11/23/24/73

MICRO 11

KEF11-B CIS DIAG
CJKDHB0

AH-S433B-MC
1 OF 2 JUL 1985
 MADE IN USA

digital

11/23/24/73

MICRO 11

KEF11-B CIS DIAG
CJKDHB0

AH-S433B-MC
2 OF 2 JUL 1985

MADE IN USA

digital

```
[ B  w
A U;                                                                    SEQ 000
1
CJKDHB0 KEF11-B CIS DIAGNOSTIC  MACRO M1200  09-APR-85 11:22  PAGE 3

                                .REM %
```

IDENTIFICATION

PRODUCT CODE: AC-S431B-MC

PRODUCT NAME: CJKDHB0 KEF11-B CIS DIAG.

PRODUCT DATE: APRIL 1985

MAINTAINER: DIAGNOSTIC ENGINEERING

```
50
51
52                                                  HISTORY
53                                                  -------
54
55
56
57                          REVISION          DATE              REASONS/CHANGES
58                          --------          --------          ---------------
59                             A              JUNE  1981        FIRST RELEASE
60                             B              APRIL 1985        MAKE UFD COMPATABLE
```

61
62
63 TABLE OF CONTENTS
64 ----------------
65
66
67

1.0 GENERAL PROGRAM INFORMATION

1.1 ABSTRACT

THIS DIAGNOSTIC VERIFIES THE FUNCTIONALITY OF THE KEF11-BA (CIS - COMMERCIAL INSTRUCTIONS SET) OPTION WITH BOTH MEMORY MANAGEMENT ENABLED AND DISABLED. IT ALLOWS THE USER TO CHECK OUT ANY COMBINATION OF CIS CHIPS. IN VIRTUALLY ALL CASES, FAULT ISOLATION IS TO THE CIS CHIP LEVEL. THE DIAGNOSTIC CAN BE RUN UNDER XXDP, ACT, AND APT MONITORS. THE SOFTWARE SWITCH REGISTER (LOCATION 176) IS USED.

THE CIS OPTION CONSISTS OF SIX MOS/LSI CONTROL CHIPS CONTAINED IN THREE 40-PIN HYBRID CARRIERS. ALL CIS INSTRUCTIONS ARE CHIP PARTITIONED, I.E. ALL THE CODE FOR A PARTICULAR INSTRUCTION IS CONTAINED ON A PARTICULAR CONTROL CHIP. THE EXCEPTION IS THAT ALL CIS INSTRUCTIONS MUST PASS THROUGH THE FIRST CIS CHIP (CONTROL CHIP DC303-004) BEFORE REACHING THEIR DESTINATION CHIP.

1.2 SYSTEM REQUIREMENTS

A. HARDWARE REQUIREMENTS

THIS DIAGNOSTIC IS DESIGNED TO RUN ON A SYSTEM IMPLEMENTING A DCF11-AA CHIP SET AND A MEMORY MANAGEMENT UNIT. ALSO REQUIRED ARE A SERIAL LINE INTERFACE, A CONSOLE TERMINAL, AND 24K OF MEMORY. THE FIRST CIS CHIP (DC303-004) MUST BE INSTALLED. ANY (OR NONE) OF THE OTHER CIS CHIPS MAY BE PRESENT.

B. SOFTWARE REQUIREMENTS

THE SOFTWARE ENVIRONMENTS UNDER WHICH THIS DIAGNOSTIC WILL OPERATE ARE APT SCRIPT, APT STAND-ALONE, ACT Q.V., ACT AUTO-ACCEPT, ACT STAND-ALONE, XXDP CHAIN, XXDP STAND-ALONE, AND ABSOLUTE BINARY PAPER TAPE.

1.3 RELATED DOCUMENTS AND STANDARDS

THE FOLLOWING DOCUMENTS WERE USED OR REFERENCED DURING THE CREATION OF THIS DIAGNOSTIC:

DIAGNOSTIC ENGINEERING STANDARDS AND CONVENTIONS PROGRAMMING PRACTICES (DOC. NO. 175-003-009-02)
ACT11/XXDP PROGRAMMING SPECIFICATION (AUTOCAT-11-QZAUB-B-D)
BIC/BIN EXTENSION LIST
STANDARD APT SYSTEM TO PDP-11 DIAGNOSTIC INTERFACE (#APT/11-317-07-09)
DEC STD 168: PDP11 EXTENDED INSTRUCTIONS
CIS4.LIS

CIS5.LIS
CIS6.LIS
CIS7.LIS
CIS8.LIS
CIS9.LIS
CONTROL CHIP SPECIFICATION
DATA CHIP SPECIFICATION
SPMAC REFERENCE MANUAL

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

IT IS ASSUMED THAT THE CPU, MEMORY MANAGEMENT, MEMORY, SERIAL LINE INTERFACE, AND CONSOLE TERMINAL HAVE ALL PASSED THE PROPER DIAGNOSTICS.

1.5 ASSUMPTIONS

THE DIAGNOSTIC WILL ATTEMPT TO TEST ALL SIX CIS CHIPS UNLESS THE SOFTWARE SWR IS ALTERED TO INDICATE A SUBSET OF CIS CHIPS TO BE TESTED. SEE SECTION 2.2.

2.0 OPERATING INSTRUCTIONS

2.1 LOADING AND STARTING PROCEDURE

A. LOADING

USE STANDARD LOADING PROCEDURE FOR PDP-11 ABSOLUTE BINARY PAPER TAPES; OR XXDP MEDIA (FILES WITH .BIC OR .BIN EXTENSIONS ONLY).

B. STARTING

THE PROGRAM SHOULD ALWAYS BE STARTED AT 200. UPON START UP (RUNNING IN NON-AUTOMATIC MODE) THE DIAGNOSTIC INITIALIZES, PRINTS OUT ITS IDENTIFICATION (FIRST TIME ONLY), AND THEN PRINTS:

"SWR = 000000 NEW = "

THE NUMBER FOLLOWING "SWR" IS THE CURRENT VALUE OF THE SOFTWARE SWITCH REGISTER. IF NO CHANGES ARE DESIRED IN THE SWITCH REGISTER THEN A CARRIAGE RETURN MAY BE HIT; IF CHANGES ARE DESIRED THEN A NEW VALUE MAY BE TYPED IN FOLLOWED BY A CARRIAGE RETURN. IF NOT ALL SIX CIS CHIPS ARE TO BE TESTED THEN THE SWR WILL HAVE TO BE CHANGED. SEE NEXT SECTION FOR SWR BIT ASSIGNMENTS.

2.2 PROGRAM OPTIONS

SWITCH REGISTER SETTINGS

LOCATION 176 IS USED FOR THE SOFTWARE SWITCH REGISTER (UNLESS RUNNING UNDER APT SCRIPT MODE). THE FOLLOWING ARE THE RESULTS OF SETTING THE RESPECTIVE SWR BITS:

BIT15	-HALT ON ERROR
BIT14	-SCOPE LOOP
BIT13	-INHIBIT ERROR TYPEOUT
BIT12	-NOT USED
BIT11	-INHIBIT SUBTEST ITERATIONS
BITS<10:8>	-NOT USED
BIT07	-MEMORY MANAGEMENT ALWAYS DISABLED
BIT06	-MEMORY MANAGEMENT ALWAYS ENABLED
BIT05	-TEST CIS CONTROL CHIP 9 (DC303-009)
BIT04	-TEST CIS CONTROL CHIP 8 (DC303-008)
BIT03	-TEST CIS CONTROL CHIP 7 (DC303-007)
BIT02	-TEST CIS CONTROL CHIP 6 (DC303-006)
BIT01	-TEST CIS CONTROL CHIP 5 (DC303-005)
BIT00	-TEST CIS CONTROL CHIP 4 (DC303-004)

AS CAN BE SEEN ABOVE, SWR BITS<5:0> ALLOW THE USER TO SELECT ANY COMBINATION OF CIS CONTROL CHIP(S) TO BE TESTED. TO SELECT A CIS CONTROL CHIP FOR TESTING, ITS CORRESPONDING BIT IN THE SWITCH REGISTER MUST BE SET. NOTE, HOWEVER, THAT IF SWR BITS<5:0> ARE ALL ZERO (DEFAULT MODE) THEN THE DIAGNOSTIC WILL TEST ALL SIX CIS

CONTROL CHIPS.

THE VALUE OF THE SOFTWARE SWITCH REGISTER MAY BE CHANGED DURING THE PROGRAM RUN. TO DO THIS A CONTROL/G IS TYPED (TYPE THE 'CTRL' AND 'G' KEYS SIMULTANEOUSLY); WITHIN A FEW SECONDS THE CURRENT VALUE OF THE SWR IS TYPED OUT AND THE PROGRAM WAITS FOR NEW INPUT.

NOTE: THE PROGRAM IS NOT RUNNING ANY TESTS WHILE WAITING FOR A NEW VALUE OF SWR TO BE INPUT.

2.3 EXECUTION TIMES

ALL TIMES WERE MEASURED ON A KDF11-A, RUNNING THE FULL COMPLEMENT OF CIS CONTROL CHIPS WITH MEMORY MANAGEMENT ENABLED AND DISABLED.

A. FIRST PASS (QV)

30 SECONDS

B. LONGEST TEST

50 SECONDS

C. ADDITIONAL TIME FOR UNITS

N/A

D. FULL PASS TIME (ITERATIONS)

90 SECONDS

3.0 ERROR INFORMATION

3.1 ERROR REPORTING PROCEDURES

WHEN A CIS INSTRUCTION FAILS, THE TYPEOUT GIVES THE ERROR PC, THE CONTROL CHIP NUMBER, AND WHAT KIND OF FAILURE OCCURRED. THE ERROR PC IS THE ACTUAL LOCATION OF THE ERROR CALL IN THE FAILING SUBTEST. THE CONTROL CHIP NUMBER REFERS TO WHICH CONTROL CHIP WAS BEING TESTED WHEN THE ERROR OCCURRED. THE CONTROL CHIP NUMBER WILL APPEAR AS 23-*04, 23-*05, 23-*06, 23-*07, 23-*08, OR 23-*09. THE * SIGNIFIES THE REVISION LEVEL OF THE CHIP AND SO MAY BE ANY DIGIT. THE LAST TWO DIGITS DISTINGUISHES ONE CIS CHIP FROM ANOTHER. CURRENTLY, THE SIX CIS CONTROL CHIPS ARE PACKAGED ON THREE HYBRID CARRIERS. THE PART NUMBERS FOR THE HYBRID ASSEMBLIES ARE: 5700002-00 (CIS CHIPS 23-*04 AND 23-*06), 5700003-00 (CIS CHIPS 23-*05 AND 23-*07) AND 5700004-00 (CIS CHIPS 23-*08 AND 23-*09). THE KINDS OF FAILURES REPORTED ARE:

"CIS INSTRUCTION FAILURE"
"CIS INTERRUPT FAILURE"
"NO TRAP ON ILLEGAL CIS CLASS OPCODE"

THE "CIS INSTRUCTION FAILURE" IS THE MOST FREQUENTLY CALLED ERROR. IT USUALLY REFERS TO A CIS INSTRUCTION THAT PRODUCES THE WRONG CONDITION CODES, THE WRONG DESTINATION OUTPUT, OR THE WRONG GENERAL PURPOSE REGISTER CONTENTS. TO DETERMINE WHAT THE SPECIFIC FAILING INSTRUCTION IS, AND THE SPECIFIC TYPE OF FAILURE, LOOK IN THE DIAGNOSTIC AT THE CODE IN THE AREA OF THE ERROR PC.

3.2 ERROR HALTS

THREE UNEXPECTED FAULTS CAN ALSO OCCUR. THEY ARE:

"AN UNEXPECTED TIME-OUT TRAP"
"AN UNEXPECTED ILLEGAL AND RESERVED INSTRUCTION TRAP"
"POWER FAIL"

IN ALL THREE CASES THE PROGRAM HALTS IMMEDIATELY AFTER TYPING OUT THE APPROPRIATE MESSAGE. HITTING "P" (PROCEED) WILL CAUSE THE DIAGNOSTIC TO START OVER IF THE FAULT WAS A POWER FAIL, AND TO CONTINUE THE DIAGNOSTIC FROM THE POINT AT WHICH IT WAS INTERRUPTED FOR THE OTHER TWO FAULTS. ANY OTHER UNEXPECTED INTERRUPTS OR TRAPS WILL BE CAUGHT BY THE TRAP CATCHER, WHICH ALSO HALTS THE DIAGNOSTIC.

SETTING SWR BIT15 WILL CAUSE THE THREE ERRORS LISTED IN SECTION 3.1 TO HALT AFTER TYPING OUT THE ERROR MESSAGE. HITTING "P" (PROCEED) WILL CONTINUE THE DIAGNOSTIC.

```
325
326
327                          4.0  PERFORMANCE AND PROGRESS REPORTS
328
329                          4.1  PERFORMANCE REPORTS
330
331                               N/A
332
333
334
335                          4.2  PROGRESS REPORTS
336
337                               AT THE END OF EACH PASS THE DIAGNOSTIC PRINTS OUT  "END  PASS  #"
338                               FOLLOWED BY THE CURRENT PASS NUMBER.
```

5.0 DEVICE INFORMATION TABLES

THE CIS INSTRUCTIONS ARE PARTITIONED SUCH THAT NEARLY ALL THE MICRO-CODE FOR AN INSTRUCTION IS LOCATED ON ONE CHIP. (THE INSTRUCTIONS THAT RESIDE ON CONTROL CHIP DC303-004 ARE CONTAINED ENTIRELY ON THAT CHIP; ALL THE OTHER INSTRUCTIONS MUST FIRST PASS THROUGH CONTROL CHIP 4 SO THERE IS A SMALL AMOUNT OF MICRO-CODE FOR THE OTHER INSTRUCTIONS ON THAT CHIP.) THE PARTITIONING IS AS FOLLOWS:

CONTROL CHIP DC303-004:		
L2DR	07602R	LOAD 2 DESCRIPTORS
L3DR	07606R	LOAD 3 DESCRIPTORS
MOVC	076030	MOVE CHARACTER
MOVCR	076031	MOVE REVERSE JUSTIFIED CHARACTER
MOVTC	076032	MOVE TRANSLATED CHARACTER
LOCC	076040	LOCATE CHARACTER
SKPC	076041	SKIP CHARACTER
SCANC	076042	SCAN CHARACTER
SPANC	076043	SPAN CHARACTER
CMPC	076044	COMPARE CHARACTER
MATC	076045	MATCH CHARACTER
CONTROL CHIP DC303-005:		
CVTNL	076053	CONVERT NUMERIC TO LONG
CVTPN	076054	CONVERT PACKED TO NUMERIC
CVTNP	076055	CONVERT NUMERIC TO PACKED
CVTLN	076057	CONVERT LONG TO NUMERIC
CVTPL	076073	CONVERT PACKED TO LONG
CVTLP	076077	CONVERT LONG TO PACKED
CONTROL CHIP DC303-006:		
ADDN	076050	ADD NUMERIC
SUBN	076051	SUBTRACT NUMERIC
CMPN	076052	COMPARE NUMERIC
ASHN	076056	ARITHMETIC SHIFT NUMERIC
CONTROL CHIP DC303-007:		
ADDP	076070	ADD PACKED
SUBP	076071	SUBTRACT PACKED
CMPP	076072	COMPARE PACKED
ASHP	076076	ARITHMETIC SHIFT PACKED
CONTROL CHIP DC303-008:		
MULP	076074	MULTIPLY PACKED
CONTROL CHIP DC303-009:		
DIVP	076075	DIVIDE PACKED

ALL OF THE ABOVE INSTRUCTIONS EXCEPT FOR L2DR AND L3DR HAVE AN IN-LINE VERSION, TOO. THE MNEMONIC IS THE SAME EXCEPT AN "I" IS APPENDED TO IT AND THE OPCODE IS THE SAME EXCEPT BIT06 IS SET. FOR EXAMPLE, THE IN-LINE VERSION OF CVTNP (076055) IS CVTNPI (076155).

```
392
393
394                          6.0  PROGRAM DESCRIPTION
395
396                               THIS PROGRAM WAS WRITTEN IN MODULES.  THE FIRST  MODULE  INCLUDES
397                               START UP AND INITIALIZATION ROUTINES, BASIC DEFINITIONS, VARIABLE
398                               DEFINITIONS, THE TRAP CATCHER, AND ACT AND APT  INFORMATION.  THE
399                               SECOND  MODULE  CONTAINS THE TESTS FOR THE CONTROL CHIP DC303-004
400                               INSTRUCTIONS, THE TEST DATA FOR THOSE  INSTRUCTIONS,  AN  ILLEGAL
401                               OPCODE  TEST,  AND  AN  INTERRUPTABILITY  TEST.  THE THIRD MODULE
402                               CONTAINS THE TESTS AND  TEST  DATA  FOR  CONTROL  CHIP  DC303-005
403                               INSTRUCTIONS,   A  SUBROUTINE  TO  CONVERT  LONG  INTEGER  TO  AN
404                               INTERMEDIATE  FORM,  AND  A  SUBROUTINE  TO  CONVERT   FROM   THE
405                               INTERMEDIATE  FORM  TO  LONG INTEGER.  THE FOURTH MODULE CONTAINS
406                               THE TESTS FOR CONTROL CHIP DC303-006 INSTRUCTIONS.  THE TESTS FOR
407                               CONTROL CHIP DC303-007 ARE IN THE FIFTH MODULE.  THE SIXTH MODULE
408                               HAS THE TESTS AND TEST  DATA  FOR  CONTROL  CHIPS  DC303-008  AND
409                               DC303-009.  FINALLY,  THE  SEVENTH  MODULE  CONTAINS END OF PASS,
410                               SCOPE, ALL THE OTHER COMMON SUBROUTINES, ASCII MESSAGES, BUFFERS,
411                               TABLES, AND TEST DATA FOR CONTROL CHIPS DC303-006 AND DC303-007.
412                               WITH BITS 6 AND 7 OF THE SWR EQUAL TO 0 THE DIAGNOSTIC WILL EXE-
413                               CUTE EVERY INSTRUCTION WITH MEMORY MANAGEMENT ENABLED AND DISABLED.
414
415                          6.1  PROGRAM EXECUTION CHARACTERISTICS
416
417                          6.2  SUBTEST SUMMARIES
418
419                               THERE IS  ONE  SUBTEST  FOR  EACH  CIS  INSTRUCTION.  FIRST,  THE
420                               REGISTER  FORM  OF THE INSTRUCTION IS TESTED WITH SEVERAL SETS OF
421                               DATA.  THEN ONE CASE OF THE IN-LINE VERSION IS TESTED.  THE  TEST
422                               DATA  FOR THE REGISTER FORM IS IN TABLES.  THE FIRST WORD OF EACH
423                               TABLE IS THE NUMBER OF TEST CASES CONTAINED  IN  THE  TABLE.  THE
424                               FORMAT  OF THE TEST DATA IS DIFFERENT FOR DIFFERENT INSTRUCTIONS;
425                               THE FIRST TEST CASE IN EACH TABLE HAS COMMENTS  IDENTIFYING  EACH
426                               ELEMENT IN THE TEST CASE.
427
428                               THE FOLLOWING IS A SUMMARY OF EACH SUBTEST.
429
430
431                                  MOVC - THE TABLE OF TEST CASES CONTAINS:
432                                              <SOURCE LENGTH>
433                                              <SOURCE ADDRESS>
434                                              <DESTINATION LENGTH>
435                                              <DESTINATION ADDRESS>
436                                              <FILL CHARACTER>
437                                              <SOURCE BUFFER STATE>
438                                              <DESTINATION BUFFER STATE>
439                                         THE SOURCE AND DESTINATION BUFFER STATES  DESIGNATE  WHAT
440                                         THOSE BUFFERS WILL BE FILLED WITH BEFORE EXECUTION OF THE
441                                         MOVC INSTRUCTION.  A '0' INDICATES EACH BYTE WILL CONTAIN
442                                         000,  A  '-1'  INDICATES  377,  AND  A  '1' INDICATES 252
443                                         (ALTERNATING  1'S  AND  0'S  BIT  PATTERN).  AN EXPECTED
444                                         DESTINATION  RESULT  IS  COMPUTED,  AS  WELL  AS EXPECTED
445                                         CONDITION CODES  AND  VALUES  OF  R0-R5.  THEN  THE  MOVC
446                                         INSTRUCTION  IS  EXECUTED  AND  THE ACTUAL RESULTS ARE
447                                         COMPARED WITH THE  EXPECTED  RESULTS.  DISAGREEMENTS  ARE
```

```
449
450
451                         FLAGGED  WITH  AN  ERROR  CALL,  WHICH  PRINTS  OUT  'CIS
452                         INSTRUCTION FAILURE', THE PC OF THE ERROR  CALL  AND  THE
453                         CONTROL  CHIP  NUMBER  (IN THIS CASE, CONTROL CHIP NUMBER
454                         DC303-004).  THIS WHOLE PROCEDURE IS  REPEATED  FOR  EACH
455                         TEST  CASE IN THE TABLE.  THEN ONE SIMPLE IN-LINE CASE IS
456                         TESTED.  THE DATA FOR THE IN-LINE CASE FOLLOWS THE  TABLE
457                         OF  TEST  CASES  FOR  THE  REGISTER  FORM OF MOVC, AND IT
458                         CONTAINS   ONLY:  SOURCE   LENGTH,   SOURCE   ADDRESS,
459                         DESTINATION LENGTH, AND DESTINATION ADDRESS.
460
461
462                 MOVRC - SAME AS MOVC
463
464
465                 MOVTC - SAME AS MOVC,
466                      EXCEPT THAT A TRANSLATION TABLE, LOCATED AT  'TRANS',  IS
467                      USED.  PRIOR  TO  THE EXECUTION OF THE MOVTC INSTRUCTION,
468                      THE TRANSLATION TABLE  IS  CLEARED.  THEN  BYTE  LOCATION
469                      TRANS+0  GETS  A  63, TRANS+252 GETS A 314, AND TRANS+377
470                      GETS A 125 (ALL NUMBERS OCTAL).
471
472
473                 LOCC - THE TABLE OF TEST CASES CONTAINS:
474                             <SOURCE LENGTH>
475                             <CHARACTER SEARCHED FOR>
476                             <OFFSET>
477                      THE SEARCH BUFFER (BUFF1) IS FILLED WITH ZEROES, THEN  AT
478                      BYTE LOCATION 'BUFF1+OFFSET' THE 'CHARACTER SEARCHED FOR'
479                      IS INSERTED.  EXPECTED RESULTS FOR  THE  CONDITION  CODES
480                      AND RO-R5 ARE COMPUTED.  THE LOCC INSTRUCTION IS EXECUTED
481                      AND EXPECTED RESULTS ARE COMPARED  WITH  ACTUAL  RESULTS.
482                      DISAGREEMENTS ARE FLAGGED WITH ERROR CALLS.
483
484
485                 SKPC - SIMILAR TO LOCC,
486                      EXCEPT THAT THE TEST  CASES  HAVE  A  'NOT  SEARCHED  FOR
487                      CHARACTER'  IN  PLACE OF A 'CHARACTER SEARCHED FOR'.  THE
488                      SEARCH  BUFFER  (BUFF1)  IS  FILLED   WITH   SEARCHED-FOR
489                      CHARACTERS, THEN AT BYTE LOCATION 'BUFF1+OFFSET' THE 'NOT
490                      SEARCHED FOR' CHARACTER IS INSERTED.
491
492
493                 SCANC - THE TABLE OF TEST CASES CONTAINS:
494                             <SOURCE LENGTH>
495                             <OFFSET>
496                      BUFF1 IS  FILLED  WITH  ZEROES,  THAN  AT  BYTE  LOCATION
497                      'BUFF1+OFFSET'  A  125 IS INSERTED.  THE TABLE 'CHRSET IS
498                      CLEARED,  THEN  A  1  IS  INSERTED   AT   BYTE   LOCATION
499                      'CHRSET+125',  TO DENOTE MEMBERSHIP IN THE CHARACTER SET.
500                      EXPECTED CONDITION CODES AND RO-R5 VALUES  ARE  COMPUTED,
501                      THE  SCANC  INSTRUCTION  IS  EXECUTED,  AND DISAGREEMENTS
502                      BETWEEN ACTUAL AND EXPECTED RESULTS CAUSE ERROR CALLS.
```

```
504
505
506                     SPANC - THE TABLE OF TEST CASES CONTAINS:
507                             <SOURCE LENGTH>
508                             <OFFSET>
509                         BUFF1 IS FILLED WITH ZEROES, THEN A 252 IS INSERTED AT
510                         BYTE LOCATION 'BUFF1+OFFSET'. CHRSET IS FILLED WITH
511                         ZEROES, THEN BYTE LOCATION 'CHRSET+O' GETS A 40 SO THAT A
512                         O IN THE SOURCE BUFFER WILL BE CONSIDERED A MEMBER OF THE
513                         CHARACTER SET AND A 252 WILL NOT. EXPECTED CONDITION
514                         CODES AND RO-R5 VALUES ARE COMPUTED, SCANC IS EXECUTED,
515                         EXPECTED RESULTS ARE COMPARED WITH ACTUAL RESULTS, AND
516                         DISAGREEMENTS CAUSE ERROR CALLS.
517
518
519                     CMPC - THE TABLE OF TEST CASES CONTAINS:
520                             <SOURCE1 LENGTH>
521                             <SOURCE2 LENGTH>
522                         BOTH SOURCE STRINGS ARE FILLED WITH 377'S AND THE FILL
523                         CHARACTER IS SET EQUAL TO 377. THEN IN THE LONGER
524                         STRING, AT 'SRC.ADR+OFFSET', A BYTE IS CLEARED. THIS
525                         BYTE WILL BE THE INEQUALITY BETWEEN THE TWO STRINGS WHICH
526                         WILL TERMINATE THE CMPC COMPARISON. ACTUAL RESULTS WILL
527                         BE COMPARED WITH PREVIOUSLY COMPUTED EXPECTED RESULTS
528                         (CONDITION CODES AND RO-R5 VALUES).
529
530
531                     MATC - THE TABLE OF TEST CASES CONTAINS:
532                             <OBJECT LENGTH>
533                             <OFFSET>
534                         THE SOURCE STRING IS THE FIRST 70 BYTES OF BUFF1. BUFF1
535                         IS FIRST FILLED WITH 377'S AND THEN CLEARED FROM
536                         'BUFF1+OFFSET' TO THE END OF BUFF1. THE OBJECT STRING
537                         STARTS AT BUFF2, IS OF LENGTH 'OBJECT LENGTH' FROM THE
538                         TABLE OF TEST CASES, AND IS ALL ZEROES. EXPECTED VALUES
539                         OF CONDITION CODES AND RO-R5 ARE COMPUTED, MATC IS
540                         EXECUTED, EXPECTED RESULTS AND ACTUAL RESULTS ARE
541                         COMPARED, AND ERROR CALLS ARE MADE FOR DISAGREEMENTS.
542
543
544                     CVTLP - THE TABLE OF TEST CASES CONTAINS:
545                             <DESTINATION DATA TYPE AND LENGTH>
546                             <LONG INTEGER HIGH, AND SIGN>
547                             <LONG INTEGER LOW>
548                         THE LONG INTEGER IN THE TEST CASE IS CONVERTED TO PACKED
549                         AND PLACED INTO BUFFER 'DS.EXP' (DECIMAL STRING EXPECTED)
550                         ACCORDING TO THE DESTINATION TYPE AND LENGTH SPECIFIED IN
551                         THE TABLE. THE EXPECTED VALUES OF THE CONDITION CODES
552                         AND RO-R5 ARE ALSO COMPUTED. THE CVTLP INSTRUCTION IS
553                         EXECUTED AND EXPECTED RESULTS ARE COMPARED WITH ACTUAL
554                         RESULTS. DISAGREEMENTS WILL CAUSE ERROR CALLS.
```

```
556
557
558                     CVTLN - SAME AS THE CVTLP TEST,
559                          EXCEPT THE LONG INTEGER IS CONVERTED TO A NUMERIC DECIMAL
560                          FORM, ACCORDING TO THE SPECIFICATION IN THE TEST CASE.
561
562
563                     CVTPL - THE TABLE OF TEST CASES CONTAINS:
564                                  <SOURCE DATA TYPE AND LENGTH>
565                                  <POINTER TO PACKED DECIMAL STRING>
566                                  <A PACKED DECIMAL STRING>
567                          THE PACKED DECIMAL STRING IS CONVERTED TO A LONG INTEGER
568                          AND THE RESULT IS PUT INTO 'R2.EXP' AND 'R3.EXP'
569                          (EXPECTED RESULTS FOR R2 AND R3). THE EXPECTED RESULTS
570                          FOR THE OTHER GENERAL PURPOSE REGISTER AND THE CONDITION
571                          CODES ARE ALSO COMPUTED. THE CVTPL INSTRUCTION IS
572                          EXECUTED, ACTUAL RESULTS ARE COMPARED WITH EXPECTED
573                          RESULTS, AND DISAGREEMENTS ARE FLAGGED BY ERROR CALLS.
574
575
576                     CVTNL - SAME AS THE CVTNL TEST,
577                          EXCEPT THAT NUMERIC DECIMAL STRINGS ARE CONVERTED INSTEAD
578                          OF PACKED DECIMAL STRINGS. ALSO, A TABLE FOR
579                          INTERPRETING ACCEPTABLE SIGN/DIGIT COMBINATIONS FOR
580                          OVERPUNCH IS INCLUDED.
581
582                     CVTPN - THE TABLE OF TEST CASES CONTAINS:
583                                  <SOURCE DATA TYPE AND LENGTH>
584                                  <POINTER TO A PACKED SOURCE STRING>
585                                  <DESTINATION LENGTH>
586                          EACH PACKED DECIMAL STRING IS CONVERTED INTO EACH OF THE
587                          SIX NUMERIC DECIMAL DATA TYPES. THUS, EACH TEST CASE IS
588                          USED SIX TIMES BEFORE GOING ON TO THE NEXT ONE. EACH
589                          TIME, THE EXPECTED CONDITION CODES AND R0-R5 VALUES ARE
590                          ALSO COMPUTED. THE CVTPN INSTRUCTION IS EXECUTED,
591                          EXPECTED RESULTS ARE COMPARED WITH ACTUAL RESULTS, AND
592                          DISAGREEMENTS ARE FLAGGED BY ERROR CALLS.
593
594                     CVTNP - THE TABLE OF TEST CASES CONTAINS:
595                                  <SOURCE DATA TYPE AND LENGTH>
596                                  <POINTER TO A SOURCE STRING>
597                                  <DESTINATION DATA TYPE AND LENGTH>
598                                  <A SOURCE STRING>
599                          THE SOURCE STRING IS IN AN INTERMEDIATE FORMAT. THE
600                          FIRST WORD INDICATES THE SIGN (0 FOR POSITIVE, -1 FOR
601                          NEGATIVE). FOLLOWING THAT IS THE MAGNITUDE, ONE
602                          BINARY-CODED-DECIMAL DIGIT PER BYTE. THE SOURCE IS
603                          CONVERTED TO A PACKED DECIMAL STRING AND PUT INTO
604                          'DS.EXP', AND THE EXPECTED VALUES OF THE CONDITION CODES
605                          AND R0-R5 ARE COMPUTED. THE CVTNP INSTRUCTION IS
606                          EXECUTED, ACTUAL RESULTS COMPARED WITH EXPECTED RESULTS,
607                          AND DISAGREEMENTS FLAGGED BY ERROR CALLS.
```

```
609
610
611                                     ADDN - THE TABLE OF TEST CASES CONTAINS:
612                                                 <SOURCE1 LENGTH>
613                                                 <SOURCE1 POINTER>
614                                                 <SOURCE2 LENGTH>
615                                                 <SOURCE2 POINTER>
616                                                 <DESTINATION LENGTH>
617                                                 <DESTINATION POINTER (SUM)>
618                                                 <EXPECTED CONDITION CODES>
619                                         THE SOURCE1, SOURCE2, AND DESTINATION STRINGS ARE  IN  AN
620                                         INTERMEDIATE  FORMAT.   EACH ONE IS TESTED IN FIVE OF THE
621                                         SIX DATA TYPE FORMATS (UNSIGNED ZONED IS  TESTED  IN  THE
622                                         INLINE  CASE  ONLY).  THIS  GIVES  A  TOTAL  OF  125 TESTS
623                                         PERFORMED ON  EACH  TEST  CASE,  IN  ORDER  TO  TEST  ALL
624                                         COMBINATIONS OF THE FIVE DATA TYPES.  SOURCE1 AND SOURCE2
625                                         ARE ADDED TOGETHER BY THE ADDN INSTRUCTION AND THE ANSWER
626                                         IS  COMPARED  TO THE EXPECTED ANSWER WHICH IS SUPPLIED IN
627                                         THE  TEST  TABLE  (POINTED  TO  BY  THE  DESTINATION
628                                         POINTER).  EXPECTED  CONDITION CODES ARE ALSO SUPPLIED IN
629                                         THE TEST CASE TABLE  AND  IS  COMPARED  WITH  THE  ACTUAL
630                                         CONDITION  CODES.  THE  EXPECTED GENERAL PURPOSE REGISTER
631                                         VALUES (R0-R5) ARE THE ONLY  EXPECTED  RESULTS  THAT  ARE
632                                         ACTUALLY  CALCULATED BY THE DIAGNOSTIC, AND THESE TOO ARE
633                                         COMPARED  WITH  THE  ACTUAL  RESULTS.  ANY  DISAGREEMENTS
634                                         BETWEEN  ACTUAL  AND  EXPECTED  RESULTS ARE FLAGGED BY AN
635                                         ERROR CALL.
636
637
638                                     SUBN - THE SAME AS ADDN,
639                                         EXCEPT THAT THE TEST CASES HAVE THE EXPECTED  DESTINATION
640                                         STRING (IN AN INTERMEDIATE FORMAT) AND EXPECTED CONDITION
641                                         CODES  FOR  A  SUBTRACT  OPERATION  INSTEAD  OF  AN   ADD
642                                         OPERATION.  THE  SOURCE1  LENGTH AND POINTER, THE SOURCE2
643                                         LENGTH  AND  POINTER,  AND  THE  DESTINATION LENGTH  ARE
644                                         IDENTICAL  TO  THOSE  USED  IN THE ADDN INSTRUCTION.  THE
645                                         TEST OPERATES IN EXACTLY THE  SAME  MANNER  AS  THE  ADDN
646                                         TEST.
647
648
649                                     CMPN - THE TABLE OF TEST CASES CONTAINS:
650                                                 <SOURCE1 LENGTH>
651                                                 <SOURCE1 POINTER>
652                                                 <SOURCE2 LENGTH>
653                                                 <SOURCE2 POINTER>
654                                         THE  TWO  SOURCE   STRINGS   ARE   IN   AN   INTERMEDIATE
655                                         FORMAT.  EACH  ONE  IS  TESTED IN FIVE OF THE SIX NUMERIC
656                                         DATA TYPE FORMATS (UNSIGNED ZONED IS TESTED IN THE INLINE
657                                         CASE  ONLY).  THIS GIVES A TOTAL OF 25 TESTS PERFORMED ON
658                                         EACH TEST CASE.  IN EACH TEST, SOURCE1 IS  ARITHMETICALLY
659                                         COMPARED  WITH  SOURCE2 AND THE CONDITIONS ARE UPDATED TO
660                                         REFLECT THE RESULTS.  THE EXPECTED  CONDITION  CODES  ARE
661                                         COMPARED  WITH  THE ACTUAL CONDITION CODES, AND R0-R5 ARE
662                                         TESTED TO SEE THAT THEY ARE CLEAR.  ANY DISCREPANCIES ARE
663                                         FLAGGED BY ERROR CALLS.
```

```
665
666
667                                     ASHN - THE TABLE OF TEST CASES CONTAINS:
668                                                <SOURCE LENGTH>
669                                                <DESTINATION LENGTH>
670                                                <SHIFT COUNT>
671                                                <ROUNDING DIGIT>
672                                          THE SOURCE STRING IS ALWAYS A STRING OF "4"'S STARTING AT
673                                          DSBUF1   AND   OF   THE  LENGTH  SPECIFIED  BY  THE  TEST
674                                          CASE.  EACH TIME, THE EXPECTED RESULTS ARE CALCULATED AND
675                                          PUT  INTO DS.EXP AND CC.EXP.  ACTUAL RESULTS ARE COMPARED
676                                          WITH EXPECTED RESULTS AND DISAGREEMENTS  ARE  FLAGGED  BY
677                                          ERROR CALLS.
678
679
680                                     ADDP - SAME AS THE ADDN TEST,
681                                          EXCEPT THAT EACH TEST CASE  IS  RUN  ONCE,  USING  SIGNED
682                                          PACKED  FORMAT.  UNSIGNED PACKED IS TESTED IN THE IN-LINE
683                                          TEST.  THE IDENTICAL TABLE OF TEST CASES USED FOR ADDN IS
684                                          USED HERE.
685
686
687                                     SUBP - SAME AS THE SUBN TEST,
688                                          EXCEPT THAT EACH TEST CASE  IS  RUN  ONCE,  USING  SIGNED
689                                          PACKED  FORMAT.  UNSIGNED PACKED IS TESTED IN THE IN-LINE
690                                          TEST.  THE IDENTICAL TABLE OF TEST CASES USED FOR SUBN IS
691                                          USED HERE.
692
693
694                                     CMPP - SAME AS THE CMPN TEST,
695                                          EXCEPT THAT EACH TEST CASE  IS  RUN  ONCE,  USING  SIGNED
696                                          PACKED  FORMAT.  UNSIGNED PACKED IS TESTED IN THE IN-LINE
697                                          TEST.  THE IDENTICAL TABLE OF TEST CASES USED FOR CMPN IS
698                                          USED HERE.
699
700
701                                     ASHP - SAME AS THE ASHN TEST,
702                                          EXCEPT THAT EACH TEST CASE  IS  RUN  ONCE,  USING  SIGNED
703                                          PACKED  FORMAT.  UNSIGNED PACKED IS TESTED IN THE IN-LINE
704                                          TEST.  THE IDENTICAL TABLE OF TEST CASES USED FOR ASHN IS
705                                          USED HERE.
```

```
707
708
709                                 MULP - THE TABLE OF TEST CASES CONTAINS:
710                                           <SOURCE1 LENGTH>
711                                           <SOURCE1 POINTER>
712                                           <SOURCE2 LENGTH>
713                                           <SOURCE2 POINTER>
714                                           <DESTINATION LENGTH>
715                                           <DESTINATION POINTER (PRODUCT)>
716                                           <CC.EXP (PRODUCT)>
717                                      THE TWO SOURCES (MULTIPLIER AND MULTIPLICAND) AND
718                                      EXPECTED DESTINATION (PRODUCT) ARE IN THE TABLE OF TEST
719                                      CASES, IN INTERMEDIATE FORMAT. THE TWO SOURCES ARE
720                                      MULTIPLIED TOGETHER (IN SIGNED PACKED FORMAT) AND THE
721                                      EXPECTED PRODUCT IS COMPARED WITH THE ACTUAL
722                                      RESULT. ALSO, THE EXPECTED VALUES OF R0-R5 AND THE
723                                      CONDITION CODES ARE COMPARED WITH THE ACTUAL
724                                      RESULTS. ANY DISCREPANCIES ARE FLAGGED BY ERROR
725                                      CALLS. UNSIGNED PACKED FORMAT IS TESTED IN THE INLINE
726                                      CASE ONLY.
727
728
729                                 DIVP - SAME AS THE MULP TEST, EXCEPT THAT A DIVP IS EXECUTED AND THE
730                                        TABLE OF TEST CASES CONTAINS QUOTIENT RESULTS FOR THE LAST TWO
731                                        TABLE ENTRIES (DESTINATION POINTER AND CC.EXP).
732
733
734                           7.0  REPAIR HISTORY
735
736                                DATE        EDIT TRAIL      PROG            REASON
737
738                                12/09/83    PA001           P. ANASTAS      TO MAKE DIAGNOSTIC COMPATIBLE
739                                                                            WITH USER FRIENDLY STRATEGY.
740
741                           ;
```

```
                        ;*COPYRIGHT (C) 1981
                        ;*DIGITAL EQUIPMENT CORP.
                        ;*MAYNARD, MASS. 01754
                        ;*
                        ;*
                        ;*THIS PROGRAM WAS ASSEMBLED USING SPMACJ
                        ;*

                        .SBTTL  BASIC DEFINITIONS

                        ;*INITIAL ADDRESS OF THE STACK POINTER *** 776 ***

000776                  STACK=  776
                        ;ERROR= EMT     ;;BASIC DEFINITION OF ERROR CALL
000004                  SCOPE   =IOT    ;;BASIC DEFINITION OF SCOPE CALL

                        ;*MISCELLANEOUS DEFINITIONS

000011                  HT=     11              ;;CODE FOR HORIZONTAL TAB
000012                  LF=     12              ;;CODE FOR LINE FEED
000015                  CR=     15              ;;CODE FOR CARRIAGE RETURN
000200                  CRLF=   200             ;;CODE FOR CARRIAGE RETURN-LINE FEED
177776                  PS=     177776          ;;PROCESSOR STATUS WORD
177776                  PSW=    PS
177774                  STKLMT= 177774          ;;STACK LIMIT REGISTER
177546                  LKS=    177546          ;;LINE CLOCK STATUS REGISTER
000100                  BUFLEN= 100             ;LENGTH OF BUFF1 AND BUFF2

                        ;*CONSOLE REGISTER DEFINITIONS

177560                  TKS=    177560
177562                  TKB=    177562
177564                  TPS=    177564
177566                  TPB=    177566

                        ;*MEMORY MANAGEMENT REGISTERS

172300                  KIPDR0= 172300
172302                  KIPDR1= 172302
172304                  KIPDR2= 172304
172306                  KIPDR3= 172306
172310                  KIPDR4= 172310
172312                  KIPDR5= 172312
172314                  KIPDR6= 172314
172316                  KIPDR7= 172316

172340                  KIPAR0= 172340
172342                  KIPAR1= 172342
172344                  KIPAR2= 172344
172346                  KIPAR3= 172346
172350                  KIPAR4= 172350
172352                  KIPAR5= 172352
172354                  KIPAR6= 172354
172356                  KIPAR7= 172356

177572                  SR1=    177572
```

```
                        ;*GENERAL PURPOSE REGISTER DEFINITIONS

000000                  R0=     %0              ;;GENERAL REGISTER
000001                  R1=     %1              ;;GENERAL REGISTER
000002                  R2=     %2              ;;GENERAL REGISTER
000003                  R3=     %3              ;;GENERAL REGISTER
000004                  R4=     %4              ;;GENERAL REGISTER
000005                  R5=     %5              ;;GENERAL REGISTER
000006                  R6=     %6              ;;GENERAL REGISTER
000007                  R7=     %7              ;;GENERAL REGISTER
000006                  SP=     %6              ;;STACK POINTER
000007                  PC=     %7              ;;PROGRAM COUNTER

                        ;*PRIORITY LEVEL DEFINITIONS

000000                  PR0=    0               ;;PRIORITY LEVEL 0
000040                  PR1=    40              ;;PRIORITY LEVEL 1
000100                  PR2=    100             ;;PRIORITY LEVEL 2
000140                  PR3=    140             ;;PRIORITY LEVEL 3
000200                  PR4=    200             ;;PRIORITY LEVEL 4
000240                  PR5=    240             ;;PRIORITY LEVEL 5
000300                  PR6=    300             ;;PRIORITY LEVEL 6
000340                  PR7=    340             ;;PRIORITY LEVEL 7

                        ;*CONDITION CODE EQUATES

000010                  NBIT=   10
000004                  ZBIT=   4
000002                  VBIT=   2
000001                  CBIT=   1

                        ;*"SWITCH REGISTER" SWITCH DEFINITIONS

100000                  SW15=   100000
040000                  SW14=   40000
020000                  SW13=   20000
010000                  SW12=   10000
004000                  SW11=   4000
002000                  SW10=   2000
001000                  SW09=   1000
000400                  SW08=   400
000200                  SW07=   200
000100                  SW06=   100
000040                  SW05=   40
000020                  SW04=   20
000010                  SW03=   10
000004                  SW02=   4
000002                  SW01=   2
000001                  SW00=   1
001000                  SW9=    SW09
000400                  SW8=    SW08
000200                  SW7=    SW07
000100                  SW6=    SW06
000040                  SW5=    SW05
000020                  SW4=    SW04
000010                  SW3=    SW03
```

BASIC DEFINITIONS

```
000004                  SW2=    SW02
000002                  SW1=    SW01
000001                  SW0=    SW00

                        ;*DATA BIT DEFINITIONS (BIT00 TO BIT15)

100000                  BIT15=  100000
040000                  BIT14=  40000
020000                  BIT13=  20000
010000                  BIT12=  10000
004000                  BIT11=  4000
002000                  BIT10=  2000
001000                  BIT09=  1000
000400                  BIT08=  400
000200                  BIT07=  200
000100                  BIT06=  100
000040                  BIT05=  40
000020                  BIT04=  20
000010                  BIT03=  10
000004                  BIT02=  4
000002                  BIT01=  2
000001                  BIT00=  1
001000                  BIT9=   BIT09
000400                  BIT8=   BIT08
000200                  BIT7=   BIT07
000100                  BIT6=   BIT06
000040                  BIT5=   BIT05
000020                  BIT4=   BIT04
000010                  BIT3=   BIT03
000004                  BIT2=   BIT02
000002                  BIT1=   BIT01
000001                  BIT0=   BIT00

                        ;*BASIC "CPU" TRAP VECTOR ADDRESSES

000004                  ERRVEC= 4               ;;TIME OUT AND OTHER ERRORS
000010                  RESVEC= 10              ;;RESERVED AND ILLEGAL INSTRUCTIONS
000014                  TBITVEC=14              ;;"T" BIT
000014                  TRTVEC= 14              ;;TRACE TRAP
000014                  BPTVEC= 14              ;;BREAKPOINT TRAP (BPT)
000020                  IOTVEC= 20              ;;INPUT/OUTPUT TRAP (IOT) **SCOPE**
000024                  PWRVEC= 24              ;;POWER FAIL
000030                  EMTVEC= 30              ;;EMULATOR TRAP (EMT)
000034                  TRAPVEC=34              ;;"TRAP" TRAP
000052                  UFDMODE=52              ;;CHECK FOR QUIET MONITOR                                   ;PA001
000001                  UFDSET=1                ;;TELL SETUP MACRO THIS IS A UFD DIAGNOSTIC                 ;PA001
000060                  TKVEC=  60              ;;TTY KEYBOARD VECTOR
000064                  TPVEC=  64              ;;TTY PRINTER VECTOR
000100                  LTCVEC= 100             ;;LTC VECTOR
000250                  MMVEC=  250             ;;MEMORY MANAGEMENT VECTOR
000000                  ASWREG  =0              ;INITIAL APT SOFTWARE SWR VALUE

                        ;*EQUATES FOR ROUTINE CALLS THAT USE TRAP INSTRUCTION

104401                  TYPE=   104401          ;TTY TYPEOUT ROUTINE
104402                  TYPOC=  104402          ;TYPE OCTAL NUMBER (WITH LEADING ZEROES) ROUTINE
```

```
                104403          TYPOS=  104403          ;TYPE OCTAL NUMBER (NO LEADING ZEROES) ROUTINE
                104404          TYPON=  104404          ;TYPE OCTAL NUMBER (AS PER LAST CALL) ROUTINE
                104405          GTSWR=  104405          ;GET SOFT-SWR SETTING ROUTINE
                104406          CKSWR=  104406          ;TEST FOR CHANGE IN SOFT-SWR ROUTINE
                100407          RDCHR=  100407          ;TTY TYPEIN CHARACTER ROUTINE
                104410          RDLIN=  104410          ;TTY TYPEIN STRING ROUTINE

                                ;;*****************************************************************
                                ;       *** COMMERCIAL INSTRUCTION SET OPCODE EQUATES ***
                                ;

                000007          MFPT    =000007                         ;MOVE FROM PROCESSOR TYPE

                                ;       COMMERCIAL LOAD DESCRIPTORS

                076020          L2D0    =76020                          ;LOAD 2 DESCRIPTORS @(R0)+
                076021          L2D1    =76021                          ;LOAD 2 DESCRIPTORS @(R1)+
                076022          L2D2    =76022                          ;LOAD 2 DESCRIPTORS @(R2)+
                076023          L2D3    =76023                          ;LOAD 2 DESCRIPTORS @(R3)+
                076024          L2D4    =76024                          ;LOAD 2 DESCRIPTORS @(R4)+
                076025          L2D5    =76025                          ;LOAD 2 DESCRIPTORS @(R5)+
                076026          L2D6    =76026                          ;LOAD 2 DESCRIPTORS @(R6)+
                076027          L2D7    =76027                          ;LOAD 2 DESCRIPTORS @(R7)+

                076060          L3D0    =76060                          ;LOAD 3 DESCRIPTORS @(R0)+
                076061          L3D1    =76061                          ;LOAD 3 DESCRIPTORS @(R1)+
                076062          L3D2    =76062                          ;LOAD 3 DESCRIPTORS @(R2)+
                076063          L3D3    =76063                          ;LOAD 3 DESCRIPTORS @(R3)+
                076064          L3D4    =76064                          ;LOAD 3 DESCRIPTORS @(R4)+
                076065          L3D5    =76065                          ;LOAD 3 DESCRIPTORS @(R5)+
                076066          L3D6    =76066                          ;LOAD 3 DESCRIPTORS @(R6)+
                076067          L3D7    =76067                          ;LOAD 3 DESCRIPTORS @(R7)+


                                ;       CHARACTER STRING INSTRUCTIONS

                076030          MOVC    =76030                          ;MOVE CHARACTER
                076031          MOVRC   =76031                          ;MOVE REVERSE CHARACTER
                076032          MOVTC   =76032                          ;MOVE TRANSLATED CHARACTER
                076040          LOCC    =76040                          ;LOCATE CHARACTER
                076041          SKPC    =76041                          ;SKIP CHARACTER
                076042          SCANC   =76042                          ;SCAN CHARACTER
                076043          SPANC   =76043                          ;SPAN CHARACTER
                076044          CMPC    =76044                          ;COMPARE CHARACTER
                076045          MATC    =76045                          ;MATCH CHARACTER

                076130          MOVCI   =76130                          ;IN-LINE VERSION OF MOVE CHARACTER
                076131          MOVRCI  =76131                          ;IN-LINE VERSION
                076132          MOVTCI  =76132                          ;       "
                076140          LOCCI   =76140                          ;       "
                076141          SKPCI   =76141                          ;       "
                076142          SCANCI  =76142                          ;       "
                076143          SPANCI  =76143                          ;       "
                076144          CMPCI   =76144                          ;       "
                076145          MATCI   =76145                          ;       "
```

```
                                ;       NUMERIC STRING INSTRUCTIONS

        076050                  ADDN    =76050                  ;ADD NUMERIC
        076051                  SUBN    =76051                  ;SUBTRACT NUMERIC
        076052                  CMPN    =76052                  ;COMPARE NUMERIC
        076053                  CVTNL   =76053                  ;CONVERT NUMERIC TO LONG
        076054                  CVTPN   =76054                  ;CONVERT PACKED TO NUMERIC
        076055                  CVTNP   =76055                  ;CONVERT NUMERIC TO PACKED
        076056                  ASHN    =76056                  ;ARITHMETIC SHIFT NUMERIC
        076057                  CVTLN   =76057                  ;CONVERT LONG TO NUMERIC

        076150                  ADDNI   =76150                  ;IN-LINE VERSION OF ADD NUMERIC
        076151                  SUBNI   =76151                  ;IN-LINE VERSION
        076152                  CMPNI   =76152                  ;       "
        076153                  CVTNLI  =76153                  ;       "
        076154                  CVTPNI  =76154                  ;       "
        076155                  CVTNPI  =76155                  ;       "
        076156                  ASHNI   =76156                  ;       "
        076157                  CVTLNI  =76157                  ;       "

                                ;       PACKED STRING INSTRUCTIONS

        076070                  ADDP    =76070                  ;ADD PACKED
        076071                  SUBP    =76071                  ;SUBTRACT PACKED
        076072                  CMPP    =76072                  ;COMPARE PACKED
        076073                  CVTPL   =76073                  ;CONVERT PACKED TO LONG
        076074                  MULP    =76074                  ;MULTIPLY PACKED
        076075                  DIVP    =76075                  ;DIVIDE PACKED
        076076                  ASHP    =76076                  ;ARITHMETIC SHIFT PACKED
        076077                  CVTLP   =76077                  ;CONVERT LONG TO PACKED

        076170                  ADDPI   =76170                  ;IN-LINE VERSION OF ADD PACKED
        076171                  SUBPI   =76171                  ;IN-LINE VERSION
        076172                  CMPPI   =76172                  ;       "
        076173                  CVTPLI  =76173                  ;       "
        076174                  MULPI   =76174                  ;       "
        076175                  DIVPI   =76175                  ;       "
        076176                  ASHPI   =76176                  ;       "
        076177                  CVTLPI  =76177                  ;       "


                                ;       CIS CLASS ILLEGAL OPCODE

        076033                  ILLOP   =76033                  ;ILLEGAL OPCODE


                                ;;**************************************************************************


                                .SBTTL  TRAP CATCHER

000000                                  .ASECT
        000000                  .       =0

                                ;*ALL UNUSED LOCATIONS FROM 4 - 776 CONTAIN A ".+2,HALT"
```

TRAP CATCHER

```
                                ;*SEQUENCE TO CATCH ILLEGAL TRAPS AND INTERRUPTS
                                ;*LOCATION 0 CONTAINS 0 TO CATCH IMPROPERLY LOADED VECTORS

                000174                  .=174

000174  000000         DISPREG: .WORD  0               ;;SOFTWARE DISPLAY REGISTER
000176  000000         SWREG:   .WORD  0               ;;SOFTWARE SWITCH REGISTER

                000100                  .=100                   ;HANDLE EVENT LINE INTERRUPT

000100  000102                  .WORD   102
000102  000002                  .WORD   2

                000200                  .=200

000200  000167  000774          JMP     START                   ;STARTING LOCATION FOR PROGRAM

                000300                  .=300

                                .SBTTL  ACT11 HOOKS

                                ;;**********************************************************************
                                ;HOOKS REQUIRED BY ACT11
                000300                  $SVPC=.                 ;SAVE PC
                000046                  .=46
000046  056232                  $ENDAD                  ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .$EOP
                000052                  .=52
000052  000000                  .WORD   0               ;;2)SET LOC.52 TO ZERO
                000300                  .=$SVPC                 ;; RESTORE PC

                                .SBTTL  APT PARAMETER BLOCK

                                ;;**********************************************************************
                                ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
                                ;;**********************************************************************
                000300                  .$X=.   ;;SAVE CURRENT LOCATION
                000024                  .=24    ;;SET POWER FAIL TO POINT TO START OF PROGRAM
000024  000200                  200     ;;FOR APT START UP
                000044                  .=44    ;;POINT TO APT INDIRECT ADDRESS PNTR.
000044  000300                  $APTHDR ;;POINT TO APT HEADER BLOCK
                000300                  .=.$X   ;;RESET LOCATION COUNTER

                                ;;**********************************************************************
                                ;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
                                ;INTERFACE SPEC.

000300                  $APTHD:
000300  000000          $HIBTS: .WORD   0       ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
000302  000632          $MBADR: .WORD   $MAIL   ;;ADDRESS OF APT MAILBOX (BITS 0-15)
000304  000300          $TSTM:  .WORD   300     ;;RUN TIM OF LONGEST TEST
000306  000300          $PASTM: .WORD   300     ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
000310  000000          $UNITM: .WORD           ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
000312  000015                  .WORD   $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)
                                .SBTTL  COMMON TAGS

                                ;;**********************************************************************
```

COMMON TAGS

```
                                ;*THIS TABLE CONTAINS VARIOUS COMMON STORAGE LOCATIONS
                                ;*USED IN THE PROGRAM.

                        .=400

000400                  $CMTAG:                         ;;START OF COMMON TAGS
000400  000000                .WORD   0
000402  000000          $TSTNM: .WORD   0               ;;CONTAINS THE TEST NUMBER
000404  000000          $TIMES: .WORD   0               ;;CONTAINS NUMBER OF ITERATIONS DESIRED
000406  000000          $ICNT:  .WORD   0               ;;CONTAINS SUBTEST ITERATION COUNT
000410  000000          $LPADR: .WORD   0               ;;CONTAINS SCOPE LOOP ADDRESS
000412  000000          $LPERR: .WORD   0               ;;CONTAINS SCOPE RETURN FOR ERRORS
000414  000000          $ERTTL: .WORD   0               ;;CONTAINS TOTAL ERRORS DETECTED
000416     001          $ERMAX: .BYTE   1               ;;CONTAINS MAX. ERRORS PER TEST
                        .EVEN
000420  000000          $ERRPC: .WORD   0               ;;CONTAINS PC OF LAST ERROR INSTRUCTION
000422     000          $AUTOB: .BYTE   0               ;;AUTOMATIC MODE INDICATOR
000423     000          $INTAG: .BYTE   0               ;;INTERRUPT MODE INDICATOR
000424  000000                .WORD   0
000426  000176          SWR:    .WORD   SWREG           ;;ADDRESS OF SWITCH REGISTER
000430  000174          DISPLAY: .WORD  DISPREG         ;;ADDRESS OF DISPLAY REGISTER
000432     000          $NULL:  .BYTE   0               ;;CONTAINS NULL CHARACTER FOR FILLS
000433     002          $FILLS: .BYTE   2               ;;CONTAINS # OF FILLER CHARACTERS REQUIRED
000434     012          $FILLC: .BYTE   12              ;;INSERT FILL CHARS. AFTER A "LINE FEED"
000435     000          $TPFLG: .BYTE   0               ;;"TERMINAL AVAILABLE" FLAG (BIT<07>=0=YES)
000436     207  377 377 $BELL:  .ASCIZ  <207><377><377> ;;CODE FOR BELL
000441     000
000442     077          $QUES:  .ASCII  /?/             ;;QUESTION MARK
000443     015          $CRLF:  .ASCII  <15>            ;;CARRIAGE RETURN
000444     012  000     $LF:    .ASCIZ  <12>            ;;LINE FEED


000446  000000          MMCYL:  .WORD   0               ;MEMORY MANAGEMENT CYCLE FLAG: 0=KTOFF, 1=KTON
000450  000000          SIGNO:  .WORD   0               ;SIGN FLAG:  0 FOR +, -1 FOR -
000452  000000          SIGN1:  .WORD   0               ;     "
000454  000000          SIGN2:  .WORD   0               ;     "
000456  000000          ERRFLG: .WORD   0               ;ERROR FLAG
000460  000000          INTFLG: .WORD   0               ;INTERRUPT FLAG
000462  000000          INTPAS: .WORD   0               ;FLAGS FIRST PASS OF INTERRUPT TEST
000464  000000          DATTYP: .WORD   0               ;DECIMAL STRING DATA TYPE FLAG
000466  000000          XXDP:   .WORD   0               ;IF NOT 0 THEN WE ARE IN XXDP CHAIN MODE
000470  000000          COUNT:  .WORD   0               ;TEMPORARY COUNTER
000472  000000          DELAY:  .WORD   0               ;TIME-DELAY COUNT STORED HERE
000474  000000          ERRTMP: .WORD   0
000476  000000          TEMP0:  .WORD   0               ;TEMPORARY VARIABLE
000500  000000          TEMP1:  .WORD   0               ;TEMPORARY VARIABLE
000502  000000          TEMP2:  .WORD   0               ;TEMPORARY VARIABLE
000504  000000          TEMP3:  .WORD   0               ;TEMPORARY VARIABLE
000506  000000          TEMP4:  .WORD   0               ;TEMPORARY VARIABLE
000510  000000          TEMP5:  .WORD   0               ;TEMPORARY VARIABLE
000512  000000          OFFSET: .WORD   0
000514  000000          DSLEN1: .WORD   0               ;FIRST DECIMAL STRING LENGTH
000516  000000          DSLEN2: .WORD   0               ;SECOND DECIMAL STRING LENGTH
000520  000000          CC:     .WORD   0               ;CONDITION CODES MAY BE SAVED HERE
000522  000000          CC.EXP: .WORD   0               ;EXPECTED CONDITION CODES
000524  000000          R0.EXP: .WORD   0               ;EXPECTED VALUE OF R0
000526  000000          R1.EXP: .WORD   0               ;                  R1
```

COMMON TAGS

```
1172 000530  000000                R2.EXP: .WORD   0               ;                   R2
1173 000532  000000                R3.EXP: .WORD   0               ;                   R3
1174 000534  000000                R4.EXP: .WORD   0               ;                   R4
1175 000536  000000                R5.EXP: .WORD   0               ;                   R5
1176 000540  000000                BF.EXP: .WORD   0               ;EXPECTED BUFFER DATA
1177 000542  000000                BFADRS: .WORD   0               ;BUFFER POINTER
1178 000544  000000                TSTCAS: .WORD   0               ;POINTER TO CURRENT TEST CASE IN TEST TABLE
1179 000546  000000                CTL:    .WORD   0               ;CONTROL CHIP NUMBER
1180 000550  000000                MSGADR: .WORD   0               ;ADDRESS OF THE ASCII TO BE PRINTED
1181 000552  000000                ERR:    .WORD   0               ;ERROR TYPE NUMBER
1182 000554     000                DSTMP0: .BYTE   0               ;DECIMAL STRING TEMPORARY
1183 000555     000                DSTMP1: .BYTE   0               ;DECIMAL STRING TEMPORARY
1184 000556     000                DSTMP2: .BYTE   0               ;DECIMAL STRING TEMPORARY
1185 000557     000                DSTMP3: .BYTE   0               ;DECIMAL STRING TEMPORARY
1186 000560     000                DSTMP4: .BYTE   0               ;DECIMAL STRING TEMPORARY
1187 000561     000                DSTMP5: .BYTE   0               ;DECIMAL STRING TEMPORARY
1188 000562  000000                DATA00: .WORD   0               ;TEST DATA GETS LOADED HERE
1189 000564  000000                DATA01: .WORD   0               ;
1190 000566  000000                DATA02: .WORD   0               ;
1191 000570  000000                DATA03: .WORD   0               ;
1192 000572  000000                DATA04: .WORD   0               ;
1193 000574  000000                DATA05: .WORD   0               ;
1194 000576  000000                DATA06: .WORD   0               ;
1195
1196 000600  000000                SAVR0:  .WORD   0               ;GENERAL PURPOSE REGISTERS MAY BE SAVED HERE
1197 000602  000000                SAVR1:  .WORD   0               ;
1198 000604  000000                SAVR2:  .WORD   0               ;
1199 000606  000000                SAVR3:  .WORD   0               ;
1200 000610  000000                SAVR4:  .WORD   0               ;
1201 000612  000000                SAVR5:  .WORD   0               ;
1202 000614  000000                SAVPSW: .WORD   0               ;PSW MAY BE SAVED HERE
1203 000616  000000                DESCR0: .WORD   0               ;TEMPS FOR DS DESCRIPTORS
1204 000620  000000                DESCR1: .WORD   0
1205 000622  000000                DESCR2: .WORD   0
1206 000624  000000                DESCR3: .WORD   0
1207 000626  000000                DESCR4: .WORD   0
1208 000630  000000                DESCR5: .WORD   0
1209
1210
1211                               ;;*******************************************************************************
1212                               .SBTTL  APT MAILBOX-ETABLE
1213
1214                               ;;*******************************************************************************
1215                               ;
1216                               ;
1217
1218                               .EVEN
1219 000632                       $MAIL:                          ;;APT MAILBOX
1220 000632  000000                $MSGTY: .WORD   000000          ;;MESSAGE TYPE CODE
1221 000634  000000                $FATAL: .WORD   000000          ;;FATAL ERROR NUMBER
1222 000636  000000                $TESTN: .WORD   000000          ;;TEST NUMBER
1223 000640  000000                $PASS:  .WORD   000000          ;;PASS COUNT
1224 000642  000000                $DEVCT: .WORD   000000          ;;DEVICE COUNT
1225 000644  000000                $UNIT:  .WORD   000000          ;;I/O UNIT NUMBER
1226 000646  000000                $MSGAD: .WORD   000000          ;;MESSAGE ADDRESS
1227 000650  000000                $MSGLG: .WORD   000000          ;;MESSAGE LENGTH
1228 000652                       $ETABLE:                        ;;APT ENVIRONMENT TABLE
```

APT MAILBOX-ETABLE

```
1229 000652     000                $ENV:   .BYTE   000     ;;ENVIRONMENT BYTE
1230 000653     000                $ENVM:  .BYTE   000     ;;ENVIRONMENT MODE BITS
1231 000654  000000                $SWREG: .WORD   000000  ;;APT SWITCH REGISTER
1232 000656  000000                $USWR:  .WORD   000000  ;;USER SWITCHES
1233 000660  000000                $CPUOP: .WORD   000000  ;;CPU TYPE,OPTIONS
1234                               ;*                      BITS 15-11=CPU TYPE
1235                               ;*                              11/04=01,11/05=02,11/20=03,11/40=04,11/45=05
1236                               ;*                              11/70=06,PDQ=07,Q=10
1237                               ;*                      BIT 10=REAL TIME CLOCK
1238                               ;*                      BIT  9=FLOATING POINT PROCESSOR
1239                               ;*                      BIT  8=MEMORY MANAGEMENT
1240 000662     000                $MAMS1: .BYTE   000     ;;HIGH ADDRESS,M.S. BYTE
1241 000663     000                $MTYP1: .BYTE   000     ;;MEM. TYPE, BLK#1
1242                               ;*                      MEM. TYPE BYTE -- (HIGH BYTE)
1243                               ;*                              900 NSEC CORE=001
1244                               ;*                              300 NSEC BIPOLAR=002
1245                               ;*                              500 NSEC MOS=003
1246 000664                        $ETEND:
1247                               .EVEN
1248
1249
1250
1251                               ;;*****************************************************************
1252                               .SBTTL  START UP AND INITIALIZATION ROUTINES
1253                               ;;*****************************************************************
1254
1255                               ;       .PSECT  CODE,REL,CON,GBL,I,RW                           -
1256         001200                .=1200
1257 001200                        START::
1258
1259                               .SBTTL  INITIALIZE THE COMMON TAGS
1260
1261                               ;;CLEAR THE COMMON TAGS ($CMTAG) AREA
1262                               .LIST   MEB
1263 001200                                .SETUP                                                  ;PA001
     001200  005767  000062       EMTSAV: TST     SAV30           ;; FIRST TIME THROUGH ?
     001204  001034                       BNE     VMKOR           ;; BRANCH IF BEEN HERE ALREADY
     001206  032737  000040 000052       BIT     #BIT5,@#52      ;; ARE WE IN UFD MODE ?
     001214  001430                       BEQ     VMKOR           ;; LEAVE IF NOT
     001216  012767  177777 000046       MOV     #-1,UFDFLG      ;; SET UFD FLAG
     001224  032737  000100 000052       BIT     #BIT6,@#52      ;; ARE WE IN QUIET MODE ?
     001232  001403                       BEQ     1$              ;; BR IF NOT
     001234  012767  177777 000032       MOV     #-1,UQUIET      ;; SET QUIET MODE
     001242  104042                1$:    EMT     42              ;; GET ADDRESS OF XXDP DCA TABLE
     001244  005060  000042              CLR     42(R0)          ;; CLR XXDP+ "DRSERR"
     001250  016767  176554 000010       MOV     30,SAV30        ;; SAVE EMULATOR ADDRESS
     001256  016767  176550 000004       MOV     32,SAV32        ;; SAVE EMULATOR PRIORITY LEVEL
     001264  000404                       BR      VMKOR           ;; GET AROUND TAG AREA
     001266  000000                SAV30:  .WORD   0               ;; PUT EMULATOR INFO HERE
     001270  000000                SAV32:  .WORD   0               ;; PUT PRIORITY LOCATION        HERE
     001272  000000                UFDFLG: .WORD   0               ;; USER FRIENDLY MODE FLAG
     001274  000000                UQUIET: .WORD   0               ;; UFD QUIET MODE FLAG
1264                               .NLIST  MEB
1265 001276  012767  177777 177770       MOV     #-1,UQUIET      ;;TO RETURN TO UFD MODE ON HALT         ;PA001
1266 001304  012706  000400              MOV     #$CMTAG,R6      ;;FIRST LOCATION TO BE CLEARED
1267 001310  005026                      CLR     (R6)+           ;;CLEAR MEMORY LOCATION
1268 001312  022706  000426              CMP     #SWR,R6         ;;DONE?
```

INITIALIZE THE COMMON TAGS

```
                001316  001374                          BNE     .-6             ;;LOOP BACK IF NO
                001320  012706  001100                  MOV     #1100,SP        ;;SETUP THE STACK POINTER

                                                ;;INITIALIZE A FEW VECTORS

                001324  012737  056354  000004          MOV     #TIMOUT,@#ERRVEC        ;TIME-OUT VECTOR
                001332  012737  000340  000006          MOV     #340,@#ERRVEC+2         ;LEVEL 7
                001340  012737  056406  000010          MOV     #ILLRES,@#RESVEC        ;ILLEGAL/RESERVED INSTR. VECTOR
                001346  012737  000340  000012          MOV     #340,@#RESVEC+2         ;LEVEL 7
                001354  012737  057376  000020          MOV     #$SCOPE,@#IOTVEC        ;;IOT VECTOR FOR SCOPE ROUTINE
                001362  012737  000340  000022          MOV     #340,@#IOTVEC+2         ;;LEVEL 7
                001370  012737  062320  000034          MOV     #$TRAP,@#TRAPVEC        ;;TRAP VECTOR FOR TRAP CALLS
                001376  012737  000340  000036          MOV     #340,@#TRAPVEC+2        ;LEVEL 7
                001404  012737  057516  000024          MOV     #$PWRDN,@#PWRVEC        ;;POWER FAILURE VECTOR
                001412  012737  000340  000026          MOV     #340,@#PWRVEC+2         ;;LEVEL 7
                001420  016767  054662  054656          MOV     $ENDCT,$EOPCT           ;;SETUP END-OF-PROGRAM COUNTER
                001426  012767  001426  176754          MOV     #.,$LPADR               ;;INITIALIZE THE LOOP ADDRESS FOR SCOPE
                001434  012737  000102  000100          MOV     #102,@#LTCVEC           ;GO TO 102
                001442  012737  000002  000102          MOV     #2,@#102                ;RTI
                001450  012737  000252  000250          MOV     #252,@#MMVEC            ;SET UP KT VECTOR
                001456  012737  177777  000446          MOV     #-1,@#MMCYL             ;INITIALIZE MEMORY MANAGEMENT CYCLE FLAG

                                                ;;SET UP SOFTWARE SWITCH REGISTER

                001464  012767  000176  176734  65$:     MOV     #SWREG,SWR              ;;POINT TO SOFTWARE SWR
                001472  012767  000174  176730          MOV     #DISPREG,DISPLAY

                001500  005067  177134                  CLR     $PASS           ;;CLEAR PASS COUNT
                001504  132767  000200  177141          BITB    #APTSIZE,$ENVM  ;;TEST USER SIZE UNDER APT
                001512  001403                          BEQ     67$             ;;YES,USE NON-APT SWITCH
                001514  012767  000654  176704          MOV     #$SWREG,SWR     ;;NO,USE APT SWITCH REGISTER
                001522                          67$:

                                                ;DETERMINE OPERATING ENVIRONMENT

                001522  032767  000040  176322          BIT     #BIT5,UFDMODE   ;ARE WE RUNNING UNDER THE UFD?          ;PA001
                001530  001403                          BEQ     6$              ;NO, NO NEED TO CLEAR DRSERR            ;PA001
                001532  104042                          EMT     42              ;YES, GET ADDRESS OF DCA INTO R0        ;PA001
                001534  005060  000042                  CLR     42(R0)          ;CLEAR DRSERR                           ;PA001
                001540  005737  000042          6$:     TST     @#42            ;ARE WE IN ACT/XXDP AUTO MODE?          ;PA001
                001544  001410                          BEQ     1$              ;BRANCH IF NO
                001546  023737  000042  000046          CMP     @#42,@#46       ;IS IT ACT AUTO MODE?
                001554  001410                          BEQ     2$              ;BRANCH IF YES
                001556  012767  000001  176702          MOV     #1,XXDP         ;SET XXDP CHAIN MODE INDICATOR          ;PA001
                001564  000404                          BR      2$              ;
                001566  126727  177060  000001  1$:     CMPB    $ENV,#1         ;ARE WE IN APT AUTO MODE?
                001574  001003                          BNE     3$              ;BRANCH IF NO
                001576  112767  000001  176616  2$:     MOVB    #1,$AUTOB       ;SET AUTO MODE INDICATOR

                                                ;PRINT TITLE IF NOT IN ACT OR APT AUTO MODE

                001604  005227  177777          3$:     INC     #-1             ;FIRST TIME?
                001610  001007                          BNE     5$              ;SKIP TITLE IF NOT
                001612  005767  176604                  TST     $AUTOB          ;ARE WE IN AUTO MODE?
                001616  001004                          BNE     5$              ;BRANCH TO TITLE TYPEOUT IF NOT
                001620  012767  062376  176722  4$:     MOV     #TITLE,MSGADR
```

```
INITIALIZE THE COMMON TAGS

   1326 001626  104401                          TYPE                    ;PRINT OUT THE TITLE
   1327
   1328                                ;GET THE VALUE IN THE SOFTWARE SWITCH REGISTER
   1329
   1330 001630  005767  176566          5$:     TST     $AUTOB          ;ARE WE IN AN AUTOMATIC MODE?
   1331 001634  001001                          BNE     BEGIN           ;BRANCH IF YES
   1332 001636  104405                          GTSWR                   ;ASK FOR SWR INPUT FROM THE CONSOLE TERMINAL
   1333
   1334 001640                          BEGIN:
   1335 001640                                  CALL MMSETUP            ;GO CYCLE MEMORY MANAGEMENT
        001640  004767  055156                                                  JSR     PC,MMSETUP
   1336
```

INITIALIZE THE COMMON TAGS

```
                        ;;*****************************************************************
                        ;;*****************************************************************
                        ;**
                        ;               C I S 4     M O D U L E
                        ;
                        ;  THIS MODULE TESTS THE LOAD DESCRIPTOR INSTRUCTIONS AND THE CHARACTER
                        ;  INSTRUCTIONS:  L2DR, L3DR, MOVC, MOVRC, MOVTC, LOCC, SKPC, SCANC,
                        ;  SPANC, CMPC, AND MATC.
                        ;--
                        ;;*****************************************************************
                        ;;*****************************************************************


                        .SBTTL  CIS4 INITIALIZATION
                        ;;*****************************************************************
                        ;
                        ;  CIS4 INITIALIZATION
                        ;
                        ;;*****************************************************************

                        ;IF SWR<5:0> ARE CLEARED OR IF SWR<0> IS SET THEN EXECUTE
                        ;  THE CIS4 MODULE; OTHERWISE, SKIP TO THE NEXT CIS MODULE.

001644                     INLINE   <SCOPE>
001644  000004                                                          SCOPE
001646                     INLINE <CKSWR>
001646  104406                                                          CKSWR

001650                  LET TEMPO := @SWR CLR.BY #177700
001650  017767  176552  176620                                         MOV     @SWR,TEMPO
001656  042767  177700  176612                                         BIC     #177700,TEMPO
001664                  IF TEMPO NE #0 THEN
001664  005767  176606                                                 TST     TEMPO
001670  001406                                                         BEQ     $50000
001672                      IF #BIT0 NOTSETIN TEMPO THEN
001672  032767  000001  176576                                         BIT     #BIT0,TEMPO
001700  001002                                                         BNE     $50001
001702                          INLINE   <JMP   CIS4END>
001702  000167  014714                                                 JMP   CIS4END
001706                      ENDIF
001706                                                          $50001:
001706                  ENDIF
001706                                                          $50000:

001706                  LET CTL := #4                ;CONTROL CHIP NUMBER
001706  012767  000004  176632                                         MOV     #4,CTL
001714                  LET ERRFLG := #0     ;INITIALIZE ERROR FLAG
001714  005067  176536                                                 CLR     ERRFLG
001720                  LET PSW := #340
001720  012767  000340  176050                                         MOV     #340,PSW
```

L2DR

```
                                .SBTTL  L2DR
                                ;;*****************************************************************
                                ;
                                ;       L2DR TEST
                                ; THIS TEST USES L2D0, L2D4, AND L2D7 TO TEST L2DR.
                                ;
                                ;;*****************************************************************
                                ROUTINE XL2DR
001726                                                                          XL2DR:
                                   INLINE  <NOP>
001726  000240                                                                      NOP
                                   INLINE  <SCOPE>
001730  000004                                                                      SCOPE
                                   LET $TIMES := #20            ;ITERATION COUNT
001732  012767  000020  176444                                                     MOV     #20,$TIMES
                                   LET COUNT := #2
001740  012767  000002  176522                                                     MOV     #2,COUNT
                                   REPEAT
001746                                                                          $50004:

                                ;+
                                ; SET UP THE GPR'S AND DO
                                ; THE CIS INSTRUCTION
                                ;-

                                      CALL CLRREG               ;CLEAR R0-R5
001746  004767  054466                                                          JSR     PC,CLRREG
                                   IF COUNT EQ #2 THEN
001752  026727  176512  000002                                                  CMP     COUNT,#2
001760  001005                                                                  BNE     $50005
                                      LET R0 := #ADR.A
001762  012700  002332                                                          MOV     #ADR.A,R0
                                      INLINE  <CCC>
001766  000257                                                                  CCC
                                      INLINE  <L2D0>
001770  076020                                                                  L2D0
                                   ELSE
001772  000404                                                                  BR      $50006
001774                                                                          $50005:
                                      LET R4 := #ADR.A
001774  012704  002332                                                          MOV     #ADR.A,R4
                                      INLINE  <SCC>
002000  000277                                                                  SCC
                                      INLINE  <L2D4>
002002  076024                                                                  L2D4
                                   ENDIF
002004                                                                          $50006:
                                   LET CC := PSW
002004  016767  175766  176506                                                  MOV     PSW,CC
                                   LET CC := CC CLR.BY #177760
002012  042767  177760  176500                                                  BIC     #177760,CC
                                   LET ERR  := #1               ;CODE FOR INSTRUCTION FAILURE
002020  012767  000001  176524                                                  MOV     #1,ERR

                                ;+
                                ;  CHECK RESULTS AND REPORT ERRORS
```

```
 1413                                   ;-
 1414
 1415 002026                            IF COUNT EQ #2 THEN
      002026  026727  176436  000002                                      CMP     COUNT,#2
      002034  001006                                                      BNE     $50007
 1416 002036                                IF CC NE #0 THEN
      002036  005767  176456                                             TST     CC
      002042  001402                                                     BEQ     $50010
 1417 002044                                    CALL ERROR
      002044  004767  055630                                             JSR     PC,ERROR
 1418 002050                                ENDIF
      002050                                                        $50010:
 1419 002050                            ELSE
      002050  000406                                                     BR      $50011
      002052                                                        $50007:
 1420 002052                                IF CC NE #17 THEN
      002052  026727  176442  000017                                      CMP     CC,#17
      002060  001402                                                     BEQ     $50012
 1421 002062                                    CALL ERROR
      002062  004767  055612                                             JSR     PC,ERROR
 1422 002066                                ENDIF
      002066                                                        $50012:
 1423 002066                            ENDIF
      002066                                                        $50011:
 1424 002066                            IF RO NE ALPHA THEN
      002066  020067  000246                                             CMP     RO,ALPHA
      002072  001402                                                     BEQ     $50013
 1425 002074                                CALL ERROR
      002074  004767  055600                                             JSR     PC,ERROR
 1426 002100                            ENDIF
      002100                                                        $50013:
 1427 002100                            IF R1 NE ALPHA+2 THEN
      002100  020167  000236                                             CMP     R1,ALPHA+2
      002104  001402                                                     BEQ     $50014
 1428 002106                                CALL ERROR
      002106  004767  055566                                             JSR     PC,ERROR
 1429 002112                            ENDIF
      002112                                                        $50014:
 1430 002112                            IF R2 NE BETA THEN
      002112  020267  000226                                             CMP     R2,BETA
      002116  001402                                                     BEQ     $50015
 1431 002120                                CALL ERROR
      002120  004767  055554                                             JSR     PC,ERROR
 1432 002124                            ENDIF
      002124                                                        $50015:
 1433 002124                            IF R3 NE BETA+2 THEN
      002124  020367  000216                                             CMP     R3,BETA+2
      002130  001402                                                     BEQ     $50016
 1434 002132                                CALL ERROR
      002132  004767  055542                                             JSR     PC,ERROR
 1435 002136                            ENDIF
      002136                                                        $50016:
 1436 002136                            IF R5 NE #0 THEN
      002136  005705                                                     TST     R5
      002140  001402                                                     BEQ     $50017
 1437 002142                                CALL ERROR
      002142  004767  055532                                             JSR     PC,ERROR
```

L2DR

```
1438 002146                                     ENDIF                                   $50017:
     002146
1439 002146                                     IF COUNT EQ #2 THEN                             CMP     COUNT,#2
     002146  026727  176316  000002                                                            BNE     $50020
     002154  001005
1440 002156                                         IF R4 NE #0 THEN                            TST     R4
     002156  005704                                                                            BEQ     $50021
     002160  001402
1441 002162                                             CALL ERROR                              JSR     PC,ERROR
     002162  004767  055512
1442 002166                                         ENDIF
     002166                                                                             $50021:
1443 002166                                     ELSE
     002166  000405                                                                            BR      $50022
     002170                                                                             $50020:
1444 002170                                         IF R4 NE #ADR.A+4 THEN                      CMP     R4,#ADR.A+4
     002170  020427  002336                                                                    BEQ     $50023
     002174  001402
1445 002176                                             CALL ERROR                              JSR     PC,ERROR
     002176  004767  055476
1446 002202                                         ENDIF
     002202                                                                             $50023:
1447 002202                                     ENDIF
     002202                                                                             $50022:
1448 002202                                     LET COUNT := COUNT - #1                         DEC     COUNT
     002202  005367  176262
1449 002206                                 UNTIL COUNT EQ #0                               TST     COUNT
     002206  005767  176256                                                                    BNE     $50004
     002212  001255
1450
1451                                    ;+
1452                                    ; TEST THE SPECIAL CASE OF USING R7
1453                                    ;-
1454
1455 002214                                     LET R4 := #0
     002214  005004                                                                         CLR     R4
1456 002216                                     LET R5 := #0
     002216  005005                                                                         CLR     R5
1457 002220                                     INLINE  <L2D7>                              L2D7
     002220  076027
1458 002222                                     INLINE  <.WORD      240>        ;NOP
     002222  000240                                                                        .WORD   240
1459 002224                                     INLINE  <.WORD      402>        ;BR TO ERROR CALL
     002224  000402                                                                        .WORD   402
1460 002226                                     INLINE  <.WORD      403>        ;BR OVER ERROR CALL
     002226  000403                                                                        .WORD   403
1461 002230                                     INLINE  <.WORD      240>        ;NOP
     002230  000240                                                                        .WORD   240
1462 002232                                     CALL ERROR                                  JSR     PC,ERROR
     002232  004767  055442
1463 002236                                     IF R0 NE @#240 THEN                         CMP     R0,@#240
     002236  020037  000240                                                                    BEQ     $50024
     002242  001402
1464 002244                                         CALL ERROR                              JSR     PC,ERROR
     002244  004767  055430
1465 002250                                     ENDIF
     002250                                                                             $50024:
```

L2DR

```
1466 002250                             IF R1 NE @#242 THEN
     002250  020137  000242                                        CMP     R1,@#242
     002254  001402                                                 BEQ     $50025
1467 002256                                 CALL ERROR
     002256  004767  055416                                         JSR     PC,ERROR
1468 002262                             ENDIF
     002262                                                     $50025:
1469 002262                             IF R2 NE @#402 THEN
     002262  020237  000402                                        CMP     R2,@#402
     002266  001402                                                 BEQ     $50026
1470 002270                                 CALL ERROR
     002270  004767  055404                                         JSR     PC,ERROR
1471 002274                             ENDIF
     002274                                                     $50026:
1472 002274                             IF R3 NE @#404 THEN
     002274  020337  000404                                        CMP     R3,@#404
     002300  001402                                                 BEQ     $50027
1473 002302                                 CALL ERROR
     002302  004767  055372                                         JSR     PC,ERROR
1474 002306                             ENDIF
     002306                                                     $50027:
1475 002306                             IF R4 NE #0 THEN
     002306  005704                                                TST     R4
     002310  001402                                                 BEQ     $50030
1476 002312                                 CALL ERROR
     002312  004767  055362                                         JSR     PC,ERROR
1477 002316                             ENDIF
     002316                                                     $50030:
1478 002316                             IF R5 NE #0 THEN
     002316  005705                                                TST     R5
     002320  001402                                                 BEQ     $50031
1479 002322                                 CALL ERROR
     002322  004767  055352                                         JSR     PC,ERROR
1480 002326                             ENDIF
     002326                                                     $50031:
1481 002326                             INLINE  <JMP XL3DR>
     002326  000167  000022                                        JMP     XL3DR
1482
1483                                    ;+
1484                                    ; TEST TABLE FOR L2DR AND L3DR
1485                                    ;-
1486
1487 002332  002340                    ADR.A:  .WORD   ALPHA
1488 002334  002344                            .WORD   BETA
1489 002336  002350                            .WORD   GAMMA
1490 002340  177777                    ALPHA:  .WORD    177777         ;ALPHA.LEN
1491 002342  052525                            .WORD    052525         ;ALPHA.ADR
1492 002344  125252                    BETA:   .WORD    125252         ;BETA.LEN
1493 002346  177777                            .WORD    177777         ;BETA.ADR
1494 002350  107070                    GAMMA:  .WORD    107070         ;GAMMA.LEN
1495 002352  070707                            .WORD    070707         ;GAMMA.ADR
1496
```

L3DR

```
1498                                    .SBTTL  L3DR
1499                                    ;;*****************************************************************
1500                                    ;
1501                                    ;       L3DR TEST
1502                                    ; THIS TEST USES L3D3, L3D5, AND L3D7 TO TEST L3DR.
1503                                    ;
1504                                    ;;*****************************************************************
1505
1506 002354                            ROUTINE XL3DR
     002354                                                                          XL3DR:
1507 002354                               INLINE  <NOP>
     002354  000240                                                                         NOP
1508 002356                               INLINE  <SCOPE>
     002356  000004                                                                         SCOPE
1509 002360                               LET $TIMES := #20          ;ITERATION COUNT
     002360  012767  000020  176016                                                         MOV     #20,$TIMES
1510 002366                               LET COUNT := #2
     002366  012767  000002  176074                                                         MOV     #2,COUNT
1511 002374                               REPEAT
     002374                                                                          $50034:
1512
1513                                    ;+
1514                                    ; SET UP THE GPR'S AND DO
1515                                    ; THE CIS INSTRUCTION
1516                                    ;-
1517
1518 002374                                   CALL CLRREG
     002374  004767  054040                                                                 JSR     PC,CLRREG
1519 002400                                   IF COUNT EQ #2 THEN
     002400  026727  176064  000002                                                         CMP     COUNT,#2
     002406  001005                                                                        BNE     $50035
1520 002410                                       LET R3 := #ADR.A
     002410  012703  002332                                                                 MOV     #ADR.A,R3
1521 002414                                       INLINE  <CCC>
     002414  000257                                                                        CCC
1522 002416                                       INLINE  <L3D3>
     002416  076063                                                                        L3D3
1523 002420                                   ELSE
     002420  000404                                                                        BR      $50036
     002422                                                                          $50035:
1524 002422                                       LET R5 := #ADR.A
     002422  012705  002332                                                                 MOV     #ADR.A,R5
1525 002426                                       INLINE  <SCC>
     002426  000277                                                                        SCC
1526 002430                                       INLINE  <L3D5>
     002430  076065                                                                        L3D5
1527 002432                                   ENDIF
     002432                                                                          $50036:
1528 002432                                   LET CC := PSW
     002432  016767  175340  176060                                                         MOV     PSW,CC
1529 002440                                   LET CC := CC CLR.BY #177760
     002440  042767  177760  176052                                                         BIC     #177760,CC
1530 002446                                   LET ERR := #1              ;CODE FOR INSTRUCTION FAILURE
     002446  012767  000001  176076                                                         MOV     #1,ERR
1531
1532                                    ;+
1533                                    ; CHECK RESULTS AND REPORT ERRORS
```

L3DR

```
1534                                    ;-
1535
1536 002454                             IF COUNT EQ #2 THEN
     002454  026727  176010  000002                                    CMP     COUNT,#2
     002462  001006                                                    BNE     $50037
1537 002464                                 IF CC NE #0 THEN
     002464  005767  176030                                            TST     CC
     002470  001402                                                    BEQ     $50040
1538 002472                                     CALL ERROR
     002472  004767  055202                                            JSR     PC,ERROR
1539 002476                                 ENDIF
     002476                                                 $50040:
1540 002476                             ELSE
     002476  000406                                                    BR      $50041
     002500                                                 $50037:
1541 002500                                 IF CC NE #17 THEN
     002500  026727  176014  000017                                    CMP     CC,#17
     002506  001402                                                    BEQ     $50042
1542 002510                                     CALL ERROR
     002510  004767  055164                                            JSR     PC,ERROR
1543 002514                                 ENDIF
     002514                                                 $50042:
1544 002514                             ENDIF
     002514                                                 $50041:
1545 002514                             IF R0 NE ALPHA THEN
     002514  020067  177620                                            CMP     R0,ALPHA
     002520  001402                                                    BEQ     $50043
1546 002522                                 CALL ERROR
     002522  004767  055152                                            JSR     PC,ERROR
1547 002526                             ENDIF
     002526                                                 $50043:
1548 002526                             IF R1 NE ALPHA+2 THEN
     002526  020167  177610                                            CMP     R1,ALPHA+2
     002532  001402                                                    BEQ     $50044
1549 002534                                 CALL ERROR
     002534  004767  055140                                            JSR     PC,ERROR
1550 002540                             ENDIF
     002540                                                 $50044:
1551 002540                             IF R2 NE BETA THEN
     002540  020267  177600                                            CMP     R2,BETA
     002544  001402                                                    BEQ     $50045
1552 002546                                 CALL ERROR
     002546  004767  055126                                            JSR     PC,ERROR
1553 002552                             ENDIF
     002552                                                 $50045:
1554 002552                             IF R3 NE BETA+2 THEN
     002552  020367  177570                                            CMP     R3,BETA+2
     002556  001402                                                    BEQ     $50046
1555 002560                                 CALL ERROR
     002560  004767  055114                                            JSR     PC,ERROR
1556 002564                             ENDIF
     002564                                                 $50046:
1557 002564                             IF R4 NE GAMMA THEN
     002564  020467  177560                                            CMP     R4,GAMMA
     002570  001402                                                    BEQ     $50047
1558 002572                                 CALL ERROR
     002572  004767  055102                                            JSR     PC,ERROR
```

L3DR

```
1559 002576                             ENDIF                                   $50047:
     002576
1560 002576                             IF R5 NE GAMMA+2 THEN                           CMP     R5,GAMMA+2
     002576  020567  177550                                                         BEQ     $50050
     002602  001402
1561 002604                                 CALL ERROR                                  JSR     PC,ERROR
     002604  004767  055070
1562 002610                             ENDIF                                   $50050:
     002610
1563 002610                             LET COUNT := COUNT - #1                         DEC     COUNT
     002610  005367  175654
1564 002614                         UNTIL COUNT EQ #0                               TST     COUNT
     002614  005767  175650                                                         BNE     $50034
     002620  001265
1565
1566                            ;+
1567                            ; TEST THE SPECIAL CASE OF USING R7
1568                            ;-
1569
1570 002622                         INLINE  <L3D7>                                  L3D7
     002622  076067
1571 002624                         INLINE      <.WORD  240>    ;NOP                .WORD   240
     002624  000240
1572 002626                         INLINE      <.WORD  240>    ;NOP                .WORD   240
     002626  000240
1573 002630                         INLINE      <.WORD  402>    ;BR TO ERROR CALL   .WORD   402
     002630  000402
1574 002632                         INLINE      <.WORD  403>    ;BR OVER ERROR CALL .WORD   403
     002632  000403
1575 002634                         INLINE      <.WORD  240>    ;NOP                .WORD   240
     002634  000240
1576 002636                         CALL ERROR                                      JSR     PC,ERROR
     002636  004767  055036
1577 002642                         IF R0 NE @#240 THEN                             CMP     R0,@#240
     002642  020037  000240                                                         BEQ     $50051
     002646  001402
1578 002650                             CALL ERROR                                  JSR     PC,ERROR
     002650  004767  055024
1579 002654                         ENDIF                                   $50051:
     002654
1580 002654                         IF R1 NE @#242 THEN                             CMP     R1,@#242
     002654  020137  000242                                                         BEQ     $50052
     002660  001402
1581 002662                             CALL ERROR                                  JSR     PC,ERROR
     002662  004767  055012
1582 002666                         ENDIF                                   $50052:
     002666
1583 002666                         IF R2 NE @#240 THEN                             CMP     R2,@#240
     002666  020237  000240                                                         BEQ     $50053
     002672  001402
1584
1585 002674                             CALL ERROR                                  JSR     PC,ERROR
     002674  004767  055000
1586 002700                         ENDIF                                   $50053:
     002700
1587 002700                         IF R3 NE @#242 THEN                             CMP     R3,@#242
     002700  020337  000242
```

```
        002704  001402                                                  BEQ     $50054
1588    002706                          CALL ERROR
        002706  004767  054766                                        JSR     PC,ERROR
1589    002712                      ENDIF
        002712                                                      $50054:
1590    002712                      IF R3 NE @#242 THEN
        002712  020337  000242                                      CMP     R3,@#242
        002716  001402                                              BEQ     $50055
1591    002720                          CALL ERROR
        002720  004767  054754                                      JSR     PC,ERROR
1592    002724                      ENDIF
        002724                                                      $50055:
1593    002724                      IF R4 NE @#402 THEN
        002724  020437  000402                                      CMP     R4,@#402
        002730  001402                                              BEQ     $50056
1594    002732                          CALL ERROR
        002732  004767  054742                                      JSR     PC,ERROR
1595    002736                      ENDIF
        002736                                                      $50056:
1596    002736                      IF R5 NE @#404 THEN
        002736  020537  000404                                      CMP     R5,@#404
        002742  001402                                              BEQ     $50057
1597    002744                          CALL ERROR
        002744  004767  054730                                      JSR     PC,ERROR
1598    002750                      ENDIF
        002750                                                      $50057:
1599    002750                      INLINE  <JMP XMOVC>
        002750  000167  000000                                      JMP     XMOVC
1600
1601
```

```
                                .SBTTL  MOVC
                                ;;*****************************************************************
                                ;
                                ;  MOVC TEST
                                ;
                                ;;*****************************************************************

002754                          ROUTINE XMOVC
002754                                                                          XMOVC:
002754                             INLINE  <NOP>
002754  000240                                                                    NOP
002756                             INLINE  <SCOPE>
002756  000004                                                                    SCOPE
002760                             LET $TIMES := #20            ;ITERATION COUNT
002760  012767  000020  175416                                                   MOV     #20,$TIMES
002766                             LET COUNT := TMOVC           ;GET TEST COUNT
002766  016767  001306  175474                                                   MOV     TMOVC,COUNT
002774                             LET TSTCAS := #TMOVC+2               ;POINT TO FIRST TEST CASE
002774  012767  004302  175542                                                   MOV     #TMOVC+2,TSTCAS
003002                             INLINE  <$5030:>
003002                                                                          $5030:

                                ;+
                                ; GET TEST DATA FROM TABLE
                                ;-

003002                             LET R0 := TSTCAS
003002  016700  175536                                                          MOV     TSTCAS,R0
003006                             LET DATA00 := (R0)+          ;SRC.LEN
003006  012067  175550                                                          MOV     (R0)+,DATA00
003012                             LET DATA01 := (R0)+          ;SRC.ADR
003012  012067  175546                                                          MOV     (R0)+,DATA01
003016                             LET DATA02 := (R0)+          ;DST.LEN
003016  012067  175544                                                          MOV     (R0)+,DATA02
003022                             LET DATA03 := (R0)+          ;DST.ADR
003022  012067  175542                                                          MOV     (R0)+,DATA03
003026                             LET DATA04 := (R0)+          ;FILL CHAR.
003026  012067  175540                                                          MOV     (R0)+,DATA04
003032                             LET DATA05 := (R0)+          ;SRC.BUFF.STATE
003032  012067  175536                                                          MOV     (R0)+,DATA05
003036                             LET DATA06 := (R0)           ;DST.BUFF.STATE
003036  011067  175534                                                          MOV     (R0),DATA06

                                ;+
                                ;  SET UP THE GPR'S
                                ;  EXPECTED GPR'S AND CONDITION CODES,
                                ;  AND BUFFERS.
                                ;-

003042                             LET R0 := DATA00
003042  016700  175514                                                          MOV     DATA00,R0
003046                             LET R1 := DATA01
003046  016701  175512                                                          MOV     DATA01,R1
003052                             LET R2 := DATA02
003052  016702  175510                                                          MOV     DATA02,R2
003056                             LET R3 := DATA03
003056  016703  175506                                                          MOV     DATA03,R3
```

```
 1641 003062                            LET R4 := DATA04
      003062  016704  175504                                                  MOV     DATA04,R4
 1642 003066                            IF #1 SETIN COUNT THEN
      003066  032767  000001  175374                                          BIT     #1,COUNT
      003074  001403                                                          BEQ     $50062
 1643 003076                                LET R5 := #177777
      003076  012705  177777                                                  MOV     #177777,R5
 1644 003102                            ELSE
      003102  000401                                                          BR      $50063
      003104                                                           $50062:
 1645 003104                                LET R5 := #0
      003104  005005                                                          CLR     R5
 1646 003106                            ENDIF
      003106                                                           $50063:
 1647 003106                            INLINE  <CMP R0,R2>
      003106  020002                                                          CMP     R0,R2
 1648 003110                            LET CC.EXP := PSW
      003110  016767  174662  175404                                          MOV     PSW,CC.EXP
 1649 003116                            LET CC.EXP := CC.EXP CLR.BY #177760
      003116  042767  177760  175376                                          BIC     #177760,CC.EXP
 1650 003124                            IF #CBIT SETIN CC.EXP OR #ZBIT SETIN CC.EXP THEN
      003124  032767  000001  175370                                          BIT     #CBIT,CC.EXP
      003132  001004                                                          BNE     $50064
      003134  032767  000004  175360                                          BIT     #ZBIT,CC.EXP
      003142  001403                                                          BEQ     $50065
      003144                                                           $50064:
 1651 003144                                LET R0.EXP := #0
      003144  005067  175354                                                  CLR     R0.EXP
 1652 003150                            ELSE
      003150  000404                                                          BR      $50066
      003152                                                           $50065:
 1653 003152                                LET R0.EXP := R0 - R2       ;SRC.LEN - DST.LEN
      003152  010067  175346                                                  MOV     R0,R0.EXP
      003156  160267  175342                                                  SUB     R2,R0.EXP
 1654 003162                            ENDIF
      003162                                                           $50066:
 1655 003162                            LET R1.EXP := #0
      003162  005067  175340                                                  CLR     R1.EXP
 1656 003166                            LET R2.EXP := #0
      003166  005067  175336                                                  CLR     R2.EXP
 1657 003172                            LET R3.EXP := #0
      003172  005067  175334                                                  CLR     R3.EXP
 1658 003176                            LET R4.EXP := R4
      003176  010467  175332                                                  MOV     R4,R4.EXP
 1659 003202                            LET R5.EXP := R5
      003202  010567  175330                                                  MOV     R5,R5.EXP
 1660 003206                            LET BFADRS := R1             ;SRC.ADR
      003206  010167  175330                                                  MOV     R1,BFADRS
 1661 003212                            IF DATA05 EQ #0 THEN
      003212  005767  175356                                                  TST     DATA05
      003216  001003                                                          BNE     $50067
 1662 003220                                CALL CLRBUF
      003220  004767  053232                                                  JSR     PC,CLRBUF
 1663 003224                            ELSE
      003224  000411                                                          BR      $50070
      003226                                                           $50067:
 1664 003226                                IF DATA05 EQ #177777 THEN
```

```
                003226  026727  175342  177777                                          CMP     DATA05,#177777
                003234  001003                                                          BNE     $50071
1665            003236                          CALL SETBUF
                003236  004767  053256                                                  JSR     PC,SETBUF
1666            003242                      ELSE
                003242  000402                                                          BR      $50072
                003244                                                          $50071:
1667            003244                          CALL ALTBUF
                003244  004767  053314                                                  JSR     PC,ALTBUF
1668            003250                      ENDIF
                003250                                                          $50072:
1669            003250                  ENDIF
                003250                                                          $50070:
1670            003250                  LET BFADRS := R3                    ;DST.ADR
                003250  010367  175266                                                  MOV     R3,BFADRS
1671            003254                  IF DATA06 EQ #0 THEN
                003254  005767  175316                                                  TST     DATA06
                003260  001003                                                          BNE     $50073
1672            003262                      CALL CLRBUF
                003262  004767  053170                                                  JSR     PC,CLRBUF
1673            003266                  ELSE
                003266  000411                                                          BR      $50074
                003270                                                          $50073:
1674            003270                      IF DATA06 EQ #177777 THEN
                003270  026727  175302  177777                                          CMP     DATA06,#177777
                003276  001003                                                          BNE     $50075
1675            003300                          CALL SETBUF
                003300  004767  053214                                                  JSR     PC,SETBUF
1676            003304                      ELSE
                003304  000402                                                          BR      $50076
                003306                                                          $50075:
1677            003306                          CALL ALTBUF
                003306  004767  053252                                                  JSR     PC,ALTBUF
1678            003312                      ENDIF
                003312                                                          $50076:
1679            003312                  ENDIF
                003312                                                          $50074:
1680            003312                  IF DATA02 EQ #0 THEN
                003312  005767  175250                                                  TST     DATA02
                003316  001003                                                          BNE     $50077
1681            003320                      LET BF.EXP := (R3)
                003320  011367  175214                                                  MOV     (R3),BF.EXP
1682            003324                  ELSE
                003324  000402                                                          BR      $50100
                003326                                                          $50077:
1683            003326                      LET BF.EXP := (R1)
                003326  011167  175206                                                  MOV     (R1),BF.EXP
1684            003332                  ENDIF
                003332                                                          $50100:
1685
1686                                    ;+
1687                                    ; DO THE CIS INSTRUCTION
1688                                    ;-
1689
1690            003332                      INLINE  <MOVC>
                003332  076030                                                          MOVC
1691
```

```
1692                                    ;*
1693                                    ; CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS
1694                                    ;-
1695
1696 003334                                LET CC := PSW
     003334  016767  174436  175156                                       MOV     PSW,CC
1697 003342                                LET CC := CC CLR.BY #177760
     003342  042767  177760  175150                                       BIC     #177760,CC
1698 003350                                IF CC NE CC.EXP THEN
     003350  026767  175144  175144                                       CMP     CC,CC.EXP
     003356  001402                                                      BEQ     $50101
1699 003360                                    CALL ERROR
     003360  004767  054314                                              JSR     PC,ERROR
1700 003364                                ENDIF
     003364                                                         $50101:
1701 003364                                IF R0 NE R0.EXP THEN
     003364  020067  175134                                              CMP     R0,R0.EXP
     003370  001402                                                      BEQ     $50102
1702 003372                                    CALL ERROR
     003372  004767  054302                                              JSR     PC,ERROR
1703 003376                                ENDIF
     003376                                                         $50102:
1704 003376                                IF R1 NE R1.EXP THEN
     003376  020167  175124                                              CMP     R1,R1.EXP
     003402  001402                                                      BEQ     $50103
1705 003404                                    CALL ERROR
     003404  004767  054270                                              JSR     PC,ERROR
1706 003410                                ENDIF
     003410                                                         $50103:
1707 003410                                IF R2 NE R2.EXP THEN
     003410  020267  175114                                              CMP     R2,R2.EXP
     003414  001402                                                      BEQ     $50104
1708 003416                                    CALL ERROR
     003416  004767  054256                                              JSR     PC,ERROR
1709 003422                                ENDIF
     003422                                                         $50104:
1710 003422                                IF R3 NE R3.EXP THEN
     003422  020367  175104                                              CMP     R3,R3.EXP
     003426  001402                                                      BEQ     $50105
1711 003430                                    CALL ERROR
     003430  004767  054244                                              JSR     PC,ERROR
1712 003434                                ENDIF
     003434                                                         $50105:
1713 003434                                IF R4 NE R4.EXP THEN
     003434  020467  175074                                              CMP     R4,R4.EXP
     003440  001402                                                      BEQ     $50106
1714 003442                                    CALL ERROR
     003442  004767  054232                                              JSR     PC,ERROR
1715 003446                                ENDIF
     003446                                                         $50106:
1716 003446                                IF R5 NE R5.EXP THEN
     003446  020567  175064                                              CMP     R5,R5.EXP
     003452  001402                                                      BEQ     $50107
1717 003454                                    CALL ERROR
     003454  004767  054220                                              JSR     PC,ERROR
1718 003460                                ENDIF
     003460                                                         $50107:
```

```
                                ;+
                                ; CHECK ENTIRE DESTINATION BUFFER FOR CORRECT RESULTS
                                ;-

003460                             LET R0 := DATA01             ;SRC.ADR
003460  016700  175100                                                    MOV     DATA01,R0
003464                             LET R1 := DATA03             ;DST.ADR
003464  016701  175100                                                    MOV     DATA03,R1
003470                             LET ERRFLG := #0
003470  005067  174762                                                    CLR     ERRFLG
003474                             IF DATA00 HIS DATA02 THEN
003474  026767  175062  175064                                           CMP     DATA00,DATA02
003502  103417                                                            BLO     $50110
003504                                 LET R3 := DATA02
003504  016703  175056                                                    MOV     DATA02,R3
003510                                 WHILE R3 NE #0 AND ERRFLG EQ #0 DO
003510                                                                $50111:
003510  005703                                                            TST     R3
003512  001412                                                            BEQ     $50112
003514  005767  174736                                                    TST     ERRFLG
003520  001007                                                            BNE     $50112
003522                                     IFB BF.EXP NE (R1)+ THEN
003522  126721  175012                                                    CMPB    BF.EXP,(R1)+
003526  001402                                                            BEQ     $50113
003530                                         CALL ERROR
003530  004767  054144                                                    JSR     PC,ERROR
003534                                     ENDIF
003534                                                                $50113:
003534                                     LET R3 := R3 - #1
003534  005303                                                            DEC     R3
003536                                 ENDDO
003536  000764                                                            BR      $50111
003540                                                                $50112:
003540                             ELSE
003540  000441                                                            BR      $50114
003542                                                                $50110:
003542                                 LET R3 := DATA00                 ;SRC.LEN
003542  016703  175014                                                    MOV     DATA00,R3
003546                                 WHILE R3 NE #0 AND ERRFLG EQ #0 DO
003546                                                                $50115:
003546  005703                                                            TST     R3
003550  001412                                                            BEQ     $50116
003552  005767  174700                                                    TST     ERRFLG
003556  001007                                                            BNE     $50116
003560                                     IFB BF.EXP NE (R1)+ THEN
003560  126721  174754                                                    CMPB    BF.EXP,(R1)+
003564  001402                                                            BEQ     $50117
003566                                         CALL ERROR
003566  004767  054106                                                    JSR     PC,ERROR
003572                                     ENDIF
003572                                                                $50117:
003572                                     LET R3 := R3 - #1
003572  005303                                                            DEC     R3
003574                                 ENDDO
003574  000764                                                            BR      $50115
003576                                                                $50116:
```

```
1743 003576                                LET R3 := DATA02  -  DATA00
     003576  016703  174764                                                MOV     DATA02,R3
     003602  166703  174754                                                SUB     DATA00,R3
1744 003606                                LET R1 := DATA03  +  DATA00
     003606  016701  174756                                                MOV     DATA03,R1
     003612  066701  174744                                                ADD     DATA00,R1
1745 003616                                REPEAT
     003616                                                          $50120:
1746 003616                                    IFB (R1)+ NE DATA04 THEN
     003616  122167  174750                                                CMPB    (R1)+,DATA04
     003622  001402                                                        BEQ     $50121
1747 003624                                        CALL ERROR
     003624  004767  054050                                                JSR     PC,ERROR
1748 003630                                    ENDIF
     003630                                                          $50121:
1749 003630                                    LET R3 := R3 - #1
     003630  005303                                                        DEC     R3
1750 003632                                UNTIL R3 EQ #0 OR ERRFLG NE #0
     003632  005703                                                        TST     R3
     003634  001403                                                        BEQ     $50122
     003636  005767  174614                                                TST     ERRFLG
     003642  001765                                                        BEQ     $50120
     003644                                                          $50122:
1751 003644                            ENDIF
     003644                                                          $50114:
1752 003644                            LET ERRFLG := #0
     003644  005067  174606                                                CLR     ERRFLG
1753 003650                            IF DATA02 LO #BUFLEN THEN
     003650  026727  174712  000100                                        CMP     DATA02,#BUFLEN
     003656  103040                                                        BHIS    $50123
1754 003660                                IF DATA06 EQ #1 THEN
     003660  026727  174712  000001                                        CMP     DATA06,#1
     003666  001004                                                        BNE     $50124
1755 003670                                    LET BF.EXP := #125252
     003670  012767  125252  174642                                        MOV     #125252,BF.EXP
1756 003676                                ELSE
     003676  000403                                                        BR      $50125
     003700                                                          $50124:
1757 003700                                    LET BF.EXP := DATA06
     003700  016767  174672  174632                                        MOV     DATA06,BF.EXP
1758 003706                                ENDIF
     003706                                                          $50125:
1759 003706                            LET R0 := DATA02 + DATA03
     003706  016700  174654                                                MOV     DATA02,R0
     003712  066700  174652                                                ADD     DATA03,R0
1760 003716                            LET TEMP0 := DATA03 + #BUFLEN
     003716  016767  174646  174552                                        MOV     DATA03,TEMP0
     003724  062767  000100  174544                                        ADD     #BUFLEN,TEMP0
1761 003732                            REPEAT
     003732                                                          $50126:
1762 003732                                IFB (R0)+ NE BF.EXP THEN
     003732  122067  174602                                                CMPB    (R0)+,BF.EXP
     003736  001402                                                        BEQ     $50127
1763 003740                                    CALL ERROR
     003740  004767  053734                                                JSR     PC,ERROR
1764 003744                                ENDIF
     003744                                                          $50127:
```

```
MOVC

1765 003744                                 UNTIL R0 EQ TEMPO OR ERRFLG NE #0
     003744  020067  174526                                                    CMP     R0,TEMPO
     003750  001403                                                            BEQ     $50130
     003752  005767  174500                                                    TST     ERRFLG
     003756  001765                                                            BEQ     $50126
     003760                                                             $50130:
1766 003760                                 ENDIF
     003760                                                             $50123:
1767 003760                                 LET TSTCAS := TSTCAS + #16
     003760  062767  000016  174556                                            ADD     #16,TSTCAS
1768 003766                                 LET COUNT := COUNT - #1
     003766  005367  174476                                                    DEC     COUNT
1769                                       ;UNTIL COUNT EQ #0
1770 003772                                 IF COUNT NE #0 THEN
     003772  005767  174472                                                    TST     COUNT
     003776  001402                                                            BEQ     $50131
1771 004000                                     INLINE  <JMP     $5030>
     004000  000167  176776                                                    JMP     $5030
1772 004004                                 ENDIF
     004004                                                             $50131:
1773
1774
1775                                       ;+
1776                                       ; DO ONE IN-LINE CASE - MOVCI
1777                                       ;-
1778
1779 004004                                 LET BFADRS := #BUFF1
     004004  012767  063236  174530                                            MOV     #BUFF1,BFADRS
1780 004012                                 CALL SETBUF
     004012  004767  052502                                                    JSR     PC,SETBUF
1781 004016                                 LET BFADRS := #BUFF2
     004016  012767  063336  174516                                            MOV     #BUFF2,BFADRS
1782 004024                                 CALL CLRBUF
     004024  004767  052426                                                    JSR     PC,CLRBUF
1783 004030                                 LET R0 := #125252
     004030  012700  125252                                                    MOV     #125252,R0
1784 004034                                 LET R1 := #125252
     004034  012701  125252                                                    MOV     #125252,R1
1785 004040                                 LET R2 := #125252
     004040  012702  125252                                                    MOV     #125252,R2
1786 004044                                 LET R3 := #125252
     004044  012703  125252                                                    MOV     #125252,R3
1787 004050                                 LET R4 := #125252
     004050  012704  125252                                                    MOV     #125252,R4
1788 004054                                 LET R5 := #125252
     004054  012705  125252                                                    MOV     #125252,R5
1789 004060                                 LET ERRFLG := #0
     004060  005067  174372                                                    CLR     ERRFLG
1790 004064                                 INLINE  <MOVCI>
     004064  076130                                                            MOVCI
1791 004066                                 INLINE  <.WORD       TMOVCI>                ;SRC.DSCR.PTR
     004066  004624                                                                    .WORD   TMOVCI
1792 004070                                 INLINE       <.WORD  TMOVCI+4>                 ;DST.DSCR.PTR
     004070  004630                                                                    .WORD   TMOVCI+4
1793 004072                                 INLINE  <.WORD       40>                    ;FILL CHARACTER
     004072  000040                                                                    .WORD   40
1794 004074                                 IFCOND CS THEN
```

```
      004074  103003                                                    BCC     $50132
 1795 004076                          IFCOND EQ THEN
      004076  001002                                                    BNE     $50133
 1796 004100                              CALL ERROR
      004100  004767  053574                                            JSR     PC,ERROR
 1797 004104                          ENDIF
      004104                                                    $50133:
 1798 004104                      ENDIF
      004104                                                    $50132:
 1799 004104                      IF R0 NE #125252 THEN
      004104  020027  125252                                            CMP     R0,#125252
      004110  001402                                                    BEQ     $50134
 1800 004112                          CALL ERROR
      004112  004767  053562                                            JSR     PC,ERROR
 1801 004116                      ENDIF
      004116                                                    $50134:
 1802 004116                      IF R1 NE #125252 THEN
      004116  020127  125252                                            CMP     R1,#125252
      004122  001402                                                    BEQ     $50135
 1803 004124                          CALL ERROR
      004124  004767  053550                                            JSR     PC,ERROR
 1804 004130                      ENDIF
      004130                                                    $50135:
 1805 004130                      IF R2 NE #125252 THEN
      004130  020227  125252                                            CMP     R2,#125252
      004134  001402                                                    BEQ     $50136
 1806 004136                          CALL ERROR
      004136  004767  053536                                            JSR     PC,ERROR
 1807 004142                      ENDIF
      004142                                                    $50136:
 1808 004142                      IF R3 NE #125252 THEN
      004142  020327  125252                                            CMP     R3,#125252
      004146  001402                                                    BEQ     $50137
 1809 004150                          CALL ERROR
      004150  004767  053524                                            JSR     PC,ERROR
 1810 004154                      ENDIF
      004154                                                    $50137:
 1811 004154                      IF R4 NE #125252 THEN
      004154  020427  125252                                            CMP     R4,#125252
      004160  001402                                                    BEQ     $50140
 1812 004162                          CALL ERROR
      004162  004767  053512                                            JSR     PC,ERROR
 1813 004166                      ENDIF
      004166                                                    $50140:
 1814 004166                      IF R5 NE #125252 THEN
      004166  020527  125252                                            CMP     R5,#125252
      004172  001402                                                    BEQ     $50141
 1815 004174                          CALL ERROR
      004174  004767  053500                                            JSR     PC,ERROR
 1816 004200                      ENDIF
      004200                                                    $50141:
 1817 004200                      IF BUFF2 NE #177777 THEN
      004200  026727  057132  177777                                    CMP     BUFF2,#177777
      004206  001402                                                    BEQ     $50142
 1818 004210                          CALL ERROR
      004210  004767  053464                                            JSR     PC,ERROR
 1819 004214                      ENDIF
```

```
004214                                                                  $50142:
004214                              IF BUFF2+2 NE #20040 THEN
004214  026727  057120  020040                                                  CMP     BUFF2+2,#20040
004222  001402                                                                  BEQ     $50143
004224                                  CALL ERROR
004224  004767  053450                                                          JSR     PC,ERROR
004230                              ENDIF
004230                                                                  $50143:
004230                              LET R0 := #BUFF2+4
004230  012700  063342                                                          MOV     #BUFF2+4,R0
004234                              LET TEMPO := #BUFF2 + #BUFLEN-2
004234  012767  063336  174234                                                  MOV     #BUFF2,TEMPO
004242  062767  000076  174226                                                  ADD     #BUFLEN-2,TEMPO
004250                              REPEAT
004250                                                                  $50144:
004250                                  IF (R0)+ NE #0 THEN
004250  005720                                                                  TST     (R0)+
004252  001402                                                                  BEQ     $50145
004254                                      CALL ERROR
004254  004767  053420                                                          JSR     PC,ERROR
004260                                  ENDIF
004260                                                                  $50145:
004260                              UNTIL R0 EQ TEMPO OR ERRFLG NE #0
004260  020067  174212                                                          CMP     R0,TEMPO
004264  001403                                                                  BEQ     $50146
004266  005767  174164                                                          TST     ERRFLG
004272  001766                                                                  BEQ     $50144
004274                                                                  $50146:
004274                              INLINE  <JMP XMOVRC>
004274  000167  000334                                                          JMP     XMOVRC

                            ;+
                            ; TABLE OF TEST CASES FOR MOVC
                            ;-

004300  000017              TMOVC:  .WORD  15.              ;# OF TEST CASES FOR MOVC

004302  000040                      .WORD   40                      ;SRC.LEN
004304  063336                      .WORD  BUFF2                    ;SRC.ADR
004306  000000                      .WORD  0                        ;DST.LEN
004310  063236                      .WORD  BUFF1                    ;DST.ADR
004312  000040                      .WORD  40                       ;FILL CHAR.
004314  000000                      .WORD   0                       ;SRC.BUFF.STATE
004316  177777                      .WORD  -1                       ;DST.BUFF.STATE

004320  000050                      .WORD  50               ;TEST CASE NUMBER 2
004322  063336                      .WORD  BUFF2
004324  000020                      .WORD  20
004326  063236                      .WORD  BUFF1
004330  000100                      .WORD  100
004332  177777                      .WORD  -1
004334  000000                      .WORD   0

004336  000045                      .WORD  45               ;TEST CASE NUMBER 3
004340  063336                      .WORD  BUFF2
004342  000030                      .WORD  30
004344  063236                      .WORD  BUFF1
```

```
1858 004346  000007                    .WORD   7
1859 004350  000000                    .WORD   0
1860 004352  000001                    .WORD   1
1861
1862 004354  000000                    .WORD   0               ;TEST CASE NUMBER 4
1863 004356  063336                    .WORD  BUFF2
1864 004360  000060                    .WORD  60
1865 004362  063236                    .WORD  BUFF1
1866 004364  000010                    .WORD  10
1867 004366  000001                    .WORD   1
1868 004370  000000                    .WORD   0
1869
1870 004372  000046                    .WORD  46               ;TEST CASE NUMBER 5
1871 004374  063336                    .WORD  BUFF2
1872 004376  000132                    .WORD  BUFLEN+32
1873 004400  063236                    .WORD  BUFF1
1874 004402  000077                    .WORD  77
1875 004404  177777                    .WORD  -1
1876 004406  000001                    .WORD   1
1877
1878 004410  000000                    .WORD   0               ;TEST CASE NUMBER 6
1879 004412  063336                    .WORD  BUFF2
1880 004414  000015                    .WORD  15
1881 004416  063236                    .WORD  BUFF1
1882 004420  000060                    .WORD  60
1883 004422  000001                    .WORD   1
1884 004424  177777                    .WORD  -1
1885
1886 004426  000043                    .WORD  43               ;TEST CASE NUMBER 7
1887 004430  063336                    .WORD  BUFF2
1888 004432  000070                    .WORD  70
1889 004434  063236                    .WORD  BUFF1
1890 004436  000370                    .WORD  370
1891 004440  000000                    .WORD   0
1892 004442  177777                    .WORD  -1
1893
1894 004444  000060                    .WORD  60               ;TEST CASE NUMBER 8
1895 004446  063236                    .WORD  BUFF1
1896 004450  000000                    .WORD   0
1897 004452  063336                    .WORD  BUFF2
1898 004454  000200                    .WORD  200
1899 004456  177777                    .WORD  -1
1900 004460  000000                    .WORD   0
1901
1902 004462  000120                    .WORD  BUFLEN+20        ;TEST CASE NUMBER 9
1903 004464  063236                    .WORD  BUFF1
1904 004466  000100                    .WORD  100
1905 004470  063336                    .WORD  BUFF2
1906 004472  000002                    .WORD   2
1907 004474  000000                    .WORD   0
1908 004476  000001                    .WORD   1
1909
1910 004500  000015                    .WORD  15               ;TEST CASE NUMBER 10
1911 004502  063236                    .WORD  BUFF1
1912 004504  000000                    .WORD   0
1913 004506  063336                    .WORD  BUFF2
1914 004510  000016                    .WORD  16
```

MOVC

```
1915 004512  000001                        .WORD   1
1916 004514  000000                        .WORD   0
1917
1918 004516  000077                        .WORD  77               ;TEST CASE NUMBER 11
1919 004520  063236                        .WORD  BUFF1
1920 004522  000021                        .WORD  21
1921 004524  063336                        .WORD  BUFF2
1922 004526  000300                        .WORD  300
1923 004530  177777                        .WORD  -1
1924 004532  000001                        .WORD   1
1925
1926 004534  000000                        .WORD   0               ;TEST CASE NUMBER 12
1927 004536  063236                        .WORD  BUFF1
1928 004540  000050                        .WORD  50
1929 004542  063336                        .WORD  BUFF2
1930 004544  000003                        .WORD   3
1931 004546  000001                        .WORD   1
1932 004550  177777                        .WORD  -1
1933
1934 004552  000014                        .WORD  14               ;TEST CASE NUMBER 13
1935 004554  063236                        .WORD  BUFF1
1936 004556  000042                        .WORD  42
1937 004560  063336                        .WORD  BUFF2
1938 004562  000030                        .WORD  30
1939 004564  000000                        .WORD   0
1940 004566  177777                        .WORD  -1
1941
1942 004570  000000                        .WORD   0               ;TEST CASE NUMBER 14
1943 004572  063236                        .WORD  BUFF1
1944 004574  000053                        .WORD  53
1945 004576  063336                        .WORD  BUFF2
1946 004600  000012                        .WORD  12
1947 004602  000001                        .WORD   1
1948 004604  000000                        .WORD   0
1949
1950 004606  000027                        .WORD  27               ;TEST CASE NUMBER 15
1951 004610  063236                        .WORD  BUFF1
1952 004612  000060                        .WORD  60
1953 004614  063336                        .WORD  BUFF2
1954 004616  000014                        .WORD  14
1955 004620  177777                        .WORD  -1
1956 004622  000001                        .WORD   1
1957
1958
1959 004624  000002              TMOVCI:   .WORD   2               ;TEST CASE FOR IN-LINE
1960 004626  063236                        .WORD  BUFF1
1961 004630  000004                        .WORD   4
1962 004632  063336                        .WORD  BUFF2
1963
```

```
                                .SBTTL  MOVRC
                                ;;******************************************************************
                                ;
                                ;  MOVRC TEST
                                ;
                                ;;******************************************************************

                                ROUTINE XMOVRC
004634                                                                          XMOVRC:
                                   INLINE  <NOP>
004634  000240                                                                 NOP
                                   INLINE  <SCOPE>
004636  000004                                                                 SCOPE
                                   LET $TIMES := #20            ;ITERATION COUNT
004640  012767  000020  173536                                                 MOV     #20,$TIMES
                                   LET COUNT := TMOVRC          ;GET TEST COUNT
004646  016767  001274  173614                                                 MOV     TMOVRC,COUNT
                                   LET TSTCAS := #TMOVRC+2              ;POINT TO FIRST TEST CASE
004654  012767  006150  173662                                                 MOV     #TMOVRC+2,TSTCAS
                                   INLINE  <$5031:>
004662                                                                          $5031:

                                ;+
                                ; GET TEST DATA FROM TABLE
                                ;-

                                   LET R0 := TSTCAS
004662  016700  173656                                                         MOV     TSTCAS,R0
                                   LET DATA00 := (R0)+          ;SRC.LEN
004666  012067  173670                                                         MOV     (R0)+,DATA00
                                   LET DATA01 := (R0)+          ;SRC.ADR
004672  012067  173666                                                         MOV     (R0)+,DATA01
                                   LET DATA02 := (R0)+          ;DST.LEN
004676  012067  173664                                                         MOV     (R0)+,DATA02
                                   LET DATA03 := (R0)+          ;DST.ADR
004702  012067  173662                                                         MOV     (R0)+,DATA03
                                   LET DATA04 := (R0)+          ;FILL CHAR.
004706  012067  173660                                                         MOV     (R0)+,DATA04
                                   LET DATA05 := (R0)+          ;SRC.BUFF.STATE
004712  012067  173656                                                         MOV     (R0)+,DATA05
                                   LET DATA06 := (R0)           ;DST.BUFF.STATE
004716  011067  173654                                                         MOV     (R0),DATA06

                                ;+
                                ; SET UP GPR'S, EXPECTED GPR'S AND
                                ; CONDITION CODES, AND BUFFERS.
                                ;-

                                   LET R0 := DATA00
004722  016700  173634                                                         MOV     DATA00,R0
                                   LET R1 := DATA01
004726  016701  173632                                                         MOV     DATA01,R1
                                   LET R2 := DATA02
004732  016702  173630                                                         MOV     DATA02,R2
                                   LET R3 := DATA03
004736  016703  173626                                                         MOV     DATA03,R3
                                   LET R4 := DATA04
```

```
      004742  016704  173624                                                    MOV     DATA04,R4
 2003 004746                          IF #1 SETIN COUNT THEN
      004746  032767  000001  173514                                            BIT     #1,COUNT
      004754  001403                                                            BEQ     $50151
 2004 004756                              LET R5 := #177777
      004756  012705  177777                                                    MOV     #177777,R5
 2005 004762                          ELSE
      004762  000401                                                            BR      $50152
      004764                                                            $50151:
 2006 004764                              LET R5 := #0
      004764  005005                                                            CLR     R5
 2007 004766                          ENDIF
      004766                                                            $50152:
 2008 004766                          INLINE  <CMP R0,R2>
      004766  020002                                                            CMP     R0,R2
 2009 004770                          LET CC.EXP := PSW
      004770  016767  173002  173524                                            MOV     PSW,CC.EXP
 2010 004776                          LET CC.EXP := CC.EXP  CLR.BY  #177760
      004776  042767  177760  173516                                            BIC     #177760,CC.EXP
 2011 005004                          IF #CBIT SETIN CC.EXP OR #ZBIT SETIN CC.EXP THEN
      005004  032767  000001  173510                                            BIT     #CBIT,CC.EXP
      005012  001004                                                            BNE     $50153
      005014  032767  000004  173500                                            BIT     #ZBIT,CC.EXP
      005022  001403                                                            BEQ     $50154
      005024                                                            $50153:
 2012 005024                              LET R0.EXP := #0
      005024  005067  173474                                                    CLR     R0.EXP
 2013 005030                          ELSE
      005030  000404                                                            BR      $50155
      005032                                                            $50154:
 2014 005032                              LET R0.EXP := R0 - R2
      005032  010067  173466                                                    MOV     R0,R0.EXP
      005036  160267  173462                                                    SUB     R2,R0.EXP
 2015 005042                          ENDIF
      005042                                                            $50155:
 2016 005042                          LET R1.EXP := #0
      005042  005067  173460                                                    CLR     R1.EXP
 2017 005046                          LET R2.EXP := #0
      005046  005067  173456                                                    CLR     R2.EXP
 2018 005052                          LET R3.EXP := #0
      005052  005067  173454                                                    CLR     R3.EXP
 2019 005056                          LET R4.EXP := R4
      005056  010467  173452                                                    MOV     R4,R4.EXP
 2020 005062                          LET R5.EXP := R5
      005062  010567  173450                                                    MOV     R5,R5.EXP
 2021 005066                          LET BFADRS := DATA01
      005066  016767  173472  173446                                            MOV     DATA01,BFADRS
 2022 005074                          IF DATA05 EQ #0 THEN
      005074  005767  173474                                                    TST     DATA05
      005100  001003                                                            BNE     $50156
 2023 005102                              CALL CLRBUF
      005102  004767  051350                                                    JSR     PC,CLRBUF
 2024 005106                          ELSE
      005106  000411                                                            BR      $50157
      005110                                                            $50156:
 2025 005110                              IF DATA05 EQ #177777 THEN
      005110  026727  173460  177777                                            CMP     DATA05,#177777
```

```
005116  001003                                                                  BNE     $50160
005120                                  CALL SETBUF
005120  004767  051374                                                          JSR     PC,SETBUF
005124                              ELSE
005124  000402                                                                  BR      $50161
005126                                                              $50160:
005126                                  CALL ALTBUF
005126  004767  051432                                                          JSR     PC,ALTBUF
005132                              ENDIF
005132                                                              $50161:
005132                          ENDIF
005132                                                              $50157:
005132                          LET BFADRS := DATA03
005132  016767  173432  173402                                                  MOV     DATA03,BFADRS
005140                          IF DATA06 EQ #0 THEN
005140  005767  173432                                                          TST     DATA06
005144  001003                                                                  BNE     $50162
005146                              CALL CLRBUF
005146  004767  051304                                                          JSR     PC,CLRBUF
005152                          ELSE
005152  000411                                                                  BR      $50163
005154                                                              $50162:
005154                              IF DATA06 EQ #177777 THEN
005154  026727  173416  177777                                                  CMP     DATA06,#177777
005162  001003                                                                  BNE     $50164
005164                                  CALL SETBUF
005164  004767  051330                                                          JSR     PC,SETBUF
005170                              ELSE
005170  000402                                                                  BR      $50165
005172                                                              $50164:
005172                                  CALL ALTBUF
005172  004767  051366                                                          JSR     PC,ALTBUF
005176                              ENDIF
005176                                                              $50165:
005176                          ENDIF
005176                                                              $50163:
005176                          LET BF.EXP := @DATA03
005176  017767  173366  173334                                                  MOV     @DATA03,BF.EXP

                        ;+
                        ; DO THE CIS INSTRUCTION
                        ;-

005204                          INLINE  <MOVRC>
005204  076031                                                                  MOVRC

                        ;+
                        ; CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS
                        ;-

005206                          LET CC := PSW
005206  016767  172564  173304                                                  MOV     PSW,CC
005214                          LET CC := CC CLR.BY #177760
005214  042767  177760  173276                                                  BIC     #177760,CC
005222                          IF CC NE CC.EXP THEN
005222  026767  173272  173272                                                  CMP     CC,CC.EXP
005230  001402                                                                  BEQ     $50166
```

```
2056 005232                             CALL ERROR                              JSR     PC,ERROR
     005232  004767  052442
2057 005236                             ENDIF
     005236                                                                  $50166:
2058 005236                             IF R0 NE R0.EXP THEN                    CMP     R0,R0.EXP
     005236  020067  173262                                                     BEQ     $50167
     005242  001402
2059 005244                                 CALL ERROR                          JSR     PC,ERROR
     005244  004767  052430
2060 005250                             ENDIF
     005250                                                                  $50167:
2061 005250                             IF R1 NE R1.EXP THEN                    CMP     R1,R1.EXP
     005250  020167  173252                                                     BEQ     $50170
     005254  001402
2062 005256                                 CALL ERROR                          JSR     PC,ERROR
     005256  004767  052416
2063 005262                             ENDIF
     005262                                                                  $50170:
2064 005262                             IF R2 NE R2.EXP THEN                    CMP     R2,R2.EXP
     005262  020267  173242                                                     BEQ     $50171
     005266  001402
2065 005270                                 CALL ERROR                          JSR     PC,ERROR
     005270  004767  052404
2066 005274                             ENDIF
     005274                                                                  $50171:
2067 005274                             IF R3 NE R3.EXP THEN                    CMP     R3,R3.EXP
     005274  020367  173232                                                     BEQ     $50172
     005300  001402
2068 005302                                 CALL ERROR                          JSR     PC,ERROR
     005302  004767  052372
2069 005306                             ENDIF
     005306                                                                  $50172:
2070 005306                             IF R4 NE R4.EXP THEN                    CMP     R4,R4.EXP
     005306  020467  173222                                                     BEQ     $50173
     005312  001402
2071 005314                                 CALL ERROR                          JSR     PC,ERROR
     005314  004767  052360
2072 005320                             ENDIF
     005320                                                                  $50173:
2073 005320                             IF R5 NE R5.EXP THEN                    CMP     R5,R5.EXP
     005320  020567  173212                                                     BEQ     $50174
     005324  001402
2074 005326                                 CALL ERROR                          JSR     PC,ERROR
     005326  004767  052346
2075 005332                             ENDIF
     005332                                                                  $50174:
2076
2077                                    ;+
2078                                    ; CHECK ENTIRE DESTINATION BUFFER FOR CORRECT RESULTS
2079                                    ;-
2080
2081 005332                                 LET ERRFLG := #0                    CLR     ERRFLG
     005332  005067  173120
2082 005336                                 LET R1 := DATA03                    MOV     DATA03,R1
     005336  016701  173226
2083 005342                                 IF DATA00 HIS DATA02 THEN           CMP     DATA00,DATA02
     005342  026767  173214  173216
```

```
     005350  103424                                                            BLO     $50175
     005352                          LET R3 := DATA02
     005352  016703  173210                                                    MOV     DATA02,R3
     005356                          LET R0 := DATA01 + DATA00 - DATA02
     005356  016700  173202                                                    MOV     DATA01,R0
     005362  066700  173174                                                    ADD     DATA00,R0
     005366  166700  173174                                                    SUB     DATA02,R0
     005372                          WHILE R3 NE #0 AND ERRFLG EQ #0 DO
     005372                                                           $50176:
     005372  005703                                                            TST     R3
     005374  001411                                                            BEQ     $50177
     005376  005767  173054                                                    TST     ERRFLG
     005402  001006                                                            BNE     $50177
     005404                              IFB (R0)+ NE (R1)+ THEN
     005404  122021                                                            CMPB    (R0)+,(R1)+
     005406  001402                                                            BEQ     $50200
     005410                                  CALL ERROR
     005410  004767  052264                                                    JSR     PC,ERROR
     005414                              ENDIF
     005414                                                           $50200:
     005414                              LET R3 := R3 - #1
     005414  005303                                                            DEC     R3
     005416                          ENDDO
     005416  000765                                                            BR      $50176
     005420                                                           $50177:
     005420                      ELSE
     005420  000455                                                            BR      $50201
     005422                                                           $50175:
     005422                          REPEAT
     005422                                                           $50202:
     005422                              IFB (R1)+ NE DATA04 THEN
     005422  122167  173144                                                    CMPB    (R1)+,DATA04
     005426  001402                                                            BEQ     $50203
     005430                                  CALL ERROR
     005430  004767  052244                                                    JSR     PC,ERROR
     005434                              ENDIF
     005434                                                           $50203:
     005434                              LET TEMPO := DATA03 + DATA02 - DATA00
     005434  016767  173130  173034                                            MOV     DATA03,TEMPO
     005442  066767  173120  173026                                            ADD     DATA02,TEMPO
     005450  166767  173106  173020                                            SUB     DATA00,TEMPO
     005456                          UNTIL R1 EQ TEMPO OR ERRFLG NE #0
     005456  020167  173014                                                    CMP     R1,TEMPO
     005462  001403                                                            BEQ     $50204
     005464  005767  172766                                                    TST     ERRFLG
     005470  001754                                                            BEQ     $50202
     005472                                                           $50204:
     005472                          LET R0 := DATA01
     005472  016700  173066                                                    MOV     DATA01,R0
     005476                          LET R1 := DATA03 + DATA02 - DATA00
     005476  016701  173066                                                    MOV     DATA03,R1
     005502  066701  173060                                                    ADD     DATA02,R1
     005506  166701  173050                                                    SUB     DATA00,R1
     005512                          LET TEMPO := DATA03 + DATA02
     005512  016767  173052  172756                                            MOV     DATA03,TEMPO
     005520  066767  173042  172750                                            ADD     DATA02,TEMPO
     005526                          WHILE R1 NE TEMPO AND ERRFLG NE #0 DO
```

```
        005526                                                                  $50205:
        005526  020167  172744                                                        CMP     R1,TEMPO
        005532  001410                                                          BEQ     $50206
        005534  005767  172716                                                  TST     ERRFLG
        005540  001405                                                          BEQ     $50206
 2103   005542                          IFB (R0)+ NE (R1)+ THEN
        005542  122021                                                          CMPB    (R0)+,(R1)+
        005544  001402                                                          BEQ     $50207
 2104   005546                              CALL ERROR
        005546  004767  052126                                                  JSR     PC,ERROR
 2105   005552                          ENDIF
        005552                                                                  $50207:
 2106   005552                      ENDDO
        005552  000765                                                          BR      $50205
        005554                                                                  $50206:
 2107   005554                  ENDIF
        005554                                                                  $50201:
 2108   005554                  IF DATA02 LO #BUFLEN THEN
        005554  026727  173006  000100                                                CMP     DATA02,#BUFLEN
        005562  103025                                                          BHIS    $50210
 2109   005564                      LET R0 := DATA03 + DATA02
        005564  016700  173000                                                  MOV     DATA03,R0
        005570  066700  172772                                                  ADD     DATA02,R0
 2110   005574                      REPEAT
        005574                                                                  $50211:
 2111   005574                          IFB (R0)+ NE BF.EXP THEN
        005574  122067  172740                                                          CMPB    (R0)+,BF.EXP
        005600  001402                                                          BEQ     $50212
 2112   005602                              CALL ERROR
        005602  004767  052072                                                  JSR     PC,ERROR
 2113   005606                          ENDIF
        005606                                                                  $50212:
 2114   005606                          LET TEMPO := DATA03 + #BUFLEN
        005606  016767  172756  172662                                                  MOV     DATA03,TEMPO
        005614  062767  000100  172654                                                  ADD     #BUFLEN,TEMPO
 2115   005622                      UNTIL R0 EQ TEMPO OR ERRFLG NE #0
        005622  020067  172650                                                          CMP     R0,TEMPO
        005626  001403                                                          BEQ     $50213
        005630  005767  172622                                                  TST     ERRFLG
        005634  001757                                                          BEQ     $50211
        005636                                                                  $50213:
 2116   005636                  ENDIF
        005636                                                                  $50210:
 2117   005636                  LET TSTCAS := TSTCAS + #16
        005636  062767  000016  172700                                                  ADD     #16,TSTCAS
 2118   005644                  LET COUNT := COUNT - #1
        005644  005367  172620                                                  DEC     COUNT
 2119                           ;UNTIL COUNT EQ #0
 2120   005650                    IF COUNT NE #0 THEN
        005650  005767  172614                                                  TST     COUNT
        005654  001402                                                          BEQ     $50214
 2121   005656                      INLINE  <JMP     $5031>
        005656  000167  177000                                                  JMP     $5031
 2122   005662                  ENDIF
        005662                                                                  $50214:
 2123
 2124
```

MOVRC

```
2125                                    ;+
2126                                    ; DO ONE INLINE CASE - MOVRCI
2127                                    ;-
2128
2129 005662                                LET BFADRS := #BUFF1
     005662  012767  063236  172652                                  MOV     #BUFF1,BFADRS
2130 005670                                CALL SETBUF
     005670  004767  050624                                         JSR     PC,SETBUF
2131 005674                                LET BFADRS := #BUFF2
     005674  012767  063336  172640                                 MOV     #BUFF2,BFADRS
2132 005702                                CALL CLRBUF
     005702  004767  050550                                         JSR     PC,CLRBUF
2133 005706                                LET R0 := #125252
     005706  012700  125252                                         MOV     #125252,R0
2134 005712                                LET R1 := #125252
     005712  012701  125252                                         MOV     #125252,R1
2135 005716                                LET R2 := #125252
     005716  012702  125252                                         MOV     #125252,R2
2136 005722                                LET R3 := #125252
     005722  012703  125252                                         MOV     #125252,R3
2137 005726                                LET R4 := #125252
     005726  012704  125252                                         MOV     #125252,R4
2138 005732                                LET R5 := #125252
     005732  012705  125252                                         MOV     #125252,R5
2139 005736                                LET ERRFLG := #0
     005736  005067  172514                                         CLR     ERRFLG
2140 005742                                INLINE  <MOVRCI>
     005742  076131                                                MOVRCI
2141 005744                                INLINE  <.WORD      TMVRCI>         ;SRC.DSCR.PTR
     005744  006240                                                .WORD   TMVRCI
2142 005746                                INLINE  <.WORD      TMVRCI+4>       ;DST.DSCR.PTR
     005746  006244                                                .WORD   TMVRCI+4
2143 005750                                INLINE  <.WORD      40>             ;FILL CHARACTER
     005750  000040                                                .WORD   40
2144 005752                                IFCOND CS THEN
     005752  103003                                                BCC     $50215
2145 005754                                    IFCOND EQ THEN
     005754  001002                                                BNE     $50216
2146 005756                                        CALL ERROR
     005756  004767  051716                                        JSR     PC,ERROR
2147 005762                                    ENDIF
     005762                                                 $50216:
2148 005762                                ENDIF
     005762                                                 $50215:
2149 005762                                IF R0 NE #125252 THEN
     005762  020027  125252                                         CMP     R0,#125252
     005766  001402                                                BEQ     $50217
2150 005770                                    CALL ERROR
     005770  004767  051704                                        JSR     PC,ERROR
2151 005774                                ENDIF
     005774                                                 $50217:
2152 005774                                IF R1 NE #125252 THEN
     005774  020127  125252                                         CMP     R1,#125252
     006000  001402                                                BEQ     $50220
2153 006002                                    CALL ERROR
     006002  004767  051672                                        JSR     PC,ERROR
2154 006006                                ENDIF
```

```
        006006                                  IF R2 NE #125252 THEN
   2155 006006  020227  125252
        006012  001402
   2156 006014      CALL ERROR
        006014  004767  051660
   2157 006020  ENDIF
        006020
   2158 006020  IF R3 NE #125252 THEN
        006020  020327  125252
        006024  001402
   2159 006026      CALL ERROR
        006026  004767  051646
   2160 006032  ENDIF
        006032
   2161 006032  IF R4 NE #125252 THEN
        006032  020427  125252
        006036  001402
   2162 006040      CALL ERROR
        006040  004767  051634
   2163 006044  ENDIF
        006044
   2164 006044  IF R5 NE #125252 THEN
        006044  020527  125252
        006050  001402
   2165 006052      CALL ERROR
        006052  004767  051622
   2166 006056  ENDIF
        006056
   2167 006056  IF BUFF2 NE #20040 THEN
        006056  026727  055254  020040
        006064  001402
   2168 006066      CALL ERROR
        006066  004767  051606
   2169 006072  ENDIF
        006072
   2170 006072  IF BUFF2+2 NE #177777 THEN
        006072  026727  055242  177777
        006100  001402
   2171 006102      CALL ERROR
        006102  004767  051572
   2172 006106  ENDIF
        006106
   2173 006106  LET ERRFLG := #0
        006106  005067  172344
   2174 006112  LET R0 := #BUFF2+4
        006112  012700  063342
   2175 006116  REPEAT
        006116
   2176 006116      IF (R0)+ NE #0 THEN
        006116  005720
        006120  001402
   2177 006122          CALL ERROR
        006122  004767  051552
   2178 006126      ENDIF
        006126
   2179 006126  UNTIL R0 EQ #BUFF2+BUFLEN-2 OR ERRFLG NE #0
```

```
$50220:
        CMP     R2,#125252
        BEQ     $50221
        JSR     PC,ERROR
$50221:
        CMP     R3,#125252
        BEQ     $50222
        JSR     PC,ERROR
$50222:
        CMP     R4,#125252
        BEQ     $50223
        JSR     PC,ERROR
$50223:
        CMP     R5,#125252
        BEQ     $50224
        JSR     PC,ERROR
$50224:
        CMP     BUFF2,#20040
        BEQ     $50225
        JSR     PC,ERROR
$50225:
        CMP     BUFF2+2,#177777
        BEQ     $50226
        JSR     PC,ERROR
$50226:
        CLR     ERRFLG
        MOV     #BUFF2+4,R0
$50227:
        TST     (R0)+
        BEQ     $50230
        JSR     PC,ERROR
$50230:
```

```
        006126  020027  063434                                                  CMP     R0,#BUFF2+BUFLEN-2
        006132  001403                                                          BEQ     $50231
        006134  005767  172316                                                  TST     ERRFLG
        006140  001766                                                          BEQ     $50227
        006142                                                          $50231:
  2180  006142                      INLINE  <JMP XMOVTC>
        006142  000167  000102                                                  JMP     XMOVTC
  2181
  2182                            ;+
  2183                            ;TABLE OF TEST CASES FOR MOVRC
  2184                            ;-
  2185
  2186  006146  000004            TMOVRC: .WORD   4               ;#OF TEST CASES FOR MOVRC
  2187
  2188  006150  000060                    .WORD   60                      ;SRC.LEN
  2189  006152  063336                    .WORD   BUFF2                   ;SRC.ADR
  2190  006154  000025                    .WORD   25                      ;DST.LEN
  2191  006156  063236                    .WORD   BUFF1                   ;DST.ADR
  2192  006160  000040                    .WORD   40                      ;FILL CHAR.
  2193  006162  000000                    .WORD   0                       ;SRC.BUFF.STATE
  2194  006164  000001                    .WORD   1                       ;DST.BUFF.STATE
  2195
  2196  006166  000017                    .WORD   17              ;TEST CASE NUMBER 2
  2197  006170  063336                    .WORD   BUFF2
  2198  006172  000050                    .WORD   50
  2199  006174  063236                    .WORD   BUFF1
  2200  006176  000100                    .WORD   100
  2201  006200  000001                    .WORD   1
  2202  006202  000000                    .WORD   0
  2203
  2204  006204  000030                    .WORD   30              ;TEST CASE NUMBER 3
  2205  006206  063236                    .WORD   BUFF1
  2206  006210  000012                    .WORD   12
  2207  006212  063336                    .WORD   BUFF2
  2208  006214  000077                    .WORD   77
  2209  006216  000000                    .WORD   0
  2210  006220  177777                    .WORD   -1
  2211
  2212  006222  000022                    .WORD   22              ;TEST CASE NUMBER 4
  2213  006224  063236                    .WORD   BUFF1
  2214  006226  000037                    .WORD   37
  2215  006230  063236                    .WORD   BUFF1
  2216  006232  000020                    .WORD   20
  2217  006234  177777                    .WORD   -1
  2218  006236  000001                    .WORD   1
  2219
  2220  006240  000002            TMVRCI: .WORD   2               ;TEST CASE FOR IN-LINE
  2221  006242  063236                    .WORD   BUFF1
  2222  006244  000004                    .WORD   4
  2223  006246  063336                    .WORD   BUFF2
  2224
```

```
                                .SBTTL  MOVTC
                                ;;*****************************************************************
                                ;
                                ;  MOVTC TEST
                                ;
                                ;;*****************************************************************

                                ROUTINE XMOVTC
                                                                                        XMOVTC:
                                   INLINE  <NOP>                                        NOP
                                   INLINE  <SCOPE>                                      SCOPE
                                   LET $TIMES := #20            ;ITERATION COUNT        MOV     #20,$TIMES
                                   LET COUNT := TMOVTC          ;GET TEST COUNT         MOV     TMOVTC,COUNT
                                   LET TSTCAS := #TMOVTC+2              ;POINT TO FIRST TEST CASE
                                                                                        MOV     #TMOVTC+2,TSTCAS
                                   INLINE  <$5032:>                                     $5032:

                                ;+
                                ; GET TEST DATA FROM TABLE
                                ;-

                                   LET R0 := TSTCAS                                     MOV     TSTCAS,R0
                                   LET DATA00 := (R0)+          ;SRC.LEN                MOV     (R0)+,DATA00
                                   LET DATA01 := (R0)+          ;SRC.ADR                MOV     (R0)+,DATA01
                                   LET DATA02 := (R0)+          ;DST.LEN                MOV     (R0)+,DATA02
                                   LET DATA03 := (R0)+          ;DST.ADR                MOV     (R0)+,DATA03
                                   LET DATA04 := (R0)+          ;FILL CHAR.             MOV     (R0)+,DATA04
                                   LET DATA05 := (R0)+          ;SRC.BUFF.STATE         MOV     (R0)+,DATA05
                                   LET DATA06 := (R0)           ;DST.BUFF.STATE         MOV     (R0),DATA06

                                ;+
                                ; SET UP GPR'S,
                                ; EXPECTED GPR'S AND CONDITION CODES,
                                ; AND BUFFERS.
                                ;-

                                   LET R0 := DATA00                                     MOV     DATA00,R0
                                   LET R1 := DATA01                                     MOV     DATA01,R1
                                   LET R2 := DATA02                                     MOV     DATA02,R2
                                   LET R3 := DATA03                                     MOV     DATA03,R3
```

H5


```
 2264 006356                                LET R4 := DATA04
      006356  016704  172210                                                            MOV     DATA04,R4
 2265 006362                                LET R5 := #TRANS
      006362  012705  063436                                                            MOV     #TRANS,R5
 2266 006366                                INLINE  <CMP R0,R2>
      006366  020002                                                                    CMP     R0,R2
 2267 006370                                LET CC.EXP := PSW
      006370  016767  171402  172124                                                    MOV     PSW,CC.EXP
 2268 006376                                LET CC.EXP := CC.EXP  CLR.BY  #177760
      006376  042767  177760  172116                                                    BIC     #177760,CC.EXP
 2269 006404                                IF #CBIT SETIN CC.EXP OR #ZBIT SETIN CC.EXP THEN
      006404  032767  000001  172110                                                    BIT     #CBIT,CC.EXP
      006412  001004                                                                    BNE     $50234
      006414  032767  000004  172100                                                    BIT     #ZBIT,CC.EXP
      006422  001403                                                                    BEQ     $50235
      006424                                                                   $50234:
 2270 006424                                    LET R0.EXP := #0
      006424  005067  172074                                                            CLR     R0.EXP
 2271 006430                                ELSE
      006430  000404                                                                    BR      $50236
      006432                                                                   $50235:
 2272 006432                                    LET R0.EXP := R0 - R2
      006432  010067  172066                                                            MOV     R0,R0.EXP
      006436  160267  172062                                                            SUB     R2,R0.EXP
 2273 006442                                ENDIF
      006442                                                                   $50236:
 2274 006442                                LET R1.EXP := #0
      006442  005067  172060                                                            CLR     R1.EXP
 2275 006446                                LET R2.EXP := #0
      006446  005067  172056                                                            CLR     R2.EXP
 2276 006452                                LET R3.EXP := #0
      006452  005067  172054                                                            CLR     R3.EXP
 2277 006456                                LET R4.EXP := R4
      006456  010467  172052                                                            MOV     R4,R4.EXP
 2278 006462                                LET R5.EXP := R5
      006462  010567  172050                                                            MOV     R5,R5.EXP
 2279 006466                                LET BFADRS := DATA01
      006466  016767  172072  172046                                                    MOV     DATA01,BFADRS
 2280 006474                                IF DATA05 EQ #0 THEN
      006474  005767  172074                                                            TST     DATA05
      006500  001003                                                                    BNE     $50237
 2281 006502                                    CALL CLRBUF
      006502  004767  047750                                                            JSR     PC,CLRBUF
 2282 006506                                ELSE
      006506  000411                                                                    BR      $50240
      006510                                                                   $50237:
 2283 006510                                    IF DATA05 EQ #177777 THEN
      006510  026727  172060  177777                                                    CMP     DATA05,#177777
      006516  001003                                                                    BNE     $50241
 2284 006520                                        CALL SETBUF
      006520  004767  047774                                                            JSR     PC,SETBUF
 2285 006524                                    ELSE
      006524  000402                                                                    BR      $50242
      006526                                                                   $50241:
 2286 006526                                        CALL ALTBUF
      006526  004767  050032                                                            JSR     PC,ALTBUF
 2287 006532                                    ENDIF
```

```
        006532                                                              $50242:
   2288 006532                              ENDIF
        006532                                                              $50240:
   2289 006532                              LET BFADRS := DATA03
        006532  016767  172032  172002                                              MOV     DATA03,BFADRS
   2290 006540                              IF DATA06 EQ #0 THEN
        006540  005767  172032                                                      TST     DATA06
        006544  001003                                                              BNE     $50243
   2291 006546                                  CALL CLRBUF
        006546  004767  047704                                                      JSR     PC,CLRBUF
   2292 006552                              ELSE
        006552  000411                                                              BR      $50244
        006554                                                              $50243:
   2293 006554                                  IF DATA06 EQ #177777 THEN
        006554  026727  172016  177777                                              CMP     DATA06,#177777
        006562  001003                                                              BNE     $50245
   2294 006564                                      CALL SETBUF
        006564  004767  047730                                                      JSR     PC,SETBUF
   2295 006570                                  ELSE
        006570  000402                                                              BR      $50246
        006572                                                              $50245:
   2296 006572                                      CALL ALTBUF
        006572  004767  047766                                                      JSR     PC,ALTBUF
   2297 006576                                  ENDIF
        006576                                                              $50246:
   2298 006576                              ENDIF
        006576                                                              $50244:
   2299 006576                              LET BF.EXP := @DATA03
        006576  017767  171766  171734                                              MOV     @DATA03,BF.EXP
   2300
   2301                                 ;+
   2302                                 ; SET UP THE TRANSLATION TABLE
   2303                                 ;-
   2304
   2305 006604                              DECR TEMPO FROM #200 TO #1 BY #1
        006604  012767  000200  171664                                              MOV     #200,TEMPO
        006612  000402                                                              BR      $50247
        006614                                                              $50250:
        006614  005367  171656                                                      DEC     TEMPO
        006620                                                              $50247:
        006620  026727  171652  000001                                              CMP     TEMPO,#1
        006626  002402                                                              BLT     $50251
   2306 006630                                  LET (R5)+ := #0
        006630  005025                                                              CLR     (R5)+
   2307 006632                              ENDDEC
        006632  000770                                                              BR      $50250
        006634                                                              $50251:
   2308 006634                              LET R5 := #TRANS
        006634  012705  063436                                                      MOV     #TRANS,R5
   2309 006640                              LET (R5) :B= #63
        006640  112715  000063                                                      MOVB    #63,(R5)
   2310 006644                              LET 252(R5) :B= #314
        006644  112765  000314  000252                                              MOVB    #314,252(R5)
   2311 006652                              LET 377(R5) :B= #125
        006652  112765  000125  000377                                              MOVB    #125,377(R5)
   2312
   2313                                 ;+
```

```
                                        ; DO THE CIS INSTRUCTION
                                        ;-

006660                                     INLINE  <MOVTC>
006660  076032                                                                  MOVTC

                                        ;+
                                        ; CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS
                                        ;-

006662                                     LET CC := PSW
006662  016767  171110  171630                                                  MOV     PSW,CC
006670                                     LET CC := CC CLR.BY #177760
006670  042767  177760  171622                                                  BIC     #177760,CC
006676                                     IF CC NE CC.EXP THEN
006676  026767  171616  171616                                                  CMP     CC,CC.EXP
006704  001402                                                                  BEQ     $50252
006706                                         CALL ERROR
006706  004767  050766                                                          JSR     PC,ERROR
006712                                     ENDIF
006712                                                                  $50252:
006712                                     IF R0 NE R0.EXP THEN
006712  020067  171606                                                          CMP     R0,R0.EXP
006716  001402                                                                  BEQ     $50253
006720                                         CALL ERROR
006720  004767  050754                                                          JSR     PC,ERROR
006724                                     ENDIF
006724                                                                  $50253:
006724                                     IF R1 NE R1.EXP THEN
006724  020167  171576                                                          CMP     R1,R1.EXP
006730  001402                                                                  BEQ     $50254
006732                                         CALL ERROR
006732  004767  050742                                                          JSR     PC,ERROR
006736                                     ENDIF
006736                                                                  $50254:
006736                                     IF R2 NE R2.EXP THEN
006736  020267  171566                                                          CMP     R2,R2.EXP
006742  001402                                                                  BEQ     $50255
006744                                         CALL ERROR
006744  004767  050730                                                          JSR     PC,ERROR
006750                                     ENDIF
006750                                                                  $50255:
006750                                     IF R3 NE R3.EXP THEN
006750  020367  171556                                                          CMP     R3,R3.EXP
006754  001402                                                                  BEQ     $50256
006756                                         CALL ERROR
006756  004767  050716                                                          JSR     PC,ERROR
006762                                     ENDIF
006762                                                                  $50256:
006762                                     IF R4 NE R4.EXP THEN
006762  020467  171546                                                          CMP     R4,R4.EXP
006766  001402                                                                  BEQ     $50257
006770                                         CALL ERROR
006770  004767  050704                                                          JSR     PC,ERROR
006774                                     ENDIF
006774                                                                  $50257:
006774                                     IF R5 NE R5.EXP THEN
```

```
        006774  020567  171536                                                  CMP     R5,R5.EXP
        007000  001402                                                          BEQ     $50260
   2344 007002                          CALL ERROR
        007002  004767  050672                                                  JSR     PC,ERROR
   2345 007006                          ENDIF
        007006                                                          $50260:
   2346
   2347                                 ;+
   2348                                 ; CHECK ENTIRE DESTINATION BUFFER FOR CORRECT RESULTS
   2349                                 ;-
   2350
   2351 007006                          LET R0 := DATA03
        007006  016700  171556                                                  MOV     DATA03,R0
   2352 007012                          LET R2 :B= @DATA01
        007012  117702  171546                                                  MOVB    @DATA01,R2
   2353 007016                          LET R2 := R2 CLR.BY #177400
        007016  042702  177400                                                  BIC     #177400,R2
   2354 007022                          LET R1 :B= TRANS(R2)
        007022  116201  063436                                                  MOVB    TRANS(R2),R1
   2355 007026                          LET ERRFLG := #0
        007026  005067  171424                                                  CLR     ERRFLG
   2356 007032                          IF DATA00 HIS DATA02 THEN
        007032  026767  171524  171526                                          CMP     DATA00,DATA02
        007040  103416                                                         BLO     $50261
   2357 007042                              LET R3 := DATA02
        007042  016703  171520                                                  MOV     DATA02,R3
   2358 007046                              WHILE R3 NE #0 AND ERRFLG EQ #0 DO
        007046                                                          $50262:
        007046  005703                                                          TST     R3
        007050  001411                                                          BEQ     $50263
        007052  005767  171400                                                  TST     ERRFLG
        007056  001006                                                         BNE     $50263
   2359 007060                                  IFB (R0)+ NE R1 THEN
        007060  122001                                                         CMPB    (R0)+,R1
        007062  001402                                                         BEQ     $50264
   2360 007064                                      CALL ERROR
        007064  004767  050610                                                 JSR     PC,ERROR
   2361 007070                                  ENDIF
        007070                                                          $50264:
   2362 007070                                  LET R3 := R3 - #1
        007070  005303                                                         DEC     R3
   2363 007072                              ENDDO
        007072  000765                                                         BR      $50262
        007074                                                          $50263:
   2364 007074                          ELSE
        007074  000440                                                         BR      $50265
        007076                                                          $50261:
   2365 007076                              LET R3 := DATA00
        007076  016703  171460                                                 MOV     DATA00,R3
   2366 007102                              WHILE R3 NE #0 AND ERRFLG EQ #0 DO
        007102                                                          $50266:
        007102  005703                                                          TST     R3
        007104  001411                                                          BEQ     $50267
        007106  005767  171344                                                  TST     ERRFLG
        007112  001006                                                         BNE     $50267
   2367 007114                                  IFB (R0)+ NE R1 THEN
        007114  122001                                                         CMPB    (R0)+,R1
```

```
        007116  001402
 2368   007120                          CALL ERROR
        007120  004767  050554
 2369   007124                      ENDIF
        007124
 2370   007124                      LET R3 := R3 - #1
        007124  005303
 2371   007126                  ENDDO
        007126  000765
        007130
 2372   007130                  LET R0 := DATA03 + DATA00
        007130  016700  171434
        007134  066700  171422
 2373   007140                  LET R3 := DATA02 - DATA00
        007140  016703  171422
        007144  166703  171412
 2374   007150                  REPEAT
        007150
 2375   007150                      IFB (R0)+ NE DATA04 THEN
        007150  122067  171416
        007154  001402
 2376   007156                          CALL ERROR
        007156  004767  050516
 2377   007162                      ENDIF
        007162
 2378   007162                      LET R3 := R3 - #1
        007162  005303
 2379   007164                  UNTIL R3 EQ #0 OR ERRFLG NE #0
        007164  005703
        007166  001403
        007170  005767  171262
        007174  001765
        007176
 2380   007176              ENDIF
        007176
 2381   007176              IF DATA02 LO #BUFLEN THEN
        007176  026727  171364  000100
        007204  103027
 2382   007206                  LET TEMP0 := DATA03 + #BUFLEN - #1
        007206  016767  171356  171262
        007214  062767  000100  171254
        007222  005367  171250
 2383   007226                  LET R0 := DATA03 + DATA02
        007226  016700  171336
        007232  066700  171330
 2384   007236                  REPEAT
        007236
 2385   007236                      IFB (R0)+ NE BF.EXP THEN
        007236  122067  171276
        007242  001402
 2386   007244                          CALL ERROR
        007244  004767  050430
 2387   007250                      ENDIF
        007250
 2388   007250                  UNTIL R0 EQ TEMP0 OR ERRFLG NE #0
        007250  020067  171222
        007254  001403
```

```
                        BEQ     $50270
                        JSR     PC,ERROR
                $50270:
                        DEC     R3
                        BR      $50266
                $50267:
                        MOV     DATA03,R0
                        ADD     DATA00,R0
                        MOV     DATA02,R3
                        SUB     DATA00,R3
                $50271:
                        CMPB    (R0)+,DATA04
                        BEQ     $50272
                        JSR     PC,ERROR
                $50272:
                        DEC     R3
                        TST     R3
                        BEQ     $50273
                        TST     ERRFLG
                        BEQ     $50271
                $50273:
                $50265:
                        CMP     DATA02,#BUFLEN
                        BHIS    $50274
                        MOV     DATA03,TEMP0
                        ADD     #BUFLEN,TEMP0
                        DEC     TEMP0
                        MOV     DATA03,R0
                        ADD     DATA02,R0
                $50275:
                        CMPB    (R0)+,BF.EXP
                        BEQ     $50276
                        JSR     PC,ERROR
                $50276:
                        CMP     R0,TEMP0
                        BEQ     $50277
```

```
        007256  005767  171174                                                  TST     ERRFLG
        007262  001765                                                          BEQ     $50275
        007264                                                          $50277:
  2389  007264                      ENDIF
        007264                                                          $50274:
  2390  007264                      LET TSTCAS := TSTCAS + #16
        007264  062767  000016  171252                                          ADD     #16,TSTCAS
  2391  007272                      LET COUNT := COUNT - #1
        007272  005367  171172                                                  DEC     COUNT
  2392                            ;UNTIL COUNT EQ #0
  2393  007276                      IF COUNT NE #0 THEN
        007276  005767  171166                                                  TST     COUNT
        007302  001402                                                          BEQ     $50300
  2394  007304                          INLINE  <JMP     $5032>
        007304  000167  176766                                                  JMP     $5032
  2395  007310                      ENDIF
        007310                                                          $50300:
  2396
  2397
  2398                            ;+
  2399                            ; DO ONE INLINE CASE - MOVTCI
  2400                            ;-
  2401
  2402  007310                      LET BFADRS := #BUFF1
        007310  012767  063236  171224                                          MOV     #BUFF1,BFADRS
  2403  007316                      CALL CLRBUF
        007316  004767  047134                                                  JSR     PC,CLRBUF
  2404  007322                      LET BFADRS := #BUFF2
        007322  012767  063336  171212                                          MOV     #BUFF2,BFADRS
  2405  007330                      CALL SETBUF
        007330  004767  047164                                                  JSR     PC,SETBUF
  2406  007334                      LET R0 := #125252
        007334  012700  125252                                                  MOV     #125252,R0
  2407  007340                      LET R1 := #125252
        007340  012701  125252                                                  MOV     #125252,R1
  2408  007344                      LET R2 := #125252
        007344  012702  125252                                                  MOV     #125252,R2
  2409  007350                      LET R3 := #125252
        007350  012703  125252                                                  MOV     #125252,R3
  2410  007354                      LET R4 := #125252
        007354  012704  125252                                                  MOV     #125252,R4
  2411  007360                      LET R5 := #125252
        007360  012705  125252                                                  MOV     #125252,R5
  2412  007364                      LET TRANS := #70707
        007364  012767  070707  054044                                          MOV     #70707,TRANS
  2413  007372                      INLINE  <MOVTCI>
        007372  076132                                                  MOVTCI
  2414  007374                      INLINE  <.WORD       TMVTCI>
        007374  007700                                                  .WORD   TMVTCI
  2415  007376                      INLINE  <.WORD       TMVTCI+4>
        007376  007704                                                  .WORD   TMVTCI+4
  2416  007400                      INLINE  <.WORD       40>
        007400  000040                                                  .WORD   40
  2417  007402                      INLINE       <.WORD  TRANS>
        007402  063436                                                  .WORD   TRANS
  2418  007404                      IFCOND CS THEN
        007404  103003                                                  BCC     $50301
```

```
2419 007406                             IFCOND EQ THEN
     007406  001002                                                    BNE     $50302
2420 007410                                 CALL ERROR
     007410  004767  050264                                           JSR     PC,ERROR
2421 007414                             ENDIF
     007414                                                         $50302:
2422 007414                         ENDIF
     007414                                                         $50301:
2423 007414                         IF R0 NE #125252 THEN
     007414  020027  125252                                         CMP     R0,#125252
     007420  001402                                                 BEQ     $50303
2424 007422                             CALL ERROR
     007422  004767  050252                                         JSR     PC,ERROR
2425 007426                         ENDIF
     007426                                                         $50303:
2426 007426                         IF R1 NE #125252 THEN
     007426  020127  125252                                         CMP     R1,#125252
     007432  001402                                                 BEQ     $50304
2427 007434                             CALL ERROR
     007434  004767  050240                                         JSR     PC,ERROR
2428 007440                         ENDIF
     007440                                                         $50304:
2429 007440                         IF R2 NE #125252 THEN
     007440  020227  125252                                         CMP     R2,#125252
     007444  001402                                                 BEQ     $50305
2430 007446                             CALL ERROR
     007446  004767  050226                                         JSR     PC,ERROR
2431 007452                         ENDIF
     007452                                                         $50305:
2432 007452                         IF R3 NE #125252 THEN
     007452  020327  125252                                         CMP     R3,#125252
     007456  001402                                                 BEQ     $50306
2433 007460                             CALL ERROR
     007460  004767  050214                                         JSR     PC,ERROR
2434 007464                         ENDIF
     007464                                                         $50306:
2435 007464                         IF R4 NE #125252 THEN
     007464  020427  125252                                         CMP     R4,#125252
     007470  001402                                                 BEQ     $50307
2436 007472                             CALL ERROR
     007472  004767  050202                                         JSR     PC,ERROR
2437 007476                         ENDIF
     007476                                                         $50307:
2438 007476                         IF R5 NE #125252 THEN
     007476  020527  125252                                         CMP     R5,#125252
     007502  001402                                                 BEQ     $50310
2439 007504                             CALL ERROR
     007504  004767  050170                                         JSR     PC,ERROR
2440 007510                         ENDIF
     007510                                                         $50310:
2441 007510                         IF BUFF2 NE #143707 THEN
     007510  026727  053622  143707                                 CMP     BUFF2,#143707
     007516  001402                                                 BEQ     $50311
2442 007520                             CALL ERROR
     007520  004767  050154                                         JSR     PC,ERROR
2443 007524                         ENDIF
     007524                                                         $50311:
```

```
 2444 007524                            IF BUFF2+2 NE #20040 THEN
      007524  026727  053610  020040                                        CMP     BUFF2+2,#20040
      007532  001402                                                       BEQ     $50312
 2445 007534                                CALL ERROR
      007534  004767  050140                                               JSR     PC,ERROR
 2446 007540                            ENDIF
      007540                                                          $50312:
 2447 007540                            LET R0 := #BUFF2 + #4
      007540  012700  063336                                               MOV     #BUFF2,R0
      007544  062700  000004                                               ADD     #4,R0
 2448 007550                            LET ERRFLG := #0
      007550  005067  170702                                               CLR     ERRFLG
 2449 007554                            REPEAT
      007554                                                          $50313:
 2450 007554                                IF (R0)+ NE #177777 THEN
      007554  022027  177777                                               CMP     (R0)+,#177777
      007560  001402                                                       BEQ     $50314
 2451 007562                                    CALL ERROR
      007562  004767  050112                                               JSR     PC,ERROR
 2452 007566                                ENDIF
      007566                                                          $50314:
 2453 007566                            UNTIL R0 EQ #BUFF2+BUFLEN OR ERRFLG NE #0
      007566  020027  063436                                               CMP     R0,#BUFF2+BUFLEN
      007572  001403                                                       BEQ     $50315
      007574  005767  170656                                               TST     ERRFLG
      007600  001765                                                       BEQ     $50313
      007602                                                          $50315:
 2454 007602                            INLINE  <JMP XLOCC>
      007602  000167  000102                                               JMP     XLOCC
 2455
 2456                                   ;+
 2457                                   ; TABLE OF TEST CASES FOR MOVTC
 2458                                   ;-
 2459
 2460 007606  000004                    TMOVTC: .WORD   4               ;#OF TEST CASES FOR MOVTC
 2461
 2462 007610  000061                            .WORD   61                      ;SRC.LEN
 2463 007612  063336                            .WORD   BUFF2                   ;SRC.ADR
 2464 007614  000027                            .WORD   27                      ;DST.LEN
 2465 007616  063236                            .WORD   BUFF1                   ;DST.ADR
 2466 007620  000040                            .WORD   40                      ;FILL CHARACTER
 2467 007622  000000                            .WORD   0                       ;SRC.BUFF.STATE
 2468 007624  000001                            .WORD   1                       ;DST.BUFF.STATE
 2469
 2470 007626  000055                            .WORD   55              ;TEST CASE NUMBER 2
 2471 007630  063336                            .WORD   BUFF2
 2472 007632  000075                            .WORD   75
 2473 007634  063236                            .WORD   BUFF1
 2474 007636  000100                            .WORD   100
 2475 007640  000001                            .WORD   1
 2476 007642  000000                            .WORD   0
 2477
 2478 007644  000032                            .WORD   32              ;TEST CASE NUMBER 3
 2479 007646  063236                            .WORD   BUFF1
 2480 007650  000031                            .WORD   31
 2481 007652  063336                            .WORD   BUFF2
 2482 007654  000073                            .WORD   73
```

MOVTC

```
 2483 007656  000000                        .WORD   0
 2484 007660  177777                        .WORD   -1
 2485
 2486 007662  000005                        .WORD   5               ;TEST CASE NUMBER 4
 2487 007664  063236                        .WORD   BUFF1
 2488 007666  000026                        .WORD   26
 2489 007670  063336                        .WORD   BUFF2
 2490 007672  000020                        .WORD   20
 2491 007674  177777                        .WORD   -1
 2492 007676  000001                        .WORD   1
 2493
 2494 007700  000002                TMVTCI: .WORD   2               ;TEST CASE FOR INLINE
 2495 007702  063236                        .WORD   BUFF1
 2496 007704  000004                        .WORD   4
 2497 007706  063336                        .WORD   BUFF2
 2498
```

```
2500                                .SBTTL  LOCC
2501                                ;;*****************************************************************
2502                                ;
2503                                ;  LOCC TEST
2504                                ;
2505                                ;;*****************************************************************
2506
2507 007710                        ROUTINE XLOCC
     007710                                                                          XLOCC:
2508 007710                           INLINE  <SCOPE>
     007710  000004                                                                 SCOPE
2509 007712                           INLINE  <NOP>
     007712  000240                                                                 NOP
2510 007714                           LET $TIMES := #20          ;ITERATION COUNT
     007714  012767  000020  170462                                                 MOV     #20,$TIMES
2511 007722                           LET COUNT := TLOCC         ;GET TEST COUNT
     007722  016767  000616  170540                                                 MOV     TLOCC,COUNT
2512 007730                           LET TSTCAS := #TLOCC+2             ;POINT TO FIRST TEST CASE
     007730  012767  010546  170606                                                 MOV     #TLOCC+2,TSTCAS
2513                                ;REPEAT
2514 007736                           INLINE  <$5040:>
     007736                                                                         $5040:
2515
2516                                ;+
2517                                ; GET TEST DATA FROM TABLE
2518                                ;
2519
2520 007736                           LET R0 := TSTCAS
     007736  016700  170602                                                         MOV     TSTCAS,R0
2521 007742                           LET DATA00 := (R0)+        ;SRC.LEN
     007742  012067  170614                                                         MOV     (R0)+,DATA00
2522 007746                           LET DATA01 := (R0)+        ;CHAR.SEARCHED FOR
     007746  012067  170612                                                         MOV     (R0)+,DATA01
2523 007752                           LET DATA02 := (R0)         ;OFFSET
     007752  011067  170610                                                         MOV     (R0),DATA02
2524
2525                                ;+
2526                                ; SET UP GPR'S,
2527                                ; EXPECTED GPR'S AND CONDITION CODES,
2528                                ; AND SOURCE BUFFER.
2529                                ;-
2530
2531 007756                           LET R0 := DATA00
     007756  016700  170600                                                         MOV     DATA00,R0
2532 007762                           LET R1 := #BUFF1
     007762  012701  063236                                                         MOV     #BUFF1,R1
2533 007766                           LET R4 := DATA01
     007766  016704  170572                                                         MOV     DATA01,R4
2534 007772                           INLINE  <BIT    #BIT00,COUNT>
     007772  032767  000001  170470                                                 BIT     #BIT00,COUNT
2535 010000                           IFCOND EQ THEN
     010000  001004                                                                 BNE     $50320
2536 010002                               LET R2 := #0
     010002  005002                                                                 CLR     R2
2537 010004                               LET R3 := #0
     010004  005003                                                                 CLR     R3
2538 010006                               LET R5 := #0
```

LOCC

```
      010006  005005
 2539 010010                          ELSE
      010010  000406
      010012
 2540 010012                              LET R2 := #177777
      010012  012702  177777
 2541 010016                              LET R3 := #177777
      010016  012703  177777
 2542 010022                              LET R5 := #177777
      010022  012705  177777
 2543 010026                          ENDIF
      010026
 2544 010026                          IF DATA02 HI DATA00 THEN
      010026  026767  170534  170526
      010034  101411
 2545 010036                              LET R0.EXP := #0
      010036  005067  170462
 2546 010042                              LET R1.EXP := #BUFF1 + DATA00
      010042  012767  063236  170456
      010050  066767  170506  170450
 2547 010056                          ELSE
      010056  000414
      010060
 2548 010060                              LET R0.EXP := DATA00 - DATA02
      010060  016767  170476  170436
      010066  166767  170474  170430
 2549 010074                              LET R1.EXP := #BUFF1 + DATA02
      010074  012767  063236  170424
      010102  066767  170460  170416
 2550 010110                          ENDIF
      010110
 2551 010110                          LET R2.EXP := R2
      010110  010267  170414
 2552 010114                          LET R3.EXP := R3
      010114  010367  170412
 2553 010120                          LET R4.EXP := R4
      010120  010467  170410
 2554 010124                          LET R5.EXP := R5
      010124  010567  170406
 2555 010130                          INLINE  <TST   R0.EXP>
      010130  005767  170370
 2556 010134                          LET CC.EXP := PSW
      010134  016767  167636  170360
 2557 010142                          LET CC.EXP := CC.EXP CLR.BY #177760
      010142  042767  177760  170352
 2558 010150                          LET BFADRS := #BUFF1
      010150  012767  063236  170364
 2559 010156                          CALL CLRBUF
      010156  004767  046274
 2560 010162                          LET TEMP0 := #BUFF1 + DATA02
      010162  012767  063236  170306
      010170  066767  170372  170300
 2561 010176                          LET @TEMP0 :B= DATA01
      010176  116777  170362  170272
 2562
 2563                             ;+
 2564                             ; DO THE CIS INSTRUCTION.
```

```
                        CLR     R5
                        BR      $50321
$50320:
                        MOV     #177777,R2
                        MOV     #177777,R3
                        MOV     #177777,R5
$50321:
                        CMP     DATA02,DATA00
                        BLOS    $50322
                        CLR     R0.EXP
                        MOV     #BUFF1,R1.EXP
                        ADD     DATA00,R1.EXP
                        BR      $50323
$50322:
                        MOV     DATA00,R0.EXP
                        SUB     DATA02,R0.EXP
                        MOV     #BUFF1,R1.EXP
                        ADD     DATA02,R1.EXP
$50323:
                        MOV     R2,R2.EXP
                        MOV     R3,R3.EXP
                        MOV     R4,R4.EXP
                        MOV     R5,R5.EXP
                        TST   R0.EXP
                        MOV     PSW,CC.EXP
                        BIC     #177760,CC.EXP
                        MOV     #BUFF1,BFADRS
                        JSR     PC,CLRBUF
                        MOV     #BUFF1,TEMP0
                        ADD     DATA02,TEMP0
                        MOVB    DATA01,@TEMP0
```

```
 2565                                   ;-
 2566
 2567 010204                               INLINE  <LOCC>                                                    LOCC
      010204  076040
 2568
 2569                                   ;+
 2570                                   ; CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS
 2571                                   ;-
 2572
 2573 010206                               LET CC := PSW                                                     MOV     PSW,CC
      010206  016767  167564  170304
 2574 010214                               LET CC := CC CLR.BY #177760                                       BIC     #177760,CC
      010214  042767  177760  170276
 2575 010222                               IF CC NE CC.EXP THEN                                              CMP     CC,CC.EXP
      010222  026767  170272  170272                                                                       BEQ     $50324
      010230  001402
 2576 010232                                   CALL ERROR                                                    JSR     PC,ERROR
      010232  004767  047442
 2577 010236                               ENDIF
      010236                                                                                          $50324:
 2578 010236                               IF R0 NE R0.EXP THEN                                              CMP     R0,R0.EXP
      010236  020067  170262                                                                               BEQ     $50325
      010242  001402
 2579 010244                                   CALL ERROR                                                    JSR     PC,ERROR
      010244  004767  047430
 2580 010250                               ENDIF
      010250                                                                                          $50325:
 2581 010250                               IF R1 NE R1.EXP THEN                                              CMP     R1,R1.EXP
      010250  020167  170252                                                                               BEQ     $50326
      010254  001402
 2582 010256                                   CALL ERROR                                                    JSR     PC,ERROR
      010256  004767  047416
 2583 010262                               ENDIF                                                       $50326:
      010262
 2584 010262                               IF R2 NE R2.EXP THEN                                              CMP     R2,R2.EXP
      010262  020267  170242                                                                               BEQ     $50327
      010266  001402
 2585 010270                                   CALL ERROR                                                    JSR     PC,ERROR
      010270  004767  047404
 2586 010274                               ENDIF                                                       $50327:
      010274
 2587 010274                               IF R3 NE R3.EXP THEN                                              CMP     R3,R3.EXP
      010274  020367  170232                                                                               BEQ     $50330
      010300  001402
 2588 010302                                   CALL ERROR                                                    JSR     PC,ERROR
      010302  004767  047372
 2589 010306                               ENDIF                                                       $50330:
      010306
 2590 010306                               IF R4 NE R4.EXP THEN                                              CMP     R4,R4.EXP
      010306  020467  170222                                                                               BEQ     $50331
      010312  001402
 2591 010314                                   CALL ERROR                                                    JSR     PC,ERROR
      010314  004767  047360
 2592 010320                               ENDIF                                                       $50331:
      010320
 2593 010320                               IF R5 NE R5.EXP THEN                                              CMP     R5,R5.EXP
      010320  020567  170212
```

```
      010324  001402                                                                BEQ     $50332
 2594 010326                              CALL ERROR
      010326  004767  047346                                                       JSR     PC,ERROR
 2595 010332                            ENDIF
      010332                                                                $50332:
 2596 010332                            LET TSTCAS := TSTCAS + #6
      010332  062767  000006  170204                                              ADD     #6,TSTCAS
 2597 010340                            LET COUNT := COUNT - #1
      010340  005367  170124                                                       DEC     COUNT
 2598                                  ;UNTIL COUNT EQ #0
 2599 010344                            IF COUNT NE #0 THEN
      010344  005767  170120                                                       TST     COUNT
      010350  001402                                                               BEQ     $50333
 2600 010352                                INLINE  <JMP     $5040>
      010352  000167  177360                                                       JMP     $5040
 2601 010356                            ENDIF
      010356                                                                $50333:
 2602
 2603
 2604                                  ;+
 2605                                  ; DO ONE INLINE CASE - LOCCI
 2606                                  ;-
 2607
 2608 010356                            LET BFADRS := #BUFF1
      010356  012767  063236  170156                                              MOV     #BUFF1,BFADRS
 2609 010364                            CALL SETBUF
      010364  004767  046130                                                       JSR     PC,SETBUF
 2610 010370                            CALL CLRREG                ;CLEAR R0 - R5
      010370  004767  046044                                                       JSR     PC,CLRREG
 2611 010374                            LET TEMPO := #BUFF1 + #25
      010374  012767  063236  170074                                              MOV     #BUFF1,TEMPO
      010402  062767  000025  170066                                              ADD     #25,TEMPO
 2612 010410                            LET @TEMPO :B= #40          ;SEARCH CHARACTER
      010410  112777  000040  170060                                              MOVB    #40,@TEMPO
 2613 010416                            INLINE  <LOCCI>
      010416  076140                                                               LOCCI
 2614 010420                            INLINE  <.WORD       TLOCCI>
      010420  010570                                                               .WORD   TLOCCI
 2615 010422                            INLINE  <.WORD       40>
      010422  000040                                                               .WORD   40
 2616 010424                            LET TEMPO := PSW
      010424  016767  167346  170044                                              MOV     PSW,TEMPO
 2617 010432                            INLINE  <BIT #17,TEMPO>
      010432  032767  000017  170036                                              BIT     #17,TEMPO
 2618 010440                            IFCOND NE THEN
      010440  001402                                                               BEQ     $50334
 2619 010442                                CALL ERROR
      010442  004767  047232                                                       JSR     PC,ERROR
 2620 010446                            ENDIF
      010446                                                                $50334:
 2621 010446                            LET TEMPO := #BUFLEN-25
      010446  012767  000053  170022                                              MOV     #BUFLEN-25,TEMPO
 2622 010454                            IF R0 NE TEMPO THEN
      010454  020067  170016                                                       CMP     R0,TEMPO
      010460  001402                                                               BEQ     $50335
 2623 010462                                CALL ERROR
      010462  004767  047212                                                       JSR     PC,ERROR
```

```
2624 010466                             ENDIF
     010466
2625 010466                             IF R1 NE #BUFF1+25 THEN
     010466  020127  063263                                                 $50335:
     010472  001402                                                                CMP     R1,#BUFF1+25
2626 010474                                 CALL ERROR                                 BEQ     $50336
     010474  004767  047200                                                         JSR     PC,ERROR
2627 010500                             ENDIF                                  $50336:
     010500
2628 010500                             IF R2 NE #0 THEN                               TST     R2
     010500  005702                                                                   BEQ     $50337
     010502  001402
2629 010504                                 CALL ERROR                                 JSR     PC,ERROR
     010504  004767  047170
2630 010510                             ENDIF                                  $50337:
     010510
2631 010510                             IF R3 NE #0 THEN                               TST     R3
     010510  005703                                                                   BEQ     $50340
     010512  001402
2632 010514                                 CALL ERROR                                 JSR     PC,ERROR
     010514  004767  047160
2633 010520                             ENDIF                                  $50340:
     010520
2634 010520                             IF R4 NE #0 THEN                               TST     R4
     010520  005704                                                                   BEQ     $50341
     010522  001402
2635 010524                                 CALL ERROR                                 JSR     PC,ERROR
     010524  004767  047150
2636 010530                             ENDIF                                  $50341:
     010530
2637 010530                             IF R5 NE #0 THEN                               TST     R5
     010530  005705                                                                   BEQ     $50342
     010532  001402
2638 010534                                 CALL ERROR                                 JSR     PC,ERROR
     010534  004767  047140
2639 010540                             ENDIF                                  $50342:
     010540
2640 010540                             INLINE  <JMP XSKPC>                            JMP     XSKPC
     010540  000167  000030
2641
2642                                ;+
2643                                ; TABLE OF TEST CASES FOR LOCC.
2644                                ;-
2645
2646 010544  000003                 TLOCC:  .WORD   3               ;# OF TEST CASES FOR LOCC
2647
2648 010546  000000                         .WORD   0                       ;SRC.LEN
2649 010550  000101                         .WORD   101                     ;CHAR.SEARCHED FOR
2650 010552  000041                         .WORD   41                      ;OFFSET
2651
2652 010554  000060                         .WORD   60              ;TEST CASE NUMBER 2
2653 010556  000040                         .WORD   40
2654 010560  000026                         .WORD   26
2655
2656 010562  000034                         .WORD   34              ;TEST CASE NUMBER 3
2657 010564  000077                         .WORD   77
2658 010566  000040                         .WORD   40
```

LOCC

```
   2659
   2660
   2661 010570  000100                TLOCCI: .WORD   BUFLEN          ;TEST CASE FOR INLINE
   2662 010572  063236                        .WORD   BUFF1
   2663
```

```
 2665                           .SBTTL  SKPC
 2666                           ;;*****************************************************************
 2667                           ;
 2668                           ;  SKPC TEST
 2669                           ;
 2670                           ;;*****************************************************************
 2671
 2672 010574                   ROUTINE XSKPC                                                  XSKPC:
      010574
 2673 010574                      INLINE  <SCOPE>                                                 SCOPE
      010574  000004
 2674 010576                      INLINE  <NOP>                                                   NOP
      010576  000240
 2675 010600                      LET $TIMES := #20          ;ITERATION COUNT                     MOV     #20,$TIMES
      010600  012767  000020  167576
 2676 010606                      LET COUNT := TSKPC         ;GET TEST COUNT                      MOV     TSKPC,COUNT
      010606  016767  000624  167654
 2677 010614                      LET TSTCAS := #TSKPC+2             ;POINT TO FIRST TEST CASE
      010614  012767  011440  167722                                                             MOV     #TSKPC+2,TSTCAS
 2678                           ;REPEAT
 2679 010622                      INLINE  <$5041:>                                                $5041:
      010622
 2680
 2681                           ;+
 2682                           ; GET TEST DATA FROM TABLE
 2683                           ;-
 2684
 2685 010622                      LET R0 := TSTCAS                                              MOV     TSTCAS,R0
      010622  016700  167716
 2686 010626                      LET DATA00 := (R0)+        ;SRC.LEN                           MOV     (R0)+,DATA00
      010626  012067  167730
 2687 010632                      LET DATA01 := (R0)+        ;NON-SEARCH CHAR                   MOV     (R0)+,DATA01
      010632  012067  167726
 2688 010636                      LET DATA02 := (R0)         ;OFFSET                            MOV     (R0),DATA02
      010636  011067  167724
 2689
 2690                           ;+
 2691                           ; SET UP GPR'S
 2692                           ; EXPECTED GPR'S AND CONDITION CODES,
 2693                           ; AND SOURCE BUFFER.
 2694                           ;-
 2695
 2696 010642                      LET R0 := DATA00                                              MOV     DATA00,R0
      010642  016700  167714
 2697 010646                      LET R1 := #BUFF1                                              MOV     #BUFF1,R1
      010646  012701  063236
 2698 010652                      LET R4 := #377                                                MOV     #377,R4
      010652  012704  000377
 2699 010656                      INLINE  <BIT    #BIT00,COUNT>                                 BIT     #BIT00,COUNT
      010656  032767  000001  167604
 2700 010664                      IFCOND EQ THEN                                                BNE     $50345
      010664  001004
 2701 010666                          LET R2 := #0                                              CLR     R2
      010666  005002
 2702 010670                          LET R3 := #0                                              CLR     R3
      010670  005003
 2703 010672                          LET R5 := #0
```

```
                010672  005005
 2704           010674
                010674  000406
                010676
 2705           010676
                010676  012702  177777
 2706           010702
                010702  012703  177777
 2707           010706
                010706  012705  177777
 2708           010712
                010712
 2709           010712
                010712  026767  167650  167642
                010720  101411
 2710           010722
                010722  005067  167576
 2711           010726
                010726  012767  063236  167572
                010734  066767  167622  167564
 2712           010742
                010742  000414
                010744
 2713           010744
                010744  016767  167612  167552
                010752  166767  167610  167544
 2714           010760
                010760  012767  063236  167540
                010766  066767  167574  167532
 2715           010774
                010774
 2716           010774
                010774  010267  167530
 2717           011000
                011000  010367  167526
 2718           011004
                011004  010467  167524
 2719           011010
                011010  010567  167522
 2720           011014
                011014  005767  167504
 2721           011020
                011020  016767  166752  167474
 2722           011026
                011026  042767  177760  167466
 2723           011034
                011034  012767  063236  167500
 2724           011042
                011042  004767  045452
 2725           011046
                011046  012767  063236  167422
                011054  066767  167506  167414
 2726           011062
                011062  116777  167476  167406
 2727
 2728
 2729
```

```
ELSE

    LET R2 := #177777

    LET R3 := #177777

    LET R5 := #177777

ENDIF

IF DATA02 HI DATA00 THEN

    LET R0.EXP := #0

    LET R1.EXP := #BUFF1 + DATA00

ELSE

    LET R0.EXP := DATA00 - DATA02

    LET R1.EXP := #BUFF1 + DATA02

ENDIF

LET R2.EXP := R2

LET R3.EXP := R3

LET R4.EXP := R4

LET R5.EXP := R5

INLINE  <TST R0.EXP>

LET CC.EXP := PSW

LET CC.EXP := CC.EXP CLR.BY #177760

LET BFADRS := #BUFF1

CALL SETBUF             ;FILL BUFFER WITH SEARCH CHAR

LET TEMP0 := #BUFF1 + DATA02

LET @TEMP0 :B= DATA01

;+
; DO THE CIS INSTRUCTION
```

```
                CLR     R5
                BR      $50346
$50345:
                MOV     #177777,R2
                MOV     #177777,R3
                MOV     #177777,R5
$50346:
                CMP     DATA02,DATA00
                BLOS    $50347
                CLR     R0.EXP
                MOV     #BUFF1,R1.EXP
                ADD     DATA00,R1.EXP
                BR      $50350
$50347:
                MOV     DATA00,R0.EXP
                SUB     DATA02,R0.EXP
                MOV     #BUFF1,R1.EXP
                ADD     DATA02,R1.EXP
$50350:
                MOV     R2,R2.EXP
                MOV     R3,R3.EXP
                MOV     R4,R4.EXP
                MOV     R5,R5.EXP
                TST     R0.EXP
                MOV     PSW,CC.EXP
                BIC     #177760,CC.EXP
                MOV     #BUFF1,BFADRS
                JSR     PC,SETBUF
                MOV     #BUFF1,TEMP0
                ADD     DATA02,TEMP0
                MOVB    DATA01,@TEMP0
```

```
 2730                                   ;-
 2731
 2732 011070                               INLINE  <SKPC>
      011070  076041                                                                SKPC
 2733
 2734                                   ;+
 2735                                   ; CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS
 2736                                   ;-
 2737
 2738 011072                               LET CC := PSW
      011072  016767  166700  167420                                                MOV     PSW,CC
 2739 011100                               LET CC := CC CLR.BY #177760
      011100  042767  177760  167412                                                BIC     #177760,CC
 2740 011106                               IF CC NE CC.EXP THEN
      011106  026767  167406  167406                                                CMP     CC,CC.EXP
      011114  001402                                                               BEQ     $50351
 2741 011116                                   CALL ERROR
      011116  004767  046556                                                       JSR     PC,ERROR
 2742 011122                               ENDIF
      011122                                                                  $50351:
 2743 011122                               IF R0 NE R0.EXP THEN
      011122  020067  167376                                                       CMP     R0,R0.EXP
      011126  001402                                                               BEQ     $50352
 2744 011130                                   CALL ERROR
      011130  004767  046544                                                       JSR     PC,ERROR
 2745 011134                               ENDIF
      011134                                                                  $50352:
 2746 011134                               IF R1 NE R1.EXP THEN
      011134  020167  167366                                                       CMP     R1,R1.EXP
      011140  001402                                                               BEQ     $50353
 2747 011142                                   CALL ERROR
      011142  004767  046532                                                       JSR     PC,ERROR
 2748 011146                               ENDIF
      011146                                                                  $50353:
 2749 011146                               IF R2 NE R2.EXP THEN
      011146  020267  167356                                                       CMP     R2,R2.EXP
      011152  001402                                                               BEQ     $50354
 2750 011154                                   CALL ERROR
      011154  004767  046520                                                       JSR     PC,ERROR
 2751 011160                               ENDIF
      011160                                                                  $50354:
 2752 011160                               IF R3 NE R3.EXP THEN
      011160  020367  167346                                                       CMP     R3,R3.EXP
      011164  001402                                                               BEQ     $50355
 2753 011166                                   CALL ERROR
      011166  004767  046506                                                       JSR     PC,ERROR
 2754 011172                               ENDIF
      011172                                                                  $50355:
 2755 011172                               IF R4 NE R4.EXP THEN
      011172  020467  167336                                                   -   CMP     R4,R4.EXP
      011176  001402                                                               BEQ     $50356
 2756 011200                                   CALL ERROR
      011200  004767  046474                                                       JSR     PC,ERROR
 2757 011204                               ENDIF
      011204                                                                  $50356:
 2758 011204                               IF R5 NE R5.EXP THEN
      011204  020567  167326                                                       CMP     R5,R5.EXP
```

```
      011210  001402                                                                BEQ     $50357
 2759 011212                          CALL ERROR
      011212  004767  046462                                                       JSR     PC,ERROR
 2760 011216                          ENDIF
      011216                                                                $50357:
 2761 011216                          LET TSTCAS := TSTCAS + #6
      011216  062767  000006  167320                                               ADD     #6,TSTCAS
 2762 011224                          LET COUNT := COUNT - #1
      011224  005367  167240                                                       DEC     COUNT
 2763                                ;UNTIL COUNT EQ #0
 2764 011230                          IF COUNT NE #0 THEN
      011230  005767  167234                                                       TST     COUNT
      011234  001402                                                               BEQ     $50360
 2765 011236                              INLINE  <JMP     $5041>
      011236  000167  177360                                                       JMP     $5041
 2766 011242                          ENDIF
      011242                                                                $50360:
 2767
 2768
 2769                                ;+
 2770                                ;DO ONE INLINE CASE - SKPCI
 2771                                ;-
 2772
 2773 011242                          LET BFADRS := #BUFF1
      011242  012767  063236  167272                                               MOV     #BUFF1,BFADRS
 2774 011250                          CALL SETBUF
      011250  004767  045244                                                       JSR     PC,SETBUF
 2775 011254                          CALL CLRREG
      011254  004767  045160                                                       JSR     PC,CLRREG
 2776 011260                          LET TEMPO := #BUFF1 + #25
      011260  012767  063236  167210                                               MOV     #BUFF1,TEMPO
      011266  062767  000025  167202                                               ADD     #25,TEMPO
 2777 011274                          LET @TEMPO :B= #40
      011274  112777  000040  167174                                               MOVB    #40,@TEMPO
 2778 011302                          INLINE  <SKPCI>
      011302  076141                                                               SKPCI
 2779 011304                          INLINE  <.WORD       TSKPCI>
      011304  011454                                                               .WORD   TSKPCI
 2780 011306                          INLINE  <.WORD       377>
      011306  000377                                                               .WORD   377
 2781 011310                          LET TEMPO := PSW
      011310  016767  166462  167160                                               MOV     PSW,TEMPO
 2782 011316                          INLINE  <BIT #17,TEMPO>
      011316  032767  000017  167152                                               BIT     #17,TEMPO
 2783 011324                          IFCOND NE THEN
      011324  001402                                                               BEQ     $50361
 2784 011326                              CALL ERROR
      011326  004767  046346                                                       JSR     PC,ERROR
 2785 011332                          ENDIF
      011332                                                                $50361:
 2786 011332                          LET TEMPO := #BUFLEN - #25
      011332  012767  000100  167136                                               MOV     #BUFLEN,TEMPO
      011340  162767  000025  167130                                               SUB     #25,TEMPO
 2787 011346                          IF R0 NE TEMPO THEN
      011346  020067  167124                                                       CMP     R0,TEMPO
      011352  001402                                                               BEQ     $50362
 2788 011354                              CALL ERROR
```

```
                011354  004767  046320                                                          JSR     PC,ERROR
2789            011360                          ENDIF
                011360                                                                  $50362:
2790            011360                          IF R1 NE #BUFF1+25 THEN
                011360  020127  063263                                                          CMP     R1,#BUFF1+25
                011364  001402                                                                  BEQ     $50363
2791            011366                              CALL ERROR
                011366  004767  046306                                                          JSR     PC,ERROR
2792            011372                          ENDIF
                011372                                                                  $50363:
2793            011372                          IF R2 NE #0 THEN
                011372  005702                                                                  TST     R2
                011374  001402                                                                  BEQ     $50364
2794            011376                              CALL ERROR
                011376  004767  046276                                                          JSR     PC,ERROR
2795            011402                          ENDIF
                011402                                                                  $50364:
2796            011402                          IF R3 NE #0 THEN
                011402  005703                                                                  TST     R3
                011404  001402                                                                  BEQ     $50365
2797            011406                              CALL ERROR
                011406  004767  046266                                                          JSR     PC,ERROR
2798            011412                          ENDIF
                011412                                                                  $50365:
2799            011412                          IF R4 NE #0 THEN
                011412  005704                                                                  TST     R4
                011414  001402                                                                  BEQ     $50366
2800            011416                              CALL ERROR
                011416  004767  046256                                                          JSR     PC,ERROR
2801            011422                          ENDIF
                011422                                                                  $50366:
2802            011422                          IF R5 NE #0 THEN
                011422  005705                                                                  TST     R5
                011424  001402                                                                  BEQ     $50367
2803            011426                              CALL ERROR
                011426  004767  046246                                                          JSR     PC,ERROR
2804            011432                          ENDIF
                011432                                                                  $50367:
2805            011432                          INLINE  <JMP        XSCANC>
                011432  000167  000022                                                          JMP     XSCANC
2806
2807                                    ;+
2808                                    ; TABLE OF TEST CASES FOR SKPC.
2809                                    ;-
2810
2811 011436  000002                     TSKPC:  .WORD   2               ;# OF TEST CASES FOR SKPC
2812
2813 011440  000040                             .WORD   40                      ;SRC.LEN
2814 011442  000100                             .WORD   100                     ;NON-SEARCH CHAR.
2815 011444  000060                             .WORD   60                      ;OFFSET
2816
2817 011446  000050                             .WORD   50              ;TEST CASE NUMBER 2
2818 011450  000077                             .WORD   77
2819 011452  000030                             .WORD   30
2820
2821
2822 011454  000100                     TSKPCI: .WORD   BUFLEN          ;TEST CASE FOR INLINE
```

SKPC

```
2823 011456  063236                          .WORD   BUFF1
2824
```

```
 2826                                .SBTTL  SCANC
 2827                                ;;*****************************************************************
 2828                                ;
 2829                                ;  SCANC TEST
 2830                                ;
 2831                                ;;*****************************************************************
 2832
 2833 011460                        ROUTINE XSCANC
      011460                                                                          XSCANC:
 2834 011460                           INLINE  <SCOPE>
      011460  000004                                                                 SCOPE
 2835 011462                           INLINE  <NOP>
      011462  000240                                                                 NOP
 2836 011464                           LET $TIMES := #20          ;ITERATION COUNT
      011464  012767  000020  166712                                                 MOV     #20,$TIMES
 2837 011472                           LET COUNT := TSCANC        ;GET TEST COUNT
      011472  016767  000710  166770                                                 MOV     TSCANC,COUNT
 2838 011500                           LET TSTCAS := #TSCANC+2            ;POINT TO FIRST TEST CASE
      011500  012767  012410  167036                                                 MOV     #TSCANC+2,TSTCAS
 2839                                ;REPEAT
 2840 011506                           INLINE  <$5042:>
      011506                                                                         $5042:
 2841
 2842                                ;+
 2843                                ; GET TEST DATA FROM TABLE
 2844                                ;-
 2845
 2846 011506                           LET R0 := TSTCAS
      011506  016700  167032                                                         MOV     TSTCAS,R0
 2847 011512                           LET DATA00 := (R0)+        ;SRC.LEN
      011512  012067  167044                                                         MOV     (R0)+,DATA00
 2848 011516                           LET DATA01 := (R0)         ;OFFSET
      011516  011067  167042                                                         MOV     (R0),DATA01
 2849
 2850                                ;+
 2851                                ; SET UP GPR'S,
 2852                                ; EXPECTED GPR'S AND CONDITION CODES
 2853                                ; SOURCE BUFFER, AND CHARACTER SET TABLE.
 2854                                ;-
 2855
 2856 011522                           LET R0 := DATA00
      011522  016700  167034                                                         MOV     DATA00,R0
 2857 011526                           LET R1 := #BUFF1
      011526  012701  063236                                                         MOV     #BUFF1,R1
 2858 011532                           LET R4 := #1               ;MASK
      011532  012704  000001                                                         MOV     #1,R4
 2859 011536                           LET R5 := #CHRSET
      011536  012705  064036                                                         MOV     #CHRSET,R5
 2860 011542                           INLINE  <BIT    #BIT00,COUNT>
      011542  032767  000001  166720                                                 BIT     #BIT00,COUNT
 2861 011550                           IFCOND EQ THEN
      011550  001003                                                                 BNE     $50372
 2862 011552                               LET R2 := #0
      011552  005002                                                                 CLR     R2
 2863 011554                               LET R3 := #0
      011554  005003                                                                 CLR     R3
 2864 011556                           ELSE
```

```
        011556  000404                                                          BR      $50373
        011560                                                          $50372:
 2865   011560                          LET R2 := #177777                       MOV     #177777,R2
        011560  012702  177777
 2866   011564                          LET R3 := #177777                       MOV     #177777,R3
        011564  012703  177777
 2867   011570                      ENDIF
        011570                                                          $50373:
 2868   011570                      IF DATA01 HI DATA00 THEN                    CMP     DATA01,DATA00
        011570  026767  166770  166764                                          BLOS    $50374
        011576  101411
 2869   011600                          LET R0.EXP := #0                        CLR     R0.EXP
        011600  005067  166720
 2870   011604                          LET R1.EXP := #BUFF1 + DATA00           MOV     #BUFF1,R1.EXP
        011604  012767  063236  166714                                          ADD     DATA00,R1.EXP
        011612  066767  166744  166706
 2871   011620                      ELSE                                        BR      $50375
        011620  000414                                                  $50374:
        011622
 2872   011622                          LET R0.EXP := DATA00 - DATA01           MOV     DATA00,R0.EXP
        011622  016767  166734  166674                                          SUB     DATA01,R0.EXP
        011630  166767  166730  166666
 2873   011636                          LET R1.EXP := #BUFF1 + DATA01           MOV     #BUFF1,R1.EXP
        011636  012767  063236  166662                                          ADD     DATA01,R1.EXP
        011644  066767  166714  166654
 2874   011652                      ENDIF                                   $50375:
        011652
 2875   011652                      LET R2.EXP := R2                            MOV     R2,R2.EXP
        011652  010267  166652
 2876   011656                      LET R3.EXP := R3                            MOV     R3,R3.EXP
        011656  010367  166650
 2877   011662                      LET R4.EXP := R4                            MOV     R4,R4.EXP
        011662  010467  166646
 2878   011666                      LET R5.EXP := R5                            MOV     R5,R5.EXP
        011666  010567  166644
 2879   011672                      INLINE  <TST   R0.EXP>                      TST   R0.EXP
        011672  005767  166626
 2880   011676                      LET CC.EXP := PSW                           MOV     PSW,CC.EXP
        011676  016767  166074  166616
 2881   011704                      LET CC.EXP := CC.EXP CLR.BY #177760         BIC     #177760,CC.EXP
        011704  042767  177760  166610
 2882   011712                      LET BFADRS := #BUFF1                        MOV     #BUFF1,BFADRS
        011712  012767  063236  166622
 2883   011720                      CALL CLRBUF                                 JSR     PC,CLRBUF
        011720  004767  044532
 2884   011724                      LET TEMP0 := #BUFF1 + DATA01                MOV     #BUFF1,TEMP0
        011724  012767  063236  166544                                          ADD     DATA01,TEMP0
        011732  066767  166626  166536
 2885   011740                      LET @TEMP0 :B= #125                         MOVB    #125,@TEMP0
        011740  112777  000125  166530
 2886   011746                      WHILE R5 LO #CHRSET+400 DO                  
        011746                                                          $50376:
        011746  020527  064436                                                  CMP     R5,#CHRSET+400
        011752  103002                                                          BHIS    $50377
 2887   011754                          INLINE  <CLR     (R5)+>         ;CLEAR CHAR SET TABLE
        011754  005025                                                          CLR     (R5)+
 2888   011756                      ENDDO
```

```
        011756  000773                                                                  BR      $50376
        011760                                                                  $50377:
 2889   011760                  LET R5 := #CHRSET                       ;RESTORE R5
        011760  012705  064036                                                          MOV     #CHRSET,R5
 2890   011764                  LET TEMPO := #CHRSET + #125
        011764  012767  064036  166504                                                  MOV     #CHRSET,TEMPO
        011772  062767  000125  166476                                                  ADD     #125,TEMPO
 2891   012000                  LET @TEMPO :B= #1                        ;SET MASK IN CHAR SET TABLE
        012000  112777  000001  166470                                                  MOVB    #1,@TEMPO
 2892
 2893                          ;+
 2894                          ; DO THE CIS INSTRUCTION
 2895                          ;-
 2896
 2897   012006                  INLINE  <SCANC>
        012006  076042                                                                  SCANC
 2898
 2899                          ;+
 2900                          ; CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS
 2901                          ;-
 2902
 2903   012010                  LET CC := PSW
        012010  016767  165762  166502                                                  MOV     PSW,CC
 2904   012016                  LET CC := CC CLR.BY #177760
        012016  042767  177760  166474                                                  BIC     #177760,CC
 2905   012024                  IF CC NE CC.EXP THEN
        012024  026767  166470  166470                                                  CMP     CC,CC.EXP
        012032  001402                                                                  BEQ     $50400
 2906   012034                      CALL ERROR
        012034  004767  045640                                                          JSR     PC,ERROR
 2907   012040                  ENDIF
        012040                                                                  $50400:
 2908   012040                  IF R0 NE R0.EXP THEN
        012040  020067  166460                                                          CMP     R0,R0.EXP
        012044  001402                                                                  BEQ     $50401
 2909   012046                      CALL ERROR
        012046  004767  045626                                                          JSR     PC,ERROR
 2910   012052                  ENDIF
        012052                                                                  $50401:
 2911   012052                  IF R1 NE R1.EXP THEN
        012052  020167  166450                                                          CMP     R1,R1.EXP
        012056  001402                                                                  BEQ     $50402
 2912   012060                      CALL ERROR
        012060  004767  045614                                                          JSR     PC,ERROR
 2913   012064                  ENDIF
        012064                                                                  $50402:
 2914   012064                  IF R2 NE R2.EXP THEN
        012064  020267  166440                                                          CMP     R2,R2.EXP
        012070  001402                                                                  BEQ     $50403
 2915   012072                      CALL ERROR
        012072  004767  045602                                                          JSR     PC,ERROR
 2916   012076                  ENDIF
        012076                                                                  $50403:
 2917   012076                  IF R3 NE R3.EXP THEN
        012076  020367  166430                                                          CMP     R3,R3.EXP
        012102  001402                                                                  BEQ     $50404
 2918   012104                      CALL ERROR
```

```
      012104  004767  045570                                                                  JSR     PC,ERROR
 2919 012110                          ENDIF
      012110                                                                          $50404:
 2920 012110                          IF R4 NE R4.EXP THEN
      012110  020467  166420                                                                  CMP     R4,R4.EXP
      012114  001402                                                                         BEQ     $50405
 2921 012116                              CALL ERROR
      012116  004767  045556                                                                 JSR     PC,ERROR
 2922 012122                          ENDIF
      012122                                                                          $50405:
 2923 012122                          IF R5 NE R5.EXP THEN
      012122  020567  166410                                                                  CMP     R5,R5.EXP
      012126  001402                                                                         BEQ     $50406
 2924 012130                              CALL ERROR
      012130  004767  045544                                                                 JSR     PC,ERROR
 2925 012134                          ENDIF
      012134                                                                          $50406:
 2926 012134                          LET TSTCAS := TSTCAS + #4
      012134  062767  000004  166402                                                         ADD     #4,TSTCAS
 2927 012142                          LET COUNT := COUNT - #1
      012142  005367  166322                                                                 DEC     COUNT
 2928                                ;UNTIL COUNT EQ #0
 2929 012146                          IF COUNT NE #0 THEN
      012146  005767  166316                                                                 TST     COUNT
      012152  001402                                                                         BEQ     $50407
 2930 012154                              INLINE  <JMP     $5042>
      012154  000167  177326                                                                 JMP     $5042
 2931 012160                          ENDIF
      012160                                                                          $50407:
 2932
 2933
 2934                                ;+
 2935                                ; DO ONE INLINE CASE - SCANCI.
 2936                                ;-
 2937
 2938 012160                          LET BFADRS := #BUFF1
      012160  012767  063236  166354                                                         MOV     #BUFF1,BFADRS
 2939 012166                          CALL CLRBUF
      012166  004767  044264                                                                 JSR     PC,CLRBUF
 2940 012172                          LET TEMPO := #BUFF1 + #50
      012172  012767  063236  166276                                                         MOV     #BUFF1,TEMPO
      012200  062767  000050  166270                                                         ADD     #50,TEMPO
 2941 012206                          LET @TEMPO :B= #27
      012206  112777  000027  166262                                                         MOVB    #27,@TEMPO
 2942 012214                          LET R0 := #CHRSET
      012214  012700  064036                                                                 MOV     #CHRSET,R0
 2943 012220                          WHILE R0 LO #CHRSET+400 DO
      012220                                                                          $50410:
      012220  020027  064436                                                                 CMP     R0,#CHRSET+400
      012224  103002                                                                         BHIS    $50411
 2944 012226                              INLINE  <CLR     (R0)+>          ;CLEAR CHAR.SET TABLE
      012226  005020                                                                         CLR     (R0)+
 2945 012230                          ENDDO
      012230  000773                                                                         BR      $50410
      012232                                                                          $50411:
 2946 012232                          LET TEMPO := #CHRSET + #27
      012232  012767  064036  166236                                                         MOV     #CHRSET,TEMPO
```

```
                012240  062767  000027  166230
2947 012246                             LET @TEMPO :B= #10
                012246  112777  000010  166222
2948 012254                             CALL CLRREG
                012254  004767  044160
2949 012260                             INLINE  <SCANCI>
                012260  076142
2950 012262                             INLINE  <.WORD       TSCNCI>
                012262  012424
2951 012264                             INLINE  <.WORD       TSCNCI+4>
                012264  012430
2952 012266                             LET TEMPO := PSW
                012266  016767  165504  166202
2953 012274                             INLINE  <BIT #17,TEMPO>
                012274  032767  000017  166174
2954 012302                             IFCOND NE THEN
                012302  001402
2955 012304                                 CALL ERROR
                012304  004767  045370
2956 012310                             ENDIF
                012310
2957 012310                             LET TEMPO := #BUFLEN-50
                012310  012767  000030  166160
2958 012316                             IF R0 NE TEMPO THEN
                012316  020067  166154
                012322  001402
2959 012324                                 CALL ERROR
                012324  004767  045350
2960 012330                             ENDIF
                012330
2961 012330                             IF R1 NE #BUFF1+50 THEN
                012330  020127  063306
                012334  001402
2962 012336                                 CALL ERROR
                012336  004767  045336
2963 012342                             ENDIF
                012342
2964 012342                             IF R2 NE #0 THEN
                012342  005702
                012344  001402
2965 012346                                 CALL ERROR
                012346  004767  045326
2966 012352                             ENDIF
                012352
2967 012352                             IF R3 NE #0 THEN
                012352  005703
                012354  001402
2968 012356                                 CALL ERROR
                012356  004767  045316
2969 012362                             ENDIF
                012362
2970 012362                             IF R4 NE #0 THEN
                012362  005704
                012364  001402
2971 012366                                 CALL ERROR
                012366  004767  045306
2972 012372                             ENDIF
```

```
                        ADD     #27,TEMPO
                                ;SET MASK IN CHAR SET TABLE
                        MOVB    #10,@TEMPO
                        JSR     PC,CLRREG
                        SCANCI
                        .WORD   TSCNCI
                        .WORD   TSCNCI+4
                        MOV     PSW,TEMPO
                        BIT     #17,TEMPO
                        BEQ     $50412
                        JSR     PC,ERROR
                $50412:
                        MOV     #BUFLEN-50,TEMPO
                        CMP     R0,TEMPO
                        BEQ     $50413
                        JSR     PC,ERROR
                $50413:
                        CMP     R1,#BUFF1+50
                        BEQ     $50414
                        JSR     PC,ERROR
                $50414:
                        TST     R2
                        BEQ     $50415
                        JSR     PC,ERROR
                $50415:
                        TST     R3
                        BEQ     $50416
                        JSR     PC,ERROR
                $50416:
                        TST     R4
                        BEQ     $50417
                        JSR     PC,ERROR
```

```
     012372                                                                      $50417:
     012372                              IF R5 NE #0 THEN
     012372  005705                                                                       TST     R5
     012374  001402                                                                       BEQ     $50420
     012376                                   CALL ERROR
     012376  004767  045276                                                               JSR     PC,ERROR
     012402                              ENDIF
     012402                                                                      $50420:
     012402                              INLINE  <JMP XSPANC>
     012402  000167  000026                                                               JMP     XSPANC


                                     ;+
                                     ; TABLE OF TEST CASES FOR SCANC.
                                     ;-

     012406  000003                  TSCANC: .WORD   3               ;#OF TEST CASES FOR SCANC

     012410  000000                          .WORD   0                       ;SRC.LEN
     012412  000020                          .WORD   20                      ;OFFSET

     012414  000100                          .WORD   BUFLEN          ;TEST CASE NUMBER 2
     012416  000030                          .WORD   30

     012420  000100                          .WORD   BUFLEN          ;TEST CASE NUMBER 3
     012422  000110                          .WORD   BUFLEN+10
                                                  .

     012424  000100                  TSCNCI: .WORD   BUFLEN          ;TEST CASE FOR INLINE
     012426  063236                          .WORD   BUFF1
     012430  000010                          .WORD   10
     012432  064036                          .WORD   CHRSET
```

```
3000                                    .SBTTL  SPANC
3001                                    ;;*****************************************************************
3002                                    ;
3003                                    ;  SPANC TEST
3004                                    ;
3005                                    ;;*****************************************************************
3006
3007 012434                             ROUTINE XSPANC
     012434                                                                          XSPANC:
3008 012434                                INLINE  <SCOPE>
     012434  000004                                                                     SCOPE
3009 012436                                INLINE  <NOP>
     012436  000240                                                                    NOP
3010 012440                                LET $TIMES := #20           ;ITERATION COUNT
     012440  012767  000020  165736                                                    MOV     #20,$TIMES
3011 012446                                LET COUNT := TSPANC         ;GET TEST COUNT
     012446  016767  000660  166014                                                    MOV     TSPANC,COUNT
3012 012454                                LET TSTCAS := #TSPANC+2             ;POINT TO FIRST TEST CASE
     012454  012767  013334  166062                                                    MOV     #TSPANC+2,TSTCAS
3013                                    ;REPEAT
3014 012462                                INLINE  <$5043:>
     012462                                                                          $5043:
3015
3016                                    ;+
3017                                    ; GET TEST DATA FROM TABLE
3018                                    ;-
3019
3020 012462                                LET R0 := TSTCAS
     012462  016700  166056                                                        MOV     TSTCAS,R0
3021 012466                                LET DATA00 := (R0)+         ;SRC.LEN
     012466  012067  166070                                                        MOV     (R0)+,DATA00
3022 012472                                LET DATA01 := (R0)          ;OFFSET
     012472  011067  166066                                                        MOV-    (R0),DATA01
3023
3024                                    ;+
3025                                    ; SET UP GPR'S,
3026                                    ; EXPECTED GPR'S AND CONDITION CODES,
3027                                    ; SOURCE BUFFER, AND CHARACTER SET TABLE.
3028                                    ;-
3029
3030 012476                                LET R0 := DATA00
     012476  016700  166060                                                        MOV     DATA00,R0
3031 012502                                LET R1 := #BUFF1
     012502  012701  063236                                                        MOV     #BUFF1,R1
3032 012506                                LET R4 := #40                       ;MASK
     012506  012704  000040                                                        MOV     #40,R4
3033 012512                                LET R5 := #CHRSET
     012512  012705  064036                                                        MOV     #CHRSET,R5
3034 012516                                INLINE  <BIT #BIT00,COUNT>
     012516  032767  000001  165744                                                BIT     #BIT00,COUNT
3035 012524                                IFCOND EQ THEN
     012524  001003                                                                BNE     $50423
3036 012526                                    LET R2 := #0
     012526  005002                                                                CLR     R2
3037 012530                                    LET R3 := #0
     012530  005003                                                                CLR     R3
3038 012532                                ELSE
```

```
        012532  000404                                                          BR      $50424
        012534                                                          $50423:
 3039   012534                          LET R2 := #177777                       MOV     #177777,R2
        012534  012702  177777
 3040   012540                          LET R3 := #177777                       MOV     #177777,R3
        012540  012703  177777
 3041   012544                      ENDIF
        012544                                                          $50424:
 3042   012544                      IF DATA01 HI DATA00 THEN                    CMP     DATA01,DATA00
        012544  026767  166014  166010                                        BLOS    $50425
        012552  101411
 3043   012554                          LET R0.EXP := #0                        CLR     R0.EXP
        012554  005067  165744
 3044   012560                          LET R1.EXP := #BUFF1 + DATA00           MOV     #BUFF1,R1.EXP
        012560  012767  063236  165740                                        ADD     DATA00,R1.EXP
        012566  066767  165770  165732
 3045   012574                      ELSE                                        BR      $50426
        012574  000414                                                  $50425:
        012576
 3046   012576                          LET R0.EXP := DATA00 - DATA01           MOV     DATA00,R0.EXP
        012576  016767  165760  165720                                        SUB     DATA01,R0.EXP
        012604  166767  165754  165712
 3047   012612                          LET R1.EXP := #BUFF1 + DATA01           MOV     #BUFF1,R1.EXP
        012612  012767  063236  165706                                        ADD     DATA01,R1.EXP
        012620  066767  165740  165700
 3048   012626                      ENDIF                                   $50426:
        012626
 3049   012626                      LET R2.EXP := R2                            MOV     R2,R2.EXP
        012626  010267  165676
 3050   012632                      LET R3.EXP := R3                            MOV     R3,R3.EXP
        012632  010367  165674
 3051   012636                      LET R4.EXP := R4                            MOV     R4,R4.EXP
        012636  010467  165672
 3052   012642                      LET R5.EXP := R5                            MOV     R5,R5.EXP
        012642  010567  165670
 3053   012646                      INLINE  <TST R0.EXP>                        TST     R0.EXP
        012646  005767  165652
 3054   012652                      LET CC.EXP := PSW                           MOV     PSW,CC.EXP
        012652  016767  165120  165642
 3055   012660                      LET CC.EXP := CC.EXP CLR.BY #177760         BIC     #177760,CC.EXP
        012660  042767  177760  165634
 3056   012666                      LET BFADRS := #BUFF1                        MOV     #BUFF1,BFADRS
        012666  012767  063236  165646
 3057   012674                      CALL CLRBUF                                 JSR     PC,CLRBUF
        012674  004767  043556
 3058   012700                      LET TEMP0 := #BUFF1 + DATA01                MOV     #BUFF1,TEMP0
        012700  012767  063236  165570                                        ADD     DATA01,TEMP0
        012706  066767  165652  165562
 3059   012714                      LET @TEMP0 :B= #252                         MOVB    #252,@TEMP0
        012714  112777  000252  165554
 3060   012722                      WHILE R5 LO #CHRSET+400 DO          $50427:
        012722                                                          CMP     R5,#CHRSET+400
        012722  020527  064436                                          BHIS    $50430
        012726  103002
 3061   012730                          INLINE  <CLR     (R5)+>         ;CLEAR CHAR. SET TABLE
        012730  005025                                                  CLR     (R5)+
 3062   012732                      ENDDO
```

```
        012732  000773                                                                  BR      $50427
        012734                                                                  $50430:
 3063   012734                  LET R5 := #CHRSET               ;RESTORE R5
        012734  012705  064036                                                          MOV     #CHRSET,R5
 3064   012740                  LET CHRSET :B= #40              ;SET MASK IN CHAR. SET
        012740  112767  000040  051070                                                 MOVB    #40,CHRSET
 3065
 3066                           ;+
 3067                           ; DO THE CIS INSTRUCTION
 3068                           ;-
 3069
 3070   012746                  INLINE  <SPANC>
        012746  076043                                                                 SPANC
 3071
 3072                           ;+
 3073                           ; CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS
 3074                           ;-
 3075
 3076   012750                  LET CC := PSW
        012750  016767  165022  165542                                                 MOV     PSW,CC
 3077   012756                  LET CC := CC CLR.BY #177760
        012756  042767  177760  165534                                                 BIC     #177760,CC
 3078   012764                  IF CC NE CC.EXP THEN
        012764  026767  165530  165530                                                 CMP     CC,CC.EXP
        012772  001402                                                                 BEQ     $50431
 3079   012774                      CALL ERROR
        012774  004767  044700                                                         JSR     PC,ERROR
 3080   013000                  ENDIF
        013000                                                                  $50431:
 3081   013000                  IF R0 NE R0.EXP THEN
        013000  020067  165520                                                          CMP     R0,R0.EXP
        013004  001402                                                                  BEQ     $50432
 3082   013006                      CALL ERROR
        013006  004767  044666                                                          JSR     PC,ERROR
 3083   013012                  ENDIF
        013012                                                                  $50432:
 3084   013012                  IF R1 NE R1.EXP THEN
        013012  020167  165510                                                          CMP     R1,R1.EXP
        013016  001402                                                                  BEQ     $50433
 3085   013020                      CALL ERROR
        013020  004767  044654                                                          JSR     PC,ERROR
 3086   013024                  ENDIF
        013024                                                                  $50433:
 3087   013024                  IF R2 NE R2.EXP THEN
        013024  020267  165500                                                          CMP     R2,R2.EXP
        013030  001402                                                                  BEQ     $50434
 3088   013032                      CALL ERROR
        013032  004767  044642                                                          JSR     PC,ERROR
 3089   013036                  ENDIF
        013036                                                                  $50434:
 3090   013036                  IF R3 NE R3.EXP THEN
        013036  020367  165470                                                          CMP     R3,R3.EXP
        013042  001402                                                                  BEQ     $50435
 3091   013044                      CALL ERROR
        013044  004767  044630                                                          JSR     PC,ERROR
 3092   013050                  ENDIF
        013050                                                                  $50435:
```

```
3093 013050                             IF R4 NE R4.EXP THEN
     013050  020467  165460                                                 CMP     R4,R4.EXP
     013054  001402                                                         BEQ     $50436
3094 013056                                 CALL ERROR
     013056  004767  044616                                                 JSR     PC,ERROR
3095 013062                             ENDIF
     013062                                                             $50436:
3096 013062                             IF R5 NE R5.EXP THEN
     013062  020567  165450                                                 CMP     R5,R5.EXP
     013066  001402                                                         BEQ     $50437
3097 013070                                 CALL ERROR
     013070  004767  044604                                                 JSR     PC,ERROR
3098 013074                             ENDIF
     013074                                                             $50437:
3099 013074                             LET TSTCAS := TSTCAS + #4
     013074  062767  000004  165442                                         ADD     #4,TSTCAS
3100 013102                             LET COUNT := COUNT - #1
     013102  005367  165362                                                 DEC     COUNT
3101                                   ;UNTIL COUNT EQ #0
3102 013106                             IF COUNT NE #0 THEN
     013106  005767  165356                                                 TST     COUNT
     013112  001402                                                         BEQ     $50440
3103 013114                                 INLINE  <JMP     $5043>
     013114  000167  177342                                                 JMP     $5043
3104 013120                             ENDIF
     013120                                                             $50440:
3105
3106
3107                                   ;+
3108                                   ; DO ONE INLINE CASE - SPANCI
3109                                   ;-
3110
3111 013120                             LET BFADRS := #BUFF1
     013120  012767  063236  165414                                         MOV     #BUFF1,BFADRS
3112 013126                             CALL CLRBUF
     013126  004767  043324                                                 JSR     PC,CLRBUF
3113 013132                             LET TEMPO := #BUFF1 + #50
     013132  012767  063236  165336                                         MOV     #BUFF1,TEMPO
     013140  062767  000050  165330                                         ADD     #50,TEMPO
3114 013146                             LET @TEMPO :B= #27
     013146  112777  000027  165322                                         MOVB    #27,@TEMPO
3115 013154                             LET R0 := #CHRSET
     013154  012700  064036                                                 MOV     #CHRSET,R0
3116 013160                             WHILE R0 LO #CHRSET+400 DO
     013160                                                             $50441:
     013160  020027  064436                                                 CMP     R0,#CHRSET+400
     013164  103002                                                         BHIS    $50442
3117 013166                                 LET (R0)+ := #0          ;CLEAR CHAR. SET TABLE
     013166  005020                                                         CLR     (R0)+
3118 013170                             ENDDO
     013170  000773                                                         BR      $50441
     013172                                                             $50442:
3119 013172                             CALL CLRREG
     013172  004767  043242                                                 JSR     PC,CLRREG
3120 013176                             LET CHRSET :B= #10
     013176  112767  000010  050632                                         MOVB    #10,CHRSET
3121 013204                             INLINE  <SPANCI>
```

```
      013204  076143
 3122 013206                          INLINE  <.WORD       TSPNCI>
      013206  013350
 3123 013210                          INLINE  <.WORD       TSPNCI+4>
      013210  013354
 3124 013212                          LET TEMPO := PSW
      013212  016767  164560  165256
 3125 013220                          INLINE  <BIT #17,TEMPO>
      013220  032767  000017  165250
 3126 013226                          IFCOND NE THEN
      013226  001402
 3127 013230                              CALL ERROR
      013230  004767  044444
 3128 013234                          ENDIF
      013234
 3129 013234                          LET TEMPO := #BUFLEN-50
      013234  012767  000030  165234
 3130 013242                          IF R0 NE TEMPO THEN
      013242  020067  165230
      013246  001402
 3131 013250                              CALL ERROR
      013250  004767  044424
 3132 013254                          ENDIF
      013254
 3133 013254                          IF R1 NE #BUFF1+50 THEN
      013254  020127  063306
      013260  001402
 3134 013262                              CALL ERROR
      013262  004767  044412
 3135 013266                          ENDIF
      013266
 3136 013266                          IF R2 NE #0 THEN
      013266  005702
      013270  001402
 3137 013272                              CALL ERROR
      013272  004767  044402
 3138 013276                          ENDIF
      013276
 3139 013276                          IF R3 NE #0 THEN
      013276  005703
      013300  001402
 3140 013302                              CALL ERROR
      013302  004767  044372
 3141 013306                          ENDIF
      013306
 3142 013306                          IF R4 NE #0 THEN
      013306  005704
      013310  001402
 3143 013312                              CALL ERROR
      013312  004767  044362
 3144 013316                          ENDIF
      013316
 3145 013316                          IF R5 NE #0 THEN
      013316  005705
      013320  001402
 3146 013322                              CALL ERROR
      013322  004767  044352
```

```
                SPANCI
                .WORD   TSPNCI
                .WORD   TSPNCI+4
                MOV     PSW,TEMPO
                BIT     #17,TEMPO
                BEQ     $50443
                JSR     PC,ERROR
$50443:
                MOV     #BUFLEN-50,TEMPO
                CMP     R0,TEMPO
                BEQ     $50444
                JSR     PC,ERROR
$50444:
                CMP     R1,#BUFF1+50
                BEQ     $50445
                JSR     PC,ERROR
$50445:
                TST     R2
                BEQ     $50446
                JSR     PC,ERROR
$50446:
                TST     R3
                BEQ     $50447
                JSR     PC,ERROR
$50447:
                TST     R4
                BEQ     $50450
                JSR     PC,ERROR
$50450:
                TST     R5
                BEQ     $50451
                JSR     PC,ERROR
```

```
   3147 013326                          ENDIF
        013326                                                                  $50451:
   3148 013326                          INLINE   <JMP        XCMPC>
        013326  000167  000026                                                      JMP     XCMPC
   3149
   3150                                 ;+
   3151                                 ; TABLE OF TEST CASES FOR SPANC
   3152                                 ;-
   3153
   3154 013332  000003                  TSPANC: .WORD   3               ;# OF TEST CASES FOR SPANC
   3155
   3156 013334  000000                          .WORD   0                       ;SRC.LEN
   3157 013336  000030                          .WORD   30                      ;OFFSET
   3158
   3159 013340  000100                          .WORD   BUFLEN          ;TEST CASE NUMBER 2
   3160 013342  000040                          .WORD   40
   3161
   3162 013344  000100                          .WORD   BUFLEN          ;TEST CASE NUMBER 3
   3163 013346  000104                          .WORD   BUFLEN+4
   3164
   3165
   3166 013350  000100                  TSPNCI: .WORD   BUFLEN          ;TEST CASE FOR INLINE
   3167 013352  063236                          .WORD   BUFF1
   3168 013354  000010                          .WORD   10
   3169 013356  064036                          .WORD   CHRSET
   3170
```

```
3172                                    .SBTTL  CMPC
3173                                    ;;*****************************************************************
3174                                    ;
3175                                    ;  CMPC TEST
3176                                    ;
3177                                    ;;*****************************************************************
3178
3179 013360                             ROUTINE XCMPC
     013360                                                                          XCMPC:
3180 013360                                INLINE  <SCOPE>
     013360  000004                                                                     SCOPE
3181 013362                                INLINE  <NOP>
     013362  000240                                                                     NOP
3182 013364                                LET $TIMES := #20           ;ITERATION COUNT
     013364  012767  000020  165012                                                     MOV     #20,$TIMES
3183 013372                                LET COUNT := TCMPC          ;GET TEST COUNT
     013372  016767  000736  165070                                                     MOV     TCMPC,COUNT
3184 013400                                LET TSTCAS := #TCMPC+2              ;POINT TO FIRST TEST CASE
     013400  012767  014336  165136                                                     MOV     #TCMPC+2,TSTCAS
3185                                    ;REPEAT
3186 013406                                INLINE  <$5044:>
     013406                                                                          $5044:
3187
3188                                    ;+
3189                                    ; GET TEST DATA FROM TABLE
3190                                    ;-
3191
3192 013406                                LET R0 := TSTCAS
     013406  016700  165132                                                         MOV     TSTCAS,R0
3193 013412                                LET DATA00 := (R0)+         ;SRC1.LEN
     013412  012067  165144                                                         MOV     (R0)+,DATA00
3194 013416                                LET DATA01 := (R0)+         ;SRC2.LEN
     013416  012067  165142                                                         MOV     (R0)+,DATA01
3195 013422                                LET DATA02 := (R0)          ;OFFSET
     013422  011067  165140                                                         MOV     (R0),DATA02
3196
3197                                    ;+
3198                                    ; SET UP GPR'S
3199                                    ;-
3200
3201 013426                                LET R0 := DATA00            ;SRC1
     013426  016700  165130                                                         MOV     DATA00,R0
3202 013432                                LET R1 := #BUFF1
     013432  012701  063236                                                         MOV     #BUFF1,R1
3203 013436                                LET R2 := DATA01            ;SRC2
     013436  016702  165122                                                         MOV     DATA01,R2
3204 013442                                LET R3 := #BUFF2
     013442  012703  063336                                                         MOV     #BUFF2,R3
3205 013446                                LET R4 := #377                      ;FILL CHARACTER
     013446  012704  000377                                                         MOV     #377,R4
3206 013452                                LET OFFSET := DATA02        ;OFFSET
     013452  016767  165110  165032                                                 MOV     DATA02,OFFSET
3207 013460                                INLINE  <BIT #BIT00,COUNT>
     013460  032767  000001  165002                                                 BIT     #BIT00,COUNT
3208 013466                                IFCOND EQ THEN
     013466  001002                                                                 BNE     $50454
3209 013470                                    LET R5 := #0
```

CMPC

```
013470  005005                          ELSE                                        CLR     R5
3210 013472
     013472  000402                                                              BR      $50455
     013474                                                                $50454:
3211 013474                                  LET R5 := #177777
     013474  012705  177777                                                      MOV     #177777,R5
3212 013500                                  ENDIF
     013500                                                                $50455:
3213
3214                                    ;+
3215                                    ; SET UP BUFFERS
3216                                    ;-
3217
3218 013500                                  LET BFADRS := #BUFF1
     013500  012767  063236  165034                                              MOV     #BUFF1,BFADRS
3219 013506                                  CALL SETBUF
     013506  004767  043006                                                      JSR     PC,SETBUF
3220 013512                                  LET BFADRS := #BUFF2
     013512  012767  063336  165022                                              MOV     #BUFF2,BFADRS
3221 013520                                  CALL SETBUF
     013520  004767  042774                                                      JSR     PC,SETBUF
3222 013524                                  IF DATA00 HI DATA01 THEN
     013524  026767  165032  165032                                              CMP     DATA00,DATA01
     013532  101412                                                            BLOS    $50456
3223 013534                                      LET TEMPO := OFFSET + #BUFF1
     013534  016767  164752  164734                                              MOV     OFFSET,TEMPO
     013542  062767  063236  164726                                              ADD     #BUFF1,TEMPO
3224 013550                                      LET CC.EXP := #1
     013550  012767  000001  164744                                              MOV     #1,CC.EXP
3225 013556                                  ELSE
     013556  000411                                                            BR      $50457
     013560                                                                $50456:
3226 013560                                      LET TEMPO := OFFSET + #BUFF2
     013560  016767  164726  164710                                              MOV     OFFSET,TEMPO
     013566  062767  063336  164702                                              ADD     #BUFF2,TEMPO
3227 013574                                      LET CC.EXP := #10
     013574  012767  000010  164720                                              MOV     #10,CC.EXP
3228 013602                                  ENDIF
     013602                                                                $50457:
3229 013602                                  LET @TEMPO :B= #0
     013602  105077  164670                                                      CLRB    @TEMPO
3230
3231                                    ;+
3232                                    ; SET UP EXPECTED GPR'S AND CONDITION CODES
3233                                    ;-
3234
3235 013606                                  LET R0.EXP := R0 - DATA02
     013606  010067  164712                                                      MOV     R0,R0.EXP
     013612  166767  164750  164704                                              SUB     DATA02,R0.EXP
3236 013620                                  LET R1.EXP := R1 + DATA02
     013620  010167  164702                                                      MOV     R1,R1.EXP
     013624  066767  164736  164674                                              ADD     DATA02,R1.EXP
3237 013632                                  LET R2.EXP := R2 - DATA02
     013632  010267  164672                                                      MOV     R2,R2.EXP
     013636  166767  164724  164664                                              SUB     DATA02,R2.EXP
3238 013644                                  LET R3.EXP := R3 + DATA02
     013644  010367  164662                                                      MOV     R3,R3.EXP
```

```
      013650  066767  164712  164654                                                    ADD     DATA02,R3.EXP
 3239 013656                          LET R4.EXP := R4
      013656  010467  164652                                                            MOV     R4,R4.EXP
 3240 013662                          LET R5.EXP := R5
      013662  010567  164650                                                            MOV     R5,R5.EXP
 3241 013666                          IF DATA02 HIS DATA00 THEN
      013666  026767  164674  164666                                                    CMP     DATA02,DATA00
      013674  103406                                                                   BLO     $50460
 3242 013676                              LET R0.EXP := #0
      013676  005067  164622                                                            CLR     R0.EXP
 3243 013702                              LET R1.EXP := R1 + R0
      013702  010167  164620                                                            MOV     R1,R1.EXP
      013706  060067  164614                                                            ADD     R0,R1.EXP
 3244 013712                          ENDIF
      013712                                                                  $50460:
 3245 013712                          IF DATA02 HIS DATA01 THEN
      013712  026767  164650  164644                                                    CMP     DATA02,DATA01
      013720  103406                                                                   BLO     $50461
 3246 013722                              LET R2.EXP := #0
      013722  005067  164602                                                            CLR     R2.EXP
 3247 013726                              LET R3.EXP := R3 + R2
      013726  010367  164600                                                            MOV     R3,R3.EXP
      013732  060267  164574                                                            ADD     R2,R3.EXP
 3248 013736                          ENDIF
      013736                                                                  $50461:
 3249 013736                          IF DATA02 HIS DATA00 AND DATA02 HIS DATA01 THEN
      013736  026767  164624  164616                                                    CMP     DATA02,DATA00
      013744  103407                                                                   BLO     $50462
      013746  026767  164614  164610                                                    CMP     DATA02,DATA01
      013754  103403                                                                   BLO     $50462
 3250 013756                              LET CC.EXP := #4
      013756  012767  000004  164536                                                    MOV     #4,CC.EXP
 3251 013764                          ENDIF
      013764                                                                  $50462:
 3252
 3253                                 ;+
 3254                                 ; DO THE CIS INSTRUCTION
 3255                                 ;-
 3256
 3257 013764                          INLINE  <CMPC>
      013764  076044                                                                   CMPC
 3258
 3259                                 ;+
 3260                                 ; CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS
 3261                                 ;-
 3262
 3263 013766                          LET CC := PSW
      013766  016767  164004  164524                                                    MOV     PSW,CC
 3264 013774                          LET CC := CC CLR.BY #177760
      013774  042767  177760  164516                                                    BIC     #177760,CC
 3265 014002                          IF CC NE CC.EXP THEN
      014002  026767  164512  164512                                                    CMP     CC,CC.EXP
      014010  001402                                                                   BEQ     $50463
 3266 014012                              CALL ERROR
      014012  004767  043662                                                            JSR     PC,ERROR
 3267 014016                          ENDIF
      014016                                                                  $50463:
```

```
3268 014016                             IF R0 NE R0.EXP THEN
     014016  020067  164502                                          CMP     R0,R0.EXP
     014022  001402                                                  BEQ     $50464
3269 014024                                 CALL ERROR
     014024  004767  043650                                          JSR     PC,ERROR
3270 014030                             ENDIF
     014030                                                     $50464:
3271 014030                             IF R1 NE R1.EXP THEN
     014030  020167  164472                                          CMP     R1,R1.EXP
     014034  001402                                                  BEQ     $50465
3272 014036                                 CALL ERROR
     014036  004767  043636                                          JSR     PC,ERROR
3273 014042                             ENDIF
     014042                                                     $50465:
3274 014042                             IF R2 NE R2.EXP THEN
     014042  020267  164462                                          CMP     R2,R2.EXP
     014046  001402                                                  BEQ     $50466
3275 014050                                 CALL ERROR
     014050  004767  043624                                          JSR     PC,ERROR
3276 014054                             ENDIF
     014054                                                     $50466:
3277 014054                             IF R3 NE R3.EXP THEN
     014054  020367  164452                                          CMP     R3,R3.EXP
     014060  001402                                                  BEQ     $50467
3278 014062                                 CALL ERROR
     014062  004767  043612                                          JSR     PC,ERROR
3279 014066                             ENDIF
     014066                                                     $50467:
3280 014066                             IF R4 NE R4.EXP THEN
     014066  020467  164442                                          CMP     R4,R4.EXP
     014072  001402                                                  BEQ     $50470
3281 014074                                 CALL ERROR
     014074  004767  043600                                          JSR     PC,ERROR
3282 014100                             ENDIF
     014100                                                     $50470:
3283 014100                             IF R5 NE R5.EXP THEN
     014100  020567  164432                                          CMP     R5,R5.EXP
     014104  001402                                                  BEQ     $50471
3284 014106                                 CALL ERROR
     014106  004767  043566                                          JSR     PC,ERROR
3285 014112                             ENDIF
     014112                                                     $50471:
3286 014112                             LET TSTCAS := TSTCAS + #6
     014112  062767  000006  164424                                  ADD     #6,TSTCAS
3287 014120                             LET COUNT := COUNT - #1
     014120  005367  164344                                          DEC     COUNT
3288                                 ;UNTIL COUNT EQ #0
3289 014124                             IF COUNT NE #0 THEN
     014124  005767  164340                                          TST     COUNT
     014130  001402                                                  BEQ     $50472
3290 014132                                 INLINE  <JMP     $5044>
     014132  000167  177250                                          JMP     $5044
3291 014136                             ENDIF
     014136                                                     $50472:
3292
3293
3294                                 ;*
```

CMPC

```
3295                                    ; DO ONE INLINE CASE - CMPCI
3296                                    ;-
3297
3298 014136                                LET BFADRS := #BUFF1
     014136  012767  063236  164376                                  MOV     #BUFF1,BFADRS
3299 014144                                CALL CLRBUF
     014144  004767  042306                                         JSR     PC,CLRBUF
3300 014150                                LET BFADRS := #BUFF2
     014150  012767  063336  164364                                 MOV     #BUFF2,BFADRS
3301 014156                                CALL CLRBUF
     014156  004767  042274                                         JSR     PC,CLRBUF
3302 014162                                CALL CLRREG
     014162  004767  042252                                         JSR     PC,CLRREG
3303 014166                                LET TEMPO := #BUFF1 + #40
     014166  012767  063236  164302                                 MOV     #BUFF1,TEMPO
     014174  062767  000040  164274                                 ADD     #40,TEMPO
3304 014202                                LET @TEMPO :B= #377
     014202  112777  000377  164266                                 MOVB    #377,@TEMPO
3305 014210                                INLINE  <CMPCI>
     014210  076144                                                 CMPCI
3306 014212                                INLINE  <.WORD      TCMPCI>
     014212  014416                                                 .WORD   TCMPCI
3307 014214                                INLINE  <.WORD      TCMPCI+4>
     014214  014422                                                 .WORD   TCMPCI+4
3308 014216                                INLINE  <.WORD      0>
     014216  000000                                                 .WORD   0
3309 014220                                LET TEMPO := PSW
     014220  016767  163552  164250                                 MOV     PSW,TEMPO
3310 014226                                LET TEMPO := TEMPO CLR.BY #177760
     014226  042767  177760  164242                                 BIC     #177760,TEMPO
3311 014234                                IF TEMPO NE #10 THEN
     014234  026727  164236  000010                                 CMP     TEMPO,#10
     014242  001402                                                 BEQ     $50473
3312 014244                                    CALL ERROR
     014244  004767  043430                                         JSR     PC,ERROR
3313 014250                                ENDIF
     014250                                                 $50473:
3314 014250                                IF R0 NE #0 THEN
     014250  005700                                                 TST     R0
     014252  001402                                                 BEQ     $50474
3315 014254                                    CALL ERROR
     014254  004767  043420                                         JSR     PC,ERROR
3316 014260                                ENDIF
     014260                                                 $50474:
3317 014260                                IF R1 NE #0 THEN
     014260  005701                                                 TST     R1
     014262  001402                                                 BEQ     $50475
3318 014264                                    CALL ERROR
     014264  004767  043410                                         JSR     PC,ERROR
3319 014270                                ENDIF
     014270                                                 $50475:
3320 014270                                IF R2 NE #0 THEN
     014270  005702                                                 TST     R2
     014272  001402                                                 BEQ     $50476
3321 014274                                    CALL ERROR
     014274  004767  043400                                         JSR     PC,ERROR
3322 014300                                ENDIF
```

```
      014300                                                                $50476:
 3323 014300                          IF R3 NE #0 THEN
      014300  005703                                                              TST     R3
      014302  001402                                                              BEQ     $50477
 3324 014304                              CALL ERROR
      014304  004767  043370                                                      JSR     PC,ERROR
 3325 014310                          ENDIF
      014310                                                                $50477:
 3326 014310                          IF R4 NE #0 THEN
      014310  005704                                                              TST     R4
      014312  001402                                                              BEQ     $50500
 3327 014314                              CALL ERROR
      014314  004767  043360                                                      JSR     PC,ERROR
 3328 014320                          ENDIF
      014320                                                                $50500:
 3329 014320                          IF R5 NE #0 THEN
      014320  005705                                                              TST     R5
      014322  001402                                                              BEQ     $50501
 3330 014324                              CALL ERROR
      014324  004767  043350                                                      JSR     PC,ERROR
 3331 014330                          ENDIF
      014330                                                                $50501:
 3332 014330                          INLINE  <JMP XMATC>
      014330  000167  000072                                                      JMP     XMATC
 3333
 3334                             ;+
 3335                             ; TABLE OF TEST CASES FOR CMPC
 3336                             ;-
 3337
 3338 014334  000010              TCMPC:  .WORD   8.              ;# OF TEST CASES FOR CMPC
 3339
 3340 014336  000000                      .WORD   0                       ;SRC1.LEN
 3341 014340  000020                      .WORD   20                      ;SRC2.LEN
 3342 014342  000040                      .WORD   40                      ;OFFSET
 3343
 3344 014344  000010                      .WORD   10              ;TEST CASE NUMBER 2
 3345 014346  000030                      .WORD   30
 3346 014350  000007                      .WORD   7
 3347
 3348 014352  000044                      .WORD   44              ;TEST CASE NUMBER 3
 3349 014354  000070                      .WORD   70
 3350 014356  000060                      .WORD   60
 3351
 3352 014360  000030                      .WORD   30              ;TEST CASE NUMBER 4
 3353 014362  000050                      .WORD   50
 3354 014364  000060                      .WORD   60
 3355
 3356 014366  000000                      .WORD   0               ;TEST CASE NUMBER 5
 3357 014370  000000                      .WORD   0
 3358 014372  000010                      .WORD   10
 3359
 3360 014374  000060                      .WORD   60              ;TEST CASE NUMBER 6
 3361 014376  000040                      .WORD   40
 3362 014400  000030                      .WORD   30
 3363
 3364 014402  000057                      .WORD   57              ;TEST CASE NUMBER 7
 3365 014404  000033                      .WORD   33
```

CMPC

```
3366 014406  000045                        .WORD   45
3367
3368 014410  000020                        .WORD   20              ;TEST CASE NUMBER 8
3369 014412  000011                        .WORD   11
3370 014414  000023                        .WORD   23
3371
3372
3373 014416  000060                TCMPCI: .WORD   60              ;TEST CASE FOR INLINE
3374 014420  063236                        .WORD   BUFF1
3375 014422  000060                        .WORD   60
3376 014424  063336                        .WORD   BUFF2
3377
```

```
                                .SBTTL  MATC
                                ;;******************************************************************
                                ;
                                ;  MATC TEST
                                ;
                                ;;******************************************************************

                                ROUTINE XMATC
014426                                                                          XMATC:
                                   INLINE  <SCOPE>
014426  000004                                                                      SCOPE
                                   INLINE  <NOP>
014430  000240                                                                      NOP
                                   LET $TIMES := #20            ;ITERATION COUNT
014432  012767  000020  163744                                                     MOV     #20,$TIMES
                                   LET COUNT := TMATC           ;GET TEST COUNT
014440  016767  000670  164022                                                     MOV     TMATC,COUNT
                                   LET TSTCAS := #TMATC+2               ;POINT TO FIRST TEST CASE
014446  012767  015336  164070                                                     MOV     #TMATC+2,TSTCAS
                                ;REPEAT
                                   INLINE  <$5045:>
014454                                                                          $5045:

                                ;+
                                ; GET TEST DATA FROM TABLE
                                ;-

                                   LET R0 := TSTCAS
014454  016700  164064                                                          MOV     TSTCAS,R0
                                   LET DATA00 := (R0)+                  ;OBJ.LEN
014460  012067  164076                                                          MOV     (R0)+,DATA00
                                   LET DATA01 := (R0)                   ;OFFSET
014464  011067  164074                                                          MOV     (R0),DATA01

                                ;+
                                ; SET UP THE GPR'S
                                ;-

                                   LET R0 := #70                        ;SRC.LEN
014470  012700  000070                                                          MOV     #70,R0
                                   LET R1 := #BUFF1                     ;SRC.ADR
014474  012701  063236                                                          MOV     #BUFF1,R1
                                   LET R2 := DATA00
014500  016702  164056                                                          MOV     DATA00,R2
                                   LET R3 := #BUFF2                     ;OBJ.LEN
014504  012703  063336                                                          MOV     #BUFF2,R3
                                   INLINE  <BIT #BIT00,COUNT>
014510  032767  000001  163752                                                  BIT     #BIT00,COUNT
                                   IFCOND EQ THEN
014516  001003                                                                  BNE     $50504
                                      LET R4 := #0
014520  005004                                                                  CLR     R4
                                      LET R5 := #0
014522  005005                                                                  CLR     R5
                                   ELSE
014524  000404                                                                  BR      $50505
014526                                                                  $50504:
```

```
3416 014526                                 LET R4 := #177777
     014526  012704  177777                                        MOV     #177777,R4
3417 014532                                 LET R5 := #177777
     014532  012705  177777                                        MOV     #177777,R5
3418 014536                              ENDIF
     014536                                                 $50505:
3419
3420                                     ;+
3421                                     ; SET UP THE BUFFERS
3422                                     ;-
3423
3424 014536                              LET BFADRS := #BUFF1
     014536  012767  063236  163776                                 MOV     #BUFF1,BFADRS
3425 014544                              CALL SETBUF
     014544  004767  041750                                         JSR     PC,SETBUF
3426 014550                              LET BFADRS := #BUFF2
     014550  012767  063336  163764                                 MOV     #BUFF2,BFADRS
3427 014556                              CALL CLRBUF
     014556  004767  041674                                         JSR     PC,CLRBUF
3428 014562                              IF DATA01 LO #70 THEN
     014562  026727  163776  000070                                 CMP     DATA01,#70
     014570  103015                                                BHIS    $50506
3429 014572                                 LET R1 := R1 + DATA01
     014572  066701  163766                                         ADD     DATA01,R1
3430 014576                                 REPEAT
     014576                                                 $50507:
3431 014576                                     INLINE  <CLRB       (R1)+>
     014576  105021                                                CLRB    (R1)+
3432 014600                                     LET TEMP0 := #BUFF1 + R0
     014600  012767  063236  163670                                 MOV     #BUFF1,TEMP0
     014606  060067  163664                                         ADD     R0,TEMP0
3433 014612                                 UNTIL R1 EQ TEMP0
     014612  020167  163660                                         CMP     R1,TEMP0
     014616  001367                                                 BNE     $50507
3434 014620                                 LET R1 := #BUFF1        ;RESTORE R1
     014620  012701  063236                                         MOV     #BUFF1,R1
3435 014624                              ENDIF
     014624                                                 $50506:
3436
3437                                     ;+
3438                                     ; SET UP EXPECTED GPR'S AND CONDITION CODES
3439                                     ;-
3440
3441 014624                              LET R0.EXP := R0
     014624  010067  163674                                         MOV     R0,R0.EXP
3442 014630                              LET R1.EXP := R1
     014630  010167  163672                                         MOV     R1,R1.EXP
3443 014634                              LET R2.EXP := R2
     014634  010267  163670                                         MOV     R2,R2.EXP
3444 014640                              LET R3.EXP := R3
     014640  010367  163666                                         MOV     R3,R3.EXP
3445 014644                              LET R4.EXP := R4
     014644  010467  163664                                         MOV     R4,R4.EXP
3446 014650                              LET R5.EXP := R5
     014650  010567  163662                                         MOV     R5,R5.EXP
3447 014654                              IF DATA00 HI #0 THEN
     014654  005767  163702                                         TST     DATA00
```

MATC

```
      014660  101432                                                          BLOS    $50510
 3448 014662                          LET TEMP0 := R0 - DATA01
      014662  010067  163610                                                  MOV     R0,TEMP0
      014666  166767  163672  163602                                          SUB     DATA01,TEMP0
 3449 014674                          IF TEMP0 LO DATA00 THEN
      014674  026767  163576  163660                                          CMP     TEMP0,DATA00
      014702  103007                                                          BHIS    $50511
 3450 014704                              LET R0.EXP := #0
      014704  005067  163614                                                  CLR     R0.EXP
 3451 014710                              LET R1.EXP := R0 + R1
      014710  010067  163612                                                  MOV     R0,R1.EXP
      014714  060167  163606                                                  ADD     R1,R1.EXP
 3452 014720                          ELSE
      014720  000412                                                          BR      $50512
      014722                                                          $50511:
 3453 014722                              LET R0.EXP := R0 - DATA01
      014722  010067  163576                                                  MOV     R0,R0.EXP
      014726  166767  163632  163570                                          SUB     DATA01,R0.EXP
 3454 014734                              LET R1.EXP := R1 + DATA01
      014734  010167  163566                                                  MOV     R1,R1.EXP
      014740  066767  163620  163560                                          ADD     DATA01,R1.EXP
 3455 014746                          ENDIF
      014746                                                          $50512:
 3456 014746                      ENDIF
      014746                                                          $50510:
 3457 014746                      INLINE  <TST R0.EXP>
      014746  005767  163552                                                  TST     R0.EXP
 3458 014752                      LET CC.EXP := PSW
      014752  016767  163020  163542                                          MOV     PSW,CC.EXP
 3459 014760                      LET CC.EXP := CC.EXP CLR.BY #177760
      014760  042767  177760  163534                                          BIC     #177760,CC.EXP
 3460
 3461                             ;+
 3462                             ; DO THE CIS INSTRUCTION
 3463                             ;-
 3464
 3465 014766                      INLINE  <MATC>
      014766  076045                                                          MATC
 3466
 3467                             ;+
 3468                             ; CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS
 3469                             ;-
 3470
 3471 014770                      LET CC := PSW
      014770  016767  163002  163522                                          MOV     PSW,CC
 3472 014776                      LET CC := CC CLR.BY #177760
      014776  042767  177760  163514                                          BIC     #177760,CC
 3473 015004                      IF CC NE CC.EXP THEN
      015004  026767  163510  163510                                          CMP     CC,CC.EXP
      015012  001402                                                          BEQ     $50513
 3474 015014                          CALL ERROR
      015014  004767  042660                                                  JSR     PC,ERROR
 3475 015020                      ENDIF
      015020                                                          $50513:
 3476 015020                      IF R0 NE R0.EXP THEN
      015020  020067  163500                                                  CMP     R0,R0.EXP
      015024  001402                                                          BEQ     $50514
```

```
3477 015026                                 CALL ERROR
     015026  004767  042646                                          JSR     PC,ERROR
3478 015032                                 ENDIF
     015032                                                         $50514:
3479 015032                                 IF R1 NE R1.EXP THEN
     015032  020167  163470                                         CMP     R1,R1.EXP
     015036  001402                                                 BEQ     $50515
3480 015040                                     CALL ERROR
     015040  004767  042634                                         JSR     PC,ERROR
3481 015044                                 ENDIF
     015044                                                         $50515:
3482 015044                                 IF R2 NE R2.EXP THEN
     015044  020267  163460                                         CMP     R2,R2.EXP
     015050  001402                                                 BEQ     $50516
3483 015052                                     CALL ERROR
     015052  004767  042622                                         JSR     PC,ERROR
3484 015056                                 ENDIF
     015056                                                         $50516:
3485 015056                                 IF R3 NE R3.EXP THEN
     015056  020367  163450                                         CMP     R3,R3.EXP
     015062  001402                                                 BEQ     $50517
3486 015064                                     CALL ERROR
     015064  004767  042610                                         JSR     PC,ERROR
3487 015070                                 ENDIF
     015070                                                         $50517:
3488 015070                                 IF R4 NE R4.EXP THEN
     015070  020467  163440                                         CMP     R4,R4.EXP
     015074  001402                                                 BEQ     $50520
3489 015076                                     CALL ERROR
     015076  004767  042576                                         JSR     PC,ERROR
3490 015102                                 ENDIF
     015102                                                         $50520:
3491 015102                                 IF R5 NE R5.EXP THEN
     015102  020567  163430                                         CMP     R5,R5.EXP
     015106  001402                                                 BEQ     $50521
3492 015110                                     CALL ERROR
     015110  004767  042564                                         JSR     PC,ERROR
3493 015114                                 ENDIF
     015114                                                         $50521:
3494 015114                                 LET TSTCAS := TSTCAS + #4
     015114  062767  000004  163422                                 ADD     #4,TSTCAS
3495 015122                                 LET COUNT := COUNT - #1
     015122  005367  163342                                         DEC     COUNT
3496                                       ;UNTIL COUNT EQ #0
3497 015126                                 IF COUNT NE #0 THEN
     015126  005767  163336                                         TST     COUNT
     015132  001402                                                 BEQ     $50522
3498 015134                                     INLINE  <JMP     $5045>
     015134  000167  177314                                         JMP     $5045
3499 015140                                 ENDIF
     015140                                                         $50522:
3500
3501
3502                                        ;+
3503                                        ; DO ONE INLINE CASE - MATCI
3504                                        ;-
3505
```

```
3506 015140                             LET BFADRS := #BUFF1
     015140  012767  063236  163374                    MOV     #BUFF1,BFADRS
3507 015146                             CALL SETBUF
     015146  004767  041346                            JSR     PC,SETBUF
3508 015152                             LET BFADRS := #BUFF2
     015152  012767  063336  163362                    MOV     #BUFF2,BFADRS
3509 015160                             CALL SETBUF
     015160  004767  041334                            JSR     PC,SETBUF
3510 015164                             LET R0 := #BUFF1
     015164  012700  063236                            MOV     #BUFF1,R0
3511 015170                             REPEAT
     015170                                          $50523:
3512 015170                                 INLINE  <CLRB    (R0)+>
     015170  105020                                    CLRB    (R0)+
3513 015172                             UNTIL R0 EQ #BUFF1+40
     015172  020027  063276                            CMP     R0,#BUFF1+40
     015176  001374                                    BNE     $50523
3514 015200                             CALL CLRREG
     015200  004767  041234                            JSR     PC,CLRREG
3515 015204                             INLINE  <MATCI>
     015204  076145                                    MATCI
3516 015206                             INLINE  <.WORD       TMATCI>
     015206  015356                                    .WORD   TMATCI
3517 015210                             INLINE  <.WORD       TMATCI+4>
     015210  015362                                    .WORD   TMATCI+4
3518 015212                             LET CC := PSW
     015212  016767  162560  163300                    MOV     PSW,CC
3519 015220                             INLINE  <BIT #17,CC>
     015220  032767  000017  163272                    BIT     #17,CC
3520 015226                             IFCOND NE THEN
     015226  001402                                    BEQ     $50524
3521 015230                                 CALL ERROR
     015230  004767  042444                            JSR     PC,ERROR
3522 015234                             ENDIF
     015234                                          $50524:
3523 015234                             IF R0 NE #30 THEN           ;SRC.LEN - 40
     015234  020027  000030                            CMP     R0,#30
     015240  001402                                    BEQ     $50525
3524 015242                                 CALL ERROR
     015242  004767  042432                            JSR     PC,ERROR
3525 015246                             ENDIF
     015246                                          $50525:
3526 015246                             IF R1 NE #BUFF1+40 THEN
     015246  020127  063276                            CMP     R1,#BUFF1+40
     015252  001402                                    BEQ     $50526
3527 015254                                 CALL ERROR
     015254  004767  042420                            JSR     PC,ERROR
3528 015260                             ENDIF
     015260                                          $50526:
3529 015260                             IF R2 NE #0 THEN
     015260  005702                                    TST     R2
     015262  001402                                    BEQ     $50527
3530 015264                                 CALL ERROR
     015264  004767  042410                            JSR     PC,ERROR
3531 015270                             ENDIF
     015270                                          $50527:
3532 015270                             IF R3 NE #0 THEN
```

```
                                                                        TST     R3
                                                                        BEQ     $50530
3533 015274                             CALL ERROR
     015274  004767  042400                                              JSR     PC,ERROR
3534 015300                             ENDIF
     015300                                                     $50530:
3535 015300                             IF R4 NE #0 THEN
     015300  005704                                                     TST     R4
     015302  001402                                                     BEQ     $50531
3536 015304                             CALL ERROR
     015304  004767  042370                                              JSR     PC,ERROR
3537 015310                             ENDIF
     015310                                                     $50531:
3538 015310                             IF R5 NE #0 THEN
     015310  005705                                                     TST     R5
     015312  001402                                                     BEQ     $50532
3539 015314                             CALL ERROR
     015314  004767  042360                                              JSR     PC,ERROR
3540 015320                             ENDIF
     015320                                                     $50532:
3541 015320                             IF R2 NE #0 THEN
     015320  005702                                                     TST     R2
     015322  001402                                                     BEQ     $50533
3542 015324                             CALL ERROR
     015324  004767  042350                                              JSR     PC,ERROR
3543 015330                             ENDIF
     015330                                                     $50533:
3544 015330                             INLINE  <JMP INTST>
     015330  000167  000032                                              JMP     INTST
3545
3546                                    ;+
3547                                    ; TABLE OF TEST DATA FOR MATC
3548                                    ;-
3549
3550 015334  000004                     TMATC:  .WORD   4               ;# OF TEST CASES FOR MATC
3551
3552 015336  000000                             .WORD   0                       ;OBJ.LEN
3553 015340  000010                             .WORD   10                      ;OFFSET
3554
3555 015342  000074                             .WORD   74              ;TEST CASE NUMBER 2
3556 015344  000010                             .WORD   10
3557
3558 015346  000040                             .WORD   40              ;TEST CASE NUMBER 3
3559 015350  000020                             .WORD   20
3560
3561 015352  000040                             .WORD   40              ;TEST CASE NUMBER 4
3562 015354  000050                             .WORD   50
3563
3564
3565 015356  000070                     TMATCI: .WORD   70              ;TEST CASE FOR INLINE
3566 015360  063236                             .WORD   BUFF1
3567 015362  000010                             .WORD   10
3568 015364  063336                             .WORD   BUFF2
3569
```

```
3571                            ;;*****************************************************************
3572                            .SBTTL  INTERRUPTABILITY SUBTEST
3573                            ;;*****************************************************************
3574
3575                            ;+
3576                            ;       THIS SUBTEST VERIFIES THAT A CIS INSTRUCTION CAN BE
3577                            ;       SUSPENDED (USING CONSOLE TERMINAL PRINTER INTERRUPT).
3578                            ;-
3579
3580 015366                     ROUTINE INTST
    015366                                                                          INTST:
3581 015366                        INLINE       <SCOPE>
    015366  000004                                                                  SCOPE
3582 015370                        INLINE  <NOP>
    015370  000240                                                                  NOP
3583 015372                        IFB $ENV EQ #1 THEN          ;IF UNDER APT THEN
    015372  126727  163254  000001                                                 CMPB    $ENV,#1
    015400  001005                                                                 BNE     $50536
3584 015402                            IF $PASS NE #0 THEN              ;   IF 1ST PASS COMPLETED THEN
    015402  005767  163232                                                         TST     $PASS
    015406  001402                                                                 BEQ     $50537
3585 015410                                INLINE <JMP  SKPTST>         ;       SKIP THE TEST
    015410  000167  001014                                                         JMP     SKPTST
3586 015414                            ENDIF                            ;ELSE
    015414                                                                      $50537:
3587 015414                        ENDIF                                ;   DO THE TEST
    015414                                                                      $50536:
3588
3589 015414                        LET $TIMES := #1             ;ITERATION COUNT
    015414  012767  000001  162762                                                 MOV     #1,$TIMES
3590 015422                        LET ERR := #2                        ;SET UP FOR ERROR MSG
    015422  012767  000002  163122                                                 MOV     #2,ERR
3591 015430                        LET SAVPSW := PSW
    015430  016767  162342  163156                                                 MOV     PSW,SAVPSW
3592 015436                        LET PSW := #PR7
    015436  012767  000340  162332                                                 MOV     #PR7,PSW
3593 015444                        LET TEMPO := #0                      ;SUSPENDED FLAG,1=INST SUSPENDED
    015444  005067  163026                                                         CLR     TEMPO
3594 015450                        LET DELAY := #13000
    015450  012767  013000  163014                                                 MOV     #13000,DELAY
3595
3596                            ;+
3597                            ;       SET UP TP VECTOR
3598                            ;-
3599
3600 015456                        LET TPS := #0                        ;MAKE SURE INTERRUPT ENABLE IS OFF
    015456  005067  162102                                                         CLR     TPS
3601 015462                        LET TPVEC := #TPSRVC
    015462  012767  016432  162374                                                 MOV     #TPSRVC,TPVEC
3602 015470                        LET TPVEC+2 := #PR7
    015470  012767  000340  162370                                                 MOV     #PR7,TPVEC+2
3603
3604                            ;+
3605                            ;       SET UP FOR MOVTC INST
3606                            ;-
3607
3608 015476                        CALL SAVREG                  ;SAVE R0-R5
```

```
     015476  004767  041234                                                                    JSR     PC,SAVREG
3609 015502                                REPEAT
     015502                                                                          $50540:
3610 015502                                    LET TEMP1 := DELAY
     015502  016767  162764  162770                                                            MOV     DELAY,TEMP1
3611 015510                                    LET DATA00 := #1700              ;SRC.LEN
     015510  012767  001700  163044                                                            MOV     #1700,DATA00
3612 015516                                    LET DATA01 := #SPACE             ;SRC.ADR
     015516  012767  066152  163040                                                            MOV     #SPACE,DATA01
3613 015524                                    LET DATA02 := #1740              ;DST.LEN
     015524  012767  001740  163034                                                            MOV     #1740,DATA02
3614 015532                                    LET DATA03 := #SPACEY            ;DST.ADR
     015532  012767  072152  163030                                                            MOV     #SPACEY,DATA03
3615 015540                                    LET DATA04 := #0         ;FILL CHAR.
     015540  005067  163026                                                                    CLR     DATA04
3616
3617 015544                                    LET R0 := DATA00
     015544  016700  163012                                                                    MOV     DATA00,R0
3618 015550                                    LET R1 := DATA01
     015550  016701  163010                                                                    MOV     DATA01,R1
3619 015554                                    LET R2 := DATA02
     015554  016702  163006                                                                    MOV     DATA02,R2
3620 015560                                    LET R3 := DATA03
     015560  016703  163004                                                                    MOV     DATA03,R3
3621 015564                                    LET R4 := DATA04
     015564  016704  163002                                                                    MOV     DATA04,R4
3622 015570                                    LET R5 := #TRANS
     015570  012705  063436                                                                    MOV     #TRANS,R5
3623
3624 015574                                    LET R0.EXP := #0
     015574  005067  162724                                                                    CLR     R0.EXP
3625 015600                                    LET R1.EXP := #0
     015600  005067  162722                                                                    CLR     R1.EXP
3626 015604                                    LET R2.EXP := #0
     015604  005067  162720                                                                    CLR     R2.EXP
3627 015610                                    LET R3.EXP := #0
     015610  005067  162716                                                                    CLR     R3.EXP
3628 015614                                    LET R4.EXP := R4
     015614  010467  162714                                                                    MOV     R4,R4.EXP
3629 015620                                    LET R5.EXP := R5
     015620  010567  162712                                                                    MOV     R5,R5.EXP
3630 015624                                    LET BFADRS := DATA01
     015624  016767  162734  162710                                                            MOV     DATA01,BFADRS
3631
3632 015632                                    CALL ALTBU1
     015632  004767  040772                                                                    JSR     PC,ALTBU1
3633 015636                                    LET BFADRS := DATA03
     015636  016767  162726  162676                                                            MOV     DATA03,BFADRS
3634 015644                                    CALL ALTBU1
     015644  004767  040760                                                                    JSR     PC,ALTBU1
3635
3636                               ;+
3637                               ; SET UP THE TRANSLATION TABLE
3638                               ;-
3639
3640 015650                                    DECR TEMPO FROM #200 TO #1 BY #1
     015650  012767  000200  162620                                                            MOV     #200,TEMPO
```

```
        015656  000402                                                          BR      $50541
        015660                                                          $50542:
        015660  005367  162612                                                  DEC     TEMP0
        015664                                                          $50541:
        015664  026727  162606  000001                                          CMP     TEMP0,#1
        015672  002402                                                          BLT     $50543
3641    015674                          LET (R5)+ := #0
        015674  005025                                                          CLR     (R5)+
3642    015676                      ENDDEC
        015676  000770                                                          BR      $50542
        015700                                                          $50543:
3643    015700                      LET R5 := #TRANS
        015700  012705  063436                                                  MOV     #TRANS,R5
3644    015704                      LET (R5) :B= #63
        015704  112715  000063                                                  MOVB    #63,(R5)
3645    015710                      LET 252(R5) :B= #314
        015710  112765  000314  000252                                          MOVB    #314,252(R5)
3646    015716                      LET 377(R5) :B= #125
        015716  112765  000125  000377                                          MOVB    #125,377(R5)
3647
3648    015724                      WHILE #BIT07 NOTSETIN TPS DO
        015724                                                          $50544:
        015724  032767  000200  161632                                          BIT     #BIT07,TPS
        015732  001001                                                          BNE     $50545
3649    015734                      ENDDO
        015734  000773                                                          BR      $50544
        015736                                                          $50545:
3650    015736                      LET TPB := #0               ;SEND CHARACTER TO THE XMIT BUFFER
        015736  005067  161624                                                  CLR     TPB
3651    015742                      INLINE<WAIT:>               ;WAIT FOR DONE TO DOUBLE BUFFER
        015742                                                          WAIT:
3652    015742                      INLINE<BIT #BIT07,TPS>   ;  XMIT
        015742  032767  000200  161614                                          BIT #BIT07,TPS
3653    015750                      INLINE<BEQ WAIT>
        015750  001774                                                          BEQ WAIT
3654    015752                      LET TPB := #0                   ;FILL SECOND XMIT BUFFFER
        015752  005067  161610                                                  CLR     TPB
3655    015756                      LET TPS := #BIT6        ;SET INTERRUPT ENABLE
        015756  012767  000100  161600                                          MOV     #BIT6,TPS
3656    015764                      LET PSW := #PR0         ;ALLOW INTERRUPTS
        015764  012767  000000  162004                                          MOV     #PR0,PSW
3657    015772                      INLINE   <LOOP2:>               ;DELAY...
        015772                                                          LOOP2:
3658    015772                      INLINE   <DEC    @#TEMP1>
        015772  005337  000500                                                  DEC     @#TEMP1
3659    015776                      INLINE   <BNE    LOOP2>
        015776  001375                                                          BNE     LOOP2
3660    016000                      INLINE  <MOVTC>
        016000  076032                                                          MOVTC
3661    016002                  UNTIL TEMPO EQ #1 OR DELAY LT #40
        016002  026727  162470  000001                                          CMP     TEMP0,#1
        016010  001404                                                          BEQ     $50546
        016012  026727  162454  000040                                          CMP     DELAY,#40
        016020  002230                                                          BGE     $50540
        016022                                                          $50546:
3662    016022                  IF TEMPO NE #1 THEN
        016022  026727  162450  000001                                          CMP     TEMP0,#1
```

```
     016030  001403                                                       BEQ    $50547
3663 016032                         CALL ERROR          ;PSW BIT08 DID NOT GET SET
     016032  004767  041642                                               JSR    PC,ERROR
3664 016036                      ELSE
     016036  000555                                                       BR     $50550
     016040                                                   $50547:
3665                           ;+
3666                           ; CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS
3667                           ;-
3668
3669 016040                         LET CC := PSW
     016040  016767  161732  162452                                       MOV    PSW,CC
3670 016046                         LET CC := CC CLR.BY #177760
     016046  042767  177760  162444                                       BIC    #177760,CC
3671 016054                         IF CC NE CC.EXP THEN
     016054  026767  162440  162440                                       CMP    CC,CC.EXP
     016062  001402                                                       BEQ    $50551
3672 016064                            CALL ERROR
     016064  004767  041610                                               JSR    PC,ERROR
3673 016070                         ENDIF
     016070                                                   $50551:
3674 016070                         IF R0 NE R0.EXP THEN
     016070  020067  162430                                               CMP    R0,R0.EXP
     016074  001402                                                       BEQ    $50552
3675 016076                            CALL ERROR
     016076  004767  041576                                               JSR    PC,ERROR
3676 016102                         ENDIF
     016102                                                   $50552:
3677 016102                         IF R1 NE R1.EXP THEN
     016102  020167  162420                                               CMP    R1,R1.EXP
     016106  001402                                                       BEQ    $50553
3678 016110                            CALL ERROR
     016110  004767  041564                                               JSR    PC,ERROR
3679 016114                         ENDIF
     016114                                                   $50553:
3680 016114                         IF R2 NE R2.EXP THEN
     016114  020267  162410                                               CMP    R2,R2.EXP
     016120  001402                                                       BEQ    $50554
3681 016122                            CALL ERROR
     016122  004767  041552                                               JSR    PC,ERROR
3682 016126                         ENDIF
     016126                                                   $50554:
3683 016126                         IF R3 NE R3.EXP THEN
     016126  020367  162400                                               CMP    R3,R3.EXP
     016132  001402                                                       BEQ    $50555
3684 016134                            CALL ERROR
     016134  004767  041540                                               JSR    PC,ERROR
3685 016140                         ENDIF
     016140                                                   $50555:
3686 016140                         IF R4 NE R4.EXP THEN
     016140  020467  162370                                               CMP    R4,R4.EXP
     016144  001402                                                       BEQ    $50556
3687 016146                            CALL ERROR
     016146  004767  041526                                               JSR    PC,ERROR
3688 016152                         ENDIF
     016152                                                   $50556:
3689 016152                         IF R5 NE R5.EXP THEN
```

```
        016152  020567  162360                                                  CMP     R5,R5.EXP
        016156  001402                                                          BEQ     $50557
3690    016160                          CALL ERROR
        016160  004767  041514                                                  JSR     PC,ERROR
3691    016164                      ENDIF
        016164                                                          $50557:
3692
3693                            ;+
3694                            ; CHECK ENTIRE DESTINATION BUFFER FOR CORRECT RESULTS
3695                            ;-
3696
3697    016164                      LET R0 := DATA03
        016164  016700  162400                                                  MOV     DATA03,R0
3698    016170                      LET R2 :B= @DATA01
        016170  117702  162370                                                  MOVB    @DATA01,R2
3699    016174                      LET R2 := R2 CLR.BY #177400
        016174  042702  177400                                                  BIC     #177400,R2
3700    016200                      LET R1 :B= TRANS(R2)
        016200  116201  063436                                                  MOVB    TRANS(R2),R1
3701    016204                      LET ERRFLG := #0
        016204  005067  162246                                                  CLR     ERRFLG
3702    016210                      LET R3 := DATA00
        016210  016703  162346                                                  MOV     DATA00,R3
3703    016214                      WHILE R3 NE #0 AND ERRFLG EQ #0 DO
        016214                                                          $50560:
        016214  005703                                                          TST     R3
        016216  001411                                                          BEQ     $50561
        016220  005767  162232                                                  TST     ERRFLG
        016224  001006                                                          BNE     $50561
3704    016226                          IFB (R0)+ NE R1 THEN
        016226  122001                                                          CMPB    (R0)+,R1
        016230  001402                                                          BEQ     $50562
3705    016232                              CALL ERROR
        016232  004767  041442                                                  JSR     PC,ERROR
3706    016236                          ENDIF
        016236                                                          $50562:
3707    016236                          LET R3 := R3 - #1
        016236  005303                                                          DEC     R3
3708    016240                      ENDDO
        016240  000765                                                          BR      $50560
        016242                                                          $50561:
3709    016242                      LET R0 := DATA03 + DATA00
        016242  016700  162322                                                  MOV     DATA03,R0
        016246  066700  162310                                                  ADD     DATA00,R0
3710    016252                      LET R3 := DATA02 - DATA00
        016252  016703  162310                                                  MOV     DATA02,R3
        016256  166703  162300                                                  SUB     DATA00,R3
3711    016262                      REPEAT
        016262                                                          $50563:
3712    016262                          IFB (R0)+ NE DATA04 THEN
        016262  122067  162304                                                  CMPB    (R0)+,DATA04
        016266  001402                                                          BEQ     $50564
3713    016270                              CALL ERROR
        016270  004767  041404                                                  JSR     PC,ERROR
3714    016274                          ENDIF
        016274                                                          $50564:
3715    016274                          LET R3 := R3 - #1
```

```
                016274  005303                                                          DEC     R3
3716            016276                          UNTIL R3 EQ #0 OR ERRFLG NE #0
                016276  005703                                                          TST     R3
                016300  001403                                                          BEQ     $50565
                016302  005767  162150                                                TST     ERRFLG
                016306  001765                                                          BEQ     $50563
                016310                                                          $50565:
3717            016310                          IF DATA02 LO #BUFLEN THEN
                016310  026727  162252  000100                                        CMP     DATA02,#BUFLEN
                016316  103025                                                        BHIS    $50566
3718            016320                              LET TEMP0 := DATA03 + #1777
                016320  016767  162244  162150                                      MOV     DATA03,TEMP0
                016326  062767  001777  162142                                      ADD     #1777,TEMP0
3719            016334                              LET R0 := DATA03 + DATA02
                016334  016700  162230                                              MOV     DATA03,R0
                016340  066700  162222                                              ADD     DATA02,R0
3720            016344                              REPEAT
                016344                                                          $50567:
3721            016344                                  IFB (R0)+ NE BF.EXP THEN
                016344  122067  162170                                              CMPB    (R0)+,BF.EXP
                016350  001402                                                      BEQ     $50570
3722            016352                                      CALL ERROR
                016352  004767  041322                                              JSR     PC,ERROR
3723            016356                                  ENDIF
                016356                                                          $50570:
3724            016356                              UNTIL R0 EQ TEMP0 OR ERRFLG NE #0
                016356  020067  162114                                              CMP     R0,TEMP0
                016362  001403                                                      BEQ     $50571
                016364  005767  162066                                              TST     ERRFLG
                016370  001765                                                      BEQ     $50567
                016372                                                          $50571:
3725            016372                              ENDIF
                016372                                                          $50566:
3726            016372                          ENDIF
                016372                                                          $50550:
3727            016372                          LET PSW := #PR7
                016372  012767  000340  161376                                      MOV     #PR7,PSW
3728            016400                          LET TPVEC := #TPVEC+2
                016400  012767  000066  161456                                      MOV     #TPVEC+2,TPVEC
3729            016406                          LET TPVEC+2 := #0           ;RESTORE CONSOLE TRAP CATCHER
                016406  005067  161454                                              CLR     TPVEC+2
3730            016412                          LET TPS := #0                       ;MAKE SURE INTERRUPT ENABLE IS OFF
                016412  005067  161146                                              CLR     TPS
3731            016416                          CALL RESREG
                016416  004767  040346                                              JSR     PC,RESREG
3732            016422                          LET PSW := SAVPSW
                016422  016767  162166  161346                                      MOV     SAVPSW,PSW
3733            016430                          INLINE <SKPTST:>            -
                016430                                                          SKPTST:
3734            016430                          INLINE  <BR  ILLTST>        ;BR AROUND INT SERVICE RTN
                016430  000421                                                      BR      ILLTST
3735
3736                                    ;+
3737                                    ;       TP INTERRUPT SERVICE ROUTINE
3738                                    ;-
3739
3740 016432                             ROUTINE TPSRVC
```

INTERRUPTABILITY SUBTEST

```
     016432                                                                  TPSRVC:
3741 016432                          LET TPS := #0                    ;TURN OFF INTERRUPT ENABLE
     016432  005067  161126                                                  CLR     TPS
3742 016436                          LET TEMP5 := 2(SP)
     016436  016667  000002  162044                                          MOV     2(SP),TEMP5
3743 016444                          IF #BIT8 SETIN TEMP5 THEN
     016444  032767  000400  162036                                          BIT     #BIT8,TEMP5
     016452  001404                                                         BEQ     $50574
3744 016454                              LET TEMP0 := #1          ;INST WAS SUSPENDED
     016454  012767  000001  162014                                          MOV     #1,TEMP0
3745 016462                          ELSE
     016462  000403                                                         BR      $50575
     016464                                                          $50574:
3746 016464                              LET DELAY := DELAY - #30 ;DECREMENT DELAY COUNTER AND TRY AGAIN
     016464  162767  000030  162000                                          SUB     #30,DELAY
3747 016472                          ENDIF
     016472                                                          $50575:
3748 016472                          INLINE  <RTI>
     016472  000002                                                         RTI
3749
```

INTERRUPTABILITY SUBTEST

```
3751                                    ;;*****************************************************************
3752                                    .SBTTL  ILLEGAL OPCODE SUBTEST
3753                                    ;;*****************************************************************
3754
3755                                    ;+
3756                                    ;       THIS SUBTEST VERIFIES THAT AN ILLEGAL CIS CLASS OPCODE
3757                                    ;       (ILLOP=76033) CAUSES A RESERVED INSTRUCTION TRAP (TRAP 10)
3758                                    ;-
3759
3760 016474                            ROUTINE ILLTST
    016474                                                                                  ILLTST:
3761 016474                               INLINE       <SCOPE>
    016474  000004                                                                                  SCOPE
3762 016476                               INLINE  <NOP>
    016476  000240                                                                                  NOP
3763 016500                               LET $TIMES := #20             ;ITERATION COUNT
    016500  012767  000020  161676                                                          MOV     #20,$TIMES
3764 016506                               LET ERRFLG := #0              ;RESET ERROR FLAG
    016506  005067  161744                                                                  CLR     ERRFLG
3765 016512                               LET ERR := #3                         ;ERROR MSG #
    016512  012767  000003  162032                                                          MOV     #3,ERR
3766 016520                               LET SAVPSW := PSW
    016520  016767  161252  162066                                                          MOV     PSW,SAVPSW
3767 016526                               LET PSW := #PR7                       ;LOCK OUT INTERRUPTS
    016526  012767  000340  161242                                                          MOV     #PR7,PSW
3768 016534                               LET RESVEC := #RESRTN
    016534  012767  016562  161246                                                          MOV     #RESRTN,RESVEC
3769 016542                               INLINE  <ILLOP>                       ;DO ILLEGAL CODE
    016542  076033                                                                          ILLOP
3770 016544                               CALL ERROR                    ;DIDNT TRAP, ERROR
    016544  004767  041130                                                                  JSR     PC,ERROR
3771 016550                               LET RESVEC := #ILLRES                 ;RESTORE RESERVED INSTRUCTION TRAP VECTOR
    016550  012767  056406  161232                                                          MOV     #ILLRES,RESVEC
3772 016556                               INLINE  <JMP CIS4END>         ;MODULE EXIT
    016556  000167  000040                                                                  JMP     CIS4END
3773
3774                                    ;+
3775                                    ;       TRAP HANDLER
3776                                    ;-
3777
3778 016562                            ROUTINE RESRTN
    016562                                                                                  RESRTN:
3779 016562                               INLINE  <NOP>
    016562  000240                                                                                  NOP
3780 016564                               IF ERRFLG EQ #0 THEN
    016564  005767  161666                                                                  TST     ERRFLG
    016570  001004                                                                          BNE     $50602
3781 016572                                   POP TEMP0,TEMP0           ;REMOVE PC,PSW PUSHED BY TRAP
    016572  012667  161700                                                                  MOV     (SP)+,TEMP0
    016576  012667  161674                                                                  MOV     (SP)+,TEMP0
3782 016602                               ENDIF
    016602                                                                                  $50602:
3783 016602                               LET ERRFLG := #0              ;ENSURE ERROR FLAG IS RESET
    016602  005067  161650                                                                  CLR     ERRFLG
3784 016606                               LET PSW := SAVPSW
    016606  016767  162002  161162                                                          MOV     SAVPSW,PSW
3785 016614                               LET RESVEC := #ILLRES                 ;RESTORE RESERVED INSTRUCTION TRAP VECTOR
```

SEQ 0113

ILLEGAL OPCODE SUBTEST

```
        016614  012767  056406  161166
   3786 016622                          INLINE  <CIS4END:>          MOV     #ILLRES,RESVEC
        016622                                                       CIS4END:
   3787
   3788
```

```
                                ;;*****************************************************************
                                ;;*****************************************************************
                                ;**
                                ;                C I S 5     M O D U L E
                                ;
                                ;  THIS MODULE TESTS THE ARITHMETIC CONVERT INSTRUCTIONS FOR
                                ;  PACKED, NUMERIC, AND LONG INTEGER DATA TYPES: CVTLP, CVTLN,
                                ;  CVTPL, CVTNL, CVTPN, AND CVTNP.
                                ;--
                                ;;*****************************************************************
                                ;;*****************************************************************


                                .SBTTL  CIS5 INITIALIZATION
                                ;;*****************************************************************
                                ;
                                ;  CIS5 INITIALIZATION
                                ;
                                ;;*****************************************************************

                                ;IF SWR<5:0> ARE CLEARED OR IF SWR<1> IS SET THEN EXECUTE
                                ;  THE CIS5 MODULE; OTHERWISE, SKIP TO THE NEXT CIS MODULE.

016622                             INLINE   <SCOPE>
016622  000004                                                                    SCOPE
                                   INLINE <CKSWR>
016624  104406                                                                    CKSWR
                                   LET TEMPO := @SWR CLR.BY #177700
016626  017767  161574  161642                                                   MOV     @SWR,TEMPO
016634  042767  177700  161634                                                   BIC     #177700,TEMPO
                                IF TEMPO NE #0 THEN
016642  005767  161630                                                           TST     TEMPO
016646  001406                                                                   BEQ     $50603
                                    IF #BIT1 NOTSETIN TEMPO THEN
016650  032767  000002  161620                                                   BIT     #BIT1,TEMPO
016656  001002                                                                   BNE     $50604
                                        INLINE   <JMP   CIS5END>
016660  000167  013736                                                           JMP   CIS5END
                                    ENDIF
016664                                                                  $50604:
                                ENDIF
016664                                                                  $50603:
                                LET CTL := #5                       ;CONTROL CHIP NUMBER
016664  012767  000005  161654                                                   MOV     #5,CTL
                                LET ERRFLG := #0                ;INITIALIZE ERROR FLAG
016672  005067  161560                                                           CLR     ERRFLG
                                LET PSW := #340
016676  012767  000340  161072                                                   MOV     #340,PSW
                                LET ERR := #1
016704  012767  000001  161640                                                   MOV     #1,ERR

```

```
                                .SBTTL  CVTLP
                                ;;*****************************************************************
                                ;
                                ;  CVTLP TEST
                                ;
                                ;;*****************************************************************

                                ROUTINE XCVTLP
                                                                                        XCVTLP:
                                   INLINE       <SCOPE>
016712  000004                                                                          SCOPE
                                   INLINE       <NOP>
016714  000240                                                                          NOP
                                   LET $TIMES := #20            ;ITERATION COUNT
016716  012767  000020  161460                                                        MOV     #20,$TIMES
                                   LET COUNT := TCVTLP          ;GET TEST COUNT
016724  016767  001274  161536                                                        MOV     TCVTLP,COUNT
                                   LET TSTCAS := #TCVTLP+2              ;POINT TO FIRST TEST CASE
016732  012767  020226  161604                                                        MOV     #TCVTLP+2,TSTCAS
                                ;REPEAT
                                   INLINE       <$5077:>
016740                                                                              $5077:

                                ;+
                                ; GET TEST DATA FROM TABLE
                                ;-

                                   LET R0 := TSTCAS
016740  016700  161600                                                              MOV     TSTCAS,R0
                                   LET DATA00 := (R0)+                  ;DST.DSCR
016744  012067  161612                                                              MOV     (R0)+,DATA00
                                   LET DATA01 := (R0)+                  ;LONG INTEGER HI
016750  012067  161610                                                              MOV     (R0)+,DATA01
                                   LET DATA02 := (R0)+                  ;LONG INTEGER LO
016754  012067  161606                                                              MOV     (R0)+,DATA02

                                ;+
                                ; PUT EXPECTED RESULTS INTO DS.EXP AND CC.EXP
                                ;-

                                   CALL CLRDSB                  ;CLEAR DECIMAL STRING BUFFER AREAS
016760  004767  037710                                                              JSR     PC,CLRDSB
                                   LET R2 := DATA01
016764  016702  161574                                                              MOV     DATA01,R2
                                   LET R3 := DATA02
016770  016703  161572                                                              MOV     DATA02,R3
                                   CALL CVTLI                   ;CONVERT LONG INTEGER TO AN INTERMEDIATE FORM
016774  004767  012304                                                              JSR     PC,CVTLI
                                   LET CC.EXP := #0
017000  005067  161516                                                              CLR     CC.EXP
                                   LET DSLEN2 := DATA00 CLR.BY #177740          ;GET DST.LEN
017004  016767  161552  161504                                                      MOV     DATA00,DSLEN2
017012  042767  177740  161476                                                      BIC     #177740,DSLEN2
                                   IF DSLEN2 EQ #0 THEN                 ;DST.LEN = 0
017020  005767  161472                                                              TST     DSLEN2
017024  001014                                                                      BNE     $50607
                                      LET CC.EXP := CC.EXP SET.BY #ZBIT
```

CVTLP

```
      017026  052767  000004  161466                                                          BIS     #ZBIT,CC.EXP
 3864 017034                          LET DS.EXP :B= #14
      017034  112767  000014  045544                                                          MOVB    #14,DS.EXP
 3865 017042                          IF DSLEN1 HI #0 THEN
      017042  005767  161446                                                                  TST     DSLEN1
      017046  101403                                                                         BLOS    $50610
 3866 017050                              LET CC.EXP := CC.EXP SET.BY #VBIT
      017050  052767  000002  161444                                                          BIS     #VBIT,CC.EXP
 3867 017056                          ENDIF
      017056                                                                          $50610:
 3868 017056                     ENDIF
      017056                                                                          $50607:
 3869 017056                     IF DSLEN2 NE #0 THEN                    ;DST.LEN > 0
      017056  005767  161434                                                                  TST     DSLEN2
      017062  001577                                                                         BEQ     $50611
 3870 017064                          IF DSLEN1 EQ #0 THEN                        ;SRC.LEN = 0
      017064  005767  161424                                                                  TST     DSLEN1
      017070  001017                                                                         BNE     $50612
 3871 017072                              LET TEMP0 := DSLEN2
      017072  016767  161420  161376                                                          MOV     DSLEN2,TEMP0
 3872 017100                              INLINE  <ASR  TEMP0>
      017100  006267  161372                                                                 ASR  TEMP0
 3873 017104                              LET R0 := #DS.EXP + TEMP0
      017104  012700  064606                                                                 MOV     #DS.EXP,R0
      017110  066700  161362                                                                 ADD     TEMP0,R0
 3874 017114                              LET (R0) :B= #14
      017114  112710  000014                                                                 MOVB    #14,(R0)
 3875 017120                              LET CC.EXP := CC.EXP SET.BY #ZBIT
      017120  052767  000004  161374                                                          BIS     #ZBIT,CC.EXP
 3876 017126                     ELSE
      017126  000555                                                                         BR      $50613
      017130                                                                          $50612:
 3877 017130                          IF DSLEN2 EQ DSLEN1 THEN                ;DST.LEN = SRC.LEN > 0
      017130  026767  161362  161356                                                          CMP     DSLEN2,DSLEN1
      017136  001010                                                                         BNE     $50614
 3878 017140                              LET R1 := #DSBUF1 + #10. - DSLEN1       ;SRC POINTER
      017140  012701  064440                                                                 MOV     #DSBUF1,R1
      017144  062701  000012                                                                 ADD     #10.,R1
      017150  166701  161340                                                                 SUB     DSLEN1,R1
 3879 017154                              LET R2 := #DS.EXP                       ;DST POINTER
      017154  012702  064606                                                                 MOV     #DS.EXP,R2
 3880 017160                          ENDIF
      017160                                                                          $50614:
 3881 017160                          IF DSLEN2 LT DSLEN1 THEN              ;DST.LEN < SRC.LEN
      017160  026767  161332  161326                                                          CMP     DSLEN2,DSLEN1
      017166  002016                                                                         BGE     $50615
 3882 017170                              LET R1 := #DSBUF1 + #10. - DSLEN2
      017170  012701  064440                                                                 MOV     #DSBUF1,R1
      017174  062701  000012                                                                 ADD     #10.,R1
      017200  166701  161312                                                                 SUB     DSLEN2,R1
 3883 017204                              LET R2 := #DS.EXP
      017204  012702  064606                                                                 MOV     #DS.EXP,R2
 3884 017210                              LET CC.EXP := CC.EXP SET.BY #VBIT
      017210  052767  000002  161304                                                          BIS     #VBIT,CC.EXP
 3885 017216                              LET DSLEN1 := DSLEN2
      017216  016767  161274  161270                                                          MOV     DSLEN2,DSLEN1
 3886 017224                          ENDIF
```

```
                                                                                $50615:
                        IF DSLEN2 GT DSLEN1 THEN                        ;DST.LEN > SRC.LEN
017224  026767  161266  161262                                                    CMP     DSLEN2,DSLEN1
017232  003426                                                                    BLE     $50616
                            LET TEMP0 := DSLEN1
017234  016767  161254  161234                                                    MOV     DSLEN1,TEMP0
                            INLINE  <ASR  TEMP0>
017242  006267  161230                                                            ASR  TEMP0
                            LET TEMP1 := DSLEN2
017246  016767  161244  161224                                                    MOV     DSLEN2,TEMP1
                            INLINE  <ASR  TEMP1>
017254  006267  161220                                                            ASR  TEMP1
                            LET R2 := @DS.EXP + TEMP1 - TEMP0
017260  012702  064606                                                            MOV     @DS.EXP,R2
017264  066702  161210                                                            ADD     TEMP1,R2
017270  166702  161202                                                            SUB     TEMP0,R2
                            LET R1 := @DSBUF1 + #10. - DSLEN1           ;SRC POINTER
017274  012701  064440                                                            MOV     @DSBUF1,R1
017300  062701  000012                                                            ADD     #10.,R1
017304  166701  161204                                                            SUB     DSLEN1,R1
                        ENDIF
017310                                                                          $50616:

                    ;LOAD REMAINING DIGITS INTO DS.EXP

                        IF DSLEN1 HI #1 THEN
017310  026727  161200  000001                                                    CMP     DSLEN1,#1
017316  101431                                                                    BLOS    $50617

                            IF #1 NOTSETIN DSLEN1 THEN
017320  032767  000001  161166                                                    BIT     #1,DSLEN1
017326  001003                                                                    BNE     $50620

                                LET (R2)+ :B= (R1)+                 ;OUTPUT TO DEST. BUFFER
017330  112122                                                                    MOVB    (R1)+,(R2)+

                                LET DSLEN1 := DSLEN1 - #1
017332  005367  161156                                                            DEC     DSLEN1

                            ENDIF
017336                                                                          $50620:

                            LET R0 := #0                            ;USE R0 TO ASSEMBLE DATA
017336  005000                                                                  CLR     R0

                            WHILE DSLEN1 HI #1 DO
017340                                                                          $50621:
017340  026727  161150  000001                                                    CMP     DSLEN1,#1
017346  101415                                                                    BLOS    $50622

                                LET R0 :B= (R1)+
017350  112100                                                                    MOVB    (R1)+,R0

                                INLINE <ASH  #4,R0>
017352  072027  000004                                                            ASH  #4,R0

                                LET TEMP0 :B= R0 OR (R1)+           ;OUTPUT TO DST. BUFFER
017356  110067  161114                                                            MOVB    R0,TEMP0
017362  152167  161110                                                            BISB    (R1)+,TEMP0

                                LET (R2)+ :B= TEMP0
017366  116722  161104                                                            MOVB    TEMP0,(R2)+

                                LET DSLEN1 := DSLEN1 - #2
017372  162767  000002  161114                                                    SUB     #2,DSLEN1

                            ENDDO
017400  000757                                                                    BR      $50621
017402                                                                          $50622:
```

```
3911 017402                                    ENDIF
     017402                                                          $50617:
3912 017402                                    LET R0 :B= (R1)+
     017402  112100                                                         MOVB    (R1)+,R0
3913 017404                                    INLINE <ASH  #4,R0>
     017404  072027  000004                                                 ASH  #4,R0
3914 017410                                    IF DATA00 HIS #70000 THEN   ;*** UNSIGNED PACKED ***
     017410  026727  161146  070000                                         CMP     DATA00,#70000
     017416  103404                                                         BLO     $50623
3915 017420                                        LET (R2) :B= R0 OR #17
     017420  110012                                                         MOVB    R0,(R2)
     017422  152712  000017                                                 BISB    #17,(R2)
3916 017426                                    ELSE
     017426  000415                                                         BR      $50624
     017430                                                          $50623:
3917 017430                                        IF SIGNO EQ #0 THEN     ;*** SIGNED PACKED ***
     017430  005767  161014                                                 TST     SIGNO
     017434  001004                                                         BNE     $50625
3918 017436                                            LET (R2) :B= R0 OR #14      ;SIGN IS +
     017436  110012                                                         MOVB    R0,(R2)
     017440  152712  000014                                                 BISB    #14,(R2)
3919 017444                                        ELSE
     017444  000406                                                         BR      $50626
     017446                                                          $50625:
3920 017446                                            LET (R2) :B= R0 OR #15      ;SIGN IS -
     017446  110012                                                         MOVB    R0,(R2)
     017450  152712  000015                                                 BISB    #15,(R2)
3921 017454                                            LET CC.EXP := CC.EXP SET.BY #NBIT
     017454  052767  000010  161040                                         BIS     #NBIT,CC.EXP
3922 017462                                        ENDIF
     017462                                                          $50626:
3923 017462                                    ENDIF
     017462                                                          $50624:
3924 017462                                ENDIF
     017462                                                          $50613:
3925 017462                            ENDIF
     017462                                                          $50611:
3926
3927                                   ;+
3928                                   ; SET UP GPR'S AND EXPECTED GPR'S.
3929                                   ;-
3930
3931 017462                            LET R0 := DATA00
     017462  016700  161074                                                 MOV     DATA00,R0
3932 017466                            LET R1 := #DS.DST                   ;DST. ADR
     017466  012701  064544                                                 MOV     #DS.DST,R1
3933 017472                            LET R2 := DATA01
     017472  016702  161066                                                 MOV     DATA01,R2
3934 017476                            LET R3 := DATA02
     017476  016703  161064                                                 MOV     DATA02,R3
3935 017502                            IF #1 SETIN COUNT THEN
     017502  032767  000001  160760                                         BIT     #1,COUNT
     017510  001405                                                         BEQ     $50627
3936 017512                                LET R4 := #177777
     017512  012704  177777                                                 MOV     #177777,R4
3937 017516                                LET R5 := #177777
     017516  012705  177777                                                 MOV     #177777,R5
```

```
                                        ELSE
017522  000402                                                          BR      $50630
017524                                                          $50627:
                                            LET R4 := #0
017524  005004                                                          CLR     R4
                                            LET R5 := #0
017526  005005                                                          CLR     R5
                                        ENDIF
017530                                                          $50630:
                                        LET R0.EXP := R0
017530  010067  160770                                                  MOV     R0,R0.EXP
                                        LET R1.EXP := R1
017534  010167  160766                                                  MOV     R1,R1.EXP
                                        LET R2.EXP := #0
017540  005067  160764                                                  CLR     R2.EXP
                                        LET R3.EXP := #0
017544  005067  160762                                                  CLR     R3.EXP
                                        LET R4.EXP := R4
017550  010467  160760                                                  MOV     R4,R4.EXP
                                        LET R5.EXP := R5
017554  010567  160756                                                  MOV     R5,R5.EXP

                                    ;*
                                    ; DO THE CIS INSTRUCTION
                                    ;-

                                       INLINE <CVTLP>
017560  076077                                                          CVTLP

                                    ;*
                                    ;  CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.
                                    ;-

                                        LET CC := PSW
017562  016767  160210  160730                                          MOV     PSW,CC
                                        LET CC := CC CLR.BY #177760
017570  042767  177760  160722                                        BIC     #177760,CC
                                        IF CC NE CC.EXP THEN
017576  026767  160716  160716                                        CMP     CC,CC.EXP
017604  001402                                                          BEQ     $50631
                                            CALL ERROR
017606  004767  040066                                                  JSR     PC,ERROR
                                        ENDIF
017612                                                          $50631:
                                        IF R0 NE R0.EXP THEN
017612  020067  160706                                                  CMP     R0,R0.EXP
017616  001402                                                          BEQ     $50632
                                            CALL ERROR
017620  004767  040054                                                  JSR     PC,ERROR
                                        ENDIF
017624                                                          $50632:
                                        IF R1 NE R1.EXP THEN
017624  020167  160676                                                  CMP     R1,R1.EXP
017630  001402                                                          BEQ     $50633
                                            CALL ERROR
017632  004767  040042                                                  JSR     PC,ERROR
                                        ENDIF
```

```
                017636
3970            017636                          IF R2 NE R2.EXP THEN
                017636  020267  160666                                          $50633:
                017642  001402                                                          CMP     R2,R2.EXP
                                                                                      BEQ     $50634
3971            017644                              CALL ERROR
                017644  004767  040030                                                  JSR     PC,ERROR
3972            017650                          ENDIF
                017650                                                          $50634:
3973            017650                          IF R3 NE R3.EXP THEN
                017650  020367  160656                                                  CMP     R3,R3.EXP
                017654  001402                                                          BEQ     $50635
3974            017656                              CALL ERROR
                017656  004767  040016                                                  JSR     PC,ERROR
3975            017662                          ENDIF
                017662                                                          $50635:
3976            017662                          IF R4 NE R4.EXP THEN
                017662  020467  160646                                                  CMP     R4,R4.EXP
                017666  001402                                                          BEQ     $50636
3977            017670                              CALL ERROR
                017670  004767  040004                                                  JSR     PC,ERROR
3978            017674                          ENDIF
                017674                                                          $50636:
3979            017674                          IF R5 NE R5.EXP THEN
                017674  020567  160636                                                  CMP     R5,R5.EXP
                017700  001402                                                          BEQ     $50637
3980            017702                              CALL ERROR
                017702  004767  037772                                                  JSR     PC,ERROR
3981            017706                          ENDIF
                017706                                                          $50637:
3982
3983                                        ;+
3984                                        ; COMPARE THE EXPECTED DECIMAL STRING BUFFER
3985                                        ; WITH ACTUAL RESULTS.
3986                                        ;-
3987
3988            017706                          LET R0 := #DS.DST
                017706  012700  064544                                                  MOV     #DS.DST,R0
3989            017712                          LET R1 := #DS.EXP
                017712  012701  064606                                                  MOV     #DS.EXP,R1
3990            017716                          LET ERRFLG := #0
                017716  005067  160534                                                  CLR     ERRFLG
3991            017722                          WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO
                017722                                                          $50640:
                017722  020027  064604                                                  CMP     R0,#DS.DST+40
                017726  001410                                                          BEQ     $50641
                017730  005767  160522                                                  TST     ERRFLG
                017734  001005                                                          BNE     $50641
3992            017736                              IFB (R0)+ NE (R1)+ THEN
                017736  122021                                                          CMPB    (R0)+,(R1)+
                017740  001402                                                          BEQ     $50642
3993            017742                                  CALL ERROR
                017742  004767  037732                                                  JSR     PC,ERROR
3994            017746                              ENDIF
                017746                                                          $50642:
3995            017746                          ENDDO
                017746  000765                                                          BR      $50640
                017750                                                          $50641:
```

CVTLP

```
 3996 017750                            LET TSTCAS := TSTCAS + #6
      017750  062767  000006  160566                                    ADD     #6,TSTCAS
 3997 017756                            LET COUNT := COUNT - #1
      017756  005367  160506                                           DEC     COUNT
 3998                                  ;UNTIL COUNT EQ #0
 3999 017762                            IF COUNT NE #0 THEN
      017762  005767  160502                                           TST     COUNT
      017766  001402                                                   BEQ     $50643
 4000 017770                                INLINE <JMP $5077>
      017770  000167  176744                                           JMP $5077
 4001 017774                            ENDIF
      017774                                                  $50643:
 4002
 4003                                  ;+
 4004                                  ; DO ONE IN-LINE CASE - CVTLPI
 4005                                  ;-
 4006
 4007 017774                            CALL CLRREG
      017774  004767  036440                                           JSR     PC,CLRREG
 4008 020000                            CALL CLRDSB
      020000  004767  036670                                           JSR     PC,CLRDSB
 4009 020004                            INLINE  <CVTLPI>
      020004  076177                                                   CVTLPI
 4010 020006                            INLINE  <.WORD       TCVLPI>
      020006  020336                                                   .WORD   TCVLPI
 4011 020010                            INLINE  <.WORD       TCVLPI+4>
      020010  020342                                                   .WORD   TCVLPI+4
 4012 020012                            LET CC := PSW
      020012  016767  157760  160500                                   MOV     PSW,CC
 4013 020020                            LET CC := CC CLR.BY #177760
      020020  042767  177760  160472                                   BIC     #177760,CC
 4014 020026                            IF CC NE #0 THEN
      020026  005767  160466                                           TST     CC
      020032  001402                                                   BEQ     $50644
 4015 020034                                CALL ERROR
      020034  004767  037640                                           JSR     PC,ERROR
 4016 020040                            ENDIF
      020040                                                  $50644:
 4017 020040                            IF R0 NE #0 THEN
      020040  005700                                                   TST     R0
      020042  001402                                                   BEQ     $50645
 4018 020044                                CALL ERROR
      020044  004767  037630                                           JSR     PC,ERROR
 4019 020050                            ENDIF
      020050                                                  $50645:
 4020 020050                            IF R1 NE #0 THEN
      020050  005701                                                   TST     R1
      020052  001402                                                   BEQ     $50646
 4021 020054                                CALL ERROR
      020054  004767  037620                                           JSR     PC,ERROR
 4022 020060                            ENDIF
      020060                                                  $50646:
 4023 020060                            IF R2 NE #0 THEN
      020060  005702                                                   TST     R2
      020062  001402                                                   BEQ     $50647
 4024 020064                                CALL ERROR
      020064  004767  037610                                           JSR     PC,ERROR
```

CVTLP

```
4025 020070                            ENDIF
     020070
4026 020070                            IF R3 NE #0 THEN
     020070  005703
     020072  001402
4027 020074                                CALL ERROR
     020074  004767  037600
4028 020100                            ENDIF
     020100
4029 020100                            IF R4 NE #0 THEN
     020100  005704
     020102  001402
4030 020104                                CALL ERROR
     020104  004767  037570
4031 020110                            ENDIF
     020110
4032 020110                            IF R5 NE #0 THEN
     020110  005705
     020112  001402
4033 020114                                CALL ERROR
     020114  004767  037560
4034 020120                            ENDIF
     020120
4035 020120                            LET R0 := #DS.DST
     020120  012700  064544
4036 020124                            IFB (R0)+ NE #1 THEN
     020124  122027  000001
     020130  001402
4037 020132                                CALL ERROR
     020132  004767  037542
4038 020136                            ENDIF
     020136
4039 020136                            REPEAT
     020136
4040 020136                                IFB (R0)+ NE #21 THEN
     020136  122027  000021
     020142  001402
4041 020144                                    CALL ERROR
     020144  004767  037530
4042 020150                                ENDIF
     020150
4043 020150                            UNTIL R0 EQ #DS.DST+5
     020150  020027  064551
     020154  001370
4044 020156                            IFB (R0)+ NE #37 THEN
     020156  122027  000037
     020162  001402
4045 020164                                CALL ERROR
     020164  004767  037510
4046 020170                            ENDIF
     020170
4047 020170                            LET ERRFLG := #0
     020170  005067  160262
4048 020174                            REPEAT
     020174
4049 020174                                IFB (R0)+ NE #0 THEN
     020174  105720
```

```
$50647:
        TST     R3
        BEQ     $50650
        JSR     PC,ERROR
$50650:
        TST     R4
        BEQ     $50651
        JSR     PC,ERROR
$50651:
        TST     R5
        BEQ     $50652
        JSR     PC,ERROR
$50652:
        MOV     #DS.DST,R0
        CMPB    (R0)+,#1
        BEQ     $50653
        JSR     PC,ERROR
$50653:
$50654:
        CMPB    (R0)+,#21
        BEQ     $50655
        JSR     PC,ERROR
$50655:
        CMP     R0,#DS.DST+5
        BNE     $50654
        CMPB    (R0)+,#37
        BEQ     $50656
        JSR     PC,ERROR
$50656:
        CLR     ERRFLG
$50657:
        TSTB    (R0)+
```

```
     020176  001402                                                          BEQ     $50660
4050 020200                           CALL ERROR
     020200  004767  037474                                                  JSR     PC,ERROR
4051 020204                       ENDIF
     020204                                                         $50660:
4052 020204                   UNTIL R0 EQ #DS.DST+40 OR ERRFLG NE #0
     020204  020027  064604                                                  CMP     R0,#DS.DST+40
     020210  001403                                                          BEQ     $50661
     020212  005767  160240                                                  TST     ERRFLG
     020216  001766                                                          BEQ     $50657
     020220                                                         $50661:
4053 020220                   INLINE        <JMP XCVTLN>
     020220  000167  000122                                                  JMP XCVTLN
4054
4055                        ;+
4056                        ; TABLE OF TEST CASES FOR CVTLP
4057                        ;-
4058
4059 020224  000014         TCVTLP: .WORD   12.             ;# OF TEST CASES FOR CVTLP
4060
4061 020226  060004                 .WORD   60004                   ;DST. DATA TYPE AND LENGTH
4062 020230  000000                 .WORD   0                       ;LONG INTEGER HIGH & SIGN
4063 020232  000000                 .WORD   0                       ;LONG INTEGER LOW
4064
4065 020234  060020                 .WORD   60020           ;TEST CASE NUMBER 2
4066 020236  000000                 .WORD   0
4067 020240  001762                 .WORD   1762
4068
4069 020242  060000                 .WORD   60000           ;TEST CASE NUMBER 3
4070 020244  000000                 .WORD   0
4071 020246  152361                 .WORD   152361
4072
4073 020250  060010                 .WORD   60010           ;TEST CASE NUMBER 4
4074 020252  074377                 .WORD   74377
4075 020254  171700                 .WORD   171700
4076
4077 020256  060006                 .WORD   60006           ;TEST CASE NUMBER 5
4078 020260  000000                 .WORD   0
4079 020262  075147                 .WORD   75147
4080
4081 020264  060020                 .WORD   60020           ;TEST CASE NUMBER 6
4082 020266  077777                 .WORD   77777
4083 020270  177777                 .WORD   177777
4084
4085 020272  060015                 .WORD   60015           ;TEST CASE NUMBER 7
4086 020274  100000                 .WORD   100000
4087 020276  000000                 .WORD   0
4088
4089 020300  060010                 .WORD   60010           ;TEST CASE NUMBER 8
4090 020302  177755                 .WORD   177755
4091 020304  130070                 .WORD   130070
4092
4093 020306  070010                 .WORD   70010           ;TEST CASE NUMBER 9
4094 020310  177755                 .WORD   177755
4095 020312  130070                 .WORD   130070
4096
4097 020314  060037                 .WORD   60037           ;TEST CASE NUMBER 10
```

```
 4098 020316  177777                        .WORD   177777
 4099 020320  177775                        .WORD   177775
 4100
 4101 020322  060003                        .WORD   60003           ;TEST CASE NUMBER 11
 4102 020324  000000                        .WORD   0
 4103 020326  000257                        .WORD   257
 4104
 4105 020330  060000                        .WORD   60000           ;TEST CASE NUMBER 12
 4106 020332  000000                        .WORD   0
 4107 020334  000000                        .WORD   0
 4108
 4109
 4110 020336                        TCVLPI:                         ;IN-LINE TEST CASE
 4111 020336  070012                        .WORD   70012                   ;DST TYPE = UNSIGNED PACKED, LENGTH = 12
 4112 020340  064544                        .WORD   DS.DST                  ;DST.ADDRESS
 4113 020342  032707                        .WORD   32707                   ;LONG INTEGER LOW WORD +
 4114 020344  041072                        .WORD   41072                   ;  LONG INTEGER HIGH WORD AND SIGN
 4115
 4116                               ;  = 1111111111 IN BCD,
 4117                               ;  = 1,21,21,21,21,21,37 IN PACKED BYTES
 4118
```

```
 4120                                   .SBTTL  CVTLN
 4121                                   ;;*****************************************************************
 4122                                   ;
 4123                                   ;  CVTLN TEST
 4124                                   ;
 4125                                   ;;*****************************************************************
 4126
 4127 020346                            ROUTINE XCVTLN
      020346                                                                          XCVTLN:
 4128 020346                               INLINE       <SCOPE>
      020346  000004                                                                          SCOPE
 4129 020350                               INLINE       <NOP>
      020350  000240                                                                          NOP
 4130 020352                               LET $TIMES := #20            ;ITERATION COUNT
      020352  012767  000020  160024                                                  MOV     #20,$TIMES
 4131 020360                               LET COUNT := TCVTLN          ;GET TEST COUNT
      020360  016767  001400  160102                                                  MOV     TCVTLN,COUNT
 4132 020366                               LET TSTCAS := #TCVTLN+2              ;POINT TO FIRST TEST CASE
      020366  012767  021766  160150                                                  MOV     #TCVTLN+2,TSTCAS
 4133                                   ;REPEAT
 4134 020374                               INLINE       <$5057:>
      020374                                                                          $5057:
 4135
 4136                                   ;+
 4137                                   ; GET TEST DATA FROM TABLE
 4138                                   ;-
 4139
 4140 020374                               LET R0 := TSTCAS
      020374  016700  160144                                                  MOV     TSTCAS,R0
 4141 020400                               LET DATA00 := (R0)+                  ;DST. DATA TYPE AND LENGTH
      020400  012067  160156                                                  MOV     (R0)+,DATA00
 4142 020404                               LET DATA01 := (R0)+                  ;LONG INTEGER HIGH
      020404  012067  160154                                                  MOV     (R0)+,DATA01
 4143 020410                               LET DATA02 := (R0)+                  ;LONG INTEGER LOW
      020410  012067  160152                                                  MOV     (R0)+,DATA02
 4144
 4145                                   ;+
 4146                                   ; PUT EXPECTED RESULTS INTO DS.EXP AND CC.EXP
 4147                                   ;-
 4148
 4149 020414                               CALL CLRDSB                  ;CLEAR DECIMAL STRING BUFFER AREA
      020414  004767  036254                                                  JSR     PC,CLRDSB
 4150 020420                               LET R2 := DATA01
      020420  016702  160140                                                  MOV     DATA01,R2
 4151 020424                               LET R3 := DATA02
      020424  016703  160136                                                  MOV     DATA02,R3
 4152 020430                               CALL CVTLI                   ;CONVERT LONG INTEGER TO AN INTERMEDIATE FORM
      020430  004767  010650                                                  JSR     PC,CVTLI
 4153 020434                               LET CC.EXP := #0
      020434  005067  160062                                                  CLR     CC.EXP
 4154 020440                               LET DSLEN2 := DATA00 CLR.BY #177740          ;GET DST.LEN
      020440  016767  160116  160050                                                  MOV     DATA00,DSLEN2
      020446  042767  177740  160042                                                  BIC     #177740,DSLEN2
 4155 020454                               LET DATTYP := DATA00 CLR.BY #107777          ;GET DATA TYPE
      020454  016767  160102  160002                                                  MOV     DATA00,DATTYP
      020462  042767  107777  157774                                                  BIC     #107777,DATTYP
 4156 020470                               IF DSLEN2 EQ #0 THEN                 ;DST.LEN = 0
```

```
        020470  005767  160022                                                  TST     DSLEN2
        020474  001053                                                          BNE     $50664
4157    020476                          LET CC.EXP := CC.EXP SET.BY #ZBIT
        020476  052767  000004  160016                                          BIS     #ZBIT,CC.EXP
4158    020504                          IF DSLEN1 HI #0 THEN                    ;INTERMEDIATE LENGTH > 0
        020504  005767  160004                                                  TST     DSLEN1
        020510  101403                                                          BLOS    $50665
4159    020512                              LET CC.EXP := CC.EXP SET.BY #VBIT
        020512  052767  000002  160002                                          BIS     #VBIT,CC.EXP
4160    020520                          ENDIF
        020520                                                                  $50665:
4161    020520                          IF DATTYP EQ #40000 THEN                ;TRAILING SEPARATE
        020520  026727  157740  040000                                          CMP     DATTYP,#40000
        020526  001015                                                          BNE     $50666
4162    020530                              IF SIGNO EQ #0 THEN
        020530  005767  157714                                                  TST     SIGNO
        020534  001004                                                          BNE     $50667
4163    020536                                  LET DS.EXP :B= #53
        020536  112767  000053  044042                                          MOVB    #53,DS.EXP
4164    020544                              ELSE
        020544  000406                                                          BR      $50670
        020546                                                                  $50667:
4165    020546                                  LET DS.EXP :B= #55
        020546  112767  000055  044032                                          MOVB    #55,DS.EXP
4166    020554                                  LET CC.EXP := CC.EXP SET.BY #NBIT
        020554  052767  000010  157740                                          BIS     #NBIT,CC.EXP
4167    020562                              ENDIF
        020562                                                                  $50670:
4168    020562                          ENDIF
        020562                                                                  $50666:
4169    020562                          IF DATTYP EQ #50000 THEN                ;LEADING SEPARATE
        020562  026727  157676  050000                                          CMP     DATTYP,#50000
        020570  001015                                                          BNE     $50671
4170    020572                              IF SIGNO EQ #0 THEN
        020572  005767  157652                                                  TST     SIGNO
        020576  001004                                                          BNE     $50672
4171    020600                                  LET DS.EXP-1 :B= #53
        020600  112767  000053  043777                                          MOVB    #53,DS.EXP-1
4172    020606                              ELSE
        020606  000406                                                          BR      $50673
        020610                                                                  $50672:
4173    020610                                  LET DS.EXP-1 :B= #55
        020610  112767  000055  043767                                          MOVB    #55,DS.EXP-1
4174    020616                                  LET CC.EXP := CC.EXP SET.BY #NBIT
        020616  052767  000010  157676                                          BIS     #NBIT,CC.EXP
4175    020624                              ENDIF
        020624                                                                  $50673:
4176    020624                          ENDIF
        020624                                                                  $50671:
4177    020624                      ENDIF
        020624                                                                  $50664:
4178    020624                      IF DSLEN2 GT #0 AND DSLEN1 EQ #0 THEN       ;DST.LEN > 0, SRC.LEN = 0
        020624  005767  157666                                                  TST     DSLEN2
        020630  003420                                                          BLE     $50674
        020632  005767  157656                                                  TST     DSLEN1
        020636  001015                                                          BNE     $50674
4179    020640                          LET R1 := #DS.EXP
```

```
      020640  012701  064606                                                      MOV     #DS.EXP,R1
 4180 020644                              LET R2 := DSLEN2
      020644  016702  157646                                                      MOV     DSLEN2,R2
 4181 020650                              WHILE R2 HI #0 DO
      020650                                                           $50675:
      020650  005702                                                              TST     R2
      020652  101404                                                              BLOS    $50676
 4182 020654                                  LET (R1)+ :B= #60
      020654  112721  000060                                                      MOVB    #60,(R1)+
 4183 020660                                  LET R2 := R2 - #1
      020660  005302                                                              DEC     R2
 4184 020662                              ENDDO
      020662  000772                                                              BR      $50675
      020664                                                           $50676:
 4185 020664                              LET CC.EXP := CC.EXP SET.BY #ZBIT
      020664  052767  000004  157630                                              BIS     #ZBIT,CC.EXP
 4186 020672                          ENDIF
      020672                                                           $50674:
 4187 020672                          IF DSLEN2 GT #0 AND DSLEN1 GT #0 THEN       ;DST.LEN > 0 AND SRC.LEN > 0
      020672  005767  157620                                                      TST     DSLEN2
      020676  003565                                                              BLE     $50677
      020700  005767  157610                                                      TST     DSLEN1
      020704  003562                                                              BLE     $50677
 4188 020706                              LET R1 := #DSBUF1 + #10. - DSLEN1
      020706  012701  064440                                                      MOV     #DSBUF1,R1
      020712  062701  000012                                                      ADD     #10.,R1
      020716  166701  157572                                                      SUB     DSLEN1,R1
 4189 020722                              LET R2 := #DS.EXP
      020722  012702  064606                                                      MOV     #DS.EXP,R2
 4190 020726                              IF DSLEN2 GT DSLEN1 THEN        ;DST.LEN > SRC.LEN > 0
      020726  026767  157564  157560                                              CMP     DSLEN2,DSLEN1
      020734  003413                                                              BLE     $50700
 4191 020736                                  LET R2 := R2 + DSLEN2 - DSLEN1
      020736  066702  157554                                                      ADD     DSLEN2,R2
      020742  166702  157546                                                      SUB     DSLEN1,R2
 4192 020746                                  LET R0 := #DS.EXP
      020746  012700  064606                                                      MOV     #DS.EXP,R0
 4193 020752                                  WHILE R0 NE R2 DO
      020752                                                           $50701:
      020752  020002                                                              CMP     R0,R2
      020754  001403                                                              BEQ     $50702
 4194 020756                                      LET (R0)+ :B= #60
      020756  112720  000060                                                      MOVB    #60,(R0)+
 4195 020762                                  ENDDO
      020762  000773                                                              BR      $50701
      020764                                                           $50702:
 4196 020764                              ENDIF
      020764                                                           $50700:
 4197 020764                              IF DSLEN2 LO DSLEN1 THEN        ;SRC.LEN > DST.LEN > 0
      020764  026767  157526  157522                                              CMP     DSLEN2,DSLEN1
      020772  103007                                                              BHIS    $50703
 4198 020774                                  LET R1 := R1 + DSLEN1 - DSLEN2
      020774  066701  157514                                                      ADD     DSLEN1,R1
      021000  166701  157512                                                      SUB     DSLEN2,R1
 4199 021004                                  LET CC.EXP := CC.EXP SET.BY #VBIT
      021004  052767  000002  157510                                              BIS     #VBIT,CC.EXP
 4200 021012                              ENDIF
```

```
      021012
 4201 021012                          REPEAT
      021012
 4202 021012                              LET TEMPO :B= (R1)+ SET.BY #60
      021012  112167  157460
      021016  152767  000060  157452
 4203 021024                              LET (R2)+ :B= TEMPO
      021024  116722  157446
 4204 021030                          UNTIL R1 EQ #DSBUF1+12
      021030  020127  064452
      021034  001366
 4205 021036                          IF DATTYP EQ #0 AND SIGNO EQ #-1 THEN
      021036  005767  157422
      021042  001012
      021044  026727  157400  177777
      021052  001006
 4206 021054                              LET CC.EXP := CC.EXP SET.BY #NBIT
      021054  052767  000010  157440
 4207 021062                              LET R2 := R2 - #1
      021062  005302
 4208 021064                              LET (R2) :B= (R2) SET.BY #160
      021064  152712  000160
 4209 021070                          ENDIF
      021070
 4210 021070                          IF DATTYP EQ #20000 OR DATTYP EQ #30000 THEN
      021070  026727  157370  020000
      021076  001404
      021100  026727  157360  030000
      021106  001026
      021110
 4211 021110                              IF DATTYP EQ #20000 THEN
      021110  026727  157350  020000
      021116  001002
 4212 021120                                  LET R2 := R2 - #1
      021120  005302
 4213 021122                              ELSE
      021122  000402
      021124
 4214 021124                                  LET R2 := #DS.EXP
      021124  012702  064606
 4215 021130                              ENDIF
      021130
 4216 021130                              LET R0 :B= (R2) CLR.BY #60
      021130  111200
      021132  142700  000060
 4217 021136                              IF SIGNO EQ #-1 THEN
      021136  026727  157306  177777
      021144  001005
 4218 021146                                  LET R0 := R0 + #10.
      021146  062700  000012
 4219 021152                                  LET CC.EXP := CC.EXP SET.BY #NBIT
      021152  052767  000010  157342
 4220 021160                              ENDIF
      021160
 4221 021160                              LET (R2) :B= OVPNCH(R0)
LE
      021160  116012  056330
 4222 021164                          ENDIF
```

```
                        $50703:
                        $50704:
                                ;TRANSFER THE DIGITS TO DS.EXP
                                MOVB    (R1)+,TEMP0
                                BISB    #60,TEMP0
                                MOVB    TEMP0,(R2)+
                                CMP     R1,#DSBUF1+12
                                BNE     $50704
                                TST     DATTYP
                                BNE     $50705
                                CMP     SIGNO,#-1
                                BNE     $50705
                                BIS     #NBIT,CC.EXP
                                DEC     R2
                                ;*** SIGNED ZONE, NEGATIVE ***
                                BISB    #160,(R2)
                        $50705:
                            ;*** OVERPUNCH ***
                                CMP     DATTYP,#20000
                                BEQ     $50706
                                CMP     DATTYP,#30000
                                BNE     $50707
                        $50706:
                                ;TRAILING
                                CMP     DATTYP,#20000
                                BNE     $50710
                                DEC     R2
                        ;LEADING
                                BR      $50711
                        $50710:
                                MOV     #DS.EXP,R2
                        $50711:
                                MOVB    (R2),R0
                                BICB    #60,R0
                        ;NEG. SIGN
                                CMP     SIGNO,#-1
                                BNE     $50712
                                ADD     #10.,R0
                                BIS     #NBIT,CC.EXP
                        $50712:
                        ;GET APPROPRIATE SIGN/DIGIT FROM TAB
                                MOVB    OVPNCH(R0),(R2)
```

```
                021164
4223            021164                  IF DATTYP EQ #40000 OR DATTYP EQ #50000 THEN
                021164  026727  157274  040000
                021172  001404
                021174  026727  157264  050C00
                021202  001023
                021204
4224            021204                      IF DATTYP EQ #50000 THEN
                021204  026727  157254  050000
                021212  001003
4225            021214
                021214  012702  064606              LET R2 := #DS.EXP - #1
                021220  005302
4226            021222                      ENDIF
                021222
4227            021222                      IF SIGNO EQ #-1 THEN
                021222  026727  157222  177777
                021230  001006
4228            021232
                021232  112712  000055              LET (R2) :B= #55
4229            021236
                021236  052767  000010  157256      LET CC.EXP := CC.EXP SET.BY #NBIT
4230            021244                      ELSE
                021244  000402
                021246
4231            021246
                021246  112712  000053              LET (R2) :B= #53
4232            021252                      ENDIF
                021252
4233            021252                  ENDIF
                021252
4234            021252              ENDIF
                021252
4235
4236                                ;+
4237                                ; SET UP GPR'S AND EXPECTED GPR'S.
4238                                ;-
4239
4240            021252                  LET R0 := DATA00                ;DATA TYPE AND LENGTH
                021252  016700  157304
4241            021256                  LET R1 := #DS.DST               ;DST. ADR
                021256  012701  064544
4242            021262                  LET R2 := DATA01
                021262  016702  157276
4243            021266                  LET R3 := DATA02
                021266  016703  157274
4244            021272                  IF #1 SETIN COUNT THEN
                021272  032767  000001  157170
                021300  001405
4245            021302                      LET R4 := #177777
                021302  012704  177777
4246            021306                      LET R5 := #177777
                021306  012705  177777
4247            021312                  ELSE
                021312  000402
                021314
4248            021314                      LET R4 := #0
```

```
$50707:
    ;*** SEPARATE ***
        CMP     DATTYP,#40000
        BEQ     $50713
        CMP     DATTYP,#50000
        BNE     $50714
$50713:
        ;LEADING
        CMP     DATTYP,#50000
        BNE     $50715

        MOV     #DS.EXP,R2
        DEC     R2
$50715:

        CMP     SIGNO,#-1
        BNE     $50716
;NEGATIVE
        MOVB    #55,(R2)

        BIS     #NBIT,CC.EXP

        BR      $50717
$50716:
;POSITIVE
        MOVB    #53,(R2)

$50717:
$50714:
$50677:

        MOV     DATA00,R0
        MOV     #DS.DST,R1
        MOV     DATA01,R2
        MOV     DATA02,R3
        BIT     #1,COUNT
        BEQ     $50720
        MOV     #177777,R4
        MOV     #177777,R5
        BR      $50721
$50720:
```

```
      021314  005004                                                        CLR     R4
 4249 021316                          LET R5 := #0
      021316  005005                                                        CLR     R5
 4250 021320                        ENDIF
      021320                                                        $50721:
 4251 021320                        LET R0.EXP := R0
      021320  010067  157200                                                MOV     R0,R0.EXP
 4252 021324                        LET R1.EXP := R1
      021324  010167  157176                                                MOV     R1,R1.EXP
 4253 021330                        LET R2.EXP := #0
      021330  005067  157174                                                CLR     R2.EXP
 4254 021334                        LET R3.EXP := #0
      021334  005067  157172                                                CLR     R3.EXP
 4255 021340                        LET R4.EXP := R4
      021340  010467  157170                                                MOV     R4,R4.EXP
 4256 021344                        LET R5.EXP := R5
      021344  010567  157166                                                MOV     R5,R5.EXP
 4257
 4258                             ;+
 4259                             ; DO THE CIS INSTRUCTION
 4260                             ;-
 4261
 4262 021350                        INLINE <CVTLN>
      021350  076057                                                        CVTLN
 4263
 4264
 4265                             ;+
 4266                             ;  CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.
 4267                             ;-
 4268
 4269 021352                        LET CC := PSW
      021352  016767  156420  157140                                        MOV     PSW,CC
 4270 021360                        LET CC := CC CLR.BY #177760
      021360  042767  177760  157132                                        BIC     #177760,CC
 4271 021366                        IF CC NE CC.EXP THEN
      021366  026767  157126  157126                                        CMP     CC,CC.EXP
      021374  001402                                                       BEQ     $50722
 4272 021376                            CALL ERROR
      021376  004767  036276                                                JSR     PC,ERROR
 4273 021402                        ENDIF
      021402                                                        $50722:
 4274 021402                        IF R0 NE R0.EXP THEN
      021402  020067  157116                                                CMP     R0,R0.EXP
      021406  001402                                                       BEQ     $50723
 4275 021410                            CALL ERROR
      021410  004767  036264                                                JSR     PC,ERROR
 4276 021414                        ENDIF
      021414                                                        $50723:
 4277 021414                        IF R1 NE R1.EXP THEN
      021414  020167  157106                                                CMP     R1,R1.EXP
      021420  001402                                                       BEQ     $50724
 4278 021422                            CALL ERROR
      021422  004767  036252                                                JSR     PC,ERROR
 4279 021426                        ENDIF
      021426                                                        $50724:
 4280 021426                        IF R2 NE R2.EXP THEN
      021426  020267  157076                                                CMP     R2,R2.EXP
```

```
      021432  001402                                                        BEQ     $50725
4281  021434                          CALL ERROR
      021434  004767  036240                                                JSR     PC,ERROR
4282  021440                      ENDIF
      021440                                                              $50725:
4283  021440                      IF R3 NE R3.EXP THEN
      021440  020367  157066                                                CMP     R3,R3.EXP
      021444  001402                                                       BEQ     $50726
4284  021446                          CALL ERROR
      021446  004767  036226                                                JSR     PC,ERROR
4285  021452                      ENDIF
      021452                                                              $50726:
4286  021452                      IF R4 NE R4.EXP THEN
      021452  020467  157056                                                CMP     R4,R4.EXP
      021456  001402                                                       BEQ     $50727
4287  021460                          CALL ERROR
      021460  004767  036214                                                JSR     PC,ERROR
4288  021464                      ENDIF
      021464                                                              $50727:
4289  021464                      IF R5 NE R5.EXP THEN
      021464  020567  157046                                                CMP     R5,R5.EXP
      021470  001402                                                       BEQ     $50730
4290  021472                          CALL ERROR
      021472  004767  036202                                                JSR     PC,ERROR
4291  021476                      ENDIF
      021476                                                              $50730:
4292
4293                              ;+
4294                              ; COMPARE THE EXPECTED DECIMAL STRING BUFFER
4295                              ; WITH ACTUAL RESULTS.
4296                              ;-
4297
4298  021476                      LET R0 := #DS.DST
      021476  012700  064544                                                MOV     #DS.DST,R0
4299  021502                      LET R1 := #DS.EXP
      021502  012701  064606                                                MOV     #DS.EXP,R1
4300  021506                      LET ERRFLG := #0
      021506  005067  156744                                                CLR     ERRFLG
4301  021512                      WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO
      021512                                                              $50731:
      021512  020027  064604                                                CMP     R0,#DS.DST+40
      021516  001410                                                       BEQ     $50732
      021520  005767  156732                                                TST     ERRFLG
      021524  001005                                                       BNE     $50732
4302  021526                          IFB (R0)+ NE (R1)+ THEN
      021526  122021                                                       CMPB    (R0)+,(R1)+
      021530  001402                                                       BEQ     $50733
4303  021532                              CALL ERROR
      021532  004767  036142                                                JSR     PC,ERROR
4304  021536                          ENDIF
      021536                                                              $50733:
4305  021536                      ENDDO
      021536  000765                                                       BR      $50731
      021540                                                              $50732:
4306  021540                      LET TSTCAS := TSTCAS + #6
      021540  062767  000006  156776                                        ADD     #6,TSTCAS
4307  021546                      LET COUNT := COUNT - #1
```

```
      021546  005367  156716                                                    DEC     COUNT
 4308                                ;UNTIL COUNT EQ #0
 4309 021552                            IF COUNT NE #0 THEN
      021552  005767  156712                                                    TST     COUNT
      021556  001402                                                           BEQ     $50734
 4310 021560                                INLINE <JMP $5057>
      021560  000167  176610                                                    JMP $5057
 4311 021564                            ENDIF
      021564                                                              $50734:
 4312
 4313                                ;+
 4314                                ; DO ONE IN-LINE CASE - CVTLNI
 4315                                ;-
 4316
 4317 021564                            CALL CLRREG
      021564  004767  034650                                                    JSR     PC,CLRREG
 4318 021570                            CALL CLRDSB
      021570  004767  035100                                                    JSR     PC,CLRDSB
 4319 021574                            INLINE <CVTLNI>
      021574  076157                                                           CVTLNI
 4320 021576                            INLINE  <.WORD       TCVLNI>
      021576  022076                                                           .WORD   TCVLNI
 4321 021600                            INLINE  <.WORD       TCVLNI+4>
      021600  022102                                                           .WORD   TCVLNI+4
 4322 021602                            LET CC := PSW
      021602  016767  156170  156710                                           MOV     PSW,CC
 4323 021610                            LET CC := CC CLR.BY #177760
      021610  042767  177760  156702                                           BIC     #177760,CC
 4324 021616                            IF CC NE #0 THEN
      021616  005767  156676                                                    TST     CC
      021622  001402                                                           BEQ     $50735
 4325 021624                                CALL ERROR
      021624  004767  036050                                                    JSR     PC,ERROP
 4326 021630                            ENDIF
      021630                                                              $50735:
 4327 021630                            IF R0 NE #0 THEN
      021630  005700                                                           TST     R0
      021632  001402                                                           BEQ     $50736
 4328 021634                                CALL ERROR
      021634  004767  036040                                                    JSR     PC,ERROR
 4329 021640                            ENDIF
      021640                                                              $50736:
 4330 021640                            IF R1 NE #0 THEN
      021640  005701                                                           TST     R1
      021642  001402                                                           BEQ     $50737
 4331 021644                                CALL ERROR
      021644  004767  036030                                                    JSR     PC,ERROR
 4332 021650                            ENDIF
      021650                                                              $50737:
 4333 021650                            IF R2 NE #0 THEN
      021650  005702                                                           TST     R2
      021652  001402                                                           BEQ     $50740
 4334 021654                                CALL ERROR
      021654  004767  036020                                                    JSR     PC,ERROR
 4335 021660                            ENDIF
      021660                                                              $50740:
 4336 021660                            IF R3 NE #0 THEN
```

```
        021660  005703                                                          TST     R3
        021662  001402                                                          BEQ     $50741
4337    021664                          CALL ERROR
        021664  004767  036010                                                  JSR     PC,ERROR
4338    021670                      ENDIF
        021670                                                          $50741:
4339    021670                      IF R4 NE #0 THEN
        021670  005704                                                          TST     R4
        021672  001402                                                          BEQ     $50742
4340    021674                          CALL ERROR
        021674  004767  036000                                                  JSR     PC,ERROR
4341    021700                      ENDIF
        021700                                                          $50742:
4342    021700                      IF R5 NE #0 THEN
        021700  005705                                                          TST     R5
        021702  001402                                                          BEQ     $50743
4343    021704                          CALL ERROR
        021704  004767  035770                                                  JSR     PC,ERROR
4344    021710                      ENDIF
        021710                                                          $50743:
4345    021710                      LET R0 := #DS.DST-1
        021710  012700  064543                                                  MOV     #DS.DST-1,R0
4346    021714                      IFB (R0)+ NE #53 THEN
        021714  122027  000053                                                  CMPB    (R0)+,#53
        021720  001402                                                          BEQ     $50744
4347    021722                          CALL ERROR
        021722  004767  035752                                                  JSR     PC,ERROR
4348    021726                      ENDIF
        021726                                                          $50744:
4349    021726                      LET ERRFLG := #0
        021726  005067  156524                                                  CLR     ERRFLG
4350    021732                      REPEAT
        021732                                                          $50745:
4351    021732                          IFB (R0)+ NE #61 THEN
        021732  122027  000061                                                  CMPB    (R0)+,#61
        021736  001402                                                          BEQ     $50746
4352    021740                              CALL ERROR
        021740  004767  035734                                                  JSR     PC,ERROR
4353    021744                          ENDIF
        021744                                                          $50746:
4354    021744                      UNTIL R0 EQ #DS.DST+12 OR ERRFLG NE #0
        021744  020027  064556                                                  CMP     R0,#DS.DST+12
        021750  001403                                                          BEQ     $50747
        021752  005767  156500                                                  TST     ERRFLG
        021756  001765                                                          BEQ     $50745
        021760                                                          $50747:
4355    021760                      INLINE      <JMP XCVTPL>
        021760  000167  000122                                                  JMP XCVTPL
4356
4357                            ;+
4358                            ; TABLE OF TEST CASES FOR CVTLN
4359                            ;-
4360
4361    021764  000014          TCVTLN: .WORD   12.             ;# OF TEST CASES FOR CVTLN
4362
4363    021766  000005                  .WORD   5                       ;DST. DATA TYPE AND LENGTH
4364    021770  000000                  .WORD   0                       ;LONG INTEGER HIGH & SIGN
```

```
4365 021772  000000                    .WORD   0                       ;LONG INTEGER LOW
4366
4367 021774  050006                    .WORD   50006           ;TEST CASE NUMBER 2
4368 021776  000000                    .WORD   0
4369 022000  000000                    .WORD   0
4370
4371 022002  020010                    .WORD   20010           ;TEST CASE NUMBER 3
4372 022004  000000                    .WORD   0
4373 022006  001751                    .WORD   1751
4374
4375 022010  030000                    .WORD   30000           ;TEST CASE NUMBER 4
4376 022012  000000                    .WORD   0
4377 022014  152361                    .WORD   152361
4378
4379 022016  040000                    .WORD   40000           ;TEST CASE NUMBER 5
4380 022020  000000                    .WORD   0
4381 022022  024137                    .WORD   24137
4382
4383 022024  050003                    .WORD   50003           ;TEST CASE NUMBER 6
4384 022026  000000                    .WORD   0
4385 022030  075147                    .WORD   75147
4386
4387 022032  020004                    .WORD   20004           ;TEST CASE NUMBER 7
4388 022034  000000                    .WORD   0
4389 022036  075147                    .WORD   75147
4390
4391 022040  000012                    .WORD   12              ;TEST CASE NUMBER 8
4392 022042  077777                    .WORD   77777
4393 022044  177777                    .WORD   177777
4394
4395 022046  000037                    .WORD   37              ;TEST CASE NUMBER 9
4396 022050  100000                    .WORD   100000
4397 022052  000000                    .WORD   0
4398
4399 022054  010021                    .WORD   10021           ;TEST CASE NUMBER 10
4400 022056  100000                    .WORD   100000
4401 022060  000000                    .WORD   0
4402
4403 022062  040012                    .WORD   40012           ;TEST CASE NUMBER 11
4404 022064  074377                    .WORD   74377
4405 022066  171700                    .WORD   171700
4406
4407 022070  030000                    .WORD   30000           ;TEST CASE NUMBER 12
4408 022072  000000                    .WORD   0
4409 022074  000000                    .WORD   0
4410
4411 022076                    TCVLNI:                          ;IN-LINE TEST CASE
4412 022076  050012                            .WORD   50012           ;DST TYPE = LEADING SEPARATE, LENGTH = 12
4413 022100  064544                            .WORD   DS.DST          ;DST.ADDRESS
4414 022102  032707                            .WORD   32707           ;LONG INTEGER LOW WORD +
4415 022104  041072                            .WORD   41072           ;  LONG INTEGER HIGH WORD AND SIGN
4416
4417                           ;  = 1111111111 IN BCD
4418
```

```
                                .SBTTL  CVTPL
                                ;*****************************************************************
                                ;
                                ;  CVTPL TEST
                                ;
                                ;*****************************************************************

                                ROUTINE XCVTPL
                                                                                        XCVTPL:
                                   INLINE       <SCOPE>
                                                                                                SCOPE
                                   INLINE       <NOP>
                                                                                                NOP
                                   LET $TIMES := #20                    ;ITERATION COUNT
                                                                                                MOV     #20,$TIMES
                                   LET COUNT := TCVTPL                  ;GET TEST COUNT
                                                                                                MOV     TCVTPL,COUNT
                                   LET TSTCAS := #TCVTPL+2              ;POINT TO FIRST TEST CASE
                                                                                                MOV     #TCVTPL+2,TSTCAS
                                ;REPEAT
                                   INLINE       <$5073:>
                                                                                                $5073:

                                ;+
                                ; GET TEST DATA FROM TABLE
                                ;-

                                   LET R0 := TSTCAS
                                                                                                MOV     TSTCAS,R0
                                   LET DATA00 := (R0)+                  ;SRC. DATA TYPE AND LENGTH
                                                                                                MOV     (R0)+,DATA00
                                   LET DATA01 := (R0)                   ;SRC. POINTER
                                                                                                MOV     (R0),DATA01

                                ;+
                                ; SET UP R0.EXP, R1.EXP, R4.EXP, R5.EXP, R4, AND R5.
                                ;-

                                   LET R0.EXP := #0
                                                                                                CLR     R0.EXP
                                   LET R1.EXP := #0
                                                                                                CLR     R1.EXP
                                   IF #1 SETIN COUNT THEN
                                                                                                BIT     #1,COUNT
                                                                                                BEQ     $50752
                                      LET R4 := #177777
                                                                                                MOV     #177777,R4
                                      LET R5 := #177777
                                                                                                MOV     #177777,R5
                                   ELSE
                                                                                                BR      $50753
                                                                                        $50752:
                                      LET R4 := #0
                                                                                                CLR     R4
                                      LET R5 := #0
                                                                                                CLR     R5
                                   ENDIF
```

```
                                                                                $50753:
                        LET R4.EXP := R4                                                MOV     R4,R4.EXP
                        LET R5.EXP := R5                                                MOV     R5,R5.EXP

                    ;+
                    ;  CONVERT PACKED NUMBER AND LOAD INTO R2.EXP AND R3.EXP
                    ;-

                        CALL  CLRDSB                                                    JSR     PC,CLRDSB
                        LET CC.EXP := #0                                                CLR     CC.EXP
                        LET R0 := #DSBUF1               ;DST.POINTER                    MOV     #DSBUF1,R0
                        LET R1 := DATA01                ;SRC.POINTER                    MOV     DATA01,R1
                        LET DSLEN1 := DATA00 CLR.BY #177740         ;GET SRC.LEN        MOV     DATA00,DSLEN1
                                                                                        BIC     #177740,DSLEN1
                        IF DSLEN1 EQ #0 THEN            ;SRC.LEN=0                      TST     DSLEN1
                                                                                        BNE     $50754
                            LET CC.EXP := CC.EXP SET.BY #ZBIT                           BIS     #ZBIT,CC.EXP
                            LET R2.EXP := #0                                            CLR     R2.EXP
                            LET R3.EXP := #0                                            CLR     R3.EXP
                        ELSE                                                            BR      $50755
                                                                                $50754:
                            LET R0 := #DSBUF1 + #10. - DSLEN1       ;DEST. FOR INTERMEDIATE RESULTS
                                                                                        MOV     #DSBUF1,R0
                                                                                        ADD     #10.,R0
                                                                                        SUB     DSLEN1,R0

                    ;LOAD UP DSBUF1 BUFFER WITH INTERMEDIATE RESULTS

                            CALL CLRDSB                 ;CLEAR DECIMAL STRING BUFFERS
                                                                                        JSR     PC,CLRDSB
                            IF #1 NOTSETIN DSLEN1 THEN                                  BIT     #1,DSLEN1
                                                                                        BNE     $50756
                                LET (R1) :B= (R1) CLR.BY #360       ;CLEAR UPPER NIBBLE BICB    #360,(R1)
                                LET (R0)+ :B= (R1)+                                     MOVB    (R1)+,(R0)+
                                LET DSLEN1 := DSLEN1 - #1                               DEC     DSLEN1
                            ENDIF
                                                                                $50756:
                            WHILE DSLEN1 HI #1 DO                               $50757:
                                                                                        CMP     DSLEN1,#1
                                                                                        BLOS    $50760
```

```
4485 022352                               LET (R0) :B= (R1)
     022352  111110                                                              MOVB    (R1),(R0)
4486 022354                               INLINE <ASRB (R0)>
     022354  106210                                                              ASRB (R0)
4487 022356                               INLINE <ASRB (R0)>
     022356  106210                                                              ASRB (R0)
4488 022360                               INLINE <ASRB (R0)>
     022360  106210                                                              ASRB (R0)
4489 022362                               INLINE <ASRB (R0)>
     022362  106210                                                              ASRB (R0)
4490 022364                               LET (R0) :B= (R0) CLR.BY #360      ;CLEAR UPPER NIBBLE
     022364  142710  000360                                                      BICB    #360,(R0)
4491 022370                               LET R0 := R0 + #1
     022370  005200                                                              INC     R0
4492 022372                               LET (R0) :B= (R1)+ CLR.BY #360
     022372  112110                                                              MOVB    (R1)+,(R0)
     022374  142710  000360                                                      BICB    #360,(R0)
4493 022400                               LET R0 := R0 + #1
     022400  005200                                                              INC     R0
4494 022402                               LET DSLEN1 := DSLEN1 - #2
     022402  162767  000002  156104                                              SUB     #2,DSLEN1
4495 022410                           ENDDO
     022410  000754                                                              BR      $50757
     022412                                                               $50760:
4496 022412                           LET (R0) :B= (R1)                       ;GET THE LSD
     022412  111110                                                              MOVB    (R1),(R0)
4497 022414                           INLINE <ASRB (R0)>
     022414  106210                                                              ASRB (R0)
4498 022416                           INLINE <ASRB (R0)>
     022416  106210                                                              ASRB (R0)
4499 022420                           INLINE <ASRB (R0)>
     022420  106210                                                              ASRB (R0)
4500 022422                           INLINE <ASRB (R0)>
     022422  106210                                                              ASRB (R0)
4501 022424                           LET (R0) :B= (R0) CLR.BY #360
     022424  142710  000360                                                      BICB    #360,(R0)
4502
4503                          ;LOAD UP R2.EXP AND R3.EXP WITH FINAL RESULT
4504
4505 022430                           CALL CVTIL
     022430  004767  007554                                                      JSR     PC,CVTIL
4506 022434                           LET DATTYP := DATA00 CLR.BY #107777
     022434  016767  156122  156022                                              MOV     DATA00,DATTYP
     022442  042767  107777  156014                                              BIC     #107777,DATTYP
4507 022450                           IF DATTYP EQ #60000 THEN          ;SIGNED PACKED
     022450  026727  156010  060000                                              CMP     DATTYP,#60000
     022456  001042                                                              BNE     $50761
4508 022460                               LET TEMP5 :B= (R1) CLR.BY #360        ;ISOLATE THE SIGN
     022460  111167  156024                                                      MOVB    (R1),TEMP5
     022464  142767  000360  156016                                              BICB    #360,TEMP5
4509 022472                               IFB TEMP5 EQ #15 OR TEMP5 EQ #13 THEN
     022472  126727  156012  000015                                              CMPB    TEMP5,#15
     022500  001404                                                              BEQ     $50762
     022502  026727  156002  000013                                              CMP     TEMP5,#13
     022510  001025                                                              BNE     $50763
     022512                                                               $50762:
4510 022512                                   LET CC.EXP := CC.EXP SET.BY #NBIT
```

CVTPL

```
      022512  052767  000010  156002
4511  022520                                LET R3.EXP := NEGATE R3.EXP                                BIS     #NBIT,CC.EXP
      022520  005467  156006                LET R2.EXP := COMP R2.EXP                                  NEG     R3.EXP
4512  022524                                IF R3.EXP EQ #0 THEN                                       COM     R2.EXP
      022524  005167  156000                                                                           TST     R3.EXP
4513  022530                                    LET R2.EXP := R2.EXP + #1                              BNE     $50764
      022530  005767  155776                ENDIF                                                      INC     R2.EXP
      022534  001002                                                                          $50764:
4514  022536                                IF R2.EXP NE #0 OR R3.EXP NE #0 THEN                       TST     R2.EXP
      022536  005267  155766                                                                           BNE     $50765
4515  022542                                    LET CC.EXP := CC.EXP SET.BY #CBIT                      TST     R3.EXP
      022542                                ENDIF                                                      BEQ     $50766
4516  022542                            ENDIF                                                  $50765:
      022542  005767  155762                                                                           BIS     #CBIT,CC.EXP
      022546  001003
      022550  005767  155756
      022554  001403
      022556                                                                                   $50766:
4517  022556                        ENDIF                                                      $50763:
      022556  052767  000001  155736
4518  022564                    ENDIF                                                          $50761:
      022564
4519  022564                ENDIF                                                              $50755:
      022564
4520  022564            ENDIF
      022564
4521  022564        ENDIF
      022564
4522
4523                ;+
4524                ; SET UP R0 THRU R3
4525                ;-
4526
4527  022564            LET R0 := DATA00          ;SRC.DSCR                                      MOV     DATA00,R0
      022564  016700  155772                                                                       MOV     DATA01,R1
4528  022570            LET R1 := DATA01          ;SRC.DSCR                                      CLR     R2
      022570  016701  155770                                                                       CLR     R3
4529  022574            LET R2 := #0
      022574  005002
4530  022576            LET R3 := #0
      022576  005003
4531
4532                ;+
4533                ; DO THE CIS INSTRUCTION
4534                ;-
4535
4536  022600            INLINE <CVTPL>                                                             CVTPL
      022600  076073
4537
4538
4539                ;+
4540                ;  CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.
4541                ;-
4542
4543  022602            LET CC := PSW                                                              MOV     PSW,CC
      022602  016767  155170  155710
4544  022610            LET CC := CC CLR.BY #177760
```

```
      022610  042767  177760  155702                          BIC     #177760,CC
 4545 022616                          IF CC NE CC.EXP THEN
      022616  026767  155676  155676                          CMP     CC,CC.EXP
      022624  001402                                          BEQ     $50767
 4546 022626                              CALL ERROR
      022626  004767  035046                                  JSR     PC,ERROR
 4547 022632                          ENDIF
      022632                                             $50767:
 4548 022632                          IF R0 NE R0.EXP THEN
      022632  020067  155666                                  CMP     R0,R0.EXP
      022636  001402                                          BEQ     $50770
 4549 022640                              CALL ERROR
      022640  004767  035034                                  JSR     PC,ERROR
 4550 022644                          ENDIF
      022644                                             $50770:
 4551 022644                          IF R1 NE R1.EXP THEN
      022644  020167  155656                                  CMP     R1,R1.EXP
      022650  001402                                          BEQ     $50771
 4552 022652                              CALL ERROR
      022652  004767  035022                                  JSR     PC,ERROR
 4553 022656                          ENDIF
      022656                                             $50771:
 4554 022656                          IF R2 NE R2.EXP THEN
      022656  020267  155646                                  CMP     R2,R2.EXP
      022662  001402                                          BEQ     $50772
 4555 022664                              CALL ERROR
      022664  004767  035010                                  JSR     PC,ERROR
 4556 022670                          ENDIF
      022670                                             $50772:
 4557 022670                          IF R3 NE R3.EXP THEN
      022670  020367  155636                                  CMP     R3,R3.EXP
      022674  001402                                          BEQ     $50773
 4558 022676                              CALL ERROR
      022676  004767  034776                                  JSR     PC,ERROR
 4559 022702                          ENDIF
      022702                                             $50773:
 4560 022702                          IF R4 NE R4.EXP THEN
      022702  020467  155626                                  CMP     R4,R4.EXP
      022706  001402                                          BEQ     $50774
 4561 022710                              CALL ERROR
      022710  004767  034764                                  JSR     PC,ERROR
 4562 022714                          ENDIF
      022714                                             $50774:
 4563 022714                          IF R5 NE R5.EXP THEN
      022714  020567  155616                                  CMP     R5,R5.EXP
      022720  001402                                          BEQ     $50775
 4564 022722                              CALL ERROR
      022722  004767  034752                                  JSR     PC,ERROR
 4565 022726                          ENDIF
      022726                                             $50775:
 4566
 4567 022726                          LET TSTCAS := TSTCAS + #4
      022726  062767  000004  155610                          ADD     #4,TSTCAS
 4568 022734                          LET COUNT := COUNT - #1
      022734  005367  155530                                  DEC     COUNT
 4569                               ;UNTIL COUNT EQ #0
 4570 022740                          IF COUNT NE #0 THEN
```

```
                                                                                    TST     COUNT
                                                                                    BEQ     $50776
4571 022746                         INLINE <JMP $5073>
     022746  000167  177162                                                         JMP $5073
4572 022752                     ENDIF
     022752                                                                 $50776:
4573
4574                            ;+
4575                            ; DO ONE INLINE CASE - CVTPLI
4576                            ;-
4577
4578 022752                         CALL CLRREG
     022752  004767  033462                                                         JSR     PC,CLRREG
4579 022756                         CALL CLRDSB
     022756  004767  033712                                                         JSR     PC,CLRDSB
4580 022762                         INLINE <CVTPLI>
     022762  076173                                                               CVTPLI
4581 022764                         INLINE  <.WORD      TCVPLI>
     022764  023210                                                               .WORD   TCVPLI
4582 022766                         INLINE  <.WORD      TCVPLI+4>
     022766  023214                                                               .WORD   TCVPLI+4
4583 022770                         LET CC := PSW
     022770  016767  155002  155522                                               MOV     PSW,CC
4584 022776                         LET CC := CC CLR.BY #177760
     022776  042767  177760  155514                                               BIC     #177760,CC
4585 023004                         IF CC NE #0 THEN
     023004  005767  155510                                                       TST     CC
     023010  001402                                                               BEQ     $50777
4586 023012                             CALL ERROR
     023012  004767  034662                                                         JSR     PC,ERROR
4587 023016                         ENDIF
     023016                                                                 $50777:
4588 023016                         IF R0 NE #0 THEN
     023016  005700                                                               TST     R0
     023020  001402                                                               BEQ     $51000
4589 023022                             CALL ERROR
     023022  004767  034652                                                         JSR     PC,ERROR
4590 023026                         ENDIF
     023026                                                                 $51000:
4591 023026                         IF R1 NE #0 THEN
     023026  005701                                                               TST     R1
     023030  001402                                                               BEQ     $51001
4592 023032                             CALL ERROR
     023032  004767  034642                                                         JSR     PC,ERROR
4593 023036                         ENDIF
     023036                                                                 $51001:
4594 023036                         IF R2 NE #0 THEN
     023036  005702                                                               TST     R2
     023040  001402                                                               BEQ     $51002
4595 023042                             CALL ERROR
     023042  004767  034632                                                         JSR     PC,ERROR
4596 023046                         ENDIF
     023046                                                                 $51002:
4597 023046                         IF R3 NE #0 THEN
     023046  005703                                                               TST     R3
     023050  001402                                                               BEQ     $51003
4598 023052                             CALL ERROR
```

```
        023052  004767  034622                                                  JSR     PC,ERROR
4599    023056                          ENDIF
        023056                                                          $51003:
4600    023056                          IF R4 NE #0 THEN
        023056  005704                                                          TST     R4
        023060  001402                                                          BEQ     $51004
4601    023062                              CALL ERROR
        023062  004767  034612                                                  JSR     PC,ERROR
4602    023066                          ENDIF
        023066                                                          $51004:
4603    023066                          IF R5 NE #0 THEN
        023066  005705                                                          TST     R5
        023070  001402                                                          BEQ     $51005
4604    023072                              CALL ERROR
        023072  004767  034602                                                  JSR     PC,ERROR
4605    023076                          ENDIF
        023076                                                          $51005:
4606    023076                          IF TCVPLI+4 NE #32707 THEN
        023076  026727  000112  032707                                          CMP     TCVPLI+4,#32707
        023104  001402                                                          BEQ     $51006
4607    023106                              CALL ERROR
        023106  004767  034566                                                  JSR     PC,ERROR
4608    023112                          ENDIF
        023112                                                          $51006:
4609    023112                          IF TCVPLI+6 NE #41072 THEN
        023112  026727  000100  041072                                          CMP     TCVPLI+6,#41072
        023120  001402                                                          BEQ     $51007
4610    023122                              CALL ERROR
        023122  004767  034552                                                  JSR     PC,ERROR
4611    023126                          ENDIF
        023126                                                          $51007:
4612    023126                      -   INLINE  <JMP XCVTNL>
        023126  000167  000076                                                  JMP XCVTNL
4613
4614                                    ;+
4615                                    ; TABLE OF TEST CASES FOR CVTPL
4616                                    ;-
4617
4618    023132  000006                  TCVTPL: .WORD   6               ;# OF TEST CASES FOR CVTPL
4619
4620    023134  060000                          .WORD   60000                   ;SRC. DATA TYPE AND LENGTH
4621    023136  023164                          .WORD   CVTPL1                  ;POINTER TO PACKED DECIMAL STRING
4622
4623    023140  060004                          .WORD   60004           ;TEST CASE NUMBER 2
4624    023142  023165                          .WORD   CVTPL2
4625
4626    023144  060004                          .WORD   60004           ;TEST CASE NUMBER 3
4627    023146  023170                          .WORD   CVTPL3
4628
4629    023150  060012                          .WORD   60012           ;TEST CASE NUMBER 4
4630    023152  023173                          .WORD   CVTPL4
4631
4632    023154  070007                          .WORD   70007           ;TEST CASE NUMBER 5
4633    023156  023201                          .WORD   CVTPL5
4634
4635    023160  060005                          .WORD   60005           ;TEST CASE NUMBER 6
4636    023162  023205                          .WORD   CVTPL6
```

```
                                CVTPL1: .BYTE   014                             ;PACKED DECIMAL STRING 1

                                CVTPL2: .BYTE   001,043,114                     ;PACKED DECIMAL STRING 2

                                CVTPL3: .BYTE   001,043,115                     ;PACKED DECIMAL STRING 3

                                CVTPL4: .BYTE   002,024,164,203,144,174         ;PACKED DECIMAL STRING 4

                                CVTPL5: .BYTE   022,000,007,055                 ;PACKED DECIMAL STRING 5

                                CVTPL6: .BYTE   061,040,174                     ;PACKED DECIMAL STRING 6


                                        .EVEN

                                TCVPLI:                         ;INLINE TEST CASE
                                        .WORD   60012           ;SRC. DATA TYPE = SIGNED PACKED, LENGTH = 12
                                        .WORD   PLI             ;POINTER TO SRC. STRING
                                        .WORD   0               ;LONG INTEGER RESULT, LOW
                                        .WORD   0               ;LONG INTEGER RESULT, HI, AND SIGN

                                PLI:    .BYTE   1,21,21,21,21,21,34     ;INLINE PACKED DECIMAL SRC. STRING

                                ;  = +1111111111 IN BCD

                                        .EVEN
```

```
                        .SBTTL  CVTNL
                        ;*****************************************************************
                        ;
                        ;  CVTNL TEST
                        ;
                        ;*****************************************************************

023230                  ROUTINE XCVTNL
023230                                                                  XCVTNL:
                           INLINE <SCOPE>
023230  000004                                                            SCOPE
                           INLINE <NOP>
023232  000240                                                          NOP
                           LET $TIMES := #20            ;ITERATION COUNT
023234  012767  000020  155142                                         MOV     #20,$TIMES
                           LET COUNT := TCVTNL          ;GET TEST COUNT
023242  016767  001246  155220                                         MOV     TCVTNL,COUNT
                           LET TSTCAS := #TCVTNL+2              ;POINT TO FIRST TEST CASE
023250  012767  024516  155266                                         MOV     #TCVTNL+2,TSTCAS
                        ;REPEAT
                           INLINE <$5053:>
023256                                                                  $5053:

                        ;+
                        ; GET TEST DATA FROM TABLE
                        ;-

                           LET R0 := TSTCAS
023256  016700  155262                                                  MOV     TSTCAS,R0
                           LET DATA00 := (R0)+          ;SRC. DATA TYPE AND LENGTH
023262  012067  155274                                                  MOV     (R0)+,DATA00
                           LET DATA01 := (R0)           ;SRC. POINTER
023266  011067  155272                                                  MOV     (R0),DATA01

                        ;+
                        ; SET UP R0.EXP, R1.EXP, R4.EXP, R5.EXP, R4, AND R5.
                        ;-

                           LET R0.EXP := #0
023272  005067  155226                                                  CLR     R0.EXP
                           LET R1.EXP := #0
023276  005067  155224                                                  CLR     R1.EXP
                           IF #1 SETIN COUNT THEN
023302  032767  000001  155160                                         BIT     #1,COUNT
023310  001405                                                          BEQ     $51012

                               LET R4 := #177777
023312  012704  177777                                                  MOV     #177777,R4
                               LET R5 := #177777
023316  012705  177777                                                  MOV     #177777,R5
                           ELSE
023322  000402                                                          BR      $51013
023324                                                          $51012:

                               LET R4 := #0
023324  005004                                                          CLR     R4
                               LET R5 := #0
023326  005005                                                          CLR     R5
                           ENDIF
```

```
 023330                                                               $51013:
 023330                            LET R4.EXP := R4
 023330  010467  155200                                                        MOV     R4,R4.EXP
 023334                            LET R5.EXP := R5
 023334  010567  155176                                                        MOV     R5,R5.EXP


                                 ;+
                                 ; CONVERT NUMERIC DECIMAL STRING AND LOAD INTO R2.EXP AND R3.EXP
                                 ;-

 023340                            LET CC.EXP := #0
 023340  005067  155156                                                        CLR     CC.EXP
 023344                            LET R0 := #DSBUF1                 ;DST. POINTER
 023344  012700  064440                                                        MOV     #DSBUF1,R0
 023350                            LET R1 := DATA01                  ;SRC. POINTER
 023350  016701  155210                                                        MOV     DATA01,R1
 023354                            LET SIGNO := #0
 023354  005067  155070                                                        CLR     SIGNO
 023360                            LET DSLEN1 := DATA00 CLR.BY #177740        ;GET SRC.LEN
 023360  016767  155176  155126                                                MOV     DATA00,DSLEN1
 023366  042767  177740  155120                                                BIC     #177740,DSLEN1
 023374                            IF DSLEN1 EQ #0 THEN              ;SRC.LEN = 0
 023374  005767  155114                                                        TST     DSLEN1
 023400  001010                                                                BNE     $51014
 023402                               LET CC.EXP := CC.EXP SET.BY #ZBIT
 023402  052767  000004  155112                                                BIS     #ZBIT,CC.EXP
 023410                               LET R2.EXP := #0
 023410  005067  155114                                                        CLR     R2.EXP
 023414                               LET R3.EXP := #0
 023414  005067  155112                                                        CLR     R3.EXP
 023420                            ELSE                              ;SRC.LEN > 0
 023420  000417                                                                BR      $51015
 023422                                                               $51014:
 023422                               LET R0 := #DSBUF1 + #10. - DSLEN1
 023422  012700  064440                                                        MOV     #DSBUF1,R0
 023426  062700  000012                                                        ADD     #10.,R0
 023432  166700  155056                                                        SUB     DSLEN1,R0
 023436                               LET TEMPO := DSLEN1
 023436  016767  155052  155032                                                MOV     DSLEN1,TEMPO
 023444                               REPEAT
 023444                                                               $51016:
 023444                                  LET (R0)+ :B= (R1)+         ;XFER TO INTERMEDIATE
 023444  112120                                                                MOVB    (R1)+,(R0)+
 023446                                  LET TEMPO := TEMPO - #1
 023446  005367  155024                                                        DEC     TEMPO
 023452                               UNTIL TEMPO EQ #0
 023452  005767  155020                                                        TST     TEMPO
 023456  001372                                                                BNE     $51016
 023460                            ENDIF
 023460                                                               $51015:
 023460                            LET DATTYP := DATA00 CLR.BY #107777
 023460  016767  155076  154776                                                MOV     DATA00,DATTYP
 023466  042767  107777  154770                                                BIC     #107777,DATTYP

                                 ;GET THE SIGN OUT OF THE INTERMEDIATE RESULTS

 023474                            IF DSLEN1 NE #0 THEN              ;SRC.LEN > 0
```

```
        023474  005767  155014                                                          TST     DSLEN1
        023500  001533                                                                  BEQ     $51017
 4733   023502
                IF DATTYP EQ #0 OR DATTYP EQ #10000 THEN ;*** ZONED ***
        023502  005767  154756                                                          TST     DATTYP
        023506  001404                                                                  BEQ     $51020
        023510  026727  154750  010000                                                  CMP     DATTYP,#10000
        023516  001020                                                                  BNE     $51021
        023520                                                                  $51020:
 4734   023520
                    LET R0 := R0 - #1                                           ;POINT TO LAST BYTE
        023520  005300                                                                  DEC     R0
 4735   023522
                    LET TEMP0 :B= (R0) CLR.BY #17
        023522  111067  154750                                                          MOVB    (R0),TEMP0
        023526  142767  000017  154742                                                  BICB    #17,TEMP0
 4736   023534
                    IFB TEMP0 EQ #160 THEN                                        ;NEGATIVE
        023534  126727  154736  000160                                                  CMPB    TEMP0,#160
        023542  001006                                                                  BNE     $51022
 4737   023544
                        LET SIGN0 := #-1
        023544  012767  177777  154676                                                  MOV     #-1,SIGN0
 4738   023552
                        LET CC.EXP := CC.EXP SET.BY #NBIT
        023552  052767  000010  154742                                                  BIS     #NBIT,CC.EXP
 4739   023560
                    ENDIF
        023560                                                                  $51022:
 4740   023560
                ENDIF
        023560                                                                  $51021:
 4741   023560
                IF DATTYP EQ #20000 OR DATTYP EQ #30000 THEN                    ;*** OVERPUNCH ***
        023560  026727  154700  020000                                                  CMP     DATTYP,#20000
        023566  001404                                                                  BEQ     $51023
        023570  026727  154670  030000                                                  CMP     DATTYP,#30000
        023576  001044                                                                  BNE     $51024
        023600                                                                  $51023:
 4742   023600
                    IF DATTYP EQ #20000 THEN                                    ;TRAILING
        023600  026727  154660  020000                                                  CMP     DATTYP,#20000
        023606  001002                                                                  BNE     $51025
 4743   023610
                        LET R0 := R0 - #1
        023610  005300                                                                  DEC     R0
 4744   023612
                    ELSE                                                        ;LEADING
        023612  000406                                                                  BR      $51026
        023614                                                                  $51025:
 4745   023614
                        LET R0 := #DSBUF1 + #10. - DSLEN1
        023614  012700  064440                                                          MOV     #DSBUF1,R0
        023620  062700  000012                                                          ADD     #10.,R0
        023624  166700  154664                                                          SUB     DSLEN1,R0
 4746   023630
                    ENDIF
        023630                                                                  $51026:
 4747   023630
                LET TEMP0 :B= (R0)                                          ;GET SIGN/DIGIT
        023630  111067  154642                                                          MOVB    (R0),TEMP0
 4748   023634
                LET R1 := #SGNDGT                                           ;TABLE ADDRESS
        023634  012701  024406                                                          MOV     #SGNDGT,R1
 4749   023640
                REPEAT
        023640                                                                  $51027:
 4750   023640
                    LET TEMP1 :B= (R1)+
        023640  112167  154634                                                          MOVB    (R1)+,TEMP1
 4751   023644
                    LET R1 := R1 + #1                                           ;SKIP OVER ONE ENTRY
        023644  005201                                                                  INC     R1
 4752   023646
                UNTIL TEMP0 EQ TEMP1
        023646  026767  154624  154624                                                  CMP     TEMP0,TEMP1
        023654  001371                                                                  BNE     $51027
```

```
4753 023656                             LET R1 := R1 - #1                       ;BACK UP TO CORRECT ENTRY
     023656  005301                                                                    DEC     R1
4754 023660                             IFB #BIT07 SETIN (R1) THEN              ;NEGATIVE
     023660  132711  000200                                                            BITB    #BIT07,(R1)
     023664  001406                                                                    BEQ     $51030
4755 023666                                 LET CC.EXP := CC.EXP SET.BY #NBIT
     023666  052767  000010  154626                                                    BIS     #NBIT,CC.EXP
4756 023674                                 LET SIGNO := #-1
     023674  012767  177777  154546                                                    MOV     #-1,SIGNO
4757 023702                             ENDIF
     023702                                                                  $51030:
4758 023702                             LET (R0) :B= (R1) CLR.BY #BIT07         ;PUT CORRECT DIGIT INTO INTERMEDIATE RESULTS
     023702  111110                                                                    MOVB    (R1),(R0)
     023704  142710  000200                                                            BICB    #BIT07,(R0)
4759 023710                         ENDIF
     023710                                                                  $51024:
4760 023710                         IF DATTYP EQ #50000 THEN        ;LEADING SEPARATE
     023710  026727  154550  050000                                                    CMP     DATTYP,#50000
     023716  001003                                                                    BNE     $51031
4761 023720                             LET R1 := DATA01 - #1               ;POINT TO SIGN
     023720  016701  154640                                                            MOV     DATA01,R1
     023724  005301                                                                    DEC     R1
4762 023726                         ENDIF
     023726                                                                  $51031:
4763 023726                         IF DATTYP EQ #50000 OR DATTYP EQ #40000 THEN
     023726  026727  154532  050000                                                    CMP     DATTYP,#50000
     023734  001404                                                                    BEQ     $51032
     023736  026727  154522  040000                                                    CMP     DATTYP,#40000
     023744  001011                                                                    BNE     $51033
     023746                                                                  $51032:
4764 023746                             IFB (R1) EQ #55 THEN                ;NEGATIVE
     023746  121127  000055                                                            CMPB    (R1),#55
     023752  001006                                                                    BNE     $51034
4765 023754                                 LET SIGNO := #-1
     023754  012767  177777  154466                                                    MOV     #-1,SIGNO
4766 023762                                 LET CC.EXP := CC.EXP SET.BY #NBIT
     023762  052767  000010  154532                                                    BIS     #NBIT,CC.EXP
4767 023770                             ENDIF
     023770                                                                  $51034:
4768 023770                         ENDIF
     023770                                                                  $51033:
4769 023770                     ENDIF
     023770                                                                  $51017:
4770
4771                            ;CLEAR OUT UPPER NIBBLE OF INTERMEDIATE RESULTS
4772
4773 023770                         INCR R0 FROM #DSBUF1 TO #DSBUF1+37 BY #1
     023770  012700  064440                                                            MOV     #DSBUF1,R0
     023774  000401                                                                    BR      $51035
     023776                                                                  $51036:
     023776  005200                                                                    INC     R0
     024000                                                                  $51035:
     024000  020027  064477                                                            CMP     R0,#DSBUF1+37
     024004  003003                                                                    BGT     $51037
4774 024006                             LET (R0) :B= (R0) CLR.BY #360
     024006  142710  000360                                                            BICB    #360,(R0)
4775 024012                         ENDINC
```

```
        024012  000771                                                                  BR      $51036
        024014                                                                  $51037:
  4776
  4777                                  ;LOAD UP R2.EXP AND R3.EXP WITH FINAL RESULT
  4778
  4779 024014                             CALL CVTIL
        024014  004767  006170                                                          JSR     PC,CVTIL
  4780 024020                             IF SIGNO EQ #-1 THEN
        024020  026727  154424  177777                                                  CMP     SIGNO,#-1
        024026  001014                                                                  BNE     $51040
  4781 024030                                 LET CC.EXP := CC.EXP SET.BY #NBIT
        024030  052767  000010  154464                                                  BIS     #NBIT,CC.EXP
  4782 024036                                 IF R2.EXP NE #0 OR R3.EXP NE #0 THEN
        024036  005767  154466                                                          TST     R2.EXP
        024042  001003                                                                  BNE     $51041
        024044  005767  154462                                                          TST     R3.EXP
        024050  001403                                                                  BEQ     $51042
        024052                                                                  $51041:
  4783 024052                                     LET CC.EXP := CC.EXP SET.BY #CBIT
        024052  052767  000001  154442                                                  BIS     #CBIT,CC.EXP
  4784 024060                                 ENDIF
        024060                                                                  $51042:
  4785 024060                             ENDIF
        024060                                                                  $51040:
  4786
  4787                                  ;+
  4788                                  ; SET UP R0 THRU R3
  4789                                  ;-
  4790
  4791 024060                             LET R0 := DATA00            ;SRC.DSCR
        024060  016700  154476                                                          MOV     DATA00,R0
  4792 024064                             LET R1 := DATA01            ;SRC.DSCR
        024064  016701  154474                                                          MOV     DATA01,R1
  4793 024070                             LET R2 := #0
        024070  005002                                                                  CLR     R2
  4794 024072                             LET R3 := #0
        024072  005003                                                                  CLR     R3
  4795
  4796                                  ;+
  4797                                  ; DO THE CIS INSTRUCTION
  4798                                  ;-
  4799
  4800 024074                             INLINE <CVTNL>
        024074  076053                                                                 CVTNL
  4801
  4802
  4803                                  ;+
  4804                                  ;  CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.
  4805                                  ;-
  4806
  4807 024076                             LET CC := PSW
        024076  016767  153674  154414                                                  MOV     PSW,CC
  4808 024104                             LET CC := CC CLR.BY #177760
        024104  042767  177760  154406                                                  BIC     #177760,CC
  4809 024112                             IF CC NE CC.EXP THEN
        024112  026767  154402  154402                                                  CMP     CC,CC.EXP
        024120  001402                                                                  BEQ     $51043
```

```
4810 024122                             CALL ERROR
     024122  004767  033552                                    JSR     PC,ERROR
4811 024126                         ENDIF
     024126                                                $51043:
4812 024126                         IF R0 NE R0.EXP THEN
     024126  020067  154372                                    CMP     R0,R0.EXP
     024132  001402                                           BEQ     $51044
4813 024134                             CALL ERROR
     024134  004767  033540                                    JSR     PC,ERROR
4814 024140                         ENDIF
     024140                                                $51044:
4815 024140                         IF R1 NE R1.EXP THEN
     024140  020167  154362                                    CMP     R1,R1.EXP
     024144  001402                                           BEQ     $51045
4816 024146                             CALL ERROR
     024146  004767  033526                                    JSR     PC,ERROR
4817 024152                         ENDIF
     024152                                                $51045:
4818 024152                         IF R2 NE R2.EXP THEN
     024152  020267  154352                                    CMP     R2,R2.EXP
     024156  001402                                           BEQ     $51046
4819 024160                             CALL ERROR
     024160  004767  033514                                    JSR     PC,ERROR
4820 024164                         ENDIF
     024164                                                $51046:
4821 024164                         IF R3 NE R3.EXP THEN
     024164  020367  154342                                    CMP     R3,R3.EXP
     024170  001402                                           BEQ     $51047
4822 024172                             CALL ERROR
     024172  004767  033502                                    JSR     PC,ERROR
4823 024176                         ENDIF
     024176                                                $51047:
4824 024176                         IF R4 NE R4.EXP THEN
     024176  020467  154332                                    CMP     R4,R4.EXP
     024202  001402                                           BEQ     $51050
4825 024204                             CALL ERROR
     024204  004767  033470                                    JSR     PC,ERROR
4826 024210                         ENDIF
     024210                                                $51050:
4827 024210                         IF R5 NE R5.EXP THEN
     024210  020567  154322                                    CMP     R5,R5.EXP
     024214  001402                                           BEQ     $51051
4828 024216                             CALL ERROR
     024216  004767  033456                                    JSR     PC,ERROR
4829 024222                         ENDIF
     024222                                                $51051:
4830
4831 024222                         LET TSTCAS := TSTCAS + #4
     024222  062767  000004  154314                            ADD     #4,TSTCAS
4832 024230                         LET COUNT := COUNT - #1
     024230  005367  154234                                    DEC     COUNT
4833                               ;UNTIL COUNT EQ #0
4834 024234                           IF COUNT NE #0 THEN
     024234  005767  154230                                    TST     COUNT
     024240  001402                                           BEQ     $51052
4835 024242                             INLINE <JMP $5053>
     024242  000167  177010                                    JMP $5053
```

```
 4836 024246                            ENDIF
      024246                                                                $51052:
 4837
 4838                                  ;+
 4839                                  ; DO ONE INLINE CASE - CVTNLI
 4840                                  ;-
 4841
 4842 024246                            CALL   CLRREG
      024246  004767  032166                                                       JSR     PC,CLRREG
 4843 024252                            CALL   CLRDSB
      024252  004767  032416                                                       JSR     PC,CLRDSB
 4844 024256                            INLINE   <CVTNLI>
      024256  076153                                                              CVTNLI
 4845 024260                            INLINE   <.WORD   TCVNLI>
      024260  025242                                                              .WORD   TCVNLI
 4846 024262                            INLINE   <.WORD   TCVNLI+4>
      024262  025246                                                              .WORD   TCVNLI+4
 4847 024264                            LET CC := PSW
      024264  016767  153506  154226                                             MOV     PSW,CC
 4848 024272                            LET CC := CC CLR.BY #177760
      024272  042767  177760  154220                                             BIC     #177760,CC
 4849 024300                            IF CC NE #0 THEN
      024300  005767  154214                                                     TST     CC
      024304  001402                                                            BEQ     $51053
 4850 024306                                CALL ERROR
      024306  004767  033366                                                       JSR     PC,ERROR
 4851 024312                            ENDIF
      024312                                                                $51053:
 4852 024312                            IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN
      024312  005700                                                             TST     R0
      024314  001004                                                            BNE     $51054
      024316  005701                                                             TST     R1
      024320  001002                                                            BNE     $51054
      024322  005702                                                             TST     R2
      024324  001402                                                            BEQ     $51055
      024326                                                                $51054:
 4853 024326                                CALL ERROR
      024326  004767  033346                                                       JSR     PC,ERROR
 4854 024332                            ENDIF
      024332                                                                $51055:
 4855 024332                            IF R3 NE #0 OR R4 NE #0 OR R5 NE #0 THEN
      024332  005703                                                             TST     R3
      024334  001004                                                            BNE     $51056
      024336  005704                                                             TST     R4
      024340  001002                                                            BNE     $51056
      024342  005705                                                             TST     R5
      024344  001402                                                            BEQ     $51057
      024346                                                                $51056:
 4856 024346                                CALL ERROR
      024346  004767  033326                                                       JSR     PC,ERROR
 4857 024352                            ENDIF
      024352                                                                $51057:
 4858 024352                            IF TCVNLI+4 NE #32707 THEN
      024352  026727  000670  032707                                             CMP     TCVNLI+4,#32707
      024360  001402                                                            BEQ     $51060
 4859 024362                                CALL ERROR
      024362  004767  033312                                                       JSR     PC,ERROR
```

```
                                        ENDIF
                                        IF TCVPLI+6 NE #41072 THEN
                                            CALL ERROR
                                        ENDIF
                                        INLINE   <JMP   XCVTPN>
                                    ;+
                                    ; TABLE FOR INTERPRETING ACCEPTABLE SIGN/DIGIT COMBINATIONS FOR OVERPUNCH
                                    ;-
024406  173  000  060  SGNDGT: .BYTE   173,0,60,0,133,0,77,0          ;+0
024411  000  133  000
024414  077  000
024416  101  001  061          .BYTE   101,1,61,1                     ;+1
024421  001
024422  102  002  062          .BYTE   102,2,62,2                     ;+2
024425  002
024426  103  003  063          .BYTE   103,3,63,3                     ;+3
024431  003
024432  104  004  064          .BYTE   104,4,64,4                     ;+4
024435  004
024436  105  005  065          .BYTE   105,5,65,5                     ;+5
024441  005
024442  106  006  066          .BYTE   106,6,66,6                     ;+6
024445  006
024446  107  007  067          .BYTE   107,7,67,7                     ;+7
024451  007
024452  110  010  070          .BYTE   110,10,70,10                   ;+8
024455  010
024456  111  011  071          .BYTE   111,11,71,11                   ;+9
024461  011
024462  175  200  135          .BYTE   175,200,135,200,41,200,72,200  ;-0
024465  200  041  200
024470  072  200
024472  112  201               .BYTE   112,201                        ;-1
024474  113  202               .BYTE   113,202                        ;-2
024476  114  203               .BYTE   114,203                        ;-3
024500  115  204               .BYTE   115,204                        ;-4
024502  116  205               .BYTE   116,205                        ;-5
024504  117  206               .BYTE   117,206                        ;-6
024506  120  207               .BYTE   120,207                        ;-7
024510  121  210               .BYTE   121,210                        ;-8
024512  122  211               .BYTE   122,211                        ;-9

                                    ;+
                                    ; TABLE OF TEST CASES FOR CVTNL
                                    ;-
024514  000055                  TCVTNL: .WORD   45.              ;# OF TEST CASES FOR CVTNL
```

```
                                    $51060:
                                            CMP     TCVPLI+6,#41072
                                            BEQ     $51061
                                            JSR     PC,ERROR
                                    $51061:
                                            JMP     XCVTPN
```

```
4898 024516  000000                    .WORD   0                       ;SRC. DATA TYPE AND LENGTH
4899 024520  025003                    .WORD   CVTNL1                  ;POINTER TO SRC. STRING
4900
4901 024522  050000                    .WORD   50000           ;TEST CASE NUMBER 2
4902 024524  025005                    .WORD   CVTNL2
4903
4904 024526  040004                    .WORD   40004           ;TEST CASE NUMBER 3
4905 024530  025007                    .WORD   CVTNL3
4906
4907 024532  050004                    .WORD   50004           ;TEST CASE NUMBER 4
4908 024534  025015                    .WORD   CVTNL4
4909
4910 024536  030007                    .WORD   30007           ;TEST CASE NUMBER 5
4911 024540  025022                    .WORD   CVTNL5
4912
4913 024542  000002                    .WORD   2               ;TEST CASE NUMBER 6
4914 024544  025032                    .WORD   CVTNL6
4915
4916 024546  020005                    .WORD   20005           ;TEST CASE NUMBER 7
4917 024550  025035                    .WORD   CVTNL7
4918
4919 024552  010005                    .WORD   10005           ;TEST CASE NUMBER 8
4920 024554  025043                    .WORD   CVTNL8
4921
4922 024556  040004                    .WORD   40004           ;TEST CASE NUMBER 9
4923 024560  025051                    .WORD   CVTNL9
4924
4925 024562  000012                    .WORD   12              ;TEST CASE NUMBER 10
4926 024564  025057                    .WORD   CVTNLA
4927
4928 024566  020002                    .WORD   20002           ;TEST CASE NUMBER 11
4929 024570  025072                    .WORD   CVTNLB
4930
4931 024572  020002                    .WORD   20002           ;TEST CASE NUMBER 12
4932 024574  025075                    .WORD   CVTNLC
4933
4934 024576  020002                    .WORD   20002           ;TEST CASE NUMBER 13
4935 024600  025100                    .WORD   CVTNLD
4936
4937 024602  020002                    .WORD   20002           ;TEST CASE NUMBER 14
4938 024604  025103                    .WORD   CVTNLE
4939
4940 024606  020002                    .WORD   20002           ;TEST CASE NUMBER 15
4941 024610  025106                    .WORD   CVTNLF
4942
4943 024612  020002                    .WORD   20002           ;TEST CASE NUMBER 16
4944 024614  025111                    .WORD   CVTNLG
4945
4946 024616  020002                    .WORD   20002           ;TEST CASE NUMBER 17
4947 024620  025114                    .WORD   CVTNLH
4948
4949 024622  020002                    .WORD   20002           ;TEST CASE NUMBER 18
4950 024624  025117                    .WORD   CVTNL$
4951
4952 024626  020002                    .WORD   20002           ;TEST CASE NUMBER 19
4953 024630  025122                    .WORD   CVTNLJ
4954
```

```
 4955 024632  020002                    .WORD   20002           ;TEST CASE NUMBER 20
 4956 024634  025125                    .WORD   CVTNLK
 4957
 4958 024636  020002                    .WORD   20002           ;TEST CASE NUMBER 21
 4959 024640  025130                    .WORD   CVTNLL
 4960
 4961 024642  020002                    .WORD   20002           ;TEST CASE NUMBER 22
 4962 024644  025133                    .WORD   CVTNLM
 4963
 4964 024646  020002                    .WORD   20002           ;TEST CASE NUMBER 23
 4965 024650  025136                    .WORD   CVTNLN
 4966
 4967 024652  020002                    .WORD   20002           ;TEST CASE NUMBER 24
 4968 024654  025141                    .WORD   CVTNLO
 4969
 4970 024656  020002                    .WORD   20002           ;TEST CASE NUMBER 25
 4971 024660  025144                    .WORD   CVTNLP
 4972
 4973 024662  020002                    .WORD   20002           ;TEST CASE NUMBER 26
 4974 024664  025147                    .WORD   CVTNLQ
 4975
 4976 024666  020002                    .WORD   20002           ;TEST CASE NUMBER 27
 4977 024670  025152                    .WORD   CVTNLR
 4978
 4979 024672  020002                    .WORD   20002           ;TEST CASE NUMBER 28
 4980 024674  025155                    .WORD   CVTNLS
 4981
 4982 024676  020002                    .WORD   20002           ;TEST CASE NUMBER 29
 4983 024700  025160                    .WORD   CVTNLT
 4984
 4985 024702  020002                    .WORD   20002           ;TEST CASE NUMBER 30
 4986 024704  025163                    .WORD   CVTNLU
 4987
 4988 024706  020002                    .WORD   20002           ;TEST CASE NUMBER 31
 4989 024710  025166                    .WORD   CVTNLV
 4990
 4991 024712  020002                    .WORD   20002           ;TEST CASE NUMBER 32
 4992 024714  025171                    .WORD   CVTNLW
 4993
 4994 024716  020002                    .WORD   20002           ;TEST CASE NUMBER 33
 4995 024720  025174                    .WORD   CVTNLX
 4996
 4997 024722  020002                    .WORD   20002           ;TEST CASE NUMBER 34
 4998 024724  025177                    .WORD   CVTNLY
 4999
 5000 024726  020002                    .WORD   20002           ;TEST CASE NUMBER 35
 5001 024730  025202                    .WORD   CVTNLZ
 5002
 5003 024732  020002                    .WORD   20002           ;TEST CASE NUMBER 36
 5004 024734  025205                    .WORD   CVTNAA
 5005
 5006 024736  020002                    .WORD   20002           ;TEST CASE NUMBER 37
 5007 024740  025210                    .WORD   CVTNBB
 5008
 5009 024742  020002                    .WORD   20002           ;TEST CASE NUMBER 38
 5010 024744  025213                    .WORD   CVTNCC
 5011
```

CVTNL

```
5012 024746  020002                        .WORD   20002           ;TEST CASE NUMBER 39
5013 024750  025216                        .WORD   CVTNDD
5014
5015 024752  020002                        .WORD   20002           ;TEST CASE NUMBER 40
5016 024754  025221                        .WORD   CVTNEE
5017
5018 024756  020002                        .WORD   20002           ;TEST CASE NUMBER 41
5019 024760  025224                        .WORD   CVTNFF
5020
5021 024762  020002                        .WORD   20002           ;TEST CASE NUMBER 42
5022 024764  025227                        .WORD   CVTNGG
5023
5024 024766  020002                        .WORD   20002           ;TEST CASE NUMBER 43
5025 024770  025232                        .WORD   CVTNHH
5026
5027 024772  020002                        .WORD   20002           ;TEST CASE NUMBER 44
5028 024774  025235                        .WORD   CVTNII
5029
5030 024776  020002                        .WORD   20002           ;TEST CASE NUMBER 45
5031 025000  025240                        .WORD   CVTNJJ
5032
5033 025002     000                        .BYTE                           ;FOR LEADING SEPARATE SIGN
5034 025003     000                CVTNL1: .BYTE                                   ;DECIMAL NUMERIC SRC STRING 1
5035
5036 025004     055                        .BYTE   055
5037 025005     000                CVTNL2: .BYTE                                   ;DECIMAL NUMERIC SRC STRING 2
5038
5039 025006     000                        .BYTE
5040 025007     061     060     061 CVTNL3: .BYTE   061,060,061,060,053             ;DECIMAL NUMERIC SRC STRING 3
     025012     060     053
5041
5042 025014     053                        .BYTE   053
5043 025015     061     060     061 CVTNL4: .BYTE   061,060,061,065                 ;DECIMAL NUMERIC SRC STRING 4
     025020     065
5044
5045 025021     000                        .BYTE
5046 025022     112     020     000 CVTNL5: .BYTE   112,020,000,000,000,007,002     ;DECIMAL NUMERIC SRC STRING 5
     025025     000     000     007
     025030     002
5047
5048 025031     000                        .BYTE
5049 025032     001     171         CVTNL6: .BYTE   001,171                         ;DECIMAL NUMERIC SRC STRING 6
5050
5051 025034     000                        .BYTE
5052 025035     003     001     002 CVTNL7: .BYTE   003,001,002,000,107             ;DECIMAL NUMERIC SRC STRING 7
     025040     000     107
5053
5054 025042     000                        .BYTE
5055 025043     001     000     002 CVTNL8: .BYTE   001,000,002,000,067             ;DECIMAL NUMERIC SRC STRING 8
     025046     000     067
5056
5057 025050     000                        .BYTE
5058 025051     061     060     061 CVTNL9: .BYTE   061,060,061,060,053             ;DECIMAL NUMERIC SRC STRING 9
     025054     060     053
5059
5060 025056     000                        .BYTE
5061 025057     002     001     004 CVTNLA: .BYTE   002,001,004,007,004,010,003,006,004,067   ;DECIMAL NUMERIC SRC STR A
```

```
                025062  007     004     010
                025065  003     006     004
                025070  067
     5062
     5063 025071  000                             .BYTE
     5064 025072  061     173        CVTNLB:     .BYTE   61,173          ;DECIMAL NUMERIC SRC STRING 11
     5065
     5066 025074  000                             .BYTE
     5067 025075  061     060        CVTNLC:     .BYTE   61,60           ;DECIMAL NUMERIC SRC STRING 12
     5068
     5069 025077  000                             .BYTE
     5070 025100  061     133        CVTNLD:     .BYTE   61,133          ;DECIMAL NUMERIC SRC STRING 13
     5071
     5072 025102  000                             .BYTE
     5073 025103  061     077        CVTNLE:     .BYTE   61,77           ;DECIMAL NUMERIC SRC STRING 14
     5074
     5075 025105  000                             .BYTE
     5076 025106  061     101        CVTNLF:     .BYTE   61,101          ;DECIMAL NUMERIC SRC STRING 15
     5077
     5078 025110  000                             .BYTE
     5079 025111  061     061        CVTNLG:     .BYTE   61,61           ;DECIMAL NUMERIC SRC STRING 16
     5080
     5081 025113  000                             .BYTE
     5082 025114  061     102        CVTNLH:     .BYTE   61,102          ;DECIMAL NUMERIC SRC STRING 17
     5083
     5084 025116  000                             .BYTE
     5085 025117  061     062        CVTNL$:     .BYTE   61,62           ;DECIMAL NUMERIC SRC STRING 18
     5086
     5087 025121  000                             .BYTE
     5088 025122  061     103        CVTNLJ:     .BYTE   61,103          ;DECIMAL NUMERIC SRC STRING 19
     5089
     5090 025124  000                             .BYTE
     5091 025125  061     063        CVTNLK:     .BYTE   61,63           ;DECIMAL NUMERIC SRC STRING 20
     5092
     5093 025127  000                             .BYTE
     5094 025130  061     104        CVTNLL:     .BYTE   61,104          ;DECIMAL NUMERIC SRC STRING 21
     5095
     5096 025132  000                             .BYTE
     5097 025133  061     064        CVTNLM:     .BYTE   61,64           ;DECIMAL NUMERIC SRC STRING 22
     5098
     5099 025135  000                             .BYTE
     5100 025136  061     105        CVTNLN:     .BYTE   61,105          ;DECIMAL NUMERIC SRC STRING 23
     5101
     5102 025140  000                             .BYTE
     5103 025141  061     065        CVTNLO:     .BYTE   61,65           ;DECIMAL NUMERIC SRC STRING 24
     5104
     5105 025143  000                             .BYTE
     5106 025144  061     106        CVTNLP:     .BYTE   61,106          ;DECIMAL NUMERIC SRC STRING 25
     5107
     5108 025146  000                             .BYTE
     5109 025147  061     066        CVTNLQ:     .BYTE   61,66           ;DECIMAL NUMERIC SRC STRING 26
     5110
     5111 025151  000                             .BYTE
     5112 025152  061     107        CVTNLR:     .BYTE   61,107          ;DECIMAL NUMERIC SRC STRING 27
     5113
     5114 025154  000                             .BYTE
     5115 025155  061     067        CVTNLS:     .BYTE   61,67           ;DECIMAL NUMERIC SRC STRING 28
```

```
                                                .BYTE
025157  000
025160  061   110           CVTNLT:  .BYTE   61,110          ;DECIMAL NUMERIC SRC STRING 29
                                                .BYTE
025162  000
025163  061   070           CVTNLU:  .BYTE   61,70           ;DECIMAL NUMERIC SRC STRING 30
                                                .BYTE
025165  000
025166  061   111           CVTNLV:  .BYTE   61,111          ;DECIMAL NUMERIC SRC STRING 31
                                                .BYTE
025170  000
025171  061   071           CVTNLW:  .BYTE   61,71           ;DECIMAL NUMERIC SRC STRING 32
                                                .BYTE
025173  000
025174  061   175           CVTNLX:  .BYTE   61,175          ;DECIMAL NUMERIC SRC STRING 33
                                                .BYTE
025176  000
025177  061   135           CVTNLY:  .BYTE   61,135          ;DECIMAL NUMERIC SRC STRING 34
                                                .BYTE
025201  000
025202  061   041           CVTNLZ:  .BYTE   61,41           ;DECIMAL NUMERIC SRC STRING 35
                                                .BYTE
025204  000
025205  061   072           CVTNAA:  .BYTE   61,72           ;DECIMAL NUMERIC SRC STRING 36
                                                .BYTE
025207  000
025210  061   112           CVTNBB:  .BYTE   61,112          ;DECIMAL NUMERIC SRC STRING 37
                                                .BYTE
025212  000
025213  061   113           CVTNCC:  .BYTE   61,113          ;DECIMAL NUMERIC SRC STRING 38
                                                .BYTE
025215  000
025216  061   114           CVTNDD:  .BYTE   61,114          ;DECIMAL NUMERIC SRC STRING 39
                                                .BYTE
025220  000
025221  061   115           CVTNEE:  .BYTE   61,115          ;DECIMAL NUMERIC SRC STRING 40
                                                .BYTE
025223  000
025224  061   116           CVTNFF:  .BYTE   61,116          ;DECIMAL NUMERIC SRC STRING 41
                                                .BYTE
025226  000
025227  061   117           CVTNGG:  .BYTE   61,117          ;DECIMAL NUMERIC SRC STRING 42
                                                .BYTE
025231  000
025232  061   120           CVTNHH:  .BYTE   61,120          ;DECIMAL NUMERIC SRC STRING 43
                                                .BYTE
025234  000
025235  061   121           CVTNII:  .BYTE   61,121          ;DECIMAL NUMERIC SRC STRING 44
                                                .BYTE
025237  000
025240  061   122           CVTNJJ:  .BYTE   61,122          ;DECIMAL NUMERIC SRC STRING 45

                                                .EVEN

025242                      TCVNLI:                  ;INLINE TEST CASE
025242  000012                       .WORD   000012          ;DATA TYPE = SIGNED ZONED, LENGTH = 12
025244  025252                       .WORD   NL1             ;ADDRESS OF DECIMAL SRC STRING
```

CVTNL

```
 5173 025246  000000                          .WORD   0               ;LONG INTEGER RESULT, LOW
 5174 025250  000000                          .WORD   0               ;LONG INTEGER RESULT, HI, AND SIGN
 5175
 5176 025252     061     061     061  NL1:    .BYTE   61,61,61,61,61,61,61,61,61,61           ;  = +1111111111 IN BCD
      025255     061     061     061
      025260     061     061     061
      025263     061
 5177                             .EVEN
 5178
```

```
                                .SBTTL  CVTPN
                                ;;*****************************************************************
                                ;
                                ;  CVTPN TEST
                                ;
                                ;;*****************************************************************

                                ROUTINE XCVTPN
                                                                                        XCVTPN:
                                   INLINE  <SCOPE>
                                                                                        SCOPE
                                   INLINE  <NOP>
                                                                                        NOP
                                   LET $TIMES := #20                    ;ITERATION COUNT
                                                                                        MOV     #20,$TIMES
                                   LET COUNT := TCVTPN                  ;GET TEST COUNT
                                                                                        MOV     TCVTPN,COUNT
                                   LET TSTCAS := #TCVTPN+2                      ;POINT TO FIRST TEST CASE
                                                                                        MOV     #TCVTPN+2,TSTCAS
                                ;REPEAT
                                   INLINE  <$5054:>
                                                                                        $5054:

                                ;+
                                ; GET TEST DATA FROM TABLE
                                ;-

                                   LET R0 := TSTCAS
                                                                                        MOV     TSTCAS,R0
                                   LET DATA00 := (R0)+                  ;SRC DATA TYPE AND LENGTH
                                                                                        MOV     (R0)+,DATA00
                                   LET DATA01 := (R0)+                  ;POINTER TO PACKED SRC STRING
                                                                                        MOV     (R0)+,DATA01
                                   LET DATA02 := (R0)                   ;DST.LEN
                                                                                        MOV     (R0),DATA02

                                ;+
                                ; SET UP R0.EXP, R1.EXP, R3.EXP, R4.EXP, R5.EXP, R4, AND R5.
                                ;-

                                   LET R0.EXP := #0
                                                                                        CLR     R0.EXP
                                   LET R1.EXP := #0
                                                                                        CLR     R1.EXP
                                   LET R3.EXP := #DS.DST
                                                                                        MOV     #DS.DST,R3.EXP
                                   IF #1 SETIN COUNT THEN
                                                                                        BIT     #1,COUNT
                                                                                        BEQ     $51064
                                      LET R4 := #-1
                                                                                        MOV     #-1,R4
                                      LET R5 := #-1
                                                                                        MOV     #-1,R5
                                   ELSE
                                                                                        BR      $51065
                                                                                $51064:
                                      LET R4 := #0
```

```
     025372  005004                                                                  CLR     R4
     025374                              LET R5 := #0
     025374  005005                                                                  CLR     R5
     025376                          ENDIF
     025376                                                               $51065:
     025376                          LET R4.EXP := R4
     025376  010467  153132                                                          MOV     R4,R4.EXP
     025402                          LET R5.EXP := R5
     025402  010567  153130                                                          MOV     R5,R5.EXP




                                 ;+
                                 ; INITIALIZE THE DECIMAL STRING BUFFERS AND CC.EXP.
                                 ;-

     025406                          CALL CLRDSB
     025406  004767  031262                                                          JSR     PC,CLRDSB
     025412                          LET CC.EXP := #0
     025412  005067  153104                                                          CLR     CC.EXP



                                 ;+
                                 ; PUT THE SOURCE LENGTH INTO DSLEN1; PUT THE DESTINATION
                                 ; LENGTH INTO DSLEN2.
                                 ;-

     025416                          LET DSLEN1 := DATA00 CLR.BY #70000
     025416  016767  153140  153070                                                  MOV     DATA00,DSLEN1
     025424  042767  070000  153062                                                  BIC     #70000,DSLEN1
     025432                          LET DSLEN2 := DATA02
     025432  016767  153130  153056                                                  MOV     DATA02,DSLEN2



                                 ;+
                                 ; LET SIGNO REFLECT THE SIGN OF THE SOURCE STRING.
                                 ;-

     025440                          LET R0 := DSLEN1
     025440  016700  153050                                                          MOV     DSLEN1,R0
     025444                          INLINE   <ASR        R0>
     025444  006200                                                                  ASR     R0
     025446                          LET R0 := R0 + DATA01               ;POINT TO THE END OF THE STRING
     025446  066700  153112                                                          ADD     DATA01,R0
     025452                          LET R1 :B= (R0) CLR.BY #360      ;GET THE SIGN NIBBLE
     025452  111001                                                                  MOVB    (R0),R1
     025454  142701  000360                                                          BICB    #360,R1
     025460                          IF DATA00 HIS #70000 OR DSLEN1 EQ #0 THEN
     025460  026727  153076  070000                                                  CMP     DATA00,#70000
     025466  103003                                                                  BHIS    $51066
     025470  005767  153020                                                          TST     DSLEN1
     025474  001003                                                                  BNE     $51067
     025476                                                               $51066:
     025476                              LET SIGNO := #0
     025476  005067  152746                                                          CLR     SIGNO
     025502                          ELSE
     025502  000411                                                                  BR      $51070
     025504                                                               $51067:
```

```
 5251 025504                            IF R1 EQ #14 THEN
      025504  020127  000014                                                    CMP     R1,#14
      025510  001003                                                            BNE     $51071
 5252 025512                                LET SIGNO := #0
      025512  005067  152732                                                    CLR     SIGNO
 5253 025516                            ELSE
      025516  000403                                                            BR      $51072
      025520                                                            $51071:
 5254 025520                                LET SIGNO := #-1
      025520  012767  177777  152722                                            MOV     #-1,SIGNO
 5255 025526                            ENDIF
      025526                                                            $51072:
 5256 025526                        ENDIF
      025526                                                            $51070:
 5257
 5258
 5259                               ;+
 5260                               ; CALCULATE THE NUMBER OF CHARACTERS TO BE MOVED INTO DS.EXP;
 5261                               ; USE R0.
 5262                               ;-
 5263
 5264 025526                            IF DSLEN2 HIS DSLEN1 THEN       ;DST.LEN >= SRC.LEN
      025526  026767  152764  152760                                            CMP     DSLEN2,DSLEN1
      025534  103403                                                            BLO     $51073
 5265 025536                                LET R0 := DSLEN1
      025536  016700  152752                                                    MOV     DSLEN1,R0
 5266 025542                            ELSE                            ;SRC.LEN > DST.LEN
      025542  000402                                                            BR      $51074
      025544                                                            $51073:
 5267 025544                                LET R0 := DSLEN2
      025544  016700  152746                                                    MOV     DSLEN2,R0
 5268 025550                            ENDIF
      025550                                                            $51074:
 5269
 5270
 5271                               ;+
 5272                               ; MAKE R1 POINT TO THE END OF THE DESTINATION IN DS.EXP;
 5273                               ; MAKE R2 POINT TO THE END OF THE SOURCE STRING IN THE
 5274                               ; TABLE OF TEST CASES.
 5275                               ;-
 5276
 5277 025550                            LET R1 := #DS.EXP + DATA02 - #1
      025550  012701  064606                                                    MOV     #DS.EXP,R1
      025554  066701  153006                                                    ADD     DATA02,R1
      025560  005301                                                            DEC     R1
 5278 025562                            LET R2 := DSLEN1
      025562  016702  152726                                                    MOV     DSLEN1,R2
 5279 025566                            INLINE   <ASR        R2>
      025566  006202                                                            ASR     R2
 5280 025570                            LET R2 := R2 + DATA01
      025570  066702  152770                                                    ADD     DATA01,R2
 5281
 5282
 5283                               ;+
 5284                               ; MOVE THE LEAST SIGNIFICANT BCD FROM THE SOURCE STRING
 5285                               ; TO THE EXPECTED DESTINATION BUFFER, IN UNPACKED FORMAT.
 5286                               ;-
```

```
 5287
 5288 025574                            LET R3 :B= (R2)
 -    025574  111203                                                      MOVB    (R2),R3
 5289 025576                            INLINE   <ASH        #-4,R3>
      025576  072327  177774                                              ASH     #-4,R3
 5290 025602                            LET (R1) :B= R3 SET.BY #60 CLR.BY #300
      025602  110311                                                      MOVB    R3,(R1)
      025604  152711  000060                                              BISB    #60,(R1)
      025610  142711  000300                                              BICB    #300,(R1)
 5291 025614                            LET R0 := R0 - #1                       ;DECREMENT COUNT
      025614  005300                                                      DEC     R0
 5292
 5293
 5294
 5295                                ;+
 5296                                ; MOVE THE REST OF THE SOURCE TO THE EXPECTED DESTINATION
 5297                                ; BUFFER, IN UNPACKED FORMAT, USING R0 AS A COUNTER FOR
 5298                                ; THE NUMBER OF CHARACTERS TO MOVE.
 5299                                ;-
 5300
 5301 025616                            WHILE R0 GT #0 DO
      025616                                                        $51075:
      025616  005700                                                      TST     R0
      025620  003425                                                      BLE     $51076
 5302 025622                                LET R1 := R1 - #1                   ;DECREMENT DEST. POINTER
      025622  005301                                                      DEC     R1
 5303 025624                                LET R2 := R2 - #1                   ;DECREMENT SRC. POINTER
      025624  005302                                                      DEC     R2
 5304 025626                                LET (R1) :B= (R2) CLR.BY #300 SET.BY #60
      025626  111211                                                      MOVB    (R2),(R1)
      025630  142711  000300                                              BICB    #300,(R1)
      025634  152711  000060                                              BISB    #60,(R1)
 5305 025640                                LET R0 := R0 - #1                   ;DECREMENT COUNT
      025640  005300                                                      DEC     R0
 5306 025642                                IF R0 HI #0 THEN                    ;COUNT > 0
      025642  005700                                                      TST     R0
      025644  101412                                                      BLOS    $51077
 5307 025646                                    LET R1 := R1 - #1               ;DECREMENT DEST. POINTER
      025646  005301                                                      DEC     R1
 5308 025650                                    LET R3 :B= (R2)
      025650  111203                                                      MOVB    (R2),R3
 5309 025652                                    INLINE   <ASH        #-4,R3>
      025652  072327  177774                                              ASH     #-4,R3
 5310 025656                                    LET (R1) :B= R3 CLR.BY #300 SET.BY #60
      025656  110311                                                      MOVB    R3,(R1)
      025660  142711  000300                                              BICB    #300,(R1)
      025664  152711  000060                                              BISB    #60,(R1)
 5311 025670                                    LET R0 := R0 - #1               ;DECREMENT COUNT
      025670  005300                                                      DEC     R0
 5312 025672                                ENDIF
      025672                                                        $51077:
 5313 025672                            ENDDO
      025672  000751                                                      BR      $51075
      025674                                                        $51076:
 5314
 5315                                ;+
 5316                                ; FILL THE REST OF THE DESTINATION WITH LEADING ZEROES (60'S).
```

```
 5317                                   ;-
 5318
 5319 025674                               WHILE R1 GT #DS.EXP DO
      025674                                                                $51100:
      025674  020127  064606                                                      CMP     R1,#DS.EXP
      025700  003403                                                              BLE     $51101
 5320 025702                                   LET -(R1) :B= #60
      025702  112741  000060                                                    MOVB    #60,-(R1)
 5321 025706                               ENDDO
      025706  000772                                                              BR      $51100
      025710                                                                $51101:
 5322
 5323
 5324                                   ;+
 5325                                   ; PUT THE EXPECTED VALUE OF THE NBIT INTO CC.EXP
 5326                                   ;-
 5327
 5328 025710                               IF DSLEN2 HI #0 AND SIGNO EQ #-1 THEN
      025710  005767  152602                                                      TST     DSLEN2
      025714  101407                                                              BLOS    $51102
      025716  026727  152526  177777                                            CMP     SIGNO,#-1
      025724  001003                                                              BNE     $51102
 5329 025726                                   LET CC.EXP := CC.EXP SET.BY #NBIT
      025726  052767  000010  152566                                            BIS     #NBIT,CC.EXP
 5330 025734                               ENDIF
      025734                                                                $51102:
 5331
 5332
 5333                                   ;+
 5334                                   ; PUT THE EXPECTED VALUE OF THE ZBIT INTO CC.EXP
 5335                                   ;-
 5336
 5337
 5338 025734                               IF DSLEN1 EQ #0 OR DSLEN2 EQ #0 THEN
      025734  005767  152554                                                      TST     DSLEN1
      025740  001403                                                              BEQ     $51103
      025742  005767  152550                                                      TST     DSLEN2
      025746  001003                                                              BNE     $51104
      025750                                                                $51103:
 5339 025750                                   LET CC.EXP := CC.EXP SET.BY #ZBIT
      025750  052767  000004  152544                                            BIS     #ZBIT,CC.EXP
 5340 025756                               ENDIF
      025756                                                                $51104:
 5341
 5342
 5343                                   ;+
 5344                                   ; PUT THE EXPECTED VALUE OF THE VBIT INTO CC.EXP
 5345                                   ;-
 5346
 5347 025756                               IF DSLEN1 HI DSLEN2 THEN             ;SRC.LEN > DST.LEN
      025756  026767  152532  152532                                            CMP     DSLEN1,DSLEN2
      025764  101403                                                              BLOS    $51105
 5348 025766                                   LET CC.EXP := CC.EXP SET.BY #VBIT
      025766  052767  000002  152526                                            BIS     #VBIT,CC.EXP
 5349 025774                               ENDIF
      025774                                                                $51105:
 5350
```

```
 5351
 5352                                   ;+
 5353                                   ; SAVE THE FIRST AND LAST BYTES OF THE EXPECTED DESTINATION
 5354                                   ; IN DSTMP0 AND DSTMP1.
 5355                                   ;-
 5356
 5357 025774                              LET DSTMP0 := DS.EXP
      025774  016767  036606  152552                                          MOV     DS.EXP,DSTMP0
 5358 026002                              LET R0 := #DS.EXP + DATA02 - #1
      026002  012700  064606                                                  MOV     #DS.EXP,R0
      026006  066700- 152554                                                  ADD     DATA02,R0
      026012  005300                                                          DEC     R0
 5359 026014                              LET DSTMP1 :B= (R0)
      026014  111067  152535                                                  MOVB    (R0),DSTMP1
 5360
 5361
 5362                                   ;+
 5363                                   ; DO THE CVTPN INSTRUCTION, CONVERTING EACH PACKED SOURCE
 5364                                   ; STRING INTO ALL SIX NUMERIC DATA TYPES.  DS.EXP WILL HAVE
 5365                                   ; TO BE ALTERED FOR EACH DATA TYPE.
 5366                                   ;-
 5367
 5368                                   ;INCR DATTYP FROM #0 TO #50000 BY #10000
 5369 026020                              INLINE   <CLR        DATTYP>
      026020  005067  152440                                                  CLR     DATTYP
 5370 026024                              INLINE   <BR $5402>
      026024  000412                                                         BR      $5402
 5371 026026                              INLINE   <$5401:>
      026026                                                                 $5401:
 5372
 5373                                   ;CLEAR THE DESTINATION BUFFER
 5374
 5375 026026                              LET R0 := #DS.DST-1
      026026  012700  064543                                                          MOV     #DS.DST-1,R0
 5376 026032                              WHILE R0 LOS #DS.DST+37 DO
      026032                                                                 $51106:
      026032  020027  064603                                                          CMP     R0,#DS.DST+37
      026036  101002                                                                  BHI     $51107
 5377 026040                                  LET (R0)+ :B= #0
      026040  105020                                                                  CLRB    (R0)+
 5378 026042                              ENDDO
      026042  000773                                                                 BR      $51106
      026044                                                                 $51107:
 5379
 5380                                   ;INCREMENT DATTYP
 5381
 5382 026044                              LET DATTYP := DATTYP + #10000
      026044  062767  010000  152412                                                  ADD     #10000,DATTYP
 5383 026052                              INLINE   <$5402:>
      026052                                                                         $5402:
 5384 026052                              IF DATTYP HI #50000 THEN
      026052  026727  152406  050000                                                  CMP     DATTYP,#50000
      026060  101402                                                                 BLOS    $51110
 5385 026062                                  INLINE   <JMP    $5403>
      026062  000167  000676                                                         JMP     $5403
 5386 026066                              ENDIF
      026066                                                                 $51110:
```

```
                                ;RESTORE FIRST AND LAST BYTES OF DESTINATION
                                
                                   LET DS.EXP :B= DSTMP0
026066  116767  152462  036512                                                  MOVB    DSTMP0,DS.EXP
                                   IF DATA02 NE #0 THEN
026074  005767  152466                                                          TST     DATA02
026100  001406                                                                  BEQ     $51111
                                      LET R0 := #DS.EXP-1 + DATA02
026102  012700  064605                                                          MOV     #DS.EXP-1,R0
026106  066700  152454                                                          ADD     DATA02,R0
                                      LET (R0) :B= DSTMP1
026112  116710  152437                                                          MOVB    DSTMP1,(R0)
                                   ENDIF
026116                                                                  $51111:
                                
                                ;*** SIGNED ZONED ***
                                
                                   IF DATTYP EQ #0 THEN
026116  005767  152342                                                          TST     DATTYP
026122  001022                                                                  BNE     $51112
                                      LET DS.EXP-1 :B= #0               ;RESTORE AFTER LEADING SEPARATE
026124  105067  036455                                                          CLRB    DS.EXP-1
                                      IF DATA02 HI #0 THEN              ;DST.LEN > 0
026130  005767  152432                                                          TST     DATA02
026134  101415                                                                  BLOS    $51113
                                         LET R0 := #DS.EXP-1 + DATA02
026136  012700  064605                                                          MOV     #DS.EXP-1,R0
026142  066700  152420                                                          ADD     DATA02,R0
                                         IF #NBIT NOTSETIN CC.EXP THEN                ;SIGN IS +
026146  032767  000010  152346                                                      BIT     #NBIT,CC.EXP
026154  001003                                                                      BNE     $51114
                                            LET (R0) :B= (R0) SET.BY #60
026156  152710  000060                                                              BISB    #60,(R0)
                                         ELSE                                 ;SIGN IS -
026162  000402                                                                      BR      $51115
026164                                                                  $51114:
                                            LET (R0) :B= (R0) SET.BY #160
026164  152710  000160                                                              BISB    #160,(R0)
                                         ENDIF
026170                                                                  $51115:
                                      ENDIF
026170                                                                  $51113:
                                   ENDIF
026170                                                                  $51112:
                                
                                ;*** UNSIGNED ZONED ***
                                
                                   IF DATTYP EQ #10000 THEN
026170  026727  152270  010000                                                  CMP     DATTYP,#10000
026176  001014                                                                  BNE     $51116
                                      IF DATA02 HI #0 THEN              ;DST.LEN > 0
026200  005767  152362                                                          TST     DATA02
026204  101406                                                                  BLOS    $51117
                                         LET R0 := #DS.EXP-1 + DATA02
026206  012700  064605                                                          MOV     #DS.EXP-1,R0
026212  066700  152350                                                          ADD     DATA02,R0
```

```
5415 026216                             LET (R0) :B= (R0) SET.BY #60   ;PUT IN UNSIGNED "SIGN"
     026216  152710  000060                                                      BISB    #60,(R0)
5416 026222                           ENDIF
     026222                                                                $51117:
5417 026222                           LET CC.EXP := CC.EXP CLR.BY #NBIT
     026222  042767  000010  152272                                              BIC     #NBIT,CC.EXP
5418 026230                         ENDIF
     026230                                                                $51116:
5419
5420                              ;*** TRAILING OVERPUNCH ***
5421
5422 026230                         IF DATTYP EQ #20000 THEN
     026230  026727  152230  020000                                              CMP     DATTYP,#20000
     026236  001047                                                            BNE     $51120
5423 026240                             IF DSLEN2 HI #0 AND SIGNO EQ #-1 THEN
     026240  005767  152252                                                    TST     DSLEN2
     026244  101407                                                            BLOS    $51121
     026246  026727  152176  177777                                            CMP     SIGNO,#-1
     026254  001003                                                            BNE     $51121
5424 026256                               LET CC.EXP := CC.EXP SET.BY #NBIT
     026256  052767  000010  152236                                            BIS     #NBIT,CC.EXP
5425 026264                             ENDIF
     026264                                                                $51121:
5426 026264                             IF DATA02 HI #0 THEN                ;DST.LEN > 0
     026264  005767  152276                                                    TST     DATA02
     026270  101432                                                            BLOS    $51122
5427 026272                                 IF DSLEN1 NE #0 AND SIGNO EQ #-1 THEN
     026272  005767  152216                                                    TST     DSLEN1
     026276  001407                                                            BEQ     $51123
     026300  026727  152144  177777                                            CMP     SIGNO,#-1
     026306  001003                                                            BNE     $51123
5428 026310                                   LET CC.EXP := CC.EXP SET.BY #NBIT
     026310  052767  000010  152204                                            BIS     #NBIT,CC.EXP
5429 026316                                 ENDIF
     026316                                                                $51123:
5430 026316                                 LET R0 := #DS.EXP-1 + DATA02    ;POINT TO LAST DST BYTE
     026316  012700  064605                                                    MOV     #DS.EXP-1,R0
     026322  066700  152240                                                    ADD     DATA02,R0
5431 026326                                 LET R1 :B= DSTMP1 CLR.BY #60          ;GET LSD
     026326  116701  152223                                                    MOVB    DSTMP1,R1
     026332  142701  000060                                                    BICB    #60,R1
5432 026336                                 IF SIGNO EQ #-1 THEN
     026336  026727  152106  177777                                            CMP     SIGNO,#-1
     026344  001002                                                            BNE     $51124
5433 026346                                     LET R1 := R1 + #10.           ;FORM OVPNCH TABLE OFFSET
     026346  062701  000012                                                    ADD     #10.,R1
5434 026352                                 ENDIF
     026352                                                                $51124:
5435 026352                                 LET (R0) :B= OVPNCH(R1)         ;GET CORRECT SIGN/DIGIT
     026352  116110  056330                                                    MOVB    OVPNCH(R1),(R0)
5436 026356                             ENDIF
     026356                                                                $51122:
5437 026356                         ENDIF
     026356                                                                $51120:
5438
5439                              ;*** LEADING OVERPUNCH ***
5440
```

```
5441 026356                          IF DATTYP EQ #30000 THEN
     026356  026727  152102  030000                                          CMP     DATTYP,#30000
     026364  001020                                                          BNE     $51125
5442 026366                              IF DATA02 HI #0 THEN          ;DST.LEN > 0
     026366  005767  152174                                                  TST     DATA02
     026372  101415                                                          BLOS    $51126
5443 026374                                  LET R1 :B= DSTMP0 CLR.BY #60          ;GET MSD
     026374  116701  152154                                                  MOVB    DSTMP0,R1
     026400  142701  000060                                                  BICB    #60,R1
5444 026404                                  IF #NBIT SETIN CC.EXP THEN
     026404  032767  000010  152110                                          BIT     #NBIT,CC.EXP
     026412  001402                                                          BEQ     $51127
5445 026414                                      LET R1 := R1 + #10.       ;FORM OVPNCH TABLE OFFSET
     026414  062701  000012                                                  ADD     #10.,R1
5446 026420                                  ENDIF
     026420                                                          $51127:
5447 026420                                  LET DS.EXP :B= OVPNCH(R1)  ;GET CORRECT SIGN/DIGIT
     026420  116167  056330  036160                                          MOVB    OVPNCH(R1),DS.EXP
5448 026426                              ENDIF
     026426                                                          $51126:
5449 026426                          ENDIF
     026426                                                          $51125:
5450
5451                                ;*** TRAILING SEPARATE ***
5452
5453 026426                          IF DATTYP EQ #40000 THEN
     026426  026727  152032  040000                                          CMP     DATTYP,#40000
     026434  001015                                                          BNE     $51130
5454 026436                              LET R0 := #DS.EXP + DATA02
     026436  012700  064606                                                  MOV     #DS.EXP,R0
     026442  066700  152120                                                  ADD     DATA02,R0
5455 026446                              IF #NBIT NOTSETIN CC.EXP THEN          ;SIGN IS +
     026446  032767  000010  152046                                          BIT     #NBIT,CC.EXP
     026454  001003                                                          BNE     $51131
5456 026456                                  LET (R0) :B= #53
     026456  112710  000053                                                  MOVB    #53,(R0)
5457 026462                              ELSE                          ;SIGN IS -
     026462  000402                                                          BR      $51132
     026464                                                          $51131:
5458 026464                                  LET (R0) :B= #55
     026464  112710  000055                                                  MOVB    #55,(R0)
5459 026470                              ENDIF
     026470                                                          $51132:
5460 026470                          ENDIF
     026470                                                          $51130:
5461
5462                                ;*** LEADING SEPARATE ***
5463
5464 026470                          IF DATTYP EQ #50000 THEN
     026470  026727  151770  050000                                          CMP     DATTYP,#50000
     026476  001020                                                          BNE     $51133
5465 026500                              LET R0 := #DS.EXP + DATA02
     026500  012700  064606                                                  MOV     #DS.EXP,R0
     026504  066700  152056                                                  ADD     DATA02,R0
5466 026510                              LET (R0) :B= #0           ;RESTORE AFTER TRAILING SEPARATE
     026510  105010                                                          CLRB    (R0)
5467 026512                              IF #NBIT NOTSETIN CC.EXP THEN          ;SIGN = +
```

```
                                                                                                                BIT     #NBIT,CC.EXP
                                                                                                                BNE     $51134
                                                LET DS.EXP-1 :B= #53
                                                                                                                MOVB    #53,DS.EXP-1
                                            ELSE                                    ;SIGN = -
                                                                                                                BR      $51135
                                                                                                        $51134:
                                                LET DS.EXP-1 :B= #55
                                                                                                                MOVB    #55,DS.EXP-1
                                            ENDIF
                                                                                                        $51135:
                                        ENDIF
                                                                                                        $51133:


                                    ;+
                                    ; SET UP R2.EXP, R0, R1, R2, AND R3; DO THE INSTRUCTION.
                                    ;-

                                        LET R0 := DATA00                    ;SRC  DATA TYPE AND LENGTH
                                                                                                                MOV     DATA00,R0
                                        LET R1 := DATA01                    ;SRC  POINTER
                                                                                                                MOV     DATA01,R1
                                        LET R2 := DATA02 + DATTYP           ;DST DATA TYPE AND LENGTH
                                                                                                                MOV     DATA02,R2
                                                                                                                ADD     DATTYP,R2
                                        LET R3 := #DS.DST                   ;DST POINTER
                                                                                                                MOV     #DS.DST,R3
                                        LET R2.EXP := R2
                                                                                                                MOV     R2,R2.EXP
                                        INLINE  <CVTPN>
                                                                                                                CVTPN


                                    ;+
                                    ;  CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.
                                    ;-

                                        LET CC := PSW
                                                                                                                MOV     PSW,CC
                                        LET CC := CC CLR.BY #177760
                                                                                                                BIC     #177760,CC
                                        IF CC NE CC.EXP THEN
                                                                                                                CMP     CC,CC.EXP
                                                                                                                BEQ     $51136
                                            CALL ERROR
                                                                                                                JSR     PC,ERROR
                                        ENDIF
                                                                                                        $51136:
                                        IF R0 NE R0.EXP THEN
                                                                                                                CMP     R0,R0.EXP
                                                                                                                BEQ     $51137
                                            CALL ERROR
                                                                                                                JSR     PC,ERROR
                                        ENDIF
                                                                                                        $51137:
                                        IF R1 NE R1.EXP THEN
                                                                                                                CMP     R1,R1.EXP
```

```
                     026640  001402
5499                 026642                          CALL ERROR
                     026642  004767  031032
5500                 026646                      ENDIF
                     026646
5501                 026646                      IF R2 NE R2.EXP THEN
                     026646  020267  151656
                     026652  001402
5502                 026654                          CALL ERROR
                     026654  004767  031020
5503                 026660                      ENDIF
                     026660
5504                 026660                      IF R3 NE R3.EXP THEN
                     026660  020367  151646
                     026664  001402
5505                 026666                          CALL ERROR
                     026666  004767  031006
5506                 026672                      ENDIF
                     026672
5507                 026672                      IF R4 NE R4.EXP THEN
                     026672  020467  151636
                     026676  001402
5508                 026700                          CALL ERROR
                     026700  004767  030774
5509                 026704                      ENDIF
                     026704
5510                 026704                      IF R5 NE R5.EXP THEN
                     026704  020567  151626
                     026710  001402
5511                 026712                          CALL ERROR
                     026712  004767  030762
5512                 026716                      ENDIF
                     026716
5513
5514                                         ;COMPARE DS.DST WITH DS.EXP
5515
5516                 026716                      LET R0 := #DS.EXP-1
                     026716  012700  064605
5517                 026722                      LET R1 := #DS.DST-1
                     026722  012701  064543
5518                 026726                      LET ERRFLG := #0
                     026726  005067  151524
5519                 026732                      WHILE R0 NE #DS.EXP+40 AND ERRFLG EQ #0 DO
                     026732
                     026732  020027  064646
                     026736  001410
                     026740  005767  151512
                     026744  001005
5520                 026746                          IFB (R0)+ NE (R1)+ THEN
                     026746  122021
                     026750  001402
5521                 026752                              CALL ERROR
                     026752  004767  030722
5522                 026756                          ENDIF
                     026756
5523                 026756                      ENDDO
                     026756  000765
```

```
                BEQ     $51140
                JSR     PC,ERROR
$51140:
                CMP     R2,R2.EXP
                BEQ     $51141
                JSR     PC,ERROR
$51141:
                CMP     R3,R3.EXP
                BEQ     $51142
                JSR     PC,ERROR
$51142:
                CMP     R4,R4.EXP
                BEQ     $51143
                JSR     PC,ERROR
$51143:
                CMP     R5,R5.EXP
                BEQ     $51144
                JSR     PC,ERROR
$51144:


                MOV     #DS.EXP-1,R0
                MOV     #DS.DST-1,R1
;CLEAR THE ERROR FLAG
                CLR     ERRFLG
$51145:
                CMP     R0,#DS.EXP+40
                BEQ     $51146
                TST     ERRFLG
                BNE     $51146
                CMPB    (R0)+,(R1)+
                BEQ     $51147
                JSR     PC,ERROR
$51147:
                BR      $51145
```

```
026760                                                                  $51146:
                                ;ENDINC
026760                             INLINE  <JMP  $5401>
026760  000167  177042                                                       JMP  $5401
026764                             INLINE  <$5403:>
026764                                                                       $5403:

026764                             LET TSTCAS := TSTCAS + #6
026764  062767  000006  151552                                               ADD     #6,TSTCAS
026772                             LET COUNT := COUNT - #1
026772  005367  151472                                                       DEC     COUNT
                                ;UNTIL COUNT EQ #0
026776                             IF COUNT NE #0 THEN
026776  005767  151466                                                       TST     COUNT
027002  001402                                                               BEQ     $51150
027004                                 INLINE  <JMP  $5054>
027004  000167  176302                                                       JMP  $5054
027010                             ENDIF
027010                                                                  $51150:

                                ;+
                                ; DO ONE INLINE CASE - CVTPNI
                                ;-

027010                             CALL CLRREG
027010  004767  027424                                                       JSR     PC,CLRREG
027014                             CALL CLRDSB
027014  004767  027654                                                       JSR     PC,CLRDSB
027020                             INLINE  <CVTPNI>
027020  076154                                                               CVTPNI
027022                             INLINE  <.WORD  TCVPNI>
027022  027336                                                               .WORD  TCVPNI
027024                             INLINE  <.WORD  TCVPNI+4>
027024  027342                                                               .WORD  TCVPNI+4
027026                             LET CC := PSW
027026  016767  150744  151464                                               MOV     PSW,CC
027034                             LET CC := CC CLR.BY #177760
027034  042767  177760  151456                                               BIC     #177760,CC
027042                             IF CC NE #10 THEN
027042  026727  151452  000010                                               CMP     CC,#10
027050  001402                                                               BEQ     $51151
027052                                 CALL ERROR
027052  004767  030622                                                       JSR     PC,ERROR
027056                             ENDIF
027056                                                                  $51151:
027056                             IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN
027056  005700                                                               TST     R0
027060  001004                                                               BNE     $51152
027062  005701                                                               TST     R1
027064  001002                                                               BNE     $51152
027066  005702                                                               TST     R2
027070  001402                                                               BEQ     $51153
027072                                                                  $51152:
027072                                 CALL ERROR
027072  004767  030602                                                       JSR     PC,ERROR
027076                             ENDIF
027076                                                                  $51153:
```

```
5552 027076                             IF R3 NE #0 OR R4 NE #0 OR R5 NE #0 THEN
     027076  005703                                                    TST     R3
     027100  001004                                                    BNE     $51154
     027102  005704                                                    TST     R4
     027104  001002                                                    BNE     $51154
     027106  005705                                                    TST     R5
     027110  001402                                                    BEQ     $51155
     027112                                                 $51154:
5553 027112                                 CALL ERROR
     027112  004767  030562                                           JSR     PC,ERROR
5554 027116                             ENDIF
     027116                                                 $51155:
5555 027116                             IFB DS.DST NE #61 THEN
     027116  126727  035422  000061                                   CMPB    DS.DST,#61
     027124  001402                                                    BEQ     $51156
5556 027126                                 CALL ERROR
     027126  004767  030546                                           JSR     PC,ERROR
5557 027132                             ENDIF
     027132                                                 $51156:
5558 027132                             IFB DS.DST+1 NE #62 THEN
     027132  126727  035407  000062                                   CMPB    DS.DST+1,#62
     027140  001402                                                    BEQ     $51157
5559 027142                                 CALL ERROR
     027142  004767  030532                                           JSR     PC,ERROR
5560 027146                             ENDIF
     027146                                                 $51157:
5561 027146                             IFB DS.DST+2 NE #63 THEN
     027146  126727  035374  000063                                   CMPB    DS.DST+2,#63
     027154  001402                                                    BEQ     $51160
5562 027156                                 CALL ERROR
     027156  004767  030516                                           JSR     PC,ERROR
5563 027162                             ENDIF
     027162                                                 $51160:
5564 027162                             IFB DS.DST+3 NE #164 THEN
     027162  126727  035361  000164                                   CMPB    DS.DST+3,#164
     027170  001402                                                    BEQ     $51161
5565 027172                                 CALL ERROR
     027172  004767  030502                                           JSR     PC,ERROR
5566 027176                             ENDIF
     027176                                                 $51161:
5567 027176                             INLINE  <JMP  XCVTNP>
     027176  000167  000150                                           JMP  XCVTNP
5568
5569                                    ;+
5570                                    ; TABLE OF TEST CASES FOR CVTPN
5571                                    ;-
5572
5573 027202  000011                     TCVTPN: .WORD   9.              ;# OF TEST CASES FOR CVTPN
5574
5575 027204  060000                             .WORD   60000                   ;SRC DATA TYPE AND LENGTH
5576 027206  027272                             .WORD   CVTPN1                  ;POINTER TO PACKED SRC STRING
5577 027210  000000                             .WORD   0                       ;DST.LEN
5578
5579 027212  060000                             .WORD   60000           ;TEST CASE NUMBER 2
5580 027214  027273                             .WORD   CVTPN2
5581 027216  000005                             .WORD   5
5582
```

```
 5583 027220  060004                        .WORD   60004           ;TEST CASE NUMBER 3
 5584 027222  027274                        .WORD   CVTPN3
 5585 027224  000000                        .WORD   0
 5586
 5587 027226  060025                        .WORD   60025           ;TEST CASE NUMBER 4
 5588 027230  027277                        .WORD   CVTPN4
 5589 027232  000014                        .WORD   14
 5590
 5591 027234  060007                        .WORD   60007           ;TEST CASE NUMBER 5
 5592 027236  027312                        .WORD   CVTPN5
 5593 027240  000003                        .WORD   3
 5594
 5595 027242  070007                        .WORD   70007           ;TEST CASE NUMBER 6
 5596 027244  027316                        .WORD   CVTPN6
 5597 027246  000003                        .WORD   3
 5598
 5599 027250  060004                        .WORD   60004           ;TEST CASE NUMBER 7
 5600 027252  027322                        .WORD   CVTPN7
 5601 027254  000037                        .WORD   37
 5602
 5603 027256  060012                        .WORD   60012           ;TEST CASE NUMBER 8
 5604 027260  027325                        .WORD   CVTPN8
 5605 027262  000012                        .WORD   12
 5606
 5607 027264  060005                        .WORD   60005           ;TEST CASE NUMBER 9
 5608 027266  027333                        .WORD   CVTPN9
 5609 027270  000005                        .WORD   5
 5610
 5611 027272     014             CVTPN1: .BYTE   014                     ;PACKED DECIMAL STRING 1
 5612
 5613 027273     015             CVTPN2: .BYTE   015                     ;PACKED DECIMAL STRING 2
 5614
 5615 027274     001     043     114     CVTPN3: .BYTE   001,043,114             ;PACKED DECIMAL STRING 3
 5616
 5617 027277     020     043     127     CVTPN4: .BYTE   020,043,127,206,224,022,003,000,100,000,014     ;PACKED DECIMAL STRING 4
      027302     206     224     022
      027305     003     000     100
      027310     000     014
 5618
 5619 027312     022     000     007     CVTPN5: .BYTE   022,000,007,055         ;PACKED DECIMAL STRING 5
      027315     055
 5620
 5621 027316     022     000     007     CVTPN6: .BYTE   022,000,007,055         ;PACKED DECIMAL STRING 6
      027321     055
 5622
 5623 027322     001     043     114     CVTPN7: .BYTE   001,043,114             ;PACKED DECIMAL STRING 7
 5624
 5625 027325     002     024     164     CVTPN8: .BYTE   002,024,164,203,144,215         ;PACKED DECIMAL STRING 8
      027330     203     144     215
 5626
 5627 027333     061     040     174     CVTPN9: .BYTE   061,040,174             ;PACKED DECIMAL STRING 9
 5628
 5629                                    .EVEN
 5630
 5631 027336                             TCVPNI:                         ;INLINE TEST CASE
 5632 027336  060004                                .WORD   060004               ;SIGNED PACKED, LENGTH = 4
 5633 027340  027346                                .WORD   PN1                  ;POINTER TO PACKED SRC STRING
```

CVTPN

```
   5634 027342  000004                          .WORD   000004                  ;SIGNED ZONED, LENGTH = 4
   5635 027344  064544                          .WORD   DS.DST                  ;POINTER TO DST
   5636
   5637 027346     001     043     113  PN1:    .BYTE   1,43,113                ;-1234
   5638                                 .EVEN
   5639
```

```
                                .SBTTL  CVTNP
                                ;;*****************************************************************
                                ;
                                ;  CVTNP TEST
                                ;
                                ;;*****************************************************************
                                ROUTINE XCVTNP
027352                                                                          XCVTNP:
                                   INLINE  <SCOPE>
027352  000004                                                                    SCOPE
                                   INLINE  <NOP>
027354  000240                                                                    NOP
                                   LET $TIMES := #20                    ;ITERATION COUNT
027356  012767  000020  151020                                                   MOV     #20,$TIMES
                                   LET COUNT := TCVTNP                  ;GET TEST COUNT
027364  016767  001460  151076                                                   MOV     TCVTNP,COUNT
                                   LET TSTCAS := #TCVTNP+2                      ;POINT TO FIRST TEST CASE
027372  012767  031052  151144                                                   MOV     #TCVTNP+2,TSTCAS
                                ;REPEAT
                                   INLINE  <$5055:>
027400                                                                          $5055:

                                ;+
                                ; GET TEST DATA FROM TABLE
                                ;-

                                   LET R0 := TSTCAS
027400  016700  151140                                                          MOV     TSTCAS,R0
                                   LET DATA00 := (R0)+                  ;SRC DATA TYPE AND LENGTH
027404  012067  151152                                                          MOV     (R0)+,DATA00
                                   LET DATA01 := (R0)+                  ;POINTER TO INTERMEDIATE SRC STRING
027410  012067  151150                                                          MOV     (R0)+,DATA01
                                   LET DATA02 := (R0)                   ;DST DATA TYPE AND LENGTH
027414  011067  151146                                                          MOV     (R0),DATA02


                                ;+
                                ; INITIALIZE THE DECIMAL BUFFERS AND CC.EXP.
                                ;-

                                   CALL CLRDSB
027420  004767  027250                                                          JSR     PC,CLRDSB
                                   LET CC.EXP := #0
027424  005067  151072                                                          CLR     CC.EXP


                                ;+
                                ; LET SIGNO REFLECT THE SIGN OF THE SOURCE STRING.
                                ;-

                                   LET SIGNO := @DATA01
027430  017767  151130  151012                                                  MOV     @DATA01,SIGNO

                                ;+
                                ; PUT SRC MAGNITUDE INTO DSBUF1.  DSBUF1 WILL BE THE SOURCE FOR THE
                                ; CVTNP INSTRUCTION.
```

```
                                ;-
                                
                                   CALL CLRDSB                                                          JSR     PC,CLRDSB
                                   LET R0 := DATA01              ;GET ADDRESS OF INTERMEDIATE SRC. STRING
                                                                                                        MOV     DATA01,R0
                                   LET R0 := R0 + #2             ;GET PAST THE SIGN WORD
                                                                                                        ADD     #2,R0
                                   LET R1 := #DSBUF1
                                                                                                        MOV     #DSBUF1,R1
                                   LET DSLEN1 := DATA00 CLR.BY #70000   ;GET SRC.LEN
                                                                                                        MOV     DATA00,DSLEN1
                                                                                                        BIC     #70000,DSLEN1
                                   INCR R2 FROM #1 TO DSLEN1 BY #1      ;TRANSFER THE MAGNITUDE
                                                                                                        MOV     #1,R2
                                                                                                        BR      $51164
                                                                                                $51165:
                                                                                                        INC     R2
                                                                                                $51164:
                                                                                                        CMP     R2,DSLEN1
                                                                                                        BGT     $51166
                                       LET (R1)+ :B= (R0)+
                                                                                                        MOVB    (R0)+,(R1)+
                                   ENDINC
                                                                                                        BR      $51165
                                                                                                $51166:

                                ;+
                                ; PUT THE PROPER SIGN INTO DSBUF1, ACCORDING TO THE SOURCE DATA TYPE.
                                ;-

                                   LET DATTYP := DATA00 CLR.BY #107777  ;GET SRC DATA TYPE
                                                                                                        MOV     DATA00,DATTYP
                                                                                                        BIC     #107777,DATTYP

                                ;*** SIGNED ZONED ***

                                   IF DATTYP EQ #0 THEN
                                                                                                        TST     DATTYP
                                                                                                        BNE     $51167
                                       IF DSLEN1 HI #0 THEN          ;SRC.LEN > 0
                                                                                                        TST     DSLEN1
                                                                                                        BLOS    $51170
                                           LET R1 := R1 - #1                 ;POINT TO SIGN BYTE
                                                                                                        DEC     R1
                                           IF SIGN0 EQ #0 THEN               ;SIGN IS +
                                                                                                        TST     SIGN0
                                                                                                        BNE     $51171
                                               LET (R1) :B= (R1) SET.BY #60
                                                                                                        BISB    #60,(R1)
                                           ELSE                          ;SIGN IS -
                                                                                                        BR      $51172
                                                                                                $51171:
                                               LET (R1) :B= (R1) SET.BY #160
                                                                                                        BISB    #160,(R1)
```

```
 5712 027566                                ENDIF
      027566                                                                $51172:
 5713 027566                             ENDIF
      027566                                                                $51170:
 5714 027566                          ENDIF
      027566                                                                $51167:
 5715
 5716
 5717                                ;*** UNSIGNED ZONED ***
 5718
 5719 027566                          IF DATTYP EQ #10000 THEN
      027566  026727  150672  010000                                        CMP     DATTYP,#10000
      027574  001006                                                        BNE     $51173
 5720 027576                              IF DSLEN1 HI #0 THEN
      027576  005767  150712                                                TST     DSLEN1
      027602  101403                                                        BLOS    $51174
 5721 027604                                  LET R1 := R1 - #1                ;POINT TO SIGN BYTE
      027604  005301                                                        DEC     R1
 5722 027606                                  LET (R1) :B= (R1) SET.BY #60
      027606  152711  000060                                                BISB    #60,(R1)
 5723 027612                              ENDIF
      027612                                                                $51174:
 5724 027612                          ENDIF
      027612                                                                $51173:
 5725
 5726
 5727                                ;*** TRAILING OVERPUNCH ***
 5728
 5729 027612                          IF DATTYP EQ #20000 THEN
      027612  026727  150646  020000                                        CMP     DATTYP,#20000
      027620  001015                                                        BNE     $51175
 5730 027622                              IF DSLEN1 HI #0 THEN
      027622  005767  150666                                                TST     DSLEN1
      027626  101412                                                        BLOS    $51176
 5731 027630                                  LET R1 := R1 - #1                ;POINT TO SIGN BYTE
      027630  005301                                                        DEC     R1
 5732 027632                                  LET R0 := (R1)
      027632  011100                                                        MOV     (R1),R0
 5733 027634                                  IF SIGN0 EQ #-1 THEN
      027634  026727  150610  177777                                        CMP     SIGN0,#-1
      027642  001002                                                        BNE     $51177
 5734 027644                                      LET R0 := R0 * #10.
      027644  062700  000012                                                ADD     #10.,R0
 5735 027650                                  ENDIF
      027650                                                                $51177:
 5736 027650                                  LET (R1) :B= OVPNCH(R0)          ;PUT IN SIGN/DIGIT COMBINATION
      027650  116011  056330                                                MOVB    OVPNCH(R0),(R1)
 5737 027654                              ENDIF
      027654                                                                $51176:
 5738 027654                          ENDIF
      027654                                                                $51175:
 5739
 5740
 5741                                ;*** LEADING OVERPUNCH ***
 5742
 5743 027654                          IF DATTYP EQ #30000 THEN
      027654  026727  150604  030000                                        CMP     DATTYP,#30000
```

```
     027662  001016                                                                   BNE     $51200
5744 027664                              IF DSLEN1 HI #0 THEN
     027664  005767  150624                                                           TST     DSLEN1
     027670  101413                                                                   BLOS    $51201
5745 027672                                 LET R0 :B= DSBUF1
     027672  116700  034542                                                           MOVB    DSBUF1,R0
5746 027676                                 IF SIGNO EQ #-1 THEN
     027676  026727  150546  177777                                                   CMP     SIGNO,#-1
     027704  001002                                                                   BNE     $51202
5747 027706                                    LET R0 := R0 + #10.
     027706  062700  000012                                                           ADD     #10.,R0
5748 027712                                 ENDIF
     027712                                                                   $51202:
5749 027712                                 LET DSBUF1 :B= OVPNCH(R0)
     027712  116067  056330  034520                                                   MOVB    OVPNCH(R0),DSBUF1
5750 027720                              ENDIF
     027720                                                                   $51201:
5751 027720                           ENDIF
     027720                                                                   $51200:
5752
5753
5754                                 ;*** TRAILING SEPARATE ***
5755
5756 027720                           IF DATTYP EQ #40000 THEN
     027720  026727  150540  040000                                                   CMP     DATTYP,#40000
     027726  001010                                                                   BNE     $51203
5757 027730                              IF SIGNO EQ #0 THEN                   ;SIGN IS +
     027730  005767  150514                                                           TST     SIGNO
     027734  001003                                                                   BNE     $51204
5758 027736                                 LET (R1) :B= #53
     027736  112711  000053                                                           MOVB    #53,(R1)
5759 027742                              ELSE                          ;SIGN IS -
     027742  000402                                                                   BR      $51205
     027744                                                                   $51204:
5760 027744                                 LET (R1) :B= #55
     027744  112711  000055                                                           MOVB    #55,(R1)
5761 027750                              ENDIF
     027750                                                                   $51205:
5762 027750                           ENDIF
     027750                                                                   $51203:
5763
5764
5765                                 ;*** LEADING SEPARATE ***
5766
5767 027750                           IF DATTYP EQ #50000 THEN
     027750  026727  150510  050000                                                   CMP     DATTYP,#50000
     027756  001012                                                                   BNE     $51206
5768 027760                              IF SIGNO EQ #0 THEN                   ;SIGN IS +
     027760  005767  150464                                                           TST     SIGNO
     027764  001004                                                                   BNE     $51207
5769 027766                                 LET DSBUF1-1 :B= #53
     027766  112767  000053  034443                                                   MOVB    #53,DSBUF1-1
5770 027774                              ELSE                          ;SIGN IS -
     027774  000403                                                                   BR      $51210
     027776                                                                   $51207:
5771 027776                                 LET DSBUF1-1 :B= #55
     027776  112767  000055  034433                                                   MOVB    #55,DSBUF1-1
```

```
 5772 030004                                ENDIF
      030004                                                                                  $51210:
 5773 030004                              ENDIF
      030004                                                                                  $51206:
 5774
 5775
 5776                                ;+
 5777                                ; LOCATE THE END OF THE EXPECTED DESTINATION, WHERE THE SIGN
 5778                                ; GOES.  GET THE DESTINATION DATA TYPE.
 5779                                ;-
 5780
 5781 030004                              LET R2 := DATA02 CLR.BY #70000                ;GET DST.LEN
      030004  016702  150556                                                                          MOV     DATA02,R2
      030010  042702  070000                                                                          BIC     #70000,R2
 5782 030014                              INLINE   <ASR        R2>
      030014  006202                                                                                  ASR     R2
 5783 030016                              LET R0 := #DS.EXP + R2
      030016  012700  064606                                                                          MOV     #DS.EXP,R0
      030022  060200                                                                                  ADD     R2,R0
 5784 030024                              LET TEMP1 := DATA02 CLR.BY #107777
      030024  016767  150536  150446                                                                  MOV     DATA02,TEMP1
      030032  042767  107777  150440                                                                  BIC     #107777,TEMP1
 5785
 5786
 5787                                ;+
 5788                                ; PUT THE SIGN INTO THE EXPECTED DESTINATION.
 5789                                ;-
 5790
 5791 030040                              LET DSLEN1 := DATA00 CLR.BY #70000            ;GET SRC.LEN
      030040  016767  150516  150446                                                                  MOV     DATA00,DSLEN1
      030046  042767  070000  150440                                                                  BIC     #70000,DSLEN1
 5792 030054                              LET DSLEN2 := DATA02 CLR.BY #70000            ;GET DST.LEN
      030054  016767  150506  150434                                                                  MOV     DATA02,DSLEN2
      030062  042767  070000  150426                                                                  BIC     #70000,DSLEN2
 5793 030070                              IF DSLEN1 EQ #0 THEN          ;SRC.LEN OR DST.LEN = 0
      030070  005767  150420                                                                          TST     DSLEN1
      030074  001003                                                                                 BNE     $51211
 5794 030076                                 LET (R0) :B= #14
      030076  112710  000014                                                                          MOVB    #14,(R0)
 5795 030102                              ELSE                                          ;SRC.LEN & DST.LEN > 0
      030102  000414                                                                                 BR      $51212
      030104                                                                                  $51211:
 5796 030104                                 IF @DATA01 EQ #0 OR DATTYP EQ #10000 THEN
      030104  005777  150454                                                                          TST     @DATA01
      030110  001404                                                                                 BEQ     $51213
      030112  026727  150346  010000                                                                  CMP     DATTYP,#10000
      030120  001003                                                                                 BNE     $51214
      030122                                                                                  $51213:
 5797 030122                                    LET (R0) :B= #14                             ;SIGN IS +
      030122  112710  000014                                                                          MOVB    #14,(R0)
 5798 030126                                 ELSE
      030126  000402                                                                                 BR      $51215
      030130                                                                                  $51214:
 5799 030130                                    LET (R0) :B= #15                             ;SIGN IS -
      030130  112710  000015                                                                          MOVB    #15,(R0)
 5800 030134                                 ENDIF
      030134                                                                                  $51215:
```

```
 5801 030134                            ENDIF
      030134                                                                $51212:
 5802 030134                            IF TEMP1 EQ #70000 THEN
      030134  026727  150340  070000                                                 CMP     TEMP1,#70000
      030142  001002                                                               BNE     $51216
 5803 030144                                LET (R0) :B= #17
      030144  112710  000017                                                       MOVB    #17,(R0)
 5804 030150                            ENDIF
      030150                                                                $51216:
 5805
 5806
 5807                                ;+
 5808                                ; PUT THE EXPECTED VALUE OF THE NBIT INTO CC.EXP
 5809                                ;-
 5810
 5811 030150                            IF DATTYP NE #10000 AND TEMP1 NE #70000 THEN
      030150  026727  150310  010000                                                 CMP     DATTYP,#10000
      030156  001415                                                               BEQ     $51217
      030160  026727  150314  070000                                                 CMP     TEMP1,#70000
      030166  001411                                                               BEQ     $51217
 5812 030170                                IFB (R0) EQ #15 AND DSLEN2 GT #0 THEN    ;SIGN NIBBLE IN DS.EXP IS NEG.
      030170  121027  000015                                                       CMPB    (R0),#15
      030174  001006                                                               BNE     $51220
      030176  005767  150314                                                       TST     DSLEN2
      030202  003403                                                               BLE     $51220
 5813 030204                                    LET CC.EXP := CC.EXP SET.BY #NBIT
      030204  052767  000010  150310                                               BIS     #NBIT,CC.EXP
 5814 030212                                ENDIF
      030212                                                                $51220:
 5815 030212                            ENDIF
      030212                                                                $51217:
 5816
 5817
 5818                                ;+
 5819                                ; PUT THE EXPECTED VALUE OF THE ZBIT INTO CC.EXP
 5820                                ;-
 5821
 5822 030212                            IF DSLEN1 EQ #0 OR DSLEN2 EQ #0 THEN         ;SRC.LEN OR DST.LEN = 0
      030212  005767  150276                                                       TST     DSLEN1
      030216  001403                                                               BEQ     $51221
      030220  005767  150272                                                       TST     DSLEN2
      030224  001003                                                               BNE     $51222
      030226                                                                $51221:
 5823 030226                                LET CC.EXP := CC.EXP SET.BY #ZBIT
      030226  052767  000004  150266                                               BIS     #ZBIT,CC.EXP
 5824 030234                            ENDIF
      030234                                                                $51222:
 5825
 5826
 5827                                ;+
 5828                                ; PUT THE EXPECTED VALUE OF THE VBIT INTO CC.EXP
 5829                                ;-
 5830
 5831 030234                            IF DSLEN1 HI DSLEN2 THEN             ;SRC.LEN > DST.LEN
      030234  026767  150254  150254                                                 CMP     DSLEN1,DSLEN2
      030242  101403                                                               BLOS    $51223
 5832 030244                                LET CC.EXP := CC.EXP SET.BY #VBIT
```

```
      030244  052767  000002  150250                                                          BIS     #VBIT,CC.EXP
 5833 030252                          ENDIF
      030252                                                                                 $51223:
 5834
 5835
 5836                                 ;+
 5837                                 ; CALCULATE THE NUMBER OF CHARACTERS TO BE MOVED INTO DS.EXP;
 5838                                 ; USE R0.
 5839                                 ;-
 5840
 5841 030252                             IF DSLEN2 HIS DSLEN1 THEN           ;DST.LEN >= SRC.LEN
      030252  026767  150240  150234                                                          CMP     DSLEN2,DSLEN1
      030260  103403                                                                         BLO     $51224
 5842 030262                                 LET R0 := DSLEN1
      030262  016700  150226                                                                 MOV     DSLEN1,R0
 5843 030266                             ELSE                                ;SRC.LEN > DST.LEN
      030266  000402                                                                         BR      $51225
      030270                                                                             $51224:
 5844 030270                                 LET R0 := DSLEN2
      030270  016700  150222                                                                 MOV     DSLEN2,R0
 5845 030274                             ENDIF
      030274                                                                             $51225:
 5846
 5847
 5848                                 ;+
 5849                                 ; MAKE R1 POINT TO THE END OF THE DESTINATION IN DS.EXP; MAKE
 5850                                 ; R2 POINT TO THE END OF THE SOURCE IN THE TABLE OF TEST CASES.
 5851                                 ;-
 5852
 5853 030274                             LET R1 := DSLEN2
      030274  016701  150216                                                                 MOV     DSLEN2,R1
 5854 030300                             INLINE   <ASR        R1>
      030300  006201                                                                         ASR     R1
 5855 030302                             LET R1 := R1 + #DS.EXP
      030302  062701  064606                                                                 ADD     #DS.EXP,R1
 5856 030306                             LET R2 := DATA01 + DSLEN1 + #1
      030306  016702  150252                                                                 MOV     DATA01,R2
      030312  066702  150176                                                                 ADD     DSLEN1,R2
      030316  005202                                                                         INC     R2
 5857
 5858
 5859                                 ;+
 5860                                 ; MOVE THE LEAST SIGNIFICANT BCD FROM THE SOURCE TO THE
 5861                                 ; EXPECTED DESTINATION BUFFER, IN PACKED FORMAT.
 5862                                 ;-
 5863
 5864 030320                             IF R0 GT #0 THEN
      030320  005700                                                                         TST     R0
      030322  003405                                                                         BLE     $51226
 5865 030324                                 LET R3 :B= (R2)
      030324  111203                                                                         MOVB    (R2),R3
 5866 030326                                 INLINE   <ASH    #4,R3>
      030326  072327  000004                                                                 ASH     #4,R3
 5867 030332                                 LET (R1) :B= (R1) SET.BY R3
      030332  150311                                                                         BISB    R3,(R1)
 5868 030334                                 LET R0 := R0 - #1                   ;DECREMENT COUNT
      030334  005300                                                                         DEC     R0
```

```
 5869 030336                            ENDIF
      030336                                                                $51226:
 5870
 5871
 5872                                ;+
 5873                                ; MOVE THE REST OF THE SOURCE TO THE EXPECTED DESTINATION
 5874                                ; BUFFER, IN PACKED FORMAT, USING R0 AS A COUNTER FOR THE NUMBER
 5875                                ; OF CHARACTERS TO MOVE.
 5876                                ;-
 5877
 5878 030336                            WHILE R0 GT #0 DO
      030336                                                                $51227:
      030336  005700                                                            TST     R0
      030340  003417                                                            BLE     $51230
 5879
 5880 030342                                LET R1 := R1 - #1               ;DECREMENT DEST. POINTER
      030342  005301                                                            DEC     R1
 5881 030344                                LET R2 := R2 - #1               ;DECREMENT SRC. POINTER
      030344  005302                                                            DEC     R2
 5882 030346                                LET (R1) :B= (R2) CLR.BY #360   ;PUT BCD INTO LOWER NIBBLE
      030346  111211                                                            MOVB    (R2),(R1)
      030350  142711  000360                                                    BICB    #360,(R1)
 5883 030354                                LET R0 := R0 - #1               ;DECREMENT COUNT
      030354  005300                                                            DEC     R0
 5884 030356                                IF R0 HI #0 THEN
      030356  005700                                                            TST     R0
      030360  101406                                                            BLOS    $51231
 5885 030362                                    LET R2 := R2 - #1           ;DECREMENT SRC. POINTER
      030362  005302                                                            DEC     R2
 5886 030364                                    LET R3 :B= (R2)
      030364  111203                                                            MOVB    (R2),R3
 5887 030366                                    INLINE   <ASH        #4,R3>
      030366  072327  000004                                                    ASH     #4,R3
 5888 030372                                    LET (R1) :B= (R1) SET.BY R3 ;PUT BCD INTO UPPER NIBBLE
      030372  150311                                                            BISB    R3,(R1)
 5889 030374                                    LET R0 := R0 - #1           ;DECREMENT COUNT
      030374  005300                                                            DEC     R0
 5890 030376                                ENDIF
      030376                                                                $51231:
 5891 030376                            ENDDO
      030376  000757                                                            BR      $51227
      030400                                                                $51230:
 5892
 5893
 5894                                ;+
 5895                                ; SET UP R0-R5 AND R0.EXP-R5.EXP; THEN DO THE CVTNP INSTRUCTION.
 5896                                ;-
 5897
 5898 030400                            LET R0 := DATA00                    ;SRC DATA TYPE AND LENGTH
      030400  016700  150156                                                    MOV     DATA00,R0
 5899 030404                            LET R1 := #DSBUF1                   ;SRC ADDRESS
      030404  012701  064440                                                    MOV     #DSBUF1,R1
 5900 030410                            LET R2 := DATA02                    ;DST DATA TYPE AND LENGTH
      030410  016702  150152                                                    MOV     DATA02,R2
 5901 030414                            LET R3 := #DS.DST                   ;DST ADDRESS
      030414  012703  064544                                                    MOV     #DS.DST,R3
 5902 030420                            LET R4 := #0
```

```
                030420  005004
 5903           030422
                030422  012705  177777                    LET R5 := #-1
 5904           030426
                030426  005067  150072                    LET R0.EXP := #0
 5905           030432
                030432  005067  150070                    LET R1.EXP := #0
 5906           030436
                030436  010267  150066                    LET R2.EXP := R2
 5907           030442
                030442  010367  150064                    LET R3.EXP := R3
 5908           030446
                030446  005067  150062                    LET R4.EXP := #0
 5909           030452
                030452  012767  177777  150056            LET R5.EXP := #-1
 5910           030460
                030460  076055                            INLINE  <CVTNP>
 5911
 5912                                            ;+
 5913                                            ;  CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.
 5914                                            ;-
 5915
 5916           030462
                030462  016767  147310  150030            LET CC := PSW
 5917           030470
                030470  042767  177760  150022            LET CC := CC CLR.BY #177760
 5918           030476
                030476  026767  150016  150016            IF CC NE CC.EXP THEN
                030504  001402
 5919           030506
                030506  004767  027166                        CALL ERROR
 5920           030512                                   ENDIF
                030512
 5921           030512                                   IF R0 NE R0.EXP THEN
                030512  020067  150006
                030516  001402
 5922           030520
                030520  004767  027154                        CALL ERROR
 5923           030524                                   ENDIF
                030524
 5924           030524                                   IF R1 NE R1.EXP THEN
                030524  020167  147776
                030530  001402
 5925           030532
                030532  004767  027142                        CALL ERROR
 5926           030536                                   ENDIF
                030536
 5927           030536                                   IF R2 NE R2.EXP THEN
                030536  020267  147766
                030542  001402                           >
 5928           030544
                030544  004767  027130                        CALL ERROR
 5929           030550                                   ENDIF
                030550
 5930           030550                                   IF R3 NE R3.EXP THEN
                030550  020367  147756
                030554  001402
```

```
                CLR     R4
                MOV     #-1,R5
                CLR     R0.EXP
                CLR     R1.EXP
                MOV     R2,R2.EXP
                MOV     R3,R3.EXP
                CLR     R4.EXP
                MOV     #-1,R5.EXP
                CVTNP

                MOV     PSW,CC
                BIC     #177760,CC
                CMP     CC,CC.EXP
                BEQ     $51232
                JSR     PC,ERROR
$51232:
                CMP     R0,R0.EXP
                BEQ     $51233
                JSR     PC,ERROR
$51233:
                CMP     R1,R1.EXP
                BEQ     $51234
                JSR     PC,ERROR
$51234:
                CMP     R2,R2.EXP
                BEQ     $51235
                JSR     PC,ERROR
$51235:
                CMP     R3,R3.EXP
                BEQ     $51236
```

```
 5931 030556                            CALL ERROR
      030556  004767  027116                                        JSR     PC,ERROR
 5932 030562                          ENDIF
      030562                                                    $51236:
 5933 030562                          IF R4 NE R4.EXP THEN
      030562  020467  147746                                        CMP     R4,R4.EXP
      030566  001402                                               BEQ     $51237
 5934 030570                            CALL ERROR
      030570  004767  027104                                        JSR     PC,ERROR
 5935 030574                          ENDIF
      030574                                                    $51237:
 5936 030574                          IF R5 NE R5.EXP THEN
      030574  020567  147736                                        CMP     R5,R5.EXP
      030600  001402                                               BEQ     $51240
 5937 030602                            CALL ERROR
      030602  004767  027072                                        JSR     PC,ERROR
 5938 030606                          ENDIF
      030606                                                    $51240:
 5939
 5940                               ;COMPARE DS.DST WITH DS.EXP
 5941
 5942 030606                            LET ERRFLG := #0
      030606  005067  147644                                        CLR     ERRFLG
 5943 030612                            LET R0 := #DS.DST
      030612  012700  064544                                        MOV     #DS.DST,R0
 5944 030616                            LET R1 := #DS.EXP
      030616  012701  064606                                        MOV     #DS.EXP,R1
 5945 030622                            WHILE ERRFLG EQ #0 AND R0 LO #DS.DST+40 DO
      030622                                                    $51241:
      030622  005767  147630                                        TST     ERRFLG
      030626  001010                                               BNE     $51242
      030630  020027  064604                                        CMP     R0,#DS.DST+40
      030634  103005                                               BHIS    $51242
 5946 030636                              IFB (R0)+ NE (R1)+ THEN
      030636  122021                                               CMPB    (R0)+,(R1)+
      030640  001402                                               BEQ     $51243
 5947 030642                                CALL ERROR
      030642  004767  027032                                        JSR     PC,ERROR
 5948 030646                              ENDIF
      030646                                                    $51243:
 5949 030646                            ENDDO
      030646  000765                                               BR      $51241
      030650                                                    $51242:
 5950 030650                            LET TSTCAS := TSTCAS + #6
      030650  062767  000006  147666                                ADD     #6,TSTCAS
 5951 030656                            LET COUNT := COUNT - #1
      030656  005367  147606                                        DEC     COUNT
 5952                               ;UNTIL COUNT EQ #0
 5953 030662                            IF COUNT NE #0 THEN
      030662  005767  147602                                        TST     COUNT
      030666  001402                                               BEQ     $51244
 5954 030670                              INLINE  <JMP  $5055>
      030670  000167  176504                                        JMP  $5055
 5955 030674                            ENDIF
      030674                                                    $51244:
 5956
 5957                               ;+
```

```
                                        ; DO ONE INLINE CASE - CVTNPI
                                        ;-
                                           CALL CLRREG                          JSR     PC,CLRREG
                                           CALL CLRDSB                          JSR     PC,CLRDSB
                                           INLINE  <CVTNPI>                     CVTNPI
                                           INLINE  <.WORD  TCVNPI>              .WORD   TCVNPI
                                           INLINE  <.WORD  TCVNPI+4>            .WORD   TCVNPI+4
                                           LET CC := PSW                        MOV     PSW,CC
                                           LET CC := CC CLR.BY #177760          BIC     #177760,CC
                                           IF CC NE #0 THEN                     TST     CC
                                                                                BEQ     $51245
                                               CALL ERROR                       JSR     PC,ERROR
                                           ENDIF                            $51245:
                                           IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN
                                                                                TST     R0
                                                                                BNE     $51246
                                                                                TST     R1
                                                                                BNE     $51246
                                                                                TST     R2
                                                                                BEQ     $51247
                                                                            $51246:
                                               CALL ERROR                       JSR     PC,ERROR
                                           ENDIF                            $51247:
                                           IF R3 NE #0 OR R4 NE #0 OR R5 NE #0 THEN
                                                                                TST     R3
                                                                                BNE     $51250
                                                                                TST     R4
                                                                                BNE     $51250
                                                                                TST     R5
                                                                                BEQ     $51251
                                                                            $51250:
                                               CALL ERROR                       JSR     PC,ERROR
                                           ENDIF                            $51251:
                                           IFB DS.DST NE #124 THEN
                                                                                CMPB    DS.DST,#124
                                                                                BEQ     $51252
                                               CALL ERROR                       JSR     PC,ERROR
                                           ENDIF                            $51252:
                                           IFB DS.DST+1 NE #62 THEN
                                                                                CMPB    DS.DST+1,#62
```

```
                031022  001402                                                          BEQ     $51253
5981            031024                          CALL ERROR
                031024  004767  026650                                                  JSR     PC,ERROR
5982            031030                      ENDIF
                031030                                                          $51253:
5983            031030                      IFB DS.DST+2 NE #37 THEN
                031030  126727  033512  000037                                          CMPB    DS.DST+2,#37
                031036  001402                                                          BEQ     $51254
5984            031040                          CALL ERROR
                031040  004767  026634                                                  JSR     PC,ERROR
5985            031044                      ENDIF
                031044                                                          $51254:
5986            031044                      INLINE  <JMP  CIS5END>
                031044  000167  001552                                                  JMP  CIS5END
5987
5988                                    ;+
5989                                    ; TABLE OF TEST CASES FOR CVTNP
5990                                    ;-
5991
5992 031050  000010                    TCVTNP: .WORD   8.              ;# OF TEST CASES FOR CVTNP
5993
5994 031052  000000                            .WORD   0                       ;SRC DATA TYPE AND LENGTH
5995 031054  031140                            .WORD   CVTNP1                  ;POINTER TO INTERMEDIATE SRC STRING
5996 031056  060000                            .WORD   60000                   ;DST DATA TYPE AND LENGTH
5997
5998 031060  050000                            .WORD   50000           ;TEST CASE NUMBER 2
5999 031062  031144                            .WORD   CVTNP2
6000 031064  060000                            .WORD   60000
6001
6002 031066  030000                            .WORD   30000           ;TEST CASE NUMBER 3
6003 031070  031150                            .WORD   CVTNP3
6004 031072  060037                            .WORD   60037
6005
6006 031074  020023                            .WORD   20023           ;TEST CASE NUMBER 4
6007 031076  031154                            .WORD   CVTNP4
6008 031100  060000                            .WORD   60000
6009
6010 031102  040010                            .WORD   40010           ;TEST CASE NUMBER 5
6011 031104  031202                            .WORD   CVTNP5
6012 031106  060020                            .WORD   60020
6013
6014 031110  010011                            .WORD   10011           ;TEST CASE NUMBER 6
6015 031112  031214                            .WORD   CVTNP6
6016 031114  060006                            .WORD   60006
6017
6018 031116  000017                            .WORD   17              ;TEST CASE NUMBER 7
6019 031120  031230                            .WORD   CVTNP7
6020 031122  060017                            .WORD   60017
6021
6022 031124  000006                            .WORD   6               ;TEST CASE NUMBER 8
6023 031126  031252                            .WORD   CVTNP8
6024 031130  070006                            .WORD   70006
6025
6026 031132  050000                            .WORD   50000           ;TEST CASE NUMBER 9
6027 031134  031262                            .WORD   CVTNP9
6028 031136  060002                            .WORD   60002
6029
```

```
                                CVTNP1: .WORD   0                       ;SIGN WORD: 0 = +, -1 = -
                                        .BYTE                           ;MAGNITUDE, ONE BCD PER BYTE
                                        .EVEN

                                CVTNP2: .WORD   -1              ;INTERMEDIATE SRC STRING 2
                                        .BYTE
                                        .EVEN

                                CVTNP3: .WORD   -1              ;INTERMEDIATE SRC STRING 3
                                        .BYTE
                                        .EVEN

                                CVTNP4: .WORD   -1              ;INTERMEDIATE SRC STRING 4
031156  011     010     007             .BYTE   011,010,007,006,005,004,003,002,001,000,001,002,003,004,005,006
031161  006     005     004
031164  003     002     001
031167  000     001     002
031172  003     004     005
031175  006
031176  007     010     011             .BYTE   007,010,011
                                        .EVEN

031202  000000                  CVTNP5: .WORD   0               ;INTERMEDIATE SRC STRING 5
031204  007     000     006             .BYTE   007,000,006,001,005,002,004,003
031207  001     005     002
031212  004     003
                                        .EVEN

031214  177777                  CVTNP6: .WORD   -1              ;INTERMEDIATE SRC STRING 6
031216  001     003     005             .BYTE   001,003,005,007,011,002,004,006,010
031221  007     011     002
031224  004     006     010
                                        .EVEN

031230  177777                  CVTNP7: .WORD   -1              ;INTERMEDIATE SRC STRING 7
031232  005     005     007             .BYTE   005,005,007,003,010,002,001,011,004,007,006,000,000,007,001
031235  003     010     002
031240  001     011     004
031243  007     006     000
031246  000     007     001
                                        .EVEN

031252  177777                  CVTNP8: .WORD   -1              ;INTERMEDIATE SRC STRING 8
031254  007     005     003             .BYTE   007,005,003,001,004,002
031257  001     004     002
                                        .EVEN

031262  000000                  CVTNP9: .WORD   0               ;INTERMEDIATE SRC STRING 9
031264  007     002                     .BYTE   007,002
                                        .EVEN


031266                          TCVNPI:                         ;INLINE TEST CASE
031266  000005                          .WORD   000005                  ;SIGNED ZONED, LENGTH = 5
031270  031276                          .WORD   NP1                     ;POINTER TO SRC
031272  070005                          .WORD   070005                  ;UNSIGNED PACKED, LENGTH = 5
```

```
 6073 031274  064544                          .WORD   DS.DST                  ;POINTER TO DST
 6074
 6075 031276     005     004     003  NP1:    .BYTE   5,4,3,2,161             ;-54321
      031301     002     161
 6076                                         .EVEN
 6077
```

```
 6079                                ;;*****************************************************************
 6080                                .SBTTL  CONVERT LONG INTEGER TO INTERMEDIATE FORM ROUTINE
 6081                                ;;*****************************************************************
 6082                                ;* THIS ROUTINE CONVERTS A LONG INTEGER INTO
 6083                                ;* AN INTERMEDIATE FORM.
 6084                                ;* THE INPUT IS THE 31 BIT LONG INTEGER (PLUS THE
 6085                                ;* SIGN BIT) IN REGISTERS R2 AND R3:
 6086                                ;*
 6087                                ;*                 ---------------------------------------------
 6088                                ;*         R2      ! SIGN !                HIGH                !
 6089                                ;*                 ---------------------------------------------
 6090                                ;*
 6091                                ;*                 ---------------------------------------------
 6092                                ;*         R3      !                       LOW                 !
 6093                                ;*                 ---------------------------------------------
 6094                                ;*
 6095                                ;*
 6096                                ;* THE OUTPUT IS:
 6097                                ;* SIGNO SET TO SIGN OF THE LONG INTEGER
 6098                                ;* (0 FOR +, 177777 FOR -); AND 10 BINARY
 6099                                ;* CODED DECIMAL (BCD) DIGITS PUT INTO MEMORY,
 6100                                ;* ONE DIGIT PER BYTE, STARTING AT DSBUF1,
 6101                                ;* MOST SIGNIFICANT DIGIT FIRST.  DSLEN1 WILL
 6102                                ;* HAVE THE LENGTH (EXCLUDING LEADING ZEROES).
 6103                                ;* THE GENERAL PURPOSE REGISTERS ARE SAVED
 6104                                ;* AT THE BEGINNING OF THE ROUTINE AND RESTORED
 6105                                ;* AT THE END.
 6106                                ;* DSBUF1 DECIMAL STRING BUFFER SHOULD BE CLEAR BEFORE
 6107                                ;* ENTERING THIS ROUTINE.
 6108
 6109 031304                        ROUTINE CVTLI
      031304                                                                          CVTLI:
 6110
 6111                                ;+
 6112                                ; ROUTINE INITIALIZATION
 6113                                ;-
 6114
 6115 031304                            CALL SAVREG                 ;SAVE THE GPR'S
      031304  004767  025426                                                         JSR     PC,SAVREG
 6116 031310                            LET SIGNO := #0
      031310  005067  147134                                                         CLR     SIGNO
 6117 031314                            LET DSLEN1 := #0
      031314  005067  147174                                                         CLR     DSLEN1
 6118
 6119                                ;+
 6120                                ; TEST FOR SPECIAL CASES OF 0 VALUE OR MOST NEGATIVE VALUE
 6121                                ;-
 6122
 6123 031320                            INLINE <TST   R2>
      031320  005702                                                                TST   R2
 6124 031322                            IFCOND MI THEN
      031322  100003                                                                BPL     $51257
 6125 031324                                LET SIGNO := #-1
      031324  012767  177777  147116                                                MOV     #-1,SIGNO
 6126 031332                            ENDIF
      031332                                                                     $51257:
 6127 031332                            LET TEMPO := R2 CLR.BY #100000
```

CONVERT LONG INTEGER TO INTERMEDIATE FORM ROUTINE

```
        031332  010267  147140                                                  MOV     R2,TEMPO
        031336  042767  100000  147132                                          BIC     #100000,TEMPO
  6128  031344                          IF R3 EQ #0 AND TEMPO EQ #0 THEN
        031344  005703                                                          TST     R3
        031346  001044                                                          BNE     $51260
        031350  005767  147122                                                  TST     TEMPO
        031354  001041                                                          BNE     $51260
  6129  031356                              IF SIGNO EQ #0 THEN
        031356  005767  147066                                                  TST     SIGNO
        031362  001003                                                          BNE     $51261
  6130                                  ;RETURN
  6131  031364                                  INLINE   <JMP  XCVTLI>
        031364  000167  000616                                                  JMP  XCVTLI
  6132  031370                              ELSE                        ;RESULT = -2,147,483,648
        031370  000433                                                          BR      $51262
        031372                                                          $51261:
  6133  031372                                  LET R0 := #DSBUF1
        031372  012700  064440                                                  MOV     #DSBUF1,R0
  6134  031376                                  LET (R0)+ :B= #2
        031376  112720  000002                                                  MOVB    #2,(R0)+
  6135  031402                                  LET (R0)+ :B= #1
        031402  112720  000001                                                  MOVB    #1,(R0)+
  6136  031406                                  LET (R0)+ :B= #4
        031406  112720  000004                                                  MOVB    #4,(R0)+
  6137  031412                                  LET (R0)+ :B= #7
        031412  112720  000007                                                  MOVB    #7,(R0)+
  6138  031416                                  LET (R0)+ :B= #4
        031416  112720  000004                                                  MOVB    #4,(R0)+
  6139  031422                                  LET (R0)+ :B= #8.
        031422  112720  000010                                                  MOVB    #8.,(R0)+
  6140  031426                                  LET (R0)+ :B= #3
        031426  112720  000003                                                  MOVB    #3,(R0)+
  6141  031432                                  LET (R0)+ :B= #6
        031432  112720  000006                                                  MOVB    #6,(R0)+
  6142  031436                                  LET (R0)+ :B= #4
        031436  112720  000004                                                  MOVB    #4,(R0)+
  6143  031442                                  LET (R0)+ :B= #8.
        031442  112720  000010                                                  MOVB    #8.,(R0)+
  6144  031446                                  LET DSLEN1 := #10.
        031446  012767  000012  147040                                          MOV     #10.,DSLEN1
  6145                                  ;RETURN
  6146  031454                                  INLINE   <JMP  XCVTLI>
        031454  000167  000526                                                  JMP  XCVTLI
  6147  031460                              ENDIF
        031460                                                          $51262:
  6148  031460                          ENDIF
        031460                                                          $51260:
  6149
  6150                                  ;+
  6151                                  ; LONG INTEGER SOURCE WAS NOT ZERO OR MOST NEGATIVE NUMBER,
  6152                                  ; SO PREPARE FOR CONVERSION ALGORITHM.
  6153                                  ;-
  6154
  6155  031460                          IF SIGNO EQ #-1 THEN
        031460  026727  146764  177777                                          CMP     SIGNO,#-1
        031466  001005                                                          BNE     $51263
  6156  031470                              LET R3 := NEGATE R3
```

```
031470  005403                                                              NEG     R3
031472                            LET R2 := COMP R2
031472  005102                                                              COM     R2
031474                            IF R3 EQ #0 THEN
031474  005703                                                              TST     R3
031476  001001                                                              BNE     $51264
031500                                LET R2 := R2 + #1
031500  005202                                                              INC     R2
031502                            ENDIF
031502                                                          $51264:
031502                        ENDIF
031502                                                          $51263:
031502                        LET R0 := #DSBUF1
031502  012700  064440                                                      MOV     #DSBUF1,R0

                    ;+
                    ; GET TENTH BCD DIGIT (MOST SIGNIFICANT DIGIT)
                    ;-

031506                        WHILE R2 HI #35632 DO
031506                                                          $51265:
031506  020227  035632                                                      CMP     R2,#35632
031512  101407                                                              BLOS    $51266
031514                            LET R3 := R3 - #145000
031514  162703  145000                                                      SUB     #145000,R3
031520                            INLINE <SBC R2>
031520  005602                                                              SBC R2
031522                            LET R2 := R2 - #35632
031522  162702  035632                                                      SUB     #35632,R2
031526                            LET (R0) :B= (R0) + #1
031526  105210                                                              INCB    (R0)
031530                        ENDDO
031530  000766                                                              BR      $51265
031532                                                          $51266:
031532                        IF R2 EQ #35632 AND R3 HIS #145000 THEN
031532  020227  035632                                                      CMP     R2,#35632
031536  001007                                                              BNE     $51267
031540  020327  145000                                                      CMP     R3,#145000
031544  103404                                                              BLO     $51267
031546                            LET R3 := R3 - #145000
031546  162703  145000                                                      SUB     #145000,R3
031552                            LET R2 := #0
031552  005002                                                              CLR     R2
031554                            LET (R0) :B= (R0) + #1
031554  105210                                                              INCB    (R0)
031556                        ENDIF
031556                                                          $51267:

                    ;+
                    ; GET NINTH BCD DIGIT
                    ;-

031556                        LET R0 := R0 + #1
031556  005200                                                              INC     R0
031560                        WHILE R2 HI #2765 DO
031560                                                          $51270:
031560  020227  002765                                                      CMP     R2,#2765
```

```
     031564  101407                                                          BLOS    $51271
     031566                           LET R3 := R3 - #160400
     031566  162703  160400                                                  SUB     #160400,R3
     031572                           INLINE <SBC R2>
     031572  005602                                                          SBC R2
     031574                           LET R2 := R2 - #2765
     031574  162702  002765                                                  SUB     #2765,R2
     031600                           LET (R0) :B= (R0) + #1
     031600  105210                                                          INCB    (R0)
     031602                        ENDDO
     031602  000766                                                          BR      $51270
     031604                                                          $51271:
     031604                        IF R2 EQ #2765 AND R3 HIS #160400 THEN
     031604  020227  002765                                                  CMP     R2,#2765
     031610  001007                                                          BNE     $51272
     031612  020327  160400                                                  CMP     R3,#160400
     031616  103404                                                          BLO     $51272
     031620                           LET R3 := R3 - #160400
     031620  162703  160400                                                  SUB     #160400,R3
     031624                           LET R2 := #0
     031624  005002                                                          CLR     R2
     031626                           LET (R0) :B= (R0) + #1
     031626  105210                                                          INCB    (R0)
     031630                        ENDIF
     031630                                                          $51272:

                               ;+
                               ;    GET EIGHTH BCD DIGIT
                               ;-

     031630                        LET R0 := R0 + #1
     031630  005200                                                          INC     R0
     031632                        WHILE R2 HI #230 DO
     031632                                                          $51273:
     031632  020227  000230                                                  CMP     R2,#230
     031636  101407                                                          BLOS    $51274
     031640                           LET R3 := R3 - #113200
     031640  162703  113200                                                  SUB     #113200,R3
     031644                           INLINE <SBC R2>
     031644  005602                                                          SBC R2
     031646                           LET R2 := R2 - #230
     031646  162702  000230                                                  SUB     #230,R2
     031652                           LET (R0) :B= (R0) + #1
     031652  105210                                                          INCB    (R0)
     031654                        ENDDO
     031654  000766                                                          BR      $51273
     031656                                                          $51274:
     031656                        IF R2 EQ #230 AND R3 HIS #113200 THEN
     031656  020227  000230                                                  CMP     R2,#230
     031662  001007                                                          BNE     $51275
     031664  020327  113200                                                  CMP     R3,#113200
     031670  103404                                                          BLO     $51275
     031672                           LET R3 := R3 - #113200
     031672  162703  113200                                                  SUB     #113200,R3
     031676                           LET R2 := #0
     031676  005002                                                          CLR     R2
     031700                           LET (R0) :B= (R0) + #1
```

```
        031700  105210
 6212   031702                          ENDIF                                                   INCB    (R0)
        031702                                                                        $51275:
 6213
 6214                                   ;+
 6215                                   ;   GET SEVENTH BCD DIGIT
 6216                                   ;-
 6217
 6218   031702                          LET R0 := R0 + #1
        031702  005200                                                                         INC     R0
 6219   031704                          WHILE R2 HI #17 DO
        031704                                                                        $51276:
        031704  020227  000017                                                                  CMP     R2,#17
        031710  101407                                                                         BLOS    $51277
 6220   031712                              LET R3 := R3 - #41100
        031712  162703  041100                                                                 SUB     #41100,R3
 6221   031716                              INLINE <SBC R2>
        031716  005602                                                                         SBC R2
 6222   031720                              LET R2 := R2 - #17
        031720  162702  000017                                                                 SUB     #17,R2
 6223   031724                              LET (R0) :B= (R0) + #1
        031724  105210                                                                         INCB    (R0)
 6224   031726                          ENDDO
        031726  000766                                                                         BR      $51276
        031730                                                                        $51277:
 6225   031730                          IF R2 EQ #17 AND R3 HIS #41100 THEN
        031730  020227  000017                                                                  CMP     R2,#17
        031734  001007                                                                         BNE     $51300
        031736  020327  041100                                                                 CMP     R3,#41100
        031742  103404                                                                         BLO     $51300
 6226   031744                              LET R3 := R3 - #41100
        031744  162703  041100                                                                 SUB     #41100,R3
 6227   031750                              LET R2 := #0
        031750  005002                                                                         CLR     R2
 6228   031752                              LET (R0) :B= (R0) + #1
        031752  105210                                                                         INCB    (R0)
 6229   031754                          ENDIF
        031754                                                                        $51300:
 6230
 6231                                   ;+
 6232                                   ;   GET SIXTH BCD DIGIT
 6233                                   ;-
 6234
 6235   031754                          LET R0 := R0 + #1
        031754  005200                                                                         INC     R0
 6236   031756                          WHILE R2 HI #1 DO
        031756                                                                        $51301:
        031756  020227  000001                                                                  CMP     R2,#1
        031762  101406                                                                         BLOS    $51302
 6237   031764                              LET R3 := R3 - #103240
        031764  162703  103240                                                                 SUB     #103240,R3
 6238   031770                              INLINE <SBC R2>
        031770  005602                                                                         SBC R2
 6239   031772                              LET R2 := R2 - #1
        031772  005302                                                                         DEC     R2
 6240   031774                              LET (R0) :B= (R0) + #1
        031774  105210                                                                         INCB    (R0)
```

CONVERT LONG INTEGER TO INTERMEDIATE FORM ROUTINE

```
6241 031776                             ENDDO
     031776  000767                                                  BR      $51301
     032000                                                $51302:
6242 032000                             IF R2 EQ #1 AND R3 HIS #103240 THEN
     032000  020227  000001                                         CMP     R2,#1
     032004  001007                                                 BNE     $51303
     032006  020327  103240                                         CMP     R3,#103240
     032012  103404                                                BLO     $51303
6243 032014                                 LET R3 := R3 - #103240
     032014  162703  103240                                         SUB     #103240,R3
6244 032020                                 LET R2 := #0
     032020  005002                                                 CLR     R2
6245 032022                                 LET (R0) :B= (R0) + #1
     032022  105210                                                 INCB    (R0)
6246 032024                             ENDIF
     032024                                                $51303:
6247
6248                                ;+
6249                                ;    GET FIFTH BCD DIGIT
6250                                ;-
6251
6252 032024                             LET R0 := R0 + #1
     032024  005200                                                 INC     R0
6253 032026                             WHILE R2 EQ #1 DO
     032026                                                $51304:
     032026  020227  000001                                         CMP     R2,#1
     032032  001005                                                 BNE     $51305
6254 032034                                 LET R3 := R3 - #23420
     032034  162703  023420                                         SUB     #23420,R3
6255 032040                                 INLINE <SBC R2>
     032040  005602                                                 SBC R2
6256 032042                                 LET (R0) :B= (R0) + #1
     032042  105210                                                 INCB    (R0)
6257 032044                             ENDDO
     032044  000770                                                 BR      $51304
     032046                                                $51305:
6258 032046                             WHILE R3 HIS #23420 DO
     032046                                                $51306:
     032046  020327  023420                                         CMP     R3,#23420
     032052  103404                                                BLO     $51307
6259 032054                                 LET R3 := R3 - #23420
     032054  162703  023420                                         SUB     #23420,R3
6260 032060                                 LET (R0) :B= (R0) + #1
     032060  105210                                                 INCB    (R0)
6261 032062                             ENDDO
     032062  000771                                                 BR      $51306
     032064                                                $51307:
6262
6263                                ;+
6264                                ;    GET FOURTH BCD DIGIT
6265                                ;-
6266
6267 032064                             LET R0 := R0 + #1
     032064  005200                                                 INC     R0
6268 032066                             WHILE R3 HIS #1750 DO
     032066                                                $51310:
     032066  020327  001750                                         CMP     R3,#1750
```

```
      032072  103404
 6269 032074                          LET R3 := R3 - #1750
      032074  162703  001750
 6270 032100                          LET (R0) :B= (R0) + #1
      032100  105210
 6271 032102                       ENDDO
      032102  000771
      032104                                                        $51311:
 6272
 6273                           ;+
 6274                           ;   GET THIRD BCD DIGIT
 6275                           ;-
 6276
 6277 032104                       LET R0 := R0 + #1
      032104  005200                                                INC     R0
 6278 032106                       WHILE R3 HIS #144 DO
      032106                                                        $51312:
      032106  020327  000144                                        CMP     R3,#144
      032112  103404                                                BLO     $51313
 6279 032114                          LET R3 := R3 - #144
      032114  162703  000144                                        SUB     #144,R3
 6280 032120                          LET (R0) :B= (R0) + #1
      032120  105210                                                INCB    (R0)
 6281 032122                       ENDDO
      032122  000771                                                BR      $51312
      032124                                                        $51313:
 6282
 6283                           ;+
 6284                           ;    GET SECOND AND FIRST BCD DIGITS
 6285                           ;-
 6286
 6287 032124                       LET R0 := R0 + #1
      032124  005200                                                INC     R0
 6288 032126                       WHILE R3 HIS #12 DO
      032126                                                        $51314:
      032126  020327  000012                                        CMP     R3,#12
      032132  103404                                                BLO     $51315
 6289 032134                          LET R3 := R3 - #12
      032134  162703  000012                                        SUB     #12,R3
 6290 032140                          LET (R0) :B= (R0) + #1
      032140  105210                                                INCB    (R0)
 6291 032142                       ENDDO
      032142  000771                                                BR      $51314
      032144                                                        $51315:
 6292 032144                       LET R0 := R0 + #1
      032144  005200                                                INC     R0
 6293 032146                       LET (R0) :B= R3
      032146  110310                                                MOVB    R3,(R0)
 6294
 6295                           ;+
 6296                           ; DETERMINE THE NUMBER OF SIGNIFICANT DIGITS.
 6297                           ;-
 6298
 6299 032150                       LET R0 := #DSBUF1
      032150  012700  064440                                        MOV     #DSBUF1,R0
 6300 032154                       WHILEB (R0) EQ #0 DO
      032154                                                        $51316:
```

CONVERT LONG INTEGER TO INTERMEDIATE FORM ROUTINE

```
        032154  105710                                                          TSTB    (R0)
        032156  001002                                                          BNE     $51317
   6301 032160                              LET R0 := R0 + #1
        032160  005200                                                          INC     R0
   6302 032162                          ENDDO
        032162  000774                                                          BR      $51316
        032164                                                          $51317:
   6303 032164                          LET R0 := R0 - #DSBUF1          ;# OF LEADING 0'S
        032164  162700  064440                                                  SUB     #DSBUF1,R0
   6304 032170                          LET DSLEN1 := #10. - R0         ;# OF MOST SIGNIFICANT DIGITS
        032170  012767  000012  146316                                          MOV     #10.,DSLEN1
        032176  160067  146312                                                  SUB     R0,DSLEN1
   6305 032202                          CALL RESREG                     ;RESTORE THE GPR'S
        032202  004767  024562                                                  JSR     PC,RESREG
   6306 032206                          INLINE   <XCVTLI:>
        032206                                                                  XCVTLI:
   6307 032206                      ENDRTN
        032206                                                          $51255:
        032206                                                          $51256:
        032206  000207                                                          RTS     PC
   6308
   6309                             ;;*****************************************************************
   6310                             .SBTTL  CONVERT INTERMEDIATE FORM TO LONG INTEGER ROUTINE
   6311                             ;;*****************************************************************
   6312                             ;* THIS ROUTINE CONVERTS A 10 BCD DIGIT INTERMEDIATE FORM INTO
   6313                             ;* LONG INTEGER FORMAT.  THE INPUT IS 10 BCD DIGITS IN THE 10 BYTES STARTING
   6314                             ;* AT DSBUF1, MOST SIGNIFICANT DIGIT FIRST; AND SIGNO.
   6315                             ;* THE OUTPUT IS A TWO'S COMPLEMENT 31 BIT BINARY LONG INTEGER IN LOCATIONS
   6316                             ;* R2.EXP AND R3.EXP.
   6317                             ;* THIS ROUTINE ASSUMES THAT THE 10 BCD DIGIT NUMBER IS SMALL ENOUGH
   6318                             ;* TO BE REPRESENTED AS A 31 BIT BINARY INTEGER.
   6319
   6320 032210                      ROUTINE CVTIL
        032210                                                          CVTIL:
   6321
   6322                             ;+
   6323                             ; ROUTINE INITIALIZATION
   6324                             ;-
   6325
   6326 032210                          CALL SAVREG                     ;SAVE THE GPR'S
        032210  004767  024522                                                  JSR     PC,SAVREG
   6327 032214                          LET R2.EXP := #0
        032214  005067  146310                                                  CLR     R2.EXP
   6328 032220                          LET R3.EXP := #0
        032220  005067  146306                                                  CLR     R3.EXP
   6329 032224                          LET R0 := #DSBUF1
        032224  012700  064440                                                  MOV     #DSBUF1,R0
   6330
   6331                             ;+
   6332                             ;+
   6333                             ; ADD IN BINARY EQUIVALENT OF TENTH BCD DIGIT
   6334                             ;-
   6335
   6336 032230                          WHILEB (R0) GT #0 DO
        032230                                                          $51322:
        032230  105710                                                          TSTB    (R0)
        032232  003412                                                          BLE     $51323
```

```
6337 032234                                 LET R3.EXP := R3.EXP + #145000
     032234  062767  145000  146270                                        ADD     #145000,R3.EXP
6338 032242                                 INLINE   <ADC  R2.EXP>
     032242  005567  146262                                              ADC  R2.EXP
6339 032246                                 LET R2.EXP := R2.EXP + #35632
     032246  062767  035632  146254                                      ADD     #35632,R2.EXP
6340 032254                                 LET (R0) :B= (R0) - #1
     032254  105310                                                      DECB    (R0)
6341 032256                               ENDDO
     032256  000764                                                      BR      $51322
     032260                                                       $51323:
6342
6343                                    ;+
6344                                    ; ADD IN BINARY EQUIVALENT OF NINTH BCD DIGIT
6345                                    ;-
6346
6347 032260                               LET R0 := R0 + #1
     032260  005200                                                      INC     R0
6348 032262                               WHILEB (R0) GT #0 DO
     032262                                                       $51324:
     032262  105710                                                      TSTB    (R0)
     032264  003412                                                      BLE     $51325
6349 032266                                   LET R3.EXP := R3.EXP + #160400
     032266  062767  160400  146236                                      ADD     #160400,R3.EXP
6350 032274                                   INLINE   <ADC  R2.EXP>
     032274  005567  146230                                              ADC  R2.EXP
6351 032300                                   LET R2.EXP := R2.EXP + #2765
     032300  062767  002765  146222                                      ADD     #2765,R2.EXP
6352 032306                                   LET (R0) :B= (R0) - #1
     032306  105310                                                      DECB    (R0)
6353 032310                               ENDDO
     032310  000764                                                      BR      $51324
     032312                                                       $51325:
6354
6355                                    ;+
6356                                    ; ADD IN BINARY EQUIVALENT OF EIGHTH BCD DIGIT
6357                                    ;-
6358
6359 032312                               LET R0 := R0 + #1
     032312  005200                                                      INC     R0
6360 032314                               WHILEB (R0) GT #0 DO
     032314                                                       $51326:
     032314  105710                                                      TSTB    (R0)
     032316  003412                                                      BLE     $51327
6361 032320                                   LET R3.EXP := R3.EXP + #113200
     032320  062767  113200  146204                                      ADD     #113200,R3.EXP
6362 032326                                   INLINE   <ADC  R2.EXP>
     032326  005567  146176                                              ADC  R2.EXP
6363 032332                                   LET R2.EXP := R2.EXP + #230
     032332  062767  000230  146170                                      ADD     #230,R2.EXP
6364 032340                                   LET (R0) :B= (R0) - #1
     032340  105310                                                      DECB    (R0)
6365 032342                               ENDDO
     032342  000764                                                      BR      $51326
     032344                                                       $51327:
6366
6367                                    ;+
```

```
                                ; ADD IN BINARY EQUIVALENT OF SEVENTH BCD DIGIT
                                ;-
                                   LET R0 := R0 + #1                                    INC     R0
 032344  005200
                                   WHILEB (R0) GT #0 DO                          $51330:
 032346                                                                                 TSTB    (R0)
 032346  105710                                                                         BLE     $51331
 032350  003412
                                       LET R3.EXP := R3.EXP + #41100                    ADD     #41100,R3.EXP
 032352  062767  041100  146152
                                       INLINE   <ADC  R2.EXP>                           ADC  R2.EXP
 032360  005567  146144
                                       LET R2.EXP := R2.EXP + #17                       ADD     #17,R2.EXP
 032364  062767  000017  146136
                                       LET (R0) :B= (R0) - #1                           DECB    (R0)
 032372  105310
                                   ENDDO                                                BR      $51330
 032374  000764                                                                  $51331:
 032376
                                ;+
                                ; ADD IN BINARY EQUIVALENT OF SIXTH BCD DIGIT
                                ;-
                                   LET R0 := R0 + #1                                    INC     R0
 032376  005200
                                   WHILEB (R0) GT #0 DO                          $51332:
 032400                                                                                 TSTB    (R0)
 032400  105710                                                                        BLE     $51333
 032402  003411
                                       LET R3.EXP := R3.EXP + #103240                   ADD     #103240,R3.EXP
 032404  062767  103240  146120
                                       INLINE   <ADC  R2.EXP>                           ADC  R2.EXP
 032412  005567  146112
                                       LET R2.EXP := R2.EXP + #1                        INC     R2.EXP
 032416  005267  146106
                                       LET (R0) :B= (R0) - #1                           DECB    (R0)
 032422  105310
                                   ENDDO                                                BR      $51332
 032424  000765                                                                  $51333:
 032426
                                ;+
                                ; ADD IN BINARY EQUIVALENT OF FIFTH BCD DIGIT
                                ;-
                                   LET R0 := R0 + #1                                    INC     R0
 032426  005200
                                   WHILEB (R0) GT #0 DO                          $51334:
 032430                                                                                 TSTB    (R0)
 032430  105710                                                                        BLE     $51335
 032432  003407
                                       LET R3.EXP := R3.EXP + #23420                    ADD     #23420,R3.EXP
 032434  062767  023420  146070
                                       INLINE   <ADC  R2.EXP>                           ADC  R2.EXP
 032442  005567  146062
```

```
 6399 032446                                LET (R0) :B= (R0) - #1                                  DECB    (R0)
      032446  105310
 6400 032450                              ENDDO                                                     BR      $51334
      032450  000767
      032452                                                                                $51335:
 6401
 6402                                  ;+
 6403                                  ; ADD IN BINARY EQUIVALENT OF FOURTH BCD DIGIT
 6404                                  ;-
 6405
 6406 032452                              LET R0 := R0 + #1
      032452  005200                                                                                INC     R0
 6407 032454                              WHILEB (R0) GT #0 DO
      032454                                                                                $51336:
      032454  105710                                                                                TSTB    (R0)
      032456  003407                                                                                BLE     $51337
 6408 032460                                  LET R3.EXP := R3.EXP + #1750
      032460  062767  001750  146044                                                                ADD     #1750,R3.EXP
 6409 032466                                  INLINE   <ADC  R2.EXP>
      032466  005567  146036                                                                        ADC  R2.EXP
 6410 032472                                  LET (R0) :B= (R0) - #1
      032472  105310                                                                                DECB    (R0)
 6411 032474                              ENDDO
      032474  000767                                                                                BR      $51336
      032476                                                                                $51337:
 6412
 6413                                  ;+
 6414                                  ; ADD IN BINARY EQUIVALENT OF THIRD BCD DIGIT
 6415                                  ;-
 6416
 6417 032476                              LET R0 := R0 + #1
      032476  005200                                                                                INC     R0
 6418 032500                              WHILEB (R0) GT #0 DO
      032500                                                                                $51340:
      032500  105710                                                                                TSTB    (R0)
      032502  003407                                                                                BLE     $51341
 6419 032504                                  LET R3.EXP := R3.EXP + #144
      032504  062767  000144  146020                                                                ADD     #144,R3.EXP
 6420 032512                                  INLINE   <ADC  R2.EXP>
      032512  005567  146012                                                                        ADC  R2.EXP
 6421 032516                                  LET (R0) :B= (R0) - #1
      032516  105310                                                                                DECB    (R0)
 6422 032520                              ENDDO
      032520  000767                                                                                BR      $51340
      032522                                                                                $51341:
 6423
 6424                                  ;+
 6425                                  ; ADD IN BINARY EQUIVALENT OF SECOND BCD DIGIT
 6426                                  ;-
 6427
 6428 032522                              LET R0 := R0 + #1
      032522  005200                                                                                INC     R0
 6429 032524                              WHILEB (R0) GT #0 DO
      032524                                                                                $51342:
      032524  105710                                                                                TSTB    (R0)
      032526  003407                                                                                BLE     $51343
 6430 032530                                  LET R3.EXP := R3.EXP + #12
```

```
      032530  062767  000012  145774
 6431 032536                          INLINE   <ADC  R2.EXP>
      032536  005567  145766
 6432 032542                          LET (R0) :B= (R0) - #1
      032542  105310
 6433 032544                        ENDDO
      032544  000767
      032546
 6434
 6435                             ;+
 6436                             ; ADD IN FIRST DIGIT
 6437                             ;-
 6438
 6439 032546                        LET R0 := R0 + #1
      032546  005200
 6440 032550                        INLINE  <SWAB        (R0)>
      032550  000310
 6441 032552                        LET R3.EXP := R3.EXP + (R0)
      032552  061067  145754
 6442 032556                        INLINE   <ADC  R2.EXP>
      032556  005567  145746
 6443 032562                        IF SIGNO EQ #-1 THEN
      032562  026727  145662  177777
      032570  001011
 6444 032572                            LET R3.EXP := NEGATE R3.EXP
      032572  005467  145734
 6445 032576                            LET R2.EXP := COMP R2.EXP
      032576  005167  145726
 6446 032602                            IF R3.EXP EQ #0 THEN
      032602  005767  145724
      032606  001002
 6447 032610                                LET R2.EXP := R2.EXP + #1
      032610  005267  145714
 6448 032614                            ENDIF
      032614
 6449 032614                        ENDIF
      032614
 6450 032614                        CALL RESREG                  ;RESTORE GPR'S
      032614  004767  024150
 6451 032620                      ENDRTN
      032620
      032620
      032620  000207
 6452
 6453 032622                        INLINE   <CISSEND:>
      032622
 6454
```

```
                        ADD     #12,R3.EXP
                        ADC  R2.EXP
                        DECB    (R0)
                        BR      $51342
                $51343:

                        INC     R0
                        SWAB    (R0)
                        ADD     (R0),R3.EXP
                        ADC  R2.EXP
                        CMP     SIGNO,#-1
                        BNE     $51344
                        NEG     R3.EXP
                        COM     R2.EXP
                        TST     R3.EXP
                        BNE     $51345
                        INC     R2.EXP
                $51345:
                $51344:
                        JSR     PC,RESREG
                $51320:
                $51321:
                        RTS     PC

                        CISSEND:
```

```
 6456                                ;*****************************************************************
 6457                                ;*****************************************************************
 6458                                ;**
 6459                                ;               C I S 6     M O D U L E
 6460                                ;
 6461                                ;  THIS MODULE TESTS THE NUMERIC DATA TYPE ARITHMETIC
 6462                                ;  INSTRUCTIONS: ADDN, SUBN, CMPN, AND ASHN.
 6463                                ;--
 6464                                ;*****************************************************************
 6465                                ;*****************************************************************
 6466
 6467
 6468
 6469                                .SBTTL  CIS6 INITIALIZATION
 6470                                ;*****************************************************************
 6471                                ;
 6472                                ;  CIS6 INITIALIZATION
 6473                                ;
 6474                                ;*****************************************************************
 6475
 6476                                ;IF SWR<5:0> ARE CLEARED OR IF SWR<2> IS SET THEN EXECUTE
 6477                                ;  THE CIS6 MODULE; OTHERWISE, SKIP TO THE NEXT CIS MODULE.
 6478
 6479 032622                           INLINE   <SCOPE>
      032622  000004                                                                  SCOPE
 6480 032624                           INLINE <CKSWR>
      032624  104406                                                                  CKSWR
 6481 032626                           LET TEMPO := @SWR CLR.BY #177700
      032626  017767  145574  145642                                                 MOV     @SWR,TEMPO
      032634  042767  177700  145634                                                 BIC     #177700,TEMPO
 6482 032642                           IF TEMPO NE #0 THEN
      032642  005767  145630                                                         TST     TEMPO
      032646  001406                                                                 BEQ     $51346
 6483 032650                               IF #BIT2 NOTSETIN TEMPO THEN
      032650  032767  000004  145620                                                 BIT     #BIT2,TEMPO
      032656  001002                                                                 BNE     $51347
 6484 032660                                   INLINE   <JMP   CIS6END>
      032660  000167  012502                                                         JMP     CIS6END
 6485 032664                               ENDIF
      032664                                                                  $51347:
 6486 032664                           ENDIF
      032664                                                                  $51346:
 6487 032664                           LET CTL := #6                 ;CONTROL CHIP NUMBER
      032664  012767  000006  145654                                                 MOV     #6,CTL
 6488 032672                           LET ERRFLG := #0      ;INITIALIZE ERROR FLAG
      032672  005067  145560                                                         CLR     ERRFLG
 6489 032676                           LET PSW := #340
      032676  012767  000340  145072                                                 MOV     #340,PSW
 6490 032704                           LET ERR := #1
      032704  012767  000001  145640                                                 MOV     #1,ERR
 6491
 6492
```

```
 6494                                   .SBTTL  ADDN
 6495                                   ;*****************************************************************
 6496                                   ;
 6497                                   ;       ADDN TEST
 6498                                   ;
 6499                                   ;*****************************************************************
 6500
 6501 032712                           ROUTINE XADDN
      032712                                                                          XADDN:
 6502 032712                              INLINE <SCOPE>
      032712  000004                                                                          SCOPE
 6503 032714                              INLINE <NOP>
      032714  000240                                                                          NOP
 6504 032716                              LET $TIMES := #10           ;ITERATION COUNT
      032716  012767  000010  145460                                                        MOV     #10,$TIMES
 6505 032724                              LET COUNT := TADD           ;# OF TEST CASES
      032724  016767  032320  145536                                                        MOV     TADD,COUNT
 6506 032732                              LET TSTCAS := #TADD+2               ;POINT TO FIRST TEST CASE
      032732  012767  065252  145604                                                        MOV     #TADD+2,TSTCAS
 6507
 6508                                   ;+
 6509                                   ;       CLR R0.EXT TO R3.EXP
 6510                                   ;-
 6511
 6512 032740                              INCR  R0 FROM #0 TO #6 BY #2
      032740  005000                                                                      CLR     R0
      032742  000402                                                                      BR      $51352
      032744                                                                      $51353:
      032744  062700  000002                                                              ADD     #2,R0
      032750                                                                      $51352:
      032750  020027  000006                                                              CMP     R0,#6
      032754  003003                                                                      BGT     $51354
 6513 032756                                  INLINE <CLR R0.EXP(R0)>
      032756  005060  000524                                                              CLR R0.EXP(R0)
 6514 032762                              ENDINC
      032762  000770                                                                      BR      $51353
      032764                                                                      $51354:
 6515                                   ;REPEAT OUTER LOOP FOR TEST CASES
 6516 032764                              INLINE <$5050:>
      032764                                                                          $5050:
 6517 032764                              CALL CLRDSB                 ;CLR DS BUFFERS
      032764  004767  023704                                                              JSR     PC,CLRDSB
 6518
 6519                                   ;+
 6520                                   ;       LOAD DATAXX AND CC.EXP FROM TEST CASE
 6521                                   ;-
 6522
 6523 032770                              LET R0 := TSTCAS
      032770  016700  145550                                                              MOV     TSTCAS,R0
 6524 032774                              LET DATA00 := (R0)+         ;SRC1 LENGTH
      032774  012067  145562                                                              MOV     (R0)+,DATA00
 6525 033000                              LET DATA01 := (R0)+         ;POINTER TO SRC1 DATA
      033000  012067  145560                                                              MOV     (R0)+,DATA01
 6526 033004                              LET DATA02 := (R0)+         ;SRC2 LENGTH
      033004  012067  145556                                                              MOV     (R0)+,DATA02
 6527 033010                              LET DATA03 := (R0)+         ;POINTER TO SRC2 DATA
      033010  012067  145554                                                              MOV     (R0)+,DATA03
```

```
6528 033014
     033014  012067  145552          LET DATA04 := (R0)+          ;DEST LENGTH                MOV     (R0)+,DATA04
6529 033020
     033020  012067  145550          LET DATA05 := (R0)+          ;POINTER TO DS.EXP DATA (SUM)
                                                                                              MOV     (R0)+,DATA05
6530 033024
     033024  012067  145472          LET CC.EXP := (R0)+          ;CC.EXP (SUM)
                                                                                              MOV     (R0)+,CC.EXP
6531
6532                                ;+
6533                                ;       LOAD DSBUF1
6534                                ;-
6535
6536 033030
     033030  005767  145526          IF DATA00 NE #0 THEN         ;CHECK FOR ZERO LENGTH BUFFER
                                                                                              TST     DATA00
     033034  001413                                                                           BEQ     $51355
6537 033036
     033036  016700  145522              LET R0 := DATA01                                    MOV     DATA01,R0
6538 033042
     033042  012067  145402              LET SIGN0 := (R0)+       ;STORE SIGN AND POINT AT FIRST BYTE
                                                                                              MOV     (R0)+,SIGN0
6539 033046
     033046  016701  145510              LET R1 := DATA00         ;# OF BYTES TO TRANSFER
                                                                                              MOV     DATA00,R1
6540 033052
     033052  012702  064440              LET R2 := #DSBUF1        ;TRANSFER DESTINATION
                                                                                              MOV     #DSBUF1,R2
6541 033056
     033056                               INLINE <1$:>                                         1$:
6542 033056
     033056  112022                       INLINE <MOVB (R0)+,(R2)+>                           MOVB (R0)+,(R2)+
6543 033060
     033060  077102                       INLINE <SOB R1,1$>                                  SOB R1,1$
6544 033062
     033062  000402                  ELSE                                                    BR      $51356
     033064                                                                       $51355:
6545 033064
     033064  005067  145360              LET SIGN0 := #0                                     CLR     SIGN0
6546 033070                         ENDIF
     033070                                                                       $51356:
6547
6548                                ;+
6549                                ;       LOAD DSBUF2
6550                                ;-
6551
6552 033070
     033070  005767  145472          IF DATA02 NE #0 THEN                                    TST     DATA02
     033074  001413                                                                           BEQ     $51357
6553 033076
     033076  016700  145466              LET R0 := DATA03                                    MOV     DATA03,R0
6554 033102
     033102  012067  145344              LET SIGN1 := (R0)+                                  MOV     (R0)+,SIGN1
6555 033106
     033106  016701  145454              LET R1 := DATA02                                    MOV     DATA02,R1
6556 033112
     033112  012702  064502              LET R2 := #DSBUF2                                   MOV     #DSBUF2,R2
6557 033116
     033116                               INLINE <2$:>                                         2$:
6558 033116
     033116  112022                       INLINE <MOVB (R0)+,(R2)+>                           MOVB (R0)+,(R2)+
6559 033120
     033120  077102                       INLINE <SOB R1,2$>                                  SOB R1,2$
```

```
033122                                  ELSE
033122  000402                                                          BR      $51360
033124                                                          $51357:
033124                                      LET SIGN1 := #0
033124  005067  145322                                                  CLR     SIGN1
033130                                  ENDIF
033130                                                          $51360:

                                    ;*
                                    ;       LOAD DS.EXP
                                    ;-

033130                              IF DATA04 NE #0 THEN
033130  005767  145436                                                  TST     DATA04
033134  001415                                                          BEQ     $51361
033136                                      LET R0 := DATA05
033136  016700  145432                                                  MOV     DATA05,R0
033142                                      LET SIGN2 := (R0)+
033142  012067  145306                                                  MOV     (R0)+,SIGN2
033146                                      LET R1 := DATA04
033146  016701  145420                                                  MOV     DATA04,R1
033152                                      LET R2 := #DS.EXP
033152  012702  064606                                                  MOV     #DS.EXP,R2
033156                                      INLINE <3$:>
033156                                                          3$:
033156                                      INLINE <MOVB     (R0)+,(R2)>
033156  112012                                                          MOVB    (R0)+,(R2)
033160                                      INLINE <BISB     #60,(R2)+>
033160  152722  000060                                                  BISB    #60,(R2)+
033164                                      INLINE <SOB R1,3$>
033164  077104                                                          SOB R1,3$
033166                                  ELSE
033166  000402                                                          BR      $51362
033170                                                          $51361:
033170                                      LET SIGN2 := #0
033170  005067  145260                                                  CLR     SIGN2
033174                                  ENDIF
033174                                                          $51362:

                                    ;*
                                    ;       LOAD DSTMP'S WITH MDS & LSD OF EACH BUFFER
                                    ;-

033174                              LET DSTMP0 :B= DSBUF1
033174  116767  031240  145352                                          MOVB    DSBUF1,DSTMP0
033202                              LET TEMP0 := #DSBUF1-1 + DATA00
033202  012767  064437  145266                                          MOV     #DSBUF1-1,TEMP0
033210  066767  145346  145260                                          ADD     DATA00,TEMP0
033216                              LET DSTMP1 :B= @TEMP0
033216  117767  145254  145331                                          MOVB    @TEMP0,DSTMP1
033224                              LET DSTMP2 :B= DSBUF2
033224  116767  031252  145324                                          MOVB    DSBUF2,DSTMP2
033232                              LET TEMP0 := #DSBUF2-1 + DATA02
033232  012767  064501  145236                                          MOV     #DSBUF2-1,TEMP0
033240  066767  145322  145230                                          ADD     DATA02,TEMP0
033246                              LET DSTMP3 :B= @TEMP0
033246  117767  145224  145303                                          MOVB    @TEMP0,DSTMP3
```

```
6591 033254                                LET DSTMP4 :B= DS.EXP
     033254  116767  031326  145276                                    MOVB    DS.EXP,DSTMP4
6592 033262                                LET TEMP0 := #DS.EXP-1 + DATA04
     033262  012767  064605  145206                                    MOV     #DS.EXP-1,TEMP0
     033270  066767  145276  145200                                    ADD     DATA04,TEMP0
6593 033276                                LET DSTMP5 :B= @TEMP0
     033276  117767  145174  145255                                    MOVB    @TEMP0,DSTMP5
6594 033304                                LET DESCR1 := #DSBUF1
     033304  012767  064440  145306                                    MOV     #DSBUF1,DESCR1
6595 033312                                LET DESCR3 := #DSBUF2
     033312  012767  064502  145304                                    MOV     #DSBUF2,DESCR3
6596 033320                                LET DESCR5 := #DS.DST
     033320  012767  064544  145302                                    MOV     #DS.DST,DESCR5
6597
6598                                  ; INCR TEMP0 FROM #1 TO #5 BY #1      SRC1 DATA TYPE LOOP
6599 033326                                INLINE <CLR @#TEMP0>
     033326  005037  000476                                            CLR @#TEMP0
6600 033332                                INLINE <INC1: INC TEMP0>
     033332  005267  145140                                            INC1: INC TEMP0
6601 033336                                IF TEMP0 GT #5 THEN
     033336  026727  145134  000005                                    CMP     TEMP0,#5
     033344  003402                                                   BLE     $51363
6602 033346                                    INLINE <JMP END1>
     033346  000167  001622                                            JMP END1
6603 033352                                ENDIF
     033352                                                       $51363:
6604
6605                                  ;+
6606                                  ;    RESTORE MSD & LSD OF DSBUF1
6607                                  ;-
6608
6609 033352                                LET DSBUF1 :B= DSTMP0          ;MSD
     033352  116767  145176  031060                                    MOVB    DSTMP0,DSBUF1
6610 033360                                LET TEMP2 := #DSBUF1-1 + DATA00
     033360  012767  064437  145114                                    MOV     #DSBUF1-1,TEMP2
     033366  066767  145170  145106                                    ADD     DATA00,TEMP2
6611 033374                                LET @TEMP2 :B= DSTMP1          ;LSD
     033374  116777  145155  145100                                    MOVB    DSTMP1,@TEMP2
6612
6613                                  ;+
6614                                  ;    SIGN DSBUF1 BY DATA TYPE
6615                                  ;-
6616
6617 033402                                SELECT TEMP0 OF 5
     033402  016746  145070                                            MOV     TEMP0,-(SP)
     033406  006316                                                   ASL     (SP)
     033410  060716                                                   ADD     PC,(SP)
     033412  063607                                                   ADD     @(SP)+,PC
     033414                                                       $51364:
     033414  000012                                                    .WORD   $51372-$51364
     033416  000052                                                    .WORD   $51371-$51364
     033420  000116                                                    .WORD   $51370-$51364
     033422  000162                                                    .WORD   $51367-$51364
     033424  000226                                                    .WORD   $51366-$51364
6618 033426                                    CASE 1
     033426                                                       $51372:
6619 033426                                        LET DESCR0 := DATA00          ;SIGNED ZONED
```

```
      033426  016767  145130  145162
 6620 033434                          IF SIGNO EQ #0 THEN                                                MOV     DATA00,DESCR0
      033434  005767  145010                                                                             TST     SIGNO
      033440  001004                                                                                     BNE     $51373
 6621 033442                              LET @TEMP2 :B= @TEMP2 SET.BY #60 ;+
      033442  152777  000060  145032                                                                     BISB    #60,@TEMP2
 6622 033450                          ELSE
      033450  000403                                                                                     BR      $51374
      033452                                                                                  $51373:
 6623 033452                              LET @TEMP2 :B= @TEMP2 SET.BY #160   ;-
      033452  152777  000160  145022                                                                     BISB    #160,@TEMP2
 6624 033460                          ENDIF
      033460                                                                                  $51374:
 6625 033460                          LET DSBUF1-1 :B= #0          ;FIX LEADING SEPARATE RESULT
      033460  105067  030753                                                                             CLRB    DSBUF1-1
 6626
 6627 033464                      CASE 2
      033464  000512                                                                                     BR      $51365
      033466                                                                                  $51371:
 6628 033466                          LET DESCR0 := DATA00 SET.BY #20000   ;TRAILING OVERPUNCH
      033466  016767  145070  145122                                                                     MOV     DATA00,DESCR0
      033474  052767  020000  145114                                                                     BIS     #20000,DESCR0
 6629 033502                          LET R5 :B= DSTMP1    ;GET LSD
      033502  116705  145047                                                                             MOVB    DSTMP1,R5
 6630 033506                          IF SIGNO EQ #-1 THEN
      033506  026727  144736  177777                                                                     CMP     SIGNO,#-1
      033514  001002                                                                                     BNE     $51375
 6631 033516                              LET R5 := R5 + #10.          ;CHANGE INDEX FOR -
      033516  062705  000012                                                                             ADD     #10.,R5
 6632 033522                          ENDIF
      033522                                                                                  $51375:
 6633 033522                          LET @TEMP2 :B= OVPNCH(R5)
      033522  116577  056330  144752                                                                         MOVB    OVPNCH(R5),@TEMP2
 6634
 6635 033530                      CASE 3
      033530  000470                                                                                     BR      $51365
      033532                                                                                  $51370:
 6636 033532                          LET DESCR0 := DATA00 SET.BY #30000   ;LEADING OVERPUNCH
      033532  016767  145024  145056                                                                     MOV     DATA00,DESCR0
      033540  052767  030000  145050                                                                     BIS     #30000,DESCR0
 6637 033546                          LET R5 :B= DSTMP0    ;GET MSD
      033546  116705  145002                                                                             MOVB    DSTMP0,R5
 6638 033552                          IF SIGNO EQ #-1 THEN
      033552  026727  144672  177777                                                                     CMP     SIGNO,#-1
      033560  001002                                                                                     BNE     $51376
 6639 033562                              LET R5 := R5 + #10.
      033562  062705  000012                                                                             ADD     #10.,R5
 6640 033566                          ENDIF
      033566                                                                                  $51376:
 6641 033566                          LET DSBUF1 :B= OVPNCH(R5)
      033566  116567  056330  030644                                                                         MOVB    OVPNCH(R5),DSBUF1
 6642
 6643
 6644 033574                      CASE 4
      033574  000446                                                                                     BR      $51365
      033576                                                                                  $51367:
 6645 033576                          LET DESCR0 := DATA00 SET.BY #40000   ;TRAILING SEPARATE
```

```
        033576  016767  144760  145012                                                  MOV     DATA00,DESCR0
        033604  052767  040000  145004                                                  BIS     #40000,DESCR0
 6646 033612                            LET R5 := TEMP2 + #1
        033612  016705  144664                                                          MOV     TEMP2,R5
        033616  005205                                                                  INC     R5
 6647 033620                            IF SIGN0 EQ #0 THEN
        033620  005767  144624                                                          TST     SIGN0
        033624  001003                                                                  BNE     $51377
 6648 033626                                LET (R5) :B= #53        ;+ SIGN
        033626  112715  000053                                                          MOVB    #53,(R5)
 6649 033632                            ELSE
        033632  000402                                                                  BR      $51400
        033634                                                                  $51377:
 6650 033634                                LET (R5) :B= #55        ;- SIGN
        033634  112715  000055                                                          MOVB    #55,(R5)
 6651 033640                            ENDIF
        033640                                                                  $51400:
 6652
 6653 033640                        CASE 5
        033640  000424                                                                  BR      $51365
        033642                                                                  $51366:
 6654 033642                            LET DESCR0 := DATA00 SET.BY #50000   ;LEADING SEPARATE
        033642  016767  144714  144746                                                  MOV     DATA00,DESCR0
        033650  052767  050000  144740                                                  BIS     #50000,DESCR0
 6655 033656                            LET R5 := TEMP2 + #1
        033656  016705  144620                                                          MOV     TEMP2,R5
        033662  005205                                                                  INC     R5
 6656 033664                            LET (R5) :B= #0  ;RESTORE TRAILING SEPARATE
        033664  105015                                                                  CLRB    (R5)
 6657 033666                            IF SIGN0 EQ #0 THEN
        033666  005767  144556                                                          TST     SIGN0
        033672  001004                                                                  BNE     $51401
 6658 033674                                LET DSBUF1-1 :B= #53
        033674  112767  000053  030535                                                  MOVB    #53,DSBUF1-1
 6659 033702                            ELSE
        033702  000403                                                                  BR      $51402
        033704                                                                  $51401:
 6660 033704                                LET DSBUF1-1 :B= #55
        033704  112767  000055  030525                                                  MOVB    #55,DSBUF1-1
 6661 033712                            ENDIF
        033712                                                                  $51402:
 6662 033712                        ENDSELECT
        033712                                                                  $51365:
 6663
 6664                           ;INCR TEMP1 FROM #1 TO #5 BY #1       SRC2 DATA TYPES
 6665 033712                        INLINE <CLR @#TEMP1>
        033712  005037  000500                                                          CLR @#TEMP1
 6666 033716                        INLINE <INC2: INC TEMP1>
        033716  005267  144556                                                          INC2: INC TEMP1
 6667 033722                        IF TEMP1 GT #5 THEN
        033722  026727  144552  000005                                                  CMP     TEMP1,#5
        033730  003402                                                                 BLE     $51403
 6668 033732                            INLINE <JMP END2>
        033732  000167  001232                                                          JMP END2
 6669 033736                        ENDIF
        033736                                                                  $51403:
 6670
```

```
                                ;+
                                ;  RESTORE MSD & LSD OF DSBUF2
                                ;-
                                   LET DSBUF2 :B= DSTMP2
033736  116767  144614  030536                                    MOVB    DSTMP2,DSBUF2
                                   LET TEMP3 := #DSBUF2-1 + DATA02
033744  012767  064501  144532                                    MOV     #DSBUF2-1,TEMP3
033752  066767  144610  144524                                    ADD     DATA02,TEMP3
                                   LET @TEMP3 :B= DSTMP3
033760  116777  144573  144516                                    MOVB    DSTMP3,@TEMP3

                                ;+
                                ;  SIGN DSBUF2 BY DATA TYPE
                                ;-

                                   SELECT TEMP1 OF 5
033766  016746  144506                                            MOV     TEMP1,-(SP)
033772  006316                                                    ASL     (SP)
033774  060716                                                    ADD     PC,(SP)
033776  063607                                                    ADD     @(SP)+,PC
034000                                                    $51404:
034000  000012                                                    .WORD   $51412-$51404
034002  000052                                                    .WORD   $51411-$51404
034004  000116                                                    .WORD   $51410-$51404
034006  000162                                                    .WORD   $51407-$51404
034010  000226                                                    .WORD   $51406-$51404
                                      CASE 1
034012                                                    $51412:
                                         LET DESCR2 := DATA02
034012  016767  144550  144602                                    MOV     DATA02,DESCR2
                                         IF SIGN1 EQ #0 THEN
034020  005767  144426                                            TST     SIGN1
034024  001004                                                    BNE     $51413
                                            LET @TEMP3 :B= @TEMP3 SET.BY #60  ;+
034026  152777  000060  144450                                    BISB    #60,@TEMP3
                                         ELSE
034034  000403                                                    BR      $51414
034036                                                    $51413:
                                            LET @TEMP3 :B= @TEMP3 SET.BY #160;-
034036  152777  000160  144440                                    BISB    #160,@TEMP3
                                         ENDIF
034044                                                    $51414:
                                         LET DSBUF2-1 :B= #0
034044  105067  030431                                            CLRB    DSBUF2-1

                                      CASE 2
034050  000512                                                    BR      $51405
034052                                                    $51411:
                                         LET DESCR2 := DATA02 SET.BY #20000   ;TRAILING OVERPUNCH
034052  016767  144510  144542                                    MOV     DATA02,DESCR2
034060  052767  020000  144534                                    BIS     #20000,DESCR2
                                         LET R5 :B= DSTMP3
034066  116705  144465                                            MOVB    DSTMP3,R5
                                         IF SIGN1 EQ #-1 THEN
034072  026727  144354  177777                                    CMP     SIGN1,#-1
034100  001002                                                    BNE     $51415
```

```
 6697 034102                                    LET R5 := R5 + #10.
      034102  062705  000012                                          ADD     #10.,R5
 6698 034106                                ENDIF
      034106                                                    $51415:
 6699 034106                                LET @TEMP3 :B= OVPNCH(R5)
      034106  116577  056330  144370                                  MOVB    OVPNCH(R5),@TEMP3
 6700
 6701 034114                              CASE 3
      034114  000470                                            BR      $51405
      034116                                                    $51410:
 6702 034116                                LET DESCR2 := DATA02 SET.BY #30000   ;LEADING OVERPUNCH
      034116  016767  144444  144476                                  MOV     DATA02,DESCR2
      034124  052767  030000  144470                                  BIS     #30000,DESCR2
 6703 034132                                LET R5 :B= DSTMP2
      034132  116705  144420                                         MOVB    DSTMP2,R5
 6704 034136                                IF SIGN1 EQ #-1 THEN
      034136  026727  144310  177777                                  CMP     SIGN1,#-1
      034144  001002                                                 BNE     $51416
 6705 034146                                    LET R5 := R5 + #10.
      034146  062705  000012                                          ADD     #10.,R5
 6706 034152                                ENDIF
      034152                                                    $51416:
 6707 034152                                LET DSBUF2 :B= OVPNCH(R5)
      034152  116567  056330  030322                                  MOVB    OVPNCH(R5),DSBUF2
 6708
 6709 034160                              CASE 4
      034160  000446                                            BR      $51405
      034162                                                    $51407:
 6710 034162                                LET DESCR2 := DATA02 SET.BY #40000   ;TRAILING SEPARATE
      034162  016767  144400  144432                                  MOV     DATA02,DESCR2
      034170  052767  040000  144424                                  BIS     #40000,DESCR2
 6711 034176                                LET R5 := TEMP3 + #1
      034176  016705  144302                                         MOV     TEMP3,R5
      034202  005205                                                 INC     R5
 6712 034204                                IF SIGN1 EQ #0 THEN
      034204  005767  144242                                         TST     SIGN1
      034210  001003                                                 BNE     $51417
 6713 034212                                    LET (R5) :B= #53
      034212  112715  000053                                         MOVB    #53,(R5)
 6714 034216                                ELSE
      034216  000402                                                 BR      $51420
      034220                                                    $51417:
 6715 034220                                    LET (R5) :B= #55
      034220  112715  000055                                         MOVB    #55,(R5)
 6716 034224                                ENDIF
      034224                                                    $51420:
 6717
 6718 034224                              CASE 5
      034224  000424                                            BR      $51405
      034226                                                    $51406:
 6719 034226                                LET DESCR2 := DATA02 SET.BY #50000   ;LEADING SEPARATE
      034226  016767  144334  144366                                  MOV     DATA02,DESCR2
      034234  052767  050000  144360                                  BIS     #50000,DESCR2
 6720 034242                                LET R5 := TEMP3 + #1
      034242  016705  144236                                         MOV     TEMP3,R5
      034246  005205                                                 INC     R5
 6721 034250                                LET (R5) :B= #0
```

```
      034250  105015                                                                      CLRB    (R5)
 6722 034252                                 IF SIGN1 EQ #0 THEN
      034252  005767  144174                                                              TST     SIGN1
      034256  001004                                                                      BNE     $51421
 6723 034260                                      LET DSBUF2-1 :B= #53
      034260  112767  000053  030213                                                      MOVB    #53,DSBUF2-1
 6724 034266                                 ELSE
      034266  000403                                                                      BR      $51422
      034270                                                                     $51421:
 6725 034270                                      LET DSBUF2-1 :B= #55
      034270  112767  000055  030203                                                      MOVB    #55,DSBUF2-1
 6726 034276                                 ENDIF
      034276                                                                     $51422:
 6727 034276                       ENDSELECT
      034276                                                                     $51405:
 6728
 6729                          ;INCR DATTYP FROM #1 TO #5 BY #1     ;DS.EXP LOOP
 6730 034276                       INLINE <CLR @#DATTYP>
      034276  005037  000464                                                              CLR @#DATTYP
 6731 034302                       INLINE <INC3: INC DATTYP>
      034302  005267  144156                                                              INC3: INC DATTYP
 6732 034306                       IF DATTYP GT #5 THEN
      034306  026727  144152  000005                                                      CMP     DATTYP,#5
      034314  003402                                                                      BLE     $51423
 6733 034316                            INLINE <JMP END3>
      034316  000167  000642                                                              JMP END3
 6734 034322                       ENDIF
      034322                                                                     $51423:
 6735
 6736                          ;+
 6737                          ;          RESTORE MSD & LSD OF DS.EXP
 6738                          ;-
 6739
 6740 034322                       LET DS.EXP :B= DSTMP4     ;MSD
      034322  116767  144232  030256                                                      MOVB    DSTMP4,DS.EXP
 6741 034330                       LET TEMP4 := #DS.EXP-1 + DATA04
      034330  012767  064605  144150                                                      MOV     #DS.EXP-1,TEMP4
      034336  066767  144230  144142                                                      ADD     DATA04,TEMP4
 6742 034344                       LET @TEMP4 :B= DSTMP5     ;LSD
      034344  116777  144211  144134                                                      MOVB    DSTMP5,@TEMP4
 6743
 6744                          ;+
 6745                          ;          SIGN DS.EXP BY DATA TYPE
 6746                          ;-
 6747
 6748 034352                       SELECT DATTYP OF 5
      034352  016746  144106                                                              MOV     DATTYP,-(SP)
      034356  006316                                                                      ASL     (SP)
      034360  060716                                                                      ADD     PC,(SP)
      034362  063607                                                                      ADD     @(SP)+,PC
      034364                                                                     $51424:
      034364  000012                                                                      .WORD   $51432-$51424
      034366  000060                                                                      .WORD   $51431-$51424
      034370  000136                                                                      .WORD   $51430-$51424
      034372  000214                                                                      .WORD   $51427-$51424
      034374  000260                                                                      .WORD   $51426-$51424
 6749 034376                            CASE 1
```

```
      034376                                                                                    $51432:
 6750 034376                          LET DESCR4 := DATA04          ;SIGNED ZONED
      034376  016767  144170  144222                                                             MOV     DATA04,DESCR4
 6751 034404                          IF DATA04 NE #0 THEN
      034404  005767  144162                                                                     TST     DATA04
      034410  001414                                                                            BEQ     $51433
 6752 034412                              IF SIGN2 EQ #0 THEN
      034412  005767  144036                                                                     TST     SIGN2
      034416  001004                                                                            BNE     $51434
 6753 034420                                  LET @TEMP4 :B= @TEMP4 SET.BY #60
      034420  152777  000060  144060                                                             BISB    #60,@TEMP4
 6754 034426                              ELSE
      034426  000403                                                                            BR      $51435
      034430                                                                                    $51434:
 6755 034430                                  LET @TEMP4 :B= @TEMP4 SET.BY #160
      034430  152777  000160  144050                                                             BISB    #160,@TEMP4
 6756 034436                              ENDIF
      034436                                                                                    $51435:
 6757 034436                              LET DS.EXP-1 :B= #0
      034436  105067  030143                                                                    CLRB    DS.EXP-1
 6758 034442                          ENDIF
      034442                                                                                    $51433:
 6759
 6760 034442                      CASE 2
      034442  000524                                                                            BR      $51425
      034444                                                                                    $51431:
 6761 034444                          LET DESCR4 := DATA04 SET.BY #20000   ;TRAILING OVERPUNCH
      034444  016767  144122  144154                                                             MOV     DATA04,DESCR4
      034452  052767  020000  144146                                                             BIS     #20000,DESCR4
 6762 034460                          IF DATA04 NE #0 THEN
      034460  005767  144106                                                                     TST     DATA04
      034464  001415                                                                            BEQ     $51436
 6763 034466                              LET R5 :B= DSTMP5
      034466  116705  144067                                                                     MOVB    DSTMP5,R5
 6764 034472                              LET R5 := R5 - #60
      034472  162705  000060                                                                     SUB     #60,R5
 6765 034476                              IF SIGN2 EQ #-1 THEN
      034476  026727  143752  177777                                                             CMP     SIGN2,#-1
      034504  001002                                                                            BNE     $51437
 6766 034506                                  LET R5 := R5 + #10.
      034506  062705  000012                                                                    ADD     #10.,R5
 6767 034512                              ENDIF
      034512                                                                                    $51437:
 6768 034512                              LET @TEMP4 :B= OVPNCH(R5)
      034512  116577  056330  143766                                                             MOVB    OVPNCH(R5),@TEMP4
 6769 034520                          ENDIF
      034520                                                                                    $51436:
 6770
 6771 034520                      CASE 3
      034520  000475                                                                            BR      $51425
      034522                                                                                    $51430:
 6772 034522                          LET DESCR4 := DATA04 SET.BY #30000   ;LEADING OVERPUNCH
      034522  016767  144044  144076                                                             MOV     DATA04,DESCR4
      034530  052767  030000  144070                                                             BIS     #30000,DESCR4
 6773 034536                          IF DATA04 NE #0 THEN
      034536  005767  144030                                                                     TST     DATA04
      034542  001415                                                                            BEQ     $51440
```

```
   6774 034544
        034544  116705  144010                      LET R5 :B= DSTMP4                                   MOVB    DSTMP4,R5
   6775 034550
        034550  162705  000060                      LET R5 := R5 - #60                                  SUB     #60,R5
   6776 034554
        034554  026727  143674  177777              IF SIGN2 EQ #-1 THEN                                CMP     SIGN2,#-1
        034562  001002                                                                                  BNE     $51441
   6777 034564
        034564  062705  000012                          LET R5 := R5 + #10.                             ADD     #10.,R5
   6778 034570                                      ENDIF
        034570                                                                          $51441:
   6779 034570
        034570  116567  056330  030010              LET DS.EXP :B= OVPNCH(R5)                           MOVB    OVPNCH(R5),DS.EXP
   6780 034576                                  ENDIF
        034576                                                                          $51440:
   6781
   6782 034576                                  CASE 4
        034576  000446                                                                                  BR      $51425
        034600                                                                          $51427:
   6783 034600                                      LET DESCR4 := DATA04 SET.BY #40000  ;TRAILING SEPARATE
        034600  016767  143766  144020                                                                  MOV     DATA04,DESCR4
        034606  052767  040000  144012                                                                  BIS     #40000,DESCR4
   6784 034614                                      LET R5 := TEMP4 + #1
        034614  016705  143666                                                                          MOV     TEMP4,R5
        034620  005205                                                                                  INC     R5
   6785 034622                                      IF SIGN2 EQ #0 THEN
        034622  005767  143626                                                                          TST     SIGN2
        034626  001003                                                                                  BNE     $51442
   6786 034630
        034630  112715  000053                          LET (R5) :B= #53                                MOVB    #53,(R5)
   6787 034634                                      ELSE
        034634  000402                                                                                  BR      $51443
        034636                                                                          $51442:
   6788 034636
        034636  112715  000055                          LET (R5) :B= #55                                MOVB    #55,(R5)
   6789 034642                                      ENDIF
        034642                                                                          $51443:
   6790
   6791 034642                                  CASE 5
        034642  000424                                                                                  BR      $51425
        034644                                                                          $51426:
   6792 034644                                      LET DESCR4 := DATA04 SET.BY #50000
        034644  016767  143722  143754                                                                  MOV     DATA04,DESCR4
        034652  052767  050000  143746                                                                  BIS     #50000,DESCR4
   6793 034660                                      LET R5 := TEMP4 + #1
        034660  016705  143622                                                                          MOV     TEMP4,R5
        034664  005205                                                                                  INC     R5
   6794 034666                                      LET (R5) :B= #0
        034666  105015                                                                                  CLRB    (R5)
   6795 034670                                      IF SIGN2 EQ #0 THEN
        034670  005767  143560                                                                          TST     SIGN2
        034674  001004                                                                                  BNE     $51444
   6796 034676
        034676  112767  000053  027701                  LET DS.EXP-1 :B= #53                            MOVB    #53,DS.EXP-1
   6797 034704                                      ELSE
        034704  000403                                                                                  BR      $51445
        034706                                                                          $51444:
```

```
ADDN

6798 034706                                  LET DS.EXP-1 :B= #55
     034706  112767  000055  027671                                        MOVB    #55,DS.EXP-1
6799 034714                                ENDIF
     034714                                                          $51445:
6800 034714                             ENDSELECT
     034714                                                          $51425:
6801
6802                                    ;+
6803                                    ;       CLR DS.DST & LOAD DESCRIPTORS
6904                                    ;-
6805
6806 034714                             LET R0 := #DS.DST-1
     034714  012700  064543                                          MOV     #DS.DST-1,R0
6807 034720                             LET R1 := #41
     034720  012701  000041                                          MOV     #41,R1
6808 034724                             INLINE <DOIT1:      CLRB    (R0)+>
     034724  105020                                                  DOIT1:  CLRB    (R0)+
6809 034726                             INLINE <SOB  R1,    DOIT1>
     034726  077102                                                  SOB     R1,     DOIT1
6810 034730                             LET R0 := DESCR0
     034730  016700  143662                                          MOV     DESCR0,R0
6811 034734                             LET R1 := DESCR1
     034734  016701  143660                                          MOV     DESCR1,R1
6812 034740                             LET R2 := DESCR2
     034740  016702  143656                                          MOV     DESCR2,R2
6813 034744                             LET R3 := DESCR3
     034744  016703  143654                                          MOV     DESCR3,R3
6814 034750                             LET R4 := DESCR4
     034750  016704  143652                                          MOV     DESCR4,R4
6815 034754                             LET R5 := DESCR5
     034754  016705  143650                                          MOV     DESCR5,R5
6816 034760                             LET R4.EXP := R4
     034760  010467  143550                                          MOV     R4,R4.EXP
6817 034764                             LET R5.EXP := R5
     034764  010567  143546                                          MOV     R5,R5.EXP
6818 034770                             INLINE <ADDN>
     034770  076050                                                  ADDN
6819
6820                                    ;+
6821                                    ; CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.
6822                                    ;-
6823
6824 034772                             LET CC := PSW
     034772  016767  143000  143520                                  MOV     PSW,CC
6825 035000                             LET CC := CC CLR.BY #177760
     035000  042767  177760  143512                                  BIC     #177760,CC
6826 035006                             IF CC NE CC.EXP THEN
     035006  026767  143506  143506                                  CMP     CC,CC.EXP
     035014  001402                                                  BEQ     $51446
6827 035016                                 CALL ERROR
     035016  004767  022656                                          JSR     PC,ERROR
6828 035022                             ENDIF
     035022                                                          $51446:
6829 035022                             IF R0 NE R0.EXP THEN
     035022  020067  143476                                          CMP     R0,R0.EXP
     035026  001402                                                  BEQ     $51447
6830 035030                                 CALL ERROR
```

```
ADDN

        035030  004767  022644                                                  JSR     PC,ERROR
 6831   035034                          ENDIF
        035034                                                          $51447:
 6832   035034                          IF R1 NE R1.EXP THEN
        035034  020167  143466                                                  CMP     R1,R1.EXP
        035040  001402                                                          BEQ     $51450
 6833   035042                              CALL ERROR
        035042  004767  022632                                                  JSR     PC,ERROR
 6834   035046                          ENDIF
        035046                                                          $51450:
 6835   035046                          IF R2 NE R2.EXP THEN
        035046  020267  143456                                                  CMP     R2,R2.EXP
        035052  001402                                                          BEQ     $51451
 6836   035054                              CALL ERROR
        035054  004767  022620                                                  JSR     PC,ERROR
 6837   035060                          ENDIF
        035060                                                          $51451:
 6838   035060                          IF R3 NE R3.EXP THEN
        035060  020367  143446                                                  CMP     R3,R3.EXP
        035064  001402                                                          BEQ     $51452
 6839   035066                              CALL ERROR
        035066  004767  022606                                                  JSR     PC,ERROR
 6840   035072                          ENDIF
        035072                                                          $51452:
 6841   035072                          IF R4 NE R4.EXP THEN
        035072  020467  143436                                                  CMP     R4,R4.EXP
        035076  001402                                                          BEQ     $51453
 6842   035100                              CALL ERROR
        035100  004767  022574                                                  JSR     PC,ERROR
 6843   035104                          ENDIF
        035104                                                          $51453:
 6844   035104                          IF R5 NE R5.EXP THEN
        035104  020567  143426                                                  CMP     R5,R5.EXP
        035110  001402                                                          BEQ     $51454
 6845   035112                              CALL ERROR
        035112  004767  022562                                                  JSR     PC,ERROR
 6846   035116                          ENDIF
        035116                                                          $51454:
 6847
 6848                                ;+
 6849                                ;       CHECK RESULTS VS EXPECTED RESULTS
 6850                                ;-
 6851
 6852   035116                          LET R0 := #DS.DST-1
        035116  012700  064543                                                  MOV     #DS.DST-1,R0
 6853   035122                          LET R1 := #DS.EXP-1
        035122  012701  064605                                                  MOV     #DS.EXP-1,R1
 6854   035126                          LET ERRFLG := #0
        035126  005067  143324                                                  CLR     ERRFLG
 6855   035132                          WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO
        035132                                                          $51455:
        035132  020027  064604                                                  CMP     R0,#DS.DST+40
        035136  001410                                                          BEQ     $51456
        035140  005767  143312                                                  TST     ERRFLG
        035144  001005                                                          BNE     $51456
 6856   035146                              IFB (R0)+ NE (R1)+ THEN
        035146  122021                                                          CMPB    (R0)+,(R1)+
```

ADDN

```
      035150  001402                                                                BEQ     $51457
 6857 035152                                  CALL ERROR
      035152  004767  022522                                                        JSR     PC,ERROR
 6858 035156                                  ENDIF
      035156                                                                $51457:
 6859 035156                                ENDDO
      035156  000765                                                                BR      $51455
      035160                                                                $51456:
 6860                                    ;ENDINC
 6861 035160                                INLINE <JMP INC3>
      035160  000167  177116                                                        JMP INC3
 6862 035164                                INLINE <END3:>
      035164                                                                        END3:
 6863                                    ;ENDINC
 6864 035164                                INLINE <JMP INC2>
      035164  000167  176526                                                        JMP INC2
 6865 035170                                INLINE <END2:>
      035170                                                                        END2:
 6866                                    ;ENDINC
 6867 035170                                INLINE <JMP INC1>
      035170  000167  176136                                                        JMP INC1
 6868 035174                                INLINE <END1:>
      035174                                                                        END1:
 6869 035174                                LET TSTCAS := TSTCAS + #22
      035174  062767  000022  143342                                                ADD     #22,TSTCAS
 6870 035202                                LET COUNT := COUNT - #1
      035202  005367  143262                                                        DEC     COUNT
 6871                                    ;UNTIL COUNT EQ #0
 6872 035206                                IF COUNT NE #0 THEN
      035206  005767  143256                                                        TST     COUNT
      035212  001402                                                                BEQ     $51460
 6873 035214                                    INLINE <JMP $5050>
      035214  000167  175544                                                        JMP $5050
 6874 035220                                ENDIF
      035220                                                                $51460:
 6875
 6876                                    ;+
 6877                                    ;       INLINE CASE - ADDNI, USING UNSIGNED ZONED DATA
 6878                                    ;-
 6879
 6880 035220                                CALL CLRDSB
      035220  004767  021450                                                        JSR     PC,CLRDSB
 6881
 6882                                    ;+
 6883                                    ;       LOAD DSBUF1
 6884                                    ;-
 6885
 6886 035224                                LET R0 := #ADD1+2                   ;SOURCE
      035224  012700  065430                                                        MOV     #ADD1+2,R0
 6887 035230                                LET R1 := TADDI CLR.BY #10000               ;# OF BYTES TO XFER
      035230  016701  030666                                                        MOV     TADDI,R1
      035234  042701  010000                                                        BIC     #10000,R1
 6888 035240                                LET R2 := #DSBUF1                   ;DESTINATION
      035240  012702  064440                                                        MOV     #DSBUF1,R2
 6889 035244                                INLINE      <1$:    MOVB    (R0)+,  (R2)+>
      035244  112022                                                        1$:     MOVB    (R0)+,  (R2)+

 6890 035246                                INLINE      <       SOB     R1,     1$>
```

ADDN

```
      035246  077102                                                                    SOB     R1,     1$
 6891
 6892                                 ;+
 6893                                 ;       LOAD DSBUF2
 6894                                 ;-
 6895
 6896 035250                             LET R0 := #ADD1A+2
      035250  012700  065452                                                           MOV     #ADD1A+2,R0
 6897 035254                             LET R1 := TADDI+4 CLR.BY #10000
      035254  016701  030646                                                           MOV     TADDI+4,R1
      035260  042701  010000                                                           BIC     #10000,R1
 6898 035264                             LET R2 := #DSBUF2
      035264  012702  064502                                                           MOV     #DSBUF2,R2
 6899 035270                             INLINE      <2$:    MOVB    (R0)+,  (R2)+>
      035270  112022                                                                   2$:     MOVB    (R0)+,  (R2)
+
 6900 035272                             INLINE      <       SOB     R1,     2$>
      035272  077102                                                                           SOB     R1,     2$
 6901
 6902                                 ;+
 6903                                 ;       LOAD DS.EXP
 6904                                 ;-
 6905
 6906 035274                             LET R0 := #ADD1B+2
      035274  012700  065476                                                           MOV     #ADD1B+2,R0
 6907 035300                             LET R1 := TADDI+10 CLR.BY #10000
      035300  016701  030626                                                           MOV     TADDI+10,R1
      035304  042701  010000                                                           BIC     #10000,R1
 6908 035310                             LET R2 := #DS.EXP
      035310  012702  064606                                                           MOV     #DS.EXP,R2
 6909 035314                             INLINE      <3$:    MOVB    (R0)+,  (R2)>
      035314  112012                                                                   3$:     MOVB    (R0)+,  (R2)
 6910 035316                             INLINE      <       BISB    #60,    (R2)+>
      035316  152722  000060                                                                 BISB    #60,    (R2)
+
 6911 035322                             INLINE      <       SOB     R1,     3$>
      035322  077104                                                                         SOB     R1,     3$
 6912
 6913 035324                             CALL CLRREG                 ;ENSURE R0-R5 ARE CLR
      035324  004767  021110                                                           JSR     PC,CLRREG
 6914
 6915 035330                             INLINE <ADDNI>
      035330  076150                                                                   ADDNI
 6916 035332                             INLINE <.WORD TADDI>
      035332  066122                                                                   .WORD TADDI
 6917 035334                             INLINE <.WORD TADDI+4>
      035334  066126                                                                   .WORD TADDI+4
 6918 035336                             INLINE <.WORD TADDI+10>
      035336  066132                                                                   .WORD TADDI+10
 6919 035340                             LET CC := PSW
      035340  016767  142432  143152                                                   MOV     PSW,CC
 6920 035346                             LET CC := CC CLR.BY #177760
      035346  042767  177760  143144                                                   BIC     #177760,CC
 6921 035354                             IF CC NE #0 THEN
      035354  005767  143140                                                           TST     CC
      035360  001402                                                                   BEQ     $51461
 6922 035362                                 CALL ERROR
      035362  004767  022312                                                           JSR     PC,ERROR
 6923 035366                             ENDIF
```

```
      035366                                                                  $51461:
 6924
 6925                                ;+
 6926                                ;       ENSURE GPR'S STILL ZERO
 6927                                ;-
 6928
 6929 035366                           IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN
      035366  005700                                                                 TST     R0
      035370  001004                                                                 BNE     $51462
      035372  005701                                                                 TST     R1
      035374  001002                                                                 BNE     $51462
      035376  005702                                                                 TST     R2
      035400  001402                                                                 BEQ     $51463
      035402                                                                  $51462:
 6930 035402                               CALL ERROR
      035402  004767  022272                                                       JSR     PC,ERROR
 6931 035406                           ENDIF
      035406                                                                  $51463:
 6932 035406                           IF R3 NE #0 OR R4 NE #0 OR  R5 NE #0 THEN
      035406  005703                                                                 TST     R3
      035410  001004                                                                 BNE     $51464
      035412  005704                                                                 TST     R4
      035414  001002                                                                 BNE     $51464
      035416  005705                                                                 TST     R5
      035420  001402                                                                 BEQ     $51465
      035422                                                                  $51464:
 6933 035422                               CALL ERROR
      035422  004767  022252                                                       JSR     PC,ERROR
 6934 035426                           ENDIF
      035426                                                                  $51465:
 6935
 6936                                ;+
 6937                                ;       CHECK RESULTS
 6938                                ;-
 6939
 6940 035426                           LET R0 := #DS.DST
      035426  012700  064544                                                         MOV     #DS.DST,R0
 6941 035432                           LET R1 := #DS.EXP             ;EXPECTED DATA
      035432  012701  064606                                                         MOV     #DS.EXP,R1
 6942 035436                           LET ERRFLG := #0
      035436  005067  143014                                                         CLR     ERRFLG
 6943 035442                           WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO
      035442                                                                  $51466:
      035442  020027  064604                                                         CMP     R0,#DS.DST+40
      035446  001410                                                                 BEQ     $51467
      035450  005767  143002                                                       TST     ERRFLG
      035454  001005                                                                 BNE     $51467
 6944 035456                               IFB (R0)+ NE (R1)+ THEN
      035456  122021                                                               CMPB    (R0)+,(R1)+
      035460  001402                                                                 BEQ     $51470
 6945 035462                                   CALL ERROR
      035462  004767  022212                                                       JSR     PC,ERROR
 6946 035466                               ENDIF
      035466                                                                  $51470:
 6947 035466                           ENDDO
      035466  000765                                                                 BR      $51466
      035470                                                                  $51467:
```

```
ADDN

   6948
   6949
```

```
                                .SBTTL  SUBN
                                ;*****************************************************************
                                ;
                                ;       SUBN TEST
                                ;
                                ;*****************************************************************

                                ROUTINE XSUBN
                                                                                        XSUBN:
                                   INLINE <SCOPE>
                                                                                                SCOPE
                                   INLINE <NOP>
                                                                                                NOP
                                   LET $TIMES := #10            ;ITERATION COUNT
                                                                                                MOV     #10,$TIMES
                                   LET COUNT := TADD            ;# OF TEST CASES
                                                                                                MOV     TADD,COUNT
                                   LET TSTCAS := #TADD+2                ;POINT TO FIRST TEST CASE
                                                                                                MOV     #TADD+2,TSTCAS

                                ;+
                                ;       CLR R0.EXT TO R3.EXP
                                ;-

                                   INCR  R0 FROM #0 TO #6 BY #2
                                                                                                CLR     R0
                                                                                                BR      $51473
                                                                                        $51474:
                                                                                                ADD     #2,R0
                                                                                        $51473:
                                                                                                CMP     R0,#6
                                                                                                BGT     $51475
                                       INLINE <CLR R0.EXP(R0)>
                                                                                                CLR R0.EXP(R0)
                                   ENDINC
                                                                                                BR      $51474
                                                                                        $51475:

                                ;REPEAT OUTER LOOP FOR TEST CASES
                                   INLINE <$5051:>
                                                                                        $5051:
                                   CALL CLRDSB                  ;CLR DS BUFFERS
                                                                                                JSR     PC,CLRDSB

                                ;+
                                ;       LOAD DATAXX AND CC.EXP FROM TEST CASE
                                ;-

                                   LET R0 := TSTCAS
                                                                                                MOV     TSTCAS,R0
                                   LET DATA00 := (R0)+          ;SRC1 LENGTH
                                                                                                MOV     (R0)+,DATA00
                                   LET DATA01 := (R0)+          ;POINTER TO SRC1 DATA
                                                                                                MOV     (R0)+,DATA01
                                   LET DATA02 := (R0)+          ;SRC2 LENGTH
                                                                                                MOV     (R0)+,DATA02
                                   LET DATA03 := (R0)+          ;POINTER TO SRC2 DATA
                                                                                                MOV     (R0)+,DATA03
```

SUBN

```
 6985 035572
      035572  012067  142774                LET DATA04 := (R0)+          ;DEST LENGTH
                                                                          MOV     (R0)+,DATA04
 6986 035576
      035576  012067  142772                LET DATA05 := (R0)+          ;POINTER TO DS.EXP DATA (SUM)
                                                                          MOV     (R0)+,DATA05
 6987 035602
      035602  012067  142714                LET CC.EXP := (R0)+          ;CC.EXP (SUM)
                                                                          MOV     (R0)+,CC.EXP
 6988 035606
      035606  012067  142762                LET DATA05 := (R0)+          ;POINTER TO DS.EXP DATA (DIFF)
                                                                          MOV     (R0)+,DATA05
 6989 035612
      035612  012067  142704                LET CC.EXP := (R0)+          ;CC.EXP (DIFF)
                                                                          MOV     (R0)+,CC.EXP
 6990
 6991                                  ;+
 6992                                  ;       LOAD DSBUF1
 6993                                  ;-
 6994
 6995 035616
      035616  005767  142740                IF DATA00 NE #0 THEN         ;CHECK FOR ZERO LENGTH BUFFER
                                                                          TST     DATA00
      035622  001413                                                      BEQ     $51476
 6996 035624
      035624  016700  142734                    LET R0 := DATA01
                                                                          MOV     DATA01,R0
 6997 035630
      035630  012067  142614                    LET SIGN0 := (R0)+       ;STORE SIGN AND POINT AT FIRST BYTE
                                                                          MOV     (R0)+,SIGN0
 6998 035634
      035634  016701  142722                    LET R1 := DATA00         ;# OF BYTES TO TRANSFER
                                                                          MOV     DATA00,R1
 6999 035640
      035640  012702  064440                    LET R2 := #DSBUF1        ;TRANSFER DESTINATION
                                                                          MOV     #DSBUF1,R2
 7000 035644
      035644                                    INLINE <1$:>
                                                                          1$:
 7001 035644
      035644  112022                            INLINE <MOVB (R0)+,(R2)+>
                                                                          MOVB (R0)+,(R2)+
 7002 035646
      035646  077102                            INLINE <SOB R1,1$>
                                                                          SOB R1,1$
 7003 035650                                ELSE
      035650  000402                                                      BR      $51477
      035652                                                        $51476:
 7004 035652
      035652  005067  142572                    LET SIGN0 := #0
                                                                          CLR     SIGN0
 7005 035656                                ENDIF
      035656                                                        $51477:
 7006
 7007                                  ;+
 7008                                  ;       LOAD DSBUF2
 7009                                  ;-
 7010
 7011 035656
      035656  005767  142704                IF DATA02 NE #0 THEN
                                                                          TST     DATA02
      035662  001413                                                      BEQ     $51500
 7012 035664
      035664  016700  142700                    LET R0 := DATA03
                                                                          MOV     DATA03,R0
 7013 035670
      035670  012067  142556                    LET SIGN1 := (R0)+
                                                                          MOV     (R0)+,SIGN1
 7014 035674
      035674  016701  142666                    LET R1 := DATA02
                                                                          MOV     DATA02,R1
 7015 035700
      035700  012702  064502                    LET R2 := #DSBUF2
                                                                          MOV     #DSBUF2,R2
 7016 035704
      035704                                    INLINE <2$:>
                                                                          2$:
```

```
                                    INLINE <MOVB (R0)+,(R2)+>                       MOVB (R0)+,(R2)+
                                    INLINE <SOB R1,2$>                              SOB R1,2$
                                ELSE                                                BR      $51501
                                                                            $51500:
                                    LET SIGN1 := #0                                 CLR     SIGN1
                                ENDIF                                       $51501:

                            ;+
                            ;       LOAD DS.EXP
                            ;-

                                IF DATA04 NE #0 THEN                                TST     DATA04
                                                                                    BEQ     $51502
                                    LET R0 := DATA05                                MOV     DATA05,R0
                                    LET SIGN2 := (R0)+                              MOV     (R0)+,SIGN2
                                    LET R1 := DATA04                                MOV     DATA04,R1
                                    LET R2 := #DS.EXP                               MOV     #DS.EXP,R2
                                    INLINE <3$:>                                    3$:
                                    INLINE <MOVB (R0)+,(R2)>                        MOVB (R0)+,(R2)
                                    INLINE <BISB     #60,    (R2)+>                 BISB    #60,    (R2)+
                                    INLINE <SOB R1,3$>                              SOB R1,3$
                                ELSE                                                BR      $51503
                                                                            $51502:
                                    LET SIGN2 := #0                                 CLR     SIGN2
                                ENDIF                                       $51503:

                            ;+
                            ;       LOAD DSTMP'S WITH MDS & LSD OF EACH BUFFER
                            ;-

                                LET DSTMP0 :B= DSBUF1                               MOVB    DSBUF1,DSTMP0
                                LET TEMP0 := #DSBUF1-1 + DATA00                     MOV     #DSBUF1-1,TEMP0
                                                                                    ADD     DATA00,TEMP0
                                LET DSTMP1 :B= @TEMP0                               MOVB    @TEMP0,DSTMP1
                                LET DSTMP2 :B= DSBUF2                               MOVB    DSBUF2,DSTMP2
                                LET TEMP0 := #DSBUF2-1 + DATA02
```

```
                                                                                                    MOV     #DSBUF2-1,TEMP0
                                                                                                    ADD     DATA02,TEMP0
 7049 036034                    LET DSTMP3 :B= @TEMP0                                               MOVB    @TEMP0,DSTMP3
 7050 036042                    LET DSTMP4 :B= DS.EXP                                               MOVB    DS.EXP,DSTMP4
 7051 036050                    LET TEMP0 := #DS.EXP-1 + DATA04                                     MOV     #DS.EXP-1,TEMP0
                                                                                                    ADD     DATA04,TEMP0
 7052 036064                    LET DSTMP5 :B= @TEMP0                                               MOVB    @TEMP0,DSTMP5
 7053 036072                    LET DESCR1 := #DSBUF1                                               MOV     #DSBUF1,DESCR1
 7054 036100                    LET DESCR3 := #DSBUF2                                               MOV     #DSBUF2,DESCR3
 7055 036106                    LET DESCR5 := #DS.DST                                               MOV     #DS.DST,DESCR5
 7056
 7057                         ; INCR TEMP0 FROM #1 TO #5 BY #1        SRC1 DATA TYPE LOOP
 7058 036114                     INLINE <CLR @#TEMP0>                                               CLR @#TEMP0
 7059 036120                    INLINE <INC21: INC TEMP0>                                           INC21: INC TEMP0
 7060 036124                    IF TEMP0 GT #5 THEN                                                 CMP     TEMP0,#5
                                                                                                    BLE     $51504
 7061 036134                        INLINE <JMP END21>                                              JMP END21
 7062 036140                    ENDIF
                                                                                          $51504:
 7063
 7064                           ;+
 7065                           ;    RESTORE MSD & LSD OF DSBUF1
 7066                           ;-
 7067
 7068 036140                      LET DSBUF1 :B= DSTMP0          ;MSD                                MOVB    DSTMP0,DSBUF1
 7069 036146                      LET TEMP2 := #DSBUF1-1 + DATA00                                   MOV     #DSBUF1-1,TEMP2
                                                                                                    ADD     DATA00,TEMP2
 7070 036162                      LET @TEMP2 :B= DSTMP1          ;LSD                              MOVB    DSTMP1,@TEMP2
 7071
 7072                           ;+
 7073                           ;    SIGN DSBUF1 BY DATA TYPE
 7074                           ;-
 7075
 7076 036170                      SELECT TEMP0 OF 5                                                 MOV     TEMP0,-(SP)
      036170  016746  142302                                                                        ASL     (SP)
      036174  006316                                                                               ADD     PC,(SP)
      036176  060716                                                                               ADD     @(SP)+,PC
      036200  063607                                                                       $51505:
      036202                                                                                        .WORD   $51513-$51505
      036202  000012                                                                               .WORD   $51512-$51505
      036204  000052                                                                               .WORD   $51511-$51505
      036206  000116                                                                               .WORD   $51510-$51505
      036210  000162
```

```
     036212  000226                                                                          .WORD    $51507-$51505
     036214                                 CASE 1
     036214                                                                        $51513:
     036214                                      LET DESCR0 := DATA00         ;SIGNED ZONED
     036214  016767  142342  142374                                                          MOV      DATA00,DESCR0
     036222                                      IF SIGN0 EQ #0 THEN
     036222  005767  142222                                                                  TST      SIGN0
     036226  001004                                                                          BNE      $51514
     036230                                           LET @TEMP2 :B= @TEMP2 SET.BY #60 ;+
     036230  152777  000060  142244                                                          BISB     #60,@TEMP2
     036236                                      ELSE
     036236  000403                                                                          BR       $51515
     036240                                                                        $51514:
     036240                                           LET @TEMP2 :B= @TEMP2 SET.BY #160   ;-
     036240  152777  000160  142234                                                          BISB     #160,@TEMP2
     036246                                      ENDIF
     036246                                                                        $51515:
     036246                                      LET DSBUF1-1 :B= #0          ;FIX LEADING SEPARATE RESULT
     036246  105067  026165                                                                  CLRB     DSBUF1-1

     036252                                 CASE 2
     036252  000512                                                                          BR       $51506
     036254                                                                        $51512:
     036254                                      LET DESCR0 := DATA00 SET.BY #20000   ;TRAILING OVERPUNCH
     036254  016767  142302  142334                                                          MOV      DATA00,DESCR0
     036262  052767  020000  142326                                                          BIS      #20000,DESCR0
     036270                                      LET R5 :B= DSTMP1    ;GET LSD
     036270  116705  142261                                                                  MOVB     DSTMP1,R5
     036274                                      IF SIGN0 EQ #-1 THEN
     036274  026727  142150  177777                                                          CMP      SIGN0,#-1
     036302  001002                                                                          BNE      $51516
     036304                                           LET R5 := R5 + #10.        ;CHANGE INDEX FOR -
     036304  062705  000012                                                                  ADD      #10.,R5
     036310                                      ENDIF
     036310                                                                        $51516:
     036310                                      LET @TEMP2 :B= OVPNCH(R5)
     036310  116577  056330  142164                                                          MOVB     OVPNCH(R5),@TEMP2

     036316                                 CASE 3
     036316  000470                                                                          BR       $51506
     036320                                                                        $51511:
     036320                                      LET DESCR0 := DATA00 SET.BY #30000   ;LEADING OVERPUNCH
     036320  016767  142236  142270                                                          MOV      DATA00,DESCR0
     036326  052767  030000  142262                                                          BIS      #30000,DESCR0
     036334                                      LET R5 :B= DSTMP0    ;GET MSD
     036334  116705  142214                                                                  MOVB     DSTMP0,R5
     036340                                      IF SIGN0 EQ #-1 THEN
     036340  026727  142104  177777                                                          CMP      SIGN0,#-1
     036346  001002                                                                          BNE      $51517
     036350                                           LET R5 := R5 + #10.
     036350  062705  000012                                                                  ADD      #10.,R5
     036354                                      ENDIF
     036354                                                                        $51517:
     036354                                      LET DSBUF1 :B= OVPNCH(R5)
     036354  116567  056330  026056                                                          MOVB     OVPNCH(R5),DSBUF1
```

SUBN

```
                                        CASE 4
                                            LET DESCRO := DATA00 SET.BY #40000   ;TRAILING SEPARATE
                                            LET R5 := TEMP2 + #1
                                            IF SIGNO EQ #0 THEN
                                                LET (R5) :B= #53        ;+ SIGN
                                            ELSE
                                                LET (R5) :B= #55        ;- SIGN
                                            ENDIF

                                        CASE 5
                                            LET DESCRO := DATA00 SET.BY #50000   ;LEADING SEPARATE
                                            LET R5 := TEMP2 + #1
                                            LET (R5) :B= #0  ;RESTORE TRAILING SEPARATE
                                            IF SIGNO EQ #0 THEN
                                                LET DSBUF1-1 :B= #53
                                            ELSE
                                                LET DSBUF1-1 :B= #55
                                            ENDIF
                                        ENDSELECT

                                    ;INCR TEMP1 FROM #1 TO #5 BY #1       SRC2 DATA TYPES
                                        INLINE <CLR @#TEMP1>
                                        INLINE <INC22: INC TEMP1>
                                        IF TEMP1 GT #5 THEN
                                            INLINE <JMP END22>
```

```
                                                    BR      $51506
                                                $51510:
                                                    MOV     DATA00,DESCRO
                                                    BIS     #40000,DESCRO
                                                    MOV     TEMP2,R5
                                                    INC     R5
                                                    TST     SIGNO
                                                    BNE     $51520
                                                    MOVB    #53,(R5)
                                                    BR      $51521
                                                $51520:
                                                    MOVB    #55,(R5)
                                                $51521:

                                                    BR      $51506
                                                $51507:
                                                    MOV     DATA00,DESCRO
                                                    BIS     #50000,DESCRO
                                                    MOV     TEMP2,R5
                                                    INC     R5
                                                    CLRB    (R5)
                                                    TST     SIGNO
                                                    BNE     $51522
                                                    MOVB    #53,DSBUF1-1
                                                    BR      $51523
                                                $51522:
                                                    MOVB    #55,DSBUF1-1
                                                $51523:
                                                $51506:

                                                    CLR @#TEMP1
                                                    INC22: INC TEMP1
                                                    CMP     TEMP1,#5
                                                    BLE     $51524
```

```
     036520  000167  001236                                                       JMP END22
7128 036524                          ENDIF
     036524                                                               $51524:
7129 036524                          LET R3 := #DSBUF2
     036524  012703  064502                                                       MOV     #DSBUF2,R3
7130
7131                                ;+
7132                                ;  RESTORE MSD & LSD OF DSBUF2
7133                                ;-
7134
7135 036530                          LET DSBUF2 :B= DSTMP2
     036530  116767  142022  025744                                               MOVB    DSTMP2,DSBUF2
7136 036536                          LET TEMP3 := #DSBUF2-1 + DATA02
     036536  012767  064501  141740                                               MOV     #DSBUF2-1,TEMP3
     036544  066767  142016  141732                                               ADD     DATA02,TEMP3
7137 036552                          LET @TEMP3 :B= DSTMP3
     036552  116777  142001  141724                                               MOVB    DSTMP3,@TEMP3
7138
7139                                ;+
7140                                ;  SIGN DSBUF2 BY DATA TYPE
7141                                ;-
7142
7143 036560                          SELECT TEMP1 OF 5
     036560  016746  141714                                                       MOV     TEMP1,-(SP)
     036564  006316                                                               ASL     (SP)
     036566  060716                                                               ADD     PC,(SP)
     036570  063607                                                               ADD     @(SP)+,PC
     036572                                                               $51525:
     036572  000012                                                               .WORD   $51533-$51525
     036574  000052                                                               .WORD   $51532-$51525
     036576  000116                                                               .WORD   $51531-$51525
     036600  000162                                                               .WORD   $51530-$51525
     036602  000226                                                               .WORD   $51527-$51525
7144 036604                             CASE 1
     036604                                                               $51533:
7145 036604                                LET DESCR2 := DATA02
     036604  016767  141756  142010                                               MOV     DATA02,DESCR2
7146 036612                                IF SIGN1 EQ #0 THEN
     036612  005767  141634                                                       TST     SIGN1
     036616  001004                                                               BNE     $51534
7147 036620                                   LET @TEMP3 :B= @TEMP3 SET.BY #60  ;+
     036620  152777  000060  141656                                               BISB    #60,@TEMP3
7148 036626                                ELSE
     036626  000403                                                               BR      $51535
     036630                                                               $51534:
7149 036630                                   LET @TEMP3 :B= @TEMP3 SET.BY #160;-
     036630  152777  000160  141646                                               BISB    #160,@TEMP3
7150 036636                                ENDIF
     036636                                                               $51535:
7151 036636                                LET DSBUF2-1 :B= #0
     036636  105067  025637                                                       CLRB    DSBUF2-1
7152
7153 036642                             CASE 2
     036642  000512                                                               BR      $51526
     036644                                                               $51532:
7154 036644                                LET DESCR2 := DATA02 SET.BY #20000   ;TRAILING OVERPUNCH
     036644  016767  141716  141750                                               MOV     DATA02,DESCR2
```

SUBN

```
        036652  052767  020000  141742                                                          BIS     #20000,DESCR2
   7155 036660                          LET R5 :B= DSTMP3                                       MOVB    DSTMP3,R5
        036660  116705  141673
   7156 036664                          IF SIGN1 EQ #-1 THEN                                    CMP     SIGN1,#-1
        036664  026727  141562  177777                                                          BNE     $51536
        036672  001002
   7157 036674                              LET R5 := R5 + #10.                                 ADD     #10.,R5
        036674  062705  000012
   7158 036700                          ENDIF
        036700                                                                          $51536:
   7159 036700                          LET @TEMP3 :B= OVPNCH(R5)
        036700  116577  056330  141576                                                          MOVB    OVPNCH(R5),@TEMP3
   7160
   7161 036706                      CASE 3                                                      BR      $51526
        036706  000470
        036710                                                                          $51531:
   7162 036710                          LET DESCR2 := DATA02 SET.BY #30000   ;LEADING OVERPUNCH
        036710  016767  141652  141704                                                          MOV     DATA02,DESCR2
        036716  052767  030000  141676                                                          BIS     #30000,DESCR2
   7163 036724                          LET R5 :B= DSTMP2                                       MOVB    DSTMP2,R5
        036724  116705  141626
   7164 036730                          IF SIGN1 EQ #-1 THEN                                    CMP     SIGN1,#-1
        036730  026727  141516  177777                                                          BNE     $51537
        036736  001002
   7165 036740                              LET R5 := R5 + #10.                                 ADD     #10.,R5
        036740  062705  000012
   7166 036744                          ENDIF
        036744                                                                          $51537:
   7167 036744                          LET DSBUF2 :B= OVPNCH(R5)
        036744  116567  056330  025530                                                          MOVB    OVPNCH(R5),DSBUF2
   7168
   7169 036752                      CASE 4                                                      BR      $51526
        036752  000446
        036754                                                                          $51530:
   7170 036754                          LET DESCR2 := DATA02 SET.BY #40000   ;TRAILING SEPARATE
        036754  016767  141606  141640                                                          MOV     DATA02,DESCR2
        036762  052767  040000  141632                                                          BIS     #40000,DESCR2
   7171 036770                          LET R5 := TEMP3 + #1                                    MOV     TEMP3,R5
        036770  016705  141510                                                                  INC     R5
        036774  005205
   7172 036776                          IF SIGN1 EQ #0 THEN                                     TST     SIGN1
        036776  005767  141450                                                                  BNE     $51540
        037002  001003
   7173 037004                              LET (R5) :B= #53                                    MOVB    #53,(R5)
        037004  112715  000053
   7174 037010                          ELSE                                                    BR      $51541
        037010  000402
        037012                                                                          $51540:
   7175 037012                              LET (R5) :B= #55                                    MOVB    #55,(R5)
        037012  112715  000055
   7176 037016                          ENDIF
        037016                                                                          $51541:
   7177
   7178 037016                      CASE 5                                                      BR      $51526
        037016  000424
        037020                                                                          $51527:
   7179 037020                          LET DESCR2 := DATA02 SET.BY #50000   ;LEADING SEPARATE
```

SUBN

```
        037020  016767  141542  141574                                                  MOV     DATA02,DESCR2
        037026  052767  050000  141566                                                  BIS     #50000,DESCR2
   7180 037034                                  LET R5 := TEMP3 + #1
        037034  016705  141444                                                          MOV     TEMP3,R5
        037040  005205                                                                  INC     R5
   7181 037042                                  LET (R5) :B= #0
        037042  105015                                                                  CLRB    (R5)
   7182 037044                                  IF SIGN1 EQ #0 THEN
        037044  005767  141402                                                          TST     SIGN1
        037050  001004                                                                  BNE     $51542
   7183 037052                                      LET DSBUF2-1 :B= #53
        037052  112767  000053  025421                                                  MOVB    #53,DSBUF2-1
   7184 037060                                  ELSE
        037060  000403                                                                  BR      $51543
        037062                                                                  $51542:
   7185 037062                                      LET DSBUF2-1 :B= #55
        037062  112767  000055  025411                                                  MOVB    #55,DSBUF2-1
   7186 037070                                  ENDIF
        037070                                                                  $51543:
   7187 037070                              ENDSELECT                          $51526:
        037070
   7188
   7189                                 ;INCR DATTYP FROM #1 TO #5 BY #1    ;DS.EXP LOOP
   7190 037070                              INLINE <CLR @#DATTYP>
        037070  005037  000464                                                          CLR @#DATTYP
   7191 037074                              INLINE <INC23: INC DATTYP>
        037074  005267  141364                                                          INC23: INC DATTYP
   7192 037100                              IF DATTYP GT #5 THEN
        037100  026727  141360  000005                                                  CMP     DATTYP,#5
        037106  003402                                                                  BLE     $51544
   7193 037110                                  INLINE <JMP END23>
        037110  000167  000642                                                          JMP END23
   7194 037114                              ENDIF
        037114                                                                  $51544:
   7195
   7196                                 ;+
   7197                                 ;       RESTORE MSD & LSD OF DS.EXP
   7198                                 ;-
   7199
   7200 037114                              LET DS.EXP :B= DSTMP4    ;MSD
        037114  116767  141440  025464                                                  MOVB    DSTMP4,DS.EXP
   7201 037122                              LET TEMP4 := #DS.EXP-1 + DATA04
        037122  012767  064605  141356                                                  MOV     #DS.EXP-1,TEMP4
        037130  066767  141436  141350                                                  ADD     DATA04,TEMP4
   7202 037136                              LET @TEMP4 :B= DSTMP5    ;LSD
        037136  116777  141417  141342                                                  MOVB    DSTMP5,@TEMP4
   7203
   7204                                 ;+
   7205                                 ;       SIGN DS.EXP BY DATA TYPE
   7206                                 ;-
   7207
   7208 037144                              SELECT DATTYP OF 5
        037144  016746  141314                                                          MOV     DATTYP,-(SP)
        037150  006316                                                                  ASL     (SP)
        037152  060716                                                                  ADD     PC,(SP)
        037154  063607                                                                  ADD     @(SP)+,PC
        037156                                                                  $51545:
```

```
        037156  000012                                                                  .WORD   $51553-$51545
        037160  000060                                                                  .WORD   $51552-$51545
        037162  000136                                                                  .WORD   $51551-$51545
        037164  000214                                                                  .WORD   $51550-$51545
        037166  000260                                                                  .WORD   $51547-$51545
   7209 037170                          CASE 1
        037170                                                                  $51553:
   7210 037170                              LET DESCR4 := DATA04          ;SIGNED ZONED
        037170  016767  141376  141430                                                  MOV     DATA04,DESCR4
   7211 037176                              IF DATA04 NE #0 THEN
        037176  005767  141370                                                          TST     DATA04
        037202  001414                                                                  BEQ     $51554
   7212 037204                                  IF SIGN2 EQ #0 THEN
        037204  005767  141244                                                          TST     SIGN2
        037210  001004                                                                  BNE     $51555
   7213 037212                                      LET @TEMP4 :B= @TEMP4 SET.BY #60
        037212  152777  000060  141266                                                  BISB    #60,@TEMP4
   7214 037220                                  ELSE
        037220  000403                                                                  BR      $51556
        037222                                                                  $51555:
   7215 037222                                      LET @TEMP4 :B= @TEMP4 SET.BY #160
        037222  152777  000160  141256                                                  BISB    #160,@TEMP4
   7216 037230                                  ENDIF
        037230                                                                  $51556:
   7217 037230                                  LET DS.EXP-1 :B= #0
        037230  105067  025351                                                          CLRB    DS.EXP-1
   7218 037234                              ENDIF
        037234                                                                  $51554:
   7219
   7220 037234                          CASE 2
        037234  000524                                                                  BR      $51546
        037236                                                                  $51552:
   7221 037236                              LET DESCR4 := DATA04 SET.BY #20000   ;TRAILING OVERPUNCH
        037236  016767  141330  141362                                                  MOV     DATA04,DESCR4
        037244  052767  020000  141354                                                  BIS     #20000,DESCR4
   7222 037252                              IF DATA04 NE #0 THEN
        037252  005767  141314                                                          TST     DATA04
        037256  001415                                                                  BEQ     $51557
   7223 037260                                  LET R5 :B= DSTMP5
        037260  116705  141275                                                          MOVB    DSTMP5,R5
   7224 037264                                  LET R5 := R5 - #60
        037264  162705  000060                                                          SUB     #60,R5
   7225 037270                                  IF SIGN2 EQ #-1 THEN
        037270  026727  141160  177777                                                  CMP     SIGN2,#-1
        037276  001002                                                                  BNE     $51560
   7226 037300                                      LET R5 := R5 + #10.
        037300  062705  000012                                                          ADD     #10.,R5
   7227 037304                                  ENDIF
        037304                                                                  $51560:
   7228 037304                                  LET @TEMP4 :B= OVPNCH(R5
        037304  116577  056330  141174                                                  MOVB    OVPNCH(R5),@TEMP4
   7229 037312                              ENDIF
        037312                                                                  $51557:
   7230
   7231 037312                          CASE 3
        037312  000475                                                                  BR      $51546
        037314                                                                  $51551:
```

```
 7232 037314                              LET DESCR4 := DATA04 SET.BY #30000   ;LEADING OVERPUNCH
      037314  016767  141252  141304                                                    MOV     DATA04,DESCR4
      037322  052767  030000  141276                                                    BIS     #30000,DESCR4
 7233 037330                              IF DATA04 NE #0 THEN
      037330  005767  141236                                                            TST     DATA04
      037334  001415                                                                    BEQ     $51561
 7234 037336                                  LET R5 :B= DSTMP4
      037336  116705  141216                                                            MOVB    DSTMP4,R5
 7235 037342                                  LET R5 := R5 - #60
      037342  162705  000060                                                            SUB     #60,R5
 7236 037346                                  IF SIGN2 EQ #-1 THEN
      037346  026727  141102  177777                                                    CMP     SIGN2,#-1
      037354  001002                                                                    BNE     $51562
 7237 037356                                      LET R5 := R5 + #10.
      037356  062705  000012                                                            ADD     #10.,R5
 7238 037362                                  ENDIF
      037362                                                                  $51562:
 7239 037362                                  LET DS.EXP :B= OVPNCH(R5)
      037362  116567  056330  025216                                                    MOVB    OVPNCH(R5),DS.EXP
 7240 037370                              ENDIF
      037370                                                                  $51561:
 7241
 7242 037370                          CASE 4
      037370  000446                                                                    BR      $51546
      037372                                                                  $51550:
 7243 037372                              LET DESCR4 := DATA04 SET.BY #40000   ;TRAILING SEPARATE
      037372  016767  141174  141226                                                    MOV     DATA04,DESCR4
      037400  052767  040000  141220                                                    BIS     #40000,DESCR4
 7244 037406                              LET R5 := TEMP4 + #1
      037406  016705  141074                                                            MOV     TEMP4,R5
      037412  005205                                                                    INC     R5
 7245 037414                              IF SIGN2 EQ #0 THEN
      037414  005767  141034                                                            TST     SIGN2
      037420  001003                                                                    BNE     $51563
 7246 037422                                  LET (R5) :B= #53
      037422  112715  000053                                                            MOVB    #53,(R5)
 7247 037426                              ELSE
      037426  000402                                                                    BR      $51564
      037430                                                                  $51563:
 7248 037430                                  LET (R5) :B= #55
      037430  112715  000055                                                            MOVB    #55,(R5)
 7249 037434                              ENDIF
      037434                                                                  $51564:
 7250
 7251 037434                          CASE 5
      037434  000424                                                                    BR      $51546
      037436                                                                  $51547:
 7252 037436                              LET DESCR4 := DATA04 SET.BY #50000
      037436  016767  141130  141162                                                    MOV     DATA04,DESCR4
      037444  052767  050000  141154                                                    BIS     #50000,DESCR4
 7253 037452                              LET R5 := TEMP4 + #1
      037452  016705  141030                                                            MOV     TEMP4,R5
      037456  005205                                                                    INC     R5
 7254 037460                              LET (R5) :B= #0
      037460  105015                                                                    CLRB    (R5)
 7255 037462                              IF SIGN2 EQ #0 THEN
      037462  005767  140766                                                            TST     SIGN2
```

```
SUBN

      037466  001004                                                                    BNE     $51565
 7256 037470                                  LET DS.EXP-1 :B= #53
      037470  112767  000053  025107                                                   MOVB    #53,DS.EXP-1
 7257 037476                                ELSE
      037476  000403                                                                    BR      $51566
      037500                                                                  $51565:
 7258 037500                                  LET DS.EXP-1 :B= #55
      037500  112767  000055  025077                                                   MOVB    #55,DS.EXP-1
 7259 037506                                ENDIF
      037506                                                                  $51566:
 7260 037506                            ENDSELECT
      037506                                                                  $51546:
 7261
 7262                                   ;+
 7263                                   ;       CLR DS.DST & LOAD DESCRIPTORS
 7264                                   ;-
 7265
 7266 037506                               LET R0 := #DS.DST-1
      037506  012700  064543                                                           MOV     #DS.DST-1,R0
 7267 037512                               LET R1 := #41
      037512  012701  000041                                                           MOV     #41,R1
 7268 037516                               INLINE <DOIT:        CLRB    (R0)+>
      037516  105020                                                           DOIT:   CLRB    (R0)+
 7269 037520                               INLINE <SOB  R1,     DOIT>
      037520  077102                                                                   SOB     R1,     DOIT
 7270 037522                               LET R0 := DESCR0
      037522  016700  141070                                                           MOV     DESCR0,R0
 7271 037526                               LET R1 := DESCR1
      037526  016701  141066                                                           MOV     DESCR1,R1
 7272 037532                               LET R2 := DESCR2
      037532  016702  141064                                                           MOV     DESCR2,R2
 7273 037536                               LET R3 := DESCR3
      037536  016703  141062                                                           MOV     DESCR3,R3
 7274 037542                               LET R4 := DESCR4
      037542  016704  141060                                                           MOV     DESCR4,R4
 7275 037546                               LET R5 := DESCR5
      037546  016705  141056                                                           MOV     DESCR5,R5
 7276 037552                               LET R4.EXP := R4
      037552  010467  140756                                                           MOV     R4,R4.EXP
 7277 037556                               LET R5.EXP := R5
      037556  010567  140754                                                           MOV     R5,R5.EXP
 7278 037562                               INLINE <SUBN>
      037562  076051                                                                   SUBN
 7279
 7280                                   ;+
 7281                                   ; CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.
 7282                                   ;-
 7283
 7284 037564                               LET CC := PSW
      037564  016767  140206  140726                                                   MOV     PSW,CC
 7285 037572                               LET CC := CC CLR.BY #177760
      037572  042767  177760  140720                                                   BIC     #177760,CC
 7286 037600                               IF CC NE CC.EXP THEN
      037600  026767  140714  140714                                                   CMP     CC,CC.EXP
      037606  001402                                                                   BEQ     $51567
 7287 037610                                   CALL ERROR
      037610  004767  020064                                                           JSR     PC,ERROR
```

```
 7288 037614                            ENDIF
      037614                                                        $51567:
 7289 037614                            IF R0 NE R0.EXP THEN
      037614  020067  140704                                                CMP     R0,R0.EXP
      037620  001402                                                       BEQ     $51570
 7290 037622                                CALL ERROR
      037622  004767  020052                                                JSR     PC,ERROR
 7291 037626                            ENDIF
      037626                                                        $51570:
 7292 037626                            IF R1 NE R1.EXP THEN
      037626  020167  140674                                                CMP     R1,R1.EXP
      037632  001402                                                       BEQ     $51571
 7293 037634                                CALL ERROR
      037634  004767  020040                                                JSR     PC,ERROR
 7294 037640                            ENDIF
      037640                                                        $51571:
 7295 037640                            IF R2 NE R2.EXP THEN
      037640  020267  140664                                                CMP     R2,R2.EXP
      037644  001402                                                       BEQ     $51572
 7296 037646                                CALL ERROR
      037646  004767  020026                                                JSR     PC,ERROR
 7297 037652                            ENDIF
      037652                                                        $51572:
 7298 037652                            IF R3 NE R3.EXP THEN
      037652  020367  140654                                                CMP     R3,R3.EXP
      037656  001402                                                       BEQ     $51573
 7299 037660                                CALL ERROR
      037660  004767  020014                                                JSR     PC,ERROR
 7300 037664                            ENDIF
      037664                                                        $51573:
 7301 037664                            IF R4 NE R4.EXP THEN
      037664  020467  140644                                                CMP     R4,R4.EXP
      037670  001402                                                       BEQ     $51574
 7302 037672                                CALL ERROR
      037672  004767  020002                                                JSR     PC,ERROR
 7303 037676                            ENDIF
      037676                                                        $51574:
 7304 037676                            IF R5 NE R5.EXP THEN
      037676  020567  140634                                                CMP     R5,R5.EXP
      037702  001402                                                       BEQ     $51575
 7305 037704                                CALL ERROR
      037704  004767  017770                                                JSR     PC,ERROR
 7306 037710                            ENDIF
      037710                                                        $51575:
 7307
 7308
 7309                                ;+
 7310                                ;       CHECK RESULTS VS EXPECTED RESULTS
 7311                                ;-
 7312
 7313 037710                            LET R0 := #DS.DST-1
      037710  012700  064543                                                MOV     #DS.DST-1,R0
 7314 037714                            LET R1 := #DS.EXP-1
      037714  012701  064605                                                MOV     #DS.EXP-1,R1
 7315 037720                            LET ERRFLG := #0
      037720  005067  140532                                                CLR     ERRFLG
 7316 037724                            WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO
```

```
                037724
                037724  020027  064604                                                  $51576:
                037730  001410                                                                  CMP     R0,@DS.DST+40
                037732  005767  140520                                                          BEQ     $51577
                037736  001005                                                                  TST     ERRFLG
        7317    037740                          IFB (R0)+ NE (R1)+ THEN                         BNE     $51577
                037740  122021                                                                  CMPB    (R0)+,(R1)+
                037742  001402                                                                  BEQ     $51600
        7318    037744                              CALL ERROR
                037744  004767  017730                                                        JSR     PC,ERROR
        7319    037750                          ENDIF
                037750                                                                  $51600:
        7320    037750                      ENDDO
                037750  000765                                                                  BR      $51576
                037752                                                                  $51577:
        7321                            ;ENDINC
        7322    037752                      INLINE <JMP INC23>
                037752  000167  177116                                                        JMP INC23
        7323    037756                      INLINE <END23:>
                037756                                                                  END23:
        7324                            ;ENDINC
        7325    037756                      INLINE <JMP INC22>
                037756  000167  176522                                                        JMP INC22
        7326    037762                      INLINE <END22:>
                037762                                                                  END22:
        7327                            ;ENDINC
        7328    037762                      INLINE <JMP INC21>
                037762  000167  176132                                                        JMP INC21
        7329    037766                      INLINE <END21:>
                037766                                                                  END21:
        7330    037766                      LET TSTCAS := TSTCAS + #22
                037766  062767  000022  140550                                              ADD     #22,TSTCAS
        7331    037774                      LET COUNT := COUNT - #1
                037774  005367  140470                                                      DEC     COUNT
        7332                            ;UNTIL COUNT EQ #0
        7333    040000                      IF COUNT NE #0 THEN
                040000  005767  140464                                                      TST     COUNT
                040004  001402                                                              BEQ     $51601
        7334    040006                          INLINE <JMP $5051>
                040006  000167  175530                                                      JMP $5051
        7335    040012                      ENDIF
                040012                                                                  $51601:
        7336
        7337                            ;+
        7338                            ;       INLINE CASE - SUBNI, USING UNSIGNED ZONED DATA
        7339                            ;-
        7340
        7341    040012                      CALL CLRDSB
                040012  004767  016656                                                      JSR     PC,CLRDSB
        7342
        7343                            ;+
        7344                            ;       LOAD DSBUF1
        7345                            ;-
        7346
        7347    040016                      LET R0 := #ADD1+2                   ;SOURCE
                040016  012700  065430                                                      MOV     #ADD1+2,R0
        7348    040022                      LET R1 := TADDI CLR.BY #10000               ;# OF BYTES TO XFER
```

```
                                                                                    MOV     TADDI,R1
                                                                                    BIC     #10000,R1
                        LET R2 := #DSBUF1                       ;DESTINATION
                                                                                    MOV     #DSBUF1,R2
                        INLINE      <1$:    MOVB    (R0)+,  (R2)+>
                                                                                    1$:     MOVB    (R0)+,  (R2)
*
                        INLINE      <       SOB     R1,     1$>
                                                                                            SOB     R1,     1$

                    ;+
                    ;       LOAD DSBUF2
                    ;-

                        LET R0 := #ADD1A+2
                                                                                    MOV     #ADD1A+2,R0
                        LET R1 := TADDI+4 CLR.BY #10000
                                                                                    MOV     TADDI+4,R1
                                                                                    BIC     #10000,R1
                        LET R2 := #DSBUF2
                                                                                    MOV     #DSBUF2,R2
                        INLINE      <2$:    MOVB    (R0)+,  (R2)+>
                                                                                    2$:     MOVB    (R0)+,  (R2)
*
                        INLINE      <       SOB     R1,     2$>
                                                                                            SOB     R1,     2$

                    ;+
                    ;       LOAD DS.EXP
                    ;-

                        LET R0 := #ADD1C+2
                                                                                    MOV     #ADD1C+2,R0
                        LET R1 := TADDI+10 CLR.BY #10000
                                                                                    MOV     TADDI+10,R1
                                                                                    BIC     #10000,R1
                        LET R2 := #DS.EXP
                                                                                    MOV     #DS.EXP,R2
                        INLINE      <3$:    MOVB    (R0)+,  (R2)>
                                                                                    3$:     MOVB    (R0)+,  (R2)
                        INLINE      <       BISB    #60,    (R2)+>
                                                                                            BISB    #60,    (R2)
*
                        INLINE      <       SOB     R1,     3$>
                                                                                            SOB     R1,     3$

                        CALL CLRREG                             ;ENSURE R0-R5 ARE CLR
                                                                                    JSR     PC,CLRREG

                        INLINE <SUBNI>
                                                                                    SUBNI
                        INLINE <.WORD TADDI>
                                                                                    .WORD TADDI
                        INLINE <.WORD TADDI+4>
                                                                                    .WORD TADDI+4
                        INLINE <.WORD TADDI+10>
                                                                                    .WORD TADDI+10
                        LET CC := PSW
                                                                                    MOV     PSW,CC
                        LET CC := CC CLR.BY #177760
```

```
SUBN

      040140  042767  177760  140352                                                BIC     #177760,CC
 7382 040146                          IF CC NE #0 THEN                              
      040146  005767  140346                                                       TST     CC
      040152  001402                                                               BEQ     $51602
 7383 040154                              CALL ERROR                                
      040154  004767  017520                                                       JSR     PC,ERROR
 7384 040160                          ENDIF                                         
      040160                                                                $51602:
 7385
 7386                               ;+
 7387                               ;       ENSURE GPR'S STILL ZERO
 7388                               ;-
 7389
 7390 040160                          IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN      
      040160  005700                                                               TST     R0
      040162  001004                                                               BNE     $51603
      040164  005701                                                               TST     R1
      040166  001002                                                               BNE     $51603
      040170  005702                                                               TST     R2
      040172  001402                                                               BEQ     $51604
      040174                                                                $51603:
 7391 040174                              CALL ERROR                                
      040174  004767  017500                                                       JSR     PC,ERROR
 7392 040200                          ENDIF                                         
      040200                                                                $51604:
 7393 040200                          IF R3 NE #0 OR R4 NE #0 OR  R5 NE #0 THEN     
      040200  005703                                                               TST     R3
      040202  001004                                                               BNE     $51605
      040204  005704                                                               TST     R4
      040206  001002                                                               BNE     $51605
      040210  005705                                                               TST     R5
      040212  001402                                                               BEQ     $51606
      040214                                                                $51605:
 7394 040214                              CALL ERROR                                
      040214  004767  017460                                                       JSR     PC,ERROR
 7395 040220                          ENDIF                                         
      040220                                                                $51606:
 7396
 7397                               ;+
 7398                               ;       CHECK RESULTS
 7399                               ;-
 7400
 7401 040220                          LET R0 := #DS.DST
      040220  012700  064544                                                       MOV     #DS.DST,R0
 7402 040224                          LET R1 := #DS.EXP               ;EXPECTED DATA
      040224  012701  064606                                                       MOV     #DS.EXP,R1
 7403 040230                          LET ERRFLG := #0
      040230  005067  140222                                                       CLR     ERRFLG
 7404 040234                          WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO
      040234                                                                $51607:
      040234  020027  064604                                                       CMP     R0,#DS.DST+40
      040240  001410                                                               BEQ     $51610
      040242  005767  140210                                                       TST     ERRFLG
      040246  001005                                                               BNE     $51610
 7405 040250                              IFB (R0)+ NE (R1)+ THEN                   
      040250  122021                                                               CMPB    (R0)+,(R1)+
      040252  001402                                                               BEQ     $51611
```

SUBN

```
   7406 040254                                  CALL ERROR
        040254  004767  017420                                    JSR     PC,ERROR
   7407 040260                                ENDIF
        040260                                                  $51611:
   7408 040260                              ENDDO
        040260  000765                                          BR      $51607
        040262                                                  $51610:
   7409
   7410
```

```
 7412                                .SBTTL  CMPN
 7413                                ;*****************************************************************
 7414                                ;
 7415                                ;  CMPN TEST
 7416                                ;
 7417                                ;*****************************************************************
 7418
 7419 040262                        ROUTINE XCMPN                                                   XCMPN:
      040262
 7420 040262                           INLINE <SCOPE>                                                       SCOPE
      040262  000004
 7421 040264                           INLINE <NOP>                                                         NOP
      040264  000240
 7422 040266                           LET $TIMES := #20                ;DO 20 ITERATIONS                   MOV     #20,$TIMES
      040266  012767  000020  140110
 7423 040274                           LET COUNT := TCMP                ;GET TEST COUNT                     MOV     TCMP,COUNT
      040274  016767  024346  140166
 7424 040302                           LET TSTCAS := #TCMP+2                    ;POINT TO FIRST TEST CASE   MOV     #TCMP+2,TSTCAS
      040302  012767  064650  140234
 7425                                ;REPEAT
 7426 040310                           INLINE <$5052:>                                                      $5052:
      040310
 7427 040310                           CALL CLRDSB                                                          JSR     PC,CLRDSB
      040310  004767  016360
 7428 040314                           LET CC.EXP := #0                                                     CLR     CC.EXP
      040314  005067  140202
 7429 040320                           LET SIGNO := #0                                                      CLR     SIGNO
      040320  005067  140124
 7430 040324                           LET DATTYP := #0                                                     CLR     DATTYP
      040324  005067  140134
 7431
 7432                                ;+
 7433                                ; GET SOURCE DESCRIPTORS FROM TABLE
 7434                                ;-
 7435
 7436 040330                           LET R0 := TSTCAS                                                     MOV     TSTCAS,R0
      040330  016700  140210
 7437 040334                           LET DATA00 := (R0)+              ;SRC1 LENGTH                        MOV     (R0)+,DATA00
      040334  012067  140222
 7438 040340                           LET DATA01 := (R0)+              ;SRC1 POINTER                       MOV     (R0)+,DATA01
      040340  012067  140220
 7439 040344                           LET DATA02 := (R0)+              ;SRC2 LENGTH                        MOV     (R0)+,DATA02
      040344  012067  140216
 7440 040350                           LET DATA03 := (R0)               ;SRC2 POINTER                       MOV     (R0),DATA03
      040350  011067  140214
 7441
 7442                                ;+
 7443                                ; LOAD SRC1 AND SRC2 INTO DSBUF1 AND DSBUF2.
 7444                                ;-
 7445
 7446 040354                           LET R2 := DATA01 + #2            ;GET SRC1 ADDRESS                   MOV     DATA01,R2
      040354  016702  140204                                                                               ADD     #2,R2
      040360  062702  000002
 7447 040364                           LET R3 := #DSBUF1                                                    MOV     #DSBUF1,R3
      040364  012703  064440
 7448 040370                           LET DSTMP0 :B= (R2)              ;SAVE SRC1 MSD                      MOVB    (R2),DSTMP0
      040370  111267  140160
```

```
CMPN

                                        DECR R1 FROM DATA00 TO #1 BY #1
040374  016701  140162                                                  MOV     DATA00,R1
040400  000401                                                          BR      $51614
040402                                                          $51615:
040402  005301                                                          DEC     R1
040404                                                          $51614:
040404  020127  000001                                                  CMP     R1,#1
040410  002402                                                          BLT     $51616
                                            LET (R3)+ :B= (R2)+
040412  112223                                                          MOVB    (R2)+,(R3)+
                                        ENDDEC
040414  000772                                                          BR      $51615
040416                                                          $51616:
                                        IF DATA00 HI #0 THEN
040416  005767  140140                                                  TST     DATA00
040422  101402                                                          BLOS    $51617
                                            LET DSTMP1 :B= -(R2)        ;SAVE SRC1 LSD
040424  114267  140125                                                  MOVB    -(R2),DSTMP1
                                        ENDIF
040430                                                          $51617:
                                        LET R2 := DATA03 + #2           ;GET SRC2 ADDRESS
040430  016702  140134                                                  MOV     DATA03,R2
040434  062702  000002                                                  ADD     #2,R2
                                        LET R3 := #DSBUF2
040440  012703  064502                                                  MOV     #DSBUF2,R3
                                        LET DSTMP2 :B= (R2)             ;SAVE SRC2 MSD
040444  111267  140106                                                  MOVB    (R2),DSTMP2
                                        DECR R1 FROM DATA02 TO #1 BY #1
040450  016701  140112                                                  MOV     DATA02,R1
040454  000401                                                          BR      $51620
040456                                                          $51621:
040456  005301                                                          DEC     R1
040460                                                          $51620:
040460  020127  000001                                                  CMP     R1,#1
040464  002402                                                          BLT     $51622
                                            LET (R3)+ :B= (R2)+
040466  112223                                                          MOVB    (R2)+,(R3)+
                                        ENDDEC
040470  000772                                                          BR      $51621
040472                                                          $51622:
                                        LET DSTMP3 :B= -(R2)                ;SAVE SRC2 LSD
040472  114267  140061                                                  MOVB    -(R2),DSTMP3

                                        ;+
                                        ; COMPARE SRC1 AND SRC2, AND SET N AND Z BITS
                                        ; IN CC.EXP ACCORDINGLY.
                                        ;-

                                        IF DATA00 EQ #0 AND DATA02 EQ #0 THEN       ;SRC1 LENGTH AND SRC2 LENGTH = 0
040476  005767  140060                                                  TST     DATA00
040502  001007                                                          BNE     $51623
040504  005767  140056                                                  TST     DATA02
040510  001004                                                          BNE     $51623
                                            LET CC.EXP := CC.EXP SET.BY #ZBIT
040512  052767  000004  140002                                          BIS     #ZBIT,CC.EXP
                                        ELSE
040520  000424                                                          BR      $51624
```

```
CMPN

        040522                                                                  $51623:
   7471 040522                          IF DATA00 EQ #0 AND @DATA03 EQ #0 THEN    ;SRC1 LENGTH = 0 AND SRC2 SIGN IS +
        040522  005767  140034                                                          TST     DATA00
        040526  001007                                                                  BNE     $51625
        040530  005777  140034                                                          TST     @DATA03
        040534  001004                                                                  BNE     $51625
   7472 040536                              LET CC.EXP := CC.EXP SET.BY #NBIT
        040536  052767  000010  137756                                                  BIS     #NBIT,CC.EXP
   7473 040544                          ELSE
        040544  000412                                                                 BR      $51626
        040546                                                                  $51625:
   7474 040546                              IF DATA02 EQ #0 AND @DATA01 EQ #-1 THEN    ;SRC2 LENGTH = 0 & SRC1 SIGN IS -
        040546  005767  140014                                                          TST     DATA02
        040552  001007                                                                  BNE     $51627
        040554  027727  140004  177777                                                  CMP     @DATA01,#-1
        040562  001003                                                                  BNE     $51627
   7475 040564                                  LET CC.EXP := CC.EXP SET.BY #NBIT
        040564  052767  000010  137730                                                  BIS     #NBIT,CC.EXP
   7476 040572                              ENDIF
        040572                                                                  $51627:
   7477 040572                          ENDIF
        040572                                                                  $51626:
   7478 040572                      ENDIF
        040572                                                                  $51624:
   7479 040572                      IF DATA00 NE #0 AND DATA02 NE #0 THEN         ;SRC1 LENGTH AND SRC2 LENGTH > 0
        040572  005767  137764                                                          TST     DATA00
        040576  001520                                                                  BEQ     $51630
        040600  005767  137762                                                          TST     DATA02
        040604  001515                                                                  BEQ     $51630
   7480 040606                          IF @DATA01 NE @DATA03 THEN                ;SRC1 SIGN AND SRC2 SIGN ARE DIFFERENT
        040606  027777  137752  137754                                                  CMP     @DATA01,@DATA03
        040614  001410                                                                  BEQ     $51631
   7481 040616                              IF @DATA01 EQ #-1 THEN                  ;SRC1 SIGN IS -
        040616  027727  137742  177777                                                  CMP     @DATA01,#-1
        040624  001003                                                                  BNE     $51632
   7482 040626                                  LET CC.EXP := CC.EXP SET.BY #NBIT
        040626  052767  000010  137666                                                  BIS     #NBIT,CC.EXP
   7483 040634                              ENDIF
        040634                                                                  $51632:
   7484 040634                          ELSE                              ;SRC1 SIGN AND SRC2 SIGN ARE EQUAL
        040634  000501                                                                 BR      $51633
        040636                                                                  $51631:
   7485 040636                              LET SIGNO := @DATA01                   ;SAVE THE COMMON SIGN
        040636  017767  137722  137604                                                  MOV     @DATA01,SIGNO
   7486 040644                              IF DATA00 LO DATA02 AND SIGNO EQ #0 THEN   ;SRC1 LENGTH < SRC2 LENGTH AND SIGNS A
RE +
        040644  026767  137712  137714                                                  CMP     DATA00,DATA02
        040652  103007                                                                 BHIS    $51634
        040654  005767  137570                                                          TST     SIGNO
        040660  001004                                                                  BNE     $51634
   7487 040662                                  LET CC.EXP := CC.EXP SET.BY #NBIT
        040662  052767  000010  137632                                                  BIS     #NBIT,CC.EXP
   7488 040670                              ELSE
        040670  000413                                                                 BR      $51635
        040672                                                                  $51634:
   7489 040672                                  IF DATA00 HI DATA02 AND SIGNO EQ #-1 THEN    ;SRC1 LENGTH > SRC2 LENGTH AND S
IGNS ARE -
        040672  026767  137664  137666                                                  CMP     DATA00,DATA02
        040700  101407                                                                 BLOS    $51636
```

CMPN

```
        040702  026727  137542  177777                                                  CMP     SIGNO,#-1
        040710  001003                                                                  BNE     $51636
   7490 040712                          LET CC.EXP := CC.EXP SET.BY #NBIT
        040712  052767  000010  137602                                                  BIS     #NBIT,CC.EXP
   7491 040720                      ENDIF
        040720                                                                      $51636:
   7492 040720                  ENDIF
        040720                                                                      $51635:
   7493 040720                  IF DATA00 EQ DATA02 THEN            ;SRC1 LENGTH AND SRC2 LENGTH ARE EQUAL
        040720  026767  137636  137640                                                  CMP     DATA00,DATA02
        040726  001044                                                                  BNE     $51637
   7494
   7495                     ;+
   7496                     ; COMPARE SRC1 AND SRC2 BYTE BY BYTE
   7497                     ;-
   7498
   7499 040730                      LET R0 := DATA00                ;SRC BUFFER LENGTH
        040730  016700  137626                                                          MOV     DATA00,R0
   7500 040734                      LET R1 := #DSBUF1               ;SRC1 BUFFER
        040734  012701  064440                                                          MOV     #DSBUF1,R1
   7501 040740                      LET R2 := #DSBUF2               ;SRC2 BUFFER
        040740  012702  064502                                                          MOV     #DSBUF2,R2
   7502 040744                      WHILEB R0 NE #0 AND (R1)+ EQ (R2)+ DO
        040744                                                                      $51640:
        040744  105700                                                                  TSTB    R0
        040746  001404                                                                  BEQ     $51641
        040750  022122                                                                  CMP     (R1)+,(R2)+
        040752  001002                                                                  BNE     $51641
   7503 040754                          LET R0 := R0 - #1           ;DECREMENT COUNTER
        040754  005300                                                                  DEC     R0
   7504 040756                      ENDDO
        040756  000772                                                                  BR      $51640
        040760                                                                      $51641:
   7505 040760                      IFB -(R1) NE -(R2) THEN  ;SOURCES WERE NOT =
        040760  124142                                                                  CMPB    -(R1),-(R2)
        040762  001423                                                                  BEQ     $51642
   7506 040764                          IFB (R1) LO (R2) AND SIGNO EQ #0 THEN
        040764  121112                                                                  CMPB    (R1),(R2)
        040766  103007                                                                  BHIS    $51643
        040770  005767  137454                                                          TST     SIGNO
        040774  001004                                                                  BNE     $51643
   7507 040776                              LET CC.EXP := CC.EXP SET.BY #NBIT
        040776  052767  000010  137516                                                  BIS     #NBIT,CC.EXP
   7508 041004                          ELSE
        041004  000411                                                                  BR      $51644
        041006                                                                      $51643:
   7509 041006                              IFB (R1) HI (R2) AND SIGNO EQ #-1 THEN
        041006  121112                                                                  CMPB    (R1),(R2)
        041010  101407                                                                  BLOS    $51645
        041012  026727  137432  177777                                                  CMP     SIGNO,#-1
        041020  001003                                                                  BNE     $51645
   7510 041022                                  LET CC.EXP := CC.EXP SET.BY #NBIT
        041022  052767  000010  137472                                                  BIS     #NBIT,CC.EXP
   7511 041030                              ENDIF
        041030                                                                      $51645:
   7512 041030                          ENDIF
        041030                                                                      $51644:
```

```
 7513 041030                                  - ELSE                          ;SOURCES WERE EQUAL
      041030  000403                                                                          BR      $51646
      041032                                                                          $51642:
 7514 041032                                      LET CC.EXP := CC.EXP SET.BY #ZBIT
      041032  052767  000004  137462                                                        BIS     #ZBIT,CC.EXP
 7515 041040                                   ENDIF
      041040                                                                          $51646:
 7516 041040                                ENDIF
      041040                                                                          $51637:
 7517 041040                             ENDIF
      041040                                                                          $51633:
 7518 041040                          ENDIF
      041040                                                                          $51630:
 7519
 7520                                ;+
 7521                                ; PERFORM THE CMPN INSTRUCTION WITH ALL COMBINATIONS
 7522                                ; OF SIGNED ZONE, TRAILING AND LEADING OVERPUNCH, AND
 7523                                ; TRAILING AND LEADING SEPARATE (UNSIGNED ZONE WILL
 7524                                ; BE TESTED IN THE INLINE PORTION OF THE TEST).
 7525                                ; CHECK THE RESULTS AND REPORT ERRORS. "DATTYP"
 7526                                ; WILL CONTAIN SRC1 DATA TYPE AND "TEMP1" WILL
 7527                                ; CONTAIN SRC2 DATA TYPE CURRENTLY BEING TESTED.
 7528                                ;-
 7529
 7530                                ;REPEAT                          ;SRC1 LOOP
 7531 041040                           INLINE  <$5201:>
      041040                                                                          $5201:
 7532 041040                           IF DATTYP EQ #10000 THEN      ;SKIP UNSIGNED ZONED
      041040  026727  137420  010000                                                   CMP     DATTYP,#10000
      041046  001003                                                                  BNE     $51647
 7533 041050                               LET DATTYP := DATTYP + #10000
      041050  062767  010000  137406                                                  ADD     #10000,DATTYP
 7534 041056                           ENDIF
      041056                                                                          $51647:
 7535
 7536 041056                           LET DSBUF1 :B= DSTMP0                 ;SRC1 MSD
      041056  116767  137472  023354                                                  MOVB    DSTMP0,DSBUF1
 7537 041064                           LET R0 := #DSBUF1-1 + DATA00
      041064  012700  064437                                                          MOV     #DSBUF1-1,R0
      041070  066700  137466                                                          ADD     DATA00,R0
 7538 041074                           LET (R0) :B= DSTMP1           ;SRC1 LSD
      041074  116710  137455                                                          MOVB    DSTMP1,(R0)
 7539
 7540 041100                           ;PUT THE PROPER SIGN INTO SRC1, ACCORDING TO DATTYP.
      041100  005767  137360             IF DATTYP EQ #0 THEN                 ;*** SIGNED ZONED ***
      041104  001017                                                                  TST     DATTYP
                                                                                      BNE     $51650
 7541 041106                               LET R0 := #DSBUF1-1 + DATA00
      041106  012700  064437                                                          MOV     #DSBUF1-1,R0
      041112  066700  137444                                                          ADD     DATA00,R0
 7542 041116                               IF @DATA01 EQ #0 THEN     ;SIGN IS +
      041116  005777  137442                                                          TST     @DATA01
      041122  001003                                                                  BNE     $51651
 7543 041124                                   LET (R0) :B= (R0)     SET.BY #60
      041124  152710  000060                                                          BISB    #60,(R0)
 7544 041130                               ELSE                                ;SIGN IS -
      041130  000402                                                                  BR      $51652
      041132                                                                  $51651:
```

CMPN

```
   7545 041132                              LET (R0) :B= (R0) SET.BY #160
        041132  152710  000160                                                BISB    #160,(R0)
   7546 041136                           ENDIF
        041136                                                          $51652:
   7547 041136                           LET R0 := #DSBUF1-1
        041136  012700  064437                                                 MOV     #DSBUF1-1,R0
   7548 041142                           LET (R0) :B= #0          ;RESTORE AFTER LEADING SEPARATE
        041142  105010                                                         CLRB    (R0)
   7549 041144                        ENDIF
        041144                                                          $51650:
   7550 041144                        IF DATTYP EQ #20000 THEN              ;*** TRAILING OVERPUNCH ***
        041144  026727  137314  020000                                         CMP     DATTYP,#20000
        041152  001021                                                         BNE     $51653
   7551 041154                           IF DATA00 HI #0 THEN
        041154  005767  137402                                                 TST     DATA00
        041160  101416                                                         BLOS    $51654
   7552 041162                              LET R0 := #DSBUF1-1 + DATA00
        041162  012700  064437                                                 MOV     #DSBUF1-1,R0
        041166  066700  137370                                                 ADD     DATA00,R0
   7553 041172                              IF @DATA01 EQ #0 THEN        ;SIGN IS +
        041172  005777  137366                                                 TST     @DATA01
        041176  001002                                                         BNE     $51655
   7554 041200                                 LET R1 := #0
        041200  005001                                                         CLR     R1
   7555 041202                              ELSE                         ;SIGN IS -
        041202  000402                                                         BR      $51656
        041204                                                          $51655:
   7556 041204                                 LET R1 := #10.
        041204  012701  000012                                                 MOV     #10.,R1
   7557 041210                              ENDIF
        041210                                                          $51656:
   7558 041210                              LET R1 :B= R1 SET.BY (R0)    ;FORM TABLE OFFSET
        041210  151001                                                         BISB    (R0),R1
   7559 041212                              LET (R0) :B= OVPNCH(R1)      ;PUT IN SIGN/DIGIT
        041212  116110  056330                                                 MOVB    OVPNCH(R1),(R0)
   7560 041216                           ENDIF
        041216                                                          $51654:
   7561 041216                        ENDIF
        041216                                                          $51653:
   7562 041216                        IF DATTYP EQ #30000 THEN              ;*** LEADING OVERPUNCH ***
        041216  026727  137242  030000                                         CMP     DATTYP,#30000
        041224  001017                                                         BNE     $51657
   7563 041226                           IF DATA00 HI #0 THEN
        041226  005767  137330                                                 TST     DATA00
        041232  101414                                                         BLOS    $51660
   7564 041234                              LET R0 := #DSBUF1
        041234  012700  064440                                                 MOV     #DSBUF1,R0
   7565 041240                              IF @DATA01 EQ #0 THEN              ;SIGN IS +
        041240  005777  137320                                                 TST     @DATA01
        041244  001002                                                         BNE     $51661
   7566 041246                                 LET R1 := #0
        041246  005001                                                         CLR     R1
   7567 041250                              ELSE                         ;SIGN IS -
        041250  000402                                                         BR      $51662
        041252                                                          $51661:
   7568 041252                                 LET R1 := #10.
        041252  012701  000012                                                 MOV     #10.,R1
```

CMPN

```
 7569 041256                            ENDIF
      041256                                                                $51662:
 7570 041256                            LET R1 :B= R1 SET.BY (R0)     ;FORM TABLE OFFSET
      041256  151001                                                        BISB    (R0),R1
 7571 041260                            LET (R0) :B= OVPNCH(R1)       ;PUT IN SIGN/DIGIT
      041260  116110  056330                                              MOVB    OVPNCH(R1),(R0)
 7572 041264                         ENDIF
      041264                                                              $51660:
 7573 041264                      ENDIF
      041264                                                              $51657:
 7574 041264                      IF DATTYP EQ #40000 THEN              ;*** TRAILING SEPARATE ***
      041264  026727  137174  040000                                         CMP     DATTYP,#40000
      041272  001014                                                        BNE     $51663
 7575 041274                         LET R0 := #DSBUF1 + DATA00
      041274  012700  064440                                                MOV     #DSBUF1,R0
      041300  066700  137256                                                ADD     DATA00,R0
 7576 041304                         IF @DATA01 EQ #0 THEN             ;SIGN IS +
      041304  005777  137254                                                TST     @DATA01
      041310  001003                                                        BNE     $51664
 7577 041312                            LET (R0) :B= #53
      041312  112710  000053                                               MOVB    #53,(R0)
 7578 041316                         ELSE                              ;SIGN IS -
      041316  000402                                                        BR      $51665
      041320                                                             $51664:
 7579 041320                            LET (R0) :B= #55
      041320  112710  000055                                               MOVB    #55,(R0)
 7580 041324                         ENDIF
      041324                                                             $51665:
 7581 041324                      ENDIF
      041324                                                             $51663:
 7582 041324                      IF DATTYP EQ #50000 THEN              ;*** LEADING SEPARATE ***
      041324  026727  137134  050000                                         CMP     DATTYP,#50000
      041332  001017                                                        BNE     $51666
 7583 041334                         LET R0 := #DSBUF1-1
      041334  012700  064437                                                MOV     #DSBUF1-1,R0
 7584 041340                         IF @DATA01 EQ #0 THEN             ;SIGN IS +
      041340  005777  137220                                                TST     @DATA01
      041344  001003                                                        BNE     $51667
 7585 041346                            LET (R0) :B= #53
      041346  112710  000053                                               MOVB    #53,(R0)
 7586 041352                         ELSE                              ;SIGN IS -
      041352  000402                                                        BR      $51670
      041354                                                             $51667:
 7587 041354                            LET (R0) :B= #55
      041354  112710  000055                                               MOVB    #55,(R0)
 7588 041360                         ENDIF
      041360                                                             $51670:
 7589 041360                         LET R0 := #DSBUF1 + DATA00
      041360  012700  064440                                                MOV     #DSBUF1,R0
      041364  066700  137172                                                ADD     DATA00,R0
 7590 041370                         LET (R0) :B= #0                   ;RESTORE AFTER TRAILING SEPARATE
      041370  105010                                                        CLRB    (R0)
 7591 041372                      ENDIF
      041372                                                             $51666:
 7592 041372                      LET TEMP1 := #0                       ;CLEAR SRC2 DATA TYPE
      041372  005067  137102                                                CLR     TEMP1
 7593                          ;REPEAT                              ;SRC2 LOOP
```

```
 7594 041376                            INLINE  <$5202:>                                                    $5202:
      041376
 7595 041376                            IF TEMP1 EQ #10000 THEN      ;SKIP UNSIGNED ZONED                    CMP     TEMP1,#10000
      041376  026727  137076  010000                                                                          BNE     $51671
      041404  001003
 7596 041406                                LET TEMP1 := TEMP1 + #10000                                       ADD     #10000,TEMP1
      041406  062767  010000  137064
 7597 041414                            ENDIF                                                          $51671:
      041414
 7598                                  ;RESTORE THE MSD AND LSD OF SRC2
 7599 041414                            LET DSBUF2 :B= DSTMP2                ;SRC2 MSD                  MOVB    DSTMP2,DSBUF2
      041414  116767  137136  023060
 7600 041422                            LET R0 := #DSBUF2-1 + DATA02                                      MOV     #DSBUF2-1,R0
      041422  012700  064501                                                                               ADD     DATA02,R0
      041426  066700  137134
 7601 041432                            LET (R0) :B= DSTMP3                  ;SRC2 LSD                  MOVB    DSTMP3,(R0)
      041432  116710  137121
 7602                                  ;PUT THE PROPER SIGN INTO SRC2, ACCORDING TO TEMP1
 7603 041436                            IF TEMP1 EQ #0 THEN                  ;*** SIGNED ZONED ***      TST     TEMP1
      041436  005767  137036                                                                               BNE     $51672
      041442  001017
 7604 041444                                LET R0 := #DSBUF2-1 + DATA02                                  MOV     #DSBUF2-1,R0
      041444  012700  064501                                                                               ADD     DATA02,R0
      041450  066700  137112
 7605 041454                                IF @DATA03 EQ #0 THEN            ;SIGN IS +                  TST     @DATA03
      041454  005777  137110                                                                               BNE     $51673
      041460  001003
 7606 041462                                    LET (R0) :B= (R0) SET.BY #60                              BISB    #60,(R0)
      041462  152710  000060
 7607 041466                                ELSE                             ;SIGN IS -                  BR      $51674
      041466  000402                                                                                  $51673:
      041470
 7608 041470                                    LET (R0) :B= (R0) SET.BY #160                             BISB    #160,(R0)
      041470  152710  000160
 7609 041474                                ENDIF                                                     $51674:
      041474
 7610 041474                                LET R0 := #DSBUF2-1                                           MOV     #DSBUF2-1,R0
      041474  012700  064501
 7611 041500                                LET (R0) :B= #0                  ;RESTORE AFTER LEADING SEPARATE
      041500  105010                                                                                       CLRB    (R0)
 7612 041502                            ENDIF                                                          $51672:
      041502
 7613 041502                            IF TEMP1 EQ #20000 THEN              ;*** TRAILING OVERPUNCH ***
      041502  026727  136772  020000                                                                        CMP     TEMP1,#20000
      041510  001016                                                                                       BNE     $51675
 7614 041512                                LET R0 := #DSBUF2-1 + DATA02                                  MOV     #DSBUF2-1,R0
      041512  012700  064501                                                                               ADD     DATA02,R0
      041516  066700  137044
 7615 041522                                IF @DATA03 EQ #0 THEN            ;SIGN IS +
      041522  005777  137042                                                                               TST     @DATA03
      041526  001002                                                                                       BNE     $51676
 7616 041530                                    LET R1 := #0
      041530  005001                                                                                       CLR     R1
 7617 041532                                ELSE                             ;SIGN IS -
      041532  000402                                                                                       BR      $51677
      041534                                                                                          $51676:
 7618 041534                                    LET R1 := #10.
```

```
                                                                                    MOV     #10.,R1
7619 041540                         ENDIF
     041540                                                                       $51677:
7620 041540                         LET R1 :B= R1 SET.BY (R0)       ;FORM TABLE OFFSET
     041540  151001                                                             BISB    (R0),R1
7621 041542                         LET (R0) :B= OVPNCH(R1)         ;PUT IN SIGN/DIGIT
     041542  116110  056330                                                     MOVB    OVPNCH(R1),(R0)
7622 041546                     ENDIF
     041546                                                                     $51675:
7623 041546                     IF TEMP1 EQ #30000 THEN             ;*** LEADING OVERPUNCH ***
     041546  026727  136726  030000                                             CMP     TEMP1,#30000
     041554  001014                                                             BNE     $51700
7624 041556                         LET R0 := #DSBUF2
     041556  012700  064502                                                     MOV     #DSBUF2,R0
7625 041562                         IF @DATA03 EQ #0 THEN           ;SIGN IS +
     041562  005777  137002                                                     TST     @DATA03
     041566  001002                                                             BNE     $51701
7626 041570                             LET R1 := #0
     041570  005001                                                             CLR     R1
7627 041572                         ELSE                            ;SIGN IS -
     041572  000402                                                             BR      $51702
     041574                                                                   $51701:
7628 041574                             LET R1 := #10.
     041574  012701  000012                                                     MOV     #10.,R1
7629 041600                         ENDIF
     041600                                                                   $51702:
7630 041600                         LET R1 :B= R1 SET.BY (R0)       ;FORM TABLE OFFSET
     041600  151001                                                             BISB    (R0),R1
7631 041602                         LET (R0) :B= OVPNCH(R1)         ;PUT IN SIGN/DIGIT
     041602  116110  056330                                                     MOVB    OVPNCH(R1),(R0)
7632 041606                     ENDIF
     041606                                                                   $51700:
7633 041606                     IF TEMP1 EQ #40000 THEN             ;*** TRAILING SEPARATE ***
     041606  026727  136666  040000                                             CMP     TEMP1,#40000
     041614  001014                                                             BNE     $51703
7634 041616                         LET R0 := #DSBUF2 + DATA02
     041616  012700  064502                                                     MOV     #DSBUF2,R0
     041622  066700  136740                                                     ADD     DATA02,R0
7635 041626                         IF @DATA03 EQ #0 THEN           ;SIGN IS +
     041626  005777  136736                                                     TST     @DATA03
     041632  001003                                                             BNE     $51704
7636 041634                             LET (R0) :B= #53
     041634  112710  000053                                                     MOVB    #53,(R0)
7637 041640                         ELSE                            ;SIGN IS -
     041640  000402                                                             BR      $51705
     041642                                                                   $51704:
7638 041642                             LET (R0) :B= #55
     041642  112710  000055                                                     MOVB    #55,(R0)
7639 041646                         ENDIF
     041646                                                                   $51705:
7640 041646                     ENDIF
     041646                                                                   $51703:
7641 041646                     IF TEMP1 EQ #50000 THEN             ;*** LEADING SEPARATE ***
     041646  026727  136626  050000                                             CMP     TEMP1,#50000
     041654  001017                                                             BNE     $51706
7642 041656                         LET R0 := #DSBUF2-1
     041656  012700  064501                                                     MOV     #DSBUF2-1,R0
```

CMPN

```
 7643 041662                        IF @DATA03 EQ #0 THEN           ;SIGN IS +
      041662  005777  136702                                                    TST     @DATA03
      041666  001003                                                            BNE     $51707
 7644 041670                            LET (R0) :B= #53
      041670  112710  000053                                                    MOVB    #53,(R0)
 7645 041674                        ELSE                            ;SIGN IS -
      041674  000402                                                            BR      $51710
      041676                                                        $51707:
 7646 041676                            LET (R0) :B= #55
      041676  112710  000055                                                    MOVB    #55,(R0)
 7647 041702                        ENDIF
      041702                                                        $51710:
 7648 041702                        LET R0 := #DSBUF2 + DATA02
      041702  012700  064502                                                    MOV     #DSBUF2,R0
      041706  066700  136654                                                    ADD     DATA02,R0
 7649 041712                        LET (R0) :B= #0          ;RESTORE AFTER TRAILING SEPARATE
      041712  105010                                                            CLRB    (R0)
 7650 041714                    ENDIF
      041714                                                        $51706:
 7651
 7652                           ;+
 7653                           ; LOAD THE GPR'S AND DO THE CMPN INSTRUCTION
 7654                           ;-
 7655
 7656 041714                    LET R0 := DATTYP SET.BY DATA00      ;SRC1.DSCR
      041714  016700  136544                                                    MOV     DATTYP,R0
      041720  056700  136636                                                    BIS     DATA00,R0
 7657 041724                    LET R1 := #DSBUF1                   ;  "
      041724  012701  064440                                                    MOV     #DSBUF1,R1
 7658 041730                    LET R2 := TEMP1 SET.BY DATA02       ;SRC2.DSCR
      041730  016702  136544                                                    MOV     TEMP1,R2
      041734  056702  136626                                                    BIS     DATA02,R2
 7659 041740                    LET R3 := #DSBUF2                   ;  "
      041740  012703  064502                                                    MOV     #DSBUF2,R3
 7660 041744                    LET R4 := #0
      041744  005004                                                            CLR     R4
 7661 041746                    LET R5 := #0
      041746  005005                                                            CLR     R5
 7662 041750                    INLINE <CMPN>
      041750  076052                                                           CMPN
 7663
 7664                           ;+
 7665                           ; CHECK THE RESULTS
 7666                           ;-
 7667
 7668 041752                    LET CC := PSW
      041752  016767  136020  136540                                            MOV     PSW,CC
 7669 041760                    LET CC := CC CLR.BY #177760
      041760  042767  177760  136532                                            BIC     #177760,CC
 7670 041766                    IF CC NE CC.EXP THEN
      041766  026767  136526  136526                                            CMP     CC,CC.EXP
      041774  001402                                                            BEQ     $51711
 7671 041776                        CALL ERROR
      041776  004767  015676                                                    JSR     PC,ERROR
 7672 042002                    ENDIF
      042002                                                        $51711:
 7673 042002                    IF R0 NE #0 THEN
```

CMPN

```
        042002  005700                                                  TST     R0
        042004  001402                                                  BEQ     $51712
   7674 042006                          CALL ERROR
        042006  004767  015666                                         JSR     PC,ERROR
   7675 042012                        ENDIF
        042012                                                  $51712:
   7676 042012                        IF R1 NE #0 THEN
        042012  005701                                                  TST     R1
        042014  001402                                                  BEQ     $51713
   7677 042016                          CALL ERROR
        042016  004767  015656                                         JSR     PC,ERROR
   7678 042022                        ENDIF
        042022                                                  $51713:
   7679 042022                        IF R2 NE #0 THEN
        042022  005702                                                  TST     R2
        042024  001402                                                  BEQ     $51714
   7680 042026                          CALL ERROR
        042026  004767  015646                                         JSR     PC,ERROR
   7681 042032                        ENDIF
        042032                                                  $51714:
   7682 042032                        IF R3 NE #0 THEN
        042032  005703                                                  TST     R3
        042034  001402                                                  BEQ     $51715
   7683 042036                          CALL ERROR
        042036  004767  015636                                         JSR     PC,ERROR
   7684 042042                        ENDIF
        042042                                                  $51715:
   7685 042042                        IF R4 NE #0 THEN
        042042  005704                                                  TST     R4
        042044  001402                                                  BEQ     $51716
   7686 042046                          CALL ERROR
        042046  004767  015626                                         JSR     PC,ERROR
   7687 042052                        ENDIF
        042052                                                  $51716:
   7688 042052                        IF R5 NE #0 THEN
        042052  005705                                                  TST     R5
        042054  001402                                                  BEQ     $51717
   7689 042056                          CALL ERROR
        042056  004767  015616                                         JSR     PC,ERROR
   7690 042062                        ENDIF
        042062                                                  $51717:
   7691 042062                        LET TEMP1 := TEMP1 + #10000     ;UPDATE SRC2 TO NEXT DATA TYPE
        042062  062767  010000  136410                                 ADD     #10000,TEMP1
   7692                             ;UNTIL TEMP1 EQ #60000
   7693 042070                        IF TEMP1 NE #60000 THEN
        042070  026727  136404  060000                                 CMP     TEMP1,#60000
        042076  001402                                                  BEQ     $51720
   7694 042100                          INLINE  <JMP  $5202>
        042100  000167  177272                                         JMP  $5202
   7695 042104                        ENDIF
        042104                                                  $51720:
   7696 042104                        LET DATTYP := DATTYP + #10000     ;UPDATE SRC1 TO NEXT DATA TYPE
        042104  062767  010000  136352                                 ADD     #10000,DATTYP
   7697                             ;UNTIL DATTYP EQ #60000
   7698 042112                        IF DATTYP NE #60000 THEN
        042112  026727  136346  060000                                 CMP     DATTYP,#60000
        042120  001402                                                  BEQ     $51721
```

```
7699 042122                             INLINE  <JMP  $5201>
     042122  000167  176712                                                    JMP  $5201
7700 042126                             ENDIF
     042126                                                              $51721:
7701
7702 042126                             LET TSTCAS := TSTCAS + #10
     042126  062767  000010  136410                                             ADD     #10,TSTCAS
7703 042134                             LET COUNT := COUNT - #1
     042134  005367  136330                                                    DEC     COUNT
7704                                   ;UNTIL COUNT EQ #0 THEN
7705 042140                               IF COUNT NE #0 THEN
     042140  005767  136324                                                    TST     COUNT
     042144  001402                                                            BEQ     $51722
7706 042146                                 INLINE <JMP $5052>
     042146  000167  176136                                                    JMP $5052
7707 042152                             ENDIF
     042152                                                              $51722:
7708
7709
7710                                   ;+
7711                                   ; DO ONE INLINE CASE - CMPNI.  USE UNSIGNED ZONED DATA TYPE.
7712                                   ;-
7713
7714 042152                             LET R0 := #-1
     042152  012700  177777                                                    MOV     #-1,R0
7715 042156                             LET R1 := #-1
     042156  012701  177777                                                    MOV     #-1,R1
7716 042162                             LET R2 := #-1
     042162  012702  177777                                                    MOV     #-1,R2
7717 042166                             LET R3 := #-1
     042166  012703  177777                                                    MOV     #-1,R3
7718 042172                             LET R4 := #-1
     042172  012704  177777                                                    MOV     #-1,R4
7719 042176                             LET R5 := #-1
     042176  012705  177777                                                    MOV     #-1,R5
7720 042202                             INLINE   <CMPNI>
     042202  076152                                                            CMPNI
7721 042204                             INLINE   <.WORD   TCMPNI>
     042204  042320                                                            .WORD   TCMPNI
7722 042206                             INLINE   <.WORD   TCMPNI+4>
     042206  042324                                                            .WORD   TCMPNI+4
7723 042210                             LET CC := PSW
     042210  016767  135562  136302                                             MOV     PSW,CC
7724 042216                             LET CC := CC CLR.BY #177760
     042216  042767  177760  136274                                             BIC     #177760,CC
7725 042224                             IF CC NE #4 THEN
     042224  026727  136270  000004                                             CMP     CC,#4
     042232  001402                                                            BEQ     $51723
7726 042234                                 CALL ERROR
     042234  004767  015440                                                    JSR     PC,ERROR
7727 042240                             ENDIF
     042240                                                              $51723:
7728 042240                             IF R0 NE #-1 OR R1 NE #-1 OR R2 NE #-1 THEN
     042240  020027  177777                                                    CMP     R0,#-1
     042244  001006                                                            BNE     $51724
     042246  020127  177777                                                    CMP     R1,#-1
     042252  001003                                                            BNE     $51724
```

CMPN

```
        042254  020227  177777                                                  CMP     R2,#-1
        042260  001402                                                           BEQ     $51725
        042262                                                          $51724:
   7729 042262                      CALL ERROR
        042262  004767  015412                                                  JSR     PC,ERROR
   7730 042266                      ENDIF
        042266                                                          $51725:
   7731 042266                      IF R3 NE #-1 OR R4 NE #-1 OR R5 NE #-1 THEN
        042266  020327  177777                                                  CMP     R3,#-1
        042272  001006                                                          BNE     $51726
        042274  020427  177777                                                  CMP     R4,#-1
        042300  001003                                                          BNE     $51726
        042302  020527  177777                                                  CMP     R5,#-1
        042306  001402                                                          BEQ     $51727
        042310                                                          $51726:
   7732 042310                      CALL ERROR
        042310  004767  015364                                                  JSR     PC,ERROR
   7733 042314                      ENDIF
        042314                                                          $51727:
   7734 042314                      INLINE <JMP XASHN>
        042314  000167  000016                                                  JMP XASHN
   7735
   7736                             ;+
   7737                             ; INLINE TEST CASE
   7738                             ;-
   7739
   7740 042320  010005              TCMPNI: .WORD   010005          ;SRC1 DATA TYPE = UNSIGNED ZONED, LENGTH = 5
   7741 042322  042330                      .WORD   CMPNI1          ;  "
   7742 042324  010005                      .WORD   010005          ;SRC2 DATA TYPE = UNSIGNED ZONED, LENGTH = 5
   7743 042326  042330                      .WORD   CMPNI1          ;  "
   7744
   7745 042330     001     002     003  CMPNI1: .BYTE   001,002,003,004,065     ;DECIMAL STRING = 12345
        042333     004     065
   7746                                     .EVEN
   7747
```

```
                                .SBTTL  ASHN
                                ;*****************************************************************
                                ;
                                ;  ASHN TEST
                                ;
                                ;*****************************************************************

042336                          ROUTINE XASHN                                           XASHN:
042336
042336                             INLINE  <SCOPE>                                      SCOPE
042336  000004
042340                             INLINE  <NOP>                                        NOP
042340  000240
042342                             LET $TIMES := #20                ;DO 20 ITERATIONS   MOV     #20,$TIMES
042342  012767  000020  136034
042350                             LET COUNT := TASH              ;GET TEST COUNT       MOV     TASH,COUNT
042350  016767  022540  136112
042356                             LET TSTCAS := #TASH+2              ;POINT TO FIRST TEST CASE
042356  012767  065116  136160                                                          MOV     #TASH+2,TSTCAS
042364                             LET SIGNO := #0                                      CLR     SIGNO
042364  005067  136060
                                ;REPEAT
042370                             INLINE  <$5056:>                                     $5056:
042370
042370                             LET CC.EXP := #0                                     CLR     CC.EXP
042370  005067  136126
042374                             LET DATTYP := #0                                     CLR     DATTYP
042374  005067  136064

                                ;*
                                ; GET TEST DATA FROM TABLE
                                ;-

042400                             LET R0 := TSTCAS                                     MOV     TSTCAS,R0
042400  016700  136140
042404                             LET DATA00 := (R0)+          ;SRC.LENGTH             MOV     (R0)+,DATA00
042404  012067  136152
042410                             LET DATA01 := (R0)+          ;DST.LENGTH             MOV     (R0)+,DATA01
042410  012067  136150
042414                             LET DATA02 := (R0)           ;SHIFT.DSCR             MOV     (R0),DATA02
042414  011067  136146
042420                             CALL CLRDSB                                          JSR     PC,CLRDSB
042420  004767  014250

                                ;*
                                ; PUT TEST STRING INTO DSBUF1 WITHOUT THE SIGN
                                ;-

042424                             LET R0 := DATA00                   ;SRC.LENGTH       MOV     DATA00,R0
042424  016700  136132
042430                             LET R1 := #DSBUF1                ;SRC.ADDRESS        MOV     #DSBUF1,R1
042430  012701  006440
042434                             WHILE R0 NE #0 DO                                    $51732:
042434
042434  005700                                                                         TST     R0
042436  001404                                                                         BEQ     $51733
042440                                LET (R1)+ :B= #4
```

ASHN

```
      042440  112721  000004                                                    MOVB    #4,(R1)+
 7786 042444                          LET R0 := R0 - #1
      042444  005300                                                            DEC     R0
 7787 042446                        ENDDO
      042446  000772                                                            BR      $51732
      042450                                                          $51733:
 7788
 7789                              ;+
 7790                              ; LOAD DS.EXP AND CC.EXP WITH EXPECTED RESULTS.
 7791                              ;-
 7792
 7793                              ;GET SHIFT COUNT MAGNITUDE AND PUT IT INTO TEMP0
 7794
 7795 042450                          LET TEMP0 := DATA02
      042450  016767  136112  136020                                            MOV     DATA02,TEMP0
 7796 042456                          IF #BIT07 SETIN TEMP0 THEN          ;NEGATIVE SHIFT COUNT
      042456  032767  000200  136012                                            BIT     #BIT07,TEMP0
      042464  001402                                                            BEQ     $51734
 7797 042466                              LET TEMP0 := NEGATE TEMP0
      042466  005467  136004                                                    NEG     TEMP0
 7798 042472                          ENDIF
      042472                                                          $51734:
 7799 042472                          LET TEMP0 := TEMP0 CLR.BY #177600
      042472  042767  177600  135776                                            BIC     #177600,TEMP0
 7800
 7801                              ;TAKE CARE OF EXPECTED NBIT
 7802
 7803 042500                          IF SIGNO EQ #-1 AND DATA00 NE #0 THEN
      042500  026727  135744  177777                                            CMP     SIGNO,#-1
      042506  001006                                                            BNE     $51735
      042510  005767  136046                                                    TST     DATA00
      042514  001403                                                            BEQ     $51735
 7804 042516                              LET CC.EXP := CC.EXP SET.BY #NBIT
      042516  052767  000010  135776                                            BIS     #NBIT,CC.EXP
 7805 042524                          ENDIF
      042524                                                          $51735:
 7806
 7807                              ;TAKE CARE OF ZERO LENGTH SOURCE OR DESTINATION
 7808
 7809 042524                          IF DATA01 NE #0 THEN
      042524  005767  136034                                                    TST     DATA01
      042530  001414                                                            BEQ     $51736
 7810 042532                              LET R0 := #DS.EXP
      042532  012700  064606                                                    MOV     #DS.EXP,R0
 7811 042536                              INCR R1 FROM #1 TO DATA01 BY #1
      042536  012701  000001                                                    MOV     #1,R1
      042542  000401                                                            BR      $51737
      042544                                                          $51740:
      042544  005201                                                            INC     R1
      042546                                                          $51737:
      042546  020167  136012                                                    CMP     R1,DATA01
      042552  003003                                                            BGT     $51741
 7812 042554                                  LET (R0)+ :B= #60
      042554  112720  000060                                                    MOVB    #60,(R0)+
 7813 042560                              ENDINC
      042560  000771                                                            BR      $51740
      042562                                                          $51741:
```

```
ASHN

 7814 042562                            ENDIF
      042562                                                                $51736:
 7815 042562                            IF DATA00 EQ #0 OR DATA01 EQ #0 THEN
      042562  005767  135774                                                        TST     DATA00
      042566  001403                                                               BEQ     $51742
      042570  005767  135770                                                        TST     DATA01
      042574  001004                                                               BNE     $51743
      042576                                                                $51742:
 7816 042576                                LET CC.EXP := CC.EXP SET.BY #ZBIT
      042576  052767  000004  135716                                               BIS     #ZBIT,CC.EXP
 7817 042604                            ELSE
      042604  000567                                                               BR      $51744
      042606                                                                $51743:
 7818
 7819                                ;TAKE CARE OF NEGATIVE SHIFT
 7820
 7821 042606                                IF #BIT07 SETIN DATA02 THEN
      042606  032767  000200  135752                                               BIT     #BIT07,DATA02
      042614  001465                                                               BEQ     $51745
 7822 042616                                    LET R0 := DATA00 - TEMP0    ;SRC.LEN - SHIFT COUNT
      042616  016700  135740                                                       MOV     DATA00,R0
      042622  166700  135650                                                       SUB     TEMP0,R0
 7823 042626                                    LET R1 := #DS.EXP + DATA01
      042626  012701  064606                                                       MOV     #DS.EXP,R1
      042632  066701  135726                                                       ADD     DATA01,R1
 7824 042636                                    LET R2 := #1
      042636  012702  000001                                                       MOV     #1,R2
 7825 042642                                    LET R3 := DATA01
      042642  016703  135716                                                       MOV     DATA01,R3
 7826 042646                                    IF R0 EQ #0 THEN
      042646  005700                                                               TST     R0
      042650  001012                                                               BNE     $51746
 7827 042652                                        IF DATA02 HIS #3000 THEN
      042652  026727  135710  003000                                               CMP     DATA02,#3000
      042660  103403                                                               BLO     $51747
 7828 042662                                            LET -(R1) :B= #61       ;ROUNDING
      042662  112741  000061                                                      MOVB    #61,-(R1)
 7829 042666                                        ELSE
      042666  000403                                                               BR      $51750
      042670                                                                $51747:
 7830 042670                                            LET CC.EXP := CC.EXP SET.BY #ZBIT
      042670  052767  000004  135624                                               BIS     #ZBIT,CC.EXP
 7831 042676                                        ENDIF
      042676                                                                $51750:
 7832 042676                                ENDIF
      042676                                                                $51746:
 7833 042676                                IF R0 LT #0 THEN
      042676  005700                                                               TST     R0
      042700  002003                                                               BGE     $51751
 7834 042702                                    LET CC.EXP := CC.EXP SET.BY #ZBIT
      042702  052767  000004  135612                                               BIS     #ZBIT,CC.EXP
 7835 042710                                ENDIF
      042710                                                                $51751:
 7836 042710                                WHILE R2 LE R0 AND R3 HI #0 DO
      042710                                                                $51752:
      042710  020200                                                               CMP     R2,R0
      042712  003017                                                               BGT     $51753
```

```
        042714  005703                                                                  TST     R3
        042716  101415                                                                  BLOS    $51753
   7837 042720                                  LET -(R1) :B= #64
        042720  112741  000064                                                          MOVB    #64,-(R1)
   7838 042724                                  IF R2 EQ #1 AND DATA02 HIS #3000 THEN
        042724  020227  000001                                                          CMP     R2,#1
        042730  001005                                                                 BNE     $51754
        042732  026727  135630  003000                                                  CMP     DATA02,#3000
        042740  103401                                                                 BLO     $51754
   7839 042742                                      LET (R1) :B= (R1) + #1          ;ROUNDING
        042742  105211                                                                 INCB    (R1)
   7840 042744                                  ENDIF
        042744                                                                      $51754:
   7841 042744                                  LET R2 := R2 + #1
        042744  005202                                                                 INC     R2
   7842 042746                                  LET R3 := R3 - #1
        042746  005303                                                                 DEC     R3
   7843 042750                              ENDDO
        042750  000757                                                                 BR      $51752
        042752                                                                      $51753:
   7844 042752                              IF R3 EQ #0 AND R2 LE R0 THEN
        042752  005703                                                                 TST     R3
        042754  001005                                                                 BNE     $51755
        042756  020200                                                                 CMP     R2,R0
        042760  003003                                                                 BGT     $51755
   7845 042762                                      LET CC.EXP := CC.EXP SET.BY #VBIT
        042762  052767  000002  135532                                                 BIS     #VBIT,CC.EXP
   7846 042770                                  ENDIF
        042770                                                                      $51755:
   7847 042770                              ENDIF
        042770                                                                      $51745:
   7848
   7849                                 ;TAKE CARE OF POSITIVE SHIFT
   7850
   7851 042770                              IF #BIT07 NOTSETIN DATA02 THEN
        042770  032767  000200  135570                                                 BIT     #BIT07,DATA02
        042776  001072                                                                 BNE     $51756
   7852 043000                                  LET TEMP1 := DATA00 + TEMP0         ;SRC.LEN + SHIFT COUNT
        043000  016767  135556  135472                                                 MOV     DATA00,TEMP1
        043006  066767  135464  135464                                                 ADD     TEMP0,TEMP1
   7853 043014                                  IF TEMP1 HI DATA01 THEN
        043014  026767  135460  135542                                                 CMP     TEMP1,DATA01
        043022  101403                                                                 BLOS    $51757
   7854 043024                                      LET CC.EXP := CC.EXP SET.BY #VBIT
        043024  052767  000002  135470                                                 BIS     #VBIT,CC.EXP
   7855 043032                                  ENDIF
        043032                                                                      $51757:
   7856 043032                                  IF TEMP0 HIS DATA01 THEN            ;SHIFT COUNT >= DST.LEN
        043032  026767  135440  135524                                                 CMP     TEMP0,DATA01
        043040  103420                                                                 BLO     $51760
   7857 043042                                      LET CC.EXP := CC.EXP SET.BY #ZBIT
        043042  052767  000004  135452                                                 BIS     #ZBIT,CC.EXP
   7858 043050                                      LET R1 := #DS.EXP
        043050  012701  064606                                                         MOV     #DS.EXP,R1
   7859 043054                                      INCR R0 FROM #1 TO DATA01 BY #1
        043054  012700  000001                                                         MOV     #1,R0
        043060  000401                                                                 BR      $51761
```

```
                043062                                                                  $51762:
                043062  005200                                                                  INC     R0
                043064                                                                  $51761:
                043064  020067  135474                                                          CMP     R0,DATA01
                043070  003003                                                                  BGT     $51763
   7860         043072                      LET (R1)+ :B= #60
                043072  112721  000060                                                          MOVB    #60,(R1)+
   7861         043076                  ENDINC
                043076  000771                                                                  BR      $51762
                043100                                                                  $51763:
   7862         043100              ELSE                                    ;SHIFT COUNT < DST.LEN
                043100  000431                                                                  BR      $51764
                043102                                                                  $51760:
   7863         043102                  LET R1 := #DS.EXP + DATA01
                043102  012701  064606                                                          MOV     #DS.EXP,R1
                043106  066701  135452                                                          ADD     DATA01,R1
   7864         043112                  INCR R0 FROM #1 TO TEMP0 BY #1
                043112  012700  000001                                                          MOV     #1,R0
                043116  000401                                                                  BR      $51765
                043120                                                                  $51766:
                043120  005200                                                                  INC     R0
                043122                                                                  $51765:
                043122  020067  135350                                                          CMP     R0,TEMP0
                043126  003003                                                                  BGT     $51767
   7865         043130                      LET -(R1) :B= #60
                043130  112741  000060                                                          MOVB    #60,-(R1)
   7866         043134                  ENDINC
                043134  000771                                                                  BR      $51766
                043136                                                                  $51767:
   7867         043136                  LET R0 := DATA00                        ;SRC.LEN
                043136  016700  135420                                                          MOV     DATA00,R0
   7868         043142                  WHILE R1 HI #DS.EXP AND R0 HI #0 DO
                043142                                                                  $51770:
                043142  020127  064606                                                          CMP     R1,#DS.EXP
                043146  101406                                                                  BLOS    $51771
                043150  005700                                                                  TST     R0
                043152  101404                                                                  BLOS    $51771
   7869         043154                      LET -(R1) :B= #64
                043154  112741  000064                                                          MOVB    #64,-(R1)
   7870         043160                      LET R0 := R0 - #1
                043160  005300                                                                  DEC     R0
   7871         043162                  ENDDO
                043162  000767                                                                  BR      $51770
                043164                                                                  $51771:
   7872         043164                      ENDIF
                043164                                                                  $51764:
   7873         043164                  ENDIF
                043164                                                                  $51756:
   7874         043164              ENDIF
                043164                                                                  $51744:
   7875         043164              IF #ZBIT SETIN CC.EXP THEN
                043164  032767  000004  135330                                                  BIT     #ZBIT,CC.EXP
                043172  001403                                                                  BEQ     $51772
   7876         043174                      LET CC.EXP := CC.EXP CLR.BY #NBIT
                043174  042767  000010  135320                                                  BIC     #NBIT,CC.EXP
   7877         043202              ENDIF
                043202                                                                  $51772:
```

```
 7878
 7879                                ;SAVE THE FIRST AND LAST EXPECTED DST DIGITS
 7880
 7881 043202                           LET DSTMP0 :B= DS.EXP
      043202  116767  021400  135344                                          MOVB    DS.EXP,DSTMP0
 7882 043210                           LET R0 := #DS.EXP + DATA01 - #1
      043210  012700  064606                                                 MOV     #DS.EXP,R0
      043214  066700  135344                                                 ADD     DATA01,R0
      043220  005300                                                         DEC     R0
 7883 043222                           LET DSTMP1 :B= (R0)
      043222  111067  135327                                                 MOVB    (R0),DSTMP1
 7884
 7885                                ;+
 7886                                ; PERFORM THE ASHN INSTRUCTION WITH ALL SIX NUMERIC
 7887                                ; DATA TYPES.  "DATTYP" WILL CONTAIN THE DATA TYPE
 7888                                ; CURRENTLY BEING TESTED.
 7889                                ;-
 7890
 7891                                ;INCR DATTYP FROM #0 TO #50000 BY #10000
 7892 043226                           INLINE  <CLR  DATTYP>
      043226  005067  135232                                                 CLR  DATTYP
 7893 043232                           INLINE  <BR  $5602>
      043232  000403                                                         BR  $5602
 7894 043234                           INLINE  <$5601:>
      043234                                                                 $5601:
 7895 043234                           LET DATTYP := DATTYP + #10000
      043234  062767  010000  135222                                         ADD     #10000,DATTYP
 7896 043242                           INLINE  <$5602:>
      043242                                                                 $5602:
 7897 043242                           IF DATTYP GT #50000 THEN
      043242  026727  135216  050000                                         CMP     DATTYP,#50000
      043250  003402                                                         BLE     $51773
 7898 043252                               INLINE  <JMP  $5603>
      043252  000167  001554                                                 JMP  $5603
 7899 043256                           ENDIF
      043256                                                            $51773:
 7900
 7901                                ;RESTORE FIRST AND LAST SRC DIGITS
 7902 043256                           IF DATA00 HI #0 THEN                    ;SRC.LEN > 0
      043256  005767  135300                                                 TST     DATA00
      043262  101411                                                         BLOS    $51774
 7903 043264                               LET DSBUF1 :B= #4
      043264  112767  000004  021146                                         MOVB    #4,DSBUF1
 7904 043272                               LET R0 := #DSBUF1-1 + DATA00
      043272  012700  064437                                                 MOV     #DSBUF1-1,R0
      043276  066700  135260                                                 ADD     DATA00,R0
 7905 043302                               LET (R0) :B= #4
      043302  112710  000004                                                 MOVB    #4,(R0)
 7906 043306                           ENDIF
      043306                                                            $51774:
 7907
 7908 043306                           IF DATA01 HI #0 THEN              ;DST.LEN > 0
      043306  005767  135252                                                 TST     DATA01
      043312  101411                                                         BLOS    $51775
 7909 043314                               LET DS.EXP :B= DSTMP0
      043314  116767  135234  021264                                         MOVB    DSTMP0,DS.EXP
 7910 043322                               LET R0 := #DS.EXP-1 + DATA01
```

```
                                                                                    MOV     #DS.EXP-1,R0
                                                                                    ADD     DATA01,R0
                        LET (R0) :B= DSTMP1
                                                                                    MOVB    DSTMP1,(R0)
                    ENDIF
                                                                                $51775:
                   ;FINISH SETTING UP EXPECTED RESULTS ACCORDING TO DATA TYPE
                      IF DATTYP EQ #0 THEN                      ;*** SIGNED ZONED ***
                                                                                    TST     DATTYP
                                                                                    BNE     $51776
                        LET DSBUF1-1 :B= #0                     ;RESTORE SRC AFTER LEADING SEPARATE
                                                                                    CLRB    DSBUF1-1
                        IF DATA00 HI #0 THEN                            ;SRC.LEN > 0
                                                                                    TST     DATA00
                                                                                    BLOS    $51777
                            LET R0 := #DSBUF1-1 + DATA00
                                                                                    MOV     #DSBUF1-1,R0
                                                                                    ADD     DATA00,R0
                            IF SIGN0 EQ #0 THEN                         ;SIGN IS +
                                                                                    TST     SIGN0
                                                                                    BNE     $52000
                                LET (R0) :B= #64
                                                                                    MOVB    #64,(R0)
                            ELSE                                        ;SIGN IS -
                                                                                    BR      $52001
                                                                                $52000:
                                LET (R0) :B= #164
                                                                                    MOVB    #164,(R0)
                            ENDIF
                                                                                $52001:
                    ENDIF
                                                                                $51777:
                    LET DS.EXP-1 :B= #0                     ;RESTORE DST AFTER LEADING SEPARATE
                                                                                    CLRB    DS.EXP-1
                    IF DATA01 HI #0 THEN                            ;DST.LEN > 0
                                                                                    TST     DATA01
                                                                                    BLOS    $52002
                        LET R0 := #DS.EXP-1 + DATA01
                                                                                    MOV     #DS.EXP-1,R0
                                                                                    ADD     DATA01,R0
                        IF SIGN0 EQ #0 OR DATA00 EQ #0 OR #ZBIT SETIN CC.EXP THEN   ;SIGN IS +
                                                                                    TST     SIGN0
                                                                                    BEQ     $52003
                                                                                    TST     DATA00
                                                                                    BEQ     $52003
                                                                                    BIT     #ZBIT,CC.EXP
                                                                                    BEQ     $52004
                                                                                $52003:
                            LET (R0) :B= (R0) SET.BY #60
                                                                                    BISB    #60,(R0)
                        ELSE                                            ;SIGN IS -
                                                                                    BR      $52005
                                                                                $52004:
                            LET (R0) :B= (R0) SET.BY #160
                                                                                    BISB    #160,(R0)
                        ENDIF
                                                                                $52005:
```

```
 7932 043466                        IFB DATA02 LO DATA00 AND #BIT07 NOTSETIN DATA02  AND SIGNO EQ #-1 THEN
      043466  126767  135074  135066                                                    CMPB    DATA02,DATA00
      043474  103012                                                                    BHIS    $52006
      043476  032767  000200  135062                                                    BIT     #BIT07,DATA02
      043504  001006                                                                    BNE     $52006
      043506  026727  134736  177777                                                    CMP     SIGNO,#-1
      043514  001002                                                                    BNE     $52006
 7933 043516                            LET (R0) :B= (R0) SET.BY #160
      043516  152710  000160                                                          BISB    #160,(R0)
 7934 043522                        ENDIF
      043522                                                                      $52006:
 7935 043522                    ENDIF
      043522                                                                      $52002:
 7936 043522                ENDIF
      043522                                                                      $51776:
 7937 043522                IF DATTYP EQ #10000 THEN                        ;*** UNSIGNED ZONED ***
      043522  026727  134736  010000                                                  CMP     DATTYP,#10000
      043530  001025                                                                 BNE     $52007
 7938 043532                    LET CC.EXP := CC.EXP CLR.BY #NBIT
      043532  042767  000010  134762                                                 BIC     #NBIT,CC.EXP
 7939 043540                    IF DATA00 HI #0 THEN                            ;SRC.LEN > 0
      043540  005767  135016                                                        TST     DATA00
      043544  101406                                                                BLOS    $52010
 7940 043546                        LET R0 := #DSBUF1-1 + DATA00
      043546  012700  064437                                                        MOV     #DSBUF1-1,R0
      043552  066700  135004                                                        ADD     DATA00,R0
 7941 043556                        LET (R0) :B= #64
      043556  112710  000064                                                        MOVB    #64,(R0)
 7942 043562                    ENDIF
      043562                                                                  $52010:
 7943 043562                    IF DATA01 HI #0 THEN                            ;DST.LEN > 0
      043562  005767  134776                                                        TST     DATA01
      043566  101406                                                                BLOS    $52011
 7944 043570                        LET R0 := #DS.EXP-1 + DATA01
      043570  012700  064605                                                        MOV     #DS.EXP-1,R0
      043574  066700  134764                                                        ADD     DATA01,R0
 7945 043600                        LET (R0) :B= (R0) SET.BY #60
      043600  152710  000060                                                        BISB    #60,(R0)
 7946 043604                    ENDIF
      043604                                                                  $52011:
 7947 043604                ENDIF
      043604                                                                  $52007:
 7948 043604                IF DATTYP EQ #20000 THEN                        ;*** TRAILING OVERPUNCH ***
      043604  026727  134654  020000                                                  CMP     DATTYP,#20000
      043612  001102                                                                 BNE     $52012
 7949 043614                    IF DATA00 HI #0 THEN                            ;SRC.LEN > 0
      043614  005767  134742                                                        TST     DATA00
      043620  101427                                                                BLOS    $52013
 7950 043622                        IF SIGNO EQ #-1 AND #ZBIT NOTSETIN CC.EXP THEN
      043622  026727  134622  177777                                                CMP     SIGNO,#-1
      043630  001007                                                                BNE     $52014
      043632  032767  000004  134662                                                BIT     #ZBIT,CC.EXP
      043640  001003                                                                BNE     $52014
 7951 043642                            LET CC.EXP := CC.EXP SET.BY #NBIT   ;RESTORE NBIT AFTER UNSIGNED ZONED
      043642  052767  000010  134652                                                BIS     #NBIT,CC.EXP
 7952 043650                        ENDIF
      043650                                                                  $52014:
```

ASHN

```
                                        LET R0 := #DSBUF1-1 + DATA00
043650  012700  064437                                                  MOV     #DSBUF1-1,R0
043654  066700  134702                                                  ADD     DATA00,R0
                                        IF SIGN0 EQ #0 THEN                     ;SIGN IS +
043660  005767  134564                                                  TST     SIGN0
043664  001003                                                          BNE     $52015
                                            LET (R0) :B= #104
043666  112710  000104                                                  MOVB    #104,(R0)
                                        ELSE                                    ;SIGN IS -
043672  000402                                                          BR      $52016
043674                                                          $52015:
                                            LET (R0) :B= #115
043674  112710  000115                                                  MOVB    #115,(R0)
                                        ENDIF
043700                                                          $52016:
                                    ENDIF
043700                                                          $52013:
                                    IF DATA01 HI #0 THEN                ;DST.LEN > 0
043700  005767  134660                                                  TST     DATA01
043704  101445                                                          BLOS    $52017
                                        LET R0 := #DS.EXP-1 + DATA01
043706  012700  064605                                                  MOV     #DS.EXP-1,R0
043712  066700  134646                                                  ADD     DATA01,R0
                                        IF SIGN0 EQ #0 OR DATA00 EQ #0 OR #ZBIT SETIN CC.EXP THEN  ;SIGN IS +
043716  005767  134526                                                  TST     SIGN0
043722  001407                                                          BEQ     $52020
043724  005767  134632                                                  TST     DATA00
043730  001404                                                          BEQ     $52020
043732  032767  000004  134562                                         BIT     #ZBIT,CC.EXP
043740  001402                                                          BEQ     $52021
043742                                                          $52020:
                                            LET R1 :B= #0
043742  105001                                                          CLRB    R1
                                        ELSE                                    ;SIGN IS -
043744  000402                                                          BR      $52022
043746                                                          $52021:
                                            LET R1 :B= #10.
043746  112701  000012                                                  MOVB    #10.,R1
                                        ENDIF
043752                                                          $52022:
                                    IFB DATA02 LO DATA00 AND #BIT07 NOTSETIN DATA02 AND SIGN0 EQ #-1 THEN
043752  126767  134610  134602                                         CMPB    DATA02,DATA00
043760  103012                                                          BHIS    $52023
043762  032767  000200  134576                                         BIT     #BIT07,DATA02
043770  001006                                                          BNE     $52023
043772  026727  134452  177777                                         CMP     SIGN0,#-1
044000  001002                                                          BNE     $52023
                                            LET R1 :B= #10.
044002  112701  000012                                                  MOVB    #10.,R1
                                        ENDIF
044006                                                          $52023:
                                    LET R1 :B= R1 SET.BY (R0)
044006  151001                                                          BISB    (R0),R1
                                    LET R1 := R1 CLR.BY #177760
044010  042701  177760                                                  BIC     #177760,R1
                                    LET (R0) :B= OVPNCH(R1)
044014  116110  056330                                                  MOVB    OVPNCH(R1),(R0)
```

```
   7973 044020                              ENDIF
        044020                                                                  $52017:
   7974 044020                          ENDIF
        044020                                                                  $52012:
   7975 044020                          IF DATTYP EQ #30000 THEN           ;*** LEADING OVERPUNCH ***
        044020  026727  134440  030000                                          CMP     DATTYP,#30000
        044026  001063                                                         BNE     $52024
   7976 044030                              IF DATA00 HI #0 THEN           ;SRC.LEN > 0
        044030  005767  134526                                                 TST     DATA00
        044034  101412                                                         BLOS    $52025
   7977 044036                                  IF SIGN0 EQ #0 THEN        ;SIGN IS +
        044036  005767  134406                                                 TST     SIGN0
        044042  001004                                                         BNE     $52026
   7978 044044                                      LET DSBUF1 :B= #104
        044044  112767  000104  020366                                         MOVB    #104,DSBUF1
   7979
   7980 044052                                  ELSE                       ;SIGN IS -
        044052  000403                                                         BR      $52027
        044054                                                                  $52026:
   7981 044054                                      LET DSBUF1 :B= #115
        044054  112767  000115  020356                                         MOVB    #115,DSBUF1
   7982 044062                                  ENDIF
        044062                                                                  $52027:
   7983 044062                              ENDIF
        044062                                                                  $52025:
   7984 044062                              IF DATA01 HI #0 THEN           ;DST.LEN > 0
        044062  005767  134476                                                 TST     DATA01
        044066  101443                                                         BLOS    $52030
   7985 044070                                  IF SIGN0 EQ #0 OR DATA00 EQ #0 OR #ZBIT SETIN CC.EXP THEN    ;SIGN IS +
        044070  005767  134354                                                 TST     SIGN0
        044074  001407                                                         BEQ     $52031
        044076  005767  134460                                                 TST     DATA00
        044102  001404                                                         BEQ     $52031
        044104  032767  000004  134410                                         BIT     #ZBIT,CC.EXP
        044112  001402                                                         BEQ     $52032
        044114                                                                  $52031:
   7986 044114                                  LET R1 :B= #0
        044114  105001                                                         CLRB    R1
   7987 044116                              ELSE                           ;SIGN IS -
        044116  000402                                                         BR      $52033
        044120                                                                  $52032:
   7988 044120                                  LET R1 :B= #10.
        044120  112701  000012                                                 MOVB    #10.,R1
   7989 044124                              ENDIF
        044124                                                                  $52033:
   7990 044124                              IFB DATA02 LO DATA00 AND #BIT07 NOTSETIN DATA02 AND SIGN0 EQ #-1 THEN
        044124  126767  134436  134430                                         CMPB    DATA02,DATA00
        044132  103012                                                         BHIS    $52034
        044134  032767  000200  134424                                         BIT     #BIT07,DATA02
        044142  001006                                                         BNE     $52034
        044144  026727  134300  177777                                         CMP     SIGN0,#-1
        044152  001002                                                         BNE     $52034
   7991 044154                                  LET R1 :B= #10.
        044154  112701  000012                                                 MOVB    #10.,R1
   7992 044160                              ENDIF
        044160                                                                  $52034:
   7993 044160                              LET R1 :B= R1 SET.BY DS.EXP
```

ASHN

```
      044160  156701  020422                                                    BISB    DS.EXP,R1
 7994 044164                          LET R1 := R1 CLR.BY #177760
      044164  042701  177760                                                    BIC     #177760,R1
 7995 044170                          LET DS.EXP :B= OVPNCH(R1)
      044170  116167  056330  020410                                            MOVB    OVPNCH(R1),DS.EXP
 7996 044176                       ENDIF
      044176                                                                $52030:
 7997 044176                    ENDIF
      044176                                                                $52024:
 7998 044176                    IF DATTYP EQ #40000 THEN                  ;*** TRAILING SEPARATE ***
      044176  026727  134262  040000                                            CMP     DATTYP,#40000
      044204  001071                                                            BNE     $52035
 7999
 8000                        ;PUT SIGN INTO SOURCE
 8001
 8002 044206                       LET TEMPO := #DSBUF1 + DATA00
      044206  012767  064440  134262                                            MOV     #DSBUF1,TEMPO
      044214  066767  134342  134254                                            ADD     DATA00,TEMPO
 8003 044222                       IF SIGNO EQ #0 THEN                       ;SIGN IS +
      044222  005767  134222                                                    TST     SIGNO
      044226  001004                                                            BNE     $52036
 8004 044230                          LET @TEMPO :B= #53
      044230  112777  000053  134240                                            MOVB    #53,@TEMPO
 8005 044236                       ELSE                                      ;SIGN IS -
      044236  000403                                                            BR      $52037
      044240                                                                $52036:
 8006 044240                          LET @TEMPO :B= #55
      044240  112777  000055  134230                                            MOVB    #55,@TEMPO
 8007 044246                       ENDIF
      044246                                                                $52037:
 8008
 8009                        ;PUT EXPECTED SIGN INTO DS.EXP
 8010
 8011 044246                       LET TEMPO := #DS.EXP + DATA01
      044246  012767  064606  134222                                            MOV     #DS.EXP,TEMPO
      044254  066767  134304  134214                                            ADD     DATA01,TEMPO
 8012 044262                       IF SIGNO EQ #0 OR #ZBIT SETIN CC.EXP THEN
      044262  005767  134162                                                    TST     SIGNO
      044266  001404                                                            BEQ     $52040
      044270  032767  000004  134224                                            BIT     #ZBIT,CC.EXP
      044276  001404                                                            BEQ     $52041
      044300                                                                $52040:
 8013 044300                          LET @TEMPO :B= #53
      044300  112777  000053  134170                                            MOVB    #53,@TEMPO
 8014 044306                       ELSE
      044306  000403                                                            BR      $52042
      044310                                                                $52041:
 8015 044310                          LET @TEMPO :B= #55
      044310  112777  000055  134160                                            MOVB    #55,@TEMPO
 8016 044316                       ENDIF
      044316                                                                $52042:
 8017 044316                       IFB DATA02 LO DATA00 AND #BIT07 NOTSETIN DATA02 AND SIGNO EQ #-1 THEN
      044316  126767  134244  134236                                            CMPB    DATA02,DATA00
      044324  103013                                                           BHIS    $52043
      044326  032767  000200  134232                                            BIT     #BIT07,DATA02
      044334  001007                                                            BNE     $52043
      044336  026727  134106  177777                                            CMP     SIGNO,#-1
```

```
      044344  001003                                                                    BNE     $52043
 8018 044346                              LET @TEMPO :B= #55
      044346  112777  000055  134122                                                   MOVB    #55,@TEMPO
 8019 044354                          ENDIF
      044354                                                                       $52043:
 8020 044354                          IF DATA01 EQ #0 THEN                  ;DST.LEN = 0
      044354  005767  134204                                                           TST     DATA01
      044360  001003                                                                   BNE     $52044
 8021 044362                              LET CC.EXP := CC.EXP CLR.BY #VBIT
      044362  042767  000002  134132                                                   BIC     #VBIT,CC.EXP
 8022 044370                          ENDIF
      044370                                                                       $52044:
 8023 044370                      ENDIF
      044370                                                                       $52035:
 8024 044370                      IF DATTYP EQ #50000 THEN              ;*** LEADING SEPARATE ***
      044370  026727  134070  050000                                                   CMP     DATTYP,#50000
      044376  001075                                                                   BNE     $52045
 8025 044400                          LET TEMPO := #DS.EXP + DATA01
      044400  012767  064606  134070                                                   MOV     #DS.EXP,TEMPO
      044406  066767  134152  134062                                                   ADD     DATA01,TEMPO
 8026 044414                          LET @TEMPO :B= #0                 ;RESTORE DS.EXP FROM TRAILING SEPARATE
      044414  105077  134056                                                          CLRB    @TEMPO
 8027 044420                          LET TEMPO := #DS.DST + DATA01
      044420  012767  064544  134050                                                   MOV     #DS.DST,TEMPO
      044426  066767  134132  134042                                                   ADD     DATA01,TEMPO
 8028 044434                          LET @TEMPO :B= #0                 ;RESTORE DS.DST FROM TRAILING SEPARATE
      044434  105077  134036                                                          CLRB    @TEMPO
 8029
 8030                          ;PUT SIGN INTO SOURCE
 8031
 8032 044440                          IF SIGNO EQ #0 THEN                   ;SIGN IS +
      044440  005767  134004                                                           TST     SIGNO
      044444  001004                                                                   BNE     $52046
 8033 044446                              LET DSBUF1-1 :B= #53
      044446  112767  000053  017763                                                   MOVB    #53,DSBUF1-1
 8034 044454                          ELSE                                  ;SIGN IS -
      044454  000403                                                                   BR      $52047
      044456                                                                   $52046:
 8035 044456                              LET DSBUF1-1 :B= #55
      044456  112767  000055  017753                                                   MOVB    #55,DSBUF1-1
 8036 044464                          ENDIF
      044464                                                                   $52047:
 8037 044464                          IF SIGNO EQ #0 OR #ZBIT SETIN CC.EXP THEN
      044464  005767  133760                                                           TST     SIGNO
      044470  001404                                                                   BEQ     $52050
      044472  032767  000004  134022                                                   BIT     #ZBIT,CC.EXP
      044500  001404                                                                   BEQ     $52051
      044502                                                                   $52050:
 8038 044502                              LET DS.EXP-1 :B= #53
      044502  112767  000053  020075                                                   MOVB    #53,DS.EXP-1
 8039 044510                          ELSE
      044510  000403                                                                   BR      $52052
      044512                                                                   $52051:
 8040 044512                              LET DS.EXP-1 :B= #55
      044512  112767  000055  020065                                                   MOVB    #55,DS.EXP-1
 8041 044520                          ENDIF
      044520                                                                   $52052:
```

ASHN

```
 8042 044520                            IFB DATA02 LO DATA00 AND #BIT07 NOTSETIN DATA02 AND SIGNO EQ #-1 THEN
      044520  126767  134042  134034                                                    CMPB    DATA02,DATA00
      044526  103013                                                                    BHIS    $52053
      044530  032767  000200  134030                                                    BIT     #BIT07,DATA02
      044536  001007                                                                    BNE     $52053
      044540  026727  133704  177777                                                    CMP     SIGNO,#-1
      044546  001003                                                                    BNE     $52053
 8043 044550                                LET DS.EXP-1 :B= #55
      044550  112767  000055  020027                                                    MOVB    #55,DS.EXP-1
 8044 044556                            ENDIF
      044556                                                                  $52053:
 8045 044556                            IF DATA01 EQ #0 THEN                ;DST.LEN = 0
      044556  005767  134002                                                    TST     DATA01
      044562  001003                                                            BNE     $52054
 8046 044564                                LET CC.EXP := CC.EXP CLR.BY #VBIT
      044564  042767  000002  133730                                            BIC     #VBIT,CC.EXP
 8047 044572                            ENDIF
      044572                                                          $52054:
 8048 044572                         ENDIF
      044572                                                          $52045:
 8049
 8050                                ;+
 8051                                ; SET UP THE GPR'S AND DO THE ASHN INSTRUCTION
 8052                                ;-
 8053
 8054 044572                            LET R0 := DATA00 SET.BY DATTYP
      044572  016700  133764                                                    MOV     DATA00,R0
      044576  056700  133662                                                    BIS     DATTYP,R0
 8055 044602                            LET R1 := #DSBUF1
      044602  012701  064440                                                    MOV     #DSBUF1,R1
 8056 044606                            LET R2 := DATA01 SET.BY DATTYP
      044606  016702  133752                                                    MOV     DATA01,R2
      044612  056702  133646                                                    BIS     DATTYP,R2
 8057 044616                            LET R3 := #DS.DST
      044616  012703  064544                                                    MOV     #DS.DST,R3
 8058 044622                            LET R4 := DATA02
      044622  016704  133740                                                    MOV     DATA02,R4
 8059 044626                            LET R5 := #-1
      044626  012705  177777                                                    MOV     #-1,R5
 8060 044632                            INLINE   <ASHN>
      044632  076056                                                           ASHN
 8061
 8062                                ;+
 8063                                ; CHECK THE RESULTS AND REPORT ERRORS
 8064                                ;-
 8065
 8066 044634                            LET CC := PSW
      044634  016767  133136  133656                                            MOV     PSW,CC
 8067 044642                            LET CC := CC CLR.BY #177760
      044642  042767  177760  133650                                            BIC     #177760,CC
 8068 044650                            IF CC NE CC.EXP THEN
      044650  026767  133644  133644                                            CMP     CC,CC.EXP
      044656  001402                                                           BEQ     $52055
 8069 044660                                CALL ERROR
      044660  004767  013014                                                    JSR     PC,ERROR
 8070 044664                            ENDIF
      044664                                                          $52055:
```

ASHN

```
 8071 044664                            IF R0 NE #0 OR R1 NE #0 OR R4 NE #0 THEN
      044664  005700                                                          TST     R0
      044666  001004                                                          BNE     $52056
      044670  005701                                                          TST     R1
      044672  001002                                                          BNE     $52056
      044674  005704                                                          TST     R4
      044676  001402                                                          BEQ     $52057
      044700                                                          $52056:
 8072 044700                                CALL ERROR
      044700  004767  012774                                                  JSR     PC,ERROR
 8073 044704                            ENDIF
      044704                                                          $52057:
 8074 044704                            LET TEMPO := DATA01 SET.BY DATTYP
      044704  016767  133654  133564                                          MOV     DATA01,TEMPO
      044712  056767  133546  133556                                          BIS     DATTYP,TEMPO
 8075 044720                            IF R2 NE TEMPO THEN
      044720  020267  133552                                                  CMP     R2,TEMPO
      044724  001402                                                          BEQ     $52060
 8076 044726                                CALL ERROR
      044726  004767  012746                                                  JSR     PC,ERROR
 8077 044732                            ENDIF
      044732                                                          $52060:
 8078 044732                            IF R3 NE #DS.DST THEN
      044732  020327  064544                                                  CMP     R3,#DS.DST
      044736  001402                                                          BEQ     $52061
 8079 044740                                CALL ERROR                  ;DS.EXP AND DS.DST DO NOT MATCH
      044740  004767  012734                                                  JSR     PC,ERROR
 8080 044744                            ENDIF
      044744                                                          $52061:
 8081                                   ;COMPARE DS.EXP WITH DS.DST
 8082 044744                               LET ERRFLG := #0                 ;CLEAR ERROR FLAG
      044744  005067  133506                                                  CLR     ERRFLG
 8083 044750                               LET R0 := #41
      044750  012700  000041                                                  MOV     #41,R0
 8084 044754                               LET R1 := #DS.DST-1
      044754  012701  064543                                                  MOV     #DS.DST-1,R1
 8085 044760                               LET R2 := #DS.EXP-1
      044760  012702  064605                                                  MOV     #DS.EXP-1,R2
 8086 044764                               WHILE R0 NE #0 AND ERRFLG EQ #0 DO
      044764                                                          $52062:
      044764  005700                                                          TST     R0
      044766  001412                                                          BEQ     $52063
      044770  005767  133462                                                  TST     ERRFLG
      044774  001007                                                          BNE     $52063
 8087 044776                                  IFB (R1)+ NE (R2)+ THEN
      044776  122122                                                          CMPB    (R1)+,(R2)+
      045000  001403                                                          BEQ     $52064
 8088 045002                                     LET ERRFLG := #-1
      045002  012767  177777  133446                                          MOV     #-1,ERRFLG
 8089 045010                                  ENDIF
      045010                                                          $52064:
 8090 045010                                  LET R0 := R0 - #1
      045010  005300                                                          DEC     R0
 8091 045012                               ENDDO
      045012  000764                                                          BR      $52062
      045014                                                          $52063:
 8092 045014                               IF ERRFLG NE #0 THEN
```

ASHN

```
      045014  005767  133436                                                    TST     ERRFLG
      045020  001402                                                            BEQ     $52065
 8093 045022                          CALL ERROR
      045022  004767  012652                                                    JSR     PC,ERROR
 8094 045026                          ENDIF
      045026                                                          $52065:
 8095                                ;ENDINC
 8096 045026                          INLINE  <JMP  $5601>
      045026  000167  176202                                                    JMP  $5601
 8097 045032                          INLINE  <$5603:>
      045032                                                                    $5603:
 8098
 8099 045032                          INCR R0 FROM #DS.EXP-1 TO #DS.EXP+37 BY #1
      045032  012700  064605                                                    MOV     #DS.EXP-1,R0
      045036  000401                                                            BR      $52066
      045040                                                          $52067:
      045040  005200                                                            INC     R0
      045042                                                          $52066:
      045042  020027  064645                                                    CMP     R0,#DS.EXP+37
      045046  003002                                                            BGT     $52070
 8100 045050                          LET (R0) :B= #0                           ;CLEAR DS.DST
      045050  105010                                                            CLRB    (R0)
 8101 045052                          ENDINC
      045052  000772                                                            BR      $52067
      045054                                                          $52070:
 8102 045054                          LET SIGNO := COMP SIGNO                   ;REPEAT THE TEST CASE WITH A NEGATIVE
      045054  005167  133370                                                    COM     SIGNO
 8103                                ;  SIGN BEFORE MOVING ON TO THE NEXT ONE
 8104 045060                          IF SIGNO EQ #0 THEN
      045060  005767  133364                                                    TST     SIGNO
      045064  001005                                                            BNE     $52071
 8105 045066                              LET TSTCAS := TSTCAS + #6
      045066  062767  000006  133450                                            ADD     #6,TSTCAS
 8106 045074                              LET COUNT := COUNT - #1
      045074  005367  133370                                                    DEC     COUNT
 8107 045100                          ENDIF
      045100                                                          $52071:
 8108                                ;UNTIL COUNT EQ #0
 8109 045100                          IF COUNT NE #0 THEN
      045100  005767  133364                                                    TST     COUNT
      045104  001402                                                            BEQ     $52072
 8110 045106                              INLINE   <JMP   $5056>
      045106  000167  175256                                                    JMP   $5056
 8111 045112                          ENDIF
      045112                                                          $52072:
 8112
 8113                                ;+
 8114                                ; DO ONE INLINE CASE - ASHNI
 8115                                ;-
 8116
 8117 045112                          CALL CLRDSB
      045112  004767  011556                                                    JSR     PC,CLRDSB
 8118
 8119                                ;LOAD SRC STRING INTO DSBUF1
 8120
 8121 045116                          LET R0 := #DSBUF1-1
      045116  012700  064437                                                    MOV     #DSBUF1-1,R0
```

ASHN

```
8122 045122                             LET (R0)+ :B= #53                       ;LEADING SEPARATE SIGN OF +
     045122  112720  000053                                                                MOVB    #53,(R0)+
8123 045126                             LET (R0)+ :B= #9.                       ;PUT SRC MAGNITUDE OF 9999
     045126  112720  000011                                                                MOVB    #9.,(R0)+
8124 045132                             LET (R0)+ :B= #9.                       ;  INTO DSBUF1
     045132  112720  000011                                                                MOVB    #9.,(R0)+
8125 045136                             LET (R0)+ :B= #9.
     045136  112720  000011                                                                MOVB    #9.,(R0)+
8126 045142                             LET (R0) :B= #9.
     045142  112710  000011                                                                MOVB    #9.,(R0)
8127 045146                             LET R0 := #0
     045146  005000                                                                        CLR     R0
8128 045150                             LET R1 := #0
     045150  005001                                                                        CLR     R1
8129 045152                             LET R2 := #0
     045152  005002                                                                        CLR     R2
8130 045154                             LET R3 := #0
     045154  005003                                                                        CLR     R3
8131 045156                             LET R4 := #0
     045156  005004                                                                        CLR     R4
8132 045160                             LET R5 := #0
     045160  005005                                                                        CLR     R5
8133 045162                             INLINE   <ASHNI>
     045162  076156                                                                       ASHNI
8134 045164                             INLINE   <.WORD  TASHNI>
     045164  045356                                                                       .WORD   TASHNI
8135 045166                             INLINE   <.WORD  TASHNI+4>
     045166  045362                                                                       .WORD   TASHNI+4
8136 045170                             INLINE   <.WORD  000002>                ;SHIFT COUNT = 2, RND DIGIT = 0
     045170  000002                                                                       .WORD   000002
8137 045172                             LET CC := PSW
     045172  016767  132600  133320                                                      MOV     PSW,CC
8138 045200                             LET CC := CC CLR.BY #177760
     045200  042767  177760  133312                                                      BIC     #177760,CC
8139 045206                             IF CC NE #6 THEN                        ;VBIT SHOULD BE SET
     045206  026727  133306  000006                                                      CMP     CC,#6
     045214  001402                                                                     BEQ     $52073
8140 045216                                 CALL ERROR
     045216  004767  012456                                                             JSR     PC,ERROR
8141 045222                             ENDIF
     045222                                                                  $52073:
8142 045222                             IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN
     045222  005700                                                                     TST     R0
     045224  001004                                                                     BNE     $52074
     045226  005701                                                                     TST     R1
     045230  001002                                                                     BNE     $52074
     045232  005702                                                                     TST     R2
     045234  001402                                                                     BEQ     $52075
     045236                                                                  $52074:
8143 045236                                 CALL ERROR
     045236  004767  012436                                                             JSR     PC,ERROR
8144 045242                             ENDIF
     045242                                                                  $52075:
8145 045242                             IF R3 NE #0 OR R4 NE #0 OR R5 NE #0 THEN
     045242  005703                                                                     TST     R3
     045244  001004                                                                     BNE     $52076
     045246  005704                                                                     TST     R4
```

```
        045250  001002                                                          BNE     $52076
        045252  005705                                                          TST     R5
        045254  001402                                                          BEQ     $52077
        045256                                                          $52076:
   8146 045256                          CALL ERROR
        045256  004767  012416                                                  JSR     PC,ERROR
   8147 045262                        ENDIF
        045262                                                          $52077:
   8148                              ;CHECK DS.DST
   8149 045262                        LET R0 := #DS.DST
        045262  012700  064544                                                  MOV     #DS.DST,R0
   8150 045266                        IFB (R0)+ NE #173 THEN
        045266  122027  000173                                                  CMPB    (R0)+,#173
        045272  001402                                                          BEQ     $52100
   8151 045274                          CALL ERROR
        045274  004767  012400                                                  JSR     PC,ERROR
   8152 045300                        ENDIF
        045300                                                          $52100:
   8153 045300                        IFB (R0)+ NE #60 THEN
        045300  122027  000060                                                  CMPB    (R0)+,#60
        045304  001402                                                          BEQ     $52101
   8154 045306                          CALL ERROR
        045306  004767  012366                                                  JSR     PC,ERROR
   8155 045312                        ENDIF
        045312                                                          $52101:
   8156 045312                        LET ERRFLG := #0
        045312  005067  133140                                                  CLR     ERRFLG
   8157 045316                        WHILE ERRFLG EQ #0 AND R0 NE #DS.DST+37 DO
        045316                                                          $52102:
        045316  005767  133134                                                  TST     ERRFLG
        045322  001013                                                          BNE     $52103
        045324  020027  064603                                                  CMP     R0,#DS.DST+37
        045330  001410                                                          BEQ     $52103
   8158 045332                          IFB (R0)+ NE #0 THEN
        045332  105720                                                          TSTB    (R0)+
        045334  001405                                                          BEQ     $52104
   8159 045336                            CALL ERROR
        045336  004767  012336                                                  JSR     PC,ERROR
   8160 045342                            LET ERRFLG := #-1
        045342  012767  177777  133106                                          MOV     #-1,ERRFLG
   8161 045350                          ENDIF
        045350                                                          $52104:
   8162 045350                        ENDDO
        045350  000762                                                          BR      $52102
        045352                                                          $52103:
   8163 045352                        INLINE . <JMP   CIS6END>
        045352  000167  000010                                                  JMP     CIS6END
   8164
   8165                              ;+
   8166                              ; INLINE TEST CASE
   8167                              ;-
   8168
   8169 045356  050004               TASHNI: .WORD   050004          ;SRC.DSCR, LEADING SEPARATE, SRC.LEN = 4
   8170 045360  064440                       .WORD   DSBUF1          ;SRC.DSCR
   8171 045362  030002                       .WORD   030002          ;DST.DSCR, LEADING OVERPUNCH, DST.LEN = 2
   8172 045364  064544                       .WORD   DS.DST          ;DST.DSCR
   8173
```

ASHN

```
 8174
 8175
 8176 045366                    INLINE   <CIS6END:>
      045366                                                    CIS6END:
 8177
 8178
```

ASHN

```
                                ;*****************************************************************
                                ;*****************************************************************
                                ;++
                                ;                C I S 7     M O D U L E
                                ;
                                ;  THIS MODULE TESTS THE PACKED DATA TYPE ARITHMETIC
                                ;  INSTRUCTIONS: ADDP, SUBP, CMPP, ASHP.
                                ;--
                                ;*****************************************************************
                                ;*****************************************************************


                                .SBTTL  CIS7 INITIALIZATION
                                ;*****************************************************************
                                ;
                                ;  CIS7 INITIALIZATION
                                ;
                                ;*****************************************************************

                                ;IF SWR<5:0> ARE CLEARED OR IF SWR<3> IS SET THEN EXECUTE
                                ;  THE CIS7 MODULE; OTHERWISE, SKIP TO THE NEXT CIS MODULE.

045366                             INLINE   <SCOPE>                                    SCOPE
045366  000004
045370                             INLINE <CKSWR>                                      CKSWR
045370  104406
045372                             LET TEMPO := @SWR CLR.BY #177700                    MOV     @SWR,TEMPO
045372  017767  133030  133076                                                         BIC     #177700,TEMPO
045400  042767  177700  133070
045406                             IF TEMPO NE #0 THEN                                 TST     TEMPO
045406  005767  133064                                                                 BEQ     $52105
045412  001406
045414                                 IF #BIT3 NOTSETIN TEMPO THEN                    BIT     #BIT3,TEMPO
045414  032767  000010  133054                                                         BNE     $52106
045422  001002
045424                                     INLINE   <JMP   CIS7END>                    JMP   CIS7END
045424  000167  005336
045430                                 ENDIF                                $52106:
045430
045430                             ENDIF                                $52105:
045430
045430                             LET CTL := #7                ;CONTROL CHIP NUMBER   MOV     #7,CTL
045430  012767  000007  133110
045436                             LET ERRFLG := #0     ;INITIALIZE ERROR FLAG        CLR     ERRFLG
045436  005067  133014
045442                             LET PSW := #340                                     MOV     #340,PSW
045442  012767  000340  132326
045450                             LET ERR := #1                                       MOV     #1,ERR
045450  012767  000001  133074
8215
```

```
                                .SBTTL  ADDP
                                ;*****************************************************************
                                ;
                                ;       ADDP TEST
                                ;
                                ;*****************************************************************

045456                          ROUTINE XADDP
045456                                                                          XADDP:
                                   INLINE <SCOPE>
045456  000004                                                                  SCOPE
                                   INLINE <NOP>
045460  000240                                                                  NOP
                                   LET $TIMES := #20            ;ITERATION COUNT
045462  012767  000020  132714                                                  MOV     #20,$TIMES
                                   LET COUNT := TADD            ;# OF TEST CASES
045470  016767  017554  132772                                                  MOV     TADD,COUNT
                                   LET TSTCAS := #TADD+2                ;POINT TO FIRST TEST CASE
045476  012767  065252  133040                                                  MOV     #TADD+2,TSTCAS

                                ;+
                                ;       CLR R0.EXT TO R3.EXP
                                ;-

                                   INCR  R0 FROM #0 TO #6 BY #2
045504  005000                                                                  CLR     R0
045506  000402                                                                  BR      $52111
045510                                                                  $52112:
045510  062700  000002                                                          ADD     #2,R0
045514                                                                  $52111:
045514  020027  000006                                                          CMP     R0,#6
045520  003003                                                                  BGT     $52113
                                       INLINE <CLR R0.EXP(R0)>
045522  005060  000524                                                          CLR R0.EXP(R0)
                                   ENDINC
045526  000770                                                                  BR      $52112
045530                                                                  $52113:

                                ;REPEAT OUTER LOOP FOR TEST CASES

                                   INLINE <$5070:>
045530                                                                  $5070:
                                   CALL CLRDSB                  ;CLR DS BUFFERS
045530  004767  011140                                                          JSR     PC,CLRDSB

                                ;+
                                ;       LOAD DATAXX AND CC.EXP FROM TEST CASE
                                ;-

                                   LET R0 := TSTCAS
045534  016700  133004                                                          MOV     TSTCAS,R0
                                   LET DATA00 := (R0)+          ;SRC1 LENGTH
045540  012067  133016                                                          MOV     (R0)+,DATA00
                                   LET DATA01 := (R0)+          ;POINTER TO SRC1 DATA
045544  012067  133014                                                          MOV     (R0)+,DATA01
                                   LET DATA02 := (R0)+          ;SRC2 LENGTH
045550  012067  133012                                                          MOV     (R0)+,DATA02
```

```
8252 045554                             LET DATA03 := (R0)+          ;POINTER TO SRC2 DATA
     045554  012067  133010                                                  MOV     (R0)+,DATA03
8253 045560                             LET DATA04 := (R0)+          ;DEST LENGTH
     045560  012067  133006                                                  MOV     (R0)+,DATA04
8254 045564                             LET DATA05 := (R0)+          ;POINTER TO DS.EXP DATA (SUM)
     045564  012067  133004                                                  MOV     (R0)+,DATA05
8255 045570                             LET CC.EXP := (R0)+          ;CC.EXP (SUM)
     045570  012067  132726                                                  MOV     (R0)+,CC.EXP
8256
8257                                ;+
8258                                ;       PACK BUFFERS FOR SRC1,SRC2, & DEST
8259                                ;-
8260
8261 045574                             LET R5 := #CHRSET                    ;R5 IS OPERAND STACK FOR ROUTINE
     045574  012705  064036                                                  MOV     #CHRSET,R5
8262 045600                             CALL PACK IN <#DSBUF1,DATA01,DATA00> ;SRC1
     045600  010546                                                          MOV     R5,-(SP)
     045602  016745  132754                                                  MOV     DATA00,-(R5)
     045606  016745  132752                                                  MOV     DATA01,-(R5)
     045612  012745  064440                                                  MOV     #DSBUF1,-(R5)
     045616  004767  011432                                                  JSR     PC,PACK
     045622  012605                                                          MOV     (SP)+,R5
8263 045624                             CALL PACK IN <#DSBUF2,DATA03,DATA02> ;SRC2
     045624  010546                                                          MOV     R5,-(SP)
     045626  016745  132734                                                  MOV     DATA02,-(R5)
     045632  016745  132732                                                  MOV     DATA03,-(R5)
     045636  012745  064502                                                  MOV     #DSBUF2,-(R5)
     045642  004767  011406                                                  JSR     PC,PACK
     045646  012605                                                          MOV     (SP)+,R5
8264 045650                             CALL PACK IN <#DS.EXP,DATA05,DATA04> ;DEST
     045650  010546                                                          MOV     R5,-(SP)
     045652  016745  132714                                                  MOV     DATA04,-(R5)
     045656  016745  132712                                                  MOV     DATA05,-(R5)
     045662  012745  064606                                                  MOV     #DS.EXP,-(R5)
     045666  004767  011362                                                  JSR     PC,PACK
     045672  012605                                                          MOV     (SP)+,R5
8265
8266                                ;+
8267                                ;       LOAD GPR'S FROM DATAXX & ADD DATA TYPE
8268                                ;-
8269
8270 045674                             LET R0 := DATA00 SET.BY #60000
     045674  016700  132662                                                  MOV     DATA00,R0
     045700  052700  060000                                                  BIS     #60000,R0
8271 045704                             LET R1 := #DSBUF1
     045704  012701  064440                                                  MOV     #DSBUF1,R1
8272 045710                             LET R2 := DATA02 SET.BY #60000
     045710  016702  132652                                                  MOV     DATA02,R2
     045714  052702  060000                                                  BIS     #60000,R2
8273 045720                             LET R3 := #DSBUF2
     045720  012703  064502                                                  MOV     #DSBUF2,R3
8274 045724                             LET R4 := DATA04 SET.BY #60000
     045724  016704  132642                                                  MOV     DATA04,R4
     045730  052704  060000                                                  BIS     #60000,R4
8275 045734                             LET R5 := #DS.DST
     045734  012705  064544                                                  MOV     #DS.DST,R5
8276
```

```
 8277                                   ;+
 8278                                   ;       FINISH SET UP & DO CIS INSTRUCTION
 8279                                   ;-
 8280
 8281
 8282 045740                               LET R4.EXP := R4
      045740  010467  132570                                                    MOV     R4,R4.EXP
 8283 045744                               LET R5.EXP := R5
      045744  010567  132566                                                    MOV     R5,R5.EXP
 8284 045750                               INLINE <ADDP>
      045750  076070                                                           ADDP
 8285
 8286                                   ;+
 8287                                   ;       CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.
 8288                                   ;-
 8289
 8290 045752                               LET CC := PSW
      045752  016767  132020  132540                                            MOV     PSW,CC
 8291 045760                               LET CC := CC CLR.BY #177760
      045760  042767  177760  132532                                            BIC     #177760,CC
 8292 045766                               IF CC NE CC.EXP THEN
      045766  026767  132526  132526                                            CMP     CC,CC.EXP
      045774  001402                                                           BEQ     $52114
 8293 045776                                   CALL ERROR
      045776  004767  011676                                                   JSR     PC,ERROR
 8294 046002                               ENDIF
      046002                                                          $52114:
 8295 046002                               IF R0 NE R0.EXP THEN
      046002  020067  132516                                                    CMP     R0,R0.EXP
      046006  001402                                                           BEQ     $52115
 8296 046010                                   CALL ERROR
      046010  004767  011664                                                   JSR     PC,ERROR
 8297 046014                               ENDIF
      046014                                                          $52115:
 8298 046014                               IF R1 NE R1.EXP THEN
      046014  020167  132506                                                    CMP     R1,R1.EXP
      046020  001402                                                           BEQ     $52116
 8299 046022                                   CALL ERROR
      046022  004767  011652                                                   JSR     PC,ERROR
 8300 046026                               ENDIF
      046026                                                          $52116:
 8301 046026                               IF R2 NE R2.EXP THEN
      046026  020267  132476                                                    CMP     R2,R2.EXP
      046032  001402                                                           BEQ     $52117
 8302 046034                                   CALL ERROR
      046034  004767  011640                                                   JSR     PC,ERROR
 8303 046040                               ENDIF
      046040                                                          $52117:
 8304 046040                               IF R3 NE R3.EXP THEN
      046040  020367  132466                                                    CMP     R3,R3.EXP
      046044  001402                                                           BEQ     $52120
 8305 046046                                   CALL ERROR
      046046  004767  011626                                                   JSR     PC,ERROR
 8306 046052                               ENDIF
      046052                                                          $52120:
 8307 046052                               IF R4 NE R4.EXP THEN
      046052  020467  132456                                                    CMP     R4,R4.EXP
```

ADDP

```
      046056  001402                                                    BEQ     $52121
 8308 046060                          CALL ERROR
      046060  004767  011614                                            JSR     PC,ERROR
 8309 046064                        ENDIF
      046064                                                    $52121:
 8310 046064                        IF R5 NE R5.EXP THEN
      046064  020567  132446                                            CMP     R5,R5.EXP
      046070  001402                                                    BEQ     $52122
 8311 046072                          CALL ERROR
      046072  004767  011602                                            JSR     PC,ERROR
 8312 046076                        ENDIF
      046076                                                    $52122:
 8313
 8314                           ;+
 8315                           ;       CHECK RESULTS VS EXPECTED RESULTS
 8316                           ;-
 8317
 8318 046076                        LET R0 := #DS.DST
      046076  012700  064544                                            MOV     #DS.DST,R0
 8319 046102                        LET R1 := #DS.EXP
      046102  012701  064606                                            MOV     #DS.EXP,R1
 8320 046106                        LET ERRFLG := #0
      046106  005067  132344                                            CLR     ERRFLG
 8321 046112                        WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO
      046112                                                    $52123:
      046112  020027  064604                                            CMP     R0,#DS.DST+40
      046116  001410                                                    BEQ     $52124
      046120  005767  132332                                            TST     ERRFLG
      046124  001005                                                    BNE     $52124
 8322 046126                          IFB (R0)+ NE (R1)+ THEN
      046126  122021                                                    CMPB    (R0)+,(R1)+
      046130  001402                                                    BEQ     $52125
 8323 046132                            CALL ERROR
      046132  004767  011542                                            JSR     PC,ERROR
 8324 046136                          ENDIF
      046136                                                    $52125:
 8325 046136                        ENDDO
      046136  000765                                                    BR      $52123
      046140                                                    $52124:
 8326 046140                        LET TSTCAS := TSTCAS + #22
      046140  062767  000022  132376                                    ADD     #22,TSTCAS
 8327 046146                        LET COUNT := COUNT - #1
      046146  005367  132316                                            DEC     COUNT
 8328                           ;UNTIL COUNT EQ #0
 8329 046152                        IF COUNT NE #0 THEN
      046152  005767  132312                                            TST     COUNT
      046156  001402                                                    BEQ     $52126
 8330 046160                          INLINE <JMP $5070>
      046160  000167  177344                                            JMP $5070
 8331 046164                        ENDIF
      046164                                                    $52126:
 8332
 8333                           ;+
 8334                           ;       INLINE CASE - ADDPI, USING UNSIGNED PACKED DATA
 8335                           ;-
 8336
 8337 046164                        CALL CLRDSB
```

```
ADDP

      046164  004767  010504                                                    JSR     PC,CLRDSB
   8338
   8339                                ;+
   8340                                ;       LOAD DATAXX AND CC.EXP FROM TEST CASE
   8341                                ;-
   8342
   8343 046170                         LET R0 := #TADD+2
      046170  012700  065252                                                    MOV     #TADD+2,R0
   8344 046174                         LET DATA00 := (R0)+         ;SRC1 LENGTH
      046174  012067  132362                                                    MOV     (R0)+,DATA00
   8345 046200                         LET DATA01 := (R0)+         ;POINTER TO SRC1 DATA
      046200  012067  132360                                                    MOV     (R0)+,DATA01
   8346 046204                         LET DATA02 := (R0)+         ;SRC2 LENGTH
      046204  012067  132356                                                    MOV     (R0)+,DATA02
   8347 046210                         LET DATA03 := (R0)+         ;POINTER TO SRC2 DATA
      046210  012067  132354                                                    MOV     (R0)+,DATA03
   8348 046214                         LET DATA04 := (R0)+         ;DEST LENGTH
      046214  012067  132352                                                    MOV     (R0)+,DATA04
   8349 046220                         LET DATA05 := (R0)+         ;POINTER TO DS.EXP DATA (SUM)
      046220  012067  132350                                                    MOV     (R0)+,DATA05
   8350
   8351                                ;+
   8352                                ;       PACK BUFFERS FOR SRC1,SRC2, & DEST
   8353                                ;-
   8354
   8355 046224                         LET R5 := #CHRSET                   ;R5 IS OPERAND STACK FOR ROUTINE
      046224  012705  064036                                                    MOV     #CHRSET,R5
   8356 046230                         CALL PACK IN <#DSBUF1,DATA01,DATA00> ;SRC1
      046230  010546                                                            MOV     R5,-(SP)
      046232  016745  132324                                                    MOV     DATA00,-(R5)
      046236  016745  132322                                                    MOV     DATA01,-(R5)
      046242  012745  064440                                                    MOV     #DSBUF1,-(R5)
      046246  004767  011002                                                    JSR     PC,PACK
      046252  012605                                                           MOV     (SP)+,R5
   8357 046254                         CALL PACK IN <#DSBUF2,DATA03,DATA02> ;SRC2
      046254  010546                                                            MOV     R5,-(SP)
      046256  016745  132304                                                    MOV     DATA02,-(R5)
      046262  016745  132302                                                    MOV     DATA03,-(R5)
      046266  012745  064502                                                    MOV     #DSBUF2,-(R5)
      046272  004767  010756                                                    JSR     PC,PACK
      046276  012605                                                           MOV     (SP)+,R5
   8358 046300                         CALL PACK IN <#DS.EXP,DATA05,DATA04> ;DEST
      046300  010546                                                            MOV     R5,-(SP)
      046302  016745  132264                                                    MOV     DATA04,-(R5)
      046306  016745  132262                                                    MOV     DATA05,-(R5)
      046312  012745  064606                                                    MOV     #DS.EXP,-(R5)
      046316  004767  010732                                                    JSR     PC,PACK
      046322  012605                                                           MOV     (SP)+,R5
   8359
   8360 046324                         CALL CLRREG                 ;ENSURE GPR'S 0-5 ARE ZERO
      046324  004767  010110                                                    JSR     PC,CLRREG
   8361
   8362 046330                         INLINE <ADDPI>
      046330  076170                                                           ADDPI
   8363 046332                         INLINE <.WORD TADDPI>
      046332  066136                                                           .WORD TADDPI
   8364 046334                         INLINE <.WORD TADDPI+4>
```

ADDP

```
      046334  066142                                                                .WORD TADDPI+4
 8365 046336                          INLINE <.WORD TADDPI+10>
      046336  066146                                                                .WORD TADDPI+10
 8366 046340                          LET CC := PSW
      046340  016767  131432  132152                                                MOV     PSW,CC
 8367 046346                          LET CC := CC CLR.BY #177760
      046346  042767  177760  132144                                                BIC     #177760,CC
 8368 046354                          IF CC NE #0 THEN
      046354  005767  132140                                                        TST     CC
      046360  001402                                                               BEQ     $52127
 8369 046362                              CALL ERROR
      046362  004767  011312                                                        JSR     PC,ERROR
 8370 046366                          ENDIF
      046366                                                                $52127:
 8371
 8372                               ;+
 8373                               ;       ENSURE GPR'S STILL ZERO
 8374                               ;-
 8375
 8376 046366                          IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN
      046366  005700                                                                TST     R0
      046370  001004                                                               BNE     $52130
      046372  005701                                                               TST     R1
      046374  001002                                                               BNE     $52130
      046376  005702                                                               TST     R2
      046400  001402                                                               BEQ     $52131
      046402                                                                $52130:
 8377 046402                              CALL ERROR
      046402  004767  011272                                                        JSR     PC,ERROR
 8378 046406                          ENDIF
      046406                                                                $52131:
 8379 046406                          IF R3 NE #0 OR R4 NE #0 OR  R5 NE #0 THEN
      046406  005703                                                                TST     R3
      046410  001004                                                               BNE     $52132
      046412  005704                                                               TST     R4
      046414  001002                                                               BNE     $52132
      046416  005705                                                               TST     R5
      046420  001402                                                               BEQ     $52133
      046422                                                                $52132:
 8380 046422                              CALL ERROR
      046422  004767  011252                                                        JSR     PC,ERROR
 8381 046426                          ENDIF
      046426                                                                $52133:
 8382
 8383                               ;+
 8384                               ;       CHECK RESULTS
 8385                               ;-
 8386
 8387 046426                          LET R0 := #DS.DST
      046426  012700  064544                                                        MOV     #DS.DST,R0
 8388 046432                          LET R1 := #DS.EXP               ;EXPECTED DATA
      046432  012701  064606                                                        MOV     #DS.EXP,R1
 8389 046436                          LET R2 := TADDPI+10 CLR.BY #70000;GET STRING LENGTH
      046436  016702  017504                                                        MOV     TADDPI+10,R2
      046442  042702  070000                                                        BIC     #70000,R2
 8390 046446                          INLINE       <ASR    R2>                ;LENGTH/2 (PACKING FACTOR)
      046446  006202                                                                ASR     R2
```

ADDP

```
 8391 046450                    LET R2 := R2 + #DS.EXP                  ;R2 POINTS TO MSD/SIGN
      046450  062702  064606                                                    ADD     #DS.EXP,R2
 8392 046454                    LET (R2) :B= (R2) SET.BY #17 ;MAKE IT UNSIGNED
      046454  152712  000017                                                   BISB    #17,(R2)
 8393 046460                    LET ERRFLG := #0
      046460  005067  131772                                                   CLR     ERRFLG
 8394 046464                    WHILE R0 NE #DS.DST+37 AND ERRFLG EQ #0 DO
      046464                                                            $52134:
      046464  020027  064603                                                   CMP     R0,#DS.DST+37
      046470  001410                                                          BEQ     $52135
      046472  005767  131760                                                   TST     ERRFLG
      046476  001005                                                          BNE     $52135
 8395 046500                        IFB (R0)+ NE (R1)+ THEN
      046500  122021                                                          CMPB    (R0)+,(R1)+
      046502  001402                                                          BEQ     $52136
 8396 046504                            CALL ERROR
      046504  004767  011170                                                   JSR     PC,ERROR
 8397 046510                        ENDIF
      046510                                                            $52136:
 8398 046510                    ENDDO
      046510  000765                                                          BR      $52134
      046512                                                            $52135:
 8399
 8400
```

```
 8402                                .SBTTL  SUBP
 8403                                ;*****************************************************************
 8404                                ;
 8405                                ;       SUBP TEST
 8406                                ;
 8407                                ;*****************************************************************
 8408
 8409 046512                         ROUTINE XSUBP
      046512                                                                          XSUBP:
 8410 046512                            INLINE <SCOPE>
      046512  000004                                                                         SCOPE
 8411 046514                            INLINE <NOP>
      046514  000240                                                                        NOP
 8412 046516                            LET $TIMES := #20           ;ITERATION COUNT
      046516  012767  000020  131660                                                        MOV     #20,$TIMES
 8413 046524                            LET COUNT := TADD           ;# OF TEST CASES
      046524  016767  016520  131736                                                        MOV     TADD,COUNT
 8414 046532                            LET TSTCAS := #TADD+2               ;POINT TO FIRST TEST CASE
      046532  012767  065252  132004                                                        MOV     #TADD+2,TSTCAS
 8415
 8416                                ;+
 8417                                ;       CLR R0.EXT TO R3.EXP
 8418                                ;-
 8419
 8420 046540                            INCR  R0 FROM #0 TO #6 BY #2
      046540  005000                                                                        CLR     R0
      046542  000402                                                                        BR      $52141
      046544                                                                    $52142:
      046544  062700  000002                                                                ADD     #2,R0
      046550                                                                    $52141:
      046550  020027  000006                                                                CMP     R0,#6
      046554  003003                                                                       BGT     $52143
 8421 046556                                INLINE <CLR R0.EXP(R0)>
      046556  005060  000524                                                                CLR R0.EXP(R0)
 8422 046562                            ENDINC
      046562  000770                                                                       BR      $52142
      046564                                                                    $52143:
 8423
 8424                                ;REPEAT OUTER LOOP FOR TEST CASES
 8425
 8426 046564                            INLINE <$5071:>
      046564                                                                             $5071:
 8427 046564                            CALL CLRDSB                 ;CLR DS BUFFERS
      046564  004767  010104                                                             JSR     PC,CLRDSB
 8428
 8429                                ;+
 8430                                ;       LOAD DATAXX AND CC.EXP FROM TEST CASE
 8431                                ;-
 8432
 8433 046570                            LET R0 := TSTCAS
      046570  016700  131750                                                            MOV     TSTCAS,R0
 8434 046574                            LET DATA00 := (R0)+         ;SRC1 LENGTH
      046574  012067  131762                                                            MOV     (R0)+,DATA00
 8435 046600                            LET DATA01 := (R0)+         ;POINTER TO SRC1 DATA
      046600  012067  131760                                                            MOV     (R0)+,DATA01
 8436 046604                            LET DATA02 := (R0)+         ;SRC2 LENGTH
      046604  012067  131756                                                            MOV     (R0)+,DATA02
```

```
SUBP

8437 046610                             LET DATA03 := (R0)+          ;POINTER TO SRC2 DATA
     046610  012067  131754                                                  MOV     (R0)+,DATA03
8438 046614                             LET DATA04 := (R0)+          ;DEST LENGTH
     046614  012067  131752                                                  MOV     (R0)+,DATA04
8439 046620                             LET DATA05 := (R0)+          ;POINTER TO DS.EXP DATA (SUM)
     046620  012067  131750                                                  MOV     (R0)+,DATA05
8440 046624                             LET CC.EXP := (R0)+          ;CC.EXP (SUM)
     046624  012067  131672                                                  MOV     (R0)+,CC.EXP
8441 046630                             LET DATA05 := (R0)+          ;POINTER TO DS.EXP DATA (DIFF)
     046630  012067  131740                                                  MOV     (R0)+,DATA05
8442 046634                             LET CC.EXP := (R0)+          ;CC.EXP (DIFF)
     046634  012067  131662                                                  MOV     (R0)+,CC.EXP
8443
8444                                ;+
8445                                ;       PACK BUFFERS FOR SRC1,SRC2, & DEST
8446                                ;-
8447
8448 046640                             LET R5 := #CHRSET                    ;R5 IS OPERAND STACK FOR ROUTINE
     046640  012705  064036                                                  MOV     #CHRSET,R5
8449 046644                             CALL PACK IN <#DSBUF1,DATA01,DATA00> ;SRC1
     046644  010546                                                          MOV     R5,-(SP)
     046646  016745  131710                                                  MOV     DATA00,-(R5)
     046652  016745  131706                                                  MOV     DATA01,-(R5)
     046656  012745  064440                                                  MOV     #DSBUF1,-(R5)
     046662  004767  010366                                                  JSR     PC,PACK
     046666  012605                                                          MOV     (SP)+,R5
8450 046670                             CALL PACK IN <#DSBUF2,DATA03,DATA02> ;SRC2
     046670  010546                                                          MOV     R5,-(SP)
     046672  016745  131670                                                  MOV     DATA02,-(R5)
     046676  016745  131666                                                  MOV     DATA03,-(R5)
     046702  012745  064502                                                  MOV     #DSBUF2,-(R5)
     046706  004767  010342                                                  JSR     PC,PACK
     046712  012605                                                          MOV     (SP)+,R5
8451 046714                             CALL PACK IN <#DS.EXP,DATA05,DATA04> ;DEST
     046714  010546                                                          MOV     R5,-(SP)
     046716  016745  131650                                                  MOV     DATA04,-(R5)
     046722  016745  131646                                                  MOV     DATA05,-(R5)
     046726  012745  064606                                                  MOV     #DS.EXP,-(R5)
     046732  004767  010316                                                  JSR     PC,PACK
     046736  012605                                                          MOV     (SP)+,R5
8452
8453
8454                                ;+
8455                                ;       LOAD GPR'S FROM DATAXX & ADD DATA TYPE
8456                                ;-
8457
8458 046740                             LET R0 := DATA00 SET.BY #60000
     046740  016700  131616                                                  MOV     DATA00,R0
     046744  052700  060000                                                  BIS     #60000,R0
8459 046750                             LET R1 := #DSBUF1
     046750  012701  064440                                                  MOV     #DSBUF1,R1
8460 046754                             LET R2 := DATA02 SET.BY #60000
     046754  016702  131606                                                  MOV     DATA02,R2
     046760  052702  060000                                                  BIS     #60000,R2
8461 046764                             LET R3 := #DSBUF2
     046764  012703  064502                                                  MOV     #DSBUF2,R3
8462 046770                             LET R4 := DATA04 SET.BY #60000
```

SUBP

```
        046770  016704  131576                                                  MOV     DATA04,R4
        046774  052704  060000                                                  BIS     #60000,R4
   8463 047000                          LET R5 := #DS.DST
        047000  012705  064544                                                  MOV     #DS.DST,R5
   8464
   8465                                 ;+
   8466                                 ;       FINISH SET UP & DO CIS INSTRUCTION
   8467                                 ;-
   8468
   8469
   8470 047004                          LET R4.EXP := R4
        047004  010467  131524                                                  MOV     R4,R4.EXP
   8471 047010                          LET R5.EXP := R5
        047010  010567  131522                                                  MOV     R5,R5.EXP
   8472 047014                          INLINE <SUBP>
        047014  076071                                                          SUBP
   8473
   8474                                 ;+
   8475                                 ;       CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.
   8476                                 ;-
   8477
   8478 047016                          LET CC := PSW
        047016  016767  130754  131474                                          MOV     PSW,CC
   8479 047024                          LET CC := CC CLR.BY #177760
        047024  042767  177760  131466                                          BIC     #177760,CC
   8480 047032                          IF CC NE CC.EXP THEN
        047032  026767  131462  131462                                          CMP     CC,CC.EXP
        047040  001402                                                         BEQ     $52144
   8481 047042                              CALL ERROR
        047042  004767  010632                                                 JSR     PC,ERROR
   8482 047046                          ENDIF
        047046                                                          $52144:
   8483 047046                          IF R0 NE R0.EXP THEN
        047046  020067  131452                                                 CMP     R0,R0.EXP
        047052  001402                                                         BEQ     $52145
   8484 047054                              CALL ERROR
        047054  004767  010620                                                 JSR     PC,ERROR
   8485 047060                          ENDIF
        047060                                                          $52145:
   8486 047060                          IF R1 NE R1.EXP THEN
        047060  020167  131442                                                 CMP     R1,R1.EXP
        047064  001402                                                         BEQ     $52146
   8487 047066                              CALL ERROR
        047066  004767  010606                                                 JSR     PC,ERROR
   8488 047072                          ENDIF
        047072                                                          $52146:
   8489 047072                          IF R2 NE R2.EXP THEN
        047072  020267  131432                                                 CMP     R2,R2.EXP
        047076  001402                                                         BEQ     $52147
   8490 047100                              CALL ERROR
        047100  004767  010574                                                 JSR     PC,ERROR
   8491 047104                          ENDIF
        047104                                                          $52147:
   8492 047104                          IF R3 NE R3.EXP THEN
        047104  020367  131422                                                 CMP     R3,R3.EXP
        047110  001402                                                         BEQ     $52150
   8493 047112                              CALL ERROR
```

```
      047112  004767  010562                                                    JSR     PC,ERROR
 8494 047116                          ENDIF
      047116                                                              $52150:
 8495 047116                          IF R4 NE R4.EXP THEN
      047116  020467  131412                                                    CMP     R4,R4.EXP
      047122  001402                                                           BEQ     $52151
 8496 047124                              CALL ERROR
      047124  004767  010550                                                    JSR     PC,ERROR
 8497 047130                          ENDIF
      047130                                                              $52151:
 8498 047130                          IF R5 NE R5.EXP THEN
      047130  020567  131402                                                    CMP     R5,R5.EXP
      047134  001402                                                           BEQ     $52152
 8499 047136                              CALL ERROR
      047136  004767  010536                                                    JSR     PC,ERROR
 8500 047142                          ENDIF
      047142                                                              $52152:
 8501
 8502
 8503                            ;+
 8504                            ;       CHECK RESULTS VS EXPECTED RESULTS
 8505                            ;-
 8506
 8507 047142                          LET R0 := #DS.DST
      047142  012700  064544                                                    MOV     #DS.DST,R0
 8508 047146                          LET R1 := #DS.EXP
      047146  012701  064606                                                    MOV     #DS.EXP,R1
 8509 047152                          LET ERRFLG := #0
      047152  005067  131300                                                    CLR     ERRFLG
 8510 047156                          WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO
      047156                                                              $52153:
      047156  020027  064604                                                    CMP     R0,#DS.DST+40
      047162  001410                                                           BEQ     $52154
      047164  005767  131266                                                    TST     ERRFLG
      047170  001005                                                           BNE     $52154
 8511 047172                              IFB (R0)+ NE (R1)+ THEN
      047172  122021                                                           CMPB    (R0)+,(R1)+
      047174  001402                                                           BEQ     $52155
 8512 047176                                  CALL ERROR
      047176  004767  010476                                                    JSR     PC,ERROR
 8513 047202                              ENDIF
      047202                                                              $52155:
 8514 047202                          ENDDO
      047202  000765                                                           BR      $52153
      047204                                                              $52154:
 8515 047204                          LET TSTCAS := TSTCAS + #22
      047204  062767  000022  131332                                            ADD     #22,TSTCAS
 8516 047212                          LET COUNT := COUNT - #1
      047212  005367  131252                                                    DEC     COUNT
 8517                            ;UNTIL COUNT EQ #0
 8518 047216                          IF COUNT NE #0 THEN
      047216  005767  131246                                                    TST     COUNT
      047222  001402                                                           BEQ     $52156
 8519 047224                              INLINE <JMP $5071>
      047224  000167  177334                                                    JMP $5071
 8520 047230                          ENDIF
      047230                                                              $52156:
```

```
8521
8522                                    ;+
8523                                    ;       INLINE CASE - SUBPI, USING UNSIGNED PACKED DATA
8524                                    ;-
8525
8526 047230                             CALL CLRDSB
     047230  004767  007440                                                          JSR     PC,CLRDSB
8527
8528                                    ;+
8529                                    ;       LOAD DATAXX AND CC.EXP FROM TEST CASE
8530                                    ;-
8531
8532 047234                             LET R0 := #TADD+2
     047234  012700  065252                                                          MOV     #TADD+2,R0
8533 047240                             LET DATA00 := (R0)+             ;SRC1 LENGTH
     047240  012067  131316                                                          MOV     (R0)+,DATA00
8534 047244                             LET DATA01 := (R0)+             ;POINTER TO SRC1 DATA
     047244  012067  131314                                                          MOV     (R0)+,DATA01
8535 047250                             LET DATA02 := (R0)+             ;SRC2 LENGTH
     047250  012067  131312                                                          MOV     (R0)+,DATA02
8536 047254                             LET DATA03 := (R0)+             ;POINTER TO SRC2 DATA
     047254  012067  131310                                                          MOV     (R0)+,DATA03
8537 047260                             LET DATA04 := (R0)+             ;DEST LENGTH
     047260  012067  131306                                                          MOV     (R0)+,DATA04
8538 047264                             LET DATA05 := (R0)+             ;POINTER TO DS.EXP DATA (SUM)
     047264  012067  131304                                                          MOV     (R0)+,DATA05
8539 047270                             LET CC.EXP := (R0)+
     047270  012067  131226                                                          MOV     (R0)+,CC.EXP
8540 047274                             LET DATA05 := (R0)+             ;POINTER TO DS.EXP DATA (DIFFERENCE)
     047274  012067  131274                                                          MOV     (R0)+,DATA05
8541
8542                                ;+
8543                                ;       PACK BUFFERS FOR SRC1,SRC2, & DEST
8544                                ;-
8545
8546 047300                             LET R5 := #CHRSET                       ;R5 IS OPERAND STACK FOR ROUTINE
     047300  012705  064036                                                          MOV     #CHRSET,R5
8547 047304                             CALL PACK IN <#DSBUF1,DATA01,DATA00> ;SRC1
     047304  010546                                                                  MOV     R5,-(SP)
     047306  016745  131250                                                          MOV     DATA00,-(R5)
     047312  016745  131246                                                          MOV     DATA01,-(R5)
     047316  012745  064440                                                          MOV     #DSBUF1,-(R5)
     047322  004767  007726                                                          JSR     PC,PACK
     047326  012605                                                                  MOV     (SP)+,R5
8548 047330                             CALL PACK IN <#DSBUF2,DATA03,DATA02> ;SRC2
     047330  010546                                                                  MOV     R5,-(SP)
     047332  016745  131230                                                          MOV     DATA02,-(R5)
     047336  016745  131226                                                          MOV     DATA03,-(R5)
     047342  012745  064502                                                          MOV     #DSBUF2,-(R5)
     047346  004767  007702                                                          JSR     PC,PACK
     047352  012605                                                                  MOV     (SP)+,R5
8549 047354                             CALL PACK IN <#DS.EXP,DATA05,DATA04> ;DEST
     047354  010546                                                                  MOV     R5,-(SP)
     047356  016745  131210                                                          MOV     DATA04,-(R5)
     047362  016745  131206                                                          MOV     DATA05,-(R5)
     047366  012745  064606                                                          MOV     #DS.EXP,-(R5)
     047372  004767  007656                                                          JSR     PC,PACK
```

SUBP

```
      047376  012605                                                                    MOV     (SP)+,R5
 8550 047400
      047400  004767  007034                  CALL CLRREG                ;ENSURE GPR'S 0-5 ARE ZERO
                                                                                       JSR     PC,CLRREG
 8551 047404                                  INLINE <SUBPI>
      047404  076171                                                                   SUBPI
 8552 047406                                  INLINE <.WORD TADDPI>
      047406  066136                                                                   .WORD TADDPI
 8553 047410                                  INLINE <.WORD TADDPI+4>
      047410  066142                                                                   .WORD TADDPI+4
 8554 047412                                  INLINE <.WORD TADDPI+10>
      047412  066146                                                                   .WORD TADDPI+10
 8555 047414                                  LET CC := PSW
      047414  016767  130356  131076                                                   MOV     PSW,CC
 8556 047422                                  LET CC := CC CLR.BY #177760
      047422  042767  177760  131070                                                   BIC     #177760,CC
 8557 047430                                  IF CC NE #0 THEN
      047430  005767  131064                                                          TST     CC
      047434  001402                                                                 BEQ     $52157
 8558 047436                                      CALL ERROR
      047436  004767  010236                                                         JSR     PC,ERROR
 8559 047442                                  ENDIF
      047442                                                                  $52157:
 8560
 8561                                 ;+
 8562                                 ;       ENSURE GPR'S STILL ZERO
 8563                                 ;-
 8564
 8565 047442                                  IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN
      047442  005700                                                                 TST     R0
      047444  001004                                                                 BNE     $52160
      047446  005701                                                                 TST     R1
      047450  001002                                                                 BNE     $52160
      047452  005702                                                                 TST     R2
      047454  001402                                                                 BEQ     $52161
      047456                                                                  $52160:
 8566 047456                                      CALL ERROR
      047456  004767  010216                                                         JSR     PC,ERROR
 8567 047462                                  ENDIF
      047462                                                                  $52161:
 8568 047462                                  IF R3 NE #0 OR R4 NE #0 OR  R5 NE #0 THEN
      047462  005703                                                                 TST     R3
      047464  001004                                                                 BNE     $52162
      047466  005704                                                                 TST     R4
      047470  001002                                                                 BNE     $52162
      047472  005705                                                                 TST     R5
      047474  001402                                                                 BEQ     $52163
      047476                                                                  $52162:
 8569 047476                                      CALL ERROR
      047476  004767  010176                                                         JSR     PC,ERROR
 8570 047502                                  ENDIF
      047502                                                                  $52163:
 8571
 8572                                 ;+
 8573                                 ;       CHECK RESULTS
 8574                                 ;-
 8575
 8576 047502                                  LET R0 := #DS.DST
```

```
                047502  012700  064544                                                          MOV     #DS.DST,R0
8577            047506                  LET R1 := #DS.EXP               ;EXPECTED DATA
                047506  012701  064606                                                          MOV     #DS.EXP,R1
8578            047512                  LET R2 := TADDPI+10 CLR.BY #70000       ;GET STRING LENGTH
                047512  016702  016430                                                          MOV     TADDPI+10,R2
                047516  042702  070000                                                          BIC     #70000,R2
8579            047522                  INLINE          <ASR    R2>             ;LENGTH/2 (PACKING FACTOR)
                047522  006202                                                                  ASR     R2
8580            047524                  LET R2 := R2 + #DS.EXP                  ;R2 POINTS TO MSD/SIGN
                047524  062702  064606                                                          ADD     #DS.EXP,R2
8581            047530                  LET (R2) :B= (R2) SET.BY #17 ;MAKE IT UNSIGNED
                047530  152712  000017                                                          BISB    #17,(R2)
8582            047534                  LET ERRFLG := #0
                047534  005067  130716                                                          CLR     ERRFLG
8583            047540                  WHILE R0 NE #DS.DST+37 AND ERRFLG EQ #0 DO
                047540                                                                  $52164:
                047540  020027  064603                                                          CMP     R0,#DS.DST+37
                047544  001410                                                                  BEQ     $52165
                047546  005767  130704                                                          TST     ERRFLG
                047552  001005                                                                  BNE     $52165
8584            047554                      IFB (R0)+ NE (R1)+ THEN
                047554  122021                                                                  CMPB    (R0)+,(R1)+
                047556  001402                                                                  BEQ     $52166
8585            047560                          CALL ERROR
                047560  004767  010114                                                          JSR     PC,ERROR
8586            047564                      ENDIF
                047564                                                                  $52166:
8587            047564                  ENDDO
                047564  000765                                                                  BR      $52164
                047566                                                                  $52165:
8588
8589
```

```
                                .SBTTL  CMPP
                                ;*****************************************************************
                                ;
                                ;  CMPP TEST
                                ;
                                ;*****************************************************************

                                ROUTINE XCMPP
047566                                                                          XCMPP:
                                   INLINE <SCOPE>
047566  000004                                                                    SCOPE
                                   INLINE <NOP>
047570  000240                                                                    NOP
                                   LET $TIMES := #20                ;ITERATION COUNT
047572  012767  000020  130604                                                    MOV     #20,$TIMES
                                   LET COUNT := TCMP                ;GET TEST COUNT
047600  016767  015042  130662                                                    MOV     TCMP,COUNT
                                   LET TSTCAS := #TCMP+2                ;POINT TO FIRST TEST CASE
047606  012767  064650  130730                                                    MOV     #TCMP+2,TSTCAS
                                ;REPEAT
                                   INLINE <$5072:>
047614                                                                          $5072:
                                   CALL CLRDSB
047614  004767  007054                                                          JSR     PC,CLRDSB
                                   LET CC.EXP := #0
047620  005067  130676                                                          CLR     CC.EXP
                                   LET SIGNO := #0
047624  005067  130620                                                          CLR     SIGNO
                                   LET DATTYP := #0
047630  005067  130630                                                          CLR     DATTYP

                                ;+
                                ; GET SOURCE DESCRIPTORS FROM TABLE
                                ;-

                                   LET R0 := TSTCAS
047634  016700  130704                                                          MOV     TSTCAS,R0
                                   LET DATA00 := (R0)+              ;SRC1 LENGTH
047640  012067  130716                                                          MOV     (R0)+,DATA00
                                   LET DATA01 := (R0)+              ;SRC1 POINTER
047644  012067  130714                                                          MOV     (R0)+,DATA01
                                   LET DATA02 := (R0)+              ;SRC2 LENGTH
047650  012067  130712                                                          MOV     (R0)+,DATA02
                                   LET DATA03 := (R0)               ;SRC2 POINTER
047654  011067  130710                                                          MOV     (R0),DATA03

                                ;+
                                ; LOAD SRC1 AND SRC2 INTO DSBUF1 AND DSBUF2
                                ; IN NON-PACKED FORMAT, FOR NOW.
                                ;-

                                   LET R2 := DATA01 + #2            ;GET SRC1 ADDRESS
047660  016702  130700                                                          MOV     DATA01,R2
047664  062702  000002                                                          ADD     #2,R2
                                   LET R3 := #DSBUF1
047670  012703  064440                                                          MOV     #DSBUF1,R3
                                   LET DSTMP0 :B= (R2)              ;SAVE SRC1 MSD
```

```
                047674  111267  130654                                                          MOVB    (R2),DSTMP0
 8629           047700                          DECR R1 FROM DATA00 TO #0 BY #1
                047700  016701  130656                                                          MOV     DATA00,R1
                047704  000401                                                                  BR      $52171
                047706                                                                  $52172:
                047706  005301                                                                  DEC     R1
                047710                                                                  $52171:
                047710  005701                                                                  TST     R1
                047712  002402                                                                  BLT     $52173
 8630           047714                              LET (R3)+ :B= (R2)+
                047714  112223                                                                  MOVB    (R2)+,(R3)+
 8631           047716                          ENDDEC
                047716  000773                                                                  BR      $52172
                047720                                                                  $52173:
 8632           047720                          LET DSTMP1 :B= -(R2)            ;SAVE SRC1 LSD
                047720  114267  130631                                                          MOVB    -(R2),DSTMP1
 8633           047724                          LET R2 := DATA03 + #2           ;GET SRC2 ADDRESS
                047724  016702. 130640                                                          MOV     DATA03,R2
                047730  062702  000002                                                          ADD     #2,R2
 8634           047734                          LET R3 := #DSBUF2
                047734  012703  064502                                                          MOV     #DSBUF2,R3
 8635           047740                          LET DSTMP2 :B= (R2)             ;SAVE SRC2 MSD
                047740  111267  130612                                                          MOVB    (R2),DSTMP2
 8636           047744                          DECR R1 FROM DATA02 TO #0 BY #1
                047744  016701  130616                                                          MOV     DATA02,R1
                047750  000401                                                                  BR      $52174
                047752                                                                  $52175:
                047752  005301                                                                  DEC     R1
                047754                                                                  $52174:
                047754  005701                                                                  TST     R1
                047756  002402                                                                  BLT     $52176
 8637           047760                              LET (R3)+ :B= (R2)+
                047760  112223                                                                  MOVB    (R2)+,(R3)+
 8638           047762                          ENDDEC
                047762  000773                                                                  BR      $52175
                047764                                                                  $52176:
 8639           047764                          LET DSTMP3 :B= -(R2)            ;SAVE SRC2 LSD
                047764  114267  130567                                                          MOVB    -(R2),DSTMP3
 8640
 8641                                   ;+
 8642                                   ; COMPARE SRC1 AND SRC2, AND SET N AND Z BITS
 8643                                   ; IN CC.EXP ACCORDINGLY.
 8644                                   ;-
 8645
 8646           047770                          IF DATA00 EQ #0 AND DATA02 EQ #0 THEN       ;SRC1 LENGTH AND SRC2 LENGTH = 0
                047770  005767  130566                                                          TST     DATA00
                047774  001007                                                                  BNE     $52177
                047776  005767  130564                                                          TST     DATA02
                050002  001004                                                                  BNE     $52177
 8647           050004                              LET CC.EXP := CC.EXP SET.BY #ZBIT
                050004  052767  000004  130510                                                  BIS     #ZBIT,CC.EXP
 8648           050012                          ELSE
                050012  000424                                                                  BR      $52200
                050014                                                                  $52177:
 8649           050014                              IF DATA00 EQ #0 AND @DATA03 EQ #0 THEN  ;SRC1 LENGTH = 0 AND SRC2 SIGN IS +
                050014  005767  130542                                                          TST     DATA00
                050020  001007                                                                  BNE     $52201
```

CMPP

```
        050022  005777  130542                                                          TST     @DATA03
        050026  001004                                                                  BNE     $52201
   8650 050030                          LET CC.EXP := CC.EXP SET.BY #NBIT
        050030  052767  000010  130464                                                  BIS     #NBIT,CC.EXP
   8651 050036                       ELSE
        050036  000412                                                                  BR      $52202
        050040                                                                  $52201:
   8652 050040                          IF DATA02 EQ #0 AND @DATA01 EQ #-1 THEN     ;SRC2 LENGTH = 0 & SRC1 SIGN IS -
        050040  005767  130522                                                          TST     DATA02
        050044  001007                                                                  BNE     $52203
        050046  027727  130512  177777                                                  CMP     @DATA01,#-1
        050054  001003                                                                  BNE     $52203
   8653 050056                             LET CC.EXP := CC.EXP SET.BY #NBIT
        050056  052767  000010  130436                                                  BIS     #NBIT,CC.EXP
   8654 050064                          ENDIF
        050064                                                                  $52203:
   8655 050064                       ENDIF
        050064                                                                  $52202:
   8656 050064                    ENDIF
        050064                                                                  $52200:
   8657 050064                    IF DATA00 NE #0 AND DATA02 NE #0 THEN       ;SRC1 LENGTH AND SRC2 LENGTH > 0
        050064  005767  130472                                                          TST     DATA00
        050070  001520                                                                  BEQ     $52204
        050072  005767  130470                                                          TST     DATA02
        050076  001515                                                                  BEQ     $52204
   8658 050100                          IF @DATA01 NE @DATA03 THEN              ;SRC1 SIGN AND SRC2 SIGN ARE DIFFERENT
        050100  027777  130460  130462                                                  CMP     @DATA01,@DATA03
        050106  001410                                                                  BEQ     $52205
   8659 050110                             IF @DATA01 EQ #-1 THEN                  ;SRC1 SIGN IS -
        050110  027727  130450  177777                                                  CMP     @DATA01,#-1
        050116  001003                                                                  BNE     $52206
   8660 050120                                LET CC.EXP := CC.EXP SET.BY #NBIT
        050120  052767  000010  130374                                                  BIS     #NBIT,CC.EXP
   8661 050126                             ENDIF
        050126                                                                  $52206:
   8662 050126                       ELSE                                      ;SRC1 SIGN AND SRC2 SIGN ARE EQUAL
        050126  000501                                                                  BR      $52207
        050130                                                                  $52205:
   8663 050130                          LET SIGNO := @DATA01                    ;SAVE THE COMMON SIGN
        050130  017767  130430  130312                                                  MOV     @DATA01,SIGNO
   8664 050136                          IF DATA00 LO DATA02 AND SIGNO EQ #0 THEN    ;SRC1 LENGTH < SRC2 LENGTH AND SIGNS
ARE *
        050136  026767  130420  130422                                                  CMP     DATA00,DATA02
        050144  103007                                                                  BHIS    $52210
        050146  005767  130276                                                          TST     SIGNO
        050152  001004                                                                  BNE     $52210
   8665 050154                             LET CC.EXP := CC.EXP SET.BY #NBIT
        050154  052767  000010  130340                                                  BIS     #NBIT,CC.EXP
   8666 050162                          ELSE
        050162  000413                                                                  BR      $52211
        050164                                                                  $52210:
   8667 050164                             IF DATA00 HI DATA02 AND SIGNO EQ #-1 THEN    ;SRC1 LENGTH > SRC2 LENGTH AND S
IGNS ARE -
        050164  026767  130372  130374                                                  CMP     DATA00,DATA02
        050172  101407                                                                  BLOS    $52212
        050174  026727  130250  177777                                                  CMP     SIGNO,#-1
        050202  001003                                                                  BNE     $52212
   8668 050204                                LET CC.EXP := CC.EXP SET.BY #NBIT
        050204  052767  000010  130310                                                  BIS     #NBIT,CC.EXP
```

```
8669 050212                                         ENDIF
     050212                                                                  $52212:
8670 050212                                     ENDIF
     050212                                                                  $52211:
8671 050212                                     IF DATA00 EQ DATA02 THEN            ;SRC1 LENGTH AND SRC2 LENGTH ARE EQUAL
     050212  026767  130344  130346                                                      CMP     DATA00,DATA02
     050220  001044                                                                     BNE     $52213
8672
8673                                    ;+
8674                                    ; COMPARE SRC1 AND SRC2 BYTE BY BYTE
8675                                    ;-
8676
8677 050222                                         LET R0 := DATA00                ;SRC BUFFER LENGTH
     050222  016700  130334                                                             MOV     DATA00,R0
8678 050226                                         LET R1 := #DSBUF1                       ;SRC1 BUFFER
     050226  012701  064440                                                             MOV     #DSBUF1,R1
8679 050232                                         LET R2 := #DSBUF2                       ;SRC2 BUFFER
     050232  012702  064502                                                             MOV     #DSBUF2,R2
8680 050236                                         WHILEB R0 NE #0 AND (R1)+ EQ (R2)+ DO
     050236                                                                  $52214:
     050236  105700                                                                     TSTB    R0
     050240  001404                                                                     BEQ     $52215
     050242  022122                                                                     CMP     (R1)+,(R2)+
     050244  001002                                                                     BNE     $52215
8681 050246                                             LET R0 := R0 - #1           ;DECREMENT COUNTER
     050246  005300                                                                     DEC     R0
8682 050250                                         ENDDO
     050250  000772                                                                     BR      $52214
     050252                                                                 $52215:
8683 050252                                         IFB -(R1) NE -(R2) THEN  ;SOURCES WERE NOT =
     050252  124142                                                                     CMPB    -(R1),-(R2)
     050254  001423                                                                     BEQ     $52216
8684 050256                                             IFB (R1) LO (R2) AND SIGNO EQ #0 THEN
     050256  121112                                                                     CMPB    (R1),(R2)
     050260  103007                                                                     BHIS    $52217
     050262  005767  130162                                                             TST     SIGNO
     050266  001004                                                                     BNE     $52217
8685 050270                                                 LET CC.EXP := CC.EXP SET.BY #NBIT
     050270  052767  000010  130224                                                     BIS     #NBIT,CC.EXP
8686 050276                                             ELSE
     050276  000411                                                                     BR      $52220
     050300                                                                 $52217:
8687 050300                                                 IFB (R1) HI (R2) AND SIGNO EQ #-1 THEN
     050300  121112                                                                     CMPB    (R1),(R2)
     050302  101407                                                                     BLOS    $52221
     050304  026727  130140  177777                                                     CMP     SIGNO,#-1
     050312  001003                                                                     BNE     $52221
8688 050314                                                     LET CC.EXP := CC.EXP SET.BY #NBIT
     050314  052767  000010  130200                                                     BIS     #NBIT,CC.EXP
8689 050322                                                 ENDIF
     050322                                                                 $52221:
8690 050322                                             ENDIF
     050322                                                                 $52220:
8691 050322                                     ELSE                        ;SOURCES WERE EQUAL
     050322  000403                                                                     BR      $52222
     050324                                                                 $52216:
8692 050324                                         LET CC.EXP := CC.EXP SET.BY #ZBIT
```

```
      050324  052767  000004  130170                                                    BIS     #ZBIT,CC.EXP
 8693 050332                                ENDIF
      050332                                                                  $52222:
 8694 050332                             ENDIF
      050332                                                                  $52213:
 8695 050332                          ENDIF
      050332                                                                  $52207:
 8696 050332                       ENDIF
      050332                                                                  $52204:
 8697
 8698                             ;+
 8699                             ; PERFORM THE CMPP INSTRUCTION WITH THE SIGNED
 8700                             ; PACKED DATA TYPE. (UNSIGNED PACKED WILL BE
 8701                             ; TESTED IN THE INLINE PORTION OF THE TEST).
 8702                             ; CHECK THE RESULTS AND REPORT ANY ERRORS.
 8703                             ;-
 8704
 8705 050332                       CALL CLRDSB
      050332  004767  006336                                                    JSR     PC,CLRDSB
 8706                             ;LOAD DSBUF1 WITH SRC1 DECIMAL STRING AND SIGN, IN PACKED FORMAT.
 8707 050336                       LET R0 := #DSBUF1
      050336  012700  064440                                                    MOV     #DSBUF1,R0
 8708 050342                       LET R1 := DATA01 + #2
      050342  016701  130216                                                    MOV     DATA01,R1
      050346  062701  000002                                                    ADD     #2,R1
 8709 050352                       LET R2 := DATA00              ;DECIMAL STRING LENGTH
      050352  016702  130204                                                    MOV     DATA00,R2
 8710 050356                       IF R2 EQ #0 THEN              ;LENGTH = 0
      050356  005702                                                            TST     R2
      050360  001011                                                            BNE     $52223
 8711 050362                          IF @DATA01 EQ #0 THEN      ;SIGN IS +
      050362  005777  130176                                                    TST     @DATA01
      050366  001003                                                            BNE     $52224
 8712 050370                             LET (R0) :B= #14
      050370  112710  000014                                                    MOVB    #14,(R0)
 8713 050374                          ELSE                                    ;SIGN IS -
      050374  000402                                                            BR      $52225
      050376                                                          $52224:
 8714 050376                             LET (R0) :B= #15
      050376  112710  000015                                                    MOVB    #15,(R0)
 8715 050402                          ENDIF
      050402                                                          $52225:
 8716 050402                       ELSE
      050402  000434                                                            BR      $52226
      050404                                                          $52223:
 8717 050404                          IF #BIT00 NOTSETIN R2 THEN              ;LENGTH IS EVEN AND > 0
      050404  032702  000001                                                    BIT     #BIT00,R2
      050410  001002                                                            BNE     $52227
 8718 050412                             LET (R0)+ :B= (R1)+     ;MOVE THE EVEN NIBBLE
      050412  112120                                                            MOVB    (R1)+,(R0)+
 8719 050414                             LET R2 := R2 - #1               ;DECREMENT COUNTER
      050414  005302                                                            DEC     R2
 8720 050416                          ENDIF
      050416                                                          $52227:
 8721 050416                          WHILE R2 NE #1 DO
      050416                                                          $52230:
      050416  020227  000001                                                    CMP     R2,#1
```

```
     050422  001410                                                                     BEQ     $52231
     050424                          LET R3 :B= (R1)+              ;GET A BCD
     050424  112103                                                                     MOVB    (R1)+,R3
     050426                          INLINE   <ASH  #4,R3>         ;SHIFT IT TO THE HIGH NIBBLE
     050426  072327  000004                                                             ASH  #4,R3
     050432                          LET R3 :B= R3 SET.BY (R1)+       ;PUT NEXT BCD INTO LOWER NIBBLE
     050432  152103                                                                     BISB    (R1)+,R3
     050434                          LET (R0)+ :B= R3
     050434  110320                                                                     MOVB    R3,(R0)+
     050436                          LET R2 := R2 - #2             ;DECREMENT  COUNTER
     050436  162702  000002                                                             SUB     #2,R2
     050442                      ENDDO
     050442  000765                                                                     BR      $52230
     050444                                                                    $52231:
     050444                      LET R3 :B= (R1)           ;GET LAST BCD
     050444  111103                                                                     MOVB    (R1),R3
     050446                      INLINE   <ASH  #4,R3>
     050446  072327  000004                                                             ASH  #4,R3
     050452                      IF @DATA01 EQ #0 THEN     ;SIGN IS +
     050452  005777  130106                                                             TST     @DATA01
     050456  001003                                                                     BNE     $52232
     050460                          LET R3 :B= R3 SET.BY #14
     050460  152703  000014                                                             BISB    #14,R3
     050464                      ELSE                      ;SIGN IS -
     050464  000402                                                                     BR      $52233
     050466                                                                    $52232:
     050466                          LET R3 :B= R3 SET.BY #15
     050466  152703  000015                                                             BISB    #15,R3
     050472                      ENDIF
     050472                                                                    $52233:
     050472                      LET (R0) :B= R3           ;PUT MSD AND SIGN INTO THE BUFFER
     050472  110310                                                                     MOVB    R3,(R0)
     050474                  ENDIF
     050474                                                                    $52226:
                         ;LOAD DSBUF2 WITH SRC2 DECIMAL STRING AND SIGN, IN PACKED FORMAT.
     050474                  LET R0 := #DSBUF2
     050474  012700  064502                                                             MOV     #DSBUF2,R0
     050500                  LET R1 := DATA03 + #2
     050500  016701  130064                                                             MOV     DATA03,R1
     050504  062701  000002                                                             ADD     #2,R1
     050510                  LET R2 := DATA02              ;DECIMAL STRING LENGTH
     050510  016702  130052                                                             MOV     DATA02,R2
     050514                  IF R2 EQ #0 THEN              ;LENGTH = 0
     050514  005702                                                                     TST     R2
     050516  001011                                                                     BNE     $52234
     050520                      IF @DATA03 EQ #0 THEN     ;SIGN IS +
     050520  005777  130044                                                             TST     @DATA03
     050524  001003                                                                     BNE     $52235
     050526                          LET (R0) :B= #14
     050526  112710  000014                                                             MOVB    #14,(R0)
     050532                      ELSE                                ;SIGN IS -
     050532  000402                                                                     BR      $52236
     050534                                                                    $52235:
     050534                          LET (R0) :B= #15
     050534  112710  000015                                                             MOVB    #15,(R0)
     050540                      ENDIF
     050540                                                                    $52236:
```

```
                                        ELSE
050540  000434                                                                  BR      $52237
050542                                                                  $52234:
                                            IF #BIT00 NOTSETIN R2 THEN      ;LENGTH IS EVEN AND > 0
050542  032702  000001                                                          BIT     #BIT00,R2
050546  001002                                                                  BNE     $52240
                                                LET (R0)+ :B= (R1)+  ;MOVE THE EVEN NIBBLE
050550  112120                                                                  MOVB    (R1)+,(R0)+
                                                LET R2 := R2 - #1           ;DECREMENT COUNTER
050552  005302                                                                  DEC     R2
                                            ENDIF
050554                                                                  $52240:
                                            WHILE R2 NE #1 DO
050554                                                                  $52241:
050554  020227  000001                                                          CMP     R2,#1
050560  001410                                                                  BEQ     $52242
                                                LET R3 :B= (R1)+            ;GET A BCD
050562  112103                                                                  MOVB    (R1)+,R3
                                                INLINE   <ASH  #4,R3>       ;SHIFT IT TO THE HIGH NIBBLE
050564  072327  000004                                                          ASH  #4,R3
                                                LET R3 :B= R3 SET.BY (R1)+      ;PUT NEXT BCD INTO LOWER NIBBLE
050570  152103                                                                  BISB    (R1)+,R3
                                                LET (R0)+ :B= R3
050572  110320                                                                  MOVB    R3,(R0)+
                                                LET R2 := R2 - #2           ;DECREMENT COUNTER
050574  162702  000002                                                          SUB     #2,R2
                                            ENDDO
050600  000765                                                                  BR      $52241
050602                                                                  $52242:
                                            LET R3 :B= (R1)
050602  111103                                                                  MOVB    (R1),R3
                                            INLINE   <ASH  #4,R3>
050604  072327  000004                                                          ASH  #4,R3
                                            IF @DATA03 EQ #0 THEN       ;SIGN IS +
050610  005777  127754                                                          TST     @DATA03
050614  001003                                                                  BNE     $52243
                                                LET R3 :B= R3 SET.BY #14
050616  152703  000014                                                          BISB    #14,R3
                                            ELSE                        ;SIGN IS -
050622  000402                                                                  BR      $52244
050624                                                                  $52243:
                                                LET R3 :B= R3 SET.BY #15
050624  152703  000015                                                          BISB    #15,R3
                                            ENDIF
050630                                                                  $52244:
                                            LET (R0) :B= R3             ;PUT MSD AND SIGN INTO THE BUFFER
050630  110310                                                                  MOVB    R3,(R0)
                                        ENDIF
050632                                                                  $52237:

                                    ;+
                                    ; LOAD THE GPR'S AND DO THE CMPP INSTRUCTION
                                    ;-

                                        LET R0 := #60000 SET.BY DATA00      ;SRC1.DSCR
050632  012700  060000                                                          MOV     #60000,R0
050636  056700  127720                                                          BIS     DATA00,R0
```

CMPP

```
                                        LET R1 := #DSBUF1               ;  "
050642  012701  064440                                                          MOV     #DSBUF1,R1
                                        LET R2 := #60000 SET.BY DATA02          ;SRC2.DSCR
050646  012702  060000                                                          MOV     #60000,R2
050652  056702  127710                                                          BIS     DATA02,R2
                                        LET R3 := #DSBUF2               ;  "
050656  012703  064502                                                          MOV     #DSBUF2,R3
                                        LET R4 := #0
050662  005004                                                                  CLR     R4
                                        LET R5 := #0
050664  005005                                                                  CLR     R5
                                        INLINE <CMPP>
050666  076072                                                                  CMPP

                                    ;+
                                    ; CHECK THE RESULTS AND REPORT ANY ERRORS
                                    ;-

                                        LET CC := PSW
050670  016767  127102  127622                                                  MOV     PSW,CC
                                        LET CC := CC CLR.BY #177760
050676  042767  177760  127614                                                  BIC     #177760,CC
                                        IF CC NE CC.EXP THEN
050704  026767  127610  127610                                                  CMP     CC,CC.EXP
050712  001402                                                                  BEQ     $52245
                                            CALL ERROR
050714  004767  006760                                                          JSR     PC,ERROR
                                        ENDIF
050720                                                                  $52245:
                                        IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN
050720  005700                                                                  TST     R0
050722  001004                                                                  BNE     $52246
050724  005701                                                                  TST     R1
050726  001002                                                                  BNE     $52246
050730  005702                                                                  TST     R2
050732  001402                                                                  BEQ     $52247
050734                                                                  $52246:
                                            CALL ERROR
050734  004767  006740                                                          JSR     PC,ERROR
                                        ENDIF
050740                                                                  $52247:
                                        IF R3 NE #0 OR R4 NE #0 OR R5 NE #0 THEN
050740  005703                                                                  TST     R3
050742  001004                                                                  BNE     $52250
050744  005704                                                                  TST     R4
050746  001002                                                                  BNE     $52250
050750  005705                                                                  TST     R5
050752  001402                                                                  BEQ     $52251
050754                                                                  $52250:
                                            CALL ERROR
050754  004767  006720                                                          JSR     PC,ERROR
                                        ENDIF
050760                                                                  $52251:
                                        LET TSTCAS := TSTCAS + #10
050760  062767  000010  127556                                                  ADD     #10,TSTCAS
                                        LET COUNT := COUNT - #1
050766  005367  127476                                                          DEC     COUNT
```

```
 8798                                ;UNTIL COUNT EQ #0
 8799 050772                           IF COUNT NE #0 THEN
      050772  005767  127472                                                    TST     COUNT
      050776  001402                                                            BEQ     $52252
 8800 051000                               INLINE <JMP $5072>
      051000  000167  176610                                                    JMP $5072
 8801 051004                           ENDIF
      051004                                                          $52252:
 8802
 8803                                ;+
 8804                                ; DO ONE INLINE CASE - CMPPI.  USE UNSIGNED PACKED DATA TYPE.
 8805                                ;-
 8806
 8807 051004                           LET R0 := #-1
      051004  012700  177777                                                    MOV     #-1,R0
 8808 051010                           LET R1 := #-1
      051010  012701  177777                                                    MOV     #-1,R1
 8809 051014                           LET R2 := #-1
      051014  012702  177777                                                    MOV     #-1,R2
 8810 051020                           LET R3 := #-1
      051020  012703  177777                                                    MOV     #-1,R3
 8811 051024                           LET R4 := #-1
      051024  012704  177777                                                    MOV     #-1,R4
 8812 051030                           LET R5 := #-1
      051030  012705  177777                                                    MOV     #-1,R5
 8813 051034                           INLINE   <CMPPI>
      051034  076172                                                            CMPPI
 8814 051036                           INLINE   <.WORD  TCMPPI>
      051036  051152                                                            .WORD   TCMPPI
 8815 051040                           INLINE   <.WORD  TCMPPI+4>
      051040  051156                                                            .WORD   TCMPPI+4
 8816 051042                           LET CC := PSW
      051042  016767  126730  127450                                            MOV     PSW,CC
 8817 051050                           LET CC := CC CLR.BY #177760
      051050  042767  177760  127442                                            BIC     #177760,CC
 8818 051056                           IF CC NE #4 THEN
      051056  026727  127436  000004                                            CMP     CC,#4
      051064  001402                                                            BEQ     $52253
 8819 051066                               CALL ERROR
      051066  004767  006606                                                    JSR     PC,ERROR
 8820 051072                           ENDIF
      051072                                                          $52253:
 8821 051072                           IF R0 NE #-1 OR R1 NE #-1 OR R2 NE #-1 THEN
      051072  020027  177777                                                    CMP     R0,#-1
      051076  001006                                                            BNE     $52254
      051100  020127  177777                                                    CMP     R1,#-1
      051104  001003                                                            BNE     $52254
      051106  020227  177777                                                    CMP     R2,#-1
      051112  001402                                                            BEQ     $52255
      051114                                                          $52254:
 8822 051114                               CALL ERROR
      051114  004767  006560                                                    JSR     PC,ERROR
 8823 051120                           ENDIF
      051120                                                          $52255:
 8824 051120                           IF R3 NE #-1 OR R4 NE #-1 OR R5 NE #-1 THEN
      051120  020327  177777                                                    CMP     R3,#-1
      051124  001006                                                            BNE     $52256
```

CMPP

```
        051126  020427  177777                                                          CMP     R4,#-1
        051132  001003                                                                  BNE     $52256
        051134  020527  177777                                                          CMP     R5,#-1
        051140  001402                                                                  BEQ     $52257
        051142                                                                  $52256:
  8825  051142                          CALL ERROR
        051142  004767  006532                                                          JSR     PC,ERROR
  8826  051146                        ENDIF
        051146                                                                  $52257:
  8827  051146                        INLINE  <JMP XASHP>
        051146  000167  000014                                                          JMP XASHP
  8828
  8829                              ;+
  8830                              ;INLINE TEST CASE
  8831                              ;-
  8832
  8833  051152  070005              TCMPPI: .WORD   070005          :SRC1 DATA TYPE = UNSIGNED PACKED, LENGTH = 5
  8834  051154  051162                      .WORD   CMPPI1          ;SRC1.DSCR
  8835  051156  070005                      .WORD   070005          ;SRC2 DATA TYPE = UNSIGNED PACKED, LENGTH = 5
  8836  051160  051162                      .WORD   CMPPI1          ;SRC2.DSCR
  8837
  8838  051162     022     064     137  CMPPI1: .BYTE   022,064,137     ;PACKED DECIMAL STRING = 12345
  8839                                      .EVEN
  8840
```

```
                                .SBTTL  ASHP
                                ;*****************************************************************
                                ;
                                ;  ASHP TEST
                                ;
                                ;*****************************************************************

051166                          ROUTINE XASHP                                           XASHP:
051166
051166                             INLINE       <SCOPE>                                 SCOPE
051166  000004
051170                             INLINE       <NOP>                                   NOP
051170  000240
051172                             LET $TIMES := #20                  ;ITERATION COUNT     MOV     #20,$TIMES
051172  012767  000020  127204
051200                             LET COUNT := TASH                  ;GET TEST COUNT      MOV     TASH,COUNT
051200  016767  013710  127262
051206                             LET TSTCAS := #TASH+2                   ;POINT TO FIRST TEST CASE
051206  012767  065116  127330                                              MOV     #TASH+2,TSTCAS
051214                             LET SIGNO := #0                         ;INITIALIZE THE SIGN WORD
051214  005067  127230                                                      CLR     SIGNO
                                ;REPEAT
051220                             INLINE       <$5076:>                                $5076:
051220
051220                             CALL CLRDSB                        ;CLEAR DECIMAL STRING BUFFERS
051220  004767  005450                                                      JSR     PC,CLRDSB
051224                             LET CC.EXP := #0                                        CLR     CC.EXP
051224  005067  127272

                                ;+
                                ; GET TEST DATA FROM TABLE
                                ;-

051230                             LET R0 := TSTCAS                                        MOV     TSTCAS,R0
051230  016700  127310
051234                             LET DATA00 := (R0)+                ;SRC LENGTH          MOV     (R0)+,DATA00
051234  012067  127322
051240                             LET DATA01 := (R0)+                ;DST LENGTH          MOV     (R0)+,DATA01
051240  012067  127320
051244                             LET DATA02 := (R0)                 ;SHIFT DSCR.         MOV     (R0),DATA02
051244  011067  127316

                                ;+
                                ;+
                                ; PUT THE TEST STRING INTO DSBUF1 IN SIGNED PACKED FORMAT.  INCLUDE
                                ; THE SIGN.
                                ;-

051250                             LET R0 := #DSBUF1                                       MOV     #DSBUF1,R0
051250  012700  064440
051254                             LET R1 := DATA00                                        MOV     DATA00,R1
051254  016701  127302
051260                             IF R1 EQ #0 THEN                        ;SRC.LEN = 0    TST     R1
051260  005701                                                                             BNE     $52262
051262  001011
051264                                 IF SIGNO EQ #0 THEN                         ;SIGN IS +  TST     SIGNO
051264  005767  127160
```

```
ASHP

        051270  001003                                                          BNE     $52263
   8880 051272                              LET (R0) :B= #14
        051272  112710  000014                                                MOVB    #14,(R0)
   8881 051276                            ELSE                          ;SIGN IS -
        051276  000402                                                      BR      $52264
        051300                                                            $52263:
   8882 051300                              LET (R0) :B= #15
        051300  112710  000015                                              MOVB    #15,(R0)
   8883 051304                            ENDIF
        051304                                                            $52264:
   8884 051304                          ELSE
        051304  000426                                                      BR      $52265
        051306                                                            $52262:
   8885 051306                              IF #BIT00 NOTSETIN R1 THEN       ;SRC.LEN IS EVEN AND > 0
        051306  032701  000001                                                  BIT     #BIT00,R1
        051312  001003                                                          BNE     $52266
   8886 051314
        051314  112720  000004                 LET (R0)+ :B= #4                MOVB    #4,(R0)+
   8887 051320                                  LET R1 := R1 - #1          ;DECREMENT COUNT
        051320  005301                                                          DEC     R1
   8888 051322                              ENDIF
        051322                                                            $52266:
   8889 051322                              WHILE R1 NE #1 DO
        051322                                                            $52267:
        051322  020127  000001                                                  CMP     R1,#1
        051326  001405                                                          BEQ     $52270
   8890 051330                                  LET (R0)+ :B= #104      ;MOVE TWO NIBBLES OF BCD
        051330  112720  000104                                                  MOVB    #104,(R0)+
   8891 051334                                  LET R1 := R1 - #2          ;DECREMENT COUNT
        051334  162701  000002                                                  SUB     #2,R1
   8892 051340                              ENDDO
        051340  000770                                                      BR      $52267
        051342                                                            $52270:
   8893 051342                              IF SIGNO EQ #0 THEN          ;SIGN IS +
        051342  005767  127102                                                  TST     SIGNO
        051346  001003                                                          BNE     $52271
   8894 051350                                  LET (R0) :B= #114
        051350  112710  000114                                                  MOVB    #114,(R0)
   8895 051354                                ELSE                       ;SIGN IS -
        051354  000402                                                      BR      $52272
        051356                                                            $52271:
   8896 051356                                  LET (R0) :B= #115
        051356  112710  000115                                                  MOVB    #115,(R0)
   8897 051362                              ENDIF
        051362                                                            $52272:
   8898 051362                          ENDIF                            $52265:
        051362
   8899
   8900                             ;+
   8901                             ; LOAD DS.EXP AND CC.EXP WITH EXPECTED RESULTS.
   8902                             ;-
   8903
   8904                             ;GET SHIFT COUNT MAGNITUDE AND PUT IT INTO TEMPO.
   8905
   8906 051362                        LET TEMPO := DATA02
        051362  016767  127200  127106                                        MOV     DATA02,TEMPO
   8907 051370                        IF #BIT07 SETIN TEMPO THEN       ;NEGATIVE SHIFT COUNT
```

ASHP

```
                                                                                        BIT     #BIT07,TEMP0
        051370  032767  000200  127100                                                BEQ     $52273
        051376  001402
 8908 051400                                LET TEMP0 := NEGATE TEMP0
        051400  005467  127072                                                        NEG     TEMP0
 8909 051404                              ENDIF
        051404                                                                  $52273:
 8910 051404                              LET TEMP0 := TEMP0 CLR.BY #177600
        051404  042767  177600  127064                                                BIC     #177600,TEMP0
 8911
 8912                                   ;TAKE CARE OF EXPECTED NBIT
 8913
 8914 051412                              IF SIGN0 EQ #-1 AND DATA00 NE #0 THEN
        051412  026727  127032  177777                                                CMP     SIGN0,#-1
        051420  001006                                                                BNE     $52274
        051422  005767  127134                                                        TST     DATA00
        051426  001403                                                                BEQ     $52274
 8915 051430                                LET CC.EXP := CC.EXP SET.BY #NBIT
        051430  052767  000010  127064                                                BIS     #NBIT,CC.EXP
 8916 051436                              ENDIF
        051436                                                                  $52274:
 8917
 8918                                   ;PUT THE SIGN NIBBLE INTO THE EXPECTED DESTINATION
 8919
 8920 051436                              LET R1 := DATA01                    ;DST.LEN
        051436  016701  127122                                                        MOV     DATA01,R1
 8921 051442                              INLINE   <ASR  R1>
        051442  006201                                                                ASR  R1
 8922 051444                              LET R1 := R1 + #DS.EXP
        051444  062701  064606                                                        ADD     #DS.EXP,R1
 8923 051450                              IF SIGN0 EQ #0 OR DATA00 EQ #0 THEN
        051450  005767  126774                                                        TST     SIGN0
        051454  001403                                                                BEQ     $52275
        051456  005767  127100                                                        TST     DATA00
        051462  001003                                                                BNE     $52276
        051464                                                                  $52275:
 8924 051464                                  LET (R1) :B= (R1) SET.BY #14     ;POSITIVE SIGN
        051464  152711  000014                                                        BISB    #14,(R1)
 8925 051470                              ELSE
        051470  000402                                                                BR      $52277
        051472                                                                  $52276:
 8926 051472                                  LET (R1) :B= (R1) SET.BY #15     ;NEGATIVE SIGN
        051472  152711  000015                                                        BISB    #15,(R1)
 8927 051476                              ENDIF
        051476                                                                  $52277:
 8928
 8929                                   ;TAKE CARE OF ZERO LENGTH SOURCE OR DESTINATION
 8930
 8931 051476                              IF DATA00 EQ #0 OR DATA01 EQ #0 THEN
        051476  005767  127060                                                        TST     DATA00
        051502  001403                                                                BEQ     $52300
        051504  005767  127054                                                        TST     DATA01
        051510  001003                                                                BNE     $52301
        051512                                                                  $52300:
 8932 051512                                  LET CC.EXP := CC.EXP SET.BY #ZBIT
        051512  052767  000004  127002                                                BIS     #ZBIT,CC.EXP
 8933 051520                              ENDIF
        051520                                                                  $52301:
```

```
ASHP

 8934 051520                            IF DATA00 NE #0 AND DATA01 NE #0 AND #BIT07 SETIN DATA02 THEN
      051520  005767  127036                                                                    TST     DATA00
      051524  001531                                                                            BEQ     $52302
      051526  005767  127032                                                                    TST     DATA01
      051532  001526                                                                            BEQ     $52302
      051534  032767  000200  127024                                                           BIT     #BIT07,DATA02
      051542  001522                                                                            BEQ     $52302
 8935
 8936                                ;TAKE CARE OF NEGATIVE SHIFT
 8937
 8938 051544                                LET R0 := DATA00 - TEMP0          ;SRC.LEN - SHIFT COUNT
      051544  016700  127012                                                                    MOV     DATA00,R0
      051550  166700  126722                                                                    SUB     TEMP0,R0
 8939 051554                                LET R1 := DATA01                  ;DST.LEN
      051554  016701  127004                                                                    MOV     DATA01,R1
 8940 051560                                INLINE   <ASR  R1>
      051560  006201                                                                            ASR  R1
 8941 051562                                LET R1 := R1 + #DS.EXP
      051562  062701  064606                                                                    ADD     #DS.EXP,R1
 8942 051566                                LET R2 := #1
      051566  012702  000001                                                                    MOV     #1,R2
 8943 051572                                LET R3 := DATA01
      051572  016703  126766                                                                    MOV     DATA01,R3
 8944 051576                                IF R0 EQ #0 THEN
      051576  005700                                                                            TST     R0
      051600  001024                                                                            BNE     $52303
 8945 051602                                    IF DATA02 HIS #3000 THEN
      051602  026727  126760  003000                                                           CMP     DATA02,#3000
      051610  103403                                                                           BLO     $52304
 8946 051612                                        LET (R1) :B= (R1) SET.BY #20
      051612  152711  000020                                                                ;ROUNDING
 8947 051616                                    ELSE                                          BISB    #20,(R1)
      051616  000415                                                                           BR      $52305
      051620                                                                            $52304:
 8948 051620                                        LET CC.EXP := CC.EXP SET.BY #ZBIT
      051620  052767  000004  126674                                                           BIS     #ZBIT,CC.EXP
 8949 051626                                        LET CC.EXP := CC.EXP CLR.BY #NBIT
      051626  042767  000010  126666                                                           BIC     #NBIT,CC.EXP
 8950 051634                                        LET R5 := DATA01
      051634  016705  126724                                                                   MOV     DATA01,R5
 8951 051640                                        INLINE   <ASR  R5>
      051640  006205                                                                           ASR  R5
 8952 051642                                        LET R5 := R5 + #DS.EXP
      051642  062705  064606                                                                   ADD     #DS.EXP,R5
 8953 051646                                        LET (R5) :B= (R5) CLR.BY #1
      051646  142715  000001                                                                   BICB    #1,(R5)
 8954 051652                                    ENDIF
      051652                                                                            $52305:
 8955 051652                            ENDIF
      051652                                                                            $52303:
 8956 051652                            IF R0 LT #0 THEN
      051652  005700                                                                           TST     R0
      051654  002015                                                                           BGE     $52306
 8957 051656                                LET CC.EXP := CC.EXP SET.BY #ZBIT
      051656  052767  000004  126636                                                           BIS     #ZBIT,CC.EXP
 8958 051664                                LET CC.EXP := CC.EXP CLR.BY #NBIT
      051664  042767  000010  126630                                                           BIC     #NBIT,CC.EXP
```

ASHP

```
8959 051672                             LET R5 := DATA01                                    MOV     DATA01,R5
     051672  016705  126666
8960 051676                             INLINE   <ASR  R5>                                  ASR  R5
     051676  006205
8961 051700                             LET R5 := R5 + #DS.EXP                              ADD     #DS.EXP,R5
     051700  062705  064606
8962 051704                             LET (R5) :B= (R5) CLR.BY #1                         BICB    #1,(R5)
     051704  142715  000001
8963 051710                         ENDIF
     051710                                                                         $52306:
8964 051710                         WHILE R2 LE R0 AND R3 HI #0 DO
     051710                                                                         $52307:
     051710  020200                                                                         CMP     R2,R0
     051712  003027                                                                         BGT     $52310
     051714  005703                                                                         TST     R3
     051716  101425                                                                         BLOS    $52310
8965 051720                             IF #BIT00 SETIN R2 THEN                             BIT     #BIT00,R2
     051720  032702  000001                                                               BEQ     $52311
     051724  001415
8966 051726                                 LET (R1) :B= (R1) SET.BY #100                   BISB    #100,(R1)
     051726  152711  000100
8967 051732                                 IF R2 EQ #1 AND DATA02 HIS #3000 THEN           CMP     R2,#1
     051732  020227  000001                                                               BNE     $52312
     051736  001006                                                                       CMP     DATA02,#3000
     051740  026727  126622  003000                                                       BLO     $52312
     051746  103402
8968 051750                                     LET (R1) :B= (R1) SET.BY #20 ;ROUNDING
     051750  152711  000020                                                           BISB    #20,(R1)
8969 051754                                 ENDIF
     051754                                                                         $52312:
8970 051754                                 LET R1 := R1 - #1
     051754  005301                                                                         DEC     R1
8971 051756                             ELSE                                                BR      $52313
     051756  000402                                                                 $52311:
     051760
8972 051760                                 LET (R1) :B= (R1) SET.BY #4                     BISB    #4,(R1)
     051760  152711  000004
8973 051764                             ENDIF
     051764                                                                         $52313:
8974 051764                             LET R2 := R2 + #1                                   INC     R2
     051764  005202
8975 051766                             LET R3 := R3 - #1                                   DEC     R3
     051766  005303
8976 051770                         ENDDO                                                   BR      $52307
     051770  000747                                                                 $52310:
     051772
8977 051772                         IF R3 EQ #0 AND R2 LE R0 THEN                           TST     R3
     051772  005703                                                                         BNE     $52314
     051774  001005                                                                       CMP     R2,R0
     051776  020200                                                                       BGT     $52314
     052000  003003
8978 052002                             LET CC.EXP := CC.EXP SET.BY #VBIT                   BIS     #VBIT,CC.EXP
     052002  052767  000002  126512
8979 052010                         ENDIF
     052010                                                                         $52314:
8980 052010                     ENDIF                                              $52302:
     052010
```

ASHP

```
8981 052010                          IF DATA00 NE #0 AND DATA01 NE #0 AND #BIT07 NOTSETIN DATA02 THEN
     052010  005767  126546                                                                  TST     DATA00
     052014  001506                                                                          BEQ     $52315
     052016  005767  126542                                                                  TST     DATA01
     052022  001503                                                                          BEQ     $52315
     052024  032767  000200  126534                                                         BIT     #BIT07,DATA02
     052032  001077                                                                          BNE     $52315
8982
8983                               ;TAKE CARE OF POSITIVE SHIFT
8984
8985 052034                             IF #BIT07 NOTSETIN DATA02 THEN
     052034  032767  000200  126524                                                         BIT     #BIT07,DATA02
     052042  001073                                                                          BNE     $52316
8986 052044                                 LET TEMP1 := DATA00 + TEMP0          ;SRC.LEN + SHIFT COUNT
     052044  016767  126512  126426                                                         MOV     DATA00,TEMP1
     052052  066767  126420  126420                                                         ADD     TEMP0,TEMP1
8987 052060                                 IF TEMP1 HI DATA01 THEN
     052060  026767  126414  126476                                                         CMP     TEMP1,DATA01
     052066  101403                                                                          BLOS    $52317
8988 052070                                     LET CC.EXP := CC.EXP SET.BY #VBIT
     052070  052767  000002  126424                                                         BIS     #VBIT,CC.EXP
8989 052076                                 ENDIF
     052076                                                                        $52317:
8990 052076                                 IF TEMP0 HIS DATA01 THEN
     052076  026767  126374  126460                                                         CMP     TEMP0,DATA01
     052104  103407                                                                          BLO     $52320
8991 052106                                     LET CC.EXP := CC.EXP SET.BY #ZBIT
     052106  052767  000004  126406                                                         BIS     #ZBIT,CC.EXP
8992 052114                                     LET CC.EXP := CC.EXP CLR.BY #NBIT
     052114  042767  000010  126400                                                         BIC     #NBIT,CC.EXP
8993 052122                              ELSE
     052122  000443                                                                        BR      $52321
     052124                                                                        $52320:
8994 052124                                     LET R4 := TEMP0 + #1                 ;SHIFT COUNT + 1
     052124  016704  126346                                                                 MOV     TEMP0,R4
     052130  005204                                                                         INC     R4
8995 052132                                     INLINE   <ASR  R4>
     052132  006204                                                                         ASR  R4
8996 052134                                     LET R1 := DATA01                     ;DST.LEN
     052134  016701  126424                                                                 MOV     DATA01,R1
8997 052140                                     INLINE   <ASR  R1>
     052140  006201                                                                         ASR  R1
8998 052142                                     LET R1 := R1 + #DS.EXP - R4
     052142  062701  064606                                                                 ADD     #DS.EXP,R1
     052146  160401                                                                         SUB     R4,R1
8999 052150                                     LET R0 := TEMP0                      ;SHIFT COUNT
     052150  016700  126322                                                                 MOV     TEMP0,R0
9000 052154                                     LET R2 := DATA00                     ;SRC.LEN
     052154  016702  126402                                                                 MOV     DATA00,R2
9001 052160                                     LET R3 := DATA01 - TEMP0             ;DST.LEN - SHIFT COUNT
     052160  016703  126400                                                                 MOV     DATA01,R3
     052164  166703  126306                                                                 SUB     TEMP0,R3
9002 052170                                     WHILE R2 NE #0 AND R3 NE #0 DO
     052170                                                                        $52322:
     052170  005702                                                                         TST     R2
     052172  001417                                                                         BEQ     $52323
     052174  005703                                                                         TST     R3
```

```
052176  001415                                                                    BEQ     $52323
052200                                 IF #BIT00 NOTSETIN RO THEN
052200  032700  000001                                                            BIT     #BIT00,R0
052204  001004                                                                    BNE     $52324
052206                                    LET (R1) :B= (R1) SET.BY #100
052206  152711  000100                                                            BISB    #100,(R1)
052212                                    LET R1 := R1 - #1
052212  005301                                                                    DEC     R1
052214                                 ELSE
052214  000402                                                                    BR      $52325
052216                                                                    $52324:
052216                                    LET (R1) :B= (R1) SET.BY #4
052216  152711  000004                                                            BISB    #4,(R1)
052222                                 ENDIF
052222                                                                    $52325:
052222                                 LET R0 := COMP R0                  ;TOGGLE SHIFT COUNT
052222  005100                                                                    COM     R0
052224                                 LET R2 := R2 - #1
052224  005302                                                                    DEC     R2
052226                                 LET R3 := R3 - #1
052226  005303                                                                    DEC     R3
052230                              ENDDO
052230  000757                                                                    BR      $52322
052232                                                                    $52323:
052232                           ENDIF
052232                                                                    $52321:
052232                        ENDIF
052232                                                                    $52316:
052232                     ENDIF
052232                                                                    $52315:

                  ;+
                  ; DO THE ASHP INSTRUCTION USING SIGNED PACKED (UNSIGNED
                  ; PACKED WILL BE TESTED IN THE INLINE PORTION OF THE TEST).
                  ; CHECK THE RESULTS AND REPORT ANY ERRORS.
                  ;-

052232                     LET R0 := DATA00 SET.BY #60000
052232  016700  126324                                                            MOV     DATA00,R0
052236  052700  060000                                                            BIS     #60000,R0
052242                     LET R1 := #DSBUF1
052242  012701  064440                                                            MOV     #DSBUF1,R1
052246                     LET R2 := DATA01 SET.BY #60000
052246  016702  126312                                                            MOV     DATA01,R2
052252  052702  060000                                                            BIS     #60000,R2
052256                     LET R3 := #DS.DST
052256  012703  064544                                                            MOV     #DS.DST,R3
052262                     LET R4 := DATA02
052262  016704  126300                                                            MOV     DATA02,R4
052266                     LET R5 := #0
052266  005005                                                                    CLR     R5
052270                     INLINE   <ASHP>
052270  076076                                                                    ASHP

                  ;CHECK RESULTS

052272                     LET CC := PSW
```

ASHP

```
      052272  016767  125500  126220
 9034 052300                          LET CC := CC CLR.BY #177760
      052300  042767  177760  126212
 9035 052306                          IF CC NE CC.EXP THEN
      052306  026767  126206  126206
      052314  001402
 9036 052316                              CALL ERROR
      052316  004767  005356
 9037 052322                          ENDIF
      052322
 9038 052322                          IF R0 NE #0 OR R1 NE #0 THEN
      052322  005700
      052324  001002
      052326  005701
      052330  001402
      052332
 9039 052332                              CALL ERROR
      052332  004767  005342
 9040 052336                          ENDIF
      052336
 9041 052336                          IF R4 NE #0 OR R5 NE #0 THEN
      052336  005704
      052340  001002
      052342  005705
      052344  001402
      052346
 9042 052346                              CALL ERROR
      052346  004767  005326
 9043 052352                          ENDIF
      052352
 9044 052352                          LET TEMP0 := DATA01 SET.BY #60000
      052352  016767  126206  126116
      052360  052767  060000  126110
 9045 052366                          IF R2 NE TEMP0 THEN
      052366  020267  126104
      052372  001402
 9046 052374                              CALL ERROR
      052374  004767  005300
 9047 052400                          ENDIF
      052400
 9048 052400                          IF R3 NE #DS.DST THEN
      052400  020327  064544
      052404  001402
 9049 052406                              CALL ERROR
      052406  004767  005266
 9050 052412                          ENDIF
      052412
 9051
 9052                            ;SEE IF DS.DST = DS.EXP
 9053
 9054 052412                            LET R0 := #40
      052412  012700  000040            LET R1 := #DS.EXP
 9055 052416                            LET R2 := #DS.DST
      052416  012701  064606            LET ERRFLG := #0
 9056 052422
      052422  012702  064544
 9057 052426
```

```
                        MOV     PSW,CC
                        BIC     #177760,CC
                        CMP     CC,CC.EXP
                        BEQ     $52326
                        JSR     PC,ERROR
                $52326:
                        TST     R0
                        BNE     $52327
                        TST     R1
                        BEQ     $52330
                $52327:
                        JSR     PC,ERROR
                $52330:
                        TST     R4
                        BNE     $52331
                        TST     R5
                        BEQ     $52332
                $52331:
                        JSR     PC,ERROR
                $52332:
                        MOV     DATA01,TEMP0
                        BIS     #60000,TEMP0
                        CMP     R2,TEMP0
                        BEQ     $52333
                        JSR     PC,ERROR
                $52333:
                        CMP     R3,#DS.DST
                        BEQ     $52334
                        JSR     PC,ERROR
                $52334:


                        MOV     #40,R0
                        MOV     #DS.EXP,R1
                        MOV     #DS.DST,R2
```

```
      052426  005067  126024                                                    CLR     ERRFLG
 9058 052432                          WHILE R0 HI #0 AND ERRFLG EQ #0 DO
      052432                                                                  $52335:
      052432  005700                                                            TST     R0
      052434  101412                                                            BLOS    $52336
      052436  005767  126014                                                    TST     ERRFLG
      052442  001007                                                            BNE     $52336
 9059 052444                              IFB (R1)+ NE (R2)+ THEN
      052444  122122                                                            CMPB    (R1)+,(R2)+
      052446  001403                                                            BEQ     $52337
 9060 052450                                  LET ERRFLG := #-1
      052450  012767  177777  126000                                            MOV     #-1,ERRFLG
 9061 052456                              ENDIF
      052456                                                                  $52337:
 9062 052456                              LET R0 := R0 - #1
      052456  005300                                                            DEC     R0
 9063 052460                          ENDDO
      052460  000764                                                            BR      $52335
      052462                                                                  $52336:
 9064 052462                          IF ERRFLG NE #0 THEN
      052462  005767  125770                                                    TST     ERRFLG
      052466  001402                                                            BEQ     $52340
 9065 052470                              CALL ERROR                      ;DS.DST AND DS.EXP DO NOT MATCH
      052470  004767  005204                                                    JSR     PC,ERROR
 9066 052474                          ENDIF
      052474                                                                  $52340:
 9067 052474                          LET SIGNO := COMP SIGNO             ;REPEAT THE TEST CASE WITH A NEGATIVE
      052474  005167  125750                                                    COM     SIGNO
 9068
 9069                                 ;  SIGN BEFORE MOVING ON TO THE NEXT TEST CASE
 9070
 9071 052500                              IF SIGNO EQ #0 THEN
      052500  005767  125744                                                    TST     SIGNO
      052504  001005                                                            BNE     $52341
 9072 052506                                  LET TSTCAS := TSTCAS + #6
      052506  062767  000006  126030                                            ADD     #6,TSTCAS
 9073 052514                                  LET COUNT := COUNT - #1
      052514  005367  125750                                                    DEC     COUNT
 9074 052520                              ENDIF
      052520                                                                  $52341:
 9075                                 ;UNTIL COUNT EQ #0
 9076 052520                              IF COUNT NE #0 THEN
      052520  005767  125744                                                    TST     COUNT
      052524  001402                                                            BEQ     $52342
 9077 052526                                  INLINE   <JMP   $5076>
      052526  000167  176466                                                    JMP   $5076
 9078 052532                              ENDIF
      052532                                                                  $52342:
 9079
 9080
 9081                                 ;+
 9082                                 ; DO ONE INLINE CASE - ASHPI.  USE UNSIGNED PACKED.
 9083                                 ;-
 9084
 9085 052532                              CALL   CLRDSB
      052532  004767  004136                                                    JSR     PC,CLRDSB
 9086 052536                              LET DSBUF1 :B= #231          ;9/9
```

```
                052536  112767  000231  011674
 9087 052544                    LET DSBUF1+1 :B= #231                ;9/9
                052544  112767  000231  011667                                    MOVB    #231,DSBUF1
 9088 052552                    LET DSBUF1+2 :B= #237                ;9/17
                052552  112767  000237  011662                                    MOVB    #237,DSBUF1+2
 9089 052560                    CALL CLRREG
                052560  004767  003654                                            JSR     PC,CLRREG
 9090 052564                    INLINE   <ASHPI>
                052564  076176                                                    ASHPI
 9091 052566                    INLINE   <.WORD   TASHPI>
                052566  052756                                                    .WORD   TASHPI
 9092 052570                    INLINE   <.WORD   TASHPI+4>
                052570  052762                                                    .WORD   TASHPI+4
 9093 052572                    INLINE   <.WORD   375>                ;SHIFT COUNT = -3, RND DIGIT = 0
                052572  000375                                                    .WORD   375
 9094 052574                    LET CC := PSW
                052574  016767  125176  125716                                    MOV     PSW,CC
 9095 052602                    LET CC := CC CLR.BY #177760
                052602  042767  177760  125710                                    BIC     #177760,CC
 9096 052610                    IF CC NE #0 THEN                     ;VBIT SHOULD BE SET
                052610  005767  125704                                            TST     CC
                052614  001402                                                    BEQ     $52343
 9097 052616                        CALL ERROR
                052616  004767  005056                                            JSR     PC,ERROR
 9098 052622                    ENDIF
                052622                                                  $52343:
 9099 052622                    IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN
                052622  005700                                                    TST     R0
                052624  001004                                                    BNE     $52344
                052626  005701                                                    TST     R1
                052630  001002                                                    BNE     $52344
                052632  005702                                                    TST     R2
                052634  001402                                                    BEQ     $52345
                052636                                                  $52344:
 9100 052636                        CALL ERROR
                052636  004767  005036                                            JSR     PC,ERROR
 9101 052642                    ENDIF
                052642                                                  $52345:
 9102 052642                    IF R3 NE #0 OR R4 NE #0 OR R5 NE #0 THEN
                052642  005703                                                    TST     R3
                052644  001004                                                    BNE     $52346
                052646  005704                                                    TST     R4
                052650  001002                                                    BNE     $52346
                052652  005705                                                    TST     R5
                052654  001402                                                    BEQ     $52347
                052656                                                  $52346:
 9103 052656                        CALL ERROR
                052656  004767  005016                                            JSR     PC,ERROR
 9104 052662                    ENDIF
                052662                                                  $52347:
 9105
 9106                         ;CHECK DS.DST
 9107
 9108 052662                      LET R0 := #DS.DST
                052662  012700  064544                                            MOV     #DS.DST,R0
 9109 052666                      IFB (R0)+ NE #11 THEN                ;FIRST BYTE = 0/9
                052666  122027  000011                                            CMPB    (R0)+,#11
```

ASHP

```
      052672  001402                                                          BEQ     $52350
 9110 052674                          CALL ERROR
      052674  004767  005000                                                  JSR     PC,ERROR
 9111 052700                        ENDIF
      052700                                                          $52350:
 9112 052700                        IFB (R0)+ NE #237 THEN          ;SECOND BYTE = 9/17
      052700  122027  000237                                                  CMPB    (R0)+,#237
      052704  001402                                                          BEQ     $52351
 9113 052706                          CALL ERROR
      052706  004767  004766                                                  JSR     PC,ERROR
 9114 052712                        ENDIF
      052712                                                          $52351:
 9115 052712                        LET ERRFLG := #0
      052712  005067  125540                                                  CLR     ERRFLG
 9116 052716                        WHILE ERRFLG NE #0 AND R0 NE #DS.DST+37 DO
      052716                                                          $52352:
      052716  005767  125534                                                  TST     ERRFLG
      052722  001413                                                          BEQ     $52353
      052724  020027  064603                                                  CMP     R0,#DS.DST+37
      052730  001410                                                          BEQ     $52353
 9117 052732                          IFB (R0)+ NE #0 THEN
      052732  105720                                                          TSTB    (R0)+
      052734  001405                                                          BEQ     $52354
 9118 052736                            LET ERRFLG := #-1
      052736  012767  177777  125512                                          MOV     #-1,ERRFLG
 9119 052744                            CALL ERROR
      052744  004767  004730                                                  JSR     PC,ERROR
 9120 052750                          ENDIF
      052750                                                          $52354:
 9121 052750                        ENDDO
      052750  000762                                                          BR      $52352
      052752                                                          $52353:
 9122 052752                        INLINE   <JMP    CIS7END>
      052752  000167  000010                                                  JMP     CIS7END
 9123
 9124                             ;+
 9125                             ; INLINE TEST CASE
 9126                             ;-
 9127
 9128 052756  070005              TASHPI: .WORD   070005          ;SRC.DSCR, UNSIGNED PACKED, SRC.LEN = 5
 9129 052760  064440                      .WORD   DSBUF1          ;SRC.DSCR
 9130 052762  070002                      .WORD   070002          ;DST.DSCR, UNSIGNED PACKED, DST.LEN = 2
 9131 052764  064544                      .WORD   DS.DST          ;DST.DSCR
 9132
 9133 052766                      INLINE   <CIS7END:>
      052766                                                          CIS7END:
 9134
```

ASHP

```
 9136                                ;*****************************************************************
 9137                                ;*****************************************************************
 9138                                ;++
 9139                                ;               C I S 8     M O D U L E
 9140                                ;
 9141                                ;  THIS MODULE TESTS THE PACKED MULTIPLY INSTRUCTION: MULP.
 9142                                ;--
 9143                                ;*****************************************************************
 9144                                ;*****************************************************************
 9145
 9146
 9147                                .SBTTL  CIS8 INITIALIZATION
 9148                                ;*****************************************************************
 9149                                ;
 9150                                ;  CIS8 INITIALIZATION
 9151                                ;
 9152                                ;*****************************************************************
 9153
 9154                                ;IF SWR<5:0> ARE CLEARED OR IF SWR<4> IS SET THEN EXECUTE
 9155                                ;  THE CIS8 MODULE; OTHERWISE, SKIP TO THE NEXT CIS MODULE.
 9156
 9157 052766                           INLINE   <SCOPE>                                          SCOPE
      052766  000004
 9158 052770                           INLINE <CKSWR>                                            CKSWR
      052770  104406
 9159 052772                           LET TEMPO := @SWR CLR.BY #177700                          MOV     @SWR,TEMPO
      052772  017767  125430  125476                                                             BIC     #177700,TEMPO
      053000  042767  177700  125470
 9160 053006                           IF TEMPO NE #0 THEN                                       TST     TEMPO
      053006  005767  125464                                                                     BEQ     $52355
      053012  001406
 9161 053014                               IF #BIT4 NOTSETIN TEMPO THEN                          BIT     #BIT4,TEMPO
      053014  032767  000020  125454                                                             BNE     $52356
      053022  001002
 9162 053024                                   INLINE   <JMP   CIS8END>                          JMP   CIS8END
      053024  000167  001062
 9163 053030                               ENDIF
      053030                                                                                $52356:
 9164 053030                           ENDIF                                                 $52355:
      053030
 9165 053030                           LET CTL := #8.                ;CONTROL CHIP NUMBER        MOV     #8.,CTL
      053030  012767  000010  125510
 9166 053036                           LET ERRFLG := #0      ;INITIALIZE ERROR FLAG              CLR     ERRFLG
      053036  005067  125414
 9167 053042                           LET PSW := #340                                           MOV     #340,PSW
      053042  012767  000340  124726
 9168 053050                           LET ERR := #1                                             MOV     #1,ERR
      053050  012767  000001  125474
 9169
```

MULP

```
                                .SBTTL  MULP
                                ;*******************************
                                ;+
                                ;       MULP TEST
                                ;-
                                ;*******************************

                                ROUTINE XMULP
053056                                                                  XMULP:
053056
                                   INLINE <SCOPE>
053056  000004                                                                SCOPE
                                   INLINE <NOP>
053060  000240                                                                NOP
                                   LET $TIMES := #20            ;ITERATION COUNT
053062  012767  000020  125314                                               MOV     #20,$TIMES
                                   LET COUNT := TAB89           ;# OF TEST CASES
053070  016767  002166  125372                                               MOV     TAB89,COUNT
                                   LET TSTCAS := #TAB89+2               ;POINT TO FIRST TEST CASE
053076  012767  055264  125440                                               MOV     #TAB89+2,TSTCAS

                                ;+
                                ;       CLR R0.EXT TO R3.EXP
                                ;-

                                   INCR  R0 FROM #0 TO #6 BY #2
053104  005000                                                                CLR     R0
053106  000402                                                                BR      $52361
053110                                                              $52362:
053110  062700  000002                                                        ADD     #2,R0
053114                                                              $52361:
053114  020027  000006                                                        CMP     R0,#6
053120  003003                                                                BGT     $52363
                                       INLINE <CLR R0.EXP(R0)>
053122  005060  000524                                                        CLR R0.EXP(R0)
                                   ENDINC
053126  000770                                                                BR      $52362
053130                                                              $52363:

                                ;REPEAT OUTER LOOP FOR TEST CASES

                                   INLINE <$5074:>
053130                                                                      $5074:
                                   CALL CLRDSB                  ;CLR DS BUFFERS
053130  004767  003540                                                        JSR     PC,CLRDSB

                                ;+
                                ;       LOAD DATAXX AND CC.EXP FROM TEST CASE
                                ;-

                                   LET R0 := TSTCAS
053134  016700  125404                                                        MOV     TSTCAS,R0
                                   LET DATA00 := (R0)+          ;SRC1 LENGTH
053140  012067  125416                                                        MOV     (R0)+,DATA00
                                   LET DATA01 := (R0)+          ;POINTER TO SRC1 DATA
053144  012067  125414                                                        MOV     (R0)+,DATA01
                                   LET DATA02 := (R0)+          ;SRC2 LENGTH
053150  012067  125412                                                        MOV     (R0)+,DATA02
```

MULP

```
 9206 053154                            LET DATA03 := (R0)+          ;POINTER TO SRC2 DATA
      053154  012067  125410                                                 MOV     (R0)+,DATA03
 9207 053160                            LET DATA04 := (R0)+          ;DEST LENGTH
      053160  012067  125406                                                 MOV     (R0)+,DATA04
 9208 053164                            LET DATA05 := (R0)+          ;POINTER TO DS.EXP DATA (PRODUCT)
      053164  012067  125404                                                 MOV     (R0)+,DATA05
 9209 053170                            LET CC.EXP := (R0)+          ;CC.EXP (PRODUCT)
      053170  012067  125326                                                 MOV     (R0)+,CC.EXP
 9210
 9211                                ;+
 9212                                ;       PACK BUFFERS FOR SRC1,SRC2, & DEST
 9213                                ;-
 9214
 9215 053174                            LET R5 := #CHRSET                    ;R5 IS OPERAND STACK FOR ROUTINE
      053174  012705  064036                                                 MOV     #CHRSET,R5
 9216 053200                            CALL PACK IN <#DSBUF1,DATA01,DATA00> ;SRC1
      053200  010546                                                         MOV     R5,-(SP)
      053202  016745  125354                                                 MOV     DATA00,-(R5)
      053206  016745  125352                                                 MOV     DATA01,-(R5)
      053212  012745  064440                                                 MOV     #DSBUF1,-(R5)
      053216  004767  004032                                                 JSR     PC,PACK
      053222  012605                                                        MOV     (SP)+,R5
 9217 053224                            CALL PACK IN <#DSBUF2,DATA03,DATA02> ;SRC2
      053224  010546                                                         MOV     R5,-(SP)
      053226  016745  125334                                                 MOV     DATA02,-(R5)
      053232  016745  125332                                                 MOV     DATA03,-(R5)
      053236  012745  064502                                                 MOV     #DSBUF2,-(R5)
      053242  004767  004006                                                 JSR     PC,PACK
      053246  012605                                                        MOV     (SP)+,R5
 9218 053250                            CALL PACK IN <#DS.EXP,DATA05,DATA04> ;DEST
      053250  010546                                                         MOV     R5,-(SP)
      053252  016745  125314                                                 MOV     DATA04,-(R5)
      053256  016745  125312                                                 MOV     DATA05,-(R5)
      053262  012745  064606                                                 MOV     #DS.EXP,-(R5)
      053266  004767  003762                                                 JSR     PC,PACK
      053272  012605                                                        MOV     (SP)+,R5
 9219
 9220                                ;+
 9221                                ;       LOAD GPR'S FROM DATAXX & ADD DATA TYPE
 9222                                ;-
 9223
 9224 053274                            LET R0 := DATA00 SET.BY #60000
      053274  016700  125262                                                 MOV     DATA00,R0
      053300  052700  060000                                                 BIS     #60000,R0
 9225 053304                            LET R1 := #DSBUF1
      053304  012701  064440                                                 MOV     #DSBUF1,R1
 9226 053310                            LET R2 := DATA02 SET.BY #60000
      053310  016702  125252                                                 MOV     DATA02,R2
      053314  052702  060000                                                 BIS     #60000,R2
 9227 053320                            LET R3 := #DSBUF2
      053320  012703  064502                                                 MOV     #DSBUF2,R3
 9228 053324                            LET R4 := DATA04 SET.BY #60000
      053324  016704  125242                                                 MOV     DATA04,R4
      053330  052704  060000                                                 BIS     #60000,R4
 9229 053334                            LET R5 := #DS.DST
      053334  012705  064544                                                 MOV     #DS.DST,R5
 9230
```

```
 9231                                   ;+
 9232                                   ;       FINISH SET UP & DO CIS INSTRUCTION
 9233                                   ;-
 9234
 9235
 9236 053340                               LET R4.EXP := R4
      053340  010467  125170                                                    MOV     R4,R4.EXP
 9237 053344                               LET R5.EXP := R5
      053344  010567  125166                                                    MOV     R5,R5.EXP
 9238 053350                               INLINE <MULP>
      053350  076074                                                            MULP
 9239
 9240                                   ;+
 9241                                   ;       CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.
 9242                                   ;-
 9243
 9244 053352                               LET CC := PSW
      053352  016767  124420  125140                                            MOV     PSW,CC
 9245 053360                               LET CC := CC CLR.BY #177760
      053360  042767  177760  125132                                            BIC     #177760,CC
 9246 053366                               IF CC NE CC.EXP THEN
      053366  026767  125126  125126                                            CMP     CC,CC.EXP
      053374  001402                                                           BEQ     $52364
 9247 053376                                   CALL ERROR
      053376  004767  004276                                                    JSR     PC,ERROR
 9248 053402                               ENDIF
      053402                                                            $52364:
 9249 053402                               IF R0 NE R0.EXP THEN
      053402  020067  125116                                                    CMP     R0,R0.EXP
      053406  001402                                                           BEQ     $52365
 9250 053410                                   CALL ERROR
      053410  004767  004264                                                    JSR     PC,ERROR
 9251 053414                               ENDIF
      053414                                                            $52365:
 9252 053414                               IF R1 NE R1.EXP THEN
      053414  020167  125106                                                    CMP     R1,R1.EXP
      053420  001402                                                           BEQ     $52366
 9253 053422                                   CALL ERROR
      053422  004767  004252                                                    JSR     PC,ERROR
 9254 053426                               ENDIF
      053426                                                            $52366:
 9255 053426                               IF R2 NE R2.EXP THEN
      053426  020267  125076                                                    CMP     R2,R2.EXP
      053432  001402                                                           BEQ     $52367
 9256 053434                                   CALL ERROR
      053434  004767  004240                                                    JSR     PC,ERROR
 9257 053440                               ENDIF
      053440                                                            $52367:
 9258 053440                               IF R3 NE R3.EXP THEN
      053440  020367  125066                                                    CMP     R3,R3.EXP
      053444  001402                                                           BEQ     $52370
 9259 053446                                   CALL ERROR
      053446  004767  004226                                                    JSR     PC,ERROR
 9260 053452                               ENDIF
      053452                                                            $52370:
 9261 053452                               IF R4 NE R4.EXP THEN
      053452  020467  125056                                                    CMP     R4,R4.EXP
```

```
053456  001402                                                  BEQ     $52371
053460                              CALL ERROR
053460  004767  004214                                          JSR     PC,ERROR
053464                          ENDIF
053464                                                  $52371:
053464                          IF R5 NE R5.EXP THEN
053464  020567  125046                                          CMP     R5,R5.EXP
053470  001402                                                  BEQ     $52372
053472                              CALL ERROR
053472  004767  004202                                          JSR     PC,ERROR
053476                          ENDIF
053476                                                  $52372:


                        ;+
                        ;       CHECK RESULTS VS EXPECTED RESULTS
                        ;-

053476                          LET R0 := #DS.DST
053476  012700  064544                                          MOV     #DS.DST,R0
053502                          LET R1 := #DS.EXP
053502  012701  064606                                          MOV     #DS.EXP,R1
053506                          LET ERRFLG := #0
053506  005067  124744                                          CLR     ERRFLG
053512                          WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO
053512                                                  $52373:
053512  020027  064604                                          CMP     R0,#DS.DST+40
053516  001410                                                  BEQ     $52374
053520  005767  124732                                          TST     ERRFLG
053524  001005                                                  BNE     $52374
053526                              IFB (R0)+ NE (R1)+ THEN
053526  122021                                                  CMPB    (R0)+,(R1)+
053530  001402                                                  BEQ     $52375
053532                                  CALL ERROR
053532  004767  004142                                          JSR     PC,ERROR
053536                              ENDIF
053536                                                  $52375:
053536                          ENDDO
053536  000765                                                  BR      $52373
053540                                                  $52374:
053540                          LET TSTCAS := TSTCAS + #22
053540  062767  000022  124776                                  ADD     #22,TSTCAS
053546                          LET COUNT := COUNT - #1
053546  005367  124716                                          DEC     COUNT
                        ;UNTIL COUNT EQ #0
053552                          IF COUNT NE #0 THEN
053552  005767  124712                                          TST     COUNT
053556  001402                                                  BEQ     $52376
053560                              INLINE <JMP $5074>
053560  000167  177344                                          JMP $5074
053564                          ENDIF
053564                                                  $52376:


                        ;+
                        ;       INLINE CASE - MULPI, USING UNSIGNED PACKED DATA
                        ;-

```

```
MULP

 9292 053564                    CALL CLRDSB
      053564  004767  003104                                                JSR     PC,CLRDSB
 9293
 9294                           ;+
 9295                           ;       LOAD DATAXX AND CC.EXP FROM TEST CASE
 9296                           ;-
 9297
 9298 053570                    LET R0 := #TAB89+2
      053570  012700  055264                                                MOV     #TAB89+2,R0
 9299 053574                    LET DATA00 := (R0)+             ;SRC1 LENGTH
      053574  012067  124762                                                MOV     (R0)+,DATA00
 9300 053600                    LET DATA01 := (R0)+             ;POINTER TO SRC1 DATA
      053600  012067  124760                                                MOV     (R0)+,DATA01
 9301 053604                    LET DATA02 := (R0)+             ;SRC2 LENGTH
      053604  012067  124756                                                MOV     (R0)+,DATA02
 9302 053610                    LET DATA03 := (R0)+             ;POINTER TO SRC2 DATA
      053610  012067  124754                                                MOV     (R0)+,DATA03
 9303 053614                    LET DATA04 := (R0)+             ;DEST LENGTH
      053614  012067  124752                                                MOV     (R0)+,DATA04
 9304 053620                    LET DATA05 := (R0)+             ;POINTER TO DS.EXP DATA (PRODUCT)
      053620  012067  124750                                                MOV     (R0)+,DATA05
 9305
 9306                           ;+
 9307                           ;       PACK BUFFERS FOR SRC1,SRC2, & DEST
 9308                           ;-
 9309
 9310 053624                    LET R5 := #CHRSET                       ;R5 IS OPERAND STACK FOR ROUTINE
      053624  012705  064036                                                MOV     #CHRSET,R5
 9311 053630                    CALL PACK IN <#DSBUF1,DATA01,DATA00> ;SRC1
      053630  010546                                                        MOV     R5,-(SP)
      053632  016745  124724                                                MOV     DATA00,-(R5)
      053636  016745  124722                                                MOV     DATA01,-(R5)
      053642  012745  064440                                                MOV     #DSBUF1,-(R5)
      053646  004767  003402                                                JSR     PC,PACK
      053652  012605                                                        MOV     (SP)+,R5
 9312 053654                    CALL PACK IN <#DSBUF2,DATA03,DATA02> ;SRC2
      053654  010546                                                        MOV     R5,-(SP)
      053656  016745  124704                                                MOV     DATA02,-(R5)
      053662  016745  124702                                                MOV     DATA03,-(R5)
      053666  012745  064502                                                MOV     #DSBUF2,-(R5)
      053672  004767  003356                                                JSR     PC,PACK
      053676  012605                                                        MOV     (SP)+,R5
 9313 053700                    CALL PACK IN <#DS.EXP,DATA05,DATA04> ;DEST
      053700  010546                                                        MOV     R5,-(SP)
      053702  016745  124664                                                MOV     DATA04,-(R5)
      053706  016745  124662                                                MOV     DATA05,-(R5)
      053712  012745  064606                                                MOV     #DS.EXP,-(R5)
      053716  004767  003332                                                JSR     PC,PACK
      053722  012605                                                        MOV     (SP)+,R5
 9314 053724                    CALL CLRREG                     ;ENSURE GPR'S 0-5 ARE ZERO
      053724  004767  002510                                                JSR     PC,CLRREG
 9315
 9316 053730                    INLINE <MULPI>
      053730  076174                                                        MULPI
 9317 053732                    INLINE <.WORD TMULPI>
      053732  056064                                                        .WORD TMULPI
 9318 053734                    INLINE <.WORD TMULPI+4>
```

MULP

```
      053734  056070                          INLINE <.WORD TMULPI+10>                                .WORD TMULPI+4
 9319 053736                                                                                          .WORD TMULPI+10
      053736  056074                          LET CC := PSW                                           MOV     PSW,CC
 9320 053740
      053740  016767  124032  124552          LET CC := CC CLR.BY #177760                             BIC     #177760,CC
 9321 053746
      053746  042767  177760  124544          IF CC NE #0 THEN                                        TST     CC
 9322 053754                                                                                          BEQ     $52377
      053754  005767  124540                      CALL ERROR                                          JSR     PC,ERROR
      053760  001402
 9323 053762                                   ENDIF                                           $52377:
      053762  004767  003712
 9324 053766
      053766                                                                                  
 9325
 9326                                 ;+
 9327                                 ;       ENSURE GPR'S STILL ZERO
 9328                                 ;-
 9329
 9330 053766                          IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN                        TST     R0
      053766  005700                                                                                 BNE     $52400
      053770  001004                                                                                 TST     R1
      053772  005701                                                                                 BNE     $52400
      053774  001002                                                                                 TST     R2
      053776  005702                                                                                 BEQ     $52401
      054000  001402                                                                          $52400:
      054002                                  CALL ERROR
 9331 054002                                                                                          JSR     PC,ERROR
      054002  004767  003672
 9332 054006                              ENDIF
      054006                                                                                  $52401:
 9333 054006                              IF R3 NE #0 OR R4 NE #0 OR  R5 NE #0 THEN                   TST     R3
      054006  005703                                                                                 BNE     $52402
      054010  001004                                                                                 TST     R4
      054012  005704                                                                                 BNE     $52402
      054014  001002                                                                                 TST     R5
      054016  005705                                                                                 BEQ     $52403
      054020  001402                                                                          $52402:
      054022
 9334 054022                                  CALL ERROR                                              JSR     PC,ERROR
      054022  004767  003652
 9335 054026                              ENDIF
      054026                                                                                  $52403:
 9336
 9337                                 ;+
 9338                                 ;       CHECK RESULTS
 9339                                 ;-
 9340
 9341 054026                              LET R0 := #DS.DST                                           MOV     #DS.DST,R0
      054026  012700  064544
 9342 054032                              LET R1 := #DS.EXP               ;EXPECTED DATA              MOV     #DS.EXP,R1
      054032  012701  064606
 9343 054036                              LET R2 := TMULPI+10 CLR.BY #70000     ;GET STRING LENGTH    MOV     TMULPI+10,R2
      054036  016702  002032                                                                         BIC     #70000,R2
      054042  042702  070000
 9344 054046                              INLINE       <ASR    R2>               ;LENGTH/2 (PACKING FACTOR)
      054046  006202                                                                                 ASR     R2
```

MULP

```
 9345 054050                        LET R2 := R2 + #DS.EXP                ;R2 POINTS TO MSD/SIGN
      054050  062702  064606                                                      ADD     #DS.EXP,R2
 9346 054054                        LET (R2) :B= (R2) SET.BY #17 ;MAKE IT UNSIGNED
      054054  152712  000017                                                     BISB    #17,(R2)
 9347 054060                        LET ERRFLG := #0
      054060  005067  124372                                                     CLR     ERRFLG
 9348 054064                        WHILE R0 NE #DS.DST+37 AND ERRFLG EQ #0 DO
      054064                                                             $52404:
      054064  020027  064603                                                     CMP     R0,#DS.DST+37
      054070  001410                                                            BEQ     $52405
      054072  005767  124360                                                     TST     ERRFLG
      054076  001005                                                            BNE     $52405
 9349 054100                            IFB (R0)+ NE (R1)+ THEN
      054100  122021                                                            CMPB    (R0)+,(R1)+
      054102  001402                                                            BEQ     $52406
 9350 054104                                CALL ERROR
      054104  004767  003570                                                    JSR     PC,ERROR
 9351 054110                            ENDIF
      054110                                                             $52406:
 9352 054110                        ENDDO
      054110  000765                                                            BR      $52404
      054112                                                             $52405:
 9353
 9354 054112                    INLINE <CIS8END:>
      054112                                                                     CIS8END:
 9355
```

```
                        ;*****************************************************************
                        ;*****************************************************************
                        ;++
                        ;                C I S 9     M O D U L E
                        ;
                        ;  THIS MODULE TESTS THE PACKED DIVIDE INSTRUCTION: DIVP.
                        ;--
                        ;*****************************************************************
                        ;*****************************************************************


                        .SBTTL  CIS9 INITIALIZATION
                        ;*****************************************************************
                        ;
                        ;  CIS9 INITIALIZATION
                        ;
                        ;*****************************************************************

                        ;IF SWR<5:0> ARE CLEARED OR IF SWR<5> IS SET THEN EXECUTE
                        ;  THE CIS9 MODULE; OTHERWISE, SKIP TO END OF PASS.

                           INLINE   <SCOPE>                                          SCOPE
054112  000004
                           INLINE <CKSWR>                                            CKSWR
054114  104406
                           LET TEMPO := @SWR CLR.BY #177700
054116  017767  124304  124352                                                      MOV     @SWR,TEMPO
054124  042767  177700  124344                                                      BIC     #177700,TEMPO
                        IF TEMPO NE #0 THEN
054132  005767  124340                                                            TST     TEMPO
054136  001406                                                                    BEQ     $52407
                            IF #BIT5 NOTSETIN TEMPO THEN
054140  032767  000040  124330                                                    BIT     #BIT5,TEMPO
054146  001002                                                                    BNE     $52410
                                INLINE   <JMP   CIS9END>
054150  000167  001724                                                            JMP   CIS9END
                            ENDIF
054154                                                                        $52410:
                        ENDIF
054154                                                                        $52407:
                        LET CTL := #9.                  ;CONTROL CHIP NUMBER
054154  012767  000011  124364                                                      MOV     #9.,CTL
                        LET ERRFLG := #0     ;INITIALIZE ERROR FLAG
054162  005067  124270                                                            CLR     ERRFLG
                        LET PSW := #340
054166  012767  000340  123602                                                    MOV     #340,PSW
                        LET ERR := #1
054174  012767  000001  124350                                                    MOV     #1,ERR
```

```
 9392                                .SBTTL  DIVP
 9393                                ;******************************
 9394                                ;*
 9395                                ;       DIVP TEST
 9396                                ;-
 9397                                ;******************************
 9398
 9399 054202                        ROUTINE XDIVP                                                   XDIVP:
      054202
 9400 054202                           INLINE <SCOPE>                                                       SCOPE
      054202  000004
 9401 054204                           INLINE <NOP>                                                         NOP
      054204  000240
 9402 054206                           LET $TIMES := #20           ;ITERATION COUNT                         MOV     #20,$TIMES
      054206  012767  000020  124170
 9403 054214                           LET COUNT := TAB89          ;# OF TEST CASES                         MOV     TAB89,COUNT
      054214  016767  001042  124246
 9404 054222                           LET TSTCAS := #TAB89+2              ;POINT TO FIRST TEST CASE
      054222  012767  055264  124314                                                                       MOV     #TAB89+2,TSTCAS
 9405
 9406                                ;*
 9407                                ;       CLR R0.EXT TO R3.EXP
 9408                                ;-
 9409
 9410 054230                           INCR  R0 FROM #0 TO #6 BY #2
      054230  005000                                                                                       CLR     R0
      054232  000402                                                                                       BR      $52413
      054234                                                                                        $52414:
      054234  062700  000002                                                                               ADD     #2,R0
      054240                                                                                        $52413:
      054240  020027  000006                                                                               CMP     R0,#6
      054244  003003                                                                                       BGT     $52415
 9411 054246                               INLINE <CLR R0.EXP(R0)>
      054246  005060  000524                                                                               CLR R0.EXP(R0)
 9412 054252                           ENDINC
      054252  000770                                                                                       BR      $52414
      054254                                                                                        $52415:
 9413                                ;REPEAT OUTER LOOP FOR TEST CASES
 9414 054254                           INLINE <$5075:>
      054254                                                                                             $5075:
 9415 054254                           CALL CLRDSB                 ;CLR DS BUFFERS                          JSR     PC,CLRDSB
      054254  004767  002414
 9416
 9417                                ;*
 9418                                ;       LOAD DATAXX AND CC.EXP FROM TEST CASE
 9419                                ;-
 9420
 9421 054260                           LET R0 := TSTCAS
      054260  016700  124260                                                                               MOV     TSTCAS,R0
 9422 054264                           LET DATA00 := (R0)+         ;SRC1 LENGTH                             MOV     (R0)+,DATA00
      054264  012067  124272
 9423 054270                           LET DATA01 := (R0)+         ;POINTER TO SRC1 DATA                    MOV     (R0)+,DATA01
      054270  012067  124270
 9424 054274                           LET DATA02 := (R0)+         ;SRC2 LENGTH                             MOV     (R0)+,DATA02
      054274  012067  124266
 9425 054300                           LET DATA03 := (R0)+         ;POINTER TO SRC2 DATA                    MOV     (R0)+,DATA03
      054300  012067  124264
```

DIVP

```
 9426 054304                        LET DATA04 := (R0)+          ;DEST LENGTH
      054304  012067  124262                                                MOV     (R0)+,DATA04
 9427 054310                        LET DATA05 := (R0)+          ;POINTER TO DS.EXP DATA (PRODUCT)
      054310  012067  124260                                                MOV     (R0)+,DATA05
 9428 054314                        LET CC.EXP := (R0)+          ;CC.EXP (PRODUCT)
      054314  012067  124202                                                MOV     (R0)+,CC.EXP
 9429 054320                        LET DATA05 := (R0)+          ;POINTER TO DS.EXP DATA (QUOTIENT)
      054320  012067  124250                                                MOV     (R0)+,DATA05
 9430 054324                        LET CC.EXP := (R0)+          ;CC.EXP (QUOTIENT)
      054324  012067  124172                                                MOV     (R0)+,CC.EXP
 9431
 9432                           ;+
 9433                           ;       PACK BUFFERS FOR SRC1,SRC2, & DEST
 9434                           ;-
 9435
 9436 054330                        LET R5 := #CHRSET                    ;R5 IS OPERAND STACK FOR ROUTINE
      054330  012705  064036                                                MOV     #CHRSET,R5
 9437 054334                        CALL PACK IN <#DSBUF1,DATA01,DATA00> ;SRC1
      054334  010546                                                        MOV     R5,-(SP)
      054336  016745  124220                                                MOV     DATA00,-(R5)
      054342  016745  124216                                                MOV     DATA01,-(R5)
      054346  012745  064440                                                MOV     #DSBUF1,-(R5)
      054352  004767  002676                                                JSR     PC,PACK
      054356  012605                                                        MOV     (SP)+,R5
 9438 054360                        CALL PACK IN <#DSBUF2,DATA03,DATA02> ;SRC2
      054360  010546                                                        MOV     R5,-(SP)
      054362  016745  124200                                                MOV     DATA02,-(R5)
      054366  016745  124176                                                MOV     DATA03,-(R5)
      054372  012745  064502                                                MOV     #DSBUF2,-(R5)
      054376  004767  002652                                                JSR     PC,PACK
      054402  012605                                                        MOV     (SP)+,R5
 9439 054404                        CALL PACK IN <#DS.EXP,DATA05,DATA04> ;DEST
      054404  010546                                                        MOV     R5,-(SP)
      054406  016745  124160                                                MOV     DATA04,-(R5)
      054412  016745  124156                                                MOV     DATA05,-(R5)
      054416  012745  064606                                                MOV     #DS.EXP,-(R5)
      054422  004767  002626                                                JSR     PC,PACK
      054426  012605                                                        MOV     (SP)+,R5
 9440
 9441                           ;+
 9442                           ;       LOAD GPR'S FROM DATAXX & ADD DATA TYPE
 9443                           ;-
 9444
 9445 054430                        LET R0 := DATA00 SET.BY #60000
      054430  016700  124126                                                MOV     DATA00,R0
      054434  052700  060000                                                BIS     #60000,R0
 9446 054440                        LET R1 := #DSBUF1
      054440  012701  064440                                                MOV     #DSBUF1,R1
 9447 054444                        LET R2 := DATA02 SET.BY #60000
      054444  016702  124116                                                MOV     DATA02,R2
      054450  052702  060000                                                BIS     #60000,R2
 9448 054454                        LET R3 := #DSBUF2
      054454  012703  064502                                                MOV     #DSBUF2,R3
 9449 054460                        LET R4 := DATA04 SET.BY #60000
      054460  016704  124106                                                MOV     DATA04,R4
      054464  052704  060000                                                BIS     #60000,R4
 9450 054470                        LET R5 := #DS.DST
```

```
DIVP
        054470  012705  064544                                                  MOV     #DS.DST,R5
   9451
   9452                                 ;+
   9453                                 ;       FINISH SET UP & DO CIS INSTRUCTION
   9454                                 ;-
   9455
   9456
   9457 054474                          LET R4.EXP := R4
        054474  010467  124034                                                  MOV     R4,R4.EXP
   9458 054500                          LET R5.EXP := R5
        054500  010567  124032                                                  MOV     R5,R5.EXP
   9459 054504                          INLINE <DIVP>
        054504  076075                                                          DIVP
   9460
   9461                                 ;+
   9462                                 ;       CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.
   9463                                 ;-
   9464
   9465 054506                          LET CC := PSW
        054506  016767  123264  124004                                          MOV     PSW,CC
   9466 054514                          LET CC := CC CLR.BY #177760
        054514  042767  177760  123776                                          BIC     #177760,CC
   9467 054522                          IF CC NE CC.EXP THEN
        054522  026767  123772  123772                                          CMP     CC,CC.EXP
        054530  001402                                                          BEQ     $52416
   9468 054532                              CALL ERROR
        054532  004767  003142                                                  JSR     PC,ERROR
   9469 054536                          ENDIF
        054536                                                          $52416:
   9470 054536                          IF R0 NE R0.EXP THEN
        054536  020067  123762                                                  CMP     R0,R0.EXP
        054542  001402                                                          BEQ     $52417
   9471 054544                              CALL ERROR
        054544  004767  003130                                                  JSR     PC,ERROR
   9472 054550                          ENDIF
        054550                                                          $52417:
   9473 054550                          IF R1 NE R1.EXP THEN
        054550  020167  123752                                                  CMP     R1,R1.EXP
        054554  001402                                                          BEQ     $52420
   9474 054556                              CALL ERROR
        054556  004767  003116                                                  JSR     PC,ERROR
   9475 054562                          ENDIF
        054562                                                          $52420:
   9476 054562                          IF R2 NE R2.EXP THEN
        054562  020267  123742                                                  CMP     R2,R2.EXP
        054566  001402                                                          BEQ     $52421
   9477 054570                              CALL ERROR
        054570  004767  003104                                                  JSR     PC,ERROR
   9478 054574                          ENDIF
        054574                                                          $52421:
   9479 054574                          IF R3 NE R3.EXP THEN
        054574  020367  123732                                                  CMP     R3,R3.EXP
        054600  001402                                                          BEQ     $52422
   9480 054602                              CALL ERROR
        054602  004767  003072                                                  JSR     PC,ERROR
   9481 054606                          ENDIF
        054606                                                          $52422:
```

```
 9482 054606                        IF R4 NE R4.EXP THEN                                    CMP     R4,R4.EXP
      054606  020467  123722                                                                BEQ     $52423
      054612  001402
 9483 054614                            CALL ERROR                                          JSR     PC,ERROR
      054614  004767  003060
 9484 054620                        ENDIF                                           $52423:
      054620                        IF R5 NE R5.EXP THEN                                    CMP     R5,R5.EXP
 9485 054620                                                                                BEQ     $52424
      054620  020567  123712
      054624  001402
 9486 054626                            CALL ERROR                                          JSR     PC,ERROR
      054626  004767  003046
 9487 054632                        ENDIF                                           $52424:
      054632
 9488
 9489
 9490                           ;+
 9491                           ;       CHECK RESULTS VS EXPECTED RESULTS
 9492                           ;-
 9493
 9494 054632                        LET R0 := #DS.DST                                       MOV     #DS.DST,R0
      054632  012700  064544
 9495 054636                        LET R1 := #DS.EXP                                       MOV     #DS.EXP,R1
      054636  012701  064606
 9496 054642                        LET ERRFLG := #0                                        CLR     ERRFLG
      054642  005067  123610
 9497 054646                        WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO      $52425:
      054646                                                                                CMP     R0,#DS.DST+40
      054646  020027  064604                                                              BEQ     $52426
      054632  001410                                                                      TST     ERRFLG
      054654  005767  123576                                                              BNE     $52426
      054660  001005
 9498 054662                            IFB (R0)+ NE (R1)+ THEN                             CMPB    (R0)+,(R1)+
      054662  122021                                                                      BEQ     $52427
      054664  001402
 9499 054666                                CALL ERROR                                      JSR     PC,ERROR
      054666  004767  003006
 9500 054672                            ENDIF                                       $52427:
      054672
 9501 054672                        ENDDO                                                   BR      $52425
      054672  000765                                                                  $52426:
      054674
 9502 054674                        LET TSTCAS := TSTCAS + #22                              ADD     #22,TSTCAS
      054674  062767  000022  123642
 9503 054702                        LET COUNT := COUNT - #1                                 DEC     COUNT
      054702  005367  123562
 9504                           ;UNTIL COUNT EQ #0
 9505 054706                        IF COUNT NE #0 THEN                                     TST     COUNT
      054706  005767  123556                                                                BEQ     $52430
      054712  001402
 9506 054714                            INLINE <JMP $5075>                                  JMP $5075
      054714  000167  177334
 9507 054720                        ENDIF                                           $52430:
      054720
 9508
 9509                           ;+
 9510                           ;       INLINE CASE - DIVPI, USING UNSIGNED PACKED DATA
```

```
                                ;-
                                ;
                                CALL CLRDSB
 054720  004767  001750                                                  JSR     PC,CLRDSB
                                ;
                                ;+
                                ;       LOAD DATAXX AND CC.EXP FROM TEST CASE
                                ;-
                                ;
                                LET R0 := #TAB89+2
 054724  012700  055264                                                  MOV     #TAB89+2,R0
                                LET DATA00 := (R0)+             ;SRC1 LENGTH
 054730  012067  123626                                                  MOV     (R0)+,DATA00
                                LET DATA01 := (R0)+             ;POINTER TO SRC1 DATA
 054734  012067  123624                                                  MOV     (R0)+,DATA01
                                LET DATA02 := (R0)+             ;SRC2 LENGTH
 054740  012067  123622                                                  MOV     (R0)+,DATA02
                                LET DATA03 := (R0)+             ;POINTER TO SRC2 DATA
 054744  012067  123620                                                  MOV     (R0)+,DATA03
                                LET DATA04 := (R0)+             ;DEST LENGTH
 054750  012067  123616                                                  MOV     (R0)+,DATA04
                                LET DATA05 := (R0)+             ;POINTER TO DS.EXP DATA (PRODUCT)
 054754  012067  123614                                                  MOV     (R0)+,DATA05
                                LET CC.EXP := (R0)+
 054760  012067  123536                                                  MOV     (R0)+,CC.EXP
                                LET DATA05 := (R0)+             ;POINTER TO DS.EXP DATA (QUOTIENT)
 054764  012067  123604                                                  MOV     (R0)+,DATA05
                                ;
                                ;+
                                ;       PACK BUFFERS FOR SRC1,SRC2, & DEST
                                ;-
                                ;
                                LET R5 := #CHRSET                       ;R5 IS OPERAND STACK FOR ROUTINE
 054770  012705  064036                                                  MOV     #CHRSET,R5
                                CALL PACK IN <#DSBUF1,DATA01,DATA00> ;SRC1
 054774  010546                                                          MOV     R5,-(SP)
 054776  016745  123560                                                  MOV     DATA00,-(R5)
 055002  016745  123556                                                  MOV     DATA01,-(R5)
 055006  012745  064440                                                  MOV     #DSBUF1,-(R5)
 055012  004767  002236                                                  JSR     PC,PACK
 055016  012605                                                          MOV     (SP)+,R5
                                CALL PACK IN <#DSBUF2,DATA03,DATA02> ;SRC2
 055020  010546                                                          MOV     R5,-(SP)
 055022  016745  123540                                                  MOV     DATA02,-(R5)
 055026  016745  123536                                                  MOV     DATA03,-(R5)
 055032  012745  064502                                                  MOV     #DSBUF2,-(R5)
 055036  004767  002212                                                  JSR     PC,PACK
 055042  012605                                                          MOV     (SP)+,R5
                                CALL PACK IN <#DS.EXP,DATA05,DATA04> ;DEST
 055044  010546                                                          MOV     R5,-(SP)
 055046  016745  123520                                                  MOV     DATA04,-(R5)
 055052  016745  123516                                                  MOV     DATA05,-(R5)
 055056  012745  064606                                                  MOV     #DS.EXP,-(R5)
 055062  004767  002166                                                  JSR     PC,PACK
 055066  012605                                                          MOV     (SP)+,R5
                                CALL CLRREG                     ;ENSURE GPR'S 0-5 ARE ZERO
 055070  004767  001344                                                  JSR     PC,CLRREG
```

```
                                                                                    DIVPI
 9538
 9539 055074                        INLINE <DIVPI>
      055074  076175                INLINE <.WORD TMULPI>                          .WORD TMULPI
 9540 055076
      055076  056064                INLINE <.WORD TMULPI+4>                        .WORD TMULPI+4
 9541 055100
      055100  056070                INLINE <.WORD TMULPI+10>                       .WORD TMULPI+10
 9542 055102
      055102  056074                LET CC := PSW                                  MOV     PSW,CC
 9543 055104
      055104  016767  122666  123406
 9544 055112                        LET CC := CC CLR.BY #177760                    BIC     #177760,CC
      055112  042767  177760  123400
 9545 055120                        IF CC NE #0 THEN                               TST     CC
      055120  005767  123374                                                       BEQ     $52431
      055124  001402
 9546 055126                            CALL ERROR                                 JSR     PC,ERROR
      055126  004767  002546
 9547 055132                        ENDIF
      055132                                                                $52431:
 9548
 9549                           ;+
 9550                           ;       ENSURE GPR'S STILL ZERO
 9551                           ;-
 9552
 9553 055132                        IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN       TST     R0
      055132  005700                                                               BNE     $52432
      055134  001004                                                               TST     R1
      055136  005701                                                               BNE     $52432
      055140  001002                                                               TST     R2
      055142  005702                                                               BEQ     $52433
      055144  001402
      055146                                                                $52432:
 9554 055146                            CALL ERROR                                 JSR     PC,ERROR
      055146  004767  002526
 9555 055152                        ENDIF
      055152                                                                $52433:
 9556 055152                        IF R3 NE #0 OR R4 NE #0 OR  R5 NE #0 THEN      TST     R3
      055152  005703                                                               BNE     $52434
      055154  001004                                                               TST     R4
      055156  005704                                                               BNE     $52434
      055160  001002                                                               TST     R5
      055162  005705                                                               BEQ     $52435
      055164  001402
      055166                                                                $52434:
 9557 055166                            CALL ERROR                                 JSR     PC,ERROR
      055166  004767  002506
 9558 055172                        ENDIF
      055172                                                                $52435:
 9559
 9560                           ;+
 9561                           ;       CHECK RESULTS
 9562                           ;-
 9563
 9564 055172                        LET R0 := #DS.DST                              MOV     #DS.DST,R0
      055172  012700  064544
 9565 055176                        LET R1 := #DS.EXP           ;EXPECTED DATA
```

DIVP

```
      055176  012701  064606                                                                  MOV     #DS.EXP,R1
 9566 055202                  LET R2 := TMULPI*10 CLR.BY #70000    ;GET STRING LENGTH
      055202  016702  000666                                                                  MOV     TMULPI*10,R2
      055206  042702  070000                                                                  BIC     #70000,R2
 9567 055212                  INLINE       <ASR    R2>             ;LENGTH/2 (PACKING FACTOR)
      055212  006202                                                                          ASR     R2
 9568 055214                  LET R2 := R2 + #DS.EXP               ;R2 POINTS TO MSD/SIGN
      055214  062702  064606                                                                  ADD     #DS.EXP,R2
 9569 055220                  LET (R2) :B= (R2) SET.BY #17 ;MAKE IT UNSIGNED
      055220  152712  000017                                                                  BISB    #17,(R2)
 9570 055224                  LET ERRFLG := #0
      055224  005067  123226                                                                  CLR     ERRFLG
 9571 055230                  WHILE R0 NE #DS.DST*37 AND ERRFLG EQ #0 DO
      055230                                                                          $52436:
      055230  020027  064603                                                                  CMP     R0,#DS.DST*37
      055234  001410                                                                          BEQ     $52437
      055236  005767  123214                                                                  TST     ERRFLG
      055242  001005                                                                          BNE     $52437
 9572 055244                      IFB (R0)+ NE (R1)+ THEN
      055244  122021                                                                          CMPB    (R0)+,(R1)+
      055246  001402                                                                          BEQ     $52440
 9573 055250                          CALL ERROR
      055250  004767  002424                                                                  JSR     PC,ERROR
 9574 055254                      ENDIF
      055254                                                                          $52440:
 9575 055254                  ENDDO
      055254  000765                                                                          BR      $52436
      055256                                                                          $52437:
 9576
 9577 055256                  INLINE <JMP CIS9END>
      055256  000167  000616                                                                      JMP CIS9END
 9578
```

```
                                .SBTTL  TEST CASES FOR CIS8 & CIS9

055262  000007          TAB89:  .WORD   7               ;# OF TEST CASES

055264  000017                  .WORD   15.             ;SRC1 LENGTH
055266  055462                  .WORD   MUL1            ;SRC1 PTR
055270  000021                  .WORD   17.             ;SRC2 LENGTH
055272  055504                  .WORD   MUL1A           ;SRC2 PTR
055274  000037                  .WORD   31.             ;DEST LENGTH
055276  055530                  .WORD   MUL1B           ;DEST PTR (PRODUCT)
055300  000000                  .WORD   0               ;CC.EXP (PRODUCT)
055302  055572                  .WORD   MUL1C           ;DEST PTR (QUOTIENT)
055304  000000                  .WORD   0               ;CC.EXP (QUOTIENT)

055306  000004                  .WORD   4
055310  055634                  .WORD   MUL2
055312  000007                  .WORD   7
055314  055642                  .WORD   MUL2A
055316  000013                  .WORD   11.
055320  055654                  .WORD   MUL2B
055322  000010                  .WORD   10
055324  055672                  .WORD   MUL2C
055326  000010                  .WORD   10

055330  000005                  .WORD   5
055332  055710                  .WORD   MUL3
055334  000002                  .WORD   2
055336  055720                  .WORD   MUL3A
055340  000006                  .WORD   6
055342  055724                  .WORD   MUL3B
055344  000000                  .WORD   0
055346  055734                  .WORD   MUL3C
055350  000004                  .WORD   4

055352  000005                  .WORD   5
055354  055744                  .WORD   MUL4
055356  000001                  .WORD   1
055360  055754                  .WORD   MUL4A
055362  000001                  .WORD   1
055364  055762                  .WORD   MUL4B
055366  000004                  .WORD   4
055370  055770                  .WORD   MUL4C
055372  000004                  .WORD   4

055374  000002                  .WORD   2
055376  055776                  .WORD   MUL5
055400  000005                  .WORD   5
055402  056002                  .WORD   MUL5A
055404  000000                  .WORD   0
055406  056012                  .WORD   MUL5B
055410  000006                  .WORD   6
055412  056016                  .WORD   MUL5C
055414  000006                  .WORD   6

055416  000005                  .WORD   5
055420  055744                  .WORD   MUL4
055422  000001                  .WORD   1
```

TEST CASES FOR CIS8 & CIS9

```
 9637 055424  055754                        .WORD   MUL4A
 9638 055426  000004                        .WORD   4
 9639 055430  055762                        .WORD   MUL4B
 9640 055432  000004                        .WORD   4
 9641 055434  055770                        .WORD   MUL4C
 9642 055436  000004                        .WORD   4
 9643
 9644 055440  000004                        .WORD   4
 9645 055442  056022                        .WORD   MUL7
 9646 055444  000005                        .WORD   5
 9647 055446  056030                        .WORD   MUL7A
 9648 055450  000010                        .WORD   8.
 9649 055452  056040                        .WORD   MUL7B
 9650 055454  000000                        .WORD   0
 9651 055456  056052                        .WORD   MUL7C
 9652 055460  000000                        .WORD   0
 9653
 9654 055462  000000                MUL1:   .WORD   0
 9655 055464     004     004     004        .BYTE   4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4
      055467     004     004     004
      055472     004     004     004
      055475     004     004     004
      055500     004     004     004
 9656                                       .EVEN
 9657
 9658 055504  000000                MUL1A:  .WORD   0
 9659 055506     001     007     004        .BYTE   1,7,4,3,2,1,6,9.,5,3,4,7,9.,4,1,6,2
      055511     003     002     001
      055514     006     011     005
      055517     003     004     007
      055522     011     004     001
      055525     006     002
 9660                                       .EVEN
 9661
 9662 055530  000000                MUL1B:  .WORD   0
 9663 055532     007     007     004        .BYTE   7,7,4,7,6,3,0,9.,0,4,3,5,2,9.,5,3
      055535     007     006     003
      055540     000     011     000
      055543     004     003     005
      055546     002     011     005
      055551     003
 9664 055552     001     004     001        .BYTE   1,4,1,2,5,7,9.,8.,4,5,3,5,9.,2,8.
      055555     002     005     007
      055560     011     010     004
      055563     005     003     005
      055566     011     002     010
 9665                                       .EVEN
 9666
 9667 055572  000000                MUL1C:  .WORD   0
 9668 055574     000     000     000        .BYTE   0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
      055577     000     000     000
      055602     000     000     000
      055605     000     000     000
      055610     000     000     000
      055613     000
 9669 055614     000     000     000        .BYTE   0,0,0,0,0,0,0,0,0,0,0,0,0,3,9.
      055617     000     000     000
```

TEST CASES FOR CIS8 & CIS9

```
        055622     000     000     000
        055625     000     000     000
        055630     000     003     011
 9670                                           .EVEN
 9671
 9672 055634  177777                    MUL2:   .WORD   -1
 9673 055636     002     007     004            .BYTE   2,7,4,1
        055641     001
 9674                                           .EVEN
 9675
 9676 055642  000000                    MUL2A:  .WORD   0
 9677 055644     010     004     006            .BYTE   8.,4,6,2,9.,7,3
        055647     002     011     007
        055652     003
 9678                                           .EVEN
 9679 055654  177777                    MUL2B:  .WORD   -1
 9680 055656     002     003     001            .BYTE   2,3,1,9.,7,0,0,8.,9.,9.,3
        055661     011     007     000
        055664     000     010     011
        055667     011     003
 9681                                           .EVEN
 9682
 9683 055672  177777                    MUL2C:  .WORD   -1
 9684 055674     000     000     000            .BYTE   0,0,0,0,0,0,0,3,0,8.,7
        055677     000     000     000
        055702     000     003     000
        055705     010     007
 9685                                           .EVEN
 9686
 9687 055710  177777                    MUL3:   .WORD   -1
 9688 055712     004     001     007            .BYTE   4,1,7,3,9.
        055715     003     011
 9689                                           .EVEN
 9690
 9691 055720  177777                    MUL3A:  .WORD   -1
 9692 055722     001     001                    .BYTE   1,1
 9693                                           .EVEN
 9694
 9695 055724  000000                    MUL3B:  .WORD   0
 9696 055726     004     005     011            .BYTE   4,5,9.,1,2,9.
        055731     001     002     011
 9697                                           .EVEN
 9698
 9699 055734  000000                    MUL3C:  .WORD   0
 9700 055736     000     000     000            .BYTE   0,0,0,0,0,0
        055741     000     000     000
 9701                                           .EVEN
 9702
 9703 055744  000000                    MUL4:   .WORD   0
 9704 055746     001     007     003            .BYTE   1,7,3,5,4
        055751     005     004
 9705                                           .EVEN
 9706
 9707 055754  000000                    MUL4A:  .WORD   0
 9708 055756     000     000     000            .BYTE   0,0,0,0
        055761     000
 9709                                           .EVEN
```

TEST CASES FOR CIS8 & CIS9

```
9710
9711 055762  000000                    MUL4B:  .WORD   0
9712 055764     000     000     000             .BYTE   0,0,0,0
     055767     000
9713                                            .EVEN
9714
9715 055770  000000                    MUL4C:  .WORD   0
9716 055772     000     000     000             .BYTE   0,0,0,0
     055775     000
9717                                            .EVEN
9718
9719 055776  000000                    MUL5:   .WORD   0
9720 056000     001     007                     .BYTE   1,7
9721                                            .EVEN
9722
9723 056002  000000                    MUL5A:  .WORD   0
9724 056004     005     002     000             .BYTE   5,2,0,0,0
     056007     000     000
9725                                            .EVEN
9726
9727 056012  000000                    MUL5B:  .WORD   0
9728 056014     000                             .BYTE   0
9729                                            .EVEN
9730
9731 056016  000000                    MUL5C:  .WORD   0
9732 056020     000                             .BYTE   0
9733                                            .EVEN
9734
9735                                    ;MUL6   USES THE MUL4 STRINGS
9736
9737 056022  000000                    MUL7:   .WORD   0
9738 056024     001     000     000             .BYTE   1,0,0,1
     056027     001
9739                                            .EVEN
9740 056030  000000                    MUL7A:  .WORD   0
9741 056032     001     000     000             .BYTE   1,0,0,0,0
     056035     000     000
9742                                            .EVEN
9743 056040  000000                    MUL7B:  .WORD   0
9744 056042     001     000     000             .BYTE   1,0,0,1,0,0,0,0
     056045     001     000     000
     056050     000     000
9745                                            .EVEN
9746 056052  000000                    MUL7C:  .WORD   0
9747 056054     000     000     000             .BYTE   0,0,0,0,0,0,0,9.
     056057     000     000     000
     056062     000     011
9748                                            .EVEN
9749
9750                                    ;+
9751                                    ;       INLINE TEST CASE FOR MULP & DIVP
9752                                    ;-
9753
9754 056064  070017                    TMULPI: .WORD   070017
9755 056066  064440                            .WORD   DSBUF1
9756 056070  070021                            .WORD   070021
9757 056072  064502                            .WORD   DSBUF2
```

```
TEST CASES FOR CIS8 & CIS9

 9758 056074  070037                          .WORD   070037
 9759 056076  064544                          .WORD   DS.DST
 9760
 9761
 9762 056100                          CIS9END:
 9763
 9764
 9765                                 ;+
 9766                                 ; BEFORE GOING TO $EOP DETERMINE IF TESTING WITH AND WITHOUT
 9767                                 ; MEMORY MANAGEMENT ENABLED HAS OCCURRED IF APPLICABLE.
 9768                                 ; IF MEMORY MANAGEMENT DISABLED THEN CLEAR TSTNM AND ITERATION
 9769                                 ; COUNT AND GO TEST WITH MEMORY MANAGEMENT ENABLED.
 9770                                 ;-
 9771
 9772 056100                             IF @#MMCYL EQ #0 AND #BIT07 NOTSETIN @SWR THEN
      056100  005737  000446                                                      TST     @#MMCYL
      056104  001012                                                              BNE     $52441
      056106  032777  000200  122312                                              BIT     #BIT07,@SWR
      056114  001006                                                              BNE     $52441
 9773 056116                                 LET $TSTNM := #0
      056116  005067  122260                                                      CLR     $TSTNM
 9774 056122                                 LET $TIMES := #0
      056122  005067  122256                                                      CLR     $TIMES
 9775 056126                                 INLINE <JMP     BEGIN>
      056126  000167  123506                                                      JMP     BEGIN
 9776 056132                             ENDIF
      056132                                                                  $52441:
 9777
 9778                                 ;.ENABL ABS
 9779                                 ;;*****************************************************************
 9780                                 .SBTTL  END OF PASS ROUTINE
 9781                                 ;;*****************************************************************
 9782                                 ;*INCREMENT THE PASS NUMBER ($PASS)
 9783                                 ;*TYPE "END PASS #XXXXX" (WHERE XXXXX IS A DECIMAL NUMBER)
 9784                                 ;*IF THERES A MONITOR GO TO IT
 9785                                 ;*IF THERE ISN'T JUMP TO BEGIN
 9786
 9787 056132                          $EOP:
 9788 056132  000004                             SCOPE
 9789 056134  005067  122242                     CLR     $TSTNM          ;;ZERO THE TEST NUMBER
 9790 056140  005067  122240                     CLR     $TIMES
 9791 056144  005267  122470                     INC     $PASS           ;;INCREMENT THE PASS NUMBER
 9792 056150  042767  100000  122462            BIC     #100000,$PASS   ;;DON'T ALLOW A NEG. NUMBER
 9793 056156  005367  000122                     DEC     $EOPCT          ;;LOOP?
 9794 056162  003027                            BGT     $DOAGN          ;;YES
 9795 056164  016767  000116  000112            MOV     $ENDCT,$EOPCT   ;;RESTORE COUNTER
 9796 056172  012767  056313  122350            MOV     #$ENDMG,MSGADR
 9797 056200  104401                            TYPE                    ;;TYPE "END PASS #"
 9798 056202  016746  122432                     MOV     $PASS,-(SP)     ;;SAVE $PASS FOR TYPEOUT
 9799 056206  104403                            TYPOS                   ;;GO TYPE--OCTAL ASCII, NO LEADING ZEROES
 9800 056210     006                            .BYTE   6
 9801 056211     000                            .BYTE   0
 9802 056212  016767  000072  122330            MOV     $ENULL,MSGADR
 9803 056220  104401                            TYPE                    ;;TYPE A NULL CHARACTER
 9804 056222  013700  000042          $GET42:   MOV     @#42,R0         ;;GET MONITOR ADDRESS
 9805 056226  001405                            BEQ     $DOAGN          ;;BRANCH IF NO MONITOR
 9806 056230  000005                            RESET                   ;;CLEAR THE WORLD
```

```
END OF PASS ROUTINE

 9807 056232  004710                  $ENDAD: JSR     PC,(R0)          ;;GO TO MONITOR
 9808 056234  000240                          NOP                      ;;SAVE ROOM
 9809 056236  000240                          NOP                      ;;FOR
 9810 056240  000240                          NOP                      ;;ACT11
 9811 056242                         $DOAGN:
 9812 056242  013737  000004  000476          MOV     @#4,@#TEMP0      ;;SAVE CONTENTS OF LOCATION 4
 9813 056250  012737  056266  000004          MOV     #1$,@#4          ;;SET UP VECTOR IN CASE OF TRAP
 9814 056256  012737  000001  164000          MOV     #1,@#164000      ;;NOTIFY MULTI-TESTER OF PASS COMPLETE
 9815 056264  000402                          BR      2$               ;;NO TRAP --> DO NOT NEED TO RESTORE STACK
 9816 056266  062706  000004          1$:     ADD     #4,SP            ;;RESET STACK BECAUSE OF TRAP
 9817 056272  013737  000476  000004  2$:     MOV     @#TEMP0,@#4      ;;RESTORE CONTENTS OF LOCATION 4
 9818 056300  000167  123334                  JMP     BEGIN            ;;RETURN
 9819 056304  000001                 $EOPCT: .WORD   1
 9820 056306  000001                 $ENDCT: .WORD   1
 9821 056310     377     377     000 $ENULL: .BYTE   -1,-1,0          ;;NULL CHARACTER STRING
 9822 056313     015     012     105 $ENDMG: .ASCIZ  <15><12>/END PASS #/
      056316     116     104     040
      056321     120     101     123
      056324     123     040     043
      056327     000
 9823
 9824
 9825                                 ;;*****************************************************************
 9826                                 ; TABLE OF SIGN/DIGIT COMBINATIONS FOR OVERPUNCH
 9827                                 ;;*****************************************************************
 9828 056330     173                 OVPNCH: .BYTE   173              ; +0
 9829 056331     101                         .BYTE   101              ; +1
 9830 056332     102                         .BYTE   102              ; +2
 9831 056333     103                         .BYTE   103              ; +3
 9832 056334     104                         .BYTE   104              ; +4
 9833 056335     105                         .BYTE   105              ; +5
 9834 056336     106                         .BYTE   106              ; +6
 9835 056337     107                         .BYTE   107              ; +7
 9836 056340     110                         .BYTE   110              ; +8
 9837 056341     111                         .BYTE   111              ; +9
 9838 056342     175                         .BYTE   175              ; -0
 9839 056343     112                         .BYTE   112              ; -1
 9840 056344     113                         .BYTE   113              ; -2
 9841 056345     114                         .BYTE   114              ; -3
 9842 056346     115                         .BYTE   115              ; -4
 9843 056347     116                         .BYTE   116              ; -5
 9844 056350     117                         .BYTE   117              ; -6
 9845 056351     120                         .BYTE   120              ; -7
 9846 056352     121                         .BYTE   121              ; -8
 9847 056353     122                         .BYTE   122              ; -9
 9848
 9849
 9850                                 ;;*****************************************************************
 9851                                 .SBTTL  UNEXPECTED TIME-OUT TRAP ROUTINE
 9852                                 ;;*****************************************************************
 9853                                 ;*ANY UNEXPECTED TIME-OUT TRAP WILL END UP HERE
 9854                                 ;*
 9855
 9856 056354  012767  062436  122166 TIMOUT: MOV     #TIMMSG,MSGADR           ;TYPE OUT ERROR MESSAGE
 9857 056362  104401                         TYPE
 9858 056364  032767  000040  121460         BIT     #BIT5,UFDMODE            ;ARE WE IN XXDP UFD          ;PA001
 9859 056372  001403                         BEQ     1$                       ;NO, SO BRANCH               ;PA001
```

```
UNEXPECTED TIME-OUT TRAP ROUTINE

 9860 056374  005000                        CLR     R0                      ;CLEAR JUNK ARGUMENT            ;PA001
 9861 056376  004767  001450                JSR     PC,ABORT                ;SYSMAC ABORT ROUTINE           ;PA001
 9862 056402  000000                1$:     HALT
 9863 056404  000002                        RTI
 9864
 9865
 9866                                ;;*****************************************************************
 9867                                .SBTTL  ILLEGAL/RESERVED INSTRUCTION TRAP ROUTINE
 9868                                ;;*****************************************************************
 9869                                ;*ANY ILLEGAL OR RESERVED INSTRUCTION TRAPS WILL END UP HERE.
 9870                                ;*
 9871
 9872 056406  012767  062504  122134 ILLRES: MOV     #ILLMSG,MSGADR          ;TYPE OUT ERROR MESSAGE
 9873 056414  104401                        TYPE
 9874 056416  032767  000040  121426        BIT     #BIT5,UFDMODE           ;ARE WE IN XXDP UFD             ;PA001
 9875 056424  001403                        BEQ     1$                      ;NO, SO BRANCH                  ;PA001
 9876 056426  005000                        CLR     R0                      ;CLEAR JUNK ARGUMENT            ;PA001
 9877 056430  004767  001416                JSR     PC,ABORT                ;SYSMAC ABORT ROUTINE           ;PA001
 9878 056434  000000                1$:     HALT
 9879 056436  000002                        RTI
 9880
 9881
 9882                                ;;*****************************************************************
 9883                                .SBTTL  CLEAR REGISTERS R0 - R5 ROUTINE
 9884                                ;;*****************************************************************
 9885                                ;*THIS ROUTINE CLEARS GENERAL PURPOSE REGISTERS R0 THROUGH R5.
 9886                                ;*
 9887
 9888 056440                        ROUTINE CLRREG
      056440                                                             CLRREG:
 9889 056440                           LET R0 := #0
      056440  005000                                                             CLR     R0
 9890 056442                           LET R1 := #0
      056442  005001                                                             CLR     R1
 9891 056444                           LET R2 := #0
      056444  005002                                                             CLR     R2
 9892 056446                           LET R3 := #0
      056446  005003                                                             CLR     R3
 9893 056450                           LET R4 := #0
      056450  005004                                                             CLR     R4
 9894 056452                           LET R5 := #0
      056452  005005                                                             CLR     R5
 9895 056454                        ENDRTN
      056454                                                             $52442:
      056454                                                             $52443:
      056454  000207                                                             RTS     PC
 9896
 9897
 9898                                ;;*****************************************************************
 9899                                .SBTTL  CLEAR BUFFER ROUTINE
 9900                                ;;*****************************************************************
 9901                                ;*THIS ROUTINE CLEARS A 100 BYTE BUFFER WHOSE STARTING ADDRESS
 9902                                ;*HAS BEEN PUT INTO 'BFADRS'.
 9903
 9904 056456                        ROUTINE  CLRBUF
      056456                                                             CLRBUF:
 9905 056456                           LET SAVR0 := R0
```

CLEAR BUFFER ROUTINE

```
                                                                                    MOV     R0,SAVR0
 9906 056462                        LET SAVR1 := R1
      056462  010167  122114                                                       MOV     R1,SAVR1
 9907 056466                        LET R0 := BFADRS
      056466  016700  122050                                                       MOV     BFADRS,R0
 9908 056472                        LET R1 := #40
      056472  012701  000040                                                       MOV     #40,R1
 9909 056476                        REPEAT
      056476                                                                $52446:
 9910 056476                            LET (R0)+ := #0
      056476  005020                                                               CLR     (R0)+
 9911 056500                            LET R1 := R1 - #1
      056500  005301                                                               DEC     R1
 9912 056502                        UNTIL R1 EQ #0
      056502  005701                                                               TST     R1
      056504  001374                                                               BNE     $52446
 9913 056506                        LET R0 := SAVR0
      056506  016700  122066                                                       MOV     SAVR0,R0
 9914 056512                        LET R1 := SAVR1
      056512  016701  122064                                                       MOV     SAVR1,R1
 9915 056516                    ENDRTN
      056516                                                                $52444:
      056516                                                                $52445:
      056516  000207                                                               RTS     PC
 9916
 9917                           ;;*****************************************************************
 9918                           .SBTTL  SET BUFFER ROUTINE
 9919                           ;;*****************************************************************
 9920                           ;*THIS ROUTINE SETS ALL BITS IN A 100 BYTE BUFFER WHOSE STARTING
 9921                           ;*ADDRESS HAS BEEN PUT INTO 'BFADRS'.
 9922                           ;*
 9923
 9924 056520                    ROUTINE  SETBUF
      056520                                                                SETBUF:
 9925 056520                        LET SAVR0 := R0
      056520  010067  122054                                                       MOV     R0,SAVR0
 9926 056524                        LET SAVR1 := R1
      056524  010167  122052                                                       MOV     R1,SAVR1
 9927 056530                        LET R0 := BFADRS
      056530  016700  122006                                                       MOV     BFADRS,R0
 9928 056534                        LET R1 := #40
      056534  012701  000040                                                       MOV     #40,R1
 9929 056540                        REPEAT
      056540                                                                $52451:
 9930 056540                            LET (R0)+ := #177777
      056540  012720  177777                                                       MOV     #177777,(R0)+
 9931 056544                            LET R1 := R1 - #1
      056544  005301                                                               DEC     R1
 9932 056546                        UNTIL R1 EQ #0
      056546  005701                                                               TST     R1
      056550  001373                                                               BNE     $52451
 9933 056552                        LET R0 := SAVR0
      056552  016700  122022                                                       MOV     SAVR0,R0
 9934 056556                        LET R1 := SAVR1
      056556  016701  122020                                                       MOV     SAVR1,R1
 9935 056562                    ENDRTN
      056562                                                                $52447:
```

SET BUFFER ROUTINE

```
        056562                                                          $52450:
        056562  000207                                                         RTS     PC
 9936
 9937
 9938                           ;;*****************************************************************
 9939                           .SBTTL  ONES AND ZEROES BUFFER ROUTINE
 9940                           ;;*****************************************************************
 9941                           ;*THIS ROUTINE SETS AN ALTERNATING ONES AND ZEROES PATTERN IN A 100 BYTE
 9942                           ;*BUFFER WHOSE STARTING ADDRESS HAS BEEN PUT INTO 'BFADRS'.
 9943                           ;*
 9944
 9945 056564                   ROUTINE  ALTBUF
        056564                                                          ALTBUF:
 9946 056564                      LET SAVR0 := R0
        056564  010067  122010                                                  MOV     R0,SAVR0
 9947 056570                      LET SAVR1 := R1
        056570  010167  122006                                                  MOV     R1,SAVR1
 9948 056574                      LET R0 := BFADRS
        056574  016700  121742                                                  MOV     BFADRS,R0
 9949 056600                      LET R1 := #40
        056600  012701  000040                                                  MOV     #40,R1
 9950 056604                      REPEAT
        056604                                                          $52454:
 9951 056604                         LET (R0)+ := #125252
        056604  012720  125252                                                  MOV     #125252,(R0)+
 9952 056610                         LET R1 := R1 - #1
        056610  005301                                                          DEC     R1
 9953 056612                      UNTIL R1 EQ #0
        056612  005701                                                          TST     R1
        056614  001373                                                          BNE     $52454
 9954 056616                      LET R0 := SAVR0
        056616  016700  121756                                                  MOV     SAVR0,R0
 9955 056622                      LET R1 := SAVR1
        056622  016701  121754                                                  MOV     SAVR1,R1
 9956 056626                   ENDRTN
        056626                                                          $52452:
        056626                                                          $52453:
        056626  000207                                                          RTS     PC
 9957
 9958                           ;;*****************************************************************
 9959                           .SBTTL  ONES AND ZEROES BUFFER ROUTINE FOR 2000 BYTE BUFFER
 9960                           ;;*****************************************************************
 9961                           ;*THIS ROUTINE SETS AN ALTERNATING ONES AND ZEROES PATTERN IN A 2000 BYTE
 9962                           ;*BUFFER WHOSE STARTING ADDRESS HAS BEEN PUT INTO 'BFADRS'.
 9963                           ;*
 9964
 9965 056630                   ROUTINE  ALTBU1
        056630                                                          ALTBU1:
 9966 056630                      LET SAVR0 := R0
        056630  010067  121744                                                  MOV     R0,SAVR0
 9967 056634                      LET SAVR1 := R1
        056634  010167  121742                                                  MOV     R1,SAVR1
 9968 056640                      LET R0 := BFADRS
        056640  016700  121676                                                  MOV     BFADRS,R0
 9969 056644                      LET R1 := #1000
        056644  012701  001000                                                  MOV     #1000,R1
 9970 056650                      REPEAT
```

ONES AND ZEROES BUFFER ROUTINE FOR 2000 BYTE BUFFER

```
      056650                                                    $52457:
 9971 056650                          LET (R0)+ := #125252                MOV     #125252,(R0)+
      056650  012720  125252
 9972 056654                          LET R1 := R1 - #1                   DEC     R1
      056654  005301
 9973 056656                       UNTIL R1 EQ #0                         TST     R1
      056656  005701                                                      BNE     $52457
      056660  001373
 9974 056662                       LET R0 := SAVR0                       MOV     SAVR0,R0
      056662  016700  121712
 9975 056666                       LET R1 := SAVR1                       MOV     SAVR1,R1
      056666  016701  121710
 9976 056672                    ENDRTN
      056672                                                    $52455:
      056672                                                    $52456:
      056672  000207                                                     RTS     PC
 9977
```

ONES AND ZEROES BUFFER ROUTINE FOR 2000 BYTE BUFFER

```
 9979                           ;;*****************************************************************
 9980                           .SBTTL  CLEAR DECIMAL STRING BUFFERS ROUTINE
 9981                           ;;*****************************************************************
 9982                           ;*THIS ROUTINE CLEARS ALL BITS IN THE FOUR 40-BYTE DECIMAL
 9983                           ;*STRING BUFFERS.
 9984                           ;*
 9985
 9986 056674                   ROUTINE CLRDSB                                          CLRDSB:
      056674
 9987 056674                      LET SAVR0 := R0                                              MOV     R0,SAVR0
      056674  010067  121700
 9988 056700                      LET SAVR1 := R1                                              MOV     R1,SAVR1
      056700  010167  121676
 9989 056704                      LET R0 := #DSBUF1-1                                          MOV     #DSBUF1-1,R0
      056704  012700  064437
 9990 056710                      LET R1 := #207                                               MOV     #207,R1
      056710  012701  000207
 9991 056714                      REPEAT                                                 $52462:
      056714
 9992 056714                         LET (R0)+ :B= #0                                          CLRB    (R0)+
      056714  105020
 9993 056716                         LET R1 := R1 - #1                                         DEC     R1
      056716  005301
 9994 056720                      UNTIL R1 EQ #0                                               TST     R1
      056720  005701                                                                          BNE     $52462
      056722  001374
 9995 056724                      LET R0 := SAVR0                                              MOV     SAVR0,R0
      056724  016700  121650
 9996 056730                      LET R1 := SAVR1                                              MOV     SAVR1,R1
      056730  016701  121646
 9997 056734                   ENDRTN                                                  $52460:
      056734                                                                          $52461:
      056734
      056734  000207                                                                      RTS     PC
 9998
 9999                           ;;*****************************************************************
10000                           .SBTTL  SAVE REGISTERS R0 - R5
10001                           ;;*****************************************************************
10002                           ;*THIS ROUTINE SAVES R0 THROUGH R5 IN LOCATIONS SAVR0 - SAVR5.
10003                           ;*
10004
10005 056736                   ROUTINE SAVREG                                          SAVREG:
      056736
10006 056736                      LET SAVR0 := R0                                              MOV     R0,SAVR0
      056736  010067  121636
10007 056742                      LET SAVR1 := R1                                              MOV     R1,SAVR1
      056742  010167  121634
10008 056746                      LET SAVR2 := R2                                              MOV     R2,SAVR2
      056746  010267  121632
10009 056752                      LET SAVR3 := R3                                              MOV     R3,SAVR3
      056752  010367  121630
10010 056756                      LET SAVR4 := R4                                              MOV     R4,SAVR4
      056756  010467  121626
10011 056762                      LET SAVR5 := R5                                              MOV     R5,SAVR5
      056762  010567  121624
10012 056766                   ENDRTN                                                  $52463:
      056766
```

```
SAVE REGISTERS R0 - R5

        056766
        056766  000207                                          $52464:
                                                                        RTS     PC
  10013
```

SAVE REGISTERS R0 - R5

```
10015                                   ;;*****************************************************************
10016                                   .SBTTL  RESTORE REGISTERS R0 - R5
10017                                   ;;*****************************************************************
10018                                   ;*THIS ROUTINE RESTORES R0 THROUGH R5 FROM LOCATIONS SAVR0 - SAVR5
10019                                   ;*
10020
10021 056770                            ROUTINE RESREG
      056770                                                                    RESREG:
10022 056770                               LET R0 := SAVR0
      056770  016700  121604                                                            MOV     SAVR0,R0
10023 056774                               LET R1 := SAVR1
      056774  016701  121602                                                            MOV     SAVR1,R1
10024 057000                               LET R2 := SAVR2
      057000  016702  121600                                                            MOV     SAVR2,R2
10025 057004                               LET R3 := SAVR3
      057004  016703  121576                                                            MOV     SAVR3,R3
10026 057010                               LET R4 := SAVR4
      057010  016704  121574                                                            MOV     SAVR4,R4
10027 057014                               LET R5 := SAVR5
      057014  016705  121572                                                            MOV     SAVR5,R5
10028 057020                            ENDRTN
      057020                                                                    $52465:
      057020                                                                    $52466:
      057020  000207                                                                    RTS     PC
10029
10030                                   ;;*****************************************************************
10031                                   .SBTTL  MEMORY MANAGEMENT SET UP ROUTINE
10032                                   ; ROUTINE WILL DETERMINE IF MEMORY MANAGEMENT SHOULD BE ENABLED OR DISABLED.
10033                                   ; IF MEMORY MANAGEMENT IS TO BE ENABLED THE APRS WILL BE MAPPED SUCH THAT THE
10034                                   ; VIRTUAL ADDRESS WILL BE MAPPED TO ITSELF AND MEMORY MANAGEMENT IS ENABLED.
10035                                   ;;*****************************************************************
10036
10037 057022                            ROUTINE MMSETUP
      057022                                                                    MMSETUP:
10038
10039                                   ;+
10040                                   ; COMPLIMENT THE MEMORY MANAGEMENT CYCLE FLAG
10041                                   ;   MMCYL = 0  -KTOFF
10042                                   ;   MMCYL = -1 -KTON
10043                                   ;-
10044
10045 057022                               LET @#MMCYL := COMP @#MMCYL
      057022  005137  000446                                                            COM     @#MMCYL
10046
10047                                   ;+
10048                                   ; DETERMINE WHETHER TO ONLY CYCLE WITH MEMORY MANAGEMENT ON OR TO CYCLE
10049                                   ; WITH MEMORY MANAGEMENT OFF.
10050                                   ;-
10051
10052 057026                               IF #BIT06 SETIN @SWR THEN
      057026  032777  000100  121372                                                    BIT     #BIT06,@SWR
      057034  001404                                                                    BEQ     $52471
10053 057036                                   LET @#MMCYL := #-1
      057036  012737  177777  000446                                                    MOV     #-1,@#MMCYL
10054 057044                               ELSE
      057044  000406                                                                    BR      $52472
      057046                                                                    $52471:
```

MEMORY MANAGEMENT SET UP ROUTINE

```
10055 057046                            IF #BIT07 SETIN @SWR THEN
      057046  032777  000200  121352                                          BIT     #BIT07,@SWR
      057054  001402                                                         BEQ     $52473
10056 057056                                LET @#MMCYL := #0
      057056  005037  000446                                                 CLR     @#MMCYL
10057 057062                            ENDIF
      057062                                                          $52473:
10058 057062                        ENDIF
      057062                                                          $52472:
10059
10060                               ;+
10061                               ; ONLY SET UP MEMORY MANAGEMENT REGISTERS IF 1ST PASS
10062                               ;-
10063
10064 057062                        IF @#$PASS EQ #0 THEN
      057062  005737  000640                                                 TST     @#$PASS
      057066  001057                                                         BNE     $52474
10065 057070                            LET @#KIPAR0 := #0
      057070  005037  172340                                                 CLR     @#KIPAR0
10066 057074                            LET @#KIPAR1 := #200
      057074  012737  000200  172342                                         MOV     #200,@#KIPAR1
10067 057102                            LET @#KIPAR2 := #400
      057102  012737  000400  172344                                         MOV     #400,@#KIPAR2
10068 057110                            LET @#KIPAR3 := #600
      057110  012737  000600  172346                                         MOV     #600,@#KIPAR3
10069 057116                            LET @#KIPAR4 := #1000
      057116  012737  001000  172350                                         MOV     #1000,@#KIPAR4
10070 057124                            LET @#KIPAR5 := #1200
      057124  012737  001200  172352                                         MOV     #1200,@#KIPAR5
10071 057132                            LET @#KIPAR6 := #1400
      057132  012737  001400  172354                                         MOV     #1400,@#KIPAR6
10072 057140                            LET @#KIPAR7 := #7600
      057140  012737  007600  172356                                         MOV     #7600,@#KIPAR7
10073
10074 057146                            LET @#KIPDR0 := #77406
      057146  012737  077406  172300                                         MOV     #77406,@#KIPDR0
10075 057154                            LET @#KIPDR1 :=  #77406
      057154  012737  077406  172302                                         MOV     #77406,@#KIPDR1
10076 057162                            LET @#KIPDR2 := #77406
      057162  012737  077406  172304                                         MOV     #77406,@#KIPDR2
10077 057170                            LET @#KIPDR3 := #77406
      057170  012737  077406  172306                                         MOV     #77406,@#KIPDR3
10078 057176                            LET @#KIPDR4 := #77406
      057176  012737  077406  172310                                         MOV     #77406,@#KIPDR4
10079 057204                            LET @#KIPDR5 := #77406
      057204  012737  077406  172312                                         MOV     #77406,@#KIPDR5
10080 057212                            LET @#KIPDR6 := #77406
      057212  012737  077406  172314                                         MOV     #77406,@#KIPDR6
10081 057220                            LET @#KIPDR7 := #77406
      057220  012737  077406  172316                                         MOV     #77406,@#KIPDR7
10082 057226                        ENDIF
      057226                                                          $52474:
10083
10084                               ;+
10085                               ; NOW TURN ON MEMORY MANAGEMENT IF IN MEMORY MANAGEMENT MEMORY MANAGEMENT CYCLE
10086                               ;-
10087
```

MEMORY MANAGEMENT SET UP ROUTINE

```
 10088 057226                            IF @#MMCYL NE #0 THEN
       057226  005737  000446                                                  TST     @#MMCYL
       057232  001404                                                          BEQ     $52475
 10089 057234                                LET @#SR1 := @#SR1 SET.BY #BIT0
       057234  052737  000001  177572                                          BIS     #BIT0,@#SR1
 10090                                   ;+
 10091                                   ; ELSE TURN OFF MEMORY MANAGEMENT
 10092                                   ;-
 10093 057242                              ELSE
       057242  000403                                                          BR      $52476
       057244                                                          $52475:
 10094 057244                                LET @#SR1 := @#SR1 CLR.BY #BIT0
       057244  042737  000001  177572                                          BIC     #BIT0,@#SR1
 10095 057252                              ENDIF
       057252                                                          $52476:
 10096 057252                            ENDRTN
       057252                                                          $52467:
       057252                                                          $52470:
       057252  000207                                                          RTS     PC
 10097
```

MEMORY MANAGEMENT SET UP ROUTINE

```
10099
10100                                   ;****************************************
10101                                   .SBTTL  PACK - PACK DECIMAL STRING ROUTINE
10102                                   ;****************************************
10103
10104                                   ;+
10105                                   ;  FUNCTIONAL DESCRIPTION
10106                                   ;
10107                                   ;       THE PACK ROUTINE ACCEPTS UNPACKED DATA FROM A TEST CASE
10108                                   ;       TABLE AND PACKS IT (TWO BCD CHARS/BYTE) INTO ONE OF THE
10109                                   ;       DECIMAL STRING BUFFERS.
10110                                   ;
10111                                   ;  INPUTS:
10112                                   ;       1. ADDRESS OF THE DESTINATION BUFFER. ** NOTE: ** THIS MUST
10113                                   ;          BE #DSBUF1, #DSBUF2, OR #DS.EXP.
10114                                   ;       2. POINTER TO SOURCE DATA IN TEST CASE TABLE. ** NOTE: **
10115                                   ;          THIS MUST BE DATA01, DATA03, OR DATA05.
10116                                   ;       3. POINTER TO DESTINATION STRING LENGTH. ** NOTE: ** THIS
10117                                   ;          MUST BE DATA00, DATA02, OR DATA04.
10118                                   ;
10119                                   ;  OUTPUTS:
10120                                   ;       NONE
10121                                   ;
10122                                   ;  PATHOLOGICAL CONNECTIONS:
10123                                   ;       NONE.
10124                                   ;
10125                                   ;  SUBORDINATE ROUTINES:
10126                                   ;       NONE.
10127                                   ;
10128                                   ;  FUNCTIONAL SIDE-EFFECTS:
10129                                   ;       NONE.
10130                                   ;
10131                                   ;  CALLING SEQUENCE:
10132                                   ;
10133                                   ;       CALL PACK IN <DSTADR,SRCADR,LENGTH>. FOR INPUT PARAMETERS
10134                                   ;       REFER TO ABOVE INPUTS:.
10135                                   ;
10136                                   ;  NOTE:
10137                                   ;       R3 IS USED AS A BYTE ASSEMBLY AREA.
10138                                   ;       R4 IS USED TO HOLD THE SIGN OF THE STRING (+ = #14,- = #15).
10139                                   ;-
10140
10141 057254                           ROUTINE PACK <DSTADR,SRCADR,LENGTH>
       057254                                                                   PACK:
10142 057254                             LET R0 := DSTADR(R5)
       057254  016500  000000                                                          MOV     DSTADR(R5),R0
10143 057260                             LET R1 := SRCADR(R5)
       057260  016501  000002                                                          MOV     SRCADR(R5),R1
10144 057264                             LET R2 := LENGTH(R5) CLR.BY #70000
       057264  016502  000004                                                          MOV     LENGTH(R5),R2
       057270  042702  070000                                                          BIC     #70000,R2
10145
10146                                    ;+
10147                                    ;       STORE SIGN OF STRING IN R4
10148                                    ;-
10149
10150 057274                             IF (R1)+ EQ #0 THEN            ;SIGN IS +
```

```
        057274  005721                                                                  TST     (R1)+
        057276  001003                                                                  BNE     $52501
  10151 057300                        LET R4 := #14
        057300  012704  000014                                                          MOV     #14,R4
  10152 057304                      ELSE                          ;SIGN IS -
        057304  000402                                                                  BR      $52502
        057306                                                                $52501:
  10153 057306                        LET R4 := #15
        057306  012704  000015                                                          MOV     #15,R4
  10154 057312                      ENDIF
        057312                                                                $52502:
  10155
  10156                             ;+
  10157                             ;     IF LENGTH IS ZERO, THEN SIGN STRING AND EXIT; ELSE IF
  10158                             ;     STRING LENGTH IS EVEN, STORE THE MSD (STRING LENGTH IS
  10159                             ;     NOW ODD).NOW STORE ALL BUT THE LAST DIGIT. BUILD LSD
  10160                             ;     & SIGN AND EXIT.
  10161                             ;-
  10162
  10163 057312                      IF R2 EQ #0 THEN              ;STRING LENGTH = 0
        057312  005702                                                                 TST     R2
        057314  001002                                                                 BNE     $52503
  10164 057316                        LET (R0) :B= R4             ;SIGN THE BUFFER AND EXIT
        057316  110410                                                                 MOVB    R4,(R0)
  10165 057320                      ELSE
        057320  000425                                                                 BR      $52504
        057322                                                                $52503:
  10166 057322                        IF #BIT00 NOTSETIN R2 THEN  ;LENGTH IS EVEN & > 0
        057322  032702  000001                                                         BIT     #BIT00,R2
        057326  001002                                                                 BNE     $52505
  10167 057330                          LET (R0)+ :B= (R1)+       ;MOVE THE EVEN NIBBLE
        057330  112120                                                                 MOVB    (R1)+,(R0)+
  10168 057332                          LET R2 := R2 - #1         ;DEC NIBBLE COUNTER
        057332  005302                                                                 DEC     R2
  10169 057334                        ENDIF
        057334                                                                $52505:
  10170 057334                        WHILE R2 NE #1 DO
        057334                                                                $52506:
        057334  020227  000001                                                         CMP     R2,#1
        057340  001410                                                                 BEQ     $52507
  10171 057342                          LET R3 :B= (R1)+          ;GET A BCD
        057342  112103                                                                 MOVB    (R1)+,R3
  10172 057344                          INLINE <ASH #4,   R3>     ;SHIFT IT TO THE HIGH NIBBLE
        057344  072327  000004                                                         ASH #4, R3
  10173 057350                          LET R3 :B= R3 SET.BY (R1)+        ;PACK IN LOW NIBBLE
        057350  152103                                                                 BISB    (R1)+,R3
  10174 057352                          LET (R0)+ :B= R3          ;STORE RESULT
        057352  110320                                                                 MOVB    R3,(R0)+
  10175 057354                          LET R2 := R2 - #2         ;DEC NIBBLE COUNTER
        057354  162702  000002                                                         SUB     #2,R2
  10176 057360                        ENDDO
        057360  000765                                                                 BR      $52506
        057362                                                                $52507:
  10177 057362                      LET R3 :B= (R1)               ;GET LAST BCD
        057362  111103                                                                 MOVB    (R1),R3
  10178 057364                      INLINE <ASH #4,     R3>     ;SHIFT IT TO THE HIGH NIBBLE
        057364  072327  000004                                                         ASH #4, R3
```

PACK - PACK DECIMAL STRING ROUTINE

```
10179 057370                            LET R3 :B= R3 SET.BY R4     ;ADD SIGN
      057370  150403                                                           BISB    R4,R3
10180 057372                            LET (R0) :B= R3             ;STORE LSD & SIGN
      057372  110310                                                           MOVB    R3,(R0)
10181 057374                          ENDIF
      057374                                                        $52504:
10182 057374                         ENDRTN
      057374                                                        $52477:
      057374                                                        $52500:
      057374  000207                                                           RTS     PC
10183
```

```
10185                                   ;;*****************************************************************
10186                                   .SBTTL  SCOPE HANDLER ROUTINE
10187                                   ;;*****************************************************************
10188                                   ;*THIS ROUTINE CONTROLS THE LOOPING OF SUBTESTS. IT WILL INCREMENT
10189                                   ;*AND LOAD THE TEST NUMBER($TSTNM) INTO THE DISPLAY REG.(DISPLAY<7:0>)
10190                                   ;*AND LOAD THE ERROR FLAG (ERRFLG) INTO DISPLAY<15:08>
10191                                   ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
10192                                   ;*SW14=1         LOOP ON TEST
10193                                   ;*SW11=1         INHIBIT ITERATIONS
10194                                   ;*
10195                                   ;*CALL
10196                                   ;*      SCOPE           ;;SCOPE=IOT
10197
10198 057376                           $SCOPE:
10199 057376  104406                            CKSWR                   ;;TEST  FOR CHANGE IN SOFT-SWR
10200 057400  032777  040000  121020            BIT     #BIT14,@SWR     ;;LOOP ON PRESENT TEST?
10201 057406  001034                            BNE     $OVER           ;;YES IF SW14=1
10202 057410  032777  004000  121010            BIT     #BIT11,@SWR     ;;INHIBIT ITERATIONS?
10203 057416  001011                            BNE     1$              ;;BRANCH IF YES
10204 057420  005767  121214                    TST     $PASS           ;;IF FIRST PASS OF PROGRAM
10205 057424  001406                            BEQ     1$              ;;      INHIBIT ITERATIONS
10206 057426  005267  120754                    INC     $ICNT           ;;INCREMENT ITERATION COUNT
10207 057432  026767  120746  120746            CMP     $TIMES,$ICNT    ;;CHECK THE NUMBER OF ITERATIONS MADE
10208 057440  002017                            BGE     $OVER           ;;BRANCH IF MORE ITERATION REQUIRED
10209 057442  012767  000001  120736  1$:       MOV     #1,$ICNT        ;;REINITIALIZE THE ITERATION COUNTER
10210 057450  016767  000040  120726            MOV     $MXCNT,$TIMES           ;;SET NUMBER OF ITERATIONS TO DO
10211 057456  005267  120720           $SVLAD:  INC     $TSTNM          ;;COUNT TEST NUMBERS
10212 057462  016767  120714  121146            MOV     $TSTNM,$TESTN   ;;SET TEST NUMBER IN APT MAILBOX
10213 057470  011667  120714                    MOV     (SP),$LPADR     ;;SAVE SCOPE LOOP ADDRESS
10214 057474  005067  120756                    CLR     ERRFLG          ;;ZERO THE ERROR FLAG
10215 057500  016777  120676  120722  $OVER:   MOV     $TSTNM,@DISPLAY ;;DISPLAY TEST NUMBER
10216 057506  016716  120676                    MOV     $LPADR,(SP)     ;;FUDGE RETURN ADDRESS
10217 057512  000002                            RTI                     ;;FIXES PS
10218 057514  000100           $MXCNT:  .WORD   100             ;;MAXIMUM NUMBER OF ITERATIONS
10219
10220
10221                                   ;;*****************************************************************
10222                                   .SBTTL  POWER DOWN AND UP ROUTINES
10223                                   ;;*****************************************************************
10224                                   ;POWER DOWN ROUTINE
10225 057516  012737  057666  000024  $PWRDN: MOV     #$ILLUP,@#PWRVEC ;;SET FOR FAST UP
10226 057524  012737  000340  000026            MOV     #340,@#PWRVEC+2 ;;PRIO:7
10227 057532  010046                            MOV     R0,-(SP)        ;;PUSH R0 ON STACK
10228 057534  010146                            MOV     R1,-(SP)        ;;PUSH R1 ON STACK
10229 057536  010246                            MOV     R2,-(SP)        ;;PUSH R2 ON STACK
10230 057540  010346                            MOV     R3,-(SP)        ;;PUSH R3 ON STACK
10231 057542  010446                            MOV     R4,-(SP)        ;;PUSH R4 ON STACK
10232 057544  010546                            MOV     R5,-(SP)        ;;PUSH R5 ON STACK
10233 057546  017746  120654                    MOV     @SWR,-(SP)      ;;PUSH @SWR ON STACK
10234 057552  010667  000114                    MOV     SP,$SAVR6       ;;SAVE SP
10235 057556  012737  057570  000024            MOV     #$PWRUP,@#PWRVEC ;;SET UP VECTOR
10236 057564  000000                            HALT
10237 057566  000776                            BR      .-2             ;;HANG UP
10238
10239                                   ;;*****************************************************************
10240                                   ;POWER UP ROUTINE
10241 057570  012737  057666  000024  $PWRUP: MOV     #$ILLUP,@#PWRVEC ;;SET FOR FAST DOWN
```

POWER DOWN AND UP ROUTINES

```
10242 057576  016706  000070                MOV     $SAVR6,SP         ;;GET SP
10243 057602  005067  000064                CLR     $SAVR6            ;;WAIT LOOP FOR THE TTY
10244 057606  005267  000060          1$:   INC     $SAVR6            ;;WAIT FOR THE INC
10245 057612  001375                        BNE     1$                ;;OF  WORD
10246 057614  012677  120606                MOV     (SP)+,@SWR        ;;POP STACK INTO @SWR
10247 057620  012605                        MOV     (SP)+,R5          ;;POP STACK INTO R5
10248 057622  012604                        MOV     (SP)+,R4          ;;POP STACK INTO R4
10249 057624  012603                        MOV     (SP)+,R3          ;;POP STACK INTO R3
10250 057626  012602                        MOV     (SP)+,R2          ;;POP STACK INTO R2
10251 057630  012601                        MOV     (SP)+,R1          ;;POP STACK INTO R1
10252 057632  012600                        MOV     (SP)+,R0          ;;POP STACK INTO R0
10253 057634  012737  057516  000024        MOV     #$PWRDN,@#PWRVEC  ;;SET UP THE POWER DOWN VECTOR
10254 057642  012737  000340  000026        MOV     #340,@#PWRVEC+2   ;;PRIO:7
10255 057650  012767  062600  120672        MOV     #PWRMSG,MSGADR
10256 057656  104401                        TYPE                      ;;REPORT THE POWER FAILURE
10257 057660  012716                        MOV     (PC)+,(SP)        ;;RESTART AT PHALT
10258 057662  057674          $PWRAD: .WORD   PHALT             ;;RESTART ADDRESS
10259 057664  000002                  RTI
10260 057666  000000          $ILLUP: HALT                      ;;THE POWER UP SEQUENCE WAS STARTED
10261 057670  000776                  BR      .-2               ;; BEFORE THE POWER DOWN WAS COMPLETE
10262 057672  000000          $SAVR6: 0                         ;;PUT THE SP HERE
10263
10264 057674                  PHALT:
10265 057674  000137  001640                JMP     @#BEGIN                 :RESTART THE TEST AT THE BEGINNING
10266
10267                         ;;*****************************************************************
10268                         .SBTTL  ERROR HANDLER ROUTINE
10269                         ;;*****************************************************************
10270                         ;*THIS ROUTINE WILL INCREMENT THE ERROR FLAG AND THE ERROR COUNT,
10271                         ;*SAVE THE ERROR ITEM NUMBER AND THE ADDRESS OF THE ERROR CALL
10272                         ;*AND GO TO $ERRTYP ON ERROR
10273                         ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
10274                         ;*SW15=1         HALT ON ERROR
10275                         ;*SW13=1         INHIBIT ERROR TYPEOUTS
10276
10277 057700             ERROR:
10278 057700  104406                CKSWR                           ;;TEST FOR CHANGE IN SOFT-SWR
10279 057702  005267  120550    7$:   INC     ERRFLG              ;;SET THE ERROR FLAG
10280 057706  001775                BEQ     7$                  ;;DON'T LET THE FLAG GO TO ZERO
10281 057710  016777  120466  120512    MOV     $TSTNM,@DISPLAY ;;DISPLAY TEST NUMBER AND ERROR FLAG
10282 057716  005267  120472    1$:   INC     $ERTTL              ;;COUNT THE NUMBER OF ERRORS
10283 057722  011667  120472          MOV     (SP),$ERRPC         ;;GET ADDRESS OF ERROR INSTRUCTION
10284 057726  162767  000004  120464    SUB     #4,$ERRPC
10285 057734  032777  020000  120464    BIT     #BIT13,@SWR         ::SKIP TYPEOUT IF SET
10286 057742  001006                BNE     20$                 ;;SKIP TYPEOUTS
10287 057744  004767  000242          JSR     PC,$ERRTYP          ;;GO TO USER ERROR ROUTINE
10288 057750  012767  000443  120572    MOV     #$CRLF,MSGADR
10289 057756  104401                TYPE
10290 057760                  20$:
10291 057760  122767  000001  120664    CMPB    #APTENV,$ENV        ;;RUNNING IN APT MODE
10292 057766  001007                BNE     2$                  ;;NO,SKIP APT ERROR REPORT
10293 057770  116767  120556  000004    MOVB    ERR,21$             ;;SET ITEM NUMBER AS ERROR NUMBER
10294 057776  004767  002110          JSR     PC,$ATY4            ;;REPORT FATAL ERROR TO APT
10295 060002     000          21$:   .BYTE   0
10296 060003     000                 .BYTE   0
10297 060004  000777          22$:   BR      22$                 ;;APT ERROR LOOP
10298 060006  005777  120414    2$:   TST     @SWR                ;;HALT ON ERROR
```

ERROR HANDLER ROUTINE

```
10299 060012  100011                    BPL     3$                ;;SKIP IF CONTINUE
10300 060014  032767  000040  120030    BIT     #BIT5,UFDMODE     ;;ARE WE IN UFD MONITOR?                ;PA001
10301 060022  001403                    BEQ     8$                ;;NO, SO BRANCH                         ;PA001
10302 060024  005000                    CLR     R0                ;;CLEAR A POSSIBLE JUNK ARGUMENT        ;PA001
10303 060026  004767  000020            JSR     PC,ABORT          ;;CALL THE XXDP ABORT ROUTINE           ;PA001
10304 060032  000000              8$:   HALT                      ;;HALT ON ERROR!
10305 060034  104406                    CKSWR                     ;;TEST FOR CHANGE IN SOFT-SWR
10306 060036                      3$:
10307 060036  022737  056232  000042    CMP     #$ENDAD,@#42      ;;ACT-11 AUTO-ACCEPT?
10308 060044  001001                    BNE     6$                ;;BRANCH IF NO
10309 060046  000000                    HALT                      ;;YES
10310 060050                      6$:
10311 060050  000207                    RTS     PC                ;;RETURN
10312
10313
10314                           ;*****************************************************************
10315                           ;*                                                               *
10316                           ;*      XXDP USER FRIENDLY ABORT ROUTINE.                        *
10317                           ;*                                                               *
10318                           ;*      THIS ROUTINE (CALLED FROM SYSMAC) WILL ABORT THE         *
10319                           ;*      DIAGNOSTIC IN CASE OF A FATAL ERROR INSTEAD OF HALT.     *
10320                           ;*      CONTROL WILL BE PASSED BACK TO XXDP.                     *
10321                           ;*                                                               *
10322                           ;*****************************************************************
10323                           .LIST   MEB
10324 060052                            $ABORT                                                   ;PA001
      060052  005767  121214    ABORT:  TST     UFDFLG            ;TEST FOR USER FRIENDLY MODE
      060056  001454                    BEQ     NOABRT            ;IF NOT UFD THEN CONTINUE NORMAL OPERATION
      060060  020027  000032            CMP     R0,#32            ;IS IT A ↑Z ?
      060064  001443                    BEQ     ABORTZ            ;JUST GO BACK TO CHAIN IF IT IS (NO ERROR)
      060066  020027  000003            CMP     R0,#3             ;IS IS A ↑C ?
      060072  001404                    BEQ     ABORTC            ;BR TO LOAD ↑C ON XXDP+ STACK (NO ERROR)
      060074  005767  121174            TST     UQUIET            ;TEST FOR USER-QUIET MODE
      060100  001443                    BEQ     NOABRT            ;IF FIELD-SERVICE MODE, CONTINUE NORMAL OPERATION
      060102  000422                    BR      ABORTE            ;SET DRSERR THEN LEAVE
      060104  016767  121156  117716    MOV     SAV30,30          ;RESTORE EMT LOCATION (30)
      060112  016767  121152  117712    MOV     SAV32,32          ;RESTORE EMT PRIORITY LOCATION (32)
      060120  104043                    EMT     +43               ;GET XXDP STACK LOC. INTO R0 FROM MONITOR
      060122  005720            1$:     TST     (R0)+             ;FIND END OF STACK
      060124  001376                    BNE     1$
      060126  112760  000057  177777    MOVB    #'/,-1(R0)        ;LOAD SLASH OVER ZERO
      060134  112720  000136            MOVB    #'↑,(R0)+         ;LOAD UPARROW
      060140  112720  000103            MOVB    #'C,(R0)+         ;LOAD C
      060144  105010                    CLRB    (R0)              ;MAKE NEW END TO STACK
      060146  000412                    BR      ABORTZ            ;NOW LEAVE
      060150  016767  121112  117652    MOV     SAV30,30          ;RESTORE EMT LOCATION (30)
      060156  016767  121106  117646    MOV     SAV32,32          ;RESTORE EMT PRIORITY LOCATION (32)
      060164  104042                    EMT     +42               ;GET DCA LOCATION INTO R0 FROM MONITOR
      060166  012760  177777  000042    MOV     #-1,42(R0)        ;SET A -1 INTO LOCATION DRSERR IN MONITOR
      060174  013700  000042    ABORTZ: MOV     @#42,R0           ;AND PUT THE MONITOR RETURN ADDRESS IN R0
      060200  005037  000042            CLR     @#42              ;CLEAR MONITOR RETURN FLAG
      060204  000167  176022            JMP     $ENDAD            ;RETURN TO MONITOR-DO NOT PUSH STACK HERE
      060210  000207            NOABRT: RTS     PC                ;IF NOTUFD RETURN TO MAINLINE
10325                           .NLIST  MEB
10326
10327
10328                           ;;*****************************************************************
```

ERROR MESSAGE TYPEOUT ROUTINE

```
10329                                   .SBTTL  ERROR MESSAGE TYPEOUT ROUTINE
10330                                   ;;*****************************************************************
10331                                   ;*THIS ROUTINE USES THE "ERROR TYPE NUMBER" (ERR) TO DETERMINE WHICH
10332                                   ;*ERROR IS TO BE REPORTED. IT THEN OBTAINS, FROM THE "ERROR TABLE" ($ERRTB),
10333                                   ;*AND REPORTS THE APPROPRIATE INFORMATION CONCERNING THE ERROR.
10334
10335 060212                           $ERRTYP:
10336 060212  012767  062750  120330          MOV     #ERRHDR,MSGADR          ;TYPE ERROR HEADER
10337 060220  104401                          TYPE
10338 060222  016767  120324  120244          MOV     ERR,ERRTMP              ;TYPE APPROPRIATE ERROR MESSAGE
10339 060230  005367  120240                  DEC     ERRTMP
10340 060234  006367  120234                  ASL     ERRTMP
10341 060240  062767  060372  120226          ADD     #ERRTBL,ERRTMP
10342 060246  017767  120222  120274          MOV     @ERRTMP,MSGADR
10343 060254  104401                          TYPE
10344 060256  012767  062630  120264          MOV     #SPACE1,MSGADR          ;TYPE ONE SPACE
10345 060264  104401                          TYPE
10346 060266  016746  120126                  MOV     $ERRPC,-(SP)            ;TYPE THE ERROR PC
10347 060272  104402                          TYPOC
10348 060274  016767  120246  120172          MOV     CTL,ERRTMP              ;TYPE FAILING CONTROL CHIP NUMBER
10349 060302  162767  000004  120164          SUB     #4,ERRTMP
10350 060310  006367  120160                  ASL     ERRTMP
10351 060314  062767  060400  120152          ADD     #CTLTBL,ERRTMP
10352 060322  017767  120146  120220          MOV     @ERRTMP,MSGADR
10353 060330  104401                          TYPE
10354 060332  012767  000443  120210          MOV     #$CRLF,MSGADR           ;TYPE CARRIAGE-RETURN/LINE-FEED
10355 060340  104401                          TYPE
10356 060342  005737  000446                  TST     @#MMCYL                 ;IF MEMORY MANAGEMENT NOT ENABLED
10357 060346  001004                          BNE     1$                      ;  THEN TYPE OUT
10358 060350  012767  063136  120172          MOV     #MMDIS,MSGADR           ;  MEMORY MANAGEMENT DISABLED MESSAGE
10359 060356  000403                          BR      2$
10360 060360  012767  063176  120162  1$:     MOV     #MMENA,MSGADR           ;  ELSE TYPE OUT MEMORY MANAGEMENT ENABLED
10361 060366  104401                  2$:     TYPE
10362 060370  000207                          RTS     PC
10363
10364 060372  063011                  ERRTBL: .WORD   ERR1                    ;ADDRESSES OF ERROR MESSAGES
10365 060374  063042                          .WORD   ERR2
10366 060376  063071                          .WORD   ERR3
10367
10368 060400  062632                  CTLTBL: .WORD   CTL4                    ;ADDRESSES OF CHIP NUMBER MESSAGES
10369 060402  062647                          .WORD   CTL5
10370 060404  062664                          .WORD   CTL6
10371 060406  062701                          .WORD   CTL7
10372 060410  062716                          .WORD   CTL8
10373 060412  062733                          .WORD   CTL9
10374
10375
10376                                   ;;*****************************************************************
10377                                   .SBTTL  TYPE ROUTINE
10378                                   ;;*****************************************************************
10379                                   ;*ROUTINE TO TYPE ASCIZ MESSAGE. MESSAGE MUST TERMINATE WITH A 0 BYTE.
10380                                   ;*THE ROUTINE WILL INSERT A NUMBER OF NULL CHARACTERS AFTER A LINE FEED.
10381                                   ;*NOTE1:        $NULL CONTAINS THE CHARACTER TO BE USED AS THE FILLER CHARACTER.
10382                                   ;*NOTE2:        $FILLS CONTAINS THE NUMBER OF FILLER CHARACTERS REQUIRED.
10383                                   ;*NOTE3:        $FILLC CONTAINS THE CHARACTER TO FILL AFTER.
10384                                   ;*
10385
```

TYPE ROUTINE

```
10386 060414  032767  000100  117430  $TYPE:  BIT     #BIT6,UFDMODE   ;;ARE WE IN QUIET XXDP?                ;PA001
10387 060422  001032                          BNE     3$              ;;YES SO EXIT                         ;PA001
10388 060424  105767  120005                  TSTB    $TPFLG          ;;IS THERE A TERMINAL?
10389 060430  100002                          BPL     1$              ;;BR IF YES
10390 060432  000000                          HALT                    ;;HALT HERE IF NO TERMINAL
10391 060434  000425                          BR      3$              ;;LEAVE
10392 060436  010046                  1$:     MOV     R0,-(SP)        ;;SAVE R0
10393 060440  016700  120104                  MOV     MSGADR,R0       ;;GET ADDRESS OF ASCIZ STRING
10394 060444  122767  000001  120200          CMPB    #APTENV,$ENV    ;;RUNNING IN APT MODE
10395 060452  001006                          BNE     62$             ;;NO,GO CHECK FOR APT CONSOLE
10396 060454  132767  000100  120171          BITB    #APTSPOOL,$ENVM ;;SPOOL MESSAGE TO APT
10397 060462  001402                          BEQ     62$             ;;NO,GO CHECK FOR CONSOLE
10398 060464  004767  001412                  JSR     PC,$ATY3        ;;SPOOL MESSAGE TO APT
10399 060470  132767  000040  120155  62$:     BITB    #APTCSUP,$ENVM  ;;APT CONSOLE SUPPRESSED
10400 060476  001003                          BNE     60$             ;;YES,SKIP TYPE OUT
10401 060500  112046                  2$:     MOVB    (R0)+,-(SP)     ;;PUSH CHARACTER TO BE TYPED ONTO STACK
10402 060502  001003                          BNE     4$              ;;BR IF IT ISN'T THE TERMINATOR
10403 060504  005726                          TST     (SP)+           ;;IF TERMINATOR POP IT OFF THE STACK
10404 060506  012600                  60$:    MOV     (SP)+,R0        ;;RESTORE R0
10405 060510  000002                  3$:     RTI                     ;;RETURN
10406 060512  122716  000011          4$:     CMPB    #HT,(SP)        ;;BRANCH IF <HT>
10407 060516  001432                          BEQ     8$
10408 060520  122716  000200                  CMPB    #CRLF,(SP)      ;;BRANCH IF NOT <CRLF>
10409 060524  001010                          BNE     5$
10410 060526  005726                          TST     (SP)+           ;;POP  <CR><LF> EQUIV
10411 060530  012767  000443  120012          MOV     #$CRLF,MSGADR
10412 060536  104401                          TYPE                    ;;TYPE A CR AND LF
10413 060540  105067  000130                  CLRB    $CHARCNT        ;;CLEAR CHARACTER COUNT
10414 060544  000755                          BR      2$              ;;GET NEXT CHARACTER
10415 060546  004767  000056          5$:     JSR     PC,$TYPEC       ;;GO TYPE THIS CHARACTER
10416 060552  126726  117656          6$:     CMPB    $FILLC,(SP)+    ;;IS IT TIME FOR FILLER CHARS.?
10417 060556  001350                          BNE     2$              ;;IF NO GO GET NEXT CHAR.
10418 060560  016746  117646                  MOV     $NULL,-(SP)     ;;GET # OF FILLER CHARS. NEEDED
10419                                                                 ;;AND THE NULL CHAR.
10420 060564  105366  000001          7$:     DECB    1(SP)           ;;DOES A NULL NEED TO BE TYPED?
10421 060570  002770                          BLT     6$              ;;BR IF NO--GO POP THE NULL OFF OF STACK
10422 060572  004767  000032                  JSR     PC,$TYPEC       ;;GO TYPE A NULL
10423 060576  105367  000072                  DECB    $CHARCNT        ;;DO NOT COUNT AS A COUNT
10424 060602  000770                          BR      7$              ;;LOOP
10425
10426                                 ;HORIZONTAL TAB PROCESSOR
10427
10428 060604  112716  000040          8$:     MOVB    #' ,(SP)        ;;REPLACE TAB WITH SPACE
10429 060610  004767  000014          9$:     JSR     PC,$TYPEC       ;;TYPE A SPACE
10430 060614  132767  000007  000052          BITB    #7,$CHARCNT     ;;BRANCH IF NOT AT
10431 060622  001372                          BNE     9$              ;;TAB STOP
10432 060624  005726                          TST     (SP)+           ;;POP SPACE OFF STACK
10433 060626  000724                          BR      2$              ;;GET NEXT CHARACTER
10434 060630  105737  177564          $TYPEC: TSTB    @#TPS           ;;WAIT UNTIL PRINTER IS READY
10435 060634  100375                          BPL     $TYPEC
10436 060636  116637  000002  177566          MOVB    2(SP),@#TPB     ;;LOAD CHAR TO BE TYPED INTO DATA REG.
10437 060644  122766  000015  000002          CMPB    #CR,2(SP)       ;;IS CHARACTER A CARRIAGE RETURN?
10438 060652  001003                          BNE     1$              ;;BRANCH IF NO
10439 060654  105067  000014                  CLRB    $CHARCNT        ;;YES--CLEAR CHARACTER COUNT
10440 060660  000406                          BR      $TYPEX          ;;EXIT
10441 060662  122766  000012  000002  1$:     CMPB    #LF,2(SP)       ;;IS CHARACTER A LINE FEED?
10442 060670  001402                          BEQ     $TYPEX          ;;BRANCH IF YES
```

TYPE ROUTINE

```
10443 060672  105227                          INCB    (PC)+           ;;COUNT THE CHARACTER
10444 060674  000000                $CHARCNT:.WORD  0               ;;CHARACTER COUNT STORAGE
10445 060676  000207                $TYPEX: RTS     PC
10446
10447                               ;;**************************************************************
10448                               .SBTTL  BINARY TO OCTAL (ASCII) AND TYPE
10449                               ;;**************************************************************
10450                               ;*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 6-DIGIT
10451                               ;*OCTAL (ASCII) NUMBER AND TYPE IT.
10452                               ;*$TYPOS---ENTER HERE TO SETUP SUPPRESS ZEROS AND NUMBER OF DIGITS TO TYPE
10453                               ;*CALL:
10454                               ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
10455                               ;*      TYPOS                   ;;CALL FOR TYPEOUT
10456                               ;*      .BYTE   N               ;;N=1 TO 6 FOR NUMBER OF DIGITS TO TYPE
10457                               ;*      .BYTE   M               ;;M=1 OR 0
10458                               ;*                                      ;;1=TYPE LEADING ZEROS
10459                               ;*                                      ;;0=SUPPRESS LEADING ZEROS
10460                               ;*
10461                               ;*$TYPON----ENTER HERE TO TYPE OUT WITH THE SAME PARAMETERS AS THE LAST
10462                               ;*$TYPOS OR $TYPOC
10463                               ;*CALL:
10464                               ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
10465                               ;*      TYPON                   ;;CALL FOR TYPEOUT
10466                               ;*
10467                               ;*$TYPOC---ENTER HERE FOR TYPEOUT OF A 16 BIT NUMBER
10468                               ;*CALL:
10469                               ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
10470                               ;*      TYPOC                   ;;CALL FOR TYPEOUT
10471
10472 060700  017646  000000        $TYPOS: MOV     @(SP),-(SP)     ;;PICKUP THE MODE
10473 060704  116667  000001  000227        MOVB    1(SP),$OFILL    ;;LOAD ZERO FILL SWITCH
10474 060712  112667  000225                MOVB    (SP)+,$OMODE+1  ;;NUMBER OF DIGITS TO TYPE
10475 060716  062716  000002                ADD     #2,(SP)         ;;ADJUST RETURN ADDRESS
10476 060722  000406                        BR      $TYPON
10477
10478 060724  112767  000001  000207  $TYPOC: MOVB    #1,$OFILL       ;;SET THE ZERO FILL SWITCH
10479 060732  112767  000006  000203        MOVB    #6,$OMODE+1     ;;SET FOR SIX(6) DIGITS
10480
10481 060740  032767  000100  117104  $TYPON: BIT     #BIT6,UFDMODE   ;;ARE WE IN QUIET XXDP?                 ;PA001
10482 060746  001401                        BEQ     9$              ;;NO, SO PRINT                          ;PA001
10483 060750  000002                        RTI                     ;;YES, SO JUST RETURN                   ;PA001
10484 060752  112767  000005  000160  9$:     MOVB    #5,$OCNT        ;;SET THE ITERATION COUNT
10485 060760  010346                        MOV     R3,-(SP)        ;;SAVE R3
10486 060762  010446                        MOV     R4,-(SP)        ;;SAVE R4
10487 060764  010546                        MOV     R5,-(SP)        ;;SAVE R5
10488 060766  116704  000151                MOVB    $OMODE+1,R4     ;;GET THE NUMBER OF DIGITS TO TYPE
10489 060772  005404                        NEG     R4
10490 060774  062704  000006                ADD     #6,R4           ;;SUBTRACT IT FOR MAX. ALLOWED
10491 061000  110467  000136                MOVB    R4,$OMODE       ;;SAVE IT FOR USE
10492 061004  116704  000131                MOVB    $OFILL,R4       ;;GET THE ZERO FILL SWITCH
10493 061010  016605  000012                MOV     12(SP),R5       ;;PICKUP THE INPUT NUMBER
10494 061014  005003                        CLR     R3              ;;CLEAR THE OUTPUT WORD
10495 061016  006105                1$:     ROL     R5              ;;ROTATE MSB INTO "C"
10496 061020  000404                        BR      3$              ;;GO DO MSB
10497 061022  006105                2$:     ROL     R5              ;;FORM THIS DIGIT
10498 061024  006105                        ROL     R5
10499 061026  006105                        ROL     R5
```

BINARY TO OCTAL (ASCII) AND TYPE

```
10500 061030  010503                          MOV     R5,R3
10501 061032  006103                  3$:     ROL     R3              ;;GET LSB OF THIS DIGIT
10502 061034  105367  000102                    DECB    $OMODE          ;;TYPE THIS DIGIT?
10503 061040  100020                          BPL     7$              ;;BR IF NO
10504 061042  042703  177770                    BIC     #177770,R3      ;;GET RID OF JUNK
10505 061046  001002                          BNE     4$              ;;TEST FOR 0
10506 061050  005704                          TST     R4              ;;SUPPRESS THIS 0?
10507 061052  001403                          BEQ     5$              ;;BR IF YES
10508 061054  005204                  4$:     INC     R4              ;;DON'T SUPPRESS ANYMORE 0'S
10509 061056  052703  000060                    BIS     #'0,R3          ;;MAKE THIS DIGIT ASCII
10510 061062  052703  000040          5$:     BIS     #' ,R3          ;;MAKE ASCII IF NOT ALREADY
10511 061066  110367  000044                    MOVB    R3,8$           ;;SAVE FOR TYPING
10512 061072  012767  061136  117450            MOV     #8$,MSGADR
10513 061100  104401                          TYPE                    ;;GO TYPE THIS DIGIT
10514 061102  105367  000032          7$:     DECB    $OCNT           ;;COUNT BY 1
10515 061106  003345                          BGT     2$              ;;BR IF MORE TO DO
10516 061110  002402                          BLT     6$              ;;BR IF DONE
10517 061112  005204                          INC     R4              ;;INSURE LAST DIGIT ISN'T A BLANK
10518 061114  000742                          BR      2$              ;;GO DO THE LAST DIGIT
10519 061116  012605                  6$:     MOV     (SP)+,R5        ;;RESTORE R5
10520 061120  012604                          MOV     (SP)+,R4        ;;RESTORE R4
10521 061122  012603                          MOV     (SP)+,R3        ;;RESTORE R3
10522 061124  016666  000002  000004          MOV     2(SP),4(SP)     ;;SET THE STACK FOR RETURNING
10523 061132  012616                          MOV     (SP)+,(SP)
10524 061134  000002                          RTI                     ;;RETURN
10525 061136     000                  8$:     .BYTE   0               ;;STORAGE FOR ASCII DIGIT
10526 061137     000                          .BYTE   0               ;;TERMINATOR FOR TYPE ROUTINE
10527 061140     000                  $OCNT:  .BYTE   0               ;;OCTAL DIGIT COUNTER
10528 061141     000                  $OFILL: .BYTE   0               ;;ZERO FILL SWITCH
10529 061142  000000                 $OMODE: .WORD   0               ;;NUMBER OF DIGITS TO TYPE
10530
10531
10532                                 ;;*******************************************************************
10533                                 .SBTTL  TTY INPUT ROUTINE
10534                                 ;;*******************************************************************
10535                                 .ENABL  LSB
10536
10537                                 ;;*******************************************************************
10538                                 ;*SOFTWARE SWITCH REGISTER CHANGE ROUTINE.
10539                                 ;*ROUTINE IS ENTERED FROM THE TRAP HANDLER, AND WILL
10540                                 ;*SERVICE THE TEST FOR CHANGE IN SOFTWARE SWITCH REGISTER TRAP CALL
10541                                 ;*WHEN OPERATING IN TTY FLAG MODE.
10542
10543 061144  022767  000176  117254 $CKSWR: CMP     #SWREG,SWR      ;;IS THE SOFT-SWR SELECTED?
10544 061152  001134                          BNE     15$             ;;BRANCH IF NO
10545 061154  105737  177560                    TSTB    @#TKS           ;;CHAR THERE?
10546 061160  100131                          BPL     15$             ;;IF NO, DON'T WAIT AROUND
10547 061162  113746  177562                    MOVB    @#TKB,-(SP)     ;;SAVE THE CHAR
10548 061166  042716  177600                    BIC     #^C177,(SP)     ;;STRIP-OFF THE ASCII
10549 061172  032767  000040  116652            BIT     #BIT5,UFDMODE   ;;ARE WE UNDER UFD?                     ;PA001
10550 061200  001422                          BEQ     21$             ;;NO, SO BRANCH                         ;PA001
10551 061202  111600                          MOVB    (SP),R0         ;;YES, GET THE CHARACTER JUST TYPED     ;PA001
10552 061204  122700  000003                    CMPB    #3,R0           ;;IS IT A CONTROL C?                    ;PA001
10553 061210  001005                          BNE     22$             ;;NO, SO BRANCH                         ;PA001
10554 061212  012767  062040  117330            MOV     #$CNTLC,MSGADR  ;;YES, GET CONTROL C TO ECHO            ;PA001
10555 061220  104401                          TYPE
10556 061222  000407                          BR      23$             ;;NOW DO THE ABORT                      ;PA001
```

TTY INPUT ROUTINE

```
10557 061224  122700  000132          22$:   CMPB   #132,R0          ;;IS IT A CONTROL Z?                    ;PA001
10558 061230  001006                         BNE    21$              ;;NO, SO BRANCH                         ;PA001
10559 061232  012767  062045  117310         MOV    #$CNTLZ,MSGADR   ;;YES, GET CONTROL Z TO ECHO            ;PA001
10560 061240  104401                         TYPE
10561 061242  004767  176604          23$:   JSR    PC,ABORT         ;;SEE IF WE NEED TO ABORT               ;PA001
10562 061246  022726  000007          21$:   CMP    #7,(SP)+         ;;IS IT A CONTROL G?
10563 061252  001074                         BNE    15$              ;;NO, RETURN TO USER
10564 061254  126727  117142  000001         CMPB   $AUTOB,#1        ;;ARE WE RUNNING IN AUTO-MODE?
10565 061262  001470                         BEQ    15$              ;;BRANCH IF YES
10566
10567 061264  012767  062033  117256         MOV    #$CNTLG,MSGADR
10568 061272  104401                         TYPE                    ;;ECHO THE CONTROL-G (↑G)
10569 061274  012767  062052  117246  $GTSWR: MOV    #$MSWR,MSGADR
10570 061302  104401                         TYPE                    ;;TYPE CURRENT CONTENTS
10571 061304  016746  116666                 MOV    SWREG,-(SP)      ;;SAVE SWREG FOR TYPEOUT
10572 061310  104402                         TYPOC                   ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
10573 061312  012767  062063  117230         MOV    #$MNEW,MSGADR
10574 061320  104401                         TYPE                    ;;PROMPT FOR NEW SWR
10575 061322  005046                 19$:    CLR    -(SP)            ;;CLEAR COUNTER
10576 061324  005046                         CLR    -(SP)            ;;THE NEW SWR
10577 061326  105737  177560          7$:    TSTB   @#TKS            ;;CHAR THERE?
10578 061332  100375                         BPL    7$               ;;IF NOT TRY AGAIN
10579
10580 061334  113746  177562                 MOVB   @#TKB,-(SP)      ;;PICK UP CHAR
10581 061340  042716  177600                 BIC    #↑C177,(SP)      ;;MAKE IT 7-BIT ASCII
10582
10583 061344  021627  000025          9$:    CMP    (SP),#25         ;;IS IT A CONTROL-U?
10584 061350  001007                         BNE    10$              ;;BRANCH IF NOT
10585 061352  012767  062026  117170         MOV    #$CNTLU,MSGADR
10586 061360  104401                         TYPE                    ;;YES, ECHO CONTROL-U (↑U)
10587 061362  062706  000006          20$:   ADD    #6,SP            ;;IGNORE PREVIOUS INPUT
10588 061366  000755                         BR     19$              ;;LET'S TRY IT AGAIN
10589
10590 061370  021627  000015          10$:   CMP    (SP),#15         ;;IS IT A <CR>?
10591 061374  001024                         BNE    16$              ;;BRANCH IF NO
10592 061376  005766  000004                 TST    4(SP)            ;;YES, IS IT THE FIRST CHAR?
10593 061402  001403                         BEQ    11$              ;;BRANCH IF YES
10594 061404  016677  000002  117014         MOV    2(SP),@SWR       ;;SAVE NEW SWR
10595 061412  062706  000006          11$:   ADD    #6,SP            ;;CLEAR UP STACK
10596 061416  012767  000443  117124  14$:   MOV    #$CRLF,MSGADR
10597 061424  104401                         TYPE                    ;;ECHO <CR> AND <LF>
10598 061426  126727  116771  000001         CMPB   $INTAG,#1        ;;RE-ENABLE TTY KBD INTERRUPTS?
10599 061434  001003                         BNE    15$              ;;BRANCH IF NOT
10600 061436  012737  000100  177560         MOV    #100,@#TKS       ;;RE-ENABLE TTY KBD INTERRUPTS
10601 061444  000002                 15$:    RTI                     ;;RETURN
10602 061446  004767  177156          16$:   JSR    PC,$TYPEC        ;;ECHO CHAR
10603 061452  021627  000060                 CMP    (SP),#60         ;;CHAR < 0?
10604 061456  002420                         BLT    18$              ;;BRANCH IF YES
10605 061460  021627  000067                 CMP    (SP),#67         ;;CHAR > 7?
10606 061464  003015                         BGT    18$              ;;BRANCH IF YES
10607 061466  042726  000060                 BIC    #60,(SP)+        ;;STRIP-OFF ASCII
10608 061472  005766  000002                 TST    2(SP)            ;;IS THIS THE FIRST CHAR
10609 061476  001403                         BEQ    17$              ;;BRANCH IF YES
10610 061500  006316                         ASL    (SP)             ;;NO, SHIFT PRESENT
10611 061502  006316                         ASL    (SP)             ;;    CHAR OVER TO MAKE
10612 061504  006316                         ASL    (SP)             ;;    ROOM FOR NEW ONE.
10613 061506  005266  000002          17$:   INC    2(SP)            ;;KEEP COUNT OF CHAR
```

TTY INPUT ROUTINE

```
10614 061512  056616  177776                BIS     -2(SP),(SP)     ;;SET IN NEW CHAR
10615 061516  000703                        BR      7$              ;;GET THE NEXT ONE
10616 061520  012767  000442  117022  18$:    MOV     #$QUES,MSGADR
10617 061526  104401                        TYPE                    ;;TYPE ?<CR><LF>
10618 061530  000714                        BR      20$             ;;SIMULATE CONTROL-U
10619                               .DSABL  LSB
10620
10621                               ;;*****************************************************************
10622                               ;*THIS ROUTINE WILL INPUT A SINGLE CHARACTER FROM THE TTY
10623                               ;*CALL:
10624                               ;*      RDCHR                   ;;INPUT A SINGLE CHARACTER FROM THE TTY
10625                               ;*      RETURN HERE             ;;CHARACTER IS ON THE STACK
10626                               ;*                              ;;WITH PARITY BIT STRIPPED OFF
10627                               ;
10628
10629 061532  032767  000100  116312  $RDCHR: BIT     #BIT6,UFDMODE   ;;ARE WE IN QUIET XXDP?         ;PA001
10630 061540  001047                        BNE     4$              ;;YES, SO JUST RETURN           ;PA001
10631 061542  011646                        MOV     (SP),-(SP)      ;;PUSH DOWN THE PC
10632 061544  016666  000004  000002        MOV     4(SP),2(SP)     ;;SAVE THE PS
10633 061552  105737  177560          1$:   TSTB    @#TKS           ;;WAIT FOR
10634 061556  100375                        BPL     1$              ;;A CHARACTER
10635 061560  113766  177562  000004        MOVB    @#TKB,4(SP)     ;;READ THE TTY
10636 061566  042766  177600  000004        BIC     #^C<177>,4(SP)  ;;GET RID OF JUNK IF ANY
10637 061574  026627  000004  000023        CMP     4(SP),#23       ;;IS IT A CONTROL-S?
10638 061602  001013                        BNE     3$              ;;BRANCH IF NO
10639 061604  105737  177560          2$:   TSTB    @#TKS           ;;WAIT FOR A CHARACTER
10640 061610  100375                        BPL     2$              ;;LOOP UNTIL ITS THERE
10641 061612  113746  177562                MOVB    @#TKB,-(SP)     ;;GET CHARACTER
10642 061616  042716  177600                BIC     #^C177,(SP)     ;;MAKE IT 7-BIT ASCII
10643 061622  022627  000021                CMP     (SP)+,#21       ;;IS IT A CONTROL-Q?
10644 061626  001366                        BNE     2$              ;;IF NOT DISCARD IT
10645 061630  000750                        BR      1$              ;;YES, RESUME
10646 061632  026627  000004  000140  3$:    CMP     4(SP),#140      ;;IS IT UPPER CASE?
10647 061640  002407                        BLT     4$              ;;BRANCH IF YES
10648 061642  026627  000004  000175        CMP     4(SP),#175      ;;IS IT A SPECIAL CHAR?
10649 061650  003003                        BGT     4$              ;;BRANCH IF YES
10650 061652  042766  000040  000004        BIC     #40,4(SP)       ;;MAKE IT UPPER CASE
10651 061660  000002                4$:     RTI                     ;;GO BACK TO USER
10652
10653                               ;;*****************************************************************
10654                               ;*THIS ROUTINE WILL INPUT A STRING FROM THE TTY
10655                               ;*CALL:
10656                               ;*      RDLIN                   ;;INPUT A STRING FROM THE TTY
10657                               ;*      RETURN HERE             ;;ADDRESS OF FIRST CHARACTER WILL BE ON THE STACK
10658                               ;*                              ;;TERMINATOR WILL BE A BYTE OF ALL 0'S
10659
10660 061662  032767  000100  116162  $RDLIN: BIT     #BIT6,UFDMODE   ;;ARE WE IN QUIET XXDP?         ;PA001
10661 061670  001401                        BEQ     11$             ;;NO, SO CONTINUE               ;PA001
10662 061672  000002                        RTI                     ;;YES, SO RETURN                ;PA001
10663
10664 061674  010346                11$:    MOV     R3,-(SP)        ;;SAVE R3
10665 061676  012703  062016          1$:   MOV     #$TTYIN,R3      ;;GET ADDRESS
10666 061702  022703  062026          2$:   CMP     #$TTYIN+8.,R3   ;;BUFFER FULL?
10667 061706  101405                        BLOS    4$              ;;BR IF YES
10668 061710  100407                        RDCHR                   ;;GO READ ONE CHARACTER FROM THE TTY
10669 061712  112613                        MOVB    (SP)+,(R3)      ;;GET CHARACTER
10670 061714  122713  000177          10$:  CMPB    #177,(R3)       ;;IS IT A RUBOUT
```

```
10671 061720 001005                          BNE     3$                ;;SKIP IF NOT
10672 061722 016767 116514 116620 4$:        MOV     $QUES,MSGADR
10673 061730 104401                          TYPE                      ;;TYPE A '?'
10674 061732 000761                          BR      1$                ;;CLEAR THE BUFFER AND LOOP
10675 061734 111367 000054          3$:      MOVB    (R3),9$           ;;ECHO THE CHARACTER
10676 061740 012767 062014 116602            MOV     #9$,MSGADR
10677 061746 104401                          TYPE
10678 061750 122723 000015                   CMPB    #15,(R3)+         ;;CHECK FOR RETURN
10679 061754 001352                          BNE     2$                ;;LOOP IF NOT RETURN
10680 061756 105063 177777                   CLRB    -1(R3)            ;;CLEAR RETURN (THE 15)
10681 061762 012767 000444 116560            MOV     #$LF,MSGADR
10682 061770 104401                          TYPE                      ;;TYPE A LINE FEED
10683 061772 012603                          MOV     (SP)+,R3          ;;RESTORE R3
10684 061774 011646                          MOV     (SP),-(SP)        ;;ADJUST THE STACK AND PUT ADDRESS OF THE
10685 061776 016666 000004 000002            MOV     4(SP),2(SP)       ;;FIRST ASCII CHARACTER ON IT
10686 062004 012766 062016 000004            MOV     #$TTYIN,4(SP)
10687 062012 000002                          RTI                       ;;RETURN
10688 062014    000                  9$:     .BYTE   0                 ;;STORAGE FOR ASCII CHAR. TO TYPE
10689 062015    000                          .BYTE   0                 ;;TERMINATOR
10690 062016                         $TTYIN: .BLKB   8.                ;;RESERVE 8 BYTES FOR TTY INPUT
10691 062026    136     125     015  $CNTLU: .ASCIZ  /↑U/<15><12>      ;;CONTROL "U"
      062031    012     000
10692 062033    136     107     015  $CNTLG: .ASCIZ  /↑G/<15><12>      ;;CONTROL "G"
      062036    012     000
10693 062040    136     103     015  $CNTLC: .ASCIZ  /↑C/<15><12>      ;;CONTROL "C"
      062043    012     000
10694 062045    136     132     015  $CNTLZ: .ASCIZ  /↑Z/<15><12>      ;;CONTROL "Z"
      062050    012     000
10695 062052    015     012     123  $MSWR:  .ASCIZ  <15><12>/SWR = /
      062055    127     122     040
      062060    075     040     000
10696 062063    040     040     116  $MNEW:  .ASCIZ  /  NEW = /
      062066    105     127     040
      062071    075     040     000
10697
10698                                ;;*****************************************************************
10699                                .SBTTL  APT COMMUNICATIONS ROUTINE
10700                                ;;*****************************************************************
10701 062074 112767 000001 000214 $ATY1:  MOVB    #1,$FFLG          ;;TO REPORT FATAL ERROR
10702 062102 112767 000001 000204 $ATY3:  MOVB    #1,$MFLG          ;;TO TYPE A MESSAGE
10703 062110 000403                       BR      $ATYC
10704 062112 112767 000001 000176 $ATY4:  MOVB    #1,$FFLG          ;;TO ONLY REPORT FATAL ERROR
10705 062120                      $ATYC:
10706 062120 010046                       MOV     R0,-(SP)          ;;PUSH R0 ON STACK
10707 062122 010146                       MOV     R1,-(SP)          ;;PUSH R1 ON STACK
10708 062124 105767 000164                TSTB    $MFLG             ;;SHOULD TYPE A MESSAGE?
10709 062130 001437                       BEQ     10$               ;;IF NOT:  BR
10710 062132 122767 000001 116512         CMPB    #APTENV,$ENV      ;;OPERATING UNDER APT?
10711 062140 001031                       BNE     3$                ;;IF NOT:  BR
10712 062142 132767 000100 116503         BITB    #APTSPOOL,$ENVM   ;;SHOULD SPOOL MESSAGES?
10713 062150 001425                       BEQ     3$                ;;IF NOT:  BR
10714 062152 017600 000004                MOV     @4(SP),R0         ;;GET MESSAGE ADDR.
10715 062156 062766 000002 000004         ADD     #2,4(SP)                  ;;BUMP RETURN ADDR.
10716 062164 005767 116442          1$:   TST     $MSGTYPE          ;;SEE IF DONE W/ LAST XMISSION?
10717 062170 001375                       BNE     1$                ;;IF NOT:  WAIT
10718 062172 010067 116450                MOV     R0,$MSGAD         ;;PUT ADDR IN MAILBOX
10719 062176 105720                 2$:   TSTB    (R0)+             ;;FIND END OF MESSAGE
```

APT COMMUNICATIONS ROUTINE

```
10720 062200  001376                          BNE     2$
10721 062202  166700  116440                  SUB     $MSGAD,R0       ;;SUB START OF MESSAGE
10722 062206  006200                          ASR     R0              ;;GET MESSAGE LNGTH IN WORDS
10723 062210  010067  116434                  MOV     R0,$MSGLGT      ;;PUT LENGTH IN MAILBOX
10724 062214  012767  000004  116410          MOV     #4,$MSGTYPE     ;;TELL APT TO TAKE MSG.
10725 062222  000402                          BR      10$
10726 062224  004767  176164          3$:     JSR     PC,$TYPE        ;;CALL TYPE MACRO
10727 062230  105767  000062          10$:    TSTB    $FFLG           ;;SHOULD REPORT FATAL ERROR?
10728 062234  001416                          BEQ     12$             ;;IF NOT:  BR
10729 062236  005767  116410                  TST     $ENV            ;;RUNNING UNDER APT?
10730 062242  001413                          BEQ     12$             ;;IF NOT:  BR
10731 062244  005767  116362          11$:    TST     $MSGTYPE        ;;FINISHED LAST MESSAGE?
10732 062250  001375                          BNE     11$             ;;IF NOT:  WAIT
10733 062252  017667  000004  116354          MOV     @4(SP),$FATAL   ;;GET ERROR #
10734 062260  062766  000002  000004          ADD     #2,4(SP)                ;;BUMP RETURN ADDR.
10735 062266  005267  116340                  INC     $MSGTYPE        ;;TELL APT TO TAKE ERROR
10736 062272  105067  000020          12$:    CLRB    $FFLG           ;;CLEAR FATAL FLAG
10737 062276  105067  000013                  CLRB    $LFLG           ;;CLEAR LOG FLAG
10738 062302  105067  000006                  CLRB    $MFLG           ;;CLEAR MESSAGE FLAG
10739 062306  012601                          MOV     (SP)+,R1        ;;POP STACK INTO R1
10740 062310  012600                          MOV     (SP)+,R0        ;;POP STACK INTO R0
10741 062312  000207                          RTS     PC              ;;RETURN
10742 062314     000          $MFLG:  .BYTE   0               ;;MESSG. FLAG
10743 062315     000          $LFLG:  .BYTE   0               ;;LOG FLAG
10744 062316     000          $FFLG:  .BYTE   0               ;;FATAL FLAG
10745                                 .EVEN
10746
10747          000200         APTSIZE=200
10748          000001         APTENV=001
10749          000100         APTSPOOL=100
10750          000040         APTCSUP=040
10751
10752                         ;;*****************************************************************
10753                         .SBTTL  TRAP DECODER
10754                         ;;*****************************************************************
10755                         ;*THIS ROUTINE WILL PICKUP THE LOWER BYTE OF THE "TRAP" INSTRUCTION
10756                         ;*AND USE IT TO INDEX THROUGH THE TRAP TABLE FOR THE STARTING ADDRESS
10757                         ;*OF THE DESIRED ROUTINE. THEN USING THE ADDRESS OBTAINED IT WILL
10758                         ;*GO TO THAT ROUTINE.
10759
10760 062320  010046          $TRAP:  MOV     R0,-(SP)        ;;SAVE R0
10761 062322  016600  000002          MOV     2(SP),R0        ;;GET TRAP ADDRESS
10762 062326  005740                  TST     -(R0)           ;;BACKUP BY 2
10763 062330  111000                  MOVB    (R0),R0         ;;GET RIGHT BYTE OF TRAP
10764 062332  006300                  ASL     R0              ;;POSITION FOR INDEXING
10765 062334  016000  062354          MOV     $TRPAD(R0),R0   ;;INDEX TO TABLE
10766 062340  000200                  RTS     R0              ;;GO TO ROUTINE
10767
10768
10769                         ;;THIS IS USE TO HANDLE THE "GETPRI" MACRO
10770
10771 062342  011646          $TRAP2: MOV     (SP),-(SP)      ;;MOVE THE PC DOWN
10772 062344  016666  000004  000002          MOV     4(SP),2(SP)     ;;MOVE THE PSW DOWN
10773 062352  000002                  RTI                     ;;RESTORE THE PSW
10774
10775                         .SBTTL  TRAP TABLE
10776
```

TRAP TABLE

```
 10777                                  ;*THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED
 10778                                  ;*BY THE "TRAP" INSTRUCTION.
 10779
 10780                                  ;       ROUTINE
 10781                                  ;       -------
 10782 062354  062342                   $TRPAD: .WORD   $TRAP2
 10783 062356  060414                           $TYPE   ;;CALL=TYPE     TRAP+1(104401)  TTY TYPEOUT ROUTINE
 10784 062360  060724                           $TYPOC  ;;CALL=TYPOC    TRAP+2(104402)  TYPE OCTAL NUMBER (WITH LEADING ZEROS)
 10785 062362  060700                           $TYPOS  ;;CALL=TYPOS    TRAP+3(104403)  TYPE OCTAL NUMBER (NO LEADING ZEROS)
 10786 062364  060740                           $TYPON  ;;CALL=TYPON    TRAP+4(104404)  TYPE OCTAL NUMBER (AS PER LAST CALL)
 10787 062366  061274                           $GTSWR  ;;CALL=GTSWR    TRAP+5(104405)  GET SOFT-SWR SETTING
 10788 062370  061144                           $CKSWR  ;;CALL=CKSWR    TRAP+6(104406)  TEST FOR CHANGE IN SOFT-SWR
 10789 062372  061532                           $RDCHR  ;;CALL=RDCHR    TRAP+7(104407)  TTY TYPEIN CHARACTER ROUTINE
 10790 062374  061662                           $RDLIN  ;;CALL=RDLIN    TRAP+10(104410) TTY TYPEIN STRING ROUTINE
 10791
 10792                                  ;;*****************************************************************
 10793                                  .SBTTL  TELETYPE MESSAGES
 10794                                  ;;*****************************************************************
 10795 062376     200     103     112   TITLE:  .ASCIZ  <200>/CJKDHB KEF11-B CIS DIAGNOSTIC/<200>
       062401     113     104     110
       062404     102     040     113
       062407     105     106     061
       062412     061     055     102
       062415     040     103     111
       062420     123     040     104
       062423     111     101     107
       062426     116     117     123
       062431     124     111     103
       062434     200     000
 10796 062436     200     101     116   TIMMSG: .ASCIZ  <200>/AN UNEXPECTED TIME-OUT TRAP OCCURRED/
       062441     040     125     116
       062444     105     130     120
       062447     105     103     124
       062452     105     104     040
       062455     124     111     115
       062460     105     055     117
       062463     125     124     040
       062466     124     122     101
       062471     120     040     117
       062474     103     103     125
       062477     122     122     105
       062502     104     000
 10797 062504     200     101     116   ILLMSG: .ASCIZ  <200>/AN UNEXPECTED ILLEGAL & RESERVED INSTRUCTION TRAP OCCURRED/
       062507     040     125     116
       062512     105     130     120
       062515     105     103     124
       062520     105     104     040
       062523     111     114     114
       062526     105     107     101
       062531     114     040     046
       062534     040     122     105
       062537     123     105     122
       062542     126     105     104
       062545     040     111     116
       062550     123     124     122
       062553     125     103     124
       062556     111     117     116
```

TELETYPE MESSAGES

```
                062561     040     124     122
                062564     101     120     040
                062567     117     103     103
                062572     125     122     122
                062575     105     104     000
 10798          062600     200     120     117  PWRMSG: .ASCIZ  <200>/POWER FAILURE OCCURRED/
                062603     127     105     122
                062606     040     106     101
                062611     111     114     125
                062614     122     105     040
                062617     117     103     103
                062622     125     122     122
                062625     105     104     000
 10799          062630     040     000          SPACE1: .ASCIZ  / /
 10800          062632     040     040     040  CTL4:   .ASCIZ  /      23-*04/
                062635     040     040     040
                062640     062     063     055
                062643     052     060     064
                062646     000
 10801          062647     040     040     040  CTL5:   .ASCIZ  /      23-*05/
                062652     040     040     040
                062655     062     063     055
                062660     052     060     065
                062663     000
 10802          062664     040     040     040  CTL6:   .ASCIZ  /      23-*06/
                062667     040     040     040
                062672     062     063     055
                062675     052     060     066
                062700     000
 10803          062701     040     040     040  CTL7:   .ASCIZ  /      23-*07/
                062704     040     040     040
                062707     062     063     055
                062712     052     060     067
                062715     000
 10804          062716     040     040     040  CTL8:   .ASCIZ  /      23-*08/
                062721     040     040     040
                062724     062     063     055
                062727     052     060     070
                062732     000
 10805          062733     040     040     040  CTL9:   .ASCIZ  /      23-*09/
                062736     040     040     040
                062741     062     063     055
                062744     052     060     071
                062747     000
 10806
 10807                                           ;;*****************************************************************
 10808                                           .SBTTL  TEST ERROR MESSAGES
 10809                                           ;;*****************************************************************
 10810          062750     200     200     105  ERRHDR: .ASCIZ  <200><200>/ERROR PC    CTL CHIP #       /
                062753     122     122     117
                062756     122     040     120
                062761     103     040     040
                062764     040     040     103
                062767     124     114     040
                062772     103     110     111
                062775     120     040     043
                063000     040     040     040
```

TEST ERROR MESSAGES

```
        063003      040     040     040
        063006      040     040     000
 10811  063011      103     111     123  ERR1:   .ASCIZ  /CIS INSTRUCTION FAILURE/<200>
        063014      040     111     116
        063017      123     124     122
        063022      125     103     124
        063025      111     117     116
        063030      040     106     101
        063033      111     114     125
        063036      122     105     200
        063041      000
 10812  063042      103     111     123  ERR2:   .ASCIZ  /CIS INTERRUPT FAILURE/<200>
        063045      040     111     116
        063050      124     105     122
        063053      122     125     120
        063056      124     040     106
        063061      101     111     114
        063064      125     122     105
        063067      200     000
 10813  063071      116     117     040  ERR3:   .ASCIZ  /NO TRAP ON ILLEGAL CIS CLASS OPCODE/<200>
        063074      124     122     101
        063077      120     040     117
        063102      116     040     111
        063105      114     114     105
        063110      107     101     114
        063113      040     103     111
        063116      123     040     103
        063121      114     101     123
        063124      123     040     117
        063127      120     103     117
        063132      104     105     200
        063135      000
 10814
 10815                                ;;*****************************************************************
 10816                                .SBTTL  MEMORY MANAGEMENT MESSAGES
 10817                                ;;*****************************************************************
 10818
 10819  063136      115     105     115  MMDIS:  .ASCIZ/MEMORY MANAGEMENT WAS DISABLED/<200>
        063141      117     122     131
        063144      040     115     101
        063147      116     101     107
        063152      105     115     105
        063155      116     124     040
        063160      127     101     123
        063163      040     104     111
        063166      123     101     102
        063171      114     105     104
        063174      200     000
 10820  063176      115     105     115  MMENA:  .ASCIZ/MEMORY MANAGEMENT WAS ENABLED/<200>
        063201      117     122     131
        063204      040     115     101
        063207      116     101     107
        063212      105     115     105
        063215      116     124     040
        063220      127     101     123
        063223      040     105     116
        063226      101     102     114
```

MEMORY MANAGEMENT MESSAGES

```
        063231      105     104     200
        063234      000
 10821
 10822                                  .EVEN
 10823                                  ;;*****************************************************************
 10824                                  .SBTTL  BUFFERS & TABLES
 10825                                  ;;*****************************************************************
 10826                                  ;
 10827 063236                           BUFF1:  .BLKB   100             ;CHARACTER STRING BUFFER 1
 10828 063336                           BUFF2:  .BLKB   100             ;CHARACTER STRING BUFFER 2
 10829 063436                           TRANS:  .BLKB   400             ;TRANSLATION TABLE
 10830 064036                           CHRSET: .BLKB   400             ;CHARACTER SET TABLE
 10831
 10832
 10833                                          .ODD
 10834 064437       000                         .BYTE   0                   ;RESERVED FOR SIGN FOR LEADING SEPARATE STRINGS
 10835 064440                           DSBUF1: .BLKB   40              ;DECIMAL STRING BUFFER 1
 10836
 10837                                          .ODD
 10838 064501       000                         .BYTE   0                   ;RESERVED FOR SIGN FOR LEADING SEPARATE STRINGS
 10839 064502                           DSBUF2: .BLKB   40              ;DECIMAL STRING BUFFER 2
 10840
 10841                                          .ODD
 10842 064543       000                         .BYTE   0                   ;RESERVED FOR SIGN FOR LEADING SEPARATE STRINGS
 10843 064544                           DS.DST: .BLKB   40              ;DESTINATION DECIMAL STRING BUFFER
 10844
 10845                                          .ODD
 10846 064605       000                         .BYTE   0                   ;RESERVED FOR SIGN FOR LEADING SEPARATE STRINGS
 10847 064606                           DS.EXP: .BLKB   40              ;DECIMAL STRING BUFFER OF EXPECTED RESULTS
 10848
 10849
 10850                                  ;;*****************************************************************
 10851                                  .SBTTL  TEST CASES FOR CMPN AND CMPP
 10852                                  ;;*****************************************************************
 10853                                  ;
 10854
 10855 064646  000007                   TCMP:   .WORD   7               ;# OF TEST CASES FOR CMPN & CMPP
 10856
 10857 064650  000000                           .WORD   0                       ;SRC1 LENGTH
 10858 064652  064740                           .WORD   CMP1                    ;SRC1 POINTER
 10859 064654  000000                           .WORD   0                       ;SRC2 LENGTH
 10860 064656  064744                           .WORD   CMP1A                   ;SRC2 POINTER
 10861
 10862 064660  000007                           .WORD   7               ;TEST CASE NUMBER 2
 10863 064662  064750                           .WORD   CMP2
 10864 064664  000000                           .WORD   0
 10865 064666  064762                           .WORD   CMP2A
 10866
 10867 064670  000000                           .WORD   0               ;TEST CASE NUMBER 3
 10868 064672  064766                           .WORD   CMP3
 10869 064674  000020                           .WORD   20
 10870 064676  064772                           .WORD   CMP3A
 10871
 10872 064700  000011                           .WORD   11              ;TEST CASE NUMBER 4
 10873 064702  065014                           .WORD   CMP4
 10874 064704  000011                           .WORD   11
 10875 064706  065030                           .WORD   CMP4A
```

TEST CASES FOR CMPN AND CMPP

```
10876
10877 064710  000006                        .WORD   6               ;TEST CASE NUMBER 5
10878 064712  065044                        .WORD   CMP5
10879 064714  000006                        .WORD   6
10880 064716  065054                        .WORD   CMP5A
10881
10882 064720  000001                        .WORD   1               ;TEST CASE NUMBER 6
10883 064722  065064                        .WORD   CMP6
10884 064724  000001                        .WORD   1
10885 064726  065070                        .WORD   CMP6A
10886
10887 064730  000006                        .WORD   6               ;TEST CASE NUMBER 7
10888 064732  065074                        .WORD   CMP7
10889 064734  000005                        .WORD   5
10890 064736  065104                        .WORD   CMP7A
10891
10892
10893 064740  000000                CMP1:   .WORD   0                               ;SRC1 SIGN
10894 064742     000                        .BYTE                                   ;DECIMAL STRING FOR SRC1
10895                                       .EVEN
10896 064744  000000                CMP1A:  .WORD   0                               ;SRC2 SIGN
10897 064746     000                        .BYTE                                   ;DECIMAL STRING FOR SRC2
10898                                       .EVEN
10899
10900 064750  000000                CMP2:   .WORD   0                               ;SRC1 SIGN
10901 064752     007     006     005        .BYTE   007,006,005,004,003,002,001     ;DECIMAL STRING FOR SRC1
      064755     004     003     002
      064760     001
10902                                       .EVEN
10903 064762  000000                CMP2A:  .WORD   0                               ;SRC2 SIGN
10904 064764     000                        .BYTE                                   ;DECIMAL STRING FOR SRC2
10905                                       .EVEN
10906
10907 064766  177777                CMP3:   .WORD   -1                              ;SRC1 SIGN
10908 064770     000                        .BYTE                                   ;DECIMAL STRING FOR SRC1
10909                                       .EVEN
10910 064772  177777                CMP3A:  .WORD   -1                              ;SRC2 SIGN
10911 064774     001     000     011        .BYTE   001,000,011,000,002,000,010,000,003,000,007,000,004,000,004,000
      064777     000     002     000
      065002     010     000     003
      065005     000     007     000
      065010     004     000     004
      065013     000
10912                                                                           ;DECIMAL STRING FOR SRC2
10913                                       .EVEN
10914
10915 065014  177777                CMP4:   .WORD   -1                                      ;SRC1 SIGN
10916 065016     001     002     003        .BYTE   001,002,003,000,005,006,007,010,011     ;DECIMAL STRING FOR SRC1
      065021     000     005     006
      065024     007     010     011
10917                                       .EVEN
10918 065030  177777                CMP4A:  .WORD   -1                                      ;SRC2 SIGN
10919 065032     001     002     003        .BYTE   001,002,003,004,005,006,007,010,011     ;DECIMAL STRING FOR SRC2
      065035     004     005     006
      065040     007     010     011
10920                                       .EVEN
10921
```

TEST CASES FOR CMPN AND CMPP

```
10922 065044  000000                  CMP5:   .WORD   0                               ;SRC1 SIGN
10923 065046     002     001     004          .BYTE   002,001,004,011,040,007         ;DECIMAL STRING FOR SRC1
      065051     011     040     007
10924                                         .EVEN
10925 065054  000000                  CMP5A:  .WORD   0                               ;SRC2 SIGN
10926 065056     002     001     004          .BYTE   002,001,004,011,040,007         ;DECIMAL STRING FOR SRC2
      065061     011     040     007
10927                                         .EVEN
10928
10929 065064  177777                  CMP6:   .WORD   -1                              ;SRC1 SIGN
10930 065066     004                          .BYTE   004                             ;DECIMAL STRING FOR SRC1
10931                                         .EVEN
10932 065070  000000                  CMP6A:  .WORD   0                               ;SRC2 SIGN
10933 065072     004                          .BYTE   004                             ;DECIMAL STRING FOR SRC2
10934                                         .EVEN
10935
10936 065074  000000                  CMP7:   .WORD   0                               ;SRC1 SIGN
10937 065076     002     003     004          .BYTE   002,003,004,005,006,007         ;DECIMAL STRING FOR SRC1
      065101     005     006     007
10938                                         .EVEN
10939 065104  000000                  CMP7A:  .WORD   0                               ;SRC2 SIGN
10940 065106     003     004     005          .BYTE   003,004,005,006,007             ;DECIMAL STRING FOR SRC2
      065111     006     007
10941                                         .EVEN
10942
10943                                 .SBTTL  TEST CASES FOR ASHN AND ASHP
10944                                 ;*****************************************************************
10945                                 ;
10946                                 ; TABLE OF TEST CASES FOR ASHN AND ASHP
10947                                 ;
10948                                 ;*****************************************************************
10949
10950 065114  000017                  TASH:   .WORD   15.                     ;# OF TEST CASES FOR ASHN & ASHP
10951
10952 065116  000005                          .WORD   5                               ;SRC LENGTH
10953 065120  000005                          .WORD   5                               ;DST LENGTH
10954 065122     000                          .BYTE   0                               ;SHIFT COUNT
10955 065123     000                          .BYTE   0                               ;ROUNDING DIGIT
10956
10957 065124  000000                          .WORD   0                       ;TEST CASE NUMBER 2
10958 065126  000017                          .WORD   17
10959 065130     000                          .BYTE   0
10960 065131     000                          .BYTE   0
10961
10962 065132  000032                          .WORD   32                      ;TEST CASE NUMBER 3
10963 065134  000032                          .WORD   32
10964 065136     000                          .BYTE   0
10965 065137     007                          .BYTE   7
10966
10967 065140  000025                          .WORD   25                      ;TEST CASE NUMBER 4
10968 065142  000025                          .WORD   25
10969 065144     377                          .BYTE   -1
10970 065145     005                          .BYTE   5
10971
10972 065146  000014                          .WORD   14                      ;TEST CASE NUMBER 5
10973 065150  000014                          .WORD   14
10974 065152     377                          .BYTE   -1
```

TEST CASES FOR ASHN AND ASHP

```
10975 065153  006                     .BYTE   6
10976
10977 065154  000006                  .WORD   6               ;TEST CASE NUMBER 6
10978 065156  000006                  .WORD   6
10979 065160  001                     .BYTE   1
10980 065161  006                     .BYTE   6
10981
10982 065162  000010                  .WORD   10              ;TEST CASE NUMBER 7
10983 065164  000011                  .WORD   11
10984 065166  001                     .BYTE   1
10985 065167  000                     .BYTE   0
10986
10987 065170  000005                  .WORD   5               ;TEST CASE NUMBER 8
10988 065172  000006                  .WORD   6
10989 065174  002                     .BYTE   2
10990 065175  000                     .BYTE   0
10991
10992 065176  000016                  .WORD   16              ;TEST CASE NUMBER 9
10993 065200  000004                  .WORD   4
10994 065202  363                     .BYTE   -15
10995 065203  000                     .BYTE   0
10996
10997 065204  000020                  .WORD   20              ;TEST CASE NUMBER 10
10998 065206  000010                  .WORD   10
10999 065210  360                     .BYTE   -20
11000 065211  000                     .BYTE   0
11001
11002 065212  000012                  .WORD   12              ;TEST CASE NUMBER 11
11003 065214  000007                  .WORD   7
11004 065216  006                     .BYTE   6
11005 065217  000                     .BYTE   0
11006
11007 065220  000037                  .WORD   37              ;TEST CASE NUMBER 12
11008 065222  000030                  .WORD   30
11009 065224  030                     .BYTE   30
11010 065225  000                     .BYTE   0
11011
11012 065226  000020                  .WORD   20              ;TEST CASE NUMBER 13
11013 065230  000010                  .WORD   10
11014 065232  373                     .BYTE   -5
11015 065233  000                     .BYTE   0
11016
11017 065234  000005                  .WORD   5               ;TEST CASE NUMBER 14
11018 065236  000005                  .WORD   5
11019 065240  373                     .BYTE   -5
11020 065241  006                     .BYTE   6
11021
11022 065242  000007                  .WORD   7               ;TEST CASE NUMBER 15
11023 065244  000007                  .WORD   7
11024 065246  366                     .BYTE   -12
11025 065247  006                     .BYTE   6
11026
11027
```

```
                                .SBTTL  ADDN/SUBN TEST DATA
                                ;;*****************************************************************
                                ;
                                ;       TABLES FOR ADDN AND SUBN
                                ;
                                ;;*****************************************************************

065250  000006         TADD:    .WORD   6               ;# OF TEST CASES

065252  000020                  .WORD   16.             ;SRC1 LENGTH
065254  065426                  .WORD   ADD1            ;SRC1 PTR
065256  000021                  .WORD   17.             ;SRC2 LENGTH
065260  065450                  .WORD   ADD1A           ;SRC2 PTR
065262  000024                  .WORD   20.             ;DEST LENGTH
065264  065474                  .WORD   ADD1B           ;DEST PTR (SUM)
065266  000000                  .WORD   0               ;CC.EXP (SUM)
065270  065522                  .WORD   ADD1C           ;DEST PTR (DIFF)
065272  000000                  .WORD   0               ;CC.EXP (DIFF)

065274  000012                  .WORD   10.             ;SRC1 >SRC2, SUM +,DIFF -
065276  065550                  .WORD   ADD2
065300  000012                  .WORD   10.
065302  065564                  .WORD   ADD2A
065304  000012                  .WORD   10.

065306  065600                  .WORD   ADD2B
065310  000000                  .WORD   0
065312  065614                  .WORD   ADD2C
065314  000010                  .WORD   10

065316  000013                  .WORD   11.             ;SRC1 +,SRC2 -, SUM & DIFF -
065320  065630                  .WORD   ADD3
065322  000013                  .WORD   11.
065324  065646                  .WORD   ADD3A
065326  000014                  .WORD   12.
065330  065664                  .WORD   ADD3B
065332  000010                  .WORD   10
065334  065702                  .WORD   ADD3C
065336  000010                  .WORD   10

065340  000020                  .WORD   16.             ;SRC1 -,SRC2 -,SUM -, DIFF +
065342  065720                  .WORD   ADD4
065344  000003                  .WORD   3
065346  065742                  .WORD   ADD4A
065350  000020                  .WORD   16.
065352  065750                  .WORD   ADD4B
065354  000010                  .WORD   10
065356  065772                  .WORD   ADD4C
065360  000000                  .WORD   0

065362  000016                  .WORD   14.             ;SRC1 = +,SRC2 0,DEST TRUNCATED WITH
                                                        ;SUM = +,DIFF = -
065364  066014                  .WORD   ADD5
065366  000001                  .WORD   1
065370  066034                  .WORD   ADD5A
065372  000011                  .WORD   9.
```

ADDN/SUBN TEST DATA

```
 11086 065374  066040                          .WORD   ADD5B
 11087 065376  000002                          .WORD   2
 11088 065400  066054                          .WORD   ADD5C
 11089 065402  000012                          .WORD   12
 11090
 11091 065404  000010                          .WORD   8.              ;DEST LENGTH = 0
 11092 065406  066070                          .WORD   ADD6
 11093 065410  000011                          .WORD   9.
 11094 065412  066102                          .WORD   ADD6A
 11095 065414  000000                          .WORD   0
 11096 065416  066116                          .WORD   ADD6B
 11097 065420  000006                          .WORD   6
 11098 065422  066116                          .WORD   ADD6B
 11099 065424  000006                          .WORD   6
 11100
 11101 065426  000000                  ADD1:   .WORD   0               ;SIGN1
 11102 065430     005     011     007          .BYTE   5,9.,7,4,3,8.,1,6,6,7,8.,1,2,9.,0,6
       065433     004     003     010
       065436     001     006     006
       065441     007     010     001
       065444     002     011     000
       065447     006
 11103                                                  ;SOURCE STRING 1
 11104
 11105                                  .EVEN
 11106 065450  000000                  ADD1A:  .WORD   0               ;SIGN2
 11107 065452     001     007     004          .BYTE   1,7,4,3,2,1,6,9.,5,3,4,7,9.,4,1,6,2
       065455     003     002     001
       065460     006     011     005
       065463     003     004     007
       065466     011     004     001
       065471     006     002
 11108                                                  ;SOURCE STRING 2
 11109
 11110                                  .EVEN
 11111 065474  000000                  ADD1B:  .WORD   0               ;SIGN3
 11112 065476     000     000     000          .BYTE   0,0,0,2,3,4,0,6,5,5,1,2,0,2
       065501     002     003     004
       065504     000     006     005
       065507     005     001     002
       065512     000     002
 11113 065514     006     000     007          .BYTE   6,0,7,0,6,8.
       065517     000     006     010
 11114                                                  ;DEST STRING
 11115                                  .EVEN
 11116
 11117 065522  000000                  ADD1C:  .WORD   0
 11118 065524     000     000     000          .BYTE   0,0,0,1,1,4,5,7,7,8.,7,8.,6
       065527     001     001     004
       065532     005     007     007
       065535     010     007     010
       065540     006
 11119 065541     006     011     010          .BYTE   6,9.,8.,1,2,5,6
       065544     001     002     005
       065547     006
 11120                                  .EVEN
 11121
```

ADDN/SUBN TEST DATA

```
11122 065550  000000                    ADD2:   .WORD   0
11123 065552     003      001     005           .BYTE   3,1,5,9.,2,6,5,3,8.,0
      065555     011      002     006
      065560     005      003     010
      065563     000
11124                                           .EVEN
11125
11126 065564  000000                    ADD2A:  .WORD   0
11127 065566     001      007     006           .BYTE   1,7,6,3,2,1,4,1,7,2
      065571     003      002     001
      065574     004      001     007
      065577     002
11128                                           .EVEN
11129
11130 065600  000000                    ADD2B:  .WORD   0
11131 065602     004      011     002           .BYTE   4,9.,2,2,4,7,9.,5,5,2
      065605     002      004     007
      065610     011      005     005
      065613     002
11132                                           .EVEN
11133
11134 065614  177777                    ADD2C:  .WORD   -1
11135 065616     001      003     011           .BYTE   1,3,9.,6,0,5,1,2,0,8.
      065621     006      000     005
      065624     001      002     000
      065627     010
11136                                           .EVEN
11137
11138 065630  000000                    ADD3:   .WORD   0
11139 065632     004      002     003           .BYTE   4,2,3,1,1,7,9.,6,7,9.,6
      065635     001      001     007
      065640     011      006     007
      065643     011      006
11140                                           .EVEN
11141
11142 065646  177777                    ADD3A:  .WORD   -1
11143 065650     007      011     003           .BYTE   7,9.,3,6,5,2,2,2,7,8.,1
      065653     006      005     002
      065656     002      002     007
      065661     010      001
11144                                           .EVEN
11145
11146 065664  177777                    ADD3B:  .WORD   -1
11147 065666     000      003     007           .BYTE   0,3,7,0,5,3,4,2,5,9.,8.,5
      065671     000      005     003
      065674     004      002     005
      065677     011      010     005
11148                                           .EVEN
11149
11150 065702  177777                    ADD3C:  .WORD   -1
11151 065704     001      002     001           .BYTE   1,2,1,6,7,7,0,1,9.,5,7,7
      065707     006      007     007
      065712     000      001     011
      065715     005      007     007
11152                                           .EVEN
11153
11154 065720  177777                    ADD4:   .WORD   -1
```

ADDN/SUBN TEST DATA

```
11155 065722     004     007     002          .BYTE   4,7,2,9.,3,2,7,7,1,6,1,7,9.,2,8.,4
      065725     011     003     002
      065730     007     007     001
      065733     006     001     007
      065736     011     002     010
      065741     004
11156                                          .EVEN
11157
11158 065742  177777                 ADD4A:  .WORD   -1
11159 065744     011     003     007          .BYTE   9.,3,7
11160                                          .EVEN
11161
11162 065750  177777                 ADD4B:  .WORD   -1
11163 065752     004     007     002          .BYTE   4,7,2,9.,3,2,7,7,1,6
      065755     011     003     002
      065760     007     007     001
      065763     006
11164 065764     001     010     000          .BYTE   1,8.,0,2,2,1
      065767     002     002     001
11165                                          .EVEN
11166
11167 065772  000000                 ADD4C:  .WORD   0
11168 065774     004     007     002          .BYTE   4,7,2,9.,3,2,7,7,1,6
      065777     011     003     002
      066002     007     007     001
      066005     006
11169 066006     001     007     010          .BYTE   1,7,8.,3,4,7
      066011     003     004     007
11170                                          .EVEN
11171
11172 066014  000000                 ADD5:   .WORD   0
11173 066016     004     000     001          .BYTE   4,0,1,7,7,7,9.,6,3,2,5,8.,8.,1
      066021     007     007     007
      066024     011     006     003
      066027     002     005     010
      066032     010     001
11174                                          .EVEN
11175
11176 066034  000000                 ADD5A:  .WORD   0
11177 066036     000                          .BYTE   0
11178                                          .EVEN
11179
11180 066040  000000                 ADD5B:  .WORD   0
11181 066042     007     011     006          .BYTE   7,9.,6,3,2,5,8.,8.,1
      066045     003     002     005
      066050     010     010     001
11182                                          .EVEN
11183
11184 066054  177777                 ADD5C:  .WORD   -1
11185 066056     007     011     006          .BYTE   7,9.,6,3,2,5,8.,8.,1
      066061     003     002     005
      066064     010     010     001
11186                                          .EVEN
11187
11188 066070  000000                 ADD6:   .WORD   0
11189 066072     001     007     011          .BYTE   1,7,9.,3,3,9.,8.,1
      066075     003     003     011
```

ADDN/SUBN TEST DATA

```
        066100     010     001
 11190                                  .EVEN
 11191
 11192 066102  000000                   ADD6A:  .WORD   0
 11193 066104     010     010     003            .BYTE   8.,8.,3,9.,2,7,6,4,9.
       066107     011     002     007
       066112     006     004     011
 11194                                  .EVEN
 11195
 11196 066116  000000                   ADD6B:  .WORD   0
 11197 066120     000                            .BYTE   0
 11198                                  .EVEN
 11199
 11200                                  ;+
 11201                                  ;       INLINE TEST CASE        (ADDNI & SUBNI)
 11202                                  ;-
 11203
 11204 066122  010020                   TADDI:  .WORD   010020          ;SRC1 DESCRIPTOR
 11205 066124  064440                           .WORD   DSBUF1
 11206 066126  010021                           .WORD   010021          ;SRC2 DESCRIPTOR
 11207 066130  064502                           .WORD   DSBUF2
 11208 066132  010024                           .WORD   010024          ;DEST DESCRIPTOR
 11209 066134  064544                           .WORD   DS.DST
 11210
 11211                                  ;+
 11212                                  ;       INLINE TEST CASE        (ADDPI & SUBPI)
 11213                                  ;-
 11214
 11215 066136  070020                   TADDPI: .WORD   070020
 11216 066140  064440                           .WORD   DSBUF1
 11217 066142  070021                           .WORD   070021
 11218 066144  064502                           .WORD   DSBUF2
 11219 066146  070024                           .WORD   070024
 11220 066150  064544                           .WORD   DS.DST
 11221
 11222 066152                           INLINE  <SPACE: .BLKW   2000>
       066152                                                           SPACE:  .BLKW   2000
 11223 072152                           INLINE  <SPACEY: .BLKW  2000>
       072152                                                           SPACEY: .BLKW   2000
 11224                                  .EVEN
 11225
 11226         001200                           .END    START
```

SYMBOL TABLE

```
ABORT   060052
ABORTC  060104
ABORTE  060150
ABORTZ  060174
ADDN  = 076050
ADDNI = 076150
ADDP  = 076070
ADDPI = 076170
ADD1    065426
ADD1A   065450
ADD1B   065474
ADD1C   065522
ADD2    065550
ADD2A   065564
ADD2B   065600
ADD2C   065614
ADD3    065630
ADD3A   065646
ADD3B   065664
ADD3C   065702
ADD4    065720
ADD4A   065742
ADD4B   065750
ADD4C   065772
ADD5    066014
ADD5A   066034
ADD5B   066040
ADD5C   066054
ADD6    066070
ADD6A   066102
ADD6B   066116
ADR.A   002332
ALPHA   002340
ALTBUF  056564
ALTBU1  056630
APTCSU= 000040
APTENV= 000001
APTSIZ= 000200
APTSPO= 000100
ASHN  = 076056
ASHNI = 076156
ASHP  = 076076
ASHPI = 076176
ASSEMB= 000010
ASWREG= 000000
BEGIN   001640
BETA    002344
BFADRS  000542
BF.EXP  000540
BIT0  = 000001
BIT00 = 000001
BIT01 = 000002
BIT02 = 000004
BIT03 = 000010
BIT04 = 000020
BIT05 = 000040
BIT06 = 000100
BIT07 = 000200
BIT08 = 000400
BIT09 = 001000
BIT1  = 000002
BIT10 = 002000
BIT11 = 004000
BIT12 = 010000
BIT13 = 020000
BIT14 = 040000
BIT15 = 100000
BIT2  = 000004
BIT3  = 000010
BIT4  = 000020
BIT5  = 000040
BIT6  = 000100
BIT7  = 000200
BIT8  = 000400
BIT9  = 001000
BPTVEC= 000014
BUFF1   063236
BUFF2   063336
BUFLEN= 000100
CBIT  = 000001
CC      000520
CC.EXP  000522
CHRSET  064036
CIS4EN  016622
CIS5EN  032622
CIS6EN  045366
CIS7EN  052766
CIS8EN  054112
CIS9EN  056100
CKSWR = 104406
CLRBUF  056456
CLRDSB  056674
CLRREG  056440
CMPC  = 076044
CMPCI = 076144
CMPN  = 076052
CMPNI = 076152
CMPNI1  042330
CMPP  = 076072
CMPPI = 076172
CMPPI1  051162
CMP1    064740
CMP1A   064744
CMP2    064750
CMP2A   064762
CMP3    064766
CMP3A   064772
CMP4    065014
CMP4A   065030
CMP5    065044
CMP5A   065054
CMP6    065064
CMP6A   065070
CMP7    065074
CMP7A   065104
COUNT   000470
CR    = 000015
CRLF  = 000200
CTL     000546
CTLTBL  060400
CTL4    062632
CTL5    062647
CTL6    062664
CTL7    062701
CTL8    062716
CTL9    062733
CVTIL   032210
CVTLI   031304
CVTLN = 076057
CVTLNI= 076157
CVTLP = 076077
CVTLPI= 076177
CVTNAA  025205
CVTNBB  025210
CVTNCC  025213
CVTNDD  025216
CVTNEE  025221
CVTNFF  025224
CVTNGG  025227
CVTNHH  025232
CVTNII  025235
CVTNJJ  025240
CVTNL = 076053
CVTNLA  025057
CVTNLB  025072
CVTNLC  025075
CVTNLD  025100
CVTNLE  025103
CVTNLF  025106
CVTNLG  025111
CVTNLH  025114
CVTNLI= 076153
CVTNLJ  025122
CVTNLK  025125
CVTNLL  025130
CVTNLM  025133
CVTNLN  025136
CVTNLO  025141
CVTNLP  025144
CVTNLQ  025147
CVTNLR  025152
CVTNLS  025155
CVTNLT  025160
CVTNLU  025163
CVTNLV  025166
CVTNLW  025171
CVTNLX  025174
CVTNLY  025177
CVTNLZ  025202
CVTNL$  025117
CVTNL1  025003
CVTNL2  025005
CVTNL3  025007
CVTNL4  025015
CVTNL5  025022
CVTNL6  025032
CVTNL7  025035
CVTNL8  025043
CVTNL9  025051
CVTNP = 076055
CVTNPI= 076155
CVTNP1  031140
CVTNP2  031144
CVTNP3  031150
CVTNP4  031154
CVTNP5  031202
CVTNP6  031214
CVTNP7  031230
CVTNP8  031252
CVTNP9  031262
CVTPL = 076073
CVTPLI= 076173
CVTPL1  023164
CVTPL2  023165
CVTPL3  023170
CVTPL4  023173
CVTPL5  023201
CVTPL6  023205
CVTPN = 076054
CVTPNI= 076154
CVTPN1  027272
CVTPN2  027273
CVTPN3  027274
CVTPN4  027277
CVTPN5  027312
CVTPN6  027316
CVTPN7  027322
CVTPN8  027325
CVTPN9  027333
DATA00  000562
DATA01  000564
DATA02  000566
DATA03  000570
DATA04  000572
DATA05  000574
DATA06  000576
DATTYP  000464
DELAY   000472
DESCR0  000616
DESCR1  000620
DESCR2  000622
DESCR3  000624
DESCR4  000626
DESCR5  000630
DIAGMC= 000000
DISPLA  000430
DISPRE  000174
DIVP  = 076075
DIVPI = 076175
DOIT    037516
DOIT1   034724
DSBUF1  064440
DSBUF2  064502
DSLEN1  000514
DSLEN2  000516
DSTADR= 000000
DSTMP0  000554
DSTMP1  000555
DSTMP2  000556
DSTMP3  000557
DSTMP4  000560
DSTMP5  000561
DS.DST  064544
DS.EXP  064606
EMTSAV  001200
EMTVEC= 000030
END1    035174
END2    035170
END21   037766
END22   037762
END23   037756
END3    035164
ERR     000552
ERRFLG  000456
ERRHDR  062750
ERROR   057700
ERRTBL  060372
ERRTMP  000474
ERRVEC= 000004
ERR1    063011
ERR2    063042
ERR3    063071
GAMMA   002350
GTSWR = 104405
HT    = 000011
ILLMSG  062504
ILLOP = 076033
ILLRES  056406
ILLTST  016474
INC1    033332
INC2    033716
INC21   036120
INC22   036504
INC23   037074
INC3    034302
INTFLG  000460
INTPAS  000462
INTST   015366
IOTVEC= 000020
KIPAR0= 172340
KIPAR1= 172342
KIPAR2= 172344
KIPAR3= 172346
KIPAR4= 172350
KIPAR5= 172352
```

SYMBOL TABLE

```
KIPAR6= 172354
KIPAR7= 172356
KIPDR0= 172300
KIPDR1= 172302
KIPDR2= 172304
KIPDR3= 172306
KIPDR4= 172310
KIPDR5= 172312
KIPDR6= 172314
KIPDR7= 172316
LENGTH= 000004
LF    = 000012
LKS   = 177546
LOCC  = 076040
LOCCI = 076140
LOOP2   015772
LTCVEC= 000100
L2D0  = 076020
L2D1  = 076021
L2D2  = 076022
L2D3  = 076023
L2D4  = 076024
L2D5  = 076025
L2D6  = 076026
L2D7  = 076027
L3D0  = 076060
L3D1  = 076061
L3D2  = 076062
L3D3  = 076063
L3D4  = 076064
L3D5  = 076065
L3D6  = 076066
L3D7  = 076067
MATC  = 076045
MATCI = 076145
MFPT  = 000007
MMCYL   000446
MMDIS   063136
MMENA   063176
MMSETU  057022
MMVEC = 000250
MOVC  = 076030
MOVCI = 076130
MOVRC = 076031
MOVRCI= 076131
MOVTC = 076032
MOVTCI= 076132
MSGADR  000550
MULP  = 076074
MULPI = 076174
MUL1    055462
MUL1A   055504
MUL1B   055530
MUL1C   055572
MUL2    055634
MUL2A   055642
MUL2B   055654
MUL2C   055672
MUL3    055710
MUL3A   055720
MUL3B   055724
MUL3C   055734
MUL4    055744
MUL4A   055754
MUL4B   055762
MUL4C   055770
MUL5    055776
MUL5A   056002
MUL5B   056012
MUL5C   056016
MUL7    056022
MUL7A   056030
MUL7B   056040
MUL7C   056052
NBIT  = 000010
NL1     025252
NOABRT  060210
NP1     031276
OFFSET  000512
OVPNCH  056330
PACK    057254
PHALT   057674
PLI     023220
PN1     027346
PR0   = 000000
PR1   = 000040
PR2   = 000100
PR3   = 000140
PR4   = 000200
PR5   = 000240
PR6   = 000300
PR7   = 000340
PS    = 177776
PSW   = 177776
PWRMSG  062600
PWRVEC= 000024
RDCHR = 100407
RDLIN = 104410
RESREG  056770
RESRTN  016562
RESVEC= 000010
R0.EXP  000524
R1.EXP  000526
R2.EXP  000530
R3.EXP  000532
R4.EXP  000534
R5.EXP  000536
R6    =%000006
R7    =%000007
SAVPSW  000614
SAVREG  056736
SAVR0   000600
SAVR1   000602
SAVR2   000604
SAVR3   000606
SAVR4   000610
SAVR5   000612
SAV30   001266
SAV32   001270
SCANC = 076042
SCANCI= 076142
SCOPE = 000004
SETBUF  056520
SGNDGT  024406
SIGN0   000450
SIGN1   000452
SIGN2   000454
SKPC  = 076041
SKPCI = 076141
SKPTST  016430
SPACE   066152
SPACEY  072152
SPACE1  062630
SPANC = 076043
SPANCI= 076143
SRCADR= 000002
SR1   = 177572
STACK = 000776
START   001200 G
STKLMT= 177774
SUBN  = 076051
SUBNI = 076151
SUBP  = 076071
SUBPI = 076171
SWR     000426
SWREG   000176
SW0   = 000001
SW00  = 000001
SW01  = 000002
SW02  = 000004
SW03  = 000010
SW04  = 000020
SW05  = 000040
SW06  = 000100
SW07  = 000200
SW08  = 000400
SW09  = 001000
SW1   = 000002
SW10  = 002000
SW11  = 004000
SW12  = 010000
SW13  = 020000
SW14  = 040000
SW15  = 100000
SW2   = 000004
SW3   = 000010
SW4   = 000020
SW5   = 000040
SW6   = 000100
SW7   = 000200
SW8   = 000400
SW9   = 001000
TAB89   055262
TADD    065250
TADDI   066122
TADDPI  066136
TASH    065114
TASHNI  045356
TASHPI  052756
TBITVE= 000014
TCMP    064646
TCMPC   014334
TCMPCI  014416
TCMPNI  042320
TCMPPI  051152
TCVLNI  022076
TCVLPI  020336
TCVNLI  025242
TCVNPI  031266
TCVPLI  023210
TCVPNI  027336
TCVTLN  021764
TCVTLP  020224
TCVTNL  024514
TCVTNP  031050
TCVTPL  023132
TCVTPN  027202
TEMP0   000476
TEMP1   000500
TEMP2   000502
TEMP3   000504
TEMP4   000506
TEMP5   000510
TIMMSG  062436
TIMOUT  056354
TITLE   062376
TKB   = 177562
TKS   = 177560
TKVEC = 000060
TLOCC   010544
TLOCCI  010570
TMATC   015334
TMATCI  015356
TMOVC   004300
TMOVCI  004624
TMOVRC  006146
TMOVTC  007606
TMULPI  056064
TMVRCI  006240
TMVTCI  007700
TPB   = 177566
TPS   = 177564
TPSRVC  016432
TPVEC = 000064
TRANS   063436
TRAPVE= 000034
TRTVEC= 000014
TSCANC  012406
TSCNCI  012424
TSKPC   011436
TSKPCI  011454
TSPANC  013332
TSPNCI  013350
TSTCAS  000544
TYPE  = 104401
TYPOC = 104402
TYPOS = 104404
TYPOS = 104403
UFDFLG  001272
UFDMOD= 000052
UFDSET= 000001
UQUIET  001274
VBIT  = 000002
VMKOR   001276
WAIT    015742
XADDN   032712
XADDP   045456
XASHN   042336
XASHP   051166
XCMPC   013360
XCMPN   040262
XCMPP   047566
XCVTLI  032206
XCVTLN  020346
XCVTLP  016712
XCVTNL  023230
XCVTNP  027352
XCVTPL  022106
XCVTPN  025264
XDIVP   054202
XLOCC   007710
XL2DR   001726
XL3DR   002354
XMATC   014426
XMOVC   002754
XMOVRC  004634
XMOVTC  006250
XMULP   053056
XSCANC  011460
XSKPC   010574
XSPANC  012434
XSUBN   035470
XSUBP   046512
XXDP    000466
ZBIT  = 000004
$APTHD  000300
$ATYC   062120
$ATY1   062074
$ATY3   062102
$ATY4   062112
$AUTOB  000422
$BELL   000436
$BGNLE= 177777
$CHARC  060674
$CKSWR  061144
```

SYMBOL TABLE

$CMTAG	000400	$ISK0 =	000001	$NSK6 =	000300	$$ERFL=	000000	$50055	002724
$CNTLC	062040	$ISK1 =	000001	$NSK7 =	000300	$$FLAG=	000001	$50056	002736
$CNTLG	062033	$ISK2 =	000001	$NULL	000432	$$FROM=	000000	$50057	002750
$CNTLU	062026	$ISK3 =	000001	$OCNT	061140	$$LOC =	057340	$50062	003104
$CNTLZ	062045	$ISK4 =	000001	$OMODE	061142	$$LOCN=	000000	$50063	003106
$CPUOP	000660	$ISK5 =	000001	$OVER	057500	$$REG =	177777	$50064	003144
$CRLF	000443	$LF	000444	$PASS	000640	$$RETU=	000000	$50065	003152
$DEVCT	000642	$LFLG	062315	$PASTM	000306	$$RTN1=	052477	$50066	003162
$DOAGN	056242	$LOCTA=	000001	$PWRAD	057662	$$RTN2=	052500	$50067	003226
$ENDAD	056232	$LPADR	000410	$PWRDN	057516	$$SRC =	000000	$50070	003250
$ENDCT	056306	$LPERR	000412	$PWRUP	057570	$$TGSV=	051546	$50071	003244
$ENDMG	056313	$LSTIN=	000001	$QUES	000442	$$TGS1=	000001	$50072	003250
$ENULL	056310	$LSTTA=	000001	$RDCHR	061532	$$TGS2=	000000	$50073	003270
$ENV	000652	$MAIL	000632	$RDLIN	061662	$$TO =	000000	$50074	003312
$ENVM	000653	$MAMS1	000662	$SAVLE=	177777	$$$TAG=	050000	$50075	003306
$EOP	056132	$MBADR	000302	$SAVR6	057672	$OFILL	061141	$50076	003312
$EOPCT	056304	$MFLG	062314	$SCOPE	057376	$50000	001706	$50077	003326
$ERFLG=	000400	$MNEW	062063	$SETUP=	000000	$50001	001706	$50100	003332
$ERMAX	000416	$MSGAD	000646	$SSK0 =	052507	$50004	001746	$50101	003364
$ERRPC	000420	$MSGLG	000650	$SVLAD	057456	$50005	001774	$50102	003376
$ERRTY	060212	$MSGTY	000632	$SVPC =	000300	$50006	002004	$50103	003410
$ERTTL	000414	$MSWR	062052	$SWREG	000654	$50007	002052	$50104	003422
$ETABL	000652	$MTYP1	000663	$TAGLE=	177777	$50010	002050	$50105	003434
$ETEND	000664	$MXCNT	057514	$TAGNU=	052510	$50011	002066	$50106	003446
$FATAL	000634	$NESTL=	000036	$TEMP =	000300	$50012	002066	$50107	003460
$FFLG	062316	$NSK0 =	000300	$TESTN	000636	$50013	002100	$50110	003542
$FILLC	000434	$NSK1 =	000300	$TIMES	000404	$50014	002112	$50111	003510
$FILLS	000433	$NSK10=	000300	$TPFLG	000435	$50015	002124	$50112	003540
FAND=	000310	$NSK11=	000300	$TRAP	062320	$50016	002136	$50113	003534
FBAD=	000401	$NSK12=	000300	$TRAP2	062342	$50017	002146	$50114	003644
FBLA=	000170	$NSK13=	000300	$TRPAD	062354	$50020	002170	$50115	003546
FCAS=	000150	$NSK14=	000300	$TSK0 =	052504	$50021	002166	$50116	003576
FDEC=	000220	$NSK15=	000300	$TSK1 =	052506	$50022	002202	$50117	003572
FDO =	000340	$NSK16=	000300	$TSK10=	051562	$50023	002202	$50120	003616
FFAL=	000405	$NSK17=	000300	$TSK2 =	052507	$50024	002250	$50121	003630
FGOO=	000400	$NSK2 =	000300	$TSK3 =	052322	$50025	002262	$50122	003644
FIF =	000110	$NSK20=	000300	$TSK4 =	052323	$50026	002274	$50123	003760
FINC=	000210	$NSK21=	000300	$TSK5 =	052325	$50027	002306	$50124	003700
FLOO=	000200	$NSK22=	000300	$TSK6 =	051546	$50030	002316	$50125	003706
FNAM=	000160	$NSK23=	000300	$TSK7 =	051566	$50031	002326	$50126	003732
FNO =	000403	$NSK24=	000300	$TSTM	000304	$50034	002374	$50127	003744
FOR =	000320	$NSK25=	000300	$TSTNM	000402	$50035	002422	$50130	003760
FRTI=	000350	$NSK26=	000300	$TTYIN	062016	$50036	002432	$50131	004004
FRTN=	000300	$NSK27=	000300	$TYPE	060414	$50037	002500	$50132	004104
FSEL=	000140	$NSK3 =	000300	$TYPEC	060630	$50040	002476	$50133	004104
FTHE=	000330	$NSK30=	000300	$TYPEX	060676	$50041	002514	$50134	004116
FTRU=	000404	$NSK31=	000300	$TYPOC	060724	$50042	002514	$50135	004130
FUNT=	000130	$NSK32=	000300	$TYPON	060740	$50043	002526	$50136	004142
FWHI=	000120	$NSK33=	000300	$TYPOS	060700	$50044	002540	$50137	004154
FYES=	000402	$NSK34=	000300	$UNIT	000644	$50045	002552	$50140	004166
$GET42	056222	$NSK35=	000300	$UNITM	000310	$50046	002564	$50141	004200
$GTSWR	061274	$NSK36=	000300	$USWR	000656	$50047	002576	$50142	004214
$HIBTS	000300	$NSK37=	000300	$$ARGC=	000006	$50050	002610	$50143	004230
$ICNT	000406	$NSK4 =	000300	$$BYTE=	000403	$50051	002654	$50144	004250
$IFLEV=	177777	$NSK40=	000110	$$CASE=	000404	$50052	002666	$50145	004260
$ILLUP	057666	$NSK41=	000120	$$DST =	000000	$50053	002700	$50146	004274
$INTAG	000423	$NSK5 =	000300	$$ELOC=	000402	$50054	002712	$50151	004764

SYMBOL TABLE

$50152	004766	$50245	006572	$50335	010466	$50430	012734	$50520	015102
$50153	005024	$50246	006576	$50336	010500	$50431	013000	$50521	015114
$50154	005032	$50247	006620	$50337	010510	$50432	013012	$50522	015140
$50155	005042	$50250	006614	$50340	010520	$50433	013024	$50523	015170
$50156	005110	$50251	006634	$50341	010530	$50434	013036	$50524	015234
$50157	005132	$50252	006712	$50342	010540	$50435	013050	$50525	015246
$50160	005126	$50253	006724	$50345	010676	$50436	013062	$50526	015260
$50161	005132	$50254	006736	$50346	010712	$50437	013074	$50527	015270
$50162	005154	$50255	006750	$50347	010744	$5044	013406	$5053	023256
$50163	005176	$50256	006762	$50350	010774	$50440	013120	$50530	015300
$50164	005172	$50257	006774	$50351	011122	$50441	013160	$50531	015310
$50165	005176	$50260	007006	$50352	011134	$50442	013172	$50532	015320
$50166	005236	$50261	007076	$50353	011146	$50443	013234	$50533	015330
$50167	005250	$50262	007046	$50354	011160	$50444	013254	$50536	015414
$50170	005262	$50263	007074	$50355	011172	$50445	013266	$50537	015414
$50171	005274	$50264	007070	$50356	011204	$50446	013276	$5054	025312
$50172	005306	$50265	007176	$50357	011216	$50447	013306	$50540	015502
$50173	005320	$50266	007102	$50360	011242	$5045	014454	$50541	015664
$50174	005332	$50267	007130	$50361	011332	$50450	013316	$50542	015660
$50175	005422	$50270	007124	$50362	011360	$50451	013326	$50543	015700
$50176	005372	$50271	007150	$50363	011372	$50454	013474	$50544	015724
$50177	005420	$50272	007162	$50364	011402	$50455	013500	$50545	015736
$50200	005414	$50273	007176	$50365	011412	$50456	013560	$50546	016022
$50201	005554	$50274	007264	$50366	011422	$50457	013602	$50547	016040
$50202	005422	$50275	007236	$50367	011432	$50460	013712	$5055	027400
$50203	005434	$50276	007250	$50372	011560	$50461	013736	$50550	016372
$50204	005472	$50277	007264	$50373	011570	$50462	013764	$50551	016070
$50205	005526	$5030	003002	$50374	011622	$50463	014016	$50552	016102
$50206	005554	$50300	007310	$50375	011652	$50464	014030	$50553	016114
$50207	005552	$50301	007414	$50376	011746	$50465	014042	$50554	016126
$50210	005636	$50302	007414	$50377	011760	$50466	014054	$50555	016140
$50211	005574	$50303	007426	$5040	007736	$50467	014066	$50556	016152
$50212	005606	$50304	007440	$50400	012040	$50470	014100	$50557	016164
$50213	005636	$50305	007452	$50401	012052	$50471	014112	$5056	042370
$50214	005662	$50306	007464	$50402	012064	$50472	014136	$50560	016214
$50215	005762	$50307	007476	$50403	012076	$50473	014250	$50561	016242
$50216	005762	$5031	004662	$50404	012110	$50474	014260	$50562	016236
$50217	005774	$50310	007510	$50405	012122	$50475	014270	$50563	016262
$50220	006006	$50311	007524	$50406	012134	$50476	014300	$50564	016274
$50221	006020	$50312	007540	$50407	012160	$50477	014310	$50565	016310
$50222	006032	$50313	007554	$5041	010622	$5050	032764	$50566	016372
$50223	006044	$50314	007566	$50410	012220	$50500	014320	$50567	016344
$50224	006056	$50315	007602	$50411	012232	$50501	014330	$5057	020374
$50225	006072	$5032	006276	$50412	012310	$50504	014526	$50570	016356
$50226	006106	$50320	010012	$50413	012330	$50505	014536	$50571	016372
$50227	006116	$50321	010026	$50414	012342	$50506	014624	$50574	016464
$50230	006126	$50322	010060	$50415	012352	$50507	014576	$50575	016472
$50231	006142	$50323	010110	$50416	012362	$5051	035542	$50602	016602
$50234	006424	$50324	010236	$50417	012372	$50510	014746	$50603	016664
$50235	006432	$50325	010250	$5042	011506	$50511	014722	$50604	016664
$50236	006442	$50326	010262	$50420	012402	$50512	014746	$50607	017056
$50237	006510	$50327	010274	$50423	012534	$50513	015020	$50610	017056
$50240	006532	$50330	010306	$50424	012544	$50514	015032	$50611	017462
$50241	006526	$50331	010320	$50425	012576	$50515	015044	$50612	017130
$50242	006532	$50332	010332	$50426	012626	$50516	015056	$50613	017462
$50243	006554	$50333	010356	$50427	012722	$50517	015070	$50614	017160
$50244	006576	$50334	010446	$5043	012462	$5052	040310	$50615	017224

SYMBOL TABLE

$50616	017310	$5071	046564	$50774	022714	$51071	025520	$51164	027502
$50617	017402	$50710	021124	$50775	022726	$51072	025526	$51165	027500
$50620	017336	$50711	021130	$50776	022752	$51073	025544	$51166	027514
$50621	017340	$50712	021160	$50777	023016	$51074	025550	$51167	027566
$50622	017402	$50713	021204	$51000	023026	$51075	025616	$51170	027566
$50623	017430	$50714	021252	$51001	023036	$51076	025674	$51171	027562
$50624	017462	$50715	021222	$51002	023046	$51077	025672	$51172	027566
$50625	017446	$50716	021246	$51003	023056	$51100	025674	$51173	027612
$50626	017462	$50717	021252	$51004	023066	$51101	025710	$51174	027612
$50627	017524	$5072	047614	$51005	023076	$51102	025734	$51175	027654
$50630	017530	$50720	021314	$51006	023112	$51103	025750	$51176	027654
$50631	017612	$50721	021320	$51007	023126	$51104	025756	$51177	027650
$50632	017624	$50722	021402	$51012	023324	$51105	025774	$51200	027720
$50633	017636	$50723	021414	$51013	023330	$51106	026032	$51201	027720
$50634	017650	$50724	021426	$51014	023422	$51107	026044	$51202	027712
$50635	017662	$50725	021440	$51015	023460	$51110	026066	$51203	027750
$50636	017674	$50726	021452	$51016	023444	$51111	026116	$51204	027744
$50637	017706	$50727	021464	$51017	023770	$51112	026170	$51205	027750
$50640	017722	$5073	022134	$51020	023520	$51113	026170	$51206	030004
$50641	017750	$50730	021476	$51021	023560	$51114	026164	$51207	027776
$50642	017746	$50731	021512	$51022	023560	$51115	026170	$51210	030004
$50643	017774	$50732	021540	$51023	023600	$51116	026230	$51211	030104
$50644	020040	$50733	021536	$51024	023710	$51117	026222	$51212	030134
$50645	020050	$50734	021564	$51025	023614	$51120	026356	$51213	030122
$50646	020060	$50735	021630	$51026	023630	$51121	026264	$51214	030130
$50647	020070	$50736	021640	$51027	023640	$51122	026356	$51215	030134
$50650	020100	$50737	021650	$51030	023702	$51123	026316	$51216	030150
$50651	020110	$5074	053130	$51031	023726	$51124	026352	$51217	030212
$50652	020120	$50740	021660	$51032	023746	$51125	026426	$51220	030212
$50653	020136	$50741	021670	$51033	023770	$51126	026426	$51221	030226
$50654	020136	$50742	021700	$51034	023770	$51127	026420	$51222	030234
$50655	020150	$50743	021710	$51035	024000	$51130	026470	$51223	030252
$50656	020170	$50744	021726	$51036	023776	$51131	026464	$51224	030270
$50657	020174	$50745	021732	$51037	024014	$51132	026470	$51225	030274
$50660	020204	$50746	021744	$51040	024060	$51133	026540	$51226	030336
$50661	020220	$50747	021760	$51041	024052	$51134	026532	$51227	030336
$50664	020624	$5075	054254	$51042	024060	$51135	026540	$51230	030400
$50665	020520	$50752	022202	$51043	024126	$51136	026622	$51231	030376
$50666	020562	$50753	022206	$51044	024140	$51137	026634	$51232	030512
$50667	020546	$50754	022300	$51045	024152	$51140	026646	$51233	030524
$50670	020562	$50755	022564	$51046	024164	$51141	026660	$51234	030536
$50671	020624	$50756	022342	$51047	024176	$51142	026672	$51235	030550
$50672	020610	$50757	022342	$51050	024210	$51143	026704	$51236	030562
$50673	020624	$5076	051220	$51051	024222	$51144	026716	$51237	030574
$50674	020672	$50760	022412	$51052	024246	$51145	026732	$51240	030606
$50675	020650	$50761	022564	$51053	024312	$51146	026760	$51241	030622
$50676	020664	$50762	022512	$51054	024326	$51147	026756	$51242	030650
$50677	021252	$50763	022564	$51055	024332	$51150	027010	$51243	030646
$5070	045530	$50764	022542	$51056	024346	$51151	027056	$51244	030674
$50700	020764	$50765	022556	$51057	024352	$51152	027072	$51245	030740
$50701	020752	$50766	022564	$51060	024366	$51153	027076	$51246	030754
$50702	020764	$50767	022632	$51061	024402	$51154	027112	$51247	030760
$50703	021012	$5077	016740	$51064	025372	$51155	027116	$51250	030774
$50704	021012	$50770	022644	$51065	025376	$51156	027132	$51251	031000
$50705	021070	$50771	022656	$51066	025476	$51157	027146	$51252	031014
$50706	021110	$50772	022670	$51067	025504	$51160	027162	$51253	031030
$50707	021164	$50773	022702	$51070	025526	$51161	027176	$51254	031044

SYMBOL TABLE

$51255	032206	$51346	032664	$51441	034570	$51534	036630	$51627	040572
$51256	032206	$51347	032664	$51442	034636	$51535	036636	$51630	041040
$51257	031332	$51352	032750	$51443	034642	$51536	036700	$51631	040636
$51260	031460	$51353	032744	$51444	034706	$51537	036744	$51632	040634
$51261	031372	$51354	032764	$51445	034714	$51540	037012	$51633	041040
$51262	031460	$51355	033064	$51446	035022	$51541	037016	$51634	040672
$51263	031502	$51356	033070	$51447	035034	$51542	037062	$51635	040720
$51264	031502	$51357	033124	$51450	035046	$51543	037070	$51636	040720
$51265	031506	$51360	033130	$51451	035060	$51544	037114	$51637	041040
$51266	031532	$51361	033170	$51452	035072	$51545	037156	$51640	040744
$51267	031556	$51362	033174	$51453	035104	$51546	037506	$51641	040760
$51270	031560	$51363	033352	$51454	035116	$51547	037436	$51642	041032
$51271	031604	$51364	033414	$51455	035132	$51550	037372	$51643	041006
$51272	031630	$51365	033712	$51456	035160	$51551	037314	$51644	041030
$51273	031632	$51366	033642	$51457	035156	$51552	037236	$51645	041030
$51274	031656	$51367	033576	$51460	035220	$51553	037170	$51646	041040
$51275	031702	$51370	033532	$51461	035366	$51554	037234	$51647	041056
$51276	031704	$51371	033466	$51462	035402	$51555	037222	$51650	041144
$51277	031730	$51372	033426	$51463	035406	$51556	037230	$51651	041132
$51300	031754	$51373	033452	$51464	035422	$51557	037312	$51652	041136
$51301	031756	$51374	033460	$51465	035426	$51560	037304	$51653	041216
$51302	032000	$51375	033522	$51466	035442	$51561	037370	$51654	041216
$51303	032024	$51376	033566	$51467	035470	$51562	037362	$51655	041204
$51304	032026	$51377	033634	$51470	035466	$51563	037430	$51656	041210
$51305	032046	$51400	033640	$51473	035526	$51564	037434	$51657	041264
$51306	032046	$51401	033704	$51474	035522	$51565	037500	$51660	041264
$51307	032064	$51402	033712	$51475	035542	$51566	037506	$51661	041252
$51310	032066	$51403	033736	$51476	035652	$51567	037614	$51662	041256
$51311	032104	$51404	034000	$51477	035656	$51570	037626	$51663	041324
$51312	032106	$51405	034276	$51500	035712	$51571	037640	$51664	041320
$51313	032124	$51406	034226	$51501	035716	$51572	037652	$51665	041324
$51314	032126	$51407	034162	$51502	035756	$51573	037664	$51666	041372
$51315	032144	$51410	034116	$51503	035762	$51574	037676	$51667	041354
$51316	032154	$51411	034052	$51504	036140	$51575	037710	$51670	041360
$51317	032164	$51412	034012	$51505	036202	$51576	037724	$51671	041414
$51320	032620	$51413	034036	$51506	036500	$51577	037752	$51672	041502
$51321	032620	$51414	034044	$51507	036430	$51600	037750	$51673	041470
$51322	032230	$51415	034106	$51510	036364	$51601	040012	$51674	041474
$51323	032260	$51416	034152	$51511	036320	$51602	040160	$51675	041546
$51324	032262	$51417	034220	$51512	036254	$51603	040174	$51676	041534
$51325	032312	$51420	034224	$51513	036214	$51604	040200	$51677	041540
$51326	032314	$51421	034270	$51514	036240	$51605	040214	$51700	041606
$51327	032344	$51422	034276	$51515	036246	$51606	040220	$51701	041574
$51330	032346	$51423	034322	$51516	036310	$51607	040234	$51702	041600
$51331	032376	$51424	034364	$51517	036354	$51610	040262	$51703	041646
$51332	032400	$51425	034714	$51520	036422	$51611	040260	$51704	041642
$51333	032426	$51426	034644	$51521	036426	$51614	040404	$51705	041646
$51334	032430	$51427	034600	$51522	036472	$51615	040402	$51706	041714
$51335	032452	$51430	034522	$51523	036500	$51616	040416	$51707	041676
$51336	032454	$51431	034444	$51524	036524	$51617	040430	$51710	041702
$51337	032476	$51432	034376	$51525	036572	$51620	040460	$51711	042002
$51340	032500	$51433	034442	$51526	037070	$51621	040456	$51712	042012
$51341	032522	$51434	034430	$51527	037020	$51622	040472	$51713	042022
$51342	032524	$51435	034436	$51530	036754	$51623	040522	$51714	042032
$51343	032546	$51436	034520	$51531	036710	$51624	040572	$51715	042042
$51344	032614	$51437	034512	$51532	036644	$51625	040546	$51716	042052
$51345	032614	$51440	034576	$51533	036604	$51626	040572	$51717	042062

SYMBOL TABLE

$51720	042104	$52012	044020	$52102	045316	$52201	050040	$52274	051436
$51721	042126	$52013	043700	$52103	045352	$52202	050064	$52275	051464
$51722	042152	$52014	043650	$52104	045350	$52203	050064	$52276	051472
$51723	042240	$52015	043674	$52105	045430	$52204	050332	$52277	051476
$51724	042262	$52016	043700	$52106	045430	$52205	050130	$52300	051512
$51725	042266	$52017	044020	$52111	045514	$52206	050126	$52301	051520
$51726	042310	$5202	041376	$52112	045510	$52207	050332	$52302	052010
$51727	042314	$52020	043742	$52113	045530	$52210	050164	$52303	051652
$51732	042434	$52021	043746	$52114	046002	$52211	050212	$52304	051620
$51733	042450	$52022	043752	$52115	046014	$52212	050212	$52305	051652
$51734	042472	$52023	044006	$52116	046026	$52213	050332	$52306	051710
$51735	042524	$52024	044176	$52117	046040	$52214	050236	$52307	051710
$51736	042562	$52025	044062	$52120	046052	$52215	050252	$52310	051772
$51737	042546	$52026	044054	$52121	046064	$52216	050324	$52311	051760
$51740	042544	$52027	044062	$52122	046076	$52217	050300	$52312	051754
$51741	042562	$52030	044176	$52123	046112	$52220	050322	$52313	051764
$51742	042576	$52031	044114	$52124	046140	$52221	050322	$52314	052010
$51743	042606	$52032	044120	$52125	046136	$52222	050332	$52315	052232
$51744	043164	$52033	044124	$52126	046164	$52223	050404	$52316	052232
$51745	042770	$52034	044160	$52127	046366	$52224	050376	$52317	052076
$51746	042676	$52035	044370	$52130	046402	$52225	050402	$52320	052124
$51747	042670	$52036	044240	$52131	046406	$52226	050474	$52321	052232
$51750	042676	$52037	044246	$52132	046422	$52227	050416	$52322	052170
$51751	042710	$52040	044300	$52133	046426	$52230	050416	$52323	052232
$51752	042710	$52041	044310	$52134	046464	$52231	050444	$52324	052216
$51753	042752	$52042	044316	$52135	046512	$52232	050466	$52325	052222
$51754	042744	$52043	044354	$52136	046510	$52233	050472	$52326	052322
$51755	042770	$52044	044370	$52141	046550	$52234	050542	$52327	052332
$51756	043164	$52045	044572	$52142	046544	$52235	050534	$52330	052336
$51757	043032	$52046	044456	$52143	046564	$52236	050540	$52331	052346
$51760	043102	$52047	044464	$52144	047046	$52237	050632	$52332	052352
$51761	043064	$52050	044502	$52145	047060	$52240	050554	$52333	052400
$51762	043062	$52051	044512	$52146	047072	$52241	050554	$52334	052412
$51763	043100	$52052	044520	$52147	047104	$52242	050602	$52335	052432
$51764	043164	$52053	044556	$52150	047116	$52243	050624	$52336	052462
$51765	043122	$52054	044572	$52151	047130	$52244	050630	$52337	052456
$51766	043120	$52055	044664	$52152	047142	$52245	050720	$52340	052474
$51767	043136	$52056	044700	$52153	047156	$52246	050734	$52341	052520
$51770	043142	$52057	044704	$52154	047204	$52247	050740	$52342	052532
$51771	043164	$52060	044732	$52155	047202	$52250	050754	$52343	052622
$51772	043202	$52061	044744	$52156	047230	$52251	050760	$52344	052636
$51773	043256	$52062	044764	$52157	047442	$52252	051004	$52345	052642
$51774	043306	$52063	045014	$52160	047456	$52253	051072	$52346	052656
$51775	043336	$52064	045010	$52161	047462	$52254	051114	$52347	052662
$51776	043522	$52065	045026	$52162	047476	$52255	051120	$52350	052700
$51777	043406	$52066	045042	$52163	047502	$52256	051142	$52351	052712
$52000	043402	$52067	045040	$52164	047540	$52257	051146	$52352	052716
$52001	043406	$52070	045054	$52165	047566	$52262	051306	$52353	052752
$52002	043522	$52071	045100	$52166	047564	$52263	051300	$52354	052750
$52003	043454	$52072	045112	$52171	047710	$52264	051304	$52355	053030
$52004	043462	$52073	045222	$52172	047706	$52265	051362	$52356	053030
$52005	043466	$52074	045236	$52173	047720	$52266	051322	$52361	053114
$52006	043522	$52075	045242	$52174	047754	$52267	051322	$52362	053110
$52007	043604	$52076	045256	$52175	047752	$52270	051342	$52363	053130
$5201	041040	$52077	045262	$52176	047764	$52271	051356	$52364	053402
$52010	043562	$52100	045300	$52177	050014	$52272	051362	$52365	053414
$52011	043604	$52101	045312	$52200	050064	$52273	051404	$52366	053426

```
CJKDHB0 KEF11-B CIS DIAGNOSTIC  MACRO M1200  09-APR-85 11:22  PAGE 62-12
                                                                                SEQ 0364
SYMBOL TABLE

$52367  053440          $52413  054240          $52434  055166          $52455  056672          $52476  057252
$52370  053452          $52414  054234          $52435  055172          $52456  056672          $52477  057374
$52371  053464          $52415  054254          $52436  055230          $52457  056650          $52500  057374
$52372  053476          $52416  054536          $52437  055256          $52460  056734          $52501  057306
$52373  053512          $52417  054550          $52440  055254          $52461  056734          $52502  057312
$52374  053540          $52420  054562          $52441  056132          $52462  056714          $52503  057322
$52375  053536          $52421  054574          $52442  056454          $52463  056766          $52504  057374
$52376  053564          $52422  054606          $52443  056454          $52464  056766          $52505  057334
$52377  053766          $52423  054620          $52444  056516          $52465  057020          $52506  057334
$52400  054002          $52424  054632          $52445  056516          $52466  057020          $52507  057362
$52401  054006          $52425  054646          $52446  056476          $52467  057252          $5401   026026
$52402  054022          $52426  054674          $52447  056562          $52470  057252          $5402   026052
$52403  054026          $52427  054672          $52450  056562          $52471  057046          $5403   026764
$52404  054064          $52430  054720          $52451  056540          $52472  057062          $5601   043234
$52405  054112          $52431  055132          $52452  056626          $52473  057062          $5602   043242
$52406  054110          $52432  055146          $52453  056626          $52474  057226          $5603   045032
$52407  054154          $52433  055152          $52454  056604          $52475  057244          .$X     = 000300
$52410  054154

. ABS.  076152     000
        000000     001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  32913 WORDS  ( 129 PAGES)
DYNAMIC MEMORY:  20060 WORDS  ( 77 PAGES)
ELAPSED TIME:  01:09:37
CJKDHB,CJKDHB/-SP=ORION/ML,SPMAC/ML,CJKDHB
```