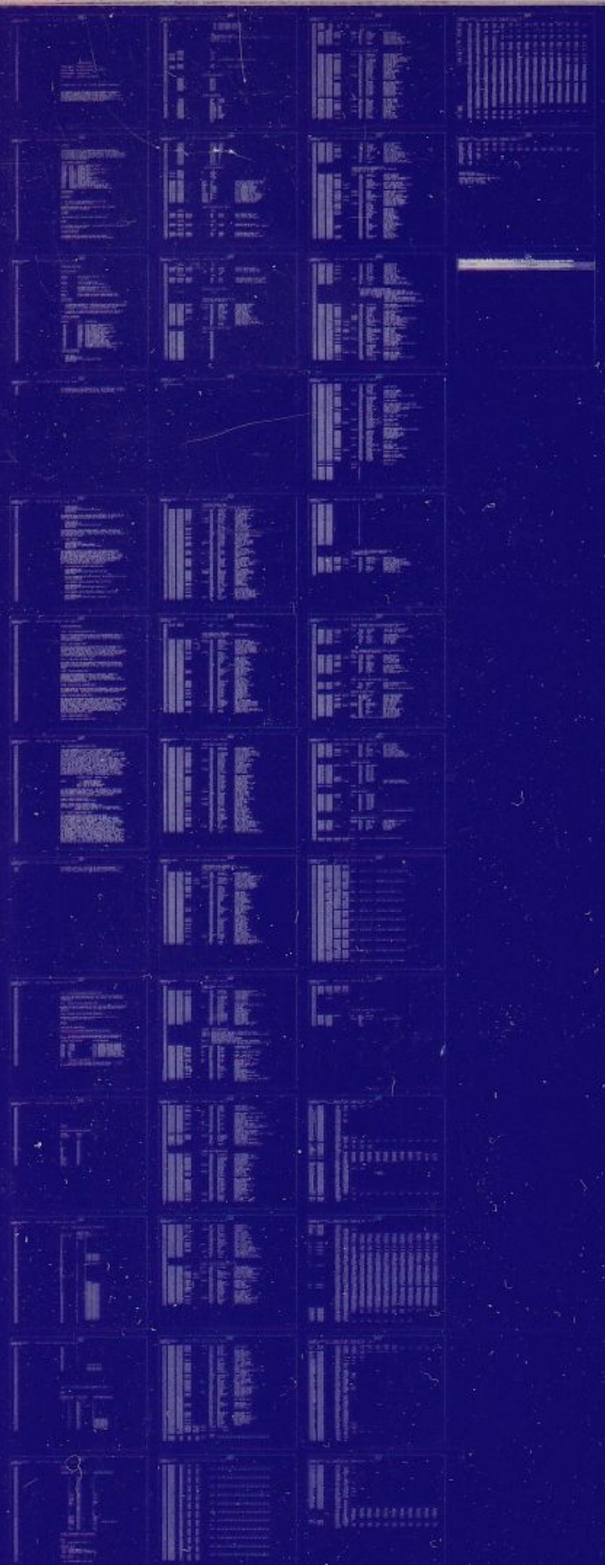
RT01/RT02

TERMINAL
MD-11-DZRTA-C

EP-DZRTA-C-DL-A

FICHE 1 OF 1

NOV 1976
digital
MADE IN USA



.REM !

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZRTA-C-D

PRODUCT NAME: RT01/RT02 TERMINAL DIAGNOSTIC

DATE RELEASED: AUGUST, 1976

MAINTAINER: DIAGNOSTIC ENGINEERING

1. ABSTRACT

THE RT01/RT02 PDP-11 LINE TERMINAL DIAGNOSTIC CONSISTS
OF A PACKAGE OF TEST PROGRAMS TO CHECKOUT EITHER TERMINAL
WHEN ATTACHED TO A PDP-11 TELETYPE CONTROL (KL11 OR EQUIVALENT).
ALL TESTS ARE INCLUDED IN ONE OBJECT TAPE. IF DESIRED TO TEST
THE RT01/RT02 ON ANOTHER OPTION, REFER TO THE ON-LINE PORTION
OF THAT OPTION'S DIAGNOSTIC.

THE AVAILABLE TESTS ARE.

TST1A	RT01	STATUS INDICATOR TEST
TST1B	RT01	DISPLAY TEST
TST1C	RT01	KEYBOARD OCTAL DISPLAY TEST
TST2A	RT02-A	DISPLAY TEST
TST2B	RT02-A	KEYBOARD OCTAL DISPLAY TEST
TS8AA	RT02-B	AUDIO ALARM TEST
TS82A	RT02-B	DISPLAY TEST
TS83	RT02-B	KEYBOARD OUTPUT TEST
TSCAA	RT02-C	DISPLAY TEST
TSC2A	RT02-C	OCTAL KEYBOARD DISPLAY TEST
TSC3	RT02-C	BADGE READER TEST
TST3	RT01/2	KEYBOARD CHARACTER DISPLAY TEST
TST4	TRANSMIT SR BITS 7-0 TO TERMINAL ONCE	
TST5	TRANSMIT SR BITS 7-0 TO TERMINAL REPEATEDLY	

2. REQUIREMENTS

2.1 EQUIPMENT

1. ANY PDP-11 LINE COMPUTER WITH 4K OF CORE.
2. A KL11 OR EQUIVALENT OPTION.

REFER TO SECTION 7 IF THE KL11 DOES NOT HAVE STANDARD
DEVICE ASSIGNMENT ADDRESSES.

2.2 STORAGE

THIS PROGRAM USES LOCATIONS 0000 THROUGH 6730

2.3 OTHER

IF AN RT02-C IS TO BE TESTED, USE MAINDEC-00-DZRTA-C-CA CARDS
FOR THE BADGE READER TEST.

3. LOADING PROCEDURE

THE BINARY TAPE IS PUNCHED IN ABSOLUTE FORMAT.
THE ABSOLUTE LOADER CAN BE USED TO LOAD THE PROGRAM.

4. OPERATING PROCEDURE

4.1 SWITCH SETTINGS

SR15=1	HALT AT END OF CURRENT TEST
SR15=0	DO NOT HALT
SR14=1	LOOP ON CURRENT TEST
SR14=0	CYCLE THROUGH TESTS NORMALLY
SR11=1	REQUIRES RTO1/RTO2 KEY STRUCK TO CONTINUE A DISPLAY
SR11=0	DISPLAY TESTS WILL CYCLE NORMALLY
SR0=1	REJECT BADGE ON RTO2-C BADGE READER TEST
SR0=0	ACCEPT BADGE ON RTO2-C BADGE READER TEST

4.2 PRELIMINARY

A. THE PROGRAM ASSUMES THE TERMINALS WILL BE CHECKED OUT IN THE CONSOLE TELETYPE SLOT. IF AN ADDITIONAL KL11 IS USED REFER TO SECTION 7 TO CHANGE THE DEVICE ASSIGNMENT CODES.

B. IF AN RTO1 IS USED WITH LESS THAN 12 NIXIES PUT THE OCTAL NUMBER INTO LOCATION 1202 BEFORE STARTING THE PROGRAM.

4.3 STARTING ADDRESSES

ADDRESS	TEST	DESCRIPTION
200	TST1A	RTO1 STATUS INDICATOR TEST
204	TST1B	RTO1 DISPLAY TEST
210	TST1C	RTO1 OCTAL KYBD DISPLAY TEST
300	TST2A	RTO2-A DISPLAY TEST
304	TST2B	RTO2-A OCTAL KYBD DISPLAY TEST
400	TSBAA	RTO2-B AUDIO ALARM TEST
404	TSB2A	RTO2-B DISPLAY TEST
410	TSB3	RTO2-B KEYBOARD OUTPUT TEST
500	TSCAA	RTO2-C DISPLAY TEST
504	TSC2A	RTO2-C OCTAL KYBRD TEST
510	TSC3	RTO2-C BADGE READER TEST
600	TST3	RTO1/RTO2 KYBD CHARACTER DISPLAY TEST
604	TST4	TRANSMIT SR BITS 7-0 ONCE
610	TST5	TRANSMIT SR BITS 7-0 REPEATEDLY

4.4 STARTING PROCEDURE

A. RTO1 TEST
   LOAD ADDRESS 200
   SEE SECTION 4.1 FOR SWITCH SETTINGS
   PRESS START

```
                        THE PROGRAM WILL CYCLE THROUGH TST1A,  TST1B AND WAIT IN TST1C
                        FOR THE OPERATOR TO PRESS A KEY ON THE RT01 KEYBOARD.  AFTER 16
                        KEYS HAVE BEEN PRESSED THE PROGRAM WILL RESTART TST1B.
```

```
                                        B.  RT02-A TESTS
                                            LOAD ADDRESS 300
                                            SEE SECTION 4.1 FOR SWITCH SETTINGS
                                            PRESS START

                                        THE PROGRAM WILL CYCLE THROUGH TST2A AND WAIT IN TST2B FOR THE
                                        OPERATOR TO PRESS A KEY ON THE RT02 KEYBOARD.  AFTER 30 KEYS
                                        HAVE BEEN PRESSED THE PROGRAM WILL RESTART AT TST2A.

                                        C.  RT02-B TESTS
                                            LOAD ADDRESS 400
                                            SEE SECTION 4.1 FOR SWITCH SETTINGS
                                            PRESS START

                                        THE PROGRAM WILL CYCLE THROUGH TSBAA,  TSB2A,  AND WAIT IN
                                        TSB3 FOR THE OPERATOR TO PRESS KEYS ON THE RT02-B KEYBOARD.
                                        SEE DESCRIPTION OF TSB3.  AFTER COMPLETION OF TSB3 THE PROGRAM
                                        WILL RESTART AT TSBAA.

                                        D.  RT02-C TESTS

                                            LOAD ADDRESS 500
                                            INSERT CARD INTO BADGE READER
                                            SEE SECTION 4.1 FOR SWITCH SETTINGS
                                            PRESS START

                                        USE MAINDEC-DD-DZRTA-C-CA CARDS FOR THE BADGE READER TEST.
                                        THE PROGRAM WILL CYCLE THROUGH TSCAA AND WAIT IN TSC2A
                                        FOR THE OPERATOR TO PRESS A KEY ON THE RT02 KEYBOARD.  AFTER
                                        30 KEYS HAVE BEEN PRESSED THE PROGRAM WILL GO TO TSC3.  THE
                                        PROGRAM WILL READ CARDS AND ACCEPT OR REJECT THEM DEPENDING
                                        ON THE SETTING OF SR0.  (SEE TEST DESCRIPTION FOR DETAILS)
                                        THE TEST IS CONTIOUS RUNNING,  SO TO EXIT FROM TSTC3
                                        HALT THE PROGRAM AND RESTART AT ADDRESS 500.

                                        E.  TST3 RT01/RT02 KEYBOARD CHARACTER TEST

                                            LOAD ADDRESS 600
                                            THERE ARE NO SWITCH SETTINGS FOR THIS TEST
                                            PRESS START

                                            THE KEY STRUCK ON THE RT01/RT02 WILL BE DISPLAYED IF A LEGAL
                                            DISPLAY CHARACTER.

                                        F.  TST4 TRANSMIT SWITCH REGISTER ONCE (RT01/RT02)

                                            LOAD ADDRESS 604
                                            SR11 ONLY SWITCH SETTING USED (SECTION 4.1)
                                            PRESS START

                                        G.  TST5 TRANSMIT SWITCH REGISTER REPEATEDLY (RT01/RT02)

                                            LOAD ADDRESS 610
                                            SR 11 ONLY SWITCH SETTING USED. (SECTION 4.1)
                                            PRESS START
```

5. PROGRAM DESCRIPTIONS

TST1A - RT01 STATUS INDICATOR TEST -

THE 4 STATUS INDICATORS WILL BE LIT IN THE SEQUENCE SHOWN IN TABLE 1. THERE IS A DELAY BETWEEN EACH TRANSMISSION BUT IF SWITCH REGISTER BIT 11 IS SET A KEY WILL HAVE TO BE PRESSED TO CONTINUE THE SEQUENCE.

TST1B - RT01 DISPLAY TEST -

DISPLAYS ALL ZEROES AND A DECIMAL POINT AT THE EXTREME LEFT. THE NIXIES ARE CLEARED AND ALL 1'S, 2'S UP TO 9'S ARE TRANSMITTED IN THE SAME MANNER. ALL 0'S ARE SENT WTH NO DECIMAL POINT. THEN SHIFTED TO THE LEFT WITH SPACES. THEN THE DISPLAY IS CLEARED AND A DECIMAL POINT IS SHIFTED ACROSS THE DISPLAY. THE TEST THEN SEQUENTIALLY DISPLAYS .0-9 AND A SPACE.

TST1C - RT01 OCTAL KEYBOARD TEST -

THE OCTAL CODE OF THE KEY STRUCK IS DISPLAYED IN THE NIXIES. THIS TEST IS RUN IN INTERRUPT MODE. AFTER 16 KEYS HAVE BEEN PRESSED THE PROGRAM RESTARTS AT TST1B. SEE TABLE 3 FOR OCTAL CODES.

TST2A - RT02-A DISPLAY TEST -

DISPLAYS THE CHARACTERS IN TABLE 2 ONCE EACH. AFTER 32 CHARACTERS ARE RECIEVED THE DISPLAY WILL BE BLANKED, UNBLANKED, THEN CLEARED. THE LAST 32 CHARACTERS WILL THEN BE DISPLAYED SEQUENTIALLY. THE TEST WILL THEN DISPLAY 32 EACH OF THE CHARACTERS IN TABLE 2.

TST2B - RT02-A OCTAL KEYBOARD TEST -

THE OCTAL CODE OF THE KEY STRUCK IS DISPLAYED. THIS TEST IS RUN IN INTERRUPT MODE. AFTER 30 KEYS HAVE BEEN PRESSED THE PROGRAM RESTARTS AT TST2A. SEE TABLE 3 FOR OCTAL CODES DISPLAYED.

TSB&A - RT02-B AUDIO ALARM TEST -

THE BELL ON THE RT02-B WILL SOUND 3 TIMES AS AN * IS DISPLAYED. THE DISPLAY WILL CLEAR, SEND 27 ZEROS, AND THE BELL SHOULD SOUD AS AN * IS SENT. THE DISPLAY WILL CLEAR AGAIN, SEND 27 ZEROS, AND THE BELL SHOULD SOUND AS AN * IS SENT. THEN WITHOUT CLEARING THE DISPLAY, 31 ONES WILL BE SENT TO CYCLE THE DISPLAY, THE BELL SHOULD SOUND WHEN THE * IS REACHED. THE DISPLAY WILL CLEAR, THEN ALL CONTROL CHARACTERS EXCEPT THE BELL AND LF WILL BE SENT. AFTER 27 ZEROS ARE SENT THE BELL SHOULD SOUND WHEN THE * IS SENT.

TSB2A - RT02-B DISPLAY TEST -

SAME AS TST2A (RT02-A DISPLAY TEST).

```
                        TSB3 - RT02-B KEYBOARD OUTPUT TEST -

                        THIS TEST COMPARES THE OUTPUT OF THE RT02 KEYBOARD AS THE
                        OPERATOR PRESSES EACH KEY,  WITH A TABLE IN MEMORY.
                        TO PRESS THE KEYS IN THE CORRECT SEQUENCE,  START IN THE UPPER
                        LEFTHAND CORNER ( ` ) AND GO FROM LEFT TO RIGHT,  LINE BY LINE.
                        ***  DO NOT HIT THE SHIFT LOCK OR SHIFT KEYS ! ***
                        AFTER HITTING THE SPACE BAR AT THE END OF THE FIRST PASS,  PRESS
                        THE SHIFT LOCK AND REPEAT.  AFTER HITTING THE SPACE BAR AT THE
                        END OF THE SECOND PASS,  PRESS THE SHIFT KEY;  THEN REPEAT
                        WHILE HOLDING DOWN THE CTL KEY.  AFTER HITTING THE SPACE BAR
                        AT THE END OF THE THIRD PASS,  THE PROGRAM WILL RETURN TO TSBAA.
                        THUS,  THE TEST CHECKS ALL THE KEYS WITH THE KEYBOARD IN THE
                        UNSHIFTED AND SHIFTED MODES AND WITH THE CTL KEY DEPRESSED.
                        IF THE RECEIVED CODE IS CORRECT,  THE CHARACTER WILL BE
                        DISPLAYED ON THE RT02 (IF IT CAN BE DISPLAYED).  IF AN ERROR
                        OCCURS DURING THE TEST,  THE BELL WILL SOUND AND THE MESSAGE:

                                        RECEIVED X=AAA    WANTED Y=BBB

                        WHERE:            X = CHARACTER RECEIVED
                                          Y = CHARACTER WANTED
                                        AAA = CODE OF CHAR RECEIVED
                                        BBB = CODE OF CHAR WANTED

                        WILL BE DISPLAYED ON THE RT02-B.  THE PROGRAM WILL THEN WAIT
                        FOR THE CORRECT CHARACTER.  IF AN ERROR MESSAGE IS DISPLAYED
                        WHEN A CORRECT CHARACTER WAS SENT,  IT WOULD INDICATE A BAD
                        KEYBAORD SWITCH (A SWITCH SENDING DOUBLE CHARACTERS).

                        TSCAA - RT02-C DISPLAY TEST -
                        SAME AS TST2A (RT02-A DISPLAY TEST).

                        TSC2A - RT02-C OCTAL KEYBOARD TEST -
                        SAME AS TST2B (RT02-A OCTAL KEYBOARD TEST)
                        *** EXCEPT:   A CARD MUST BE INSERTED IN THE BADGE READER
                                      BEFORE STARTING THE TEST - TO ENABLE THE KYBRD.

                        TSC3 - RT02-C BADGE READER TEST -

                        USE MAINDEC-00-DZRTA-C-CA CARDS FOR THIS TEST.
                        THIS TEST READS THE 22 CHARACTERS ON THE BADGE AND CHECKS
                        FOR THE PROPER FIRST AND LAST CONTROL CHARACTERS,  AS WELL
                        AS THE CORRECT NUMBER OF CHARACTERS.  IF THE FIRST CHARACTER
                        IS NOT AN ASCII SOH,  THE MESSAGE:  "FIRST CHAR NOT ASCII SOH"
                        WILL BE DISPLAYED.  IF THE LAST CHARACTER IS NOT A CR,  THE
                        MESSAGE:  "LAST CHAR NOT ASCII CR"  WILL BE DISPLAYED.
                        IF 22 CHARACTERS ARE NOT RECEIVED BETWEEN THE SOH AND CR,
                        THE MESSAGE:  "DID NOT GET RIGHT # OF CHARS"  WILL BE DISPLAYED.
                        THE BADGE BEING READ CAN BE ACCEPTED OR REJECTED BY SETTING
                        SR0 AS DESIRED.  WITH SR0=1 THE BADGE WILL BE REJECTED,  AFTER
                        THE DATA READ IS DISPLAYED,  THE PROGRAM WILL HALT.  PRESS
                        THE CONTINUE KEY ON THE CONSOLE TO REREAD THE BADGE.  WHEN THE
                        BADGE IS REREAD,  IT WILL BE ACCEPTED AND REJECTED AFTER THE
                        ACCEPT LIGHT IS LIT.  WITH SR0=0,  CARDS CAN BE READ ONE AFTER
                        ANOTHER AND WILL AUTOMATICALLY BE EJECTED AFTER BEING READ AND
```

```
                                THE ACCEPT LIGHT LIT.  THE TEST IS CONTINUOUS RUNNING,  SO TO
                                EXIT THIS TEST - PRESS HALT AND RESTART AT ADDRESS 500.
                                SR3 CAN BE CHANGED BEFORE INSERTING EACH CARD IN THE READER.
```

TST3 - RT01/RT02 KEYBOARD CHARACTER TEST -

THIS IS A CONTINUOUS RUNNING TEST THAT OUTPUTS THE CHARACTER STRUCK ON THE RT01/RT02 KEYBOARD. SEE TABLE 3 FOR DISPLAYABLE CHARACTERS.

TST4 - TRANSMIT SWITCH REGISTER ONCE -

OUTPUTS THE ASCII CHARACTER PUT INTO THE CONSOLE SWITCH REGISTER BITS 7-0. THE TEST WILL OUTPUT ONE CHARACTER FOR EACH SWITCH SETTING.

TST5 - TRANSMIT SWITCH REGISTER REPEATEDLY -

REPEATEDLY OUTPUTS THE ASCII CHARACTER PUT INTO BITS 7-0 OF THE CONSOLE SWITCH REGISTER.

6. ERRORS

THERE ARE NO ERROR HALTS

7. TESTING AT NON-STANDARD ADDRESSES AND/OR VECTORS

A. IF THE KL11 USED HAS NON-STANDARD DEVICE ASSIGNMENTS, CHANGE THE FOLLOWING LOCATIONS AFTER LOADING THE PROGRAM:

LOCATION	FROM STANDARD	TO NON-STANDARD
1206	177566	KL11 TRANSMIT BUFFER ADDRESS
1210	177564	KL11 TRANSMIT STATUS ADDRESS
1212	177562	KL11 RECEIVER BUFFER ADDRESS
1214	177560	KL11 RECEIVER STATUS ADDRESS
1216	60	KL11 RECEIVER INTERRUPT VECTOR
1220	200	KL11 PRIORITY LEVEL RECEIVER
1224	64	KL11 TRANSMIT INTERRUPT VECTOR
1226	200	KL11 PRIORITY LEVEL TRANSMITTER

B. PROCEED TO USE SYSTEM, OR

C. DUMP OUT ENTIRE PROGRAM IN ABSOLUTE FORMAT, TO HAVE AN UPDATED OBJECT TAPE THAT REFLECTS YOUR SYSTEM.

8. TABLE 1

RTO1 STATUS INDICATOR SEQUENCE

STATUS INDICATOR	ASCII CODE
1	121
2	122
3	124
4	130
1+2	123
2+3	126
3+4	134
1+4	131
1+3	125
2+4	132
1+2+3	127
1+3+4	135
1+2+4	133
2+3+4	136
1+2+3+4	137

TABLE 2 CHARACTERS DISPLAYED ON RTO1/RTO2

ASCII CODE	RTO2-A DISPLAY	RTO1 DISPLAY
40	(SPACE)	0
41	!	1
42	"	2
43	#	3
44	$	4
45	%	5
46	&	6
47	'	7
50	(	8
51	)	9
52	*	(CLEAR DISPLAY)
53	+	(CLEAR DISPLAY)
54	,	(CLEAR DISPLAY)
55	-	(CLEAR DISPLAY)
56	.	.
57	/	(CLEAR DISPLAY)
60	0	0
61	1	1
62	2	2
63	3	3
64	4	4
65	5	5
66	6	6
67	7	7
70	8	8
71	9	9
72	:	(SPACE)
73	;	(CLEAR DISPLAY)
74	<	(CLEAR DISPLAY)
75	=	(CLEAR DISPLAY)
76	>	(CLEAR DISPLAY)
77	?	(CLEAR DISPLAY)
100	@	(CLEAR DISPLAY)
101	A	(CLEAR DISPLAY)
102	B	(CLEAR DISPLAY)
103	C	(CLEAR DISPLAY)
104	D	(CLEAR DISPLAY)
105	E	(CLEAR DISPLAY)
106	F	(CLEAR DISPLAY)
107	G	(CLEAR DISPLAY)
110	H	(CLEAR DISPLAY)
111	I	(CLEAR DISPLAY)
112	J	(CLEAR DISPLAY)
113	K	(CLEAR DISPLAY)
114	L	(CLEAR DISPLAY)
115	M	(CLEAR DISPLAY)
116	N	(CLEAR DISPLAY)
117	O	(CLEAR DISPLAY)
120	P	
121	Q	

```
                                        122             R
                                        123             S
                                        124
                                        125             U
                                        126             V
                                        127             W
                                        130             X
                                        131             Y
                                        132             Z
                                        133             [
                                        134                     (MODIFIED ASCII)
                                        135             ]
                                        136                     (MODIFIED ASCII)
                                        137                     (MODIFIED ASCII)
```

TABLE 3 RT01/RT02 KEYBOARD TRANSMITTED CODES

RT01

KEYBOARD NAME	OCTAL CODE	CHARACTER DISPLAYED
0	060	0
1	061	1
2	062	2
3	063	3
4	064	4
5	065	5
6	066	6
7	067	7
8	070	8
9	071	9
A	101	(CLEAR DISPLAY)
B	102	(CLEAR DISPLAY)
C	103	(CLEAR DISPLAY)
D	104	(CLEAR DISPLAY)
E	105	(CLEAR DISPLAY)
F	106	(CLEAR DISPLAY)

```
                        R[02-A/C
                        --------

                        KEYBOARD NAME   OCTAL CODE      CHARACTER DISPLAYED
                        -------------   ----------      -------------------

                              0         060             0
                              1         261             1
                              2         262             2
                              3         063             3
                              4         264             4
                              5         065             5
                              6         066             6
                              7         267             7
                              8         270             8
                              9         071             9
                              SHIFT     NONE            NONE
                              .         056             .
                              SPACE     240             NONE
                              TOTAL     324             T
                              YES       006             NONE
                              SEND      215             NONE
                              ASK       104             D
                              X         252             *
                              -         055             -
                              +         053             +
                              @         300             @
                              %         245             %
                              ?         077             ?
                              BREAK     000             NONE
                              REL       033             NONE
                              BEGIN     202             NONE
                              ERROR     377                             (MODIFIED ASCII)
                              CLEAR     012             NONE
                                        257             /
                              GO        107             G
                              STOP      003             NONE


                        CONTROL CHARACTERS (NOT DISPLAYED)
                        ----------------------------------

                        RT01
                        ----
                         100   CLEAR DISPLAY (ASCII @)

                        RT02-A,B, OR C
                        --------------
                          012   CLEAR DISPLAY (LF)
                          016   BLANK DISPLAY (CONTROL N)
                          017   UNBLANK DISPLAY (CONTROL O)

                        RT02-C (ONLY)
                        -------------

                           005  REREAD BADGE (ASCII ENQ)
```

```
                                        006  ACCEPT BADGE (ASCII ACK)
                                        007  REJECT BADGE (ASCII BEL)
                                        013  EJECT BADGE (ASCII VT)
                                        025  REREAD BADGE (ASCII NAK)

                                .ABS
                                .TITLE MAINDEC-11-DZRTA-C
                                ;COPYRIGHT 1972, 1973, 1974 BY DIGITAL EQUIPMENT CORPORATION
                                .LIST ME
                                .NLIST MC,MD,CND

                    000000      .=0
                    000002      .=.+2
000002  005000      HALT

000004  000006      .+2
000006  030000      HALT    ;BUS ERROR TRAP TO PREVIOUS LOCATION

                    000024      .=24
000024  006076      PFAIL           ;POWER FAIL ROUTINE
000026  000000      0


                ;REGISTER ASSIGNMENTS

                    000000          R0=%0
                    000001          R1=%1
                    000002          R2=%2
                    000003          R3=%3
                    000004          R4=%4
                    000005          R5=%5
                    000006          R6=%6
                    000007          R7=%7


                ;EQUATE STATEMENTS

                    177570          CSR=    177570
                    177776          CC=     177776
                    177776          PSW=    177776
                    000000          OPEN=   0
                    000340          PRTY7=  340
                    000300          PRTY6=  300
                    000240          PRTY5=  240
                    000200          PRTY4=  200
                    000140          PRTY3=  140
                    100000          BIT15=  100000
                    040000          BIT14=  40000
                    020000          BIT13=  20000
```

```
                        010000                  BIT12=  10000
                        004000                  BIT11=  4000
                        002000                  BIT10=  2000
                        001000                  BIT9=   1000
                        000400                  BIT8=   400
                        000200                  BIT7=   200
                        000100                  BIT6=   100
                        000040                  BIT5=   40
                        000020                  BIT4=   20
                        000010                  BIT3=   10
                        000004                  BIT2=   4
                        000002                  BIT1=   2
                        000001                  BIT0=   1

                                        ;LOCATION ASSIGNMENTS

                        000046                  .=46
000046  006746                                  $ENDAD
                        000052                  .=52
000052  020000                                  20000
                        001200                  .=1200

001200  000000          BUFF:   OPEN            ;PREVIOUS LOCATIONS FOR STACK
001202  000014          CNT:    14              ;NO. OF RTOI NIXIES IN OCTAL
001204  000020          BRCTA:  20              ;HOLDS DELAY COUNT
001206  177566          TPB:    177566          ;KL11 TRANSMIT BUFFER
001210  177564          TPS:    177564          ;KL11 TRANSMIT STATUS
001212  177562          TKB:    177562          ;KL11 RECEIVE BUFFER
001214  177560          TKS:    177560          ;KL11 RECEIVE STATUS
001216  000060          TKVTR:  60              ;KL11 RECEIVE INT. VECTOR
001220  000200          TKLVL:  PRTY4           ;KL11 RECIEVER INT. PRIORITY
001222  000064          TPVTR:  64              ;KL11 TRANSMIT INT. VECTOR.
001224  000200          TPLVL:  PRTY4           ;KL11 TRANSMIT INT PRIORITY
001226  000000          CRBUFT: 0               ;HOLDS TRANSMITTED CHAR
001230  000000          CRBUFR: 0               ;HOLDS RECIEVED CHAR.


                                        ;ENTRY POINTS FOR ALL TESTS

                        000200                  .=200

000200  000167  001174                          JMP     TST1A           ;RTOI STATUS IND. TEST
000204  000167  001302                          JMP     TST1B           ;RTOI DISPLAY TEST1
000210  000167  001564                          JMP     TST1C           ;RTOI OCTAL KYBD DISPLAY

                        000300                  .=300

000300  000167  001704                          JMP     TST2A           ;RT02-A DISPLAY TEST1
000304  000167  002146                          JMP     TST2B           ;RT02-A KYBD OCTAL  KYBD  DISPLAY

                        000400                  .=400

000400  000167  002254                          JMP     TSBAA           ;RT02-B AUDIO ALARM TEST
000404  000167  002562                          JMP     TSB2A           ;RT02-B DISPLAY TEST
000410  000167  003014                          JMP     TSB3            ;RT02-B KEYBOARD OUTPUT TEST
```

```
                    000500                          .=500
                    
000500  000167  003620                              JMP     TSCAA           ;RTO2-C DISPLAY TEST
000504  000167  004062                              JMP     TSC2A           ;RTO2-C OCTAL KYBRD TEST
000510  000167  004322                              JMP     TSC3            ;RTO2-C BADGE READER TEST
                    
                    000600                          .=600
                    
000600  000167  005000                              JMP     TST3            ;RTO1/RTO2 DISPLAY CHAR. STRUCK
000604  000167  005040                              JMP     TST4            ;TRANSMIT SR BITS 7-0 ONCE
000610  000167  005072                              JMP     TST5            ;TRANSMIT SR BITS 7-0 REPEATEDLY
                    
                    
                                                ;BEGINNING OF MAIN TESTS
                    
                    001400                          .=1400
                    
                    
                    
                                                ;TST1A RTO1 STATUS INDICATOR TEST
                                                ;OUTPUTS ASCII 120-137
                    
001400  012706  001200                  TST1A:  MOV     #BUFF,R6        ;SETUP STACK BUFFER
001404  012704  001446                          MOV     #TABR, R4       ;TABLE PNTR TO R4
001410  011405                          TST1AA: MOV     @R4,R5          ;STATUS CODE-R5
001412  004767  004336                          JSR     R7,TYP          ;TRANSMIT R5
001416  004767  004370                          JSR     R7,DELAYA       ;DELAY W/O OPTION
001422  004767  004342                          JSR     R7, DELAY       ;DELAY AWHILE
001426  005724                                  TST     (R4)+           ;END OF TABLE?
001430  001367                                  BNE     TST1AA          ;BR IF NOT DONE
001432  004767  004404                  TST1AP: JSR     R7, SCOP        ;CHECK SWITCH SETTINGS
001436  000760                                  BR      TST1A           ;LOOP ON TST1A
001440  000000                                  HALT                    ;NORMAL END OF TEST HALT
001442  000773                                  BR      TST1AP          ;CHECK SWITCH SETTINGS AGAIN
001444  000422                                  BR      TST1B           ;DO NEXT TEST
                    
                                                ;RTO1 STATUS INDICATOR TABLE
                    
001446  000320                          TABR:   320
001450  000321                                  321
001452  000322                                  322
001454  000324                                  324
001456  000330                                  330
001460  000323                                  323
001462  000326                                  326
001464  000334                                  334
001466  000331                                  331
001470  000325                                  325
001472  000332                                  332
001474  000327                                  327
001476  000335                                  335
001500  000333                                  333
001502  000336                                  336
001504  000337                                  337
001506  000320                                  320
```

```
   743  001510  000000                          0
   744
   745
```

```
                                        ;TST1B RTU1 DISPLAY TEST
                                        ;OUTPUTS ASCII CODES 60-72. 100,56 12 TIMES EACH

001512  012706  001200        TST1B:  MOV     #BUFF,R6        ;SETUP STACK BUFFER
001516  016700  177460                MOV     CNT,R0          ;NO. OF NIXIES
001522  012704  000060                MOV     #60,R4          ;1ST CHAR-R4
001526  012705  000100                MOV     #100,R5         ;CLEAR DISPLAY CODE
001532  004767  004216                JSR     R7, TYP         ;TRANSMIT R5
001536  012705  000056        TST1BA: MOV     #56,R5          ;DEC PT ASCII-R5
001542  004767  004206                JSR     R7,TYP          ;TRANSMIT R5
001546  010405                TST1BB: MOV     R4,R5           ;CHAR-R5
001550  004767  004200                JSR     R7,TYP          ;TRANSMIT R5
001554  005300                        DEC     R0              ;DEC. DISPLAY CNT
001556  001373                        BNE     TST1BB          ;BR IF NOT DONE
001560  004767  004204                JSR     R7, DELAY       ;DELAY AWHILE
001564  016700  177412                MOV     CNT,R0          ;NO. OF NIXIES
001570  005204                        INC     R4
001572  122704  000072                CMPB    #72,R4          ;GET NEXT CHAR.
001576  001405                        BEQ     TST1BC          ;BR IF 9'S BEING DISPLAYED
001600  012705  000100                MOV     #100,R5         ;CLEAR DISPLAY
001604  004767  004144                JSR     R7,TYP          ;TRANSMIT R5
001610  000752                        BR      TST1BA          ;DISPLAY NEXT CHAR
001612  016700  177364        TST1BC: MOV     CNT,R0          ;GET NO. OF NIXIES.
001616  012705  000060        1$:     MOV     #60,R5          ;ASCII CODE FOR 0
001622  004767  004126                JSR     R7,TYP          ;SEND R5
001626  005300                        DEC     R0              ;DONE?
001630  001372                        BNE     1$              ;NO,  CONTINUE
001632  016700  177344                MOV     CNT,R0          ;GET NO.  OF NIXIES
001636  012705  000040        2$:     MOV     #40,R5          ;ASCII FOR SPACE
001642  004767  004106                JSR     R7,TYP          ;SEND SPACE
001646  005300                        DEC     R0              ;DONE?
001650  001372                        BNE     2$              ;NO,  CONTINUE
001652  012705  000100                MOV     #100,R5         ;CLEAR DISPLAY
001656  012700  000015                MOV     #15,R0          ;SET DECIMAL PT.  COUNT
001662  012705  000056        3$:     MOV     #56,R5          ;DECIMAL PT.  CODE
001666  004767  004062                JSR     R7,TYP          ;SEND DECIMAL POINT
001672  004767  004072                JSR     R7,DELAY        ;DELAY AWHILE
001676  005300                        DEC     R0              ;DONE?
001700  001370                        BNE     3$              ;NO,  CONTINUE
001702  012706  001200        TST1BD: MOV     #BUFF,R6        ;SETUP STACK BUFFER
001706  012705  000100        TST1BF: MOV     #100,R5         ;CLEAR DISPLAY CODE
001712  004767  004036                JSR     R7,TYP          ;TRANSMIT R5
001716  012704  000060                MOV     #60,R4          ;ASCII 0-R4
001722  012705  000056                MOV     #56,R5          ;DEC. POINT TO R5
001726  004767  004022                JSR     R7,TYP          ;TRANSMIT R5
001732  010405                TST1BE: MOV     R4,R5           ;CHAR-R5
001734  004767  004014                JSR     R7,TYP          ;TRANSMIT R5
001740  004767  004024                JSR     R7, DELAY       ;DELAY AWHILE
001744  005204                        INC     R4
001746  122704  000073                CMPB    #73,R4          ;GET NEXT CHAR
001752  001367                        BNE     TST1BE          ;BR IF NOT SPACE CHAR
001754  005737  000042                TST     @#42            ;CHECK LOAD BY ACT11
001760  001401                        BEQ     .+4             ;BR. IF NOT LOADED BY ACT11
001762  000167  004752                JMP     ACT             ;JMP TO ACT11 ROUTINE
001766  004767  004050        TS1BP:  JSR     R7, SCOP        ;CHECK SWITCH SETTINGS
001772  000647                        BR      TST1B           ;LOOP ON TST1B
```

```
                                        HALT                    ;NORMAL END OF TEST
001776  000773                          BR      TS1BP           ;CHECK SWITCH SETTINGS AGAIN


                                ;TST1C RTOJ OCTAL KYBD TEST
                                ;OCTAL EQUIV OF KEY PRESSED WILL BE DISPLAYED
                                ;INTERRUPT MODE IS USED

002000  012700  000020          TST1C:  MOV     #20,R0          ;SETUP CHAR CNT
002004  012706  001200          TST1CA: MOV     #BUFF,R6        ;SETUP STACK BUFFER
002010  012777  002036  177200          MOV     #TST1CB,@TKVTR  ;SETUP INT. VECTOR RETURN
002016  005077  177176                  CLR     @TKLVL          ;CLR INT. PRIORITY
002022  005077  177164                  CLR     @TKB            ;CLR DONE FLAG READER
002026  012777  000100  177160          MOV     #100,@TKS       ;ENABLE KYBD INTERRUPT
002034  000001                          WAIT                    ;WAIT FOR KEY PRESSED
002036  005077  177152          TST1CB: CLR     @TKS            ;DISABLE KYBD INTERRUPT
002042  017704  177144                  MOV     @TKB,R4         ;READ CHAR
002046  010405                          MOV     R4,R5
002050  042705  177770                  BIC     #177770,R5      ;SAVE BITS 2-0
002054  062705  000060                  ADD     #60,R5          ;MAKE ASCII
002060  010503                          MOV     R5,R3           ;R3=3RD DIGIT
002062  010405                          MOV     R4,R5           ;GET CHAR AGAIN
002064  006005                          ROR     R5              ;SHIFT REMAINING,
002066  006005                          ROR     R5              ;2 DIGITS,
002070  006005                          ROR     R5              ;INTO BITS 5-0.
002072  010504                          MOV     R5,R4           ;SAVE DIGITS AWHILE
002074  042705  177770                  BIC     #177770,R5      ;SAVE 2ND DIGIT
002100  062705  000060                  ADD     #60,R5          ;MAKE ASCII
002104  010502                          MOV     R5,R2           ;R2=2ND DIGIT
002106  006004                          ROR     R4              ;SHIFT
002110  006004                          ROR     R4              ;1ST DIGIT,
002112  006004                          ROR     R4              ;TO BITS 2-0
002114  042704  177770                  BIC     #177770,R4      ;SAVE 1ST DIGIT
002120  062704  000060                  ADD     #60,R4          ;MAKE ASCII
002124  012705  000100                  MOV     #100,R5         ;CLEAR DISPLAY CODE
002130  004767  003620                  JSR     R7,TYP          ;TRANSMIT R5
002134  010405                          MOV     R4,R5           ;1ST DIGIT-R5
002136  004767  003612                  JSR     R7,TYP          ;TRANSMIT R5
002142  010205                          MOV     R2,R5           ;2ND DIGIT-R5
002144  004767  003604                  JSR     R7,TYP          ;TRANSMIT R5
002150  010305                          MOV     R3,R5           ;3RD DIGIT TO R5
002152  004767  003576                  JSR     R7,TYP          ;TRANSMIT R5
002156  005300                          DEC     R0              ;DEC CNT
002160  001311                          BNE     TST1CA          ;BR. IF NOT DONE
002162  004767  003624                  JSR     R7,DELAYA       ;DELAY NO SWITCH OPTION
002166  004767  003620                  JSR     R7,DELAYA       ;DELAY NO SWITCH OPTION
002172  004767  003644          TS1CP:  JSR     R7,SCOP         ;CHECK SWITCH SETTINGS
002176  000700                          BR      TST1C           ;LOOP ON TEST
002200  000000                          HALT                    ;NORMAL END OF TEST HALT
002202  000773                          BR      TS1CP           ;CHECK SWITCH SETTINGS AGAIN
002204  000167  177302                  JMP     TST1B           ;RESTART TST1A
```

```
                                        ;TST2A RTU2-A DISPLAY TEST
                                
002210  012706  001200          TST2A:  MOV     #BUFF,R6        ;SETUP STACK BUFFER
002214  004467  003754                  JSR     R4,PRINTM       ;PRINT MESSAGE
002220  006212                          MSG1                    ;"TST2A DISPLAY TEST"
002222  004767  003564                  JSR     R7,DELAYA       ;DELAY NO SWITCH OPTIONS
002226  004767  003560                  JSR     R7,DELAYA       ;DELAY NO SWITCH OPTIONS
002232  012705  000012                  MOV     #12,R5          ;CLR RTO2 CODE
002236  004767  003512                  JSR     R7,TYP          ;TRANSMIT R5
002242  012706  001200          TST2AC: MOV     #BUFF,R6        ;SETUP STACK BUFFER
002246  012700  000040                  MOV     #40,R0          ;32 COUNT
002252  012704  000040                  MOV     #40,R4          ;1ST ASCII CHAR.
002256  012705  000012                  MOV     #12,R5          ;CLEAR DISPLAY-R5
002262  004767  003466                  JSR     R7,TYP          ;TRANSMIT R5
002266  010405                  TST2AD: MOV     R4,R5           ;CHAR TO R5
002270  004767  003460                  JSR     R7,TYP          ;TRANSMIT R5
002274  004767  003470                  JSR     R7, DELAY       ;DELAY AWHILE
002300  005300                          DEC     R0
002302  001020                          BNE     TST2AE          ;BR. IF NOT DONE 32
002304  012705  000016                  MOV     #16,R5          ;BLANK DISPLAY-R5
002310  004767  003440                  JSR     R7,TYP          ;TRANSMIT R5
002314  004767  003450                  JSR     R7,DELAY        ;DELAY AWHILE
002320  012705  000017                  MOV     #17,R5          ;UNBLANK DISPLAY -R5
002324  004767  003424                  JSR     R7,TYP          ;TRANSMIT R5
002330  004767  003434                  JSR     R7,DELAY        ;DELAY AWHILE
002334  012705  000012                  MOV     #12,R5          ;CLEAR DISPLAY CODE
002340  004767  003410                  JSR     R7,TYP          ;TRANSMIT R5
002344  005204                  TST2AE: INC     R4              ;GET NEXT CHAR
002346  122704  000140                  CMPB    #140,R4         ;GET NEXT CHAR.
002352  001345                          BNE     TST2AD          ;BR IF NOT DONE
002354  012704  000040                  MOV     #40,R4          ;1ST CHAR
002360  012705  000012          TST2AA: MOV     #12,R5          ;CLR DISPLAY CODE-R5
002364  004767  003364                  JSR     R7,TYP          ;TRANSMIT R5
002370  012700  000040                  MOV     #40,R0          ;INIT 32 COUNT
002374  010405                  TST2AB: MOV     R4,R5           ;CHAR-R5
002376  004767  003352                  JSR     R7,TYP          ;TRANSMIT R5
002402  005300                          DEC     R0              ;DEC. 32 CNT
002404  001373                          BNE     TST2AB          ;BR IF NOT DONE
002406  004767  003356                  JSR     R7,DELAY        ;DELAY AWHILE
002412  005204                          INC     R4              ;INC TO NEXT CHAR
002414  122704  000140                  CMPB    #140,R4         ;DONE CHECK
002420  001357                          BNE     TST2AA          ;BR. IF NOTDONE
002422  004767  003364                  JSR     R7,DELAYA       ;DELAY W/O OPTIONS
002426  004767  003336                  JSR     R7,DELAY        ;DELAY AWHILE
002432  005737  000042                  TST     @#42            ;CHECK LOAD BY ACT11
002436  001402                          BEQ     .+6             ;BR. IF NOT LOADED BY ACT11
002440  000167  004274                  JMP     ACT             ;JMP TO ACT11 ROUTINE
002444  004767  003372          TS2AP:  JSR     R7, SCOP        ;CHECK SWITCH SETTINGS
002450  000657                          BR      TST2A           ;LOOP ON TST23
002452  000000                          HALT                    ;NORMAL END OF TEST HALT
002454  000773                          BR      TS2AP           ;CHECK SWITCH SETTINGS AGAIN
```

```
                                        ;TST2B RTU2-A OCTAL KEYBOARD TEST.
                                        ;OCTAL EQUIV OF KEY PRESSED WILL BE DISPLAYED
                                        ;INTERRUPT MODE IS USED

002456  012700  000036          TST2B:  MOV     #36,R0          ;SETUPT CHAR CNT
002462  004467  003506                  JSR     R4,PRINTM       ;PRINT MESSAGE
002466  006246                          MSG2                    ;"TST2B OCTAL KEYBOARD TEST"
002470  012706  001200          TST2BA: MOV     #BUFF,R6        ;SETUP STACK BUFFER
002474  012777  002516  176514          MOV     #TST2BB,@TKVTR  ;SETUP INT. VECTOR RETURN
002502  005077  176512                  CLR     @TKLVL          ;CLR INT. PRIORITY
002506  012777  000100  176500          MOV     #100,@TKS       ;ENABLE KYBD INTERRUPT
002514  000001                          WAIT                    ;WAIT FOR KEY PRESSED
002516  005077  176472          TST2BB: CLR     @TKS            ;DISABLE KYBD INTERRUPT
002522  012705  000012                  MOV     #12,R5          ;CLEAR DISPLAY CODE
002526  004767  003222                  JSR     R7,TYP
002532  017704  176454                  MOV     @TKB,R4         ;READ CHAR
002536  010405                          MOV     R4,R5           ;CHAR TO R5
002540  042705  177770                  BIC     #177770,R5      ;SAVE BITS 2-0
002544  062705  000060                  ADD     #60,R5          ;MAKE ASCII
002550  010503                          MOV     R5,R3           ;R3=3RD DIGIT
002552  010405                          MOV     R4,R5           ;GET CHAR AGAIN
002554  006005                          ROR     R5              ;SHIFT REMAINING,
002556  006005                          ROR     R5              ;2 DIGITS,
002560  006005                          ROR     R5              ;INTO BITS 5-0.
002562  010504                          MOV     R5,R4           ;SAVE DIGITS AWHILE
002564  042705  177770                  BIC     #177770,R5      ;SAVE 2ND DIGIT
002570  062705  000060                  ADD     #60,R5          ;MAKE ASCII
002574  010502                          MOV     R5,R2           ;R2=2ND DIGIT
002576  006004                          ROR     R4              ;SHIFT,
002600  006004                          ROR     R4              ;1ST DIGIT,
002602  006004                          ROR     R4              ;TO BITS 2-0
002604  042704  177770                  BIC     #177770,R4      ;SAVE 1ST DIGIT
002610  062704  000060                  ADD     #60,R4          ;MAKE ASCII
002614  010405                          MOV     R4,R5           ;1ST DIGIT-R5
002616  004767  003132                  JSR     R7,TYP          ;TRANSMIT R5
002622  010205                          MOV     R2,R5           ;2ND DIGIT-R5
002624  004767  003124                  JSR     R7,TYP          ;TRANSMIT R5
002630  010305                          MOV     R3,R5           ;3RD DIGIT TO R5
002632  004767  003116                  JSR     R7,TYP          ;TRANSMIT R5
002636  005300                          DEC     R0              ;DEC. CHAR CNT
002640  001313                          BNE     TST2BA          ;BR IF NOT DONE
002642  004767  003174          TS2BP:  JSR     R7,SCOP         ;CHECK SWITCH SETTINGS
002646  000703                          BR      TST2B           ;LOOP ON TEST
002650  000000                          HALT                    ;NORMAL END OF TEST HALT
002652  000773                          BR      TS2BP           ;CHECK SWITCH SETTINGS AGAIN
002654  000167  177330                  JMP     TST2A           ;RESTART RTO2 TESTS
```

```
                                        ;TSBAA  RT02-B  AUDIO ALARM TEST
                                        ;ALARM CHECK
                                        
002660  012706  001200          TSBAA:  MOV     #BUFF,R6        ;SETUP STACK BUFFER
002664  004467  003304                  JSR     R4,PRINTM       ;PRINT MESSAGE
002670  006311                          MSG3                    ;"TSBAA RT02-B AUDIO ALARM TEST"
002672  004767  003114                  JSR     R7,DELAYA       ;DELAY AWHILE
002676  004767  003110                  JSR     R7,DELAYA       ;DELAY AWHILE
002702  012705  000012                  MOV     #12,R5          ;CLEAR DISPLAY
002706  004767  003042                  JSR     R7,TYP
002712  004767  003074                  JSR     R7,DELAYA       ;DELAY AWHILE
002716  012702  000003                  MOV     #3,R2           ;CHAR COUNT TO SEND 3 BELLS
002722  012705  000052          1$:     MOV     #52,R5          ;* TO R5
002726  004767  003022                  JSR     R7,TYP          ;TRANSMIT *
002732  012705  000007                  MOV     #7,R5           ;BELL CODE TO R5
002736  004767  003012                  JSR     R7,TYP          ;TRANSMIT BELL
002742  004767  003044                  JSR     R7,DELAYA       ;DELAY AWHILE
002746  004767  003040                  JSR     R7,DELAYA       ;DELAY AWHILE
002752  004767  003034                  JSR     R7,DELAYA       ;DELAY AWHILE
002756  004767  003030                  JSR     R7,DELAYA       ;DELAY AWHILE
002762  004767  003024                  JSR     R7,DELAYA       ;DELAY AWHILE
002766  005302                          DEC     R2              ;DECREMENT CHAR, COUNT
002770  001354                          BNE     1$              ;BRANCH IF NOT DONE
                                        
                                        
                                        ;TSBAA2 - TEST DECRIPTIONS
                                        
                                        ;PASS 1 = SENDS 27 ZEROS THEN BELL SOUNDS AS * IS SENT.
                                        ;PASS 2 = CLEAR DISPLAY & REPEAT (27 ZEROS & BELL SOUNDS AS * SENT)
                                        ;PASS 3 = DOES NOT CLEAR DISPLAY
                                        ;         SENDS 31 ONES TO CYCLE DISPLAY
                                        ;         BELL SHOULD RING AS * IS SENT
                                        ;PASS 4 = CLEARS DISPLAY AND SENDS ALL CONTROL CHARACTERS.
                                        ;         THEN SENDS 27 ZEROS AND THE BELL SHOULD SOUND AS * IS SENT.
                                        
002772  005000                  TSBAA2: CLR     R0              ;CLEAR PASS COUNT
002774  012705  000012          1$:     MOV     #12,R5          ;CLEAR DISPLAY
003000  004767  002750                  JSR     R7,TYP
003004  012702  000033          2$:     MOV     #33,R2          ;CHAR.  COUNT FOR 27 ZEROS
003010  000405                          BR      4$
003012  012702  000037          3$:     MOV     #37,R2          ;CHAR.  COUNT FOR 31 ONES
003016  012705  000061                  MOV     #61,R5          ;ONE CODE TO R5
003022  000402                          BR      5$
003024  012705  000060          4$:     MOV     #60,R5          ;ZERO CODE TO R5
003030  004767  002720          5$:     JSR     R7,TYP          ;SEND CHAR
003034  005302                          DEC     R2              ;DECREMENT CHAR.  COUNT
003036  001374                          BNE     5$              ;BRANCH IF NOT DONE
003040  012705  000052                  MOV     #52,R5          ;*CODE TO R5
003044  004767  002704                  JSR     R7,TYP          ;SEND *
003050  004767  002736                  JSR     R7,DELAYA       ;DELAY AWHILE
003054  005200                          INC     R0              ;INCREMENT PASS COUNT
003056  022700  000001                  CMP     #1,R0           ;FIRST PASS?
003062  001744                          BEQ     1$              ;CLEAR DISPLAY AND REPEAT
003064  022700  000002                  CMP     #2,R0           ;SECOND PASS?
003070  001750                          BEQ     3$              ;DON'T CLEAR DISPLAY, CYCLE DISPLAY
                                                                ;AND CONTINUE
```

```
1007  003072  022700  000004            CMP     #4,R0           ;FOURTH PASS?
1008  003076  001424                    BEQ     TSB2AP          ;CHECK SWITCH OPTIONS
1009  003100  012705  000012            MOV     #12,R5          ;CLEAR DISPLAY
1010  003104  004767  002644            JSR     R7,TYP
1011  003110  005005                    CLR     R5              ;CONTROL CHAR.  CODE 0 TO R5
1012  003112  004767  002636    6$:     JSR     R7,TYP          ;SEND CHAR.
1013  003116  005205                    INC     R5              ;NEXT CONTROL CHAR.  CODE TO R5
1014  003120  022705  000007            CMP     #7,R5           ;BELL CODE?
1015  003124  001001                    BNE     8$              ;BRANCH IF NOT BELL CODE
1016  003126  005205            7$:     INC     R5              ;INCREMENT R5
1017  003130  022705  000012    8$:     CMP     #12,R5          ;DISPLAY CLEAR CODE?
1018  003134  001001                    BNE     9$              ;BRANCH IF NOT DISPLAY CLEAR CODE
1019  003136  005205                    INC     R5              ;INC R5 IF DISPLAY CLEAR CODE
1020  003140  022705  000037    9$:     CMP     #37,R5          ;FINISHED?
1021  003144  001362                    BNE     6$              ;BRANCH IF NOT FINISHED
1022  003146  000716                    BR      2$              ;BRANCH TO SEND ZEROS
1023                                                            ;THEN BELL SHOULD SOUND WHEN * IS SENT
1024  003150  004767  002636    TSB2AP: JSR     R7,DELAYA       ;DELAY AWHILE
1025  003154  004767  002632            JSR     R7,DELAYA       ;DELAY AWHILE
1026  003160  004767  002656            JSR     R7,SCOP         ;CHECK SWITCH SETTINGS
1027  003164  000635                    BR      TSBAA           ;LOOP ON TEST
1028  003166  000000                    HALT                    ;NORMAL END OF TEST HALT
1029  003170  000767                    BR      TSB2AP          ;CHECK SWITCH SETTINGS AGAIN
1030
1031
1032
1033                            ;TSB2A RT02-B DISPLAY TEST
1034
1035  003172  004467  002776    TSB2A:  JSR     R4,PRINTM       ;PRINT MESSAGE
1036  003176  001351                    MSG4                    ;"TSB2A DISPLAY TEST"
1037  003200  004767  002606            JSR     R7,DELAYA       ;DELAY NO SWITCH OPTIONS
1038  003204  004767  002602            JSR     R7,DELAYA       ;DELAY AWHILE
1039  003210  012705  000012            MOV     #12,R5          ;CLR RT02 CODE
1040  003214  004767  002534            JSR     R7,TYP          ;TRANSMIT R5
1041  003220  012706  001200    TSB2AC: MOV     #BUFF,R6        ;SETUP STACK BUFFER
1042  003224  012700  000040            MOV     #40,R0          ;32 COUNT
1043  003230  012704  000040            MOV     #40,R4          ;1ST ASCII CHAR.
1044  003234  012705  000012            MOV     #12,R5          ;CLEAR DISPLAY-R5
1045  003240  010405            TSB2AD: MOV     R4,R5           ;CHAR TO R5
1046  003242  004767  002506            JSR     R7,TYP          ;TRANSMIT R5
1047  003246  004767  002516            JSR     R7,DELAY        ;DELAY A WHILE
1048  003252  005300                    DEC     R0
1049  003254  001020                    BNE     TSB2AE          ;BR.  IF NOT DONE 32
1050  003256  012705  000016            MOV     #16,R5          ;BLANK DISPLAY-R5
1051  003262  004767  002466            JSR     R7,TYP          ;TRANSMIT R5
1052  003266  004767  002476            JSR     R7,DELAY        ;DELAY AWHILE
1053  003272  012705  000017            MOV     #17,R5          ;UNBLANK DISPLAY -R5
1054  003276  004767  002452            JSR     R7,TYP          ;TRANSMIT R5
1055  003302  004767  002462            JSR     R7,DELAY        ;DELAY AWHILE
1056  003306  012705  000012            MOV     #12,R5          ;CLEAR DISPLAY CODE
1057  003312  004767  002436            JSR     R7,TYP          ;TRANSMIT R5
1058  003316  005204            TSB2AE: INC     R4              ;GET NEXT CHAR
1059  003320  122704  000140            CMPB    #140,R4         ;DONE?
1060  003324  001345                    BNE     TSB2AD          ;BR IF NOT DONE
1061  003326  012704  000040            MOV     #40,R4          ;1ST CHAR
1062  003332  012705  000012    TSB2AA: MOV     #12,R5          ;CLR DISPLAY CODE-R5
```

```
 1063  003336  004767  002412              JSR     R7,TYP          ;TRANSMIT R5
 1064  003342  012700  000040              MOV     #40,R0          ;CHAR COUNT TO 32
 1065  003346  010405              TSB2AB: MOV     R4,R5           ;CHAR-R5
 1066  003350  004767  002400              JSR     R7,TYP          ;TRANSMIT R5
 1067  003354  005300                      DEC     R0              ;DEC.  32 CNT
 1068  003356  001373                      BNE     TSB2AB          ;BR IF NOT DONE
 1069  003360  004767  002404              JSR     R7,DELAY        ;DELAY AWHILE
 1070  003364  005204                      INC     R4              ;INC TO NEXT CHAR
 1071  003366  122704  000140              CMPB    #140,R4         ;DONE CHECK
 1072  003372  001357                      BNE     TSB2AA          ;BR.  IF NOT DONE
 1073  003374  004767  002412              JSR     R7,DELAYA       ;DELAY W/O OPTIONS
 1074  003400  004767  002364              JSR     R7,DELAY        ;DELAY AWHILE
 1075  003404  005737  000042              TST     @#42            ;CHECK LOAD BY ACT11
 1076  003410  001402                      BEQ     .+6             ;BR.  IF NOT LOADED BY ACT11
 1077  003412  000167  003322              JMP     ACT             ;JMP TO ACT11 ROUTINE
 1078  003416  004767  002420  TB2AP:     JSR     R7,SCOP         ;CHECK SWITCH SETTINGS
 1079  003422  000663                      BR      TSB2A           ;LOOP ON TSB2A
 1080  003424  000000                      HALT                    ;NORMAL END OF TEST HALT
 1081  003426  000773                      BR      TB2AP           ;CHECK SWITCH SETTING AGAIN
 1082
 1083
 1084
 1085                          ;TSB3   RT02-B KEYBOARD OUTPUT TEST
 1086
 1087  003430  004467  002540  TSB3:      JSR     R4,PRINTM       ;PRINT MESSAGE
 1088  003434  006405                      MSG5                    ;"TSB3 RT02 KYBD OUTPUT TEST"
 1089  003436  005000                      CLR     R0              ;SET TABLE POINTER
 1090  003440  012706  001200  TSB3A:     MOV     #BUFF,R6        ;SETUP STACK BUFFER
 1091  003444  012777  003466  175544      MOV     #TSB3B,@TKVTR   ;SETUP INT. VECTOR RETURN
 1092  003452  005077  175542              CLR     @TKLVL          ;CLR INT. PRIORITY
 1093  003456  012777  000100  175530      MOV     #100,@TKS       ;ENABLE KYBD INTERRUPT
 1094  003464  000001                      WAIT
 1095  003466  005077  175522  TSB3B:     CLR     @TKS            ;DISABLE KYBD INTERRUPT
 1096  003472  017704  175514              MOV     @TKB,R4         ;READ CHAR.
 1097  003476  126004  004026              CMPB    CODE(R0),R4     ;CHECK CHAR.
 1098  003502  001020                      BNE     ERROR           ;BRANCH IF ERROR
 1099  003504  012705  000012              MOV     #12,R5          ;CLEAR DISPLAY
 1100  003510  004767  002240              JSR     R7,TYP
 1101  003514  020427  000240              CMP     R4,#240         ;ASCII CODE >240?
 1102  003520  002403                      BLT     TSB3BB          ;NO,  BRANCH
 1103  003522  010405                      MOV     R4,R5           ;MOVE CHAR TO R5
 1104  003524  004767  002224              JSR     R7,TYP          ;TRANSMIT CHAR
 1105  003530  005200  TSB3BB:    INC     R0              ;INCREMENT TABLE POINTER
 1106  003532  105760  004026              TSTB    CODE(R0)        ;CHECK TABLE DATA
 1107  003536  001340                      BNE     TSB3A           ;BRANCH IF NOT FINISHED
 1108  003540  000167  000532              JMP     TSB3C           ;JUMP IF FINISHED
 1109
```

```
1110  003544  012705  000012          ERROR:  MOV     #12,R5          ;CLEAR DISPLAY
1111  003550  004767  002200                  JSR     R7,TYP
1112  003554  012705  000007                  MOV     #7,R5           ;BELL CODE
1113  003560  004767  002170                  JSR     R7,TYP          ;SEND BELL CODE
1114  003564  010401                          MOV     R4,R1           ;STORE CHAR IN R1
1115  003566  004467  002402                  JSR     R4,PRINTM       ;PRINT MESSAGE
1116  003572  003776                          RMSG                    ;"RECEIVED  "
1117  003574  010105                          MOV     R1,R5           ;CHAR CODE TO R5
1118  003576  020527  000240                  CMP     R5,#240         ;CODE >240?
1119  003602  002002                          BGE     1$              ;YES,  TYPE CHAR
1120  003604  012705  000240                  MOV     #240,R5         ;NO,  TYPE SPACE
1121  003610  004767  002140          1$:     JSR     R7,TYP          ;TYPE CHAR OR SPACE
1122  003614  012705  000075                  MOV     #75,R5          ;ASCII FOR =
1123  003620  004767  002130                  JSR     R7,TYP          ;SEND =
1124  003624  004367  000054                  JSR     R3,ASC          ;PRINT ASCII FOR CODE OF CHAR
1125  003630  004467  002340                  JSR     R4,PRINTM       ;PRINT MESSAGE
1126  003634  004011                          WMSG                    ;"    WANTED  "
1127  003636  116005  004026                  MOVB    CODE(R0),R5     ;WANTED CHAR TO R5
1128  003642  020527  000240                  CMP     R5,#240         ;CODE >240?
1129  003646  002002                          BGE     2$              ;YES,  TYPE CHAR
1130  003650  012705  000240                  MOV     #240,R5         ;NO,  TYPE SPACE
1131  003654  004767  002074          2$:     JSR     R7,TYP          ;SEND CHAR OR SPACE
1132  003660  012705  000075                  MOV     #75,R5          ;ASCII FOR =
1133  003664  004767  002064                  JSR     R7,TYP          ;SEND =
1134  003670  116001  004026                  MOVB    CODE(R0),R1     ;STORE CHAR WANTED IN R1
1135  003674  004367  000004                  JSR     R3,ASC          ;PRINT ASCII CODE
1136  003700  000167  177534                  JMP     TSB3A           ;WAIT FOR NEW INPUT
1137  003704  010105          ASC:    MOV     R1,R5           ;STORE CHAR IN R5
1138  003706  042701  177770                  BIC     #177770,R1      ;SAVE LAST DIGIT
1139  003712  062701  000260                  ADD     #260,R1         ;MAKE ASCII
1140  003716  006005                          ROR     R5
1141  003720  006005                          ROR     R5              ;GET SECOND DIGIT
1142  003722  006005                          ROR     R5
1143  003724  010502                          MOV     R5,R2           ;SAVE IN R2
1144  003726  042702  177770                  BIC     #177770,R2      ;SAVE SECOND DIGIT
1145  003732  062702  000260                  ADD     #260,R2         ;MAKE ASCII
1146  003736  006005                          ROR     R5              ;GET FIRST DIGIT
1147  003740  006005                          ROR     R5
1148  003742  006005                          ROR     R5
1149  003744  042705  177774                  BIC     #177774,R5      ;SAVE FIRST DIGIT
1150  003750  062705  000260                  ADD     #260,R5         ;MAKE ASCII
1151  003754  004767  001774                  JSR     R7,TYP          ;SEND FIRST DIGIT
1152  003760  010205                          MOV     R2,R5           ;GET SECOND DIGIT CODE
1153  003762  004767  001766                  JSR     R7,TYP          ;SEND SECOND DIGIT
1154  003766  010105                          MOV     R1,R5           ;GET LAST DIGIT
1155  003770  004767  001760                  JSR     R7,TYP          ;SEND LAST DIGIT
1156  003774  000203                          RTS     R3              ;RETURN
1157  003776  042522  042503  053111  RMSG:   .ASCIZ  'RECEIVED  '
1158  004004  042105  020040    000
1159  004011    040  020040  053440  WMSG:   .ASCIZ  '    WANTED  '
1160  004016  047101  042524  020104
1161  004024  000040
1162                                  .EVEN
1163  004026     333     261     262  CODE:   .BYTE   333,261,262,263,264,265,266,267,270,271,260,255,337
1164  004031     263     264     265
1165  004034     266     267     270
```

```
1166  004037  271  260  255
1167  004042  337
1168  004043  335  211  233        .BYTE   335,211,233,361,367,345,362,364,371,365,351,357,360
1169  004046  361  367  345
1170  004051  362  364  371
1171  004054  365  351  357
1172  004057  360
1173  004060  300  212  215        .BYTE   300,212,215,336,341,363,344,346,347,350,352,353,354
1174  004063  336  341  363
1175  004066  344  346  347
1176  004071  350  352  353
1177  004074  354
1178  004075  273  272  377        .BYTE   273,272,377,372,370,343,366,342,356,355,254,256,257
1179  004100  372  370  343
1180  004103  366  342  356
1181  004106  355  254  256
1182  004111  257
1183  004112  334  240  373        .BYTE   334,240,373,241,242,243,244,245,246,247,250.251,260
1184  004115  241  242  243
1185  004120  244  245  246
1186  004123  247  250  251
1187  004126  260
1188  004127  275  337  375        .BYTE   275,337,375,211,233,321,327,305,322,324,331,325,311
1189  004132  211  233  321
1190  004135  327  305  322
1191  004140  324  331  325
1192  004143  311
1193  004144  317  320  340        .BYTE   317,320,340,212,215,376,301,323,304,306,307,310,312
1194  004147  212  215  376
1195  004152  301  323  304
1196  004155  306  307  310
1197  004160  312
1198  004161  313  314  253        .BYTE   313,314,253,252,377,332,330,303,326,302,316,315,274
1199  004164  252  377  332
1200  004167  330  303  326
1201  004172  302  316  315
1202  004175  274
1203  004176  276  277  374        .BYTE   276,277,374,240,233,261,262,263,264,265,266,267,270
1204  004201  240  233  261
1205  004204  262  263  264
1206  004207  265  266  267
1207  004212  270
1208  004213  271  260  215        .BYTE   271,260,215,237,235,211,233,221,227,205,222,224,231
1209  004216  237  235  211
1210  004221  233  221  227
1211  004224  205  222  224
1212  004227  231
1213  004230  225  211  217        .BYTE   225,211,217,220,200,212,215,236,201,223,204,206,207
1214  004233  220  200  212
1215  004236  215  236  201
1216  004241  223  204  206
1217  004244  207
1218  004245  210  212  213        .BYTE   210,212,213,214,273,272,377,232,230,203,226,202,216
1219  004250  214  273  272
1220  004253  377  232  230
1221  004256  203  226  202
```

```
1222  004261      216
1223  004262      215     254     256         .BYTE   215,254,256,257,234,240
1224  004265      257     234     240
1225  004270      000                         .BYTE   0
1226          004272                        .EVEN
1227  004272  000167  177132        TSB3D:  JMP     TSB3            ;LOOP ON TEST
1228  004276  004767  001510        TSB3C:  JSR     R7,DELAYA       ;DELAY W/O OPTIONS
1229  004302  004767  001462                JSR     R7,DELAY        ;DELAY AWHILE
1230  004306  004767  001530        TSB3CC: JSR     R7,SCOP         ;CHECK SWITCH SETTINGS
1231  004312  000767                        BR      TSB3D           ;LOOP ON TEST TSB3
1232  004314  000000                        HALT                    ;NORMAL END OF TEST
1233  004316  000773                        BR      TSB3CC          ;CHECK SWITCH SETTINGS AGAIN
1234  004320  000167  176334                JMP     TSBAA           ;RESTART RT02-B TEST
1235
1236
1237
1238                                ;TSCAA RT02-C DISPLAY TEST
1239
1240  004324  012706  001200        TSCAA:  MOV     #BUFF,R6        ;SETUP STACK BUFFER
1241  004330  004467  001640                JSR     R4,PRINTM       ;PRINT MESSAGE
1242  004334  006444                        MSG6                    ;"TSCAA RT02-C  DISPLAY TEST"
1243  004336  004767  001450                JSR     R7,DELAYA       ;DELAY NO SWITCH OPTIONS
1244  004342  004767  001444                JSR     R7,DELAYA       ;DELAY NO SWITCH OPTIONS
1245  004346  012705  000012                MOV     #12,R5          ;CLR RT02 CODE
1246  004352  004767  001376                JSR     R7,TYP          ;TRANSMIT R5
1247  004356  012706  001200        TSC2AC: MOV     #BUFF,R6        ;SETUP STACK BUFFER
1248  004362  012700  000040                MOV     #40,R0          ;32 COUNT
1249  004366  012704  000040                MOV     #40,R4          ;1ST ASCII CHAR.
1250  004372  012705  000012                MOV     #12,R5          ;CLEAR DISPLAY-R5
1251  004376  004767  001352                JSR     R7,TYP          ;TRANSMIT R5
1252  004402  010405                TSC2AD: MOV     R4,R5           ;CHAR TO R5
1253  004404  004767  001344                JSR     R7,TYP          ;TRANSMIT R5
1254  004410  004767  001354                JSR     R7, DELAY       ;DELAY AWHILE
1255  004414  005300                        DEC     R0
1256  004416  001020                        BNE     TSC2AE          ;BR. IF NOT DONE 32
1257  004420  012705  000016                MOV     #16,R5          ;BLANK DISPLAY-R5
1258  004424  004767  001324                JSR     R7,TYP          ;TRANSMIT R5
1259  004430  004767  001334                JSR     R7,DELAY        ;DELAY AWHILE
1260  004434  012705  000017                MOV     #17,R5          ;UNBLANK DISPLAY -R5
1261  004440  004767  001310                JSR     R7,TYP          ;TRANSMIT R5
1262  004444  004767  001320                JSR     R7,DELAY        ;DELAY AWHILE
1263  004450  012705  000012                MOV     #12,R5          ;CLEAR DISPLAY CODE
1264  004454  004767  001274                JSR     R7,TYP          ;TRANSMIT R5
1265
1266  004460  005204                TSC2AE: INC     R4              ;GET NEXT CHAR
1267  004462  122704  000140                CMPB    #140,R4         ;GET NEXT CHAR.
1268  004466  001345                        BNE     TSC2AD          ;BR IF NOT DONE
1269  004470  012704  000040                MOV     #40,R4          ;1ST CHAR
1270  004474  012705  000012        TSC2AA: MOV     #12,R5          ;CLR DISPLAY CODE-R5
1271  004500  004767  001250                JSR     R7,TYP          ;TRANSMIT R5
1272  004504  012700  000040                MOV     #40,R0          ;IHIT 32 COUNT
1273  004510  010405                TSC2AB: MOV     R4,R5           ;CHAR-R5
1274  004512  004767  001236                JSR     R7,TYP          ;TRANSMIT R5
1275  004516  005300                        DEC     R0              ;DEC. 32 CNT
1276  004520  001373                        BNE     TSC2AB          ;BR IF NOT DONE
1277  004522  004767  001242                JSR     R7,DELAY        ;DELAY AWHILE
```

```
1278  004526  005204                          INC     R4              ;INC TO NEXT CHAR
1279  004530  122704  000140                  CMPB    #140,R4         ;DONE CHECK
1280  004534  001357                          BNE     TSC2AA          ;BR. IF NOTDONE
1281  004536  004767  001250                  JSR     R7,DELAYA       ;DELAY W/O OPTIONS
1282  004542  004767  001222                  JSR     R7,DELAY        ;DELAY AWHILE
1283  004546  005737  000042                  TST     @#42            ;CHECK LOAD BY ACT11
1284  004552  001402                          BEQ     .+6             ;BR. IF NOT LOADED BY ACT11
1285  004554  000167  002160                  JMP     ACT             ;JMP TO ACT11 ROUTINE
1286  004560  004767  001256          TC2AP:  JSR     R7, SCOP        ;CHECK SWITCH SETTINGS
1287  004564  000657                          BR      TSCAA           ;LOOP ON TSCAA
1288  004566  000000                          HALT                    ;NORMAL END OF TEST HALT
1289  004570  000773                          BR      TC2AP           ;CHECK SWITCH SETTINGS AGAIN
1290
1291
1292                                  ;TSC2A RT02-C OCTAL KEYBOARD TEST.
1293                                  ;OCTAL EQUIV OF KEY PRESSED WILL BE DISPLAYED
1294                                  ;INTERRUPT MODE IS USED
1295
1296  004572  012700  000036          TSC2A:  MOV     #36,R0          ;SETUPT CHAR CNT
1297  004576  004467  001372                  JSR     R4,PRINTM       ;PRINT MESSAGE
1298  004602  006500                          MSG7                    ;"TSC2A OCTAL KEYBOARD TEST"
1299  004604  012705  000006                  MOV     #6,R5           ;ENABLE KYBRD ON RT02-C
1300  004610  004767  001140                  JSR     R7,TYP
1301  004614  012706  001200                  MOV     #BUFF,R6        ;SETUP STACK BUFFER
1302  004620  012777  004642  174370          MOV     #TSC2BC,@TKVTR  ;SET INTERRUPT RETURN VECTOR
1303  004626  005077  174366                  CLR     @TKLVL          ;CLR INT. PRIORITY
1304  004632  012777  000100  174354          MOV     #100,@TKS       ;ENABLE KYBRD INTERRUPT
1305  004640  000001                          WAIT
1306  004642  005077  174346          TSC2BC: CLR     @TKS            ;DISABLE KYBD INTERRUPT
1307  004646  005077  174340                  CLR     @TKB            ;CLEAR CHAR
1308  004652  012706  001200          TSC2BA: MOV     #BUFF,R6        ;SETUP STACK BUFFER
1309  004656  012777  004700  174332          MOV     #TSC2BB,@TKVTR  ;SETUP INT. VECTOR RETURN
1310  004664  005077  174330                  CLR     @TKLVL          ;CLR INT. PRIORITY
1311  004670  012777  000100  174316          MOV     #100,@TKS       ;ENABLE KYBD INTERRUPT
1312  004676  000001                          WAIT                    ;WAIT FOR KEY PRESSED
1313  004700  005077  174310          TSC2BB: CLR     @TKS            ;DISABLE KYBD INTERRUPT
1314  004704  012705  000012                  MOV     #12,R5          ;CLEAR DISPLAY CODE
1315  004710  004767  001040                  JSR     R7,TYP
1316  004714  017704  174272                  MOV     @TKB,R4         ;READ CHAR
1317  004720  010405                          MOV     R4,R5           ;CHAR TO R5
1318  004722  042705  177770                  BIC     #177770,R5      ;SAVE BITS 2-0
1319  004726  062705  000060                  ADD     #60,R5          ;MAKE ASCII
1320  004732  010503                          MOV     R5,R3           ;R3=3RD DIGIT
1321  004734  010405                          MOV     R4,R5           ;GET CHAR AGAIN
1322  004736  006005                          ROR     R5              ;SHIFT REMAINING
1323  004740  006005                          ROR     R5              ;2 DIGITS,
1324  004742  006005                          ROR     R5              ;INTO BITS 5-0.
1325  004744  010504                          MOV     R5,R4           ;SAVE DIGITS AWHILE
1326  004746  042705  177770                  BIC     #177770,R5      ;SAVE 2ND DIGIT
1327  004752  062705  000060                  ADD     #60,R5          ;MAKE ASCII
1328  004756  010502                          MOV     R5,R2           ;R2=2ND DIGIT
1329  004760  006004                          ROR     R4              ;SHIFT,
1330  004762  006004                          ROR     R4              ;1ST DIGIT,
1331  004764  006004                          ROR     R4              ;TO BITS 2-0
1332  004766  042704  177770                  BIC     #177770,R4      ;SAVE 1ST DIGIT
1333  004772  062704  000060                  ADD     #60,R4          ;MAKE ASCII
```

```
1334  004776  010405                          MOV     R4,R5           ;1ST DIGIT-R5
1335  005000  004767  000750                  JSR     R7,TYP          ;TRANSMIT R5
1336  005004  010205                          MOV     R2,R5           ;2ND DIGIT-R5
1337  005006  004767  000742                  JSR     R7,TYP          ;TRANSMIT R5
1338  005012  010305                          MOV     R3,R5           ;3RD DIGIT TO R5
1339  005014  004767  000734                  JSR     R7,TYP          ;TRANSMIT R5
1340  005020  005300                          DEC     R0              ;DEC. CHAR CNT
1341  005022  001313                          BNE     TSC2BA          ;BR IF NOT DONE
1342  005024  004767  001012          TC2BP:  JSR     R7,SCOP         ;CHECK SWITCH SETTINGS
1343  005030  000660                          BR      TSC2A           ;LOOP ON TEST
1344  005032  000000                          HALT                    ;NORMAL END OF TEST HALT
1345  005034  000773                          BR      TC2BP           ;CHECK SWITCH SETTINGS AGAIN
1346
1347
1348                                          ;TSC3  RT02-C BADGE READER TEST
1349                                          ;USE MAINDEC-00-DZRTA-C-CA CARDS FOR THIS TEST
1350                                          ;BADGE WILL BE READ AND INFORMATION DISPLAYED ON RT02
1351                                          ;BADGE WILL BE ACCEPTED OR REJECTED DEPENDING ON THE SETTING
1352                                          ;OF SR0 ----- IF SR0=0;  ACCEPT BADGE
1353                                          ;             IF SR0=1;  REJECT BADGE AND PROGRAM HALTS -
1354                                          ;                        PRESS CONTINUE TO REREAD BADGE
1355                                          ;                        CARD WILL BE ACCEPTED AND THEN EJECTED.
1356
1357  005036  004467  001132          TSC3:   JSR     R4,PRINTM       ;PRINT MESSAGE
1358  005042  006540                          MSG8                    ;"TSC3 RT02-C BADGE READER TEST"
1359  005044  004767  000742                  JSR     R7,DELAYA       ;DELAYA AWHILE
1360  005050  004767  000736                  JSR     R7,DELAYA
1361  005054  012705  000012                  MOV     #12,R5          ;CLEAR RT02 DISPLAY
1362  005060  004767  000670                  JSR     R7,TYP          ;TRANSMIT R5
1363  005064  000167  000376                  JMP     TSC3K           ;EJECT CARD
1364  005070  005037  005502          TSC3AA: CLR     @#LPFLG         ;CLEAR LOOP FLAG
1365  005074  005000                  TSC3A:  CLR     R0              ;CLEAR CHAR COUNT; TABLE POINTER
1366  005076  012706  001200          TSC3B:  MOV     #BUFF,R6        ;SETUP STACK BUFFER
1367  005102  012777  005124  174106          MOV     #TSC3C,@TKVTR   ;SETUP INT. VECTOR RETURN
1368  005110  005077  174104                  CLR     @TKLVL          ;CLR INT. PRIORITY
1369  005114  012777  000100  174072          MOV     #100,@TKS       ;ENABLE KYBD INTERRUPT
1370  005122  000001                          WAIT
1371  005124  005077  174064          TSC3C:  CLR     @TKS            ;DISABLE KYBD INTERRUPT
1372  005130  017760  174056  005504          MOV     @TKB,STORE(R0)  ;STORE CHAR
1373  005136  022760  000215  005504          CMP     #215,STORE(R0)  ;CHECK CHAR
1374  005144  001403                          BEQ     TSC3D           ;BRANCH IF CR
1375  005146  022720  000066                  CMP     #66,(R0)+       ;CHECK # OF CHARS
1376  005152  001351                          BNE     TSC3B           ;BRANCH IF LESS THAN 33
1377  005154  022737  000201  005504  TSC3D:  CMP     #201,@#STORE    ;CHECK FIRST CHAR
1378  005162  001413                          BEQ     TSC3E           ;BRANCH IF CORRECT
1379  005164  004467  001004                  JSR     R4,PRINTM       ;PRINT MESSAGE
1380  005170  006600                          MSG9                    ;"FIRST CHAR NOT ASCII SOH (001)"
1381  005172  004767  000614                  JSR     R7,DELAYA       ;DELAYA AWHILE
1382  005176  004767  000610                  JSR     R7,DELAYA
1383  005202  012705  000012                  MOV     #12,R5          ;CLEAR RT02 DISPLAY
1384  005206  004767  000542                  JSR     R7,TYP
1385  005212  022760  000215  005504  TSC3E:  CMP     #215,STORE(R0)  ;CHECK LAST CHAR
1386  005220  001413                          BEQ     TSC3F           ;BRANCH IF CORRECT
1387  005222  004467  000746                  JSR     R4,PRINTM       ;PRINT MESSAGE
1388  005226  006641                          MSG10                   ;"LAST CHAR NOT ASCII CR (015)"
1389  005230  004767  000556                  JSR     R7,DELAYA       ;DELAY AWHILE
```

```
1390  005234  004767  000552                  JSR     R7,DELAYA
1391  005240  012705  000012                  MOV     #12,R5          ;CLEAR DISPLAY
1392  005244  004767  000504                  JSR     R7,TYP
1393  005250  022700  000056          TSC3F:  CMP     #56,R0          ;CHECK # OF CHARS
1394  005254  001413                          BEQ     TSC3G           ;BRANCH IF CORRECT
1395  005256  004467  000712                  JSR     R4,PRINTM       ;PRINT MESSAGE
1396  005262  006700                          MSG11                   ;"DID NOT GET RIGHT # OF CHARS"
1397  005264  004767  000522                  JSR     R7,DELAYA       ;DELAY AWHILE
1398  005270  004767  000516                  JSR     R7,DELAYA
1399  005274  012705  000012                  MOV     #12,R5          ;CLEAR DISPLAY
1400  005300  004767  000450                  JSR     R7,TYP
1401  005304  012700  000002          TSC3G:  MOV     #2,R0           ;SET TABLE POINTER TO FIRST CHAR
1402  005310  005737  005502                  TST     @#LPFLG         ;CHECK LOOP FLAG
1403  005314  001040                          BNE     TSC3I           ;BRANCH IF REJECT & SECOND TIME READING
1404  005316  032767  000001  172244          BIT     #BIT0,CSR       ;CHECK IF ACCEPT OR REJECT
1405  005324  001434                          BEQ     TSC3I           ;BRANCH IF ACCEPT
1406  005326  012705  000012                  MOV     #12,R5          ;CLEAR DISPLAY
1407  005332  004767  000416                  JSR     R7,TYP
1408  005336  012705  000007                  MOV     #7,R5           ;BELL CODE - REJECT CODE
1409  005342  004767  000406                  JSR     R7,TYP
1410  005346  016005  005504          TSC3H:  MOV     STORE(R0),R5    ;GET NEXT CHAR
1411  005352  004767  000376                  JSR     R7,TYP          ;SEND CHAR
1412  005356  026027  005504  000215          CMP     STORE(R0),#215
1413  005364  001403                          BEQ     TSC3HA          ;BRANCH IF DONE
1414  005366  022720  000066                  CMP     #66,(R0)+
1415  005372  001365                          BNE     TSC3H           ;BRANCH IF NOT DONE
1416  005374  000000                  TSC3HA: HALT                    ;WAIT FOR REREAD COMMAND (CONTINUE)
1417  005376  005237  005502                  INC     @#LPFLG         ;SET LOOP FLAG
1418  005402  012705  000025                  MOV     #25,R5          ;REREAD COMMAND (NAK)
1419  005406  004767  000342                  JSR     R7,TYP          ;SEND REREAD COMMAND
1420  005412  000167  177456                  JMP     TSC3A           ;GO REREAD CARD
1421  005416  012705  000012          TSC3I:  MOV     #12,R5          ;CLEAR DISPLAY
1422  005422  004767  000326                  JSR     R7,TYP
1423  005426  012705  000006                  MOV     #6,R5           ;SEND ACK
1424  005432  004767  000316                  JSR     R7,TYP
1425  005436  016005  005504          TSC3J:  MOV     STORE(R0),R5    ;GET NEXT CHAR
1426  005442  004767  000306                  JSR     R7,TYP          ;SEND CHAR
1427  005446  005720                          TST     (R0)+           ;INCREMENT TABLE POINTER
1428  005450  026027  005504  000215          CMP     STORE(R0),#215
1429  005456  001403                          BEQ     TSC3K           ;BRANCH IF DONE
1430  005460  020027  000066                  CMP     R0,#66
1431  005464  002764                          BLT     TSC3J           ;BRANCH IF NOT DONE
1432  005466  012705  000013          TSC3K:  MOV     #13,R5          ;SEND VT TO EJECT CARD
1433  005472  004767  000256                  JSR     R7,TYP
1434  005476  000167  177366                  JMP     TSC3AA          ;RESTART TEST
1435
1436  005502  000000                  LPFLG:  0                       ;LOOP FLAG
1437
1438  005504  000000                  STORE:  0
1439  005506  000000                          0
1440  005510  000000                          0
1441  005512  000000                          0
1442  005514  000000                          0
1443  005516  000000                          0
1444  005520  000000                          0
1445  005522  000000                          0
```

```
1446  005524  000000                          0
1447  005526  000000                          0
1448  005530  000000                          0
1449  005532  000000                          0
1450  005534  000000                          0
1451  005536  000000                          0
1452  005540  000000                          0
1453  005542  000000                          0
1454  005544  000000                          0
1455  005546  000000                          0
1456  005550  000000                          0
1457  005552  000000                          0
1458  005554  000000                          0
1459  005556  000000                          0
1460  005560  000000                          0
1461  005562  000000                          0
1462  005564  000000                          0
1463  005566  000000                          0
1464  005570  000000                          0
1465  005572  000000                          0
1466  005574  000000                          0
1467  005576  000000                          0
1468  005600  000000                          0
1469  005602  000000                          0
1470
1471
1472
1473                                  ;TST 3  RT01/RT02 KEYBOARD CHARACTER TEST
1474                                  ; TRANSMITS THE KEY THAT IS PRESSED,
1475                                  ;ON THE RT01/RT02
1476
1477  005604  012706  001200          TST3:   MOV     #BUFF,R6        ;SETUP STACK
1478  005610  012777  005632  173400          MOV     #TST3A,@TKVTR   ;SET INT. VECTOR RETURN
1479  005616  005077  173376                  CLR     @TKLVL          ;CLR INT. PRIORITY
1480  005622  012777  000100  173364          MOV     #100,@TKS       ;ENABLE KEYBOARD INTERRUPT.
1481  005630  000001                          WAIT                    ;WAIT FOR KEY TO BE PRESSED.
1482  005632  005077  173356          TST3A:  CLR     @TKS            ;CLEAR KYBD FLAG
1483  005636  017705  173350                  MOV     @TKB,R5         ;CHAR TYPED TO R5
1484  005642  004767  000106                  JSR     R7,TYP          ;TRANSMIT R5
1485  005646  000756                          BR      TST3            ;GET ANOTHER CHAR.
1486
1487
```

```
1488                                        ;TST 4  TRANSMITS BITS 7-0 OF CONSOLE SWITCH REGISTER
1489                                        ;CHARS. ARE TRANS. ONCE FOR EACH DIFFERENT CODE
1490
1491  005650  116777  171714  173330  TST4:   MOVB    CSR,@TPB        ;TRANS SR. TO RT01/RT02
1492  005656  116700  171706          TST4A:  MOVB    CSR,R0          ;GET CHAR FOR COMPARE
1493  005662  105777  173322                  TSTB    @TPS
1494  005666  100375                          BPL     .-4             ;WAIT FOR DONE
1495  005670  116701  171674                  MOVB    CSR,R1          ;GET ANOTHER CHAR FOR COMPARE
1496  005674  120001                          CMPB    R0,R1           ;COMPARE CHARS.
1497  005676  001774                          BEQ     TST4A           ;BR IF EQUAL
1498  005700  004767  000064                  JSR     R7,DELAY        ;DELAY AWHILE
1499  005704  000761                          BR      TST4            ;DO IT AGAIN
1500
1501
1502
1503                                        ;TST 5  TRANSMITS BITS 7-0 OF CONSOLE SWITCH REGISTER.
1504                                        ;TRANSMITS CHARACTER REPEATEDLY
1505
1506  005706  116700  171656          TST5:   MOVB    CSR,R0          ;SAVE FOR CHECK
1507  005712  116777  171652  173266  TST5A:  MOVB    CSR,@TPB        ;TRANSMIT SR
1508  005720  105777  173264                  TSTB    @TPS            ;CHECK DONE FLAG
1509  005724  100375                          BPL     .-4             ;BR. IF NOT DONE
1510  005726  116701  171636          TST5B:  MOVB    CSR,R1          ;FOR DATA CHECK
1511  005732  042700  000200                  BIC     #200,R0
1512  005736  042701  000200                  BIC     #200,R1         ;CLEAR OUT PARITY
1513  005742  120001                          CMPB    R0,R1           ;SWITCH REG CHANGED?
1514  005744  001762                          BEQ     TST5A           ;BR. IF NOT
1515  005746  004767  000016                  JSR     R7,DELAY        ;GIVE TIME TO CHANGE CHAR.
1516  005752  000755                          BR      TST5            ;GET ANOTHER CAR.
1517
1518
1519
1520                                        ;SUBR. TO OUTPUT CHAR CONTAINED IN R5
1521
1522  005754  010577  173226          TYP:    MOV     R5,@TPB         ;CHAR TO DISPLAY BUFFER
1523  005760  105777  173224                  TSTB    @TPS            ;CHECK DONE FLAG
1524  005764  100375                          BPL     .-4             ;DONE
1525  005766  000207                          RTS     R7              ;YES RETURN TO CALLING ROUTINE
1526
1527                                        ;ROUTINE TO DELAY AWHILE
1528
1529  005770  032767  004000  171572  DELAY:  BIT #BIT11,CSR          ;CHECK ENDLESS DELAY
1530  005776  001405                          BEQ DELAYA              ;BR IF  NOT WANTED
1531  006000  005077  173206                  CLR @TKB                ;CLEAR RYBD FLAG
1532  006004  105777  173204                  TSTB    @TKS            ;CHECK DONE FLAG
1533  006010  100375                          BPL     .-4             ;WAIT FOR DONE FLAG
1534  006012  016703  173166          DELAYA: MOV     BRCTA,R3        ;INIT  2ND CNT
1535  006016  012767  007777  000014  DELAYB: MOV     #7777,BRCT      ;INIT 1ST CNT
1536  006024  005367  000010                  DEC     BRCT            ;DEC 1ST CNT
1537  006030  001375                          BNE     .-4             ;DELAY APPROX .2 SEC
1538  006032  005303                          DEC     R3              ;BR. IF NOT DONE
1539  006034  001370                          BNE     DELAYB          ;BR. IF NOT DONE
1540  006036  000207                          RTS     R7              ;RETURN TO CALLER
1541  006040  000000          BRCT:   OPEN                            ;DELAY CNT HOLDER
1542                                        ;REPEAT AND HALT CHECK OF THE SWITCH REGISTER
1543
```

```
1544  006042  032767  040000  171520  SCOP:   BIT     #BIT14,CSR      ;LOOP ON TEST?
1545  006050  001401                          BEQ     .+4             ;BR. IF NO LOOP
1546  006052  000207                          RTS     R7              ;EXIT TO HALT
1547  006054  062716  000002                  ADD     #2,@R6          ;SKIP PAST LOOP
1548  006060  032767  100000  171502          BIT     #BIT15,  CSR    ;HALT @ END OF TEST?
1549  006066  001002                          BNE     .+6             ;BR. IF HALT  DESIRED
1550  006070  062716  000004                  ADD     #4,@R6          ;SKIP HALT AND BRANCH BACK
1551  006074  000207                          RTS     R7              ;RETURN TO CALLING TEST
1552
1553
1554                                  ;POWER FAIL ROUTINE
1555
1556  006076  012737  000340  177776  PFAIL:  MOV     #340,@#PSW
1557  006104  010046                          MOV     R0,-(6)
1558  006106  010146                          MOV     R1,-(6)
1559  006110  010246                          MOV     R2,-(6)
1560  006112  010346                          MOV     R3,-(6)
1561  006114  010446                          MOV     R4,-(6)
1562  006116  010546                          MOV     R5,-(6)
1563  006120  016746  171700                  MOV     24,-(6)
1564  006124  010667  000010                  MOV     R6,SAVR6
1565  006130  012767  006142  171666          MOV     #RES,24         ;SAVE POWER UP ADDRESS.
1566  006136  000000                          HALT                    ;WAIT FOR POWER TO GO AWAY
1567
1568  006140  000000                  SAVR6:  0
1569
1570
1571                                  ;POWER UP ROUTINE
1572
1573  006142  016706  177772          RES:    MOV     SAVR6,R6
1574  006146  012667  171652                  MOV     (6)+,24
1575  006152  012605                          MOV     (6)+,R5
1576  006154  012604                          MOV     (6)+,R4
1577  006156  012603                          MOV     (6)+,R3
1578  006160  012602                          MOV     (6)+,R2
1579  006162  012601                          MOV     (6)+,R1
1580  006164  012600                          MOV     (6)+,R0
1581  006166  005037  177776                  CLR     @#PSW
1582  006172  000002                          RTI                     ;GO BACK TO INTERRUPTED ROUTINE
1583
1584
1585
1586                                  ;PRINTS A MESSAGE ON THE TTY
1587
1588  006174  012403                  PRINTM: MOV     (R4)+,R3        ;GET ADDRESS OF MESSAGE
1589  006176  112305                  PRINTA: MOVB    (R3)+,R5        ;CHAR TO R5
1590  006200  001403                          BEQ     EXIT            ;EXIT IF 000 FOUND
1591  006202  004767  177546                  JSR     R7,TYP          ;TRANSMIT R5
1592  006206  000773                          BR      PRINTA          ;NOT DONE
1593  006210  000204                  EXIT:   RTS     R4              ;GO BACK
1594
1595
1596
1597
1598  006212  005015  051524  031124  MSG1:   .ASCIZ<15><12>  /TST2A RT02-A DISPLAY TEST/
1599  006220  020101  052122  031060
```

```
1600  006226  040455  042040  051511
1601  006234  046120  054501  052040
1602  006242  051505  000124
1603  006246  005015  051524  031124  MSG2:   .ASCIZ<15><12>  /TST2B RT02-A OCTAL KEYBOARD TEST/
1604  006254  020102  052122  031060
1605  006262  040455  047440  052103
1606  006270  046101  045440  054505
1607  006276  047502  051101  020104
1608  006304  042524  052123     000
1609  006311     015  052012  041123  MSG3:   .ASCIZ<15><12>  /TSBAA RT02-B AUDIO ALARM TEST/
1610  006316  040501  051040  030124
1611  006324  026462  020102  052501
1612  006332  044504  020117  046101
1613  006340  051101  020115  042524
1614  006346  052123     000
1615  006351     015  052012  041123  MSG4:   .ASCIZ<15><12>  /TSB2A RT02-B DISPLAY TEST/
1616  006356  040462  051040  030124
1617  006364  026462  020102  044504
1618  006372  050123  040514  020131
1619  006400  042524  052123     000
1620  006405     015  052012  041123  MSG5:   .ASCIZ<15><12>  /TSB3 RT02-B KYBD OUTPUT TEST/
1621  006412  020063  052122  031060
1622  006420  041055  045440  041131
1623  006426  020104  052517  050124
1624  006434  052125  052040  051505
1625  006442  000124
1626  006444  005015  051524  040503  MSG6:   .ASCIZ<15><12>  /TSCAA RT02-C DISPLAY TEST/
1627  006452  020101  052122  031060
1628  006460  041455  042040  051511
1629  006466  046120  054501  052040
1630  006474  051505  000124
1631  006500  005015  051524  031103  MSG7:   .ASCIZ<15><12>  /TSC2A RT02-C OCTAL KYBRD TEST/
1632  006506  020101  052122  031060
1633  006514  041455  047440  052103
1634  006522  046101  045440  041131
1635  006530  042122  052040  051505
1636  006536  000124
1637  006540  005015  051524  031503  MSG8:   .ASCIZ<15><12>  /TSC3 RT02-C BADGE READER TEST/
1638  006546  051040  030124  026462
1639  006554  020103  040502  043504
1640  006562  020105  042522  042101
1641  006570  051105  052040  051505
1642  006576  000124
1643  006600  005015  044506  051522  MSG9:   .ASCIZ<15><12>  /FIRST CHAR NOT ASCII SOH (001)/
1644  006606  020124  044103  051101
1645  006614  047040  052117  040440
1646  006622  041523  044511  051440
1647  006630  044117  024040  030060
1648  006636  024461     000
1649  006641     015  046012  051501  MSG10:  .ASCIZ<15><12>  /LAST CHAR NOT ASCII CR (015)/
1650  006646  020124  044103  051101
1651  006654  047040  052117  040440
1652  006662  041523  044511  041440
1653  006670  020122  030050  032461
1654  006676  000051
1655  006700  005015  044504  020104  MSG11:  .ASCIZ<15><12>  /DID NOT GET RIGHT # OF CHARS/
```

```
 1656  006706  047516  020124  042507
 1657  006714  020124  044522  044107
 1658  006722  020124  020043  043117
 1659  006730  041440  040510  051522
 1660  006736     000
 1661
 1662
 1663          006740                          .EVEN
 1664
 1665
 1666
 1667
 1668
 1669
 1670                                  ;FOLLOWING IN CASE LOADED BY ACT11
 1671
 1672
 1673  006740  013700  000042          ACT:    MOV     @#42,R0
 1674  006744  000005                          RESET
 1675  006746  004710                  $ENDAD: JSR     R7,@R0          ;EXIT TO MONITOR
 1676  006750  000240                          NOP
 1677  006752  000240                          NOP
 1678  006754  000240                          NOP
 1679          000001                          .END
```

```
ACT     006740            799     900    1077    1285    1673#
ASC     003704           1124    1135    1137#
BIT0  = 000001            643#    1404
BIT1  = 000002            642#
BIT10 = 002000            633#
BIT11 = 004000            632#    1529
BIT12 = 010000            631#
BIT13 = 020000            630#
BIT14 = 040000            629#    1544
BIT15 = 100000            628#    1548
BIT2  = 000004            641#
BIT3  = 000010            640#
BIT4  = 000020            639#
BIT5  = 000040            638#
BIT6  = 000100            637#
BIT7  = 000200            636#
BIT8  = 000400            635#
BIT9  = 001000            634#
BRCT    006040           1535*   1536*   1541#
BRCTA   001204            655#    1534
BUFF    001200 -          653#     710     749     785     812     856     863     912     954    1041    1090    1240    1247
                         1301    1308    1366    1477
CC    = 177776            620#
CNT     001202            654#     750     761     768     773
CODE    004026           1097    1106    1127    1134    1163#
CRBUFR  001230            665#
CRBUFT  001226            664#
CSR   = 177570            619#    1404    1491    1492    1495    1506    1507    1510    1529    1544    1548
DELAY   005770            715     760     782     793     870     875     878     892     897    1047    1052    1055    1069
                         1074    1229    1254    1259    1262    1277    1282    1498    1515    1529#
DELAYA  006012            714     847     848     859     860     896     957     958     961     967     968     969     970
                          971    1000    1024    1025    1037    1038    1073    1228    1243    1244    1281    1359    1360
                         1381    1382    1389    1390    1397    1398    1530    1534#
DELAYB  006016           1535#   1539
ERROR   003544           1098    1110#
EXIT    006210           1590    1593#
LPFLG   005502           1364*   1402    1417*   1436#
MSG1    006212            858    1598#
MSG10   006641           1388    1649#
MSG11   006700           1396    1655#
MSG2    006246            911    1603#
MSG3    006311            956    1609#
MSG4    006351           1036    1615#
MSG5    006405           1088    1620#
MSG6    006444           1242    1626#
MSG7    006500           1298    1631#
MSG8    006540           1358    1637#
MSG9    006600           1380    1643#
OPEN  = 000000            622#     653    1541
PFAIL   006076            600    1556#
PRINTA  006176           1589#   1592
PRINTM  006174            857     910     955    1035    1087    1115    1125    1241    1297    1357    1379    1387    1395
                         1588#
PRTY3 = 000140            627#
PRTY4 = 000200            626#     661     663
PRTY5 = 000240            625#
```

```
PRTY6 = 000300          624#
PRTY7 = 000340          623#
PSW   = 177776          621#  1556*  1581*
RES     006142         1565   1573#
RMSG    003776         1116   1157#
R0    =%000000          607#   750*   758*   761*   768*   771*   773*   776*   779*   793*   811*   845*   864*
                        871*   887*   890*   909*   944*   986*  1001*  1002   1004   1007  1042*  1048*  1064*
                       1067*  1089*  1097   1105*  1106   1127   1134   1248*  1255*  1272*  1275*  1296*  1340*
                       1365*  1372*  1373   1375   1385   1393   1401*  1410   1412   1414   1425   1427   1428
                       1430   1492*  1496   1506*  1511*  1513   1557   1580*  1673*  1675
R1    =%000001          608#  1114*  1117   1134*  1137   1138*  1139*  1154   1495*  1496   1510*  1512*  1513
                       1558   1579*
R2    =%000002          609#   831*   841    932*   940    962*   972*   989*   991*   996*  1143*  1144*  1145*
                       1152   1328*  1336   1559   1578*
R3    =%000003          610#   823*   843    924*   942   1124*  1135*  1156*  1320*  1338   1534*  1538*  1560
                       1577*  1588*  1589
R4    =%000004          611#   711*   712    716    751*   756    762*   763    788*   791    794*   795    819*
                        820    824    828*   832*   833*   834*   835*   836*   839    857*   865*   868    881*
                        882    884*   888    893*   894    910*   920*   921    925    929*   933*   934*   935*
                        936*   937*   938    955*  1035*  1043*  1045   1058*  1059   1061*  1065   1070*  1071
                       1087*  1096*  1097   1101   1103   1114   1115*  1125*  1241*  1249*  1252   1266*  1267
                       1269*  1273   1278*  1279   1297*  1316*  1317   1321   1325*  1329*  1330*  1331*  1332*
                       1333*  1334   1357*  1379*  1387*  1395*  1561   1576*  1588   1593*
R5    =%000005          612#   712*   752*   754*   756*   765*   769*   774*   778*   780*   786*   789*   791*
                        820*   821*   822*   823    824*   825*   826*   827*   828    829*   830*   831    837*
                        839*   841*   843*   861*   866*   868*   873*   876*   879*   885*   888*   918*   921*
                        922*   923*   924    925*   926*   927*   928*   929    930*   931*   932    938*   940*
                        942*   959*   963*   965*   987*   992*   994*   998*  1009*  1011*  1013*  1014   1016*
                       1017   1019*  1020   1039*  1044*  1045*  1050*  1053*  1056*  1062*  1065*  1099*  1103*
                       1110*  1112*  1117*  1118   1120*  1122*  1127*  1128   1130*  1132*  1137*  1140*  1141*
                       1142*  1143   1146*  1147*  1148*  1149*  1150*  1152*  1154*  1245*  1250*  1252*  1257*
                       1260*  1263*  1270*  1273*  1299*  1314*  1317*  1318*  1319*  1320   1321*  1322*  1323*
                       1324*  1325   1326*  1327*  1328   1334*  1336*  1338*  1361*  1383*  1391*  1399*  1406*
                       1408*  1410*  1418*  1421*  1423*  1425*  1432*  1483*  1522   1562   1575*  1589*
R6    =%000006          613#   710*   749*   785*   812*   856*   863*   912*   954*  1041*  1090*  1240*  1247*
                       1301*  1308*  1366*  1477*  1547*  1550*  1564   1573*
R7    =%000007          614#   713*   714*   715*   718*   753*   755*   757*   760*   766*   770*   775*   781*
                        782*   787*   790*   792*   793*   800*   838*   840*   842*   844*   847*   848*   849*
                        859*   860*   862*   867*   869*   870*   874*   875*   877*   878*   880*   886*   889*
                        892*   896*   897*   901*   919*   939*   941*   943*   946*   957*   958*   960*   961*
                        964*   966*   967*   968*   969*   970*   971*   988*   995*   999*  1000*  1010*  1012*
                       1024*  1025*  1026*  1037*  1038*  1040*  1046*  1047*  1051*  1052*  1054*  1055*  1057*
                       1063*  1066*  1069*  1073*  1074*  1078*  1100*  1104*  1111*  1113*  1121*  1123*  1131*
                       1133*  1151*  1153*  1155*  1228*  1229*  1230*  1243*  1244*  1246*  1251*  1253*  1254*
                       1258*  1259*  1261*  1262*  1264*  1271*  1274*  1277*  1281*  1282*  1286*  1300*  1315*
                       1335*  1337*  1339*  1342*  1359*  1360*  1362*  1381*  1382*  1384*  1389*  1390*  1392*
                       1397*  1398*  1400*  1407*  1409*  1411*  1419*  1422*  1424*  1426*  1433*  1484*  1498*
                       1515*  1525*  1540*  1546*  1551*  1591*  1675*
SAVR6   006140         1564*  1568#  1573
SCOP    006042          718    800    849    901    946   1026   1078   1230   1286   1342   1544#
STORE   005504         1372*  1373   1377   1385   1410   1412   1425   1428   1438#
TABR    001446          711    726#
TB2AP   003416         1078#  1081
TC2AP   004560         1286#  1289
TC2BP   005024         1342#  1345
TKB     001212          658#   815*   819    920   1096   1307*  1316   1372   1483   1531*
```

```
TKLVL   001220          661*    814*    914*   1092*   1303*   1310*   1368*   1479*
TKS     001214          659*    816*    818*    915*    917*   1093*   1095*   1304*   1306*   1311*   1313*   1369*   1371*
                       1480*   1482*   1532
TKVTR   001216          660*    813*    913*   1091*   1302*   1309*   1367*   1478*
TPB     001206          656*   1491*   1507*   1522*
TPLVL   001224          663*
TPS     001210          657*   1493    1508    1523
TPVTR   001222          662*
TSBAA   002660          683     954*   1027    1234
TSBAA2  002772          986*
TSB2A   003172          684    1035*   1079
TSB2AA  003332         1062*   1072
TSB2AB  003346         1065*   1068
TSB2AC  003220         1041*
TSB2AD  003240         1045*   1060
TSB2AE  003316         1049    1058*
TSB2AP  003150         1008    1024*   1029
TSB3    003430          685    1087*   1227
TSB3A   003440         1090*   1107    1136
TSB3B   003466         1091    1095*
TSB3BB  003530         1102    1105*
TSB3C   004276         1108    1228*
TSB3CC  004306         1230*   1233
TSB3D   004272         1227*   1231
TSCAA   004324          689    1240*   1287
TSC2A   004572          690    1296*   1343
TSC2AA  004474         1270*   1280
TSC2AB  004510         1273*   1276
TSC2AC  004356         1247*
TSC2AD  004402         1252*   1268
TSC2AE  004460         1256    1266*
TSC2BA  004652         1308*   1341
TSC2BB  004700         1309    1313*
TSC2BC  004642         1302    1306*
TSC3    005036          691    1357*
TSC3A   005074         1365*   1420
TSC3AA  005070         1364*   1434
TSC3B   005076         1366*   1376
TSC3C   005124         1367    1371*
TSC3D   005154         1374    1377*
TSC3E   005212         1378    1385*
TSC3F   005250         1386    1393*
TSC3G   005304         1394    1401*
TSC3H   005346         1410*   1415
TSC3HA  005374         1413    1416*
TSC3I   005416         1403    1405    1421*
TSC3J   005436         1425*   1431
TSC3K   005466         1363    1429    1432*
TST1A   001400          672     710*    719
TST1AA  001410          712*    717
TST1B   001512          573     722     749*    801     853
TST1BA  001536          754*    767
TST1BB  001546          756*    759
TST1BC  001612          764     768*
TST1BD  001702          785*
TST1BE  001732          791*    796
```

DZRTAC.SRC CROSS REFERENCE TABLE -- USER SYMBOLS

```
TST1BF  001706            786#
TST1C   002000            674     811#    850
TST1CA  002004            812#    846
TST1CB  002036            813     818#
TST2A   002210            678     856#    902     950
TST2AA  002360            885#    895
TST2AB  002374            888#    891
TST2AC  002242            863#
TST2AD  002266            868#    883
TST2AE  002344            872     881#
TST2B   002456            679     909#    947
TST2BA  002470            912#    945
TST2BB  002516            913     917#
TST3    005604            695    1477#   1485
TST3A   005632           1478    1482#
TST4    005650            696    1491#   1499
TST4A   005670           1495#   1497
TST5    005706            697    1506#   1516
TST5A   005712           1507#   1514
TST5B   005726           1510#
TS1AP   001432            718#    721
TS1BP   001766            800#    803
TS1CP   002172            849#    852
TS2AP   002444            901#    904
TS2BP   002642            946#    949
TYP     005754            713     753     755     757     766     770     775     781     787     790     792     838     840
                          842     844     862     867     869     874     877     880     886     889     919     939     941
                          943     960     964     966     988     995     999    1010    1012    1040    1046    1051    1054
                         1057    1063    1066    1100    1104    1111    1113    1121    1123    1131    1133    1151    1153
                         1155    1246    1251    1253    1258    1261    1264    1271    1274    1300    1315    1335    1337
                         1339    1362    1384    1392    1400    1407    1409    1411    1419    1422    1424    1426    1433
                         1484    1522#   1591
WMSG    004011           1126    1159#
$ENDAD  006746            648    1675#
.     = 006756            591#    592#    595     598#    599#    647#    649#    651#    670#    676#    681#    687#    693#
                          703#    798     899    1076    1226#   1284    1494    1509    1524    1533    1537    1545    1549
                         1663#
```

```
ADC      822   830   836   923   931   937  1139  1145  1150  1319  1327  1333  1547  1550
BEQ      764   798   899  1003  1005  1008  1076  1294  1374  1378  1386  1394  1405  1413  1429
        1497  1514  1530  1545  1590
BGE     1119  1129
BIC      821   829   835   922   930   936  1138  1144  1149  1318  1326  1332  1511  1512
BIT     1404  1529  1544  1548
BLT     1102  1431
BNE      717   759   772   777   784   796   846   872   883   891   895   945   973   997  1015
        1018  1021  1049  1060  1068  1072  1098  1107  1256  1268  1276  1280  1341  1376  1403
        1415  1537  1539  1549
BPL     1494  1509  1524  1533
BR       719   721   722   767   801   803   850   852   902   904   947   949   990   993  1022
        1027  1029  1079  1081  1231  1233  1287  1289  1343  1345  1485  1499  1516  1592
CLR      814   815   818   914   917   986  1011  1089  1092  1095  1203  1306  1307  1310  1313
        1364  1365  1368  1371  1479  1482  1531  1581
CMP     1002  1004  1007  1014  1017  1020  1101  1118  1128  1373  1375  1377  1385  1393  1412
        1414  1428  1430
CMPB     763   795   882   894  1059  1071  1097  1267  1279  1496  1513
DEC      758   771   776   783   845   871   890   944   972   996  1048  1067  1255  1275  1340
        1536  1538
HALT     593   596   598   720   802   851   903   948  1028  1080  1232  1288  1344  1416  1566
INC      762   794   881   893  1001  1013  1016  1019  1058  1070  1105  1266  1278  1417
JMP      672   673   674   678   679   683   684   685   689   690   691   695   696   697   799
         853   900   950  1077  1108  1136  1227  1234  1285  1363  1420  1434
JSR      713   714   715   718   753   755   757   760   766   770   775   781   782   787   790
         792   793   800   838   840   842   844   847   848   849   857   859   860   862   867
         869   870   874   875   877   878   880   886   889   892   896   897   901   910   919
         939   941   943   946   955   957   958   960   961   964   966   967   968   969   970
         971   988   995   999  1000  1010  1012  1024  1025  1026  1035  1037  1038  1040  1046
        1047  1051  1052  1054  1055  1057  1063  1066  1069  1073  1074  1078  1087  1100  1104
        1111  1113  1115  1121  1123  1124  1125  1131  1133  1135  1151  1153  1155  1228  1229
        1230  1241  1243  1244  1246  1251  1253  1254  1258  1259  1261  1262  1264  1271  1274
        1277  1281  1282  1286  1297  1300  1315  1335  1337  1339  1342  1357  1359  1360  1362
        1379  1381  1382  1384  1387  1389  1390  1392  1395  1397  1398  1400  1407  1409  1411
        1419  1422  1424  1426  1433  1484  1498  1515  1591  1675
MOV      710   711   712   749   750   751   752   754   756   761   765   768   769   773   774
         778   779   780   785   786   788   789   791   811   812   813   816   819   820   823
         824   828   831   837   839   841   843   856   861   863   864   865   866   868   873
         876   879   884   885   887   888   909   912   913   915   918   920   921   924   925
         929   932   938   940   942   954   959   962   963   965   987   989   991   992   994
         998  1009  1039  1041  1042  1043  1044  1045  1050  1053  1056  1061  1062  1064  1065
        1090  1091  1093  1096  1099  1103  1110  1112  1114  1117  1120  1122  1130  1132  1137
        1143  1152  1154  1240  1245  1247  1248  1249  1250  1252  1257  1260  1263  1269  1270
        1272  1273  1296  1299  1301  1302  1304  1308  1309  1311  1314  1316  1317  1320  1321
        1325  1328  1334  1336  1338  1361  1366  1367  1369  1372  1383  1391  1399  1401  1406
        1408  1410  1418  1421  1423  1425  1432  1477  1478  1480  1483  1522  1534  1535  1556
        1557  1558  1559  1560  1561  1562  1563  1564  1565  1573  1574  1575  1576  1577  1578
        1579  1580  1588  1673
MOVB    1127  1134  1491  1492  1495  1506  1507  1510  1589
NOP     1676  1677  1678
RESET   1674
ROR      825   826   827   832   833   834   926   927   928   933   934   935  1140  1141  1142
        1146  1147  1148  1322  1323  1324  1329  1330  1331
RTI     1582
RTS     1156  1525  1540  1546  1551  1593
TST      716   797   898  1075  1283  1402  1427
```

DZRTAC.SRC CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

```
TSTB     1106    1493    1508    1523    1532
WAIT      817     916    1094    1305    1312    1370    1481
.ABS      581
.ASCIZ   1157    1159    1598    1603    1609    1615    1620    1626    1631    1637    1643    1649    1655
.BYTE    1163    1168    1173    1178    1183    1188    1193    1198    1203    1208    1213    1218    1223    1225
.END     1679
.EVEN    1162    1226    1663
.LIST     584     598
.NLIST    585     598
.PAGE     746     951    1110
.REM        1
.REPT     598
.TITLE    592
```

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

```
*DZRTAC,DZRTAC/SOL/CRF,PAGNUM=DZRTAC.SRC
RUN-TIME:  5 10 1 SECONDS
RUN-TIME RATIO:  98/17=5.6
CORE USED:  7K  (13 PAGES)
```

Spooler runtime 6 Seconds, [illegible] disk reads, 3 disk writes, 41 pages