DUP11

OVERLAY FOR INTER TEST

MD-11-DZDPF-B

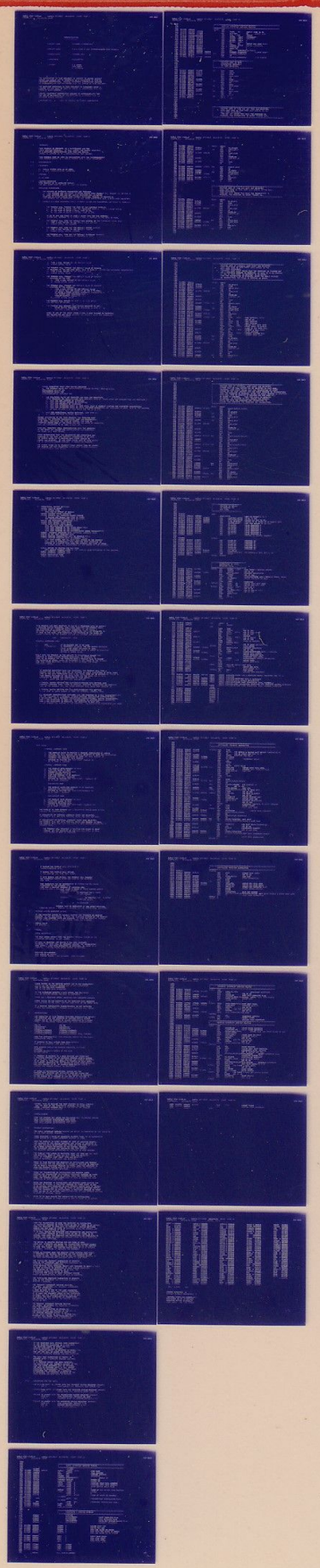
EP DZDPF B DL

JAN 1978

digital

FICHE 1 OF 1

MADE IN USA



IDENTIFICATION

PRODUCT CODE: MAYNDEC-11-DZDPF-B-D

PRODUCT NAME: DUP11 OVERLAY FOR INTERPROCESSOR TEST PROGRAM

PROGRAM DATE: OCTOBER 1976

MAINTAINER: DIAGNOSTICS

AUTHORS: R A JONES
JOHN EGOLF
R. OCHESTER

A1

1.0 ABSTRACT.

THIS PROGRAM IS DESIGNED AS A MAINTENANCE AID FOR FIELD SERVICE PERSONNEL. IT WILL VERIFY THE PROPER OPERATION OF A COMPLETE COMMUNICATION LINK FROM ONE PDP-11 SYSTEM TO ANOTHER OR TO A COMMUNICATION TEST CENTER.

THIS PROGRAM MUST BE USED IN CONJUNCTION WITH THE INTERPROCESSOR TEST PROGRAM(DZITP) ON A PDP-11 SYSTEM WITH A DL-11 INTERFACE.

2.0 REQUIREMENTS.

2.1 EQUIPMENT

A. PDP-11 SYSTEM WITH 4K OF CORE.
B. A DUP11 COMMUNICATION INTERFACE.

2.2 STORAGE.

4K OF CORE

3.0 LOADING PROCEDURE
THIS PROGRAM IS IN ABSOLUTE FORMAT.
THE ABS LOADER MUST BE USED TO LOAD THE PROGRAM.

4.0 OPERATING PROCEDURES.

A. TWO METHODS OF ENTERING PARAMETERS ARE PROVIDED
1. LOAD ADDRESS 200 AND START TO ENTER PARAMS FROM CONSOLE TTY, PROCEED TO SECTION B.
2. LOAD ADDRESS 200 AND SET SWITCH REGISTER BIT 15 BEFORE STARTING TO ENTER PARAMS FROM CONSOLE SWITCHES, PROCEED TO SECTION C.
*THE PROGRAM MAY BE RESTARTED AT LOC 204 (ONCE PARAMETERS HAVE ALREADY BEEN SELECTED)

B. CONSOLE DIALOGUE PARAMETER INPUT (CURRENT VALUES FOR PARAMETERS ARE FOUND IN OVERLAY)

1. THE PROGRAM WILL TYPEOUT THE NAME OF THE VARIABLE OVERLAY.
   A. IF YOU WISH TO SETUP JUST THE INDICATED OVERLAY, TYPE A CARAGE RETURN
   B. IF YOU WISH TO SETUP A DN11, TYPE IN DN.
   C. IF YOU WISH TO SETUP A DMC11A, TYPE IN DMA.

   IF DN OR DMA WAS TYPED IN STEP 1 ABOVE THEN THE BUS ADDRESS, VECTOR ETC. REFERED TO IN STEPS 2 THRU 7, PERTAIN TO THE DN11 OR DMAA.

2. THE PROGRAM WILL TYPE THE DEFAULT BUS ADDRESS OF THE INTERFACE UNDER TEST.
   A. TYPE A CAR. RETURN TO USE DEFAULT BUS ADDRESS
   B. TYPEIN ACTUAL BUS ADDRESS

3. THE PROGRAM WILL TYPE OUT THE DEFAULT VECTOR ADDRESS
   A. TYPE A CAR. RETURN TO USE DEFAULT ADDRESS
   B. TYPEIN ACTUAL VECTOR ADDRESS

4. THE PROGRAM WILL TYPE OUT THE DEFAULT INTERFACE PRIORITY
   NOTE: 200=PRIO 4, 240=PRIO 5, 300=PRIO 6, ETC.

B1

A. TYPE A CAR. RETURN TO USE DEFAULT VALUE
B. TYPEIN ACTUAL VALUE

5. THE PROGRAM WILL TYPEOUT THE DEFAULT VALUE OF PARAM01
IF REQUIRED BY THE ISR.(SEE SECT. 1B.P IN OVERLAY LISTING FOR PARAMETER DESCRIPTION)
A. TYPE A CAR. RETURN TO USE DEFAULT VALUE
B. TYPEIN ACTUAL VALUE

6. THE PROGRAM WILL TYPEOUT THE DEFAULT VALUE OF PARAM02
IF REQUIRED BY THE ISR.
A. TYPE A CAR. RETURN TO USE DEFAULT VALUE
B. ENTER ACTUAL VALUE

7. THE PROGRAM WILL TYPEOUT THE DEFAULT VALUE OF PARAM03
IF REQUIRED BY THE OVERLAY.
A. TYPE A CAR. RETURN TO USE DEFAULT VALUE
THE DN-11 WILL USE PARAM 03 AS THE # TO DIAL.
IF USING A MODEM WITHOUT AUTOMATIC HANDSHAKING,
THE NUMBER MUST TERMINATE WITH A
"END-OF-NUMBER" CHARACTER (;).
B. ENTER ACTUAL VALUE.

8. THE PROGRAM WILL RETURN TO STEP #1 IF THIS SETUP
WAS FOR DN11 OR DM11BB.

9. THE PROGRAM WILL REQUEST THAT SWITCH REGISTER BE SET.
A. SETUP SWITCH REGISTER AS SPECIFIED IN STEP 0.
AND TYPE A CAR. RETURN.

NOTE: IF ANY OF THE ABOVE ITEMS 2 THRU 7 WERE CHANGED BY ENTERING
NEW VALUES,THE NEW VALUE BECOMES THE DEFAULT VALUE FOR SUBSEQUENT
RESTARTS OF THE PROGRAM.

C. MANUAL PARAMETER INPUT FROM SWITCH REGISTER
    1. THE PROGRAM HALTS FOR ISR(INTERFACE SERVICE ROUTINE) SPECIFICATION
    SWR14=SETUP DH-11B ISR
    SWR13=SETUP DN-11 ISR
    SWR=XXXXXX0=SETUP VARIABLE ISR

    2. THE FOLLOWING HALTS ARE REPEATED FOR EACH ISR SPECIFIED.
        SETUP SEQUENCE IS: DN11,DM11-BB THEN V ARIABLE OVERLAY. (EACH ENTRY SET SWICHES THEN HIT CONTINUE.)
        A. HALT FOR BUS ADDRESS OF INTERFACE
        B. HALT FOR VECTOR ADDRESS OF INTERFACE
        C. HALT FOR PRIORITY OF INTERFACE
        D. HALT FOR INTERFACE PARAM #1 (SEE SECT. 1A.A IN OVERLAY LISTING FOR PARAMETER DESCRIPTION)
        E. HALT FOR INTERFACE PARAM #2 (DN11 AND DMBB PARAMETERS ARE DISCUSSED IN SECT. 1A.A OF THE MONITOR.
        F. GO BACK TO STEP A IF THIS SETUP WAS FOR DN OR DMB.

    3. HALT FOR OPERATIONAL SWITCH SETTINGS. (SEE STEP D.)
        A. PRESS CONTINUE TO START TESTING

BEFORE ATTEMPTING TO RUN THIS PROGRAM, THE OPERATOR MUST
ACCERTAIN THE COMPLETE COMMUNICATION LOOP AND PROCEDURES
TO BE USED, INCLUDING THE TYPE OF MODEMS, THE TYPE OF
INTERFACE BEING USED AT THE OTHER CPU AND THE MODES OF OPERATION,
DATA AND PARAMETERS TO BE USED AT EACH CPU.

THIS WILL REQUIRED VOCAL COMMUNICATION WITH THE OPERATOR
AT THE OTHER CPU UNLESS ITS CONFIGURATION AND OPERATION
ARE FIXED AS A TEST CENTER.

AFTER DETERMINING THAT THE EQUIPMENTS ARE COMPATIBLE AND
AGREEING ON THE MODE AND VARIABLE PARAMETERS TO BE USED,
THE SYSTEM WHICH IS TO RECEIVE DATA FIRST SHOULD BE
LOADED AND STARTED. IF THE MODEM BEING USED ON THIS SYSTEM
HAS AN AUTOMATIC ANSWER FEATURE, IT SHOULD BE ENABLED.

THE SYSTEM WHICH IS TO TRANSMIT FIRST SHOULD THEN BE LOADED
AND STARTED AND THE CONNECTION ESTABLISHED EITHER MANUALLY
OR AUTOMATICALLY (VIA DN-11).

```
0.  OPERATIONAL SWITCH SETTINGS.
    SW15=1 HALT ON ERROR
    SW14=1 SINGLE PASS
        SW14 HAS NO EFFECT IF SW04=0
    SW13=1 INHIBIT ERROR TYPEOUTS
    SW12=1 INHIBIT ALL TYPEOUTS EXCEPT ERRORS
        IF SW12=0 AND SW04=1 END PASS IS TYPED
        AND TRANSMITTED/RECEIVED DATA IS TYPED.
    SW11=1 USE PREVIOUSLY SPECIFIED DATA
    SW10=1 DATA SELECT (WITH SW09)
    SW09=1 DATA SELECT (WITH SW10)
        00=1 GET DATA FROM OPERATOR
        01=1 TEST MESSAGE #1 (3A QUICK BROWN FOX)
        10=1 TEST MESSAGE #2 (3B NUMERICS)
        11=1 TEST MESSAGE #3 (3C CONTEST/QUICK BROWN FOX/NUMERICS)
    SW08=1  TRANSMIT RECEIVED DATA (INTERNAL LOOPBACK MODE)
    SW07=1 DO NOT TEST RECEIVED DATA
    SW06=1 MONITOR TRANSMITTED DATA ON CONSOLE TTY.*
    SW05=1 MONITOR RECEIVED DATA ON CONSOLE TTY.*
        * IN MANY CASES, NOT ALL DATA WILL APPEAR ON THE CONSOLE
        TTY. THIS IS ESPECIALLY TRUE WHEN THE COMM INTERFACE IS
        RUNNING AT A FASTER BAUD THAN THE CONSOLE, BUT EVEN AT EQUAL
        OR SLOWER BAUDS, ALL CHARACTERS MAY NOT APPEAR ON THE CONSOLE.

    SW04=1 RETURN TO MONITOR FOR END PASS
        WHEN SW04=0 PROGRAM LOOPS IN THE OVERLAY NEVER RETURNING TO THE MONITOR.
    SW03=1 INTERNAL LOOPBACK MODE
    SW02=1 EXTERNAL LOOPBACK MODE
    SW01=1 ONE-WAY-IN MODE
    SW00=1 ONE-WAY-OUT MODE
```

THIS PROGRAM HAS BEEN MODIFIED TO RUN ON A PROCESSOR WITH OR WITHOUT
A HARDWARE SWITCH REGISTER. WHEN FIRST EXECUTED THE PROGRAM TESTS
THE EXISTENCE OF A HARDWARE SWITCH REGISTER. IF NOT FOUND A
SOFTWARE SWITCH REGISTER LOCATION (SWREG=LOC. 176) IS DEFAULTED TO.
IF THIS IS THE CASE, UPON EXECUTION THE CONTENTS OF THE SWREG ARE
DUMPED IN OCTAL ON THE CONSOLE TTY AND ANY CHANGES ARE REQUESTED

(IE) SWR=XXXXXX NEW=

POSSIBLE RESPONSES ARE:

1. <CR> IF NO CHANGES ARE TO BE MADE
2. 6 DIGITS 0-7 TO REPRESENT IN OCTAL THE NEW SWITCH REGISTER VALUE (LAST DIGIT FOLLOWED BY <CR>.
3. ^U TO ALLOW REENTERING VALUE IF ERROR IS COMMITTED KEYING IN SWREG VALUE.

BUILT INTO THE PROGRAM IS THE ABILITY TO DYNAMICALLY CHANGE THE
CONTENTS OF SWREG DURING PROGRAM EXECUTION. BY STRIKING ^G
(CNTL G) ON CONSOLE TTY THE OPERATOR SETS A REQUEST FLAG TO CHANGE
THE CONTENTS OF SWREG, WHICH IS PROCESSED IN KEY AREAS OF THE PROGRAM
CODE (IE) ERROR ROUTINES, AFTER HALTS END OF PASS, AND OTHER
APPLICABLE AREAS.

IF OPERATOR SPECIFIED DATA WAS INDICATED, THE PROGRAM WILL TYPE A
REQUEST FOR THE DATA. DATA MAY BE ENTERED AS ASCII CHARACTERS OR OCTAL CODE.
TYPE IN THE DATA TERMINATED WITH A CR. OCTAL CODE MAY BE ENTERED BY TYPING AN
^(UP ARROW) FOLLOWED BY THE OCTAL CODE (IN THE RANGE 000 TO 377)
SEPERATED BY SPACES AND TERMINATED BY ^(UP ARROW).
  I.E. ABCD^ 000 123 377^ EPG (CAR.RETURN)

A TYPICAL SWITCH SETTING FOR HALF-DUPLEX=003150 THIS SETTING USES
INTERNAL LOOPBACK MODE, LOOPS IN OVERLAY, MONITORS TRANSMITTED AND RECEIVED
DATA ON THE CONSOLE TTY, AND TESTS RECEIVED DATA USING TEST MESSAGE #3.

A TYPICAL SWITCH SETTING FOR FULL-DUPLEX=003144 THIS SETTING
IS THE SAME AS ABOVE EXCEPT IT USES THE EXTERNAL LOOPBACK MODE.

ALL STANDARD MESSAGES(TEST MESSAGES 1-9) ARE PRECEDED BY 2 FILL CHARACTERS(177),
AND ARE FOLLOWED BY A CR(015), LF(012),RECEIVE TERMINATING CHARACTER(001),
& FILLS(177), AND A TRANSMIT TERMINATING CHARACTER(000). DURING TRANSMISSION,
WHEN A 000 CHARACTER IS SEEN THE TRANSMISSION IS STOPPED.DURING RECEPTION,
WHEN A 001 CHARACTER IS RECEIVED, THE RECEIVER IS SHUT OFF.
IF THE MESSAGE WAS INPUTED BY THE OPERATER, THE TERMINATING CHARACTERS ARE ADDED.

TEST MODES

INTERNAL LOOPBACK MODE

1. THE OVERLAY WAITS TO RECEIVE A MESSAGE (TERMINATED BY <BRK>)
2. VERFIES THE DATA AGAINST THE DATA SELECTED BY SW09 AND SW10(SW7=0)
3. TRANSMIT THE DATA SELECTED BY SW09 AND SW10 (SW8=0) OR
   TRANSMIT THE RECEIVED DATA (SW8=1)
4. RETURNS TO MONITOR FOR "END PASS" (SW4=1) OR
   GO TO STEP 1. (SW4=0)

EXTERNAL LOOPBACK MODE

1. THE OVERLAY SETS REQUEST TO SEND
2. WAIT FOR CLEAR TO SEND
3. TRANSMITS THE SELECTED DATA
4. RESETS REQUEST TO SEND
5. WAIT FOR MESSAGE TO BE RECEIVED
6. VERIFIES THE DATA (SW07=0)
7. RETURNS TO MONITOR FOR "END PASS". (SW04=1) OR
   GO TO STEP 1(SW04=0)

ONE-WAY-IN MODE

1. THE OVERLAY WAITS FOR MESSAGE TO BE RECEIVED.
2. VERIFIES THE DATA(SW07=0)
3. RETURNS TO MONITOR FOR "END PASS"(SW04=1) OR
   GO TO STEP 1 (SW04=0)

ONE-WAY-OUT MODE

1. THE OVERLAY SETS REQUEST TO SEND
2. WAITS FOR CLEAR TO SEND
3. TRANSMITS SELECTED DATA
4. RETURNS TO MONITOR FOR "END PASS". (SW04=1) OR
   GO TO STEP 1 (SW04=0)

E. THE OVERLAY IS THEN ENTERED AND A CONNECTION ESTABLISHED EITHER MANUALLY OR AUTOMATICALLY.

IF ONE-WAY-IN OR INTERNAL LOOPBACK MODES ARE SELECTED,
THE OVERLAY WILL SET DATA TERMINAL READY AND WAIT FOR DATA.

IF ONE-WAY-OUT OR EXTERNAL LOOPBACK MODES WERE SELECTED,
THE OVERLAY WILL SET DATA TERMINAL READY AND REQUEST TO SEND.
THE OVERLAY WILL THEN WAIT FOR CLEAR TO SEND BEFORE ATTEMPTING TO
TRANSMIT DATA.

THE PROGRAM WILL PRINTOUT A "WAITING FOR CLEAR TO SEND"
MESSAGE AND THE CONTENTS OF THE XMIT CSR EVERY 60 SECS.
UNTIL CLEAR TO SEND IS ASSERTED.

F. IF SW06=0 THE OVERLAY WILL CONTINUE TO
TRANSMIT/RECEIVE DATA.

IF SW06=1 THE OVERLAY WILL RETURN
TO THE MONITOR AND TYPE "END PASS".

IF BOTH SW06=1 AND SW14=1, THE PROGRAM WILL REQUEST
NEW INTERFACE PARAMS AFTER ONE PASS OF THE SELECTED
TEST MODE.

TEST EXECUTION MAY BE INTERRUPTED BY TYPING THE FOLLOWING
CHARACTERS ON THE CONSOLE TTY.
LINE FEED = RESTART PROGRAM AT LOCATION 200.
QUESTION MARK = PRINTOUT FIRST 8 WORDS OF INPUT BUFFER.(ASCII)
THEN TYPE EITHER:

=WXXXXXX	TO PRINTOUT THE 8 WORDS AT LOC XXXXXX.
=BXXXXXX	TO PRINTOUT THE 16 BYTES AFTER LOC XXXXXX.
=C	TO CONTINUE

PROGRAM MUST BE RESTARTED AT 200 AFTER PRINTING.
CARRIAGE RETURN = RESTART AT REQUEST FOR NEW OPERATIONAL SWITCHES.

5.0 PROGRAM AND/OR OPERATOR ACTION

IF THE OPERATOR WISHES TO MANUALY EXAMINE THE TRANSMIT OR RECEIVE
BUFFERS, DO THE FOLLOWING; TO FIND THE STARTING ADDRESS OF THE RECEIVE
BUFFER, LOAD ADDRESS 11020 AND EXAMINE. TO FIND THE STARTING ADDRESS
OF THE TRANSMIT BUFFER, LOAD ADDRESS 11022 AND EXAMINE.

5.1 NORMAL HALTS
SEE SECTION 4.

6.0 ERRORS

6.1 ERROR REPORTING

THE ONLY ERROR REPORT FROM THE CONTROL PROGRAM OCCURS IF THE
INTERFACE SPECIFIED IS NOT LOADED.

IF DATA IS RECEIVED AND SWITCH 7 (NO DATA COMPARE)
IS RESET, THE DATA WILL BE COMPARED AGAINST THE PRESELECTED
DATA AFTER A LINE FEED CHARACTER IS RECEIVED. IF THERE IS A
MISMATCH, THE FOLLOWING ERROR REPORT IS PRINTED:

RECEIVED DATA=RRRRRR
DATA SHOULD BE TTTTTT
DATA COMPARE ERROR; BAD DATA=BBB GOOD DATA=GGG

WHERE RRRRRR IS THE RECEIVE BUFFER (UP TO 512 CHARACTERS)
TTTTTT IS THE TRANSMIT BUFFER (UP TO 512 CHARACTERS)
BBB IS THE BAD DATA CHARACTER
GGG IS THE GOOD DATA CHARACTER

IF THE INTERFACE DETECTS A DATA ERROR, THE FOLLOWING
WILL BE PRINTED BEFORE THE DATA IS COMPARED:

THERE WAS A RECEIVER ERROR, RECEIVER DATA REGISTER =XXXXXX

WHERE XXXXXX IS THE CONTENTS OF THE RECEIVER DATA REGISTER
THE LOW BYTE IS THE DATA, AND THE HIGH BYTE IS THE ERROR BITS.

IF A RECEIVE TERMINATING CHARACTER<CR> IS NOT DETECTED
WITHIN 512 CHARACTERS A "BUFFER FULL" PRINTOUT WILL OCCUR.

7.0 RESTRICTIONS

THE OPERATION OF THIS PROGRAM REQUIRES COORDINATION BETWEEN
THE OPERATOR AND THE OPERATOR OF ANOTHER PDP-11 SYSTEM
UNLESS ONE OF THE SYSTEMS IS ALWAYS OPERATING IN A FIXED
MODE. THE FOLLOWING TABLE LISTS THE VALID COMBINATIONS:

CPU #1	CPU #2
ONE-WAY-OUT	ONE-WAY-IN
ONE-WAY-IN	ONE-WAY-OUT
EXTERNAL-LOOPBACK	INTERNAL-LOOPBACK
INTERNAL-LOOPBACK	EXTERNAL-LOOPBACK
EXTERNAL-LOOPBACK	EXTERNAL-LOOPBACK (FULL DUPLEX)

WHEN THE COMMUNICATION LINK INVOLVES MODEMS THE FOLLOWING
RESTRICTION APPLY:

IF RUNNING IN FULL DUPLEX MODE BOTH SYSTEMS
MUST BE IN EXTERNAL LOOP BACK MODE.

BOTH SYSTEMS SHOULD BE RUNNING IDENTICAL ROUTINES.
EXAMPLE:
SWITCHES 14,13,7,6 SHOULD BE THE SAME
ON BOTH CPU'S

IF PROGRAM IS WAITING IN A SCAN ROUTINE AND TYPES OUT
A "WAITING MESSAGE", IF AN INCOMING MESSAGE STARTS DURING
THE TYPE OUT, IT WILL BE LOST BECAUSE THE TYPEOUT PRIORITY
IS AT LEVEL 7. THIS WILL RESULT IN OVERRUN OR SILO OVER-
RUN ERRORS, DEPENDING ON THE DEVICE. TO AVOID THIS SITUATION
RUN WITH SWITCH 13 UP. IF OVERRUN DOES OCCURE DURING A
TYPEOUT THE PROGRAM SHOULD BE RESTARTED.

IF USING AN ASYNCRONOUS DEVICE, MODEMS AND THE
MAYNARD TEST STATION AND INITALIZE DOES NOT CLEAR THE
CONNECTION (EXAMPLE THE DJ11) IF THE PROGRAM IS RESTARTED
IN THE MIDDLE OF A MESSAGE AT LOC 204 OR BY HITTING CR
AN IMMEDIATE ERROR MESSAGE FROM MAYNARD WILL BE RE-

GETVFD. THIS IS BECAUSE THE TEST STATION IS STILL LOOKING
FOR THE REST OF THE INTERRUPTED MESSAGE. TO AVOID THIS
ERROR , RESTART PROGRAM ONLY AT THE END OF THE MESSAGE
CURRENTLY BEING TRANSMITTED.

8.0 MISCELLANEOUS

ITEP WAS CHECKED OUT USING THE FOLLOWING BELL TELEPHONE MODEMS.
201A (HALF-DUPLEX SYNCHRONOUS 2000 BAUD)
202C (HALF-DUPLEX ASYNCHRONOUS 1200 BAUD)
103A (FULL-DUPLEX ASYNCHRONOUS 110 BAUD)

9.0 PROGRAM DESCRIPTION

9.1 THE DUP11 INTERFACE SERVICE PARAMS ARE SETUP, AS SPECIFIED BY THE OPERATOR,
BY THE ITEP CONTROL PROGRAM.

TIME: PROVIDES A MEANS OF MEASURING ELASPED TIME. IT IS INCREMENTED
EVERY SECOND BY A CLOCK INTERRUPT ROUTINE IN ITEP.

9.2 WHEN THE OVERLAY IS FIRST ENTERED BY ITEP AT LOCATION START:,
THE CONTENTS OF THE SWITCH REGISTER ARE STORED IN REGISTER 0.
THE MODE AND DATA SELECTIONS ARE FIXED AT THIS TIME AND CANNOT
BE ALTERED WITHOUT RETURNING TO THE CONTROL PROGRAM.
THE INTERRUPT VECTORS AND VARIABLES ARE THEN SETUP.
THE SELECTED ROUTINE DETERMINED BY THE MODE IS THEN ENTERED

9.3 THE OVERLAY THEN LOOPS IN ROUTINES: SOWI ,IF "ONE WAY IN" MODE
WAS SELECTED. SOWO.IF "ONE WAY OUT" MODE WAS SELECTED.
SILB. IF "INTERNAL LOOP BACK" MODE WAS SELECTED.
SXLB.IF "EXTERNAL LOOP BACK" WAS SELECTED.

9.31 SOWI: IN THIS ROUTINE THE RECEIVER IS INITIALIZED AND PROGRAM
LOOPS WAITING FOR THE RECEIVER TO FINISH.IF NOTHING IS RECEIVED
FOR 43 SECS A "WAITING" MESSAGE IS TYPED. WHEN THE RECEIVER IS
DONE.THE PROGRAM CHECKS DATA IF SWITCHES PERMIT,AND TYPES END
PASS DEPENDING ON SWITCH SETTINGS.

9.32 SOWO: THE TRANSMITTER IS INITIALIZED AND PROGRAM LOOPS
WAITING FOR TRANSMITTER TO FINISH,A "WAITING" MESSAGE IS TYPED
EVERY 60 SECS IF THERE IS NO ACTION. WHEN THE TRANSMITTER IS
DONE. THE PROGRAM EITHER LOOPS BACK TO SOWO OR TYPES END PASS
DEPENDING ON SWITCH SETTINGS.

9.33 SILB: THE RECEIVER IS INITIALIZED AND PROGRAM LOOPS WAITING FOR
RECEIVER TO FINISH. A "WAITING" MESSAGE IS TYPED EVERY 60 SEC
IF NO ACTION. WHEN RECEIVER IS DONE PROGRAM CHECKS DATA IF SWITCH
SETTINGS PERMIT, AND END PASS IS TYPED IF SWITCH SETTINGS PERMIT.
THEN THE TRANSMITTER IS INITIALIZED, A "WAITING" MESSAGE IS TYPED
EVERY 60 SEC IF NO ACTION. WHEN TRANSMITTER IS DONE PROGRAM RETURNS
TO START OF ROUTINE. (SILB)

9.34 SXLB: IF IN HALF DUPLEX THE TRANSMITTER IS INITIALIZED.
A "WAITING MESSAGE IS TYPED EVERY 60 SEC IF THERE IS NO ACTION

WHEN THE TRANSMITTER IS DONE THE RECEIVER IS INITIALIZED
,A "WAITING" MESSAGE IS TYPED EVERY 60 SEC IF THERE IS NO ACTION.
WHEN THE RECEIVER IS DONE,DATA IS CHECKED IF SWITCH SETTINGS
PERMIT AND END PASS IS TYPED IF SWITCHES ALLOW.THE PROGRAM NOW
REPEATS CYCLE STARTING AT $XLB.
IF IN FULL DUPLEX THE RECEIVER AND TRANSMITTER ARE INITIALIZED
, A "WAITING" MESSAGE IS TYPED EVERY 60 SEC IF THERE IS NO
ACTION. WHEN BOTH THE RECEIVER AND TRANSMITTER ARE DONE,DATA IS
CHECKED, END PASS IS TYPED AND PROGRAM LOOPS TO $XLA DEPENDING
ON THE SWITCH SETTINGS.

9.4 THE RETURN TO MONITOR ROUTINE FOR END PASS AT EOP:
LOCKS OUT INTERUPTS AND SAVES THE TRANSMITTER INTERUPT ENABLE
BIT AND ALL GENERAL REGISTERS. IT THEN RETURNS TO THE MONITOR
TO TYPE "END PASS". THE MONITOR CHECKS SW14 IF UP IT RETURNS
TO ENTER:, OTHERWISE IT RESTARTS THE PROGRAM.

9.5 ENTER: IS ENTERED FROM THE MONITOR AFTER TYPEING "END PASS",
IT RESTORES THE GENERAL REGISTERS AND THE TRANSMITTER CSR
AS SAVED IN EOP. THE DELAY FLAG IS SET AND PROGRAM RETURNS TO
THE SCAN ROUTINE(OWO.OWI,ILA,XLA) WHERE IT CAME FROM.

9.6 THE INITIALIZE TRANSMIT SUBROUTINE AT START:
SETS UP THE INTERFACE AND POINTERS NECESSARY TO
INITIATE A TRANSMIT OPERATION.
AFTER SETTING "DATA TERMINAL READY" AND "REQUEST TO SEND" A CHECK
IS MADE ON PARAM2 TO DETERMINE IF HALF DUPLEX OPERATION
WAS SELECTED BY THE OPERATOR. IF IT WAS, THE
SUBROUTINE WAITS FOR CLEAR TO SEND.
A 'WAITING FOR CLEAR TO SEND' PRINTOUT OCCURS
EVERY 30 SECONDS UNTIL CLEAR TO SEND IS ASSERTED.

9.7 THE INITIALIZE RECEIVED SUBROUTINE AT START:
SETS UP THE INTERFACE AND POINTERS NECESSARY TO
RECEIVE A MESSAGE.

9.8 THE TRANSMIT INTERRUPT SERVICE ROUTINE,
AT XTSR:. IS ENTERED VIA TRANSMIT INTERRUPTS
FROM THE INTERFACE.
A TEST IS MADE TO SEE IF THE LAST CHARACTER
TRANSMITTED WAS A NULL (ALL ZEROS) CHARACTER.
IF IT WAS; THE TRANSMIT LOGIC IN THE INTERFACE
IS RESET AND THE TRANSMIT COMPLETE FLAG IS SET.
AT XTSR1: THE NEXT CHARACTER IS TRANSMITTED
AND PRINTED ON THE TTY IF THE MONITOR TRANSMIT
SWITCH IS SET.

9.9 THE RECEIVE INTERRUPT SERVICE ROUTINE
.AT RISR:, IS ENTERED VIA RECEIVER INTERRUPTS
FROM THE INTERFACE.
THE RECEIVED CHARACTER IS STORED IN
THE INPUT BUFFER AND PRINTED ON THE TTY IF
THE MONITOR RECEIVER SWITCH IS SET.
IF THE INPUT BUFFER IS FULL, A 'BUFFER FULL'
PRINTOUT WILL OCCUR. THIS INDICATES THAT A
LINE FEED CHARACTER WAS NOT RECOGNIZED

K1

IN THE RECEIVED DATA (WITHIN 1000 CHARACTERS).
IF THE RECEIVED CHARACTER IS A LINE FEED,
THE RECEIVED LOGIC IS RESET AND THE
RECEIVE COMPLETE FLAG IS SET.
IF A 'RECEIVE ERROR' IS DETECTED AT RISR1, THE
CSR AND DBR WILL BE SAVED AND PRINTED OUT
AFTER THE COMPLETE MESSAGE HAS BEEN RECEIVED.

9.10 THE DATA TEST SUBROUTINE AT TESTD1 IS
ENTERED AFTER A COMPLETE MESSAGE HAS BEEN
RECEIVED.
IF A 'RECEIVE ERROR' HAD BEEN DETECTED,
THE CONTENTS OF THE 'RECEIVE BUFFER' AT THE
TIME THE ERROR OCCURRED WILL BE PRINTED.
THE DATA IS COMPARED UNTIL A 'ALL ZEROS'
CHARACTER IS RECOGNIZED. 'FILL' (ALL ONES)
CHARACTERS ARE IGNORED. IF A MISMATCH
IS DETECTED, THE COMPLETE CONTENTS OF THE
INPUT BUFFER AND GOOD DATA IS PRINTED.

10.3 PARAMETERS FOR THE DUP11

PARAM1(LOW BYTE) IS LOADED INTO THE TRANSMIT STATUS REGISTER (TXCSR).
BIT 3 HALF DUPLEX (1), DEFAULTS HALF DUPLEX (1)

PARAM1(HIGH BYTE) IS LOADED INTO THE RECEIVER STATUS REGISTER (RXCSR).
BIT A STRIP SYNC (1), DEFAULTS STRIP SYNC (1)

PARAM2 IS LOADED INTO THE TRANSMITTER BUFFER REGISTER (TXDBUF).
BITS 0-7 SYNC CHARACTER, DEFAULTS 26 (26)
BIT 8 TRANSMIT START OF MESSAGE,DEFAULT = (1)

PARAM3 IS LOADED INTO THE PARAMETER STATUS REGISTER (PARCSR).
BITS 0-7 SYNC CHARACTER, DEFAULT = 26
BIT 15 DECMODE SELECT,DEFAULT = (1)
BIT 9 CRC INHIBIT,DEFAULT = (1)

```
                                ;*****************************************************************
                                ;       DUP11 INTERFACE SERVICE PARAMS
                                ;*****************************************************************
                                        .=110000
110000  052504  000120 DUP11:  .ASCII  /DUP/              ;ISR NAME
110004  160050 BA:      160050                           ;BUS ADDRESS
110006  000300 VIV:     300                              ;VECTOR ADDRESS
110010  000240 PRIOR:   240                              ;PRIORITY
110012  000410 PARAM1:  STPSYNIMMIEN                      ;PARAM #1
110014  000426 PARAM2:  TSOM124                          ;PARAM #2
110016  101026 PARAM3:  101026                           ;PARAM #3
110020  000000 IRDA:    .WORD   0                        ;INITIAL READ DATA ADDRESS
110022  000000 IXDA:    .WORD   0                        ;INITIAL XMIT DATA ADDRESS
110024  000000 SETTLE:  .WORD   0                        ;LINE SETTLE DELAY FLAG
                        .WORD   0                        ;
110026  000000 B2D16:   .WORD   0                        ;ADDR OF BIN TO DCT TYPE ROUTINE
110030  000000 TIME:    .WORD   0                        ;TIMER
                        .WORD   0                        ;
110032  111102          .WORD   START                    ;ADDR OF START OF PROGRAM
110034                  TX.TERM:
110034  000           .BYTE   000                        ;TRANSMITTER TERMINATING CHAR.
110035                  RX.TERM:
110035  001           .BYTE   001                        ;RECEIVER TERMINATING CHAR.
110036  000000 FLAG:    .WORD   0
110040  177570 SWR:     177570
110042  177570 DISPLAY: 177570
                                ;*****************************************************************
                                ;       CONSTANTS & WORKING STORAGE
                                ;*****************************************************************
                                        STAT=R0
                                        XFLG=100000                     ;XMIT COMPLETE FLAG
                                        RFLG=040000                     ;RCV COMPLETE FLAG
                                        DSFLG=020000                    ;DATA SET STATUS CHANGE FLAG
                                        RIT13=020000                    ;INHIBIT PRINTOUTS
110044  000000 SXCSR:   0                               ;SAVED XMIT CSR
110046  000000 SRCSR:   0                               ;SAVED RCV CSR
110050  000000 ERCSR:   0                               ;RCV CSR SAVED ON ERROR
110052  000000 ERDBR:   0                               ;RCV DATA REG SAVED ON ERROR
110054  000000 DSSTAT:  0                               ;RCV CSR SAVED ON DS CHANGE

110056  000000 XCC:     0                               ;XMIT CHAR COUNT
110060  000000 RCC:     0                               ;RCV CHAR COUNT
110062  000000 RDA:     0                               ;RCV DATA ADDR.
110064  000000 XDA:     0                               ;XMIT DATA ADDR.

110066  177560 TKS:     177560
110070  177562 TKB:     177562
110072  177564 TPS:     177564
110074  177566 TPB:     177566

                        FULL.DUPLEX=000001
```

```
                                        ;*****************************************************************
                                        ;       DUP11-X INTERFACE SERVICE ROUTINE
                                        ;*****************************************************************
011102  000240          START:  NOP
011104  017700  177734          MOV     @SWR,   R0      ;SETUP MODE IN R0
011110  042700  177400          BIC     #177400,R0      ;STRIP JUNK
011114  013702  011006          MOV     RIV,    R2      ;SETUP
011120  012722  013606          MOV     $RTSR,  (R2)+   ;INTERRUPT
011124  013722  011010          MOV     PRTOR,  (R2)+   ;VECTORS
011130  012722  013510          MOV     $XTSR,  (R2)+   ;
011134  013722  011010          MOV     PRTOR,  (R2)+   ;
011140  013704  011004          MOV     RA,     R4      ;SETUP BUS ADDR INDEX
011144  013714  011012          MOV     PARAM1, @RCSR   ;SETUP VARIABLES
011150  013702  011014          MOV     PARAM2,R2       ;
011154  042702  000001          BIC     #000001,R2      ;
011160  010264  000004          MOV     R2,     XCSR(R4);IN CSR'S
011164  052764  000400  000004  BIS     #MRESET,TXCSR(R4)       ;MASTER RESET
011172  032764  000400  000004  18:     BIT     #MRESET,TXCSR(R4)
011200  001374                  BNE     1$
011202  013764  011016  000002          MOV     PARAM3,PARCSR(R4)       ;LOAD PARCSR

                                        ;*****************************************
                                        ;       ROUTINE USED TO GOTO
                                        ;       SUBROUTINE DEPENDENT
                                        ;       ON MODE SELECTED.
                                        ;*****************************************

011210  005037  011032          GO:     CLR     TIME
011214  005037  013110                  CLR     DELAY
011220  005037  013114                  CLR     STOP
011224  032700  000001                  BIT     #BDMO,MODE
011230  001402                          BEQ     1$
011232  000137  011406                  JMP     $DMO
011236  032700  000002          18:     BIT     #BDMI,MODE
011242  001402                          BEQ     2$
011244  000137  011300                  JMP     $DMI
011250  032700  000010          28:     BIT     #ILB,MODE
011254  001402                          BEQ     3$
011256  000137  011504                  JMP     $TLB
011262  032700  000004          38:     BIT     #XLB,MODE
011266  001402                          BEQ     4$
011270  000137  011734                  JMP     $XLB
011274  000000                  48:     HALT
011276  000776                          BR      .-2

                                        ;*****************************************************************
                                        ;       ROUTINE USED IF "ONE WAY IN" MODE WAS SELECTED.
                                        ;       NOTE THAT WHEN IN THIS MODE HALF DUPLEX IS THE
                                        ;       ONLY MODE AVAILABLE.
                                        ;       "ONE WAY IN" MEANS THAT ONLY THE RECEIVER IS
                                        ;       ENABLED. THE TRANSMITTER IS NEVER "TURNED ON".
                                        ;*****************************************************************
```

```
691
692
693  011300  104416                    SOWI:   KBDIN
694  011302  004737  013414                    JSR     PC,STARTR
695  011306  032700  040000            1$:     BIT     #RPLG,STAT
696  011312  001013                            BNE     2$
697  011314  023727  011032  000100            CMP     TIME,#100
698  011322  103771                            BLO     1$
699  011324  011402                            MOV     @RCSR,R2
700  011326  016403  000004                    MOV     XCSR(R4),R3
701  011332  104001                            WLT     1
702  011334  005037  011032                    CLR     TIME
703  011340  000762                            BR      1$
704
705  011342  032777  000200  177674  2$:     BIT     #NODAT,@SWR
706  011350  001002                            BNE     3$
707  011352  004737  012324                    JSR     PC,TESTD
708  011356  042700  040000            3$:     BIC     #RPLG,STAT
709  011362  032777  000020  177654            BIT     #LOOP,@SWR
710  011370  001405                            BEQ     4$
711  011372  012737  011404  013112            MOV     #4$,BACK
712  011400  000137  012164                    JMP     EOP
713  011404  000755                    4$:     BR      SOWI
714
715
716                                    ;*****************************************************************
717                                    ;       ROUTINE USED IF "ONE WAY OUT" WAS SELECTED.
718                                    ;       NOTE THAT WHEN IN THIS MODE HALF DUPLEX IS THE ONLY
719                                    ;       MODE AVAILABLE.
720                                    ;       "ONE WAY OUT" MEANS THAT ONLY THE TRANSMITTER IS
721                                    ;       ENABLED, THE RECEIVER IS NEVER "TURNED ON.
722                                    ;*****************************************************************
723
724  011406  104416                    SOWO:   KBDIN
725  011410  004737  013116                    JSR     PC,STARTX
726  011414  005037  011032                    CLR     TIME
727  011420  032700  100000            1$:     BIT     #XPLG,STAT
728  011424  001013                            BNE     2$
729  011426  023727  011032  000100            CMP     TIME,#10P
730  011434  103771                            BLO     1$
731  011436  011402                            MOV     @RCSR,R2
732  011440  016403  000004                    MOV     XCSR(R4),R3
733  011444  104001                            WLT     1
734  011446  005037  011032                    CLR     TIME
735  011452  000762                            BR      1$
736  011454  042700  100000            2$:     BIC     #XPLG,STAT
737  011460  032777  000020  177356            BIT     #LOOP,@SWR
738  011466  001405                            BEQ     3$
739  011470  012737  011502  013112            MOV     #3$,BACK
740  011476  000137  012164                    JMP     EOP
741  011502  000741                    3$:     BR      SOWO
742
743
744
```

B2

```
                                        ;*****************************************************************
                                        ;       ROUTINE USED IF INTERNAL LOOP BACK" WAS SELECTED.
                                        ;       NOTE THAT WHEN IN THIS MODE; HALF DUPLEX IS THE
                                        ;       ONLY MODE AVAILABLE.
                                        ;       "INTERNAL LOOP BACK" MEANS THAT THE RECEIVER IS "TURNED ON"
                                        ;       AND A COMLPETE MESSAGE IS RECEIVED. IF DATA IS TO BE CHECKED
                                        ;       IT IS; IF "END PASS" IS DESIRED; IT IS GIVEN.
                                        ;       THEN THE TRANSMITTER IS ENABLED. AFTER THE WHOLE MESSAGE
                                        ;       IS TRANSMITTED; THE CYCLE IS REPETED AS ABOVE.
                                        ;*****************************************************************

011504  104416                  $ILB1:  KBDIN
011506  004737  013414                  JSR     PC,STARTR
011512  005037  011032                  CLR     TIME
011516  032700  040000          1$:     BIT     #RFLG,STAT
011522  001013                          BNE     2$
011524  023727  011032  000100          CMP     TIME,#100
011532  103771                          BLO     1$
011534  011402                          MOV     #RCSR,R2
011536  016403  000004                  MOV     XCSR(R4),R3
011542  104001                          WLT     1
011544  005037  011032                  CLR     TIME
011550  000762                          BR      1$
011552  032777  000200  177264  2$:     BIT     #NODAT,@SWR
011560  001002                          BNE     3$
011562  004737  012324                  JSR     PC,TESTD
011566  042700  040000          3$:     BIC     #RFLG,STAT
011572  032777  000020  177244          BIT     #LOOP,@SWR
011600  001405                          BEQ     4$
011602  012737  011614  013112          MOV     #4$,BACK
011610  000137  012164                  JMP     END
011614  032777  000400  177222  4$:     BIT     #400,   @SWR    ;USE EXTERNAL DATA?
011622  001416                          BEQ     7$              ;BR IF NO
011624  013702  011020                  MOV     IPDA,   R2      ;SET POINTER
011630  013703  011022                  MOV     TXDA,   R3      ;SET POINTER
011634  010337  011070                  MOV     R3,     XDA     ;SETUP XMIT DATA ADDR
011640  112223                          MOVB    (R2)+,  (R3)+   ;MOVE INPUT TO OUTPUT
011642  001376                          BNE     .-2             ;LOOP IF NOT ZERO CHAR
011644  112743  000177                  MOVB    #177,   -(R3)   ;INSERT A FILL CHAR
011650  005203                          INC     R3              ;BUMP ADDRESS
011652  112723  000177                  MOVB    #177,   (R3)+   ;INSERT ANOTHER FILL
011656  105023                          CLRB    (R3)+           ;INSERT ZERO CHAR
011660  005037  011032          7$:     CLR     TIME
011664  004737  013116                  JSR     PC,STARTX
011670  032700  100000          5$:     BIT     #XFLG,STAT
011674  001013                          BNE     6$
011676  023727  011032  000100          CMP     TIME,#100
011704  103771                          BLO     5$
011706  011402                          MOV     #RCSR,R2
011710  016403  000004                  MOV     XCSR(R4),R3
011714  104001                          WLT     1
011716  005037  011032                  CLR     TIME
011722  000762                          BR      5$
011724  042700  100000          6$:     BIC     #XFLG,STAT
011730  000137  011504                  JMP     $ILB
```

C2

```
                        ;*****************************************************************
                        ;       ROUTINE USED IF "EXTERNAL LOOP BACK" WAS SELECTED.
                        ;       EITHER HALF OR FULL DUPLEX MAY BE SELECTED IN THIS MODE.
                        ;       "EXTERNAL LOOP BACK" MEANS THAT THE TRANSMITTER IS FIRST
                        ;       TURNED ON (IF HALF DUPLEX) AND THE WHOLE MESSAGE IS TRANSMITTED;
                        ;       THEN THE RECEIVER IS ENABLED. AFTER THE WHOLE MESSAGE IS RECEIVED
                        ;       DATA WILL THEN BE CHECKED IF DESIRED AND END PASS WILL
                        ;       BE GIVEN IF DESIRED.    THEN THE CYCLE IS REPEATED
                        ;       AS ABOVE.  IF RUNNING IN FULL DUPLEX THE PROGRAM
                        ;       WAITS FOR BOTH THE RECEIVER AND TRANSMITTER TO
                        ;       FINISH THEN RESTARTS THE RECEIVER AND TRANSMITTER.
                        ;*****************************************************************

A11734  104416          SXL0:   KBDIN
A11736  A32737  000010  A11012          BIT     #HALF,DUPLEX,PARAM1
A11744  001002                          BNE     1$
A11746  004737  A13414                  JSR     PC,STARTR
A11752  004737  A13116          1$:     JSR     PC,STARTX
A11756  005037  A11032                  CLR     TIME
A11762  A32700  100000          2$:     BIT     #XPLG,STAT
A11766  001016                          BNE     3$
A11770  A32700  040000          7$:     BIT     #RPLG,STAT
A11774  001024                          BNE     4$
A11776  023727  A11032  000100          CMP     TIME,#100
A12004  103766                          BLO     2$
A12006  A11402                          MOV     @RCSR,R2
A12010  016403  000004                  MOV     XCSR(R4),R3
A12014  104001                          WLT     1
A12016  005037  A11032                  CLR     TIME
A12022  000757                          BR      2$
A12024  A32737  000010  A11012  3$:     BIT     #HALF,DUPLEX,PARAM1
A12032  001756                          BEQ     7$
A12034  042700  100000                  BIC     #XPLG,STAT
A12040  004737  A13414                  JSR     PC,STARTR
A12044  000746                          BR      2$
A12046  A32737  000010  A11012  4$:     BIT     #HALF,DUPLEX,PARAM1
A12054  001020                          BNE     6$
A12056  A32700  100000                  BIT     #XPLG,STAT
A12062  001413                          BNE     6$
A12064  023727  A11032  000100          CMP     TIME,#100
A12072  103765                          BLO     4$
A12074  011402                          MOV     @RCSR,R2
A12076  016403  000004                  MOV     XCSR(R4),R3
A12102  104001                          WLT     1
A12104  005037  A11032                  CLR     TIME
A12110  000756                          BR      4$
A12112  042700  100000          6$:     BIC     #XPLG,STAT
A12116  042700  040000          8$:     BIC     #RPLG,STAT
A12122  005037  A11032                  CLR     TIME
A12126  032777  000200  176710          BIT     #NODAT,@SWR
A12134  001002                          BNE     5$
A12136  004737  A12324                  JSR     PC,TESTD
A12142  032777  000020  176674  5$:     BIT     #LOOP,@SWR
A12150  001671                          BEQ     SXL0
A12152  012737  A11734  013112          MOV     #SXLB,BACK
A12160  000137  A12164                  JMP     FOP
```

D2

```
856                                     ;*******************************
857                                     ;       ROUTINE TO RETURN
858                                     ;       TO MONITOR FOR
859                                     ;       END PASS.
860                                     ;*******************************
861
862  012164                             EOP:
863  012164  104414  000340                    STPS,PRTY7              ;SET PS PRIORITY TO 7
864  012170  016477  000004  012322            MOV     XCSR(R4),OTPIE  ;SAVE TX CSR
865  012176  042737  177677  012322            BIC     #^C<TIE>,OTPIE  ;CLEAR ALL BUT TX IE.
866  012204  042764  000100  000004            BIC     #TIE,XCSR(R4)   ;CLEAR TX IE (EVEN IF IT WASN'T SET)
867  012212  012766  012252  000002            MOV     #ENTER,2(SP)    ;SET FOR RETURN IF SW 14=1
868  012220  010067  013074                    MOV     R0,SAVR0        ;SAVE REGISTER 0
869  012224  010167  013076                    MOV     R1,SAVR1        ;SAVE REGISTER 1
870  012230  010267  013100                    MOV     R2,SAVR2        ;SAVE REGISTER 2
871  012234  010367  013102                    MOV     R3,SAVR3        ;SAVE REGISTER 3
872  012240  010467  013104                    MOV     R4,SAVR4        ;SAVE REGISTER 4
873  012244  010567  013106                    MOV     R5,SAVR5        ;SAVE REGISTER 5
874  012250  000207                            RTS     PC              ;RETURN TO CONTROL PROGRAM
875
876  012252                             ENTER:
877  012252  013700  013074                    MOV     SAVR0,R0        ;RESTORE R0
878  012256  013701  013076                    MOV     SAVR1,R1        ;RESTORE R1
879  012262  013702  013100                    MOV     SAVR2,R2        ;RESTORE R2
880  012266  013703  013102                    MOV     SAVR3,R3        ;RESTORE R3
881  012272  013704  013104                    MOV     SAVR4,R4        ;RESTORE R4
882  012276  013705  013106                    MOV     SAVR5,R5        ;RESTORE R5
883  012302  012737  177777  013110            MOV     #-1,DELAY
884  012310  052764  012322  000004            BIS     OTPIE,XCSR(R4)  ;IF ORGINALLY SET; SET TX IE
885  012316  000177  000578                    JMP     @#ACK
886  012322  000000                    OTPIE:  .WORD   0
887
888
889
890                                     ;*******************************
891                                     ;       SUBROUTINE TO CHECK
892                                     ;       RECEIVER DATA.
893                                     ;*******************************
894  012324  013746  011056            TESTD:  MOV     ERROR,  -(SP)   ;WAS THERE A RECEIVE ERROR?
895  012330  001413                            BEQ     TSTDAT          ;BR IF NO
896  012332  032777  020000  176584            BIT     #BIT13,@#WR     ;INHIBIT PRINTOUTS?
897  012340  001007                            BNE     TSTDAT          ;BR IF YES
898  012342  104400  012534                    TYPE    ,#MSG8          ;<15><12>THERE WAS A RECEIVE ERROR, RBUF=
899  012346  004077  176456                    JSR     R0,@#2016       ;PRINT CONTENTS OF RBUF
900  012352  005746                            TST     -(SP)
901  012354  104400  012615                    TYPE    ,#MSG1          ;<15><12>
902  012360  013701  011022            TSTDAT: MOV     INDA,   R1      ;SETUP XMIT DATA ADDR
903  012364  013702  011020                    MOV     IND,    R2      ;SETUP RCV DATA ADDR
904  012370  122122                    SCAN4:  CMPB    (R1)+,  (R2)+   ;DATA OK ?
905  012372  001776                            BEQ     SCAN4           ;BR IF OK
906  012374  123741  011040                    CMPB    TX.TERM,-(R1)   ;IS IT END OF DATA
907  012400  001403                            BEQ     TESTDX          ;BR IF YES
908  012402  122742  000002                    CMPB    #2,-(R2)
909  012406  001005                            BNE     2$
910  012410  012737  012416                    MOV     #2,1$
911  012416  104400                            TYPE
```

E2

```
912  012416  000000                  18:     .WORD   0
913  012420  000443                          RR      TESTDX
914  012422                          28:
915  012422  105712                          TSTB    (R2)            ;
916  012424  001441                          BEQ     TESTDX          ;BR IF YES
917  012426  122721  000177                  CMPB    #177,   (R1)+   ;IS IT FILL CHAR?
918  012432  001756                          BEQ     SCAN4           ;BR IF YES
919  012434  005301                          DEC     R1              ;BACKUP
920  012436  122722  000177                  CMPB    #177,   (R2)+   ;IS IT FILL?
921  012442  001752                          BEQ     SCAN4           ;BR IF YES
922  012444  105742                          TSTB    -(R2)           ;BACK UP POINTER
923  012446  123722  011014                  CMPB    PARAM2,(R2)+
924  012452  001746                          BEQ     SCAN4           ;BR IF CHAR WAS SYNC
925  012454  000240                  SCANS:  NOP                     ;DATA ERROR
926  012456  032777  020000  176360          BIT     #BIT13,@SWR     ;INHIBIT PRINTOUTS
927  012464  001016                          BNE     DERR            ;BR IF YES
928  012466  104400  012620                  TYPE    ,MSG2           ;<15><12>RECEIVED DATA = <15><12>
929  012472  013737  011020  012502          MOV     IXDA,   RDAX    ;SETUP DATA ADDRESS
930  012500  104400                          TYPE                    ;PRINT RECEIVED DATA
931  012502  000000                  RDAX:   0                       ;RECEIVED DATA ADDR.
932  012504  104400  012645                  TYPE    ,MSG3           ;<15><12>DATA SHOULD BE<15><12>
933  012510  013737  011022  012520          MOV     IXDA,   .+10    ;SETUP ADDR.
934  012516  104400                          TYPE                    ;PRINT GOOD DATA
935  012520  011022                          IXDA
936  012522  111103                  DERR:   MOVB    (R1),R3 ;SETUP XMIT DATA
937  012524  114202                          MOVB    -(R2),R2        ;SETUP RCV DATA
938  012526  104007                          HLT+7                   ;DATA ERROR HALT
939  012530  005726                  TESTDX: TST     (SP)+           ;POP STACK
940  012532  000207                          RTS     PC              ;RETURN FROM SUB/ROUT
941
942  012534  050015  044124  051105  MSG0:   .ASCIZ  <15><12>/THERE WAS A RECEIVER ERROR, REGISTER (SEL 2) =/
(1)  012615      015  000012          MSG1:   .ASCIZ  <15><12>
(1)  012620  050015  042522  042503  MSG2:   .ASCIZ  <15><12>/RECEIVED DATA = /<15><12>
(1)  012645      015  042012  052101  MSG3:   .ASCIZ  <15><12>/DATA SHOULD BE/<15><12>
(1)  012670  050015  046120  040505  MSG4:   .ASCII <15><12>/PLEASE MAKE CONNECTION (DIAL NUMBER)./
(1)  012757      015  053412  042510          .ASCIZ <15><12>/WHEN CONNECTION COMPLETE; HIT CONTINUE SWITCH./<15><12>
(1)  013022  050015  046120  040505  MSG5:   .ASCIZ <15><12>/PLEASE MAKE CONNECTION (DIAL NUMBER)./<15><12>
(1)                                  .EVEN
(1)  013074  000000                  SAVR0:  0
943  013076  000000                  SAVR1:  0
944  013100  000000                  SAVR2:  0
945  013102  000000                  SAVR3:  0
946  013104  000000                  SAVR4:  0
947  013106  000000                  SAVR5:  0
948  013110  000000                  DELAY:  0
949  013112  000000                  BACK:   0
950  013114  000000                  STOP:   0
951
```

F2

```
                                ;*****************************************************************
                                ;       INITIALIZE TRANSMIT SUBROUTINE
                                ;*****************************************************************
                                
013116  005037  011032          STARTX: CLR     TIME
013122  005737  013110                  TST     DELAY   ;IF SW04=1 & SW14=0 WAIT BEFORE TURNING ON TX
013126  001416                          BEQ     1$      ;NO GO AHEAD AND TURN ON TX
013130  005037  013504                  CLR     TEMP1   ;PREPARE FOR DELAY
013134  012737  000007  013506          MOV     #7,TEMP2
013142  062737  000001  013504          ADD     #1,TEMP1        ;INCREMENT DELAY....
013150  001374                          BNE     .-6
013152  005337  013506                  DEC     TEMP2
013156  001371                          BNE     .-14
013160  005037  013110                  CLR     DELAY   ;ZERO DELAY
013164  013737  011022  011070  1$:     MOV     TXDA,XDA        ;SETUP XMIT DATA ADDR.
013172  042700  100000                  BIC     #XPLG,R0        ;RESET XMIT COMPLETE FLAG
013176  052714  000002                  BIS     #DTR,@RXCSR     ;SET DTR
013202  005737  013114                  TST     STOP
013206  001005                          BNE     2$
013210  104400  012670                  TYPE    ,MSG4
013214  000000                          HALT
013216  005137  013114                  COM     STOP
013222  032737  000010  011012  2$:     BIT     #WALF,DUPLEX,PARAM1
013230  001403                          BEQ     3$
013232  032714  100000          4$:     BIT     #100000,@RXCSR  ;IS CARRIER UP
013236  001375                          BNE     4$      ;BR IF YES
013240  052714  000004          3$:     BIS     #RTS,@RXCSR     ;SET RTS
013244  032714  020000          CTSW1:  BIT     #CTS,@RXCSR;IS CLEAR TO SEND SET?
013250  001017                          BNE     CTSOK           ;BR IF YES
013252  023727  011032  000036  CTSWN1: CMP     TIME,#36        ;30 SECS ELASPED?
013260  103771                          BLO     CTSW            ;BR IF NO
013262  011402                          MOV     @RXCSR,R2       ;SETUP RECEIVE CSR
013264  016403  000004                  MOV     TXCSR(R4),R3    ;SETUP XMIT CSR
013270  032777  010000  175546          BIT     #SW12,@SWR      ;INHIBIT PRINTOUTS?
013276  001001                          BNE     1$              ;BR IF YES
013300  104402                          MLT+2                   ;PRINTOUT "WAITING TO XMIT" MESSAGE
013302  005037  011032          1$:     CLR     TIME            ;RESET TIMER
013306  000756                          BR      CTSW            ;WAIT SOME MORE
013310  005037  011032          CTSOK:  CLR     TIME
013314  012737  000005  013412          MOV     #5,SYNCNO               ;SETUP SYNC. COUNTER
013322  032737  000010  011012          BIT     #WALF,DUPLEX,PARAM1
013330  001404                          BEQ     2$
013332  052764  000030  000004          BIS     #SEND!MOXEN,TXCSR(R4)
013340  000403                          BR      1$
013342  012764  000020  000004  2$:     MOV     #SEND,TXCSR(R4) ;SET SEND
013350  013764  011014  000006  1$:     MOV     PARAM2,TXDBUF(R4);LOAD A SYNC CHAR

013356  105764  000004                  TSTB    TXCSR(R4)       ;IS XMIT READY(TXDONE=1?)
013362  100375                          BPL     .-6             ;BR IF NO
013364  005337  013412                  DEC     SYNCNO          ;DECREMENT COUNTER
013370  001367                          BNE     1$              ;BR IF NOT ZERO
013372  042764  000400  000006          BIC     #TADM,TXDBUF(R4)
013400  052764  000100  000004          BIS     #TXINTE,TXCSR(R4);SET XMIT INTERRUPT ENABLE
013406  000240                          NOP
013410  000207                          RTS     PC              ;EXIT FROM SUBROUTINE
013412  000000          SYNCNO: 0
```

G2

```
1008
1009                                    ;*****************************************************************
1010                                    ;       INITIALIZE RECEIVER SUBROUTINE
1011                                    ;*****************************************************************
1012  013414                            START0: 
1013  013414  005737  013114                   TST     STOP            ;FIRST TIME HERE?
1014  013420  001006                           BNE     10              ;BR IF NO
1015  013422  052714  000002                   BIS     #DTR,@RXCSR     ;SET DTR
1016  013426  104400  013022                   TYPE    ,MSG5           ;MAKE CONNECTION
1017  013432  005137  013114                   COM     STOP
1018  013436  005037  011032           10$:    CLR     TIME
1019  013442  013737  011020  011066          MOV     IRDA,RDA        ;SETUP RCV DATA ADDR
1020  013450  012737  001000  011064          MOV     #1000,RCC       ;SETUP RCV CHAR COUNT
1021  013456  042700  000000                   BIC     #RFLG,R0        ;RESET RCV COMPLETE FLAG
1022  013462  005037  011050                   CLR     ERCSR           ;RESET ERROR RECORDS
1023  013466  005037  011052                   CLR     ERDAR
1024  013472  005764  000002                   TST     RXDBUF(R4)      ;CLR ANY RXDONE
1025  013476  052714  000522                   BIS     #RINTEN!RCVEN!STRPSYN!DTR,@RXCSR ;SET INTER ENABLE & STRIP SYNC &DTR
1026  013502  000207                           RTS     PC              ;EXIT FROM SUBROUTINE
1027  013504  000000                   TEMP1:  0
1028  013506  000000                   TEMP2:  0
```

```
1029                                    ;*****************************************************************
1030                                    ;       TRANSMIT INTERRUPT SERVICE ROUTNE
1031                                    ;*****************************************************************
1032  013510  000240          XISR:   NOP
1033  013512  127737  175352  011040          CMPB    @XDA,TX,TERM            ;FINISHED XMITTING?
1034  013520  001010                  BNE     XISR1           ;BR IF NO
1035  013522  052700  100000          BIS     @XFLG,R0        ;SET XMIT COMPLETE FLAG
1036  013526  042764  000120  000004          BIC     @TXINTF!SEND,TXCSR(R4)  ;RESET XMIT INTR ENABLE
1037  013534  042714  000004          BIC     @RTS,@RXCSR     ;RESET RTS
1038  013540  000417                  BR      XISR2           ;
1039                                    ;
1040  013542  117764  175322  000006  XISR1:  MOVB    @XDA,TXDBUF(R4);XMIT NEXT CHAR.
1041  013550  032777  000100  175266          BIT     #100,@SWR       ;MONITOR OUTPUT?
1042  013556  001406                  BEQ     NOXMON          ;BR IF NO
1043  013560  105777  175312          TSTB    @TPS    ;IS TTY AVAILABLE
1044  013564  100003                  BPL     NOXMON          ;BR IF NO
1045  013566  117777  175276  175304          MOVB    @XDA,@TPB;TYPE THE CHAR
1046  013574                  NOXMON:
1047  013574  005237  011070          INC     XDA             ;INCREMENT ADDRESS
1048  013600  005037  011032  XISR2:  CLR     TIME            ;RESET TIMER
1049  013604  000002                  RTI                     ;RETURN FROM INTERRUPT
1050                                    ;*****************************************************************
1051                                    ;       RECEIVE INTERRUPT SERVICE ROUTINE
1052                                    ;*****************************************************************
1053  013606  000240          RISR:   NOP
1054  013610  011402                  MOV     @RXCSR,R2       ;SAVE RXCSR CONTENTS
1055  013612  016403  000004          MOV     TXCSR(R4),R3    ;SAVE TXCSR CONTENTS
1056  013616  016437  000002  013504  MOV     RXDBUF(R4),TEMP1        ;STORE CHAR AND ERROR FLAGS IF ANY
1057  013624  113737  013504  013506  MOVB    TEMP1,TEMP2
1058  013632  113777  013504  175226          MOVB    TEMP1,@RDA      ;MOVE CHAR TO INBUF
1059  013640  032777  000040  175176          BIT     #40,@SWR        ;MONITOR INPUT?
1060  013646  001406                  BEQ     NORMON          ;BR IF NO
1061  013650  105777  175222          TSTB    @TPS    ;IS TTY AVAILABLE?
1062  013654  100003                  BPL     NORMON          ;BR IF NO
1063  013656  113777  013506  175214          MOVB    TEMP2,@TPB;TYPE THE CHAR
1064  013664                  NORMON:
1065  013664  005237  011066          INC     RDA             ;BUMP POINTER
1066  013670  105077  175172          CLRB    @RDA            ;CLEAR NEXT CHAR POSITION
1067  013674  005337  011064          DEC     RCC             ;DECREMENT CHAR. COUNTER
1068  013700  001005                  BNE     1$              ;BR IF BUFFER NOT FULL
1069  013702  042714  000520          BIC     #RINTEN!RCVEN!STPSYN,@RXCSR     ;RESET INTERRUPT ENAB
1070  013706  104006                  HLT+6                   ;RECEIVER BUFFER FULL
1071  013710  004737  013414          JSR     PC,STARTR       ;INITIALIZE RECEIVER
1072  013714  123737  011041  013506  1$:     CMPB    RX,TERM,TEMP2   ;IS IT LINE FEED?
1073  013722  001004                  BNE     RISR1           ;BR IF NO
1074  013724  042714  000520          BIC     #RINTEN!RCVEN!STPSYN,@RXCSR     ;DISABLE INTERRUPT
1075  013730  052700  040000          BIS     #RFLG,R0        ;SET RCVR COMPLETE FLAG
1076  013734  005737  013504  RISR1:  TST     TEMP1   ;IS THERE A DATA ERROR
1077  013740  100005                  BPL     RISR2           ;BR IF NO
1078  013742  010237  011054          MOV     R2,ERCSR        ;SAVE RXCSR
1079  013746  013737  013504  011056  MOV     TEMP1,ERDBR     ;SAVE RXDBUF
1080  013754  005702          RISR2:  TST     R2      ;IS THERE A DATA SET STATUS CHANGE
1081  013756  100004                  BPL     RISR3   ;BR IF NO
1082  013760  010237  011060          MOV     R2,DSSTAT       ;SAVE STATUS
1083  013764  052700  020000          BIS     #DSFLG,R0       ;SET FLAG
1084  013770                  RISR3:
```

I.2

```
  1005  013770  005037  011032                  CLR     TIME            ;RESET TIMER
  1006  013774  000002                          RTI                     ;RETURN FROM INTERRUPT
  1007          000001                  .END
```

```
ABORT = 002000
BA      011004
BACK    013112
BITW  = 002000
BIT0  = 000001
BIT1  = 000002
BIT10 = 002000
BIT11 = 004000
BIT12 = 010000
BIT13 = 020000
BIT14 = 040000
BIT15 = 100000
BIT2  = 000004
BIT3  = 000010
BIT4  = 000020
BIT5  = 000040
BIT6  = 000100
BIT7  = 000200
BIT8  = 000400
BIT9  = 001000
B2016   011030
CARDET= 010000
CLK   = 020000
CRCEN = 001000
CRCERR= 010000
CTS   = 020000
CTSOK   013310
CTSW    013244
CTSWN   013252
DECMOD= 100000
DELAY   013110
DERR    012522
DISPLA  011046
DNA   = 100000
DSC   = 100000
DSC2  = 000001
DSFLG = 020000
DSINTE= 000040
DSR   = 001000
DSSTAT  011060
DTR   = 000002
DUP11   011000
ENTER   012252
EOP     012164
ERCSR   011054
ERDBR   011056
FLAG    011042
FULL.D= 000001
GO      011210
HALF.D= 000010
HDXEN = 000010
ILS   = 000010
IRDA    011020
IXDA    011022
KBDIN = 104416
LOOP  = 000020
NEXT  = 010000
NINT  = 014000
NMODEA= 004000
NMODEB= 010000
NODE  =%000000
NRESET= 000400
NSS0    012534
NSS1    012614
NSS2    012620
NSS3    012644
NSS4    012670
NSS5    013022
NXDATA= 040000
NODAT = 000200
NORMON  013664
NOXMON  013574
OVRRUN= 040000
OWI   = 000002
OWO   = 000001
PARAM1  011012
PARAM2  011014
PARAM3  011016
PARCSR= 000002
PRIOR   011010
PRISEC= 010000
PRTV0 = 000000
PRTV1 = 000040
PRTV2 = 000100
PRTV3 = 000140
PRTV4 = 000200
PRTV5 = 000240
PRTV6 = 000300
PRTV7 = 000340
QTPIE   012322
RABORT= 002000
RCC     011064
RCSR  =%000004
RCVEN = 000020
RDA     011066
RMAX    012502
RECACT= 004000
REOM  = 001000
RFLG  = 040000
RING  = 040000
RINTEN= 000100
RISR    013606
RISR1   013734
RISR2   013754
RISR3   013770
RIV     011006
RSOM  = 000400
RTS   = 000004
RWAIT = 104410
RXCSR =%000004
RXDBUF= 000002
RXDONE= 000200
RXERR = 100000
RX.TER  011041
R6    =%000006
R7    =%000007
SAVR0   013074
SAVR1   013076
SAVR2   013100
SAVR3   013102
SAVR4   013104
SAVR5   013106
SCAN4   012370
SCAN5   012454
SEND  = 000020
SETTLE  011024
SRCSR   011052
SRD   = 002000
START   011102
STARTR  013414
STARTX  013116
STAT  =%000000
STD   = 000010
STOP    013114
STPS  = 104414
STPSYN= 000400
SWR     011044
SW12  = 010000
SXCSR   011050
SYNCNO  013412
SYSTST= 004000
TEMP1   013504
TEMP2   013506
TEOM  = 001000
TESTB   012324
TESTDX  012330
TIE   = 000100
TIME    011032
TKB     011074
TKS     011072
TPB     011100
TPS     011076
TSOM  = 000400
TSTDAT  012360
TXACT = 001000
TXCSR = 000004
TXDBUF= 000006
TXDONE= 000200
TXINTE= 000100
TX.TER  011040
TYPE  = 104400
USER  = 000000
XCC     011062
XCSR  = 000004
XDA     011070
XFLG  = 100000
XISR    013510
XISR1   013542
XISR2   013600
XLB   = 000004
XWAIT = 104412
SILB    011504
SOWI    011300
SOWO    011406
SXLB    011734
.     = 013776

. ABS.  013776      000
```

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

,DUP/SOL:ITEP1,MAC,DZDPF0.P11
RUN-TIME: 10 13 .4 SECONDS
RUN-TIME RATIO: 67/24=2.7
CORE USED: 15K (29 PAGES)

K2