

# DH11

OVERLAY FOR INTERPRO TEST  
MD-11-DZDHL-B

EP DZDHL B DL  
COPYRIGHT 1978  
FICHE 1 OF 1

JAN 1978  
**digital**  
MADE IN USA



IDENTIFICATION  
.....

PRODUCT CODE:            MAINDEC-11-DZDHL-R-D  
  
PRODUCT NAME:            DH11 OVERLAY FOR INTERPROCESSOR TEST PROGRAM  
  
PROGRAM DATE:            OCTOBER 1976  
  
MAINTAINER:              DIAGNOSTICS  
  
AUTHORS:                 R A JONES  
                           JOHN FOLF  
                           FAY BASHAW

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED UNDER A LICENSE AND MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE.

DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.

COPYRIGHT (C) 1975, 1976, BY DIGITAL EQUIPMENT CORPORATION

1.0 ABSTRACT.

THIS PROGRAM IS DESIGNED AS A MAINTENANCE AID FOR FIELD SERVICE PERSONEL. 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 DN11 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 DN11PB, TYPE IN DMB.

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

- 2. THE PROGRAM WILL TYPE THE DEFAULT BUS ADDRESS OF THE INTERFACE UNDER TEST.
  - A. TYPE A CAP. RETURN TO USE DFFAULT BUS ADDRESS
  - B. TYPE IN ACTUAL BUS ADDRESS
- 3. THE PROGRAM WILL TYPE OUT THE DEFAULT VECTOR ADDRESS
  - A. TYPE A CAP. RETURN TO USE DEFAULT ADDRESS
  - B. TYPE IN ACTUAL VECTOR ADDRESS

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

- A. TYPE A CAR, RETURN TO USE DEFAULT VALUE
  - B. TYPE IN ACTUAL VALUE
5. THE PROGRAM WILL TYPEOUT THE DEFAULT VALUE OF PARAM#1  
IF REQUIRED BY THE ISP, (SEE SECT. 10.0 IN OVERLAY LISTING FOR PARAMETER DESCRIPTION)
- A. TYPE A CAR, RETURN TO USE DEFAULT VALUE
  - B. TYPE IN ACTUAL VALUE
6. THE PROGRAM WILL TYPEOUT THE DEFAULT VALUE OF PARAM#2  
IF REQUIRED BY THE ISP.
- A. TYPE A CAR, RETURN TO USE DEFAULT VALUE
  - B. ENTER ACTUAL VALUE
7. THE PROGRAM WILL TYPEOUT THE DEFAULT VALUE OF PARAM#3  
IF REQUIRED BY THE OVERLAY.
- A. TYPE A CAR, RETURN TO USE DEFAULT VALUE  
THE DN-11 WILL USE PARAM #3 AS THE # TO DIAL.  
IF USING A MODEM WITHOUT AUTOMATIC HANDSHAKING,  
THE NUMBER MUST TERMINATE WITH A  
"END-OF-NUMBER" CHARACTER (1).
  - B. ENTER ACTUAL VALUE.
8. THE PROGRAM WILL RETURN TO STEP 61 IF THIS SETUP  
WAS FOR DN11 OR DN11BB.
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 ISP (INTERFACE SERVICE ROUTINE) SPECIFICATION  
SAR14=SETUP DN=11R ISP  
SAR13=SETUP DN=11 ISP  
SAR00000000=SETUP VARIABLE ISP
  2. THE FOLLOWING HALTS ARE REPEATED FOR EACH ISP SPECIFIED.  
SETUP SEQUENCE IS: DN11, DN11-RR THEN VARIABLE OVERLAY. (EACH ENTRY SET SWITCHES 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. 10.0 IN OVERLAY LISTING FOR PARAMETER DESCRIPTION)
    - E. HALT FOR INTERFACE PARAM #2 (DN11 AND DNBB PARAMETERS ARE DISCUSSED IN SECT. 10.0 OF THE MONITOR.)
    - F. GO BACK TO STEP A IF THIS SETUP WAS FOR DN OR DNBB.
  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).

D. OPERATIONAL SWITCH SETTINGS.

- SW15=1 HALT ON ERROR
- SW14=1 SINGLE PASS
  - S=14 HAS NO EFFECT IF SW14=0
- SW13=1 INHIBIT ERROR TIMEOUTS
- SW12=1 INHIBIT ALL TIMEOUTS EXCEPT ERRORS  
IF SW12=0 AND SW14=1 END PASS IS TYPED  
AND TRANSMITTED/RECEIVED DATA IS TYPED.
- SW11=1 USE PREVIOUSLY SPECIFIED DATA
- SW10=1 DATA SELECT (WITH SW10)
- SW09=1 DATA SELECT (WITH SW10)
  - 00=1 GET DATA FROM OPERATOR
  - 01=1 TEST MESSAGE #1 (8A QUICK BROWN FOX)
  - 10=1 TEST MESSAGE #2 (8B NUMERICS)
  - 11=1 TEST MESSAGE #3 (8C 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 SW14=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

(IF) S=P=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 (CNTRL 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) EPROP 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.    APCD^ 000 123 377^ FFG (CAP,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 03.

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 1-3) ARE PRECEDED BY 2 FILL CHARACTERS(177), AND ARE FOLLOWED BY A CR(015), LF(012), RECEIVE TERMINATING CHARACTER(001), 4 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 <CR>)
2. VERIFIES THE DATA AGAINST THE DATA SELECTED BY SW09 AND SW10(SW7=0)
3. TRANSMIT THE DATA SELECTED BY SW09 AND SW10 (SW0=0) OR TRANSMIT THE RECEIVED DATA (SW0=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 CSP EVERY 60 SFCS.  
UNTIL CLEAR TO SEND IS ASSERTED.



F. IF SW04=0 THE OVERLAY WILL CONTINUE TO TRANSMIT/RECEIVE DATA.

IF SW04=1 THE OVERLAY WILL RETURN TO THE MONITOR AND TYPE "END PASS".

IF BOTH SW04=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:

\*XXXXXX                      TO PRINTOUT THE 8 WORDS AT LOC XXXXXX.

\*BXXXXX                      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/OP OPERATOR ACTION

IF THE OPERATOR WISHES TO MANUALLY 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=RRRPRP  
DATA SHOULD BE TTITTT  
DATA COMPARE ERROR; BAD DATA=BRB GOOD DATA=GCG

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 RECEIVED 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 LOOPBACK MODE.

BOTH SYSTEMS SHOULD BE RUNNING IDENTICAL ROUTINES.

EXAMPLE:

SWITCHES 14,13,7,4 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 TIMEOUT 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 ASYNCHRONOUS DEVICE, MODEMS AND THE  
MAYNARD TEST STATION AND INITIALIZE 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-

11

CEIVED. 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 DH11 INTERFACE SERVICE PAPAMS ARE SETUP, AS SPECIFIED BY THE OPERATOR, BY THE ITEP CONTROL PROGRAM.

TIME; PROVIDES A MEANS OF MEASURING ELAPSED 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 P. 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. XLB, 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 60 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 XLB; 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 8XLB.  
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 8XLB DEPENDING  
ON THE SWITCH SETTINGS.

- 9.4 THE RETURN TO MONITOR ROUTINE FOR END PASS AT EOP;  
LOCKS OUT INTERRUPTS AND SAVES THE TRANSMITTER INTERRUPT 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(0W0,0W1,ILA,XLR) WHERE IT CAME FROM.
- 9.6 THE INITIALIZE TRANSMIT SUBROUTINE AT STARTX;  
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 STARTR;  
SETS UP THE INTERFACE AND POINTERS NECESSARY TO  
RECEIVE A MESSAGE.
- 9.8 THE TRANSMIT INTERRUPT SERVICE ROUTINE,  
AT XISR; 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 XISR; 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

M1

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 RISP<sub>i</sub>, THE  
CSR AND DBR WILL BE SAVED AND PRINTED OUT  
AFTER THE COMPLETE MESSAGE HAS BEEN RECEIVED.

9.10 THE DATA TEST SUBROUTINE AT TESTD<sub>i</sub> 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.

#### DH11 RESTRICTIONS

IF A DM11RB EXISTS IN THE SYSTEM WITH THE DH11 BEING  
TESTED, BUT MODEM CONTROL IS NOT DESIRED AND THE DM11RB  
WAS NOT INITIALIZED BY ITEP, THE PROGRAM WILL HANG IN THE  
DH11 TRANSMITTER INITIALIZATION ROUTINE. TO CORRECT THIS  
LOAD LOCATION "DM11B" WITH AN ADDRESS THAT WILL TIME OUT (NO  
SLAVE SYNC RESPONSE). THE ADDRESS OF DM11B CAN BE FOUND  
IN THE CROSS REFERENCE TABLE IN THE BACK OF THIS LISTING.

575

#### 10.0 PARAMETERS FOR THE DH11

PARAM#1 IS LOADED INTO THE SYSTEM CONTROL REGISTER.(SCR)  
BITS 0-3      LINE SELECTION, DEFAULT= LINE 0 (0000)

PARAM#2 IS LOADED INTO THE LINE PARAMETER REGISTER.(LPR)  
BITS 0,1      CHARACTER LENGTH, DEFAULT= 8 BITS (11)  
BIT 2      STOP BITS, DEFAULT= 2 STOP BITS (1)  
BIT 4      PARITY ENABLED (1), DEFAULT= (0)  
BIT 5      ODD PARITY (1), DEFAULT= (0)  
BITS 6-9      RECEIVER SPEED, DEFAULT= 110 BAUD (0011)  
BITS 10-13      TRANSMIT SPEED, DEFAULT= 110 BAUD (0011)  
BIT 14      HALF DUPLEX (1), DEFAULT= FULL DUPLEX (0)

PARAM#3 IS NOT USED (177777)

```

595
596 ;.....
597 ; DH11 INTERFACE SERVICE PARAMS
598 ;.....
599          #11000          ,#11000
600 011000 044104 000040 DH11: .ASCIZ /DH /          ;ISR NAME
601 011004 160020          BA: 160020          ;BUS ADDRESS
602 011006 000300          RIV: 300          ;VECTOR ADDRESS
603 011010 000240          PRIOR: 240          ;PRIORITY
604 011012 000000          PARAM1: 0          ;PARAM 01
605 011014 006307          PARAM2: 006307          ;PARAM 02
606 011016 177777          PARAM3: 177777          ;PARAM 03
607 011020 000000          IRDA: .WORD 0          ;INITIAL READ DATA ADDRESS
608 011022 000000          IXDA: .WORD 0          ;INITIAL XMIT DATA ADDRESS
609 011024 000000          SETTLE: .WORD 0          ;LINE SETTLE DELAY FLAG
610 011026 000000          ;
611 011030 000000          B2016: .WORD 0          ;ADDR OF BIN TO OCT TYPE ROUTINE
612 011032 000000          TIME1: .WORD 0          ;TIMER
613 011034 000000          ;
614 011036 011102          ;ADDR OF START OF PROGRAM
615 011040          TX_TERM:          ;
616 011042          .BYTE 000          ;TRANSMITTED TERMINATING CHAR.
617 011044          ;
618 011046          .BYTE 001          ;RECEIVER TERMINATING CHAR.
619 011048 000000          FLAG: .WORD 0
620 011044 177570          SWR: 177570
621 011046 177570          DISPLAY: 177570
622
623 ;.....
624 ; CONSTANTS + WORKING STORAGE
625 ;.....
626          000000          STAT=0
627          100000          XFLG=100000          ;XMIT COMPLETE FLAG
628          040000          RFLG=40000          ;RCV COMPLETE FLAG
629          020000          DSFLG=20000          ;DATA SET STATUS CHANGE FLAG
630          020000          BIT13=20000          ;INHIBIT PRINTOUTS
631
632 011050 000000          SXCSR: 0          ;SAVED XMIT CSR
633 011052 000000          SRCSP: 0          ;SAVED RCV CSR
634 011054 000000          ERCSF: 0          ;RCV CSF SAVED ON ERROR
635 011056 000000          ERDBP: 0          ;RCV DATA REG SAVED ON ERROR
636 011060 000000          DSSTAT: 0          ;RCV CSR SAVED ON DS CHANGE
637
638 011062 000000          XCC: 0          ;XMIT CHAR COUNT
639 011064 000000          RCC: 0          ;RCV CHAR COUNT
640 011066 000000          PDA: 0          ;RCV DATA ADDR.
641 011070 000000          XDA: 0          ;XMIT DATA ADDR.
642
643 011072 177560          TFS: 177560
644 011074 177562          TKB: 177562
645 011076 177564          TPS: 177564
646 011100 177566          TPB: 177566
647
648          000001          FULL_DUPLEX=000001
  
```

M

```

649 ;.....
650 ;   DH11-X INTERFACE SERVICE ROUTINE
651 ;.....
652 #11102 000240 START: NOP
653 #11104 #17700 177734 MOV     #SWP,  R0      ;SETUP MODE IN R0
654 #11110 042700 177400 BIC     #177400,  R0   ;STRIP JUNK
655 011114 013702 #11006 MOV     RIV,   R2      ;SETUP
656 #11120 012722 014060 MOV     #PISR, (R2)+  ;INTERRUPT
657 011124 #13722 #11010 MOV     PRIOR, (R2)+  ;VECTORS
658 #11130 012722 013510 MOV     #XISR, (R2)+  ;
659 #11134 #13722 #11010 MOV     PPOR,  (R2)+  ;
660 011140 013704 #11004 MOV     BA,    R4      ;SETUP BUS ADDR INDEX
661 011144 #13714 011012 MOV     PARAM1, #PCSR ;SETUP VARIABLES
662 #11150 #13702 #11014 MOV     PARAM2,R2      ;
663 011154 042702 #11001 BIC     #0001,R2      ;
664 011160 010264 #00000 MOV     R2,    XCSR(R4); IN CSR'S
665 #11164 #12714 #14000 MOV     #MC, #PCSR
666 011170 #53714 #11012 BIS     PARAM1, #PCSR
667 011174 #53764 #11014 HIS     PARAM2, LPR(R4)
668 #11202 123727 011012 #00004 #00017 CMPH   PARAM1, #17
669 011210 101402 HLOS   16
670 #11212 000000 HALT
671 #11214 #20776      BP     #-2           ;NUMBER IN PARAM1 IS TOO LARGE!
672 011216 #14046      18:  MOV     R0, -(SP)      ;MUST BE 17 OR LESS (LINE# IN OCTAL)
673 #11220 012700 #00001      MOV     #1, R0      ;SAVE R0
674 011224 013701 #11012      MOV     PARAM1, R1
675 #11230 #05701      28:  TST     R1
676 #11232 #01403      RFO    38           ;CALCULATE PAP BIT
677 #11234 #06300      ASL   R0
678 011236 #05301      DEC   R1
679 #11240 #00773      BR    28
680 #11242 #10037 013734 38:  MOV     R0, BARTMP
681 #11246 #12600      MOV     (SP)+, R0
682
683 ;.....
684 ;   ROUTINE USED TO GOTO
685 ;   SUBROUTINE DEPENDENT
686 ;   ON MODE SELECTED.
687 ;.....
688
689 #11250 #05037 011032 G0:   CLR     TIME
690 #11254 #05037 013140      CLR     DELAY
691 011260 #05037 013144      CLR     STOP
692 #11264 #32700 #100001      BIT     #000, MODE
693 #11270 #01402      BEO    16
694 #11272 #00137 011446      JMP     #000
695 011276 #32700 #00002      18:  HIT     #001, MODE
696 #11302 #01402      RFO    28
697 #11304 #00137 011340      JMP     #001
698 #11310 #32700 #100010      28:  HIT     #100, MODE
699 #11314 #01402      BEO    38
700 #11316 #00137 011544      JMP     #100
701 011322 #32700 #100004      38:  HIT     #100, MODE
702 #11326 #01402      BEO    48
703 #11330 #00137 011774      JMP     #100
704 #11336 #10000      48:  HALT
    
```

705 011336 000776 BR -2

706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719

```

;.....
; 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
; ENABLD. THE TRANSMITTER IS NEVER "TURNED ON".
;.....
    
```

720	011340	104416				80WI:	KBDIN	
721	011342	004737	013746				JSP	PC,STAPTR
722	011346	032700	040000			18:	BIT	0RFLG,STAT
723	011352	001013					BNE	28
724	011354	023727	011032	000100			CMP	TIME,0100
725	011362	103771					BLO	18
726	011364	011402					MOV	0RCSR,R2
727	011366	016403	000000				MOV	XCSR(R4),R3
728	011372	104001					HLT	1
729	011374	005037	011032				CLR	TIME
730	011300	000762					BR	18
731								
732	011402	032777	000200	177434		28:	BIT	0NODAT,0SWP
733	011410	001002					BNE	38
734	011412	004737	012364				JSP	PC,TESTD
735	011416	042700	040000			38:	HIC	0RFLG,STAT
736	011422	032777	000020	177414			BIT	0LOOP,0SWR
737	011430	001405					RFQ	48
738	011432	012737	011444	013142			MOV	048,BACK
739	011440	000137	012224				JMP	EOP
740	011444	000735				48:	BP	80WI

741  
742  
743  
744  
745  
746  
747  
748  
749

```

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

750								
751	011446	104416				80WI:	KBDIN	
752	011450	004737	013146				JSP	PC,STARTX
753	011454	005037	011032				CLR	TIME
754	011460	032700	100000			18:	BIT	0XFLG,STAT
755	011464	001013					BNE	28
756	011466	023727	011032	000100			CMP	TIME,0100
757	011474	103771					BLO	18
758	011476	011402					MOV	0RCSR,R2
759	011500	016403	000000				MOV	XCSR(R4),R3
760	011504	104001					HLT	1

B2



761	011506	005037	011032			CLR	TIME
762	011512	000702				BR	18
763	011514	042700	100000	281		BIC	0XFLG,STAT
764	011520	032777	000020	177316		BIT	0LOOP,0SWR
765	011526	001425				BFO	38
766	011530	012737	011542	013142		MOV	038,BACK
767	011536	000137	012224			JMP	EOP
768	011542	000741		381		HR	00W0
769							
770							
771							

```

772 ;.....
773 ; ROUTINE USED IF INTERNAL LOOP BACK" WAS SELECTED.
774 ; NOTE THAT WHEN IN THIS MODE, HALF DUPLEX IS THE
775 ; ONLY MODE AVAILABLE.
776 ; "INTERNAL LOOP BACK" MEANS THAT THE RECEIVER IS "TURNED ON"
777 ; AND A COMPLETE MESSAGE IS RECEIVED. IF DATA IS TO BE CHECKED
778 ; IT IS; IF "END PASS" IS DESIRED; IT IS GIVEN.
779 ; THEN THE TRANSMITTER IS ENABLD. AFTER THE WHOLE MESSAGE
780 ; IS TRANSMITTED; THE CYCLE IS REPEATED AS ABOVE.
781 ;.....
782
783 011544 104316      8ILB:  KRDIN
784 011546 004737 013746      JSR   PC,STARTR
785 011552 005037 011032      CLR   TIME
786 011556 032700 040000      18:  BIT   0RFLG,STAT
787 011562 001013      BNE   28
788 011564 023727 011032 000100      CMP   TIME,0100
789 011572 103771      RLO   18
790 011574 011402      MOV   0RCSR,R2
791 011576 016403 000000      MOV   XCSP(R4),R3
792 011602 104001      HLT   1
793 011604 005037 011032      CLM   TIME
794 011610 000762      BR    18
795 011612 032777 000200 177224 28:  BIT   0NODAT,0SWR
796 011620 001002      RNE   38
797 011622 004737 012364      JSR   PC,TESTD
798 011626 042700 040000      38:  BIC   0RFLG,STAT
799 011632 032777 000020 177204      BIT   0LOOP,0SWP
800 011640 001405      BFG   48
801 011642 012737 011654 013142      MOV   048,0ACK
802 011650 000137 012224      JMP   EOP
803 011654 032777 000400 177162 48:  BIT   0400, 0SWP ;USE EXTERNAL DATA?
804 011662 001416      BEO   78 ;BR IF NO
805 011664 013702 011020      MOV   IRDA, R2 ;SET POINTER
806 011670 013703 011022      MOV   IXDA, R3 ;SET POINTER
807 011674 010337 011070      MOV   R3, XDA ;SETUP XMIT DATA ADDR
808 011700 112223      MOVVB (R2)+, (R3)+ ;MOVE INPUT TO OUTPUT
809 011702 001376      BNE   -2 ;LOOP IF NOT ZERO CHAR
810 011704 112763 000177      MOVVB 0177, -(R3) ;INSERT A FILL CHAR
811 011710 005203      INC   R3 ;BUMP ADDRESS
812 011712 112723 000177      MOVVB 0177, (R3)+ ;INSERT ANOTHER FILL
813 011716 105023      CLPB (R3)+ ;INSERT ZERO CHAR
814 011720 005037 011032      78:  CIR   TIME
815 011724 004737 013146      JSR   PC,STARTX
816 011730 032700 100000      58:  BIT   0RFLG,STAT
817 011734 001013      RNE   68
818 011736 023727 011032 000100      CMP   TIME,0100
819 011744 103771      RLO   58
820 011746 011402      MOV   0RCSR,R2
821 011750 016403 000000      MOV   XCSP(R4),R3
822 011754 104001      HLT   1
823 011756 005037 011032      CLM   TIME
824 011762 000762      BR    58
825 011764 042700 100000      68:  BIC   0RFLG,STAT
826 011770 000137 011544      JMP   8ILB

```

```

027 ;
028 ;
029 ;
030 ;
031 ;
032 ;
033 ;
034 ;
035 ;
036 ;
037 ;
038 ;
039 ;
040 ;
041 ;
042 ;
043 ;
044 ;
045 ;
046 ;
047 ;
048 ;
049 ;
050 ;
051 ;
052 ;
053 ;
054 ;
055 ;
056 ;
057 ;
058 ;
059 ;
060 ;
061 ;
062 ;
063 ;
064 ;
065 ;
066 ;
067 ;
068 ;
069 ;
070 ;
071 ;
072 ;
073 ;
074 ;
075 ;
076 ;
077 ;
078 ;
079 ;
080 ;
081 ;
082 ;

```

```

;.....
; 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.
;.....

```

```

040 J11774 104416 040000 011014 8XLR: KBDIN
041 011776 032737 040000 011014 BIT 0HALF,DUPLEX,PARAM2
042 012004 001002 BNE 10
043 012006 004737 013746 JSR PC,STARTR
044 012012 004737 013146 10: JSR PC,STARTX
045 012016 005037 011032 CLR TIME
046 012022 032737 100000 20: BIT 0XFLG,STAT
047 012026 001016 BNE 30
048 012030 032700 040000 70: BIT 0RFLG,STAT
049 012034 001024 RNE 40
050 012036 023727 011032 000100 CMP TIME,0100
051 012044 103766 BLO 20
052 012046 011402 MOV 0RCSR,R2
053 012050 016433 000000 MOV XCSR(R4),R3
054 012054 104031 HLT 1
055 012056 005037 011032 CLR TIME
056 012062 000757 BP 20
057 012064 032737 040000 011014 30: BIT 0HALF,DUPLEX,PARAM2
058 012072 001756 BEQ 70
059 012074 042700 100000 BIC 0XFLG,STAT
060 012100 004737 013746 JSR PC,STARTR
061 012104 000746 BP 20
062 012106 032737 040000 011014 40: BIT 0HALF,DUPLEX,PARAM2
063 012114 001020 RNE 00
064 012116 032700 100000 BIT 0XFLG,STAT
065 012122 001013 BNE 60
066 012124 023727 011032 000100 CMP TIME,0100
067 012132 103765 BLO 40
068 012134 011402 MOV 0RCSR,R2
069 012136 016433 000000 MOV XCSR(R4),R3
070 012142 104031 HLT 1
071 012144 005037 011032 CLR TIME
072 012150 000756 BP 40
073 012152 042700 100000 60: BIC 0XFLG,STAT
074 012156 042700 040000 00: BIC 0RFLG,STAT
075 012162 005037 011032 CLR TIME
076 012166 032777 000200 176650 BIT 0NODAT,0S=R
077 012174 001002 BNE 50
078 012176 004737 012364 JSR PC,TESTD
079 012202 032777 000020 176634 50: BIT 0LOOP,0S=R
080 012210 001071 BFO 8XLB
081 012212 012737 011774 013142 MOV 08XLB,BACK
082 012220 000137 012224 JMP EOP

```

```
003 ;.....
004 ; ROUTINE TO RETURN
005 ; TO MONITOR FOR
006 ; END PASS.
007 ;.....
008
009 EOP:
010 012224 104414 000340 STFS,PPTY7 ;SET PS PRIORITY TO 7
011 012230 016437 000000 012362 MOV XCSR(R4),QTPIE ;SAVE TX CSR
012 012236 042737 157777 012362 BIC 0<C<TIE>,QTPIE ;CLFAR ALL BUT TX IE.
013 012244 042764 020000 000000 BIC 0TIE,XCSR(R4) ;CLEAR TX IE (EVEN IF IT WASN'T SET)
014 012252 012766 012312 000000 MOV 0ENTER,2(SP) ;SET FOR RETURN IF SW 14=1
015 012260 010037 013124 MOV R0,SAVR0 ;SAVE REGISTER 0
016 012264 010137 013126 MOV R1,SAVR1 ;SAVE REGISTER 1
017 012270 010237 013130 MOV R2,SAVR2 ;SAVE REGISTER 2
018 012274 010337 013132 MOV R3,SAVR3 ;SAVE REGISTER 3
019 012300 010437 013134 MOV R4,SAVR4 ;SAVE REGISTER 4
020 012304 010537 013136 MOV R5,SAVR5 ;SAVE REGISTER 5
021 012310 000207 RTS PC ;RETURN TO CONTROL PROGRAM
022
023 ENTER:
024 012312 013700 013124 MOV SAVR0,R0 ;RESTORE R0
025 012316 013701 013126 MOV SAVR1,R1 ;RESTORE R1
026 012322 013702 013130 MOV SAVR2,R2 ;RESTORE R2
027 012326 013703 013132 MOV SAVR3,R3 ;RESTORE R3
028 012332 013704 013134 MOV SAVR4,R4 ;RESTORE R4
029 012336 013705 013136 MOV SAVR5,R5 ;RESTORE R5
030 012342 012737 177777 013140 MOV 0-1,DELAY
031 012350 053764 012362 000000 BIS QTPIE,XCSR(R4) ;IF ORGINALLY SET; SET TX IF
032 012356 000177 000500 JMP 0BACK
033 012362 000000 QTPIE: 000000
034
035
036
037 ;.....
038 ; SUBROUTINE TO CHECK
039 ; RECEIVER DATA.
040 ;.....
041 TESTD: MOV ERDBR, -(SP) ;WAS THERE A RECEIVE ERROR?
042 012370 001413 000000 176444 BEQ TSTDAT ;BR IF NO
043 012372 032777 020000 0176444 BIT 0BIT13,0SWR ;INHIBIT PRINTOUTS?
044 012400 001007 000000 RNE TSTDAT ;BR IF YES
045 012402 104400 012564 TYPE ,4MSG0 ;<15><12>THERE WAS A RECEIVE ERROR, RBUF=
046 012406 004077 176416 JSR R0,0R2016 ;PRINT CONTENTS OF RBUF
047 012412 005746 TST -(SP)
048 012414 104400 012645 TYFF ,4MSG1 ;<15><12>
049 012420 013701 011022 TSTDAT: MOV IXDA, R1 ;SETUP XMIT DATA ADDR
050 012424 013702 011020 MOV IRDA, R2 ;SETUP RCV DATA ADDR
051 012430 122122 SCAN4: CMPR (R1)+, (R2)+ ;DATA OK ?
052 012432 001776 BEQ SCAN4 ;BR IF OK
053 012434 123741 011040 CMPR TX,TERM,-(R1) ;IS IT END OF DATA
054 012440 001447 BEQ TESTDX ;BR IF YES
055 012442 122742 000000 CMPR 0002,-(R2)
056 012446 001005 RNE 28
057 012450 010237 012456 MOV R2,18
058 012454 104400 TYPE
```

```

939 012456 000000      18:  ,WORD  0
940 012460 000437      BR      TESTDX
941 012462      20:
942 012462 105712      TSTR   (R2)      ;
943 012464 001435      BEQ    TESTDX    ;BR IF YES
944 012466 122721 000177  CMPB   0177, (P1)+ ;IS IT FILL CHAR?
945 012472 001756      BEQ    SCAN4     ;BR IF YES
946 012474 005301      DEC    R1        ;BACKUP
947 012476 122722 000177  CMPB   0177, (R2)+ ;IS IT FILL?
948 012502 001752      BEQ    SCAN4     ;BR IF YES
949 012504 000240      SCANS1: NOP        ;DATA ERROR
950 012506 032777 020000 176330  BIT    0BIT13,0SWR ;INHIBIT PRINTOUTS
951 012514 001016      BNE    DERR      ;BR IF YES
952 012516 104400 012650  TYPE   ,MSG2     ;<15><12>RECEIVED DATA = <15><12>
953 012522 013737 011020 012532  MOV    IRDA, RDAX ;SETUP DATA ADRESS
954 012530 104400      TYPE   ;PRINT RECEIVED DATA
955 012532 000000      RDAX1: 0         ;RECEIVED DATA ADDR.
956 012534 104400 012675  TYPE   ,MSG3     ;<15><12>DATA SHOULD RE<15><12>
957 012540 013737 011022 012550  MOV    IXDA, .+10 ;SETUP ADDR.
958 012546 104400      TYPE   ;PRINT GOOD DATA
959 012550 011022      IXDA
960 012552 111103      DERR1: MOVR   (R1),R3 ;SETUP XMIT DATA
961 012554 110202      MOVR   -(R2),R2    ;SETUP RCV DATA
962 012556 104007      HLT+7  ;DATA ERROR HALT
963 012560 005726      TESTDX: TST    (SP)+ ;POP STACK
964 012562 000207      RTS     PC        ;RETURN FROM SUB/ROUT
965
966 012564 005015 044124 051105  MSG01: ,ASCIZ <15><12>/THERE WAS A RECEIVER ERROR. REGISTER (SEL 2) =/
(1) 012645 015 000012  MSG11: ,ASCIZ <15><12>
(1) 012650 005015 042522 042503  MSG21: ,ASCIZ <15><12>/RECEIVED DATA = /<15><12>
(1) 012675 015 042012 052101  MSG31: ,ASCIZ <15><12>/DATA SHOULD RE/<15><12>
(1) 012720 005015 046120 040505  MSG41: ,ASCII <15><12>/PLEASE MAKE CONNECTION (DIAL NUMBER),/
(1) 012767 015 053412 042510  ,ASCIZ <15><12>/WHEN CONNECTION COMPLETE, HIT CONTINUE SWITCH,/<15><12>
(1) 013052 005015 046120 040505  MSG51: ,ASCIZ <15><12>/PLEASE MAKE CONNECTION (DIAL NUMBER),/<15><12>
(1)
(1) 013124 000000      ,EVEN
967 013126 000000      SAV0: 0
968 013130 000000      SAV1: 0
969 013132 000000      SAV2: 0
970 013134 000000      SAV3: 0
971 013136 000000      SAV4: 0
972 013140 000000      SAV5: 0
973 013142 000000      DELAY: 0
974 013144 000000      BACK: 0
975 013146 000000      STOP: 0
    
```

```

976 ;*****
977 ; TRANSMITTER INITIALIZATION SUBROUTINE
978 ;*****
979
980 013146 005737 011024 STARTX: TST SETTLE
981 013152 001404 BNE 68
982 013154 005737 013140 TST DELAY
983 013160 001015 BNE 58
984 013162 000434 BR 18
985 013164 005037 013730 68: CLR TEMP1 ;PREPARE FOR DELAY
986 013170 012737 000007 013732 MOV #7,TEMP2
987 013176 002737 000001 013730 ADD #1,TEMP1 ;INC DELAY
988 013204 001374 BNE .-6
989 013206 005337 013732 DEC TEMP2
990 013212 001371 BNE .-14
991 013214 005037 013730 58: CLR TEMP1 ;PREPARE FOR DELAY
992 013220 012737 000007 013732 MOV #7,TEMP2
993 013226 002737 000001 013730 ADD #1,TEMP1 ;INC DELAY
994 013234 001374 BNE .-6
995 013236 005337 013732 DEC TEMP2
996 013242 001371 BNE .-14
997 013244 005037 013140 CLR DELAY
998 013250 005037 011024 CLR SETTLE
999 013254 002737 000000 011014 18: BIT #HALF,DUPLEX,PARAM2 ;HALF DUPLEX?
1000 013262 001440 BEQ 48 ;BR IF NO
1001 013264 013746 000004 MOV #4,-(SP) ;SAVE LOC 4
1002 013270 013746 000006 MOV #6,-(SP) ;SAVE LOC 6
1003 013274 012737 013352 000004 MOV #38,#4 ;SET UP TRAP CATCHER
1004 013302 005037 000006 CLR #6 ;CLEAR VECT+2
1005 013306 005737 013144 TST STOP ;FIRST TIME HFREQ?
1006 013312 001407 BEQ 88 ;BR IF YES
1007 013314 012737 177777 013144 MOV #-1,STOP
1008 013322 002777 000000 000406 BIT #100,#DMBB ;CARRIER UP?
1009 013330 001374 BNE .-6 ;BR IF YES
1010 013332 005277 000004 000376 88: BIS #BIT2,#DMBB ;SET PQTS IN DMBB
1011 013340 002777 000004 000370 28: BIT #BITS,#DMBB ;SPIN ON CTS
1012 013346 001774 BEQ 28
1013 013350 024646 CMP -(SP),-(SP) ;ADJUST STACK
1014 013352 022026 38: CMP (SP)+,(SP)+ ;POP STACK
1015 013354 012037 000006 MOV (SP)+,#6 ;RESTORE LOC 6
1016 013360 012037 000004 MOV (SP)+,#4 ;RESTORE LOC 4
1017 013364 013737 011022 011070 48: MOV IXDA,XDA ;SET UP XMIT DATA ADD
1018 013372 002700 100000 BIC #XFLG,STAT ;CLEAR XFLG
1019 013376 013764 011070 000006 MOV XDA,CAR(R4) ;LOAD CURRENT ADDRESS REG
1020 013404 002737 000000 011014 BIT #HALF,DUPLEX,PARAM2 ;HALF DUPLEX?
1021 013412 001022 BNE 78 ;BR IF YES
1022 013414 002730 000004 BIT #XLB,MODE ;XLB MODE?
1023 013420 001417 BEQ 78 ;BR IF NO
1024 013422 012737 177777 013742 MOV #-1,TRNFLG ;SET SOFTWARE FLAG
1025 013430 012764 177777 000010 MOV #-1,RCR(R4)
1026 013436 005274 020000 BIS #TIE,#RCSP
1027 013442 013764 013730 000012 MOV BARTMP,BAR(R4)
1028 013450 000001 WAIT
1029 013452 005737 013740 TST SNCFLG ;HAS RECEIVER GOT FIRST CHART?
1030 013456 001375 BNE .-4 ;NO WAIT FOR IT
1031 013460 013764 011070 000006 78: MOV XDA,CAR(R4) ;LOAD CURRENT ADDRESS REG
    
```

H2

```

1032 013466 012764 177777 000010      MOV      0-1,BCR(P4)      ;LOAD BYTE COUNT REG
1033 013474 052714 020000      BIS      STIE,0RC5R      ;SET INTERRUPT ENABLE
1034 013500 013764 013734 000012      MOV      BARTMP,BAR(R4)  ;LOAD BAR REG
1035 013506 000207      RTS      PC
1036
1037 013510 042714 100000      XISP1:  BIC      STI,0RC5R      ;CLEAR XMIT DONE
1038 013514 032714 002000      BIT      0NEM,0RC5R      ;NON-EXISTENT MEM ERROR?
1039 013520 001407      BEQ      10      ;BR IF NO
1040 013522 011402      MOV      0RC5R,R2      ;SAVE CSR FOR TYPE OUT
1041 013524 005003      CLR      R3
1042 013526 104010      HLT      10      ;ERROR HLT
1043 013530 104404 014337      TYPE    ,NONEX      ;TYPE ERROR MESS
1044 013534 000000      HALT
1045 013536 000776      BP
1046 013540 127737 175324 011040 10:      CMPR    0XDA,TX,TERM      ;IS CHAR TERMINATION CHAR?
1047 013546 001033      RNE
1048 013550 052700 100000      BIS      0XFLG,STAT      ;SET XMIT DONE FLAG
1049 013554 042714 020000      BIC      STIE,0RC5R      ;CLEAR INTERRUPT ENABLE
1050 013560 032737 040000 011014      BIT      0HALF,DUPLEX,PARAM2 ;HALF DUPLEX?
1051 013566 001422      BEQ      30      ;BR IF NO
1052 013570 013746 000004      MOV      004,-(SP)      ;SAVE LOC 4
1053 013574 013746 000006      MOV      006,-(SP)      ;SAVE LOC 6
1054 013600 012737 013622 000004      MOV      020,004      ;SET UP TPAP CATCHER
1055 013606 005037 000006      CLR      006      ;CLEAR VECT+2
1056 013612 042777 000004 000116      NIC      0BIT2,0DMRB      ;CLEAR RTS
1057 013620 024646      CMP      -(SP),-(SP)      ;ADJUST STACK
1058 013622 022626 20:      CMP      (SP)+,(SP)+      ;POP STACK
1059 013624 012637 000006      MOV      (SP)+,006      ;RESTORE LOC 6
1060 013630 012637 000004      MOV      (SP)+,004      ;RESTORE LOC 4
1061 013634 000430 30:      BP
1062 013636 032777 000100 175200 XISP1:  BIT      010R,0SWP      ;MONITOR XMIT DATA?
1063 013644 001406      REU      NOXMON      ;BR IF NO
1064 013646 105777 175224      TSTR    0TPS      ;TTY READY?
1065 013652 100003      BPL      NOXMON      ;BR IF NO
1066 013654 117777 175210 175216      MOVB    0XDA,0TPB      ;TYPE CHAR
1067 013662 005237 011070 NOXMON: INC      XDA      ;INC TXBUF POINTER
1068 013666 013764 011070 000006      MOV      XDA,CAR(R4)      ;LOAD CURRENT ADDRESS REG
1069 013674 005737 013742      TST     TRNFLG      ;IS THIS FIRST TIME?
1070 013700 001006      BNE     XISP2      ;BR IF YFS
1071 013702 012764 177777 000010      MOV      0-1,BCR(P4)      ;LOAD BYTE COUNT REG
1072 013710 013764 013734 000017      MOV      BARTMP,BAR(R4)  ;SET BAR BIT
1073 013716 005037 011032 XISP2:  CLR      TIME
1074 013722 005037 013742      CLR     TRNFLG
1075 013726 000002      RTI
1076 013730 000000      TEMP1:0
1077 013732 000000      TEMP2:0
1078 013734 000000      BARTMP:0
1079 013736 170502      DMDB:  170502      ;LINE STATUS REG IN DMDB
1080 013740 000000      SNCFLG:0
1081 013742 000000      TRNFLG:0
1082 013744      177      FILL:  ,BYTE 177
1083      013746      .EVEN
    
```

```

1004 ;.....
1005 ; PFCEIVER INITIALIZATION SUBROUTINE
1006 ;.....
1007
1008 013746 032737 040000 011014 STARTR: BIT 0HALF,DUPLEX,PARAM2 ;HALF DUPLEX?
1009 013754 001010 BNE 28 ;BR IF YES
1010 013756 032700 000004 BIT 0XLB,MODE ;XLB MODF?
1011 013762 001405 BEQ 28 ;BR IF NO
1012 013764 005037 013730 CLR TEMP1 ;START DELAY
1013 013770 005237 013730 18: INC TEMP1
1014 013774 001375 BNE 18
1015 013776 042700 040000 28: BIC 0RFLG,STAT ;CLEAR RFLG
1016 014002 013737 011020 011066 MOV IRDA,RDA ;SET UP RECEIVER DATA ADD
1017 014010 012737 001000 011064 MOV 01000,RCC ;SET UP BUFFER LIMIT
1018 014016 012737 177777 013740 MOV 0-1,SNCFLG ;SET SOFTWARE FLAG
1019 014024 005037 011054 CLR ERCSR ;CLEAR ERROR RECORDS
1020 014030 005037 011056 CLR ERDRR
1021 014034 052714 004000 BIS 0BIT11,0RCSP ;MASTER CLEAR
1022 014040 053714 011012 RIS PARAM1,0RCSP ;SET LINE NUMBER
1023 014044 053764 011014 000004 BIS PARAM2,LPR(R4) ;LINE PARAMETERS
1024 014052 052714 010100 RIS 0PIE+SIE,0RCSP ;SET INTERRUPT ENABLES
1025 014056 000207 RTS PC
1026
1027 014060 032714 040000 RISR: RIT 0SI,0RCSP ;SILO OVERFLOW?
1028 014064 001407 BEO 18 ;BR IF NO
1029 014066 011402 MOV 0RCSP,R2 ;SAVE CSR FOR TYPEOUT
1030 014070 005003 CLR R3
1031 014072 104010 HLT 10 ;ERROR HLT
1032 014074 104000 014310 TYPF ,SILO ;TYPE ERROR MESS
1033 014100 000000 HALT
1034 014102 000776 BR 0-2 ;RR HALT
1035 014104 016401 000002 18: MOV 0RCCR(R4),R1 ;PUT CHAR IN R1
1036 014110 042701 000200 BIC 0200,R1 ;STPIP A BIT
1037 014114 005701 TST R1 ;VALID DATA?
1038 014116 100403 RMI 48 ;BR IF YES
1039 014120 011402 MOV 0PCSR,R2 ;SAVE CSR FOR TYPFOUT
1040 014122 005003 CLR R3
1041 014124 104010 HLT 10 ;ERROR HLT
1042 014126 032701 070000 48: HIT 0DO+FE+PE,R1 ;OVERPUN,FRAMING OR PARITY ERROR?
1043 014132 001404 BEQ 38 ;RR IF NO
1044 014134 011437 011054 MOV 0RCSP,ERCSR ;SAVE CSR
1045 014140 010137 011056 MOV R1,ERDRR ;SAVE CHAR
1046 014144 110177 174716 38: MOVR R1,0RDA ;STORE CHAR IN BUFFER
1047 014150 032777 000000 174666 RIT 0RIT5,0SWP ;MONITOR RECEIVE DATA?
1048 014156 001405 BFO 0NORMON ;RR IF NO
1049 014160 105777 174712 TSTR 0TPS ;TTY READY?
1050 014164 100402 RPL 0NORMON ;RR IF NO
1051 014166 110177 174706 MOVR R1,0TPB ;TYPE CHAR
1052 014172 005237 011066 NORMON: INC RDA ;INC RECEIVER BUFFER POINTER
1053 014176 105077 174664 CLRR 0RDA ;CLEAR NEXT LOCATION
1054 014202 005337 011064 DEC RCC ;DEC CHAR COUNT
1055 014206 001405 BNE 18 ;BR IF BUFFER NOT FULL
1056 014210 000005 RESET
1057 014212 104000 HLT 0 ;STOP THE SHOW,BUFFER OVERFLOWED!
1058 014214 104006 HLT+6 ;RECEIVER BUFFER FULL
1059 014216 000000 HALT
    
```



```
1140 014220 000776          BP          .-2          ;BP HALT
1141 014222 123701 011041    101          CMPA          RX,TERM,R1      ;IS CHAR PCV TERMINATION CHAR
1142 014226 001004          BNE          RISR1      ;BR IF NOT
1143 014230 042714 010100    BIC          BR1E+SIE,0RCSR ;CLEAR INTERRUPT ENABLES
1144 014234 052700 040000    HIS          0PFLG,STAT    ;SET PCV DONE FLAG
1145 014240 005037 011032    RISR1:      CLP          TIME
1146 014244 005037 013740    CLR          SNCFLG        ;CLEAR FLAG
1147 014250 000002          RTI
1148 014252 005015 051105 047522 MFULL:      .ASCIZ<15><12>/ERROR! RECEIVER BUFFER FULL/
      014310 005015 051105 047522 SILO:       .ASCIZ<15><12>/ERROR! SILO OVERFLOW/
      014337      015 047012 047117 NONEX:     .ASCIZ<15><12>/NON EXISTENT MEMORY ERROR/
      014373      015 050012 042514 LINES:    .ASCIZ<15><12>/PLFASE SELECT ONLY ONE LINE AT A TIME(PARAM)/
      014454          .EVEN
      000001          .END
```

K2

BA	#11004	DSFLG	# 020000	NODAT	# 000200	RIV	011006	TEMP1	013730
BACK	#13142	DSSTAT	011060	NONEX	014337	RWAIT	# 104410	TEMP2	013732
BAR	# 000012	ENTER	012312	NORMON	014172	RX,TER	011041	TESTD	012360
BARTMP	013734	LOP	012224	NOXMON	013662	R6	#000006	TESTDX	012560
BCR	# 000010	ERCSP	011054	NPCR	# 000002	R7	#000007	TI	# 100000
BTA	# 000001	ERDBR	011056	OWI	# 000002	SAVR0	013124	TIE	# 020000
BIT1	# 000002	FE	# 020000	OWO	# 000001	SAVR1	013126	TIME	011032
BIT10	# 002000	FILL	013744	PARAM1	011012	SAVR2	013130	TKB	011074
BIT11	# 004000	FLAG	011042	PARAM2	011014	SAVR3	013132	TKS	011072
BIT12	# 010000	FULL,D	# 000001	PARAM3	011016	SAVR4	013134	TPB	011100
BIT13	# 020000	GO	011250	PE	# 010000	SAVR5	013136	TPS	011076
BIT14	# 040000	HALF,D	# 040000	PRIOP	011010	SCAN4	012430	TRNFLG	013742
BIT15	# 100000	ILB	# 000010	PRTY0	# 000000	SCAN5	012504	TSTDAT	012420
BIT2	# 000004	IPDA	011020	PRTY1	# 000040	SETTLE	011024	TX,TER	011040
BIT3	# 000014	IXDA	011022	PRTY2	# 000100	SI	# 040000	TYPE	# 104400
BIT4	# 000020	KRDIN	# 104416	PRTY3	# 000140	SIE	# 010000	XCC	011062
BIT5	# 000040	LINES	014373	PRTY4	# 000200	SILO	014310	XCSR	# 000000
BIT6	# 000100	LOOP	# 000020	PRTY5	# 000240	SNCFLG	013740	XDA	011070
BIT7	# 000200	LPR	# 000004	PRTY6	# 000300	SRCSP	011052	XFLG	# 100000
BIT8	# 000400	MC	# 004000	PRTY7	# 000340	SSP	# 000016	XISR	013510
BIT9	# 001000	MFULL	014252	OTPIE	012362	START	011102	XISR1	013636
B2016	011030	MODE	#000000	RCC	011064	STARTP	013746	XISR2	013716
CAR	# 000006	MSG0	012564	RCSP	#000004	STARTX	013146	XLB	# 000004
DELAY	#13140	MSG1	012645	RDA	011066	STAT	#000000	XWAIT	# 104412
DERR	012552	MSG2	012650	RDAX	012532	STOP	013144	YILB	011544
DH11	011000	MSG3	012675	RFLG	# 040000	STPS	# 104414	YOWI	011340
DISPLA	011046	MSG4	012720	RIE	# 000100	SWR	011044	YOWO	011440
DMBB	013736	MSG5	013052	RISR	014060	SW12	# 010000	YXLB	011774
DO	# 040000	NEM	# 002000	RISP1	014240	SXCSR	011050	.	# 014454

. ABS. #14454 #00

ERRORS DETECTED: 0  
 DEFAULT GLOBALS GENERATED: 0

DHLB,DHLB/SOL\_ITEP1,MAC,DZDHLB,P11  
 RUN-TIME: 10 13 .3 SECONDS  
 RUN-TIME RATIO: 66/25=2.5  
 CORE USED: 16K (31 PAGES)