```
      1                                .SBTTL DOCUMENTATION
        .REM_
```

IDENTIFICATION

PRODUCT CODE:	AC T802A-MC
PRODUCT NAME:	CXDMUAO DMU-11 DEC/X11 MODULE
PRODUCT DATE:	15 DECEMBER 1983
MAINTAINER:	ENE - DIAGNOSTICS, READING
AUTHOR:	A.HYATT
VERSION:	A.0

REVISION HISTORY

REV.	DATE	AUTHOR	COMMENTS
A.0	15-DEC-83	A.HYATT	FIRST RELEASE

1. ABSTRACT:

THIS DOCUMENT DESCRIBES THE DHU11 DEC/X11 EXERCISER MODULE. THE PROGRAM EXERCISES UP TO FOUR (CONSECUTIVELY ADDRESSED) DHU11 ASYNCHRONOUS INTERFACES. MAINTENANCE MODE IS USED TO TRANSMIT AND RECEIVE A BINARY COUNT PATTERN TRANSMITTED AND RECEIVED IN 108 CHARACTER BURSTS. ALL DEVICES SELECTED FOR TESTING ARE ACTIVATED AND RUN CONCURRENTLY. ALL SIXTEEN LINES ARE RUN ON EACH SELECTED DEVICE.

THIS PROGRAM WILL RUN ONLY ON A UNIBUS PROCESSOR. THIS PROGRAM WILL NOT SUPPORT ANY DEVICE OTHER THAN THE DHU11.

THE DHU11 IS AN ASYNCHRONOUS MULTIPLEXER THAT PROVIDES AN INTERFACE BETWEEN SIXTEEN ASYNCHRONOUS SERIAL DATA COMMUNICATIONS CHANNELS AND ANY UNIBUS PROCESSOR.

THE DHU11 PROVIDES RS232-A AND RS423-A INTERFACES WITH ENOUGH MODEM CONTROL TO PERMIT FULL DUPLEX POINT TO POINT OPERATION OR DIAL UP (AUTO ANSWER) OPERATION OVER THE PUBLIC SWITCHED TELEPHONE NETWORK.

FOR EACH LINE THE CHARACTERISTICS ARE SET UNDER PROGRAM CONTROL, THE ONLY INFORMATION THAT IS SET VIA SWITCHES ON THE CARD BEING THE UNIBUS ADDRESS AND THE VECTOR ADDRESS.

2. REQUIREMENTS:

HARDWARE:
- O A PDP SYSTEM WITH UNIBUS
- O 32K BYTES OF MEMORY (TOTAL) ON THE SYSTEM.
- O 1-4 DHU11 INTERFACES. NO WRAPAROUND OR MAINTENANCE CABLE IS NEEDED IF RUNNING IN LOCAL LOOPBACK MODE.

STORAGE: THIS MODULE (DHU11) REQUIRES:

1. DECIMAL WORDS: 1122
2. OCTAL WORDS: 2142
3. OCTAL BYTES: 4304

SOFTWARE: LATEST REVISION OF THE DEC/X11 MONITOR & LINKER.

3. PASS DEFINITION:

ONE PASS OF THE DHU11 MODULE CONSISTS OF A NUMBER (DEFAULT OF 8) OF INTERATIONS.

AN ITERATION CONSISTS OF FOUR BASIC TESTS ON EACH DHU11, INCLUDING THE TRANSMITTING AND RECIEVING OF 108 CHARACTERS ON EACH LINE.

4. EXECUTION TIME:

THE EXECUTION TIME (TIME TO MAKE ONE PASS) DEPENDS ON THE NUMBER OF DHU11'S, THE NUMBER OF LINES SELECTED, THE BAUD RATE AND THE OTHER MODULES CONFIGURED IN. THE AVERAGE TIME FOR A MODULE RUNNING STAND ALONE ON A PDP 11/24 WITH BAUD RATES ETC. SET TO DEFAULT IS ONE MINUTE.

5. CONFIGURATION PARAMETERS:

MEANING OF PARAMETERS:

DVA: ADDRESS OF FIRST DHU11 CSR REG.
VCT: VECTOR ADDRESS OF FIRST DHU11
BR1: PRIORITY OF DHU11'S
BR2: SECOUND PRIORITY (NOT USED)
DVC: NO. OF DHU11'S IF GREATER THAN 1
SR1: FIRST SOFTWARE REGISTER

DEFAULT PARAMETERS:

DVA: 1, VCT: 1, BR1: 5, BR2: 5, DVC: 1, SR1: 0

REQUIRED PARAMETERS:

AT CONFIGURATION TIME THE USER MUST SPECIFY:

DVA: ADDRESS OF FIRST DHU11 CSR REG.
VCT: VECTOR ADDRESS OF FIRST DHU11
DVC: NO. OF DHU11'S IF GREATER THAN 1

MODULE LOCATION DVID1 (HOLDS DVC) MAY BE MODIFIED (MOD DHUA 14) TO EXERCISE ANY COMBINATION OF DHU11'S. (SEE ITEM #8, "OPERATOR OPTIONS").

6 DEVICE OPTION SETUP:

NONE REQUIRED

7. MODULE OPERATION:

THE DHU11 DEC/X11 MODULE IS ASSEMBLED AND LINKED ACCORDING TO STANDARDS OUTLINED IN THE DEC/X11 MODULE PROGRAMMER'S GUIDE.

THE MAIN ALGORITHM:

THE TESTING IS PERFORMED WITH INTERNAL LOOPBACK SELECTED, VIA MAINTENANCE MODE.
THE CHARACTERS 224 - 377 (OCTAL) ARE TRANSMITTED ON 16 LINES SIMULTANEOUSLY VIA DIRECT MEMORY ACCESS. THE RECEIVER, NOT OPERATING ON A DMA BASIS, WILL RECEIVE CHARACTERS ON A PER/CHARACTER BASIS INTO THE RECEIVE FIFO. IF SUCCESSFUL, 108 (DECIMAL) CHARACTERS PER LINE SHOULD BE RECEIVED. EACH RECEIVED LINE WILL BE CHECKED FOR PROPERLY RECEIVED DATA BY COMPARING EACH RECEIVED CHARACTER TO ITS EXPECTED VALUE.

START:

- SETUP NUMBER OF DHU11'S TO BE TESTED.

RESTRT:

- SETUP DEVICE ADDRESSES & VECTORS.

DHUT01:

- RESET DHU11 AND TEST THE RESET BIT CLEARS.

DHUT02:

- SEND ONE CHARACTER AROUND THE LOOP AND TEST THAT THE BOARD INTERRUPTS. DONE FOR EACH LINE ON ALL BOARDS.

DHUT03:

- RUN INTERNAL SELF TEST DIAGNOSTICS ON ALL DHU11'S SELECTED AND TEST FOR PASS OR FAIL.

DHUT04:

- GET LINES TO BE TESTED ON THIS DHU11.
- SET-UP WRITE COUNT PATTERN INTO XMIT BUFFER.
- SET-UP LINE PARAMETERS (9600 BAUD, 8 BIT/CHAR., ENABLE RECEIVER).

- START ALL LINES/DMU11 SELECTED.

TXSERV:

- CHECK FOR VALID INTERRUPT AND REPORT ERROR AS REQUIRED.

RXSERV:

- RECEIVES EACH CHARACTER OF EACH LINE UPON GETTING FIFO INTERRUPTS.
- VERIFIES CORRECT DATA REPORT ERROR AS REQUIRED.
- CHECK FOR END OF DATA AND BOARD FINISHED.
- WHEN ALL DONE DECLARE END OF ITERATION.

8. OPERATOR OPTIONS:

NOTE
SR1 , 2 AND 3 APPLY TO ALL DMU11'S SELECTED

1). TESTING MULTIPLE DMU11'S

MODULE LOCATION DVID1 (MOD DMU 14) MAY BE MODIFIED TO EXERCISE ANY COMBINATION OF FOUR DMU11'S:

DEVICE TO TEST	DVID1 =
1	1
2	2
3	4
4	10
1,2	3
1,3	5
1,4	11
2,3	6
2,4	12
3,4	14
1,2,3	7
1,2,4	13
1,3,4	15
2,3,4	16
1,2,3,4	17

2). SELECTABLE BAUD RATE:

THE BAUD RATE MAY BE SELECTED BASED ON THE CODE BELOW. THE SELECTION

PROCEDURE IS VERY SIMPLE.

IF YOU WANT TO USE 9600 BITS/SEC. THEN SIMPLY LEAVE BOTH SR1 & SR2 "0". THIS IS THE DEFAULT VALUE.

IF YOU WISH TO USE OTHER BAUD RATES THEN FOLLOW THIS PROCEDURE:

A. SELECT THE BAUD RATE YOU DESIRE FROM THE BAUD RATE TABLE. WE WILL USE 38.4K FOR EXAMPLE.

   NOTE 1: THE TRANSMIT & RECEIVE BAUD RATES WILL BE THE SAME.

   2: IF USING 38.4K BAUD RATE, A MAXIMUM OF 2 LINES MAY BE USED AT ONE TIME.

   IF USING 19.2K BAUD RATE, A MAXIMUM OF 4 LINES MAY BE USED AT ONE TIME.

   THE VARIABLE "LINESELECT" MUST BE CHANGED TO REFLECT THE MAXIMUM ALLOWABLE # OF LINES THAT MAY BE TESTED. (2 LINES @ 38.4, 4 LINES @ 19.2)

B. PLACE THE BINARY CODE FOR THE PARTICULAR BAUD RATE INTO SR1. FOR OUR EXAMPLE 38.4K HAS A BINARY CODE VALUE OF 17.

C. PLACE A "1" IN SR2. SR2 MUST ALWAYS CONTAIN A ONE WHENEVER A BAUD RATE (OTHER THAN THE DEFAULT) IS SELECTED.

D. IN ORDER FOR THE PROGRAM TO RUN FOR 1 MINUTE, THE VALUE "ICONT" MUST BE MODIFIED FOR EACH SELECTED BAUD RATE. THE APPROPRIATE VALUE FOR "ICONT" MAY BE FOUND IN THE BAUD RATE TABLE. THE 1 MINUTE RUNTIME IS BASED ON RUNNING 1 DHU11. THE RUNTIME WILL BE PROPORTIONATLY LONGER FOR EACH ADDITIONAL DHU11 SELECTED FOR TESTING.

THE BAUD RATE MAY BE SELECTED AT TIME OF CONFIGURATION OR AT RUNTIME BY CHANGING THE CONTENTS OF SR1, SR2, & ICONT TO SELECT A PARTICULAR BAUD RATE.

LOCATION SR1 (MOD DHUA 16).
LOCATION SR2 (MOD DHUA 20).
LOCATION ICONT (MOD DHUA 36).

(NOTE: IF SR2 CONTAINS A "0", THE BAUD RATE SELECTED WILL BE 9600 NO MATTER WHAT IS IN SR1!!)

BAUD RATE TABLE

DATA RATE	BINARY CODE	ICONT *

(BITS/SEC)	(SR1 -)	
50	0	1
75	1	1
110	2	2
134.5	3	2
150	4	3
300	5	4
600	6	5
1200	7	6
1800	10	6
2000	11	7
2400	12	7
4800	13	7
7200	14	7
9600	15	7
19200	16	10
38400	17	10

3). MAY CHOOSE BETWEEN INTERNAL LOOPBACK OR NORMAL OPERATION (NORMAL OPERATION REQUIRES LOOPBACK CONNECTORS TO BE FITTED).

A. INTERNAL LOOPBACK REQUIRES NO CABLES, ONLY THE MODULE IS TESTED.

B. NORMAL OPERATION ALLOWS THE OPERATOR TO TEST NOT ONLY THE MODULE, BUT THE CABLES TOO. A H325 LOOPBACK CONNECTOR IS REQUIRED FOR EACH LINE SET TO NORMAL, IF THIS IS PLACED ON THE DISTRIBUTION PANEL ONLY THE INTERNAL CABLES ARE TESTED. THE H325 CAN ALSO BE PLACED AT THE END OF THE BC05L CABLE (FROM THE DISTRIBUTION PANEL) AND THE BC05L CABLE IS THEN TESTED.

INTERNAL LOOPBACK:
LEAVE SR3 "0".

NORMAL OPERATION:
PLACE INTO SR3 THE BIT MASK OF THE LINE DESIRED FOR TESTING. (SEE BIT MASK VALUES BELOW). (THE SAME LINE NUMBER WILL BE TESTED ON ALL DEVICES SELECTED).

LOCATION SR3: (MOD DHUA 22).

BIT MASK VALUES FOR TESTING A LINE IN NORMAL OPERATION

LINE # DESIRED	SR3
0	1

1	2
2	4
3	10
4	20
5	40
6	100
7	200
8	400
9	1000
10	2000
11	4000
12	10000
13	20000
14	40000
15	100000

4). INDIVIDUAL & MULTIPLE LINE TESTING

MODULE VARIABLE "LINESELECT" MAY BE MODIFIED TO EXERCISE ANY COMBINATION OF SIXTEEN LINES:

LINE NUMBER	LINESELECT
0	1
1	2
2	4
3	10
4	20
5	40
6	100
7	200
8	400
9	1000
10	2000
11	4000
12	10000
13	20000
14	40000
15	100000
1,2	3
1,3	5
1,4	11
ETC.	
ALL 16 LINES	177777

9 ERROR REPORTS:

STANDARD ERROR REPORTS ARE GENERATED EXCEPT AS NOTED UNDER THE DESCRIPTION FOR NON-STANDARD PRINTOUTS. SEE THE DEC/X11 USERS GUIDE FOR EXAMPLES.

10. NON-STANDARD PRINTOUTS:

A. HARD ERROR REPORT

ASTAT (STATC) IS THE CONTENTS OF THE RECEIVER BUFFER.

B. DATA ERROR REPORT

WRADR IS THE LINE NUMBER.
RDADR HAS NO MEANING AND WILL ALWAYS EQUAL '0".

11. MNEMONICS:

THE FOLLOWING INFORMATION SHOULD BE USEFUL IN UNDERSTANDING
NAMES GIVEN TO VARIABLES IN THIS PROGRAM:

```
LINESELECT:   .WORD   177777  ;  INDICATES LINES SELECTED. (16 LINES UNLESS
                              ;  CHANGED BY OPERATOR).
LINESDONE:    .BLKW   4       ;  USED IN RXSERV AS A BITMAP OF LINES FINISHED
LINECOUNT:    .WORD   0       ;  NUMBER OF LINES BEING TESTED.

DHUSDONE:     .WORD   0       ;  INDICATES HOW MANY DHU11'S ARE DONE USED IN DHUT04
DHUSLEFT:     .WORD   0       ;  INDICATES DHU11'S LEFT TO DO USED IN DHUT04
DHU.LIVE:     .WORD   0       ;  BIT MAP OF NUMBER OF DHU11'S TO TEST.
DHUNOW:       .WORD   0       ;  CURRENT DHU11 USED IN DHUT04
DHUCSR:       .WORD   0       ;  DHU11 CSR.

POINT:        .WORD   0       ;  POINTS TO CURRENT DHU11 BEING TESTED.
OFFSET:       .WORD   0       ;  OFFSET TO INDICATE SPECIFIC UNIT.
BUFADR:       .WORD   0       ;  HOLDS CURRENT BUFFER ADDRESS FOR BINARY COUNT PATTERN.
CDHUCSR:      .WORD   0       ;  HOLDS CSR.
LCKOUT:       .WORD   0       ;  ALLOWS ONLY 1 ERROR REPORT AT A TIME
                              ;  I.E. SET TO 1 WHEN REPORTING AN ERROR
TXSPEED:      .WORD   0       ;  TRANSMIT SPEED (BAUD RATE).
RCVSPEED:     .WORD   0       ;  RECEIVE  SPEED (BAUD RATE).

TXBUF:        .BLKB   256.    ;  HOLDS TRANSMIT MESSAGE
;
;
;  THE DHR BUFFERS ARE USED IN DHUT04.
;  THE FIRST 16 BYTES HOLD THE NEXT EXPECTED CHARACTER ON
;  THE CORRESPONDING LINE
;  THE REST OF THE BUFFER IS USED TO HOLD ERROR DATA FOR DHUT04
;
DHR0BUF:      .BLKB   256.    ;  HOLDS THE BINARY COUNT PATTERN FOR DEVICE #1.
DHR1BUF:      .BLKB   256.    ;  HOLDS THE BINARY COUNT PATTERN FOR DEVICE #2.
DHR2BUF:      .BLKB   256.    ;  HOLDS THE BINARY COUNT PATTERN FOR DEVICE #3.
```

```
        DHP3BUF:    .BLKB   256.    ;  HOLDS THE BINARY COUNT PATTERN FOR DEVICE #4.
        DHUVECT:    .WORD   0       ;  HOLDS THE VECTOR ADDRESS OF CURRENT DHU11.

        VA:         .WORD   0       ;  VIRTUAL ADDRESS.
        PA:         .WORD   0       ;  PHYSICAL ADDRESS.
        EA:         .WORD   0       ;  EXTENDED ADDRESS.
        PA22:       .WORD   0       ;  PHYSICAL ADDRESS (22 BIT).
        EA22:       .WORD   0       ;  EXTENDED ADDRESS (22 BIT).

        ADDR0:      .WORD   0       ;  HOLDS CSR ADDRESS OF DHU11 #1.
        ADDR1:      .WORD   0       ;  HOLDS CSR ADDRESS OF DHU11 #2.
        ADDR2:      .WORD   0       ;  HOLDS CSR ADDRESS OF DHU11 #3.
        ADDR3:      .WORD   0       ;  HOLDS CSR ADDRESS OF DHU11 #4.

        FLAG:       .WORD   0       ;  INDICATES IF DEVICE CAN INTERRUPT O.K.
        DHULPR:     .WORD   30      ;  LPR BEFORE SETTING BAUD RATES.
        LPRSELECT:  .WORD   0       ;  LPR AFTER SETTING BAUD RATES.  NEEDED BECAUSE
                                    ;  THE BAUD RATES ARE SET TO DEFAULTS DURING SELF TEST.


538
539
540                                 .SBTTL MODULE DEFINITIONS
```

```
542                                     .LIST SEQ,LOC,BIN
543                                             .TITLE  DHUA DEC/X11 SYSTEM EXERCISER MODULE
                                        ;       DDXCOM  VERSION  6      23 MAY-78
                                                .LIST   BIN
                                        ;*****************************************************************************
                                        BEGIN:
000000                                  MODNAM: .ASCII  /DHUA / ;MODULE NAME.
000000     104     110     125
000003     101     040
000005     000                          XFLAG:  .BYTE   OPEN            ;USED TO KEEP TRACK OF WBUFF USAGE
000006  000001                          ADDR:   1+0             ;1ST DEVICE ADDR.
000010  000001                          VECTOR: 1+0                     ;1ST DEVICE VECTOR.
000012     240                          BR1:    .BYTE   PRTY5+0         ;1ST BR LEVEL.
000013     240                          BR2:    .BYTE   PRTY5+0         ;2ND BR LEVEL.
000014  000020                          DVID1:  17+1                    ;DEVICE INDICATOR 1.
000016  000000                          SR1:    OPEN                    ;SWITCH REGISTER 1
000020  000000                          SR2:    OPEN                    ;SWITCH REGISTER 2
000022  000000                          SR3:    OPEN                    ;SWITCH REGISTER 3
000024  000000                          SR4:    OPEN                    ;SWITCH REGISTER 4
                                        ;*****************************************************************************
000026  140000                          STAT:   140000                  ;STATUS WORD.
000030  004340'                         INIT:   START                   ;MODULE START ADDR.
000032  000222'                         SPOINT: MODSP                   ;MODULE STACK POINTER.
000034  000000                          PASCNT: 0                       ;PASS COUNTER.
000036  000007                          ICONT:  7                       ;# OF ITERATIONS PER PASS=7
000040  000000                          ICOUNT: 0                       ;LOC TO COUNT ITERATIONS
000042  000000                          SOFCNT: 0                       ;LOC TO SAVE TOTAL SOFT ERRORS
000044  000000                          HRDCNT: 0                       ;LOC TO SAVE TOTAL HARD ERRORS
000046  000000                          SOFPAS: 0                       ;LOC TO SAVE SOFT ERRORS PER PASS
000050  000000                          HRDPAS: 0                       ;LOC TO SAVE HARD ERRORS PER PASS
000052  000000                          SYSCNT: 0                       ;# OF SYS ERRORS ACCUMULATED
000054  000000                          RANNUM: 0                       ;HOLDS RANDOM # WHEN RAND MACRO IS CALLED
000056                                  CONFIG:                         ;RESERVED FOR MONITOR USE
000056  000000                          RES1:   0                       ;RESERVED FOR MONITOR USE
000060  000000                          RES2:   0                       ;RESERVED FOR MONITOR USE
000062  000000                          SVR0:   OPEN                    ;LOC TO SAVE R0.
000064  000000                          SVR1:   OPEN                    ;LOC TO SAVE R1.
000066  000000                          SVR2:   OPEN                    ;LOC TO SAVE R2.
000070  000000                          SVR3:   OPEN                    ;LOC TO SAVE R3.
000072  000000                          SVR4:   OPEN                    ;LOC TO SAVE R4.
000074  000000                          SVR5:   OPEN                    ;LOC TO SAVE R5.
000076  000000                          SVR6:   OPEN                    ;LOC TO SAVE R6.
000100  000000                          CSRA:   OPEN                    ;ADDR OF CURRENT CSR.
000102                                  SBADR:                          ;ADDR OF GOOD DATA, OR
000102  000000                          ACSR:   OPEN                    ;CONTENTS OF CSR.
000104                                  WASADR:                         ;ADDR OF BAD DATA, OR
000104  000000                          ASTAT:  OPEN                    ;STATUS REG CONTENTS.
000106                                  ERRTYP:                         ;TYPE OF ERROR
000106  000000                          ASB:    OPEN                    ;EXPECTED DATA.
000110  000000                          AWAS:   OPEN                    ;ACTUAL DATA.
000112  004374'                         RSTRT:  RESTRT                  ;RESTART ADDRESS AFTER END OF PASS
000114  000000                          WDTO:   OPEN                    ;WORDS TO MEMORY PER ITERATION
000116  000000                          WDFR:   OPEN                    ;WORDS FROM MEMORY PER ITERATION
000120  000000                          INTR:   OPEN                    ;# OF INTERRUPTS PER ITERATION
000122                                  IDNUM:                          ;MODULE IDENTIFICATION NUMBER=
           000040                               .REPT   SPSIZ              ;MODULE STACK STARTS HERE.
000222                                  MODSP:
                                        ;*****************************************************************************
```

```
                                ;  BIT DEFINITIONS

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


                                ;  OFFSETS FOR REGISTERS (BASE IS CSR).

000000                                  CSR             =  0
000002                                  RBUF            =  2
000004                                  LPR             =  4
000006                                  TXCHAR          =  6
000010                                  LNCTRL          =  10
000012                                  TX1BUFADR       =  12
000014                                  TX2BUFADR       =  14
000016                                  TXBUFCNTR       =  16



                                ;  REGISTER DEFINITIONS.

                                ;  CSR(R2)              =  CSR.
                                ;  RBUF(R2)             =  RECEIVE BUFFER.
                                ;  TXCHAR(R2)           =  TRANSMIT CHARACTER FIFO.
                                ;  LPR(R2)              =  LINE PARAMETER REGISTER.
                                ;  LNCTRL(R2)           =  LINE CONTROL REGISTER.
                                ;  TX1BUFADR(R2)        =  TRANSMIT BUFFER ADDRESS REGISTER 1.
                                ;  TX2BUFADR(R2)        =  TRANSMIT BUFFER ADDRESS REGISTER 2.
                                ;  TXBUFCNTR(R2)        =  TRANSMIT DMA BUFFER COUNTER.


                                ;  CSR BIT DEFINITIONS.
```

```
                                        TX.ACTION       = BIT15
                                        TXINTR.ENABLE   = BIT14
                                        DIAG.FAIL       = BIT13
                                        TXDMA.ERROR     = BIT12
                                        RCVDATA.AVAIL   = BIT7
                                        RCVINTR.ENABLE  = BIT6
                                        MASTER.RESET    = BIT5


                                ;  RECEIVE BUFFER BIT DEFINITIONS.

                                        DATA.VALID      = BIT15
                                        OVERRUN.ERROR   = BIT14
                                        FRAMING.ERROR   = BIT13
                                        PARITY.ERROR    = BIT12


                                ;  TRANSMIT CHARACTER USED IN DHUT02

                                        TXDATA          = 224


                                ;  LINE CONTROL REGISTER BIT DEFINITIONS.

                                        LOCAL.LOOP      = BIT7
                                        RX.ENABLE       = BIT2
                                        AUTO0           = BIT4
                                        AUTO1           = BIT1


                                ;  TRANSMIT BUFFER ADDRESS 2 BIT DEFINITIONS.

                                        TX.ENABLE       = BIT15
                                        DMA.START       = BIT7


                                ;  THESE ARE THE PROGRAM PARAMETERS:

000222  177777                  LINESELECT:   .WORD   177777  ;  INDICATES LINES SELECTED. (16 LINES UNLESS
                                                               ;  CHANGED BY OPERATOR).
000224                          LINESDONE:    .BLKW   4       ;  USED IN RXSERV AS A BITMAP OF LINES FINISHED
000234  000000                  LINECOUNT:    .WORD   0       ;  NUMBER OF LINES BEING TESTED.

000236  000000                  DMUSDONE:     .WORD   0       ;  INDICATES HOW MANY DHU11'S ARE DONE USED IN DHUT04
000240  000000                  DMUSLEFT:     .WORD   0       ;  INDICATES DHU11'S LEFT TO DO USED IN DHUT04
000242  000000                  DMU.LIVE:     .WORD   0       ;  BIT MAP OF NUMBER OF DHU11'S TO TEST.
000244  000000                  DMUNOW:       .WORD   0       ;  CURRENT DHU11 USED IN DHUT04
```

```
658 000246  000000                  DHUCSR:     .WORD   0       ;  DHU11 CSR.
659
660 000250  000000                  POINT:      .WORD   0       ;  POINTS TO CURRENT DHU11 BEING TESTED.
661 000252  000000                  OFFSET:     .WORD   0       ;  OFFSET TO INDICATE SPECIFIC UNIT.
662 000254  000000                  BUFADR:     .WORD   0       ;  HOLDS CURRENT BUFFER ADDRESS FOR BINARY COUNT PATTEPN.
663 000256  000000                  CDHUCSR:    .WORD   0       ;  HOLDS CSR.
664 000260  000000                  LCKOUT:     .WORD   0       ;  ALLOWS ONLY 1 ERROR REPORT AT A TIME
665                                                             ;  I.E. SET TO 1 WHEN REPORTING AN ERROR
666 000262  000000                  TXSPEED:    .WORD   0       ;  TRANSMIT SPEED (BAUD RATE).
667 000264  000000                  RCVSPEED:   .WORD   0       ;  RECEIVE  SPEED (BAUD RATE).
668
669 000266                          TXBUF:      .BLKB   256.    ;  HOLDS TRANSMIT MESSAGE
670                                 ;
671                                 ;
672                                 ;  THE DHR BUFFERS ARE USED IN DHUT04.
673                                 ;  THE FIRST 16 BYTES HOLD THE NEXT EXPECTED CHARACTER ON
674                                 ;  THE CORRESPONDING LINE
675                                 ;  THE REST OF THE BUFFER IS USED TO HOLD ERROR DATA FOR DHUT04
676                                 ;
677 000666                          DHR0BUF:    .BLKB   256.    ;  HOLDS THE BINARY COUNT PATTERN FOR DEVICE #1.
678 001266                          DHR1BUF:    .BLKB   256.    ;  HOLDS THE BINARY COUNT PATTERN FOR DEVICE #2.
679 001666                          DHR2BUF:    .BLKB   256.    ;  HOLDS THE BINARY COUNT PATTERN FOR DEVICE #3.
680 002266                          DHR3BUF:    .BLKB   256.    ;  HOLDS THE BINARY COUNT PATTERN FOR DEVICE #4.
681 002666  000000                  DHUVECT:    .WORD   0       ;  HOLDS THE VECTOR ADDRESS OF CURRENT DHU11.
682
683 002670  000000                  VA:         .WORD   0       ;  VIRTUAL ADDRESS.
684 002672  000000                  PA:         .WORD   0       ;  PHYSICAL ADDRESS.
685 002674  000000                  EA:         .WORD   0       ;  EXTENDED ADDRESS.
686 002676  000000                  PA22:       .WORD   0       ;  PHYSICAL ADDRESS (22 BIT).
687 002700  000000                  EA22:       .WORD   0       ;  EXTENDED ADDRESS (22 BIT).
688
689 002702  000000                  ADDR0:      .WORD   0       ;  HOLDS CSR ADDRESS OF DHU11 #1.
690 002704  000000                  ADDR1:      .WORD   0       ;  HOLDS CSR ADDRESS OF DHU11 #2.
691 002706  000000                  ADDR2:      .WORD   0       ;  HOLDS CSR ADDRESS OF DHU11 #3.
692 002710  000000                  ADDR3:      .WORD   0       ;  HOLDS CSR ADDRESS OF DHU11 #4.
693
694 002712  000000                  FLAG:       .WORD   0       ;  INDICATES IF DEVICE CAN INTERRUPT O.K.
695 002714  000030                  DHULPR:     .WORD   30      ;  LPR BEFORE SETTING BAUD RATES.
696 002716  000000                  LPRSELECT:  .WORD   0       ;  LPR AFTER SETTING BAUD RATES.  NEEDED BECAUSE
697                                                             ;  THE BAUD RATES ARE SET TO DEFAULTS DURING SELF TEST.
698
699
700
701
702                                 ;  THESE ARE THE ERROR MESSAGES
703
704 002720     045     106     101  ERR.A:  .ASCIZ  '%FATAL ERROR!!  CANNOT SELECT MORE THAN 4 LINES @19.2 K BAUD'!%
    002723     124     101     114
    002726     040     105     122
    002731     122     117     122
    002734     041     041     040
    002737     040     103     101
    002742     116     116     117
    002745     124     040     123
    002750     105     114     105
    002753     103     124     040
    002756     115     117     122
```

```
                002761  105     040     124
                002764  110     101     116
                002767  040     064     040
                002772  114     111     116
                002775  105     123     040
                003000  100     061     071
                003003  056     062     040
                003006  113     040     102
                003011  101     125     104
                003014  041     041     045
                003017  000
    705         003020  045     106     101     ERR.B:  .ASCIZ  '%FATAL ERROR!!  CANNOT SELECT MORE THAN 2 LINES @38.4 K BAUD!!%'
                003023  124     101     114
                003026  040     105     122
                003031  122     117     122
                003034  041     041     040
                003037  040     103     101
                003042  116     116     117
                003045  124     040     123
                003050  105     114     105
                003053  103     124     040
                003056  115     117     122
                003061  105     040     124
                003064  110     101     116
                003067  040     062     040
                003072  114     111     116
                003075  105     123     040
                003100  100     063     070
                003103  056     064     040
                003106  113     040     102
                003111  101     125     104
                003114  041     041     045
                003117  000
    706         003120  045     106     101     ERR.C:  .ASCIZ  '%FATAL ERROR!!  SELF-TEST TIMED OUT!!  DROP THE UNIT!!%'
                003123  124     101     114
                003126  040     105     122
                003131  122     117     122
                003134  041     041     040
                003137  040     123     105
                003142  114     106     055
                003145  124     105     123
                003150  124     040     124
                003153  111     115     105
                003156  104     040     117
                003161  125     124     041
                003164  041     040     040
                003167  104     122     117
                003172  120     040     124
                003175  110     105     040
                003200  125     116     111
                003203  124     041     041
                003206  045     000
    707         003210  045     106     101     ERR.D:  .ASCIZ  '%FATAL ERROR!!  SELF-TEST DIAGNOSTICS FAILED!!  DROP THE UNIT!!%'
                003213  124     101     114
                003216  040     105     122
                003221  122     117     122
                003224  041     041     040
```

MODULE DEFINITIONS

```
        003227  040     123     105
        003232  114     106     055
        003235  124     105     123
        003240  124     040     104
        003243  111     101     107
        003246  116     117     123
        003251  124     111     103
        003254  123     040     106
        003257  101     111     114
        003262  105     104     041
        003265  041     040     040
        003270  104     122     117
        003273  120     040     124
        003276  110     105     040
        003301  125     116     111
        003304  124     041     041
        003307  045     000
    708 003311  045     106     101  ERR.E:  .ASCIZ  '%FATAL ERROR!!  DATA VALID DID NOT SET!!  DROP THE UNIT!!%'
        003314  124     101     114
        003317  040     105     122
        003322  122     117     122
        003325  041     041     040
        003330  040     104     101
        003333  124     101     040
        003336  126     101     114
        003341  111     104     040
        003344  104     111     104
        003347  040     116     117
        003352  124     040     123
        003355  105     124     041
        003360  041     040     040
        003363  104     122     117
        003366  120     040     124
        003371  110     105     040
        003374  125     116     111
        003377  124     041     041
        003402  045     000
    709 003404  045     106     101  ERR.F:  .ASCIZ  '%FATAL ERROR!!  OVERRUN,PARITY, OR FRAMING ERROR!!  DROP THE UNIT!!%'
        003407  124     101     114
        003412  040     105     122
        003415  122     117     122
        003420  041     041     040
        003423  040     117     126
        003426  105     122     122
        003431  125     116     054
        003434  120     101     122
        003437  111     124     131
        003442  054     040     117
        003445  122     040     106
        003450  122     101     115
        003453  111     116     107
        003456  040     105     122
        003461  122     117     122
        003464  041     041     040
        003467  040     104     122
        003472  117     120     040
        003475  124     110     105
```

```
        003500  040  125  116
        003503  111  124  041
        003506  041  045  000
    710 003511  045  106  101  ERR.G:  .ASCIZ  '%FATAL ERROR!!  SELF-TEST ERROR CODE INDICATES AN ERROR!!  DROP THE UNIT!!%

        003514  124  101  114
        003517  040  105  122
        003522  122  117  122
        003525  041  041  040
        003530  040  123  105
        003533  114  106  055
        003536  124  105  123
        003541  124  040  105
        003544  122  122  117
        003547  122  040  103
        003552  117  104  105
        003555  040  111  116
        003560  104  111  103
        003563  101  124  105
        003566  123  040  101
        003571  116  040  105
        003574  122  122  117
        003577  122  041  041
        003602  040  040  104
        003605  122  117  120
        003610  040  124  110
        003613  105  040  125
        003616  116  111  124
        003621  041  041  045
        003624  000
    711 003625  045  105  122  ERR.H:  .ASCIZ  '%ERROR!!  ILLEGAL TRANSMIT INTERRUPT!!%'
        003630  122  117  122
        003633  041  041  040
        003636  040  111  114
        003641  114  105  107
        003644  101  114  040
        003647  124  122  101
        003652  116  123  115
        003655  111  124  040
        003660  111  116  124
        003663  105  122  122
        003666  125  120  124
        003671  041  041  045
        003674  000
    712 003675  045  105  122  ERR.I:  .ASCIZ  '%ERROR!!  OVERRUN,PARITY, OR FRAMING ERROR!!%'
        003700  122  117  122
        003703  041  041  040
        003706  040  117  126
        003711  105  122  122
        003714  125  116  054
        003717  120  101  122
        003722  111  124  131
        003725  054  040  117
        003730  122  040  106
        003733  122  101  115
        003736  111  116  107
        003741  040  105  122
        003744  122  117  122
```

```
        003747  041     041     045
        003752  000
    713 003753  045     105     122  ERR.J:  .ASCIZ  '%ERROR!!  DATA ERROR!!%'
        003756  122     117     122
        003761  041     041     040
        003764  040     104     101
        003767  124     101     040
        003772  105     122     122
        003775  117     122     041
        004000  041     045     000
    714 004003  045     105     122  ERR.K:  .ASCIZ  '%ERROR!!  NO LINES SELECTED!!%'
        004006  122     117     122
        004011  041     041     040
        004014  040     116     117
        004017  040     114     111
        004022  116     105     123
        004025  040     123     105
        004030  114     105     103
        004033  124     105     104
        004036  041     041     045
        004041  000
    715 004042  045     106     101  ERR.L:  .ASCIZ  '%FATAL ERROR!!  DEVICE DID NOT INTERRUPT PROPERLY!!  DROP THE UNIT!!%'
        004045  124     101     114
        004050  040     105     122
        004053  122     117     122
        004056  041     041     040
        004061  040     104     105
        004064  126     111     103
        004067  105     040     104
        004072  111     104     040
        004075  116     117     124
        004100  040     111     116
        004103  124     105     122
        004106  122     125     120
        004111  124     040     120
        004114  122     117     120
        004117  105     122     114
        004122  131     041     041
        004125  040     040     104
        004130  122     117     120
        004133  040     124     110
        004136  105     040     125
        004141  116     111     124
        004144  041     041     045
        004147  000
    716 004150  045     106     101  ERR.M:  .ASCII  '%FATAL ERROR!!  DEVICE DID NOT INTERRUPT PROPERLY!!%'
        004153  124     101     114
        004156  040     105     122
        004161  122     117     122
        004164  041     041     040
        004167  040     104     105
        004172  126     111     103
        004175  105     040     104
        004200  111     104     040
        004203  116     117     124
        004206  040     111     116
        004211  124     105     122
```

```
        004214  122     125     120
        004217  124     040     120
        004222  122     117     120
        004225  105     122     114
        004230  131     041     041
        004233  045
717     004234  045     103     110             .ASCIZ   %CHECK EXTERNAL CABLE OR TURNAROUND CONNECTION!!  DROP THE UNIT!!%'
        004237  105     103     113
        004242  040     105     130
        004245  124     105     122
        004250  116     101     114
        004253  040     103     101
        004256  102     114     105
        004261  040     117     122
        004264  040     124     125
        004267  122     116     101
        004272  122     117     125
        004275  116     104     040
        004300  103     117     116
        004303  116     105     103
        004306  124     111     117
        004311  116     041     041
        004314  040     040     104
        004317  122     117     120
        004322  040     124     110
        004325  105     040     125
        004330  116     111     124
        004333  041     041     045
        004336  000
718
719                                     .EVEN
720
721                                     .SBTTL INITIAL SETUP
```

```
                                ;) **
                                ;)   NAME:                        START: RESTART:
                                ;)
                                ;)   FUNCTIONAL DESCRIPTION:     START AND RESTART ARE ENTRY POINTS
                                ;)                               INTO THE CODE.
                                ;)                               START SETS DHU.LIVE UP FROM DVID1
                                ;)                               RESTART DOES THE 18 BIT CONVERTION
                                ;)                               ON THE TXBUF ADDRESS FOR DMA
                                ;)
                                ;)
                                ;)   CALLING SEQUENCE:          IN LINE.
                                ;)
                                ;)   INPUTS REQUIRED:           MODULE HEADER
                                ;)
                                ;)   OUTPUTS GIVEN:             "DHU.LIVE" AND THE 18 BIT ADDRESSES
                                ;)
                                ;)   REGISTERS USED,UNRESTORED: NONE.
                                ;)
                                ;)   SUBROUTINES USED:          NONE.
                                ;)
                                ;)   RESTRICTIONS:             NONE
                                ;)
                                ;) --
                                ;
004340                          START:
004340  016767  173450  173674          MOV     DVID1,DHU.LIVE          ;  SETUP # OF DHU11'S TO BE TESTED.
004346  001012                        BNE     RESTRT                  ;  IF THERE ARE NO DHU11'S TO TEST,
004350                          DROP:
004350  012701  000600                 MOV     #600,R1
004354                          DROP1:
004354  104407  000000'                BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
004360  104407  000000'                BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
004364  005301                        DEC     R1
004366  001372                        BNE     DROP1
004370  104410  000000'                END$,BEGIN              ;
004374                          RESTRT:
004374  042767  177760  173640          BIC     #^C<17>,DHU.LIVE        ;  # OF DHU11'S TO BE TESTED.
                                ;  WE MUST CONVERT THE 16 BIT VIRTUAL ADDRESS (TXBUF) TO AN 18 PHYSICAL
                                ;  ADDRESS
004402  012767  000266' 176260          MOV     #TXBUF,VA               ;  GET THE TRANSMIT BUFFER ADDRESS.
004410  104415  000000' 002670'        GETPA$,BEGIN, VA                ;GET PHYSICAL ADDRESS FROM 16-BIT VA
004416  016767  176250  176252          MOV     PA,PA22                 ;  NOW WE HAVE THE 18 BIT PHYSICAL ADDRESS AND
004424  016767  176244  176246          MOV     EA,EA22                 ;  THE 18 BIT EXTENDED ADDRESS.
004432  006267  176242                 ASR     EA22                    ;  SHIFT THE EXTENDED ADDRESS BITS
```

```
779 004436  006267  176236          ASR     EA22        ;  SO THEY ARE CORRECT FOR THE DMU11
780 004442  006267  176232          ASR     EA22        ;  NOTE GETPA PUTS THE EXTENDED ADDRESS
781 004446  006267  176226          ASR     EA22        ;  BITS IN BITS 4 AND 5
782
```

```
                                ;) **
                                ;)   NAME:                        SETUP1: ( INCLUDING SETUP2: )
                                ;)
                                ;)   FUNCTIONAL DESCRIPTION:     POINT TO FIRST BOARD
                                ;)                               AND SET UP THE VECTORS
                                ;)
                                ;)
                                ;)   CALLING SEQUENCE:           IN LINE.
                                ;)
                                ;)   INPUTS REQUIRED:            "DHU.LIVE" AND THE MODULE HEADER
                                ;)
                                ;)   OUTPUTS GIVEN:              'POINT"
                                ;)                               R1 = ADRRSES OF POINTER TO CURRENT DHU11
                                ;)                               R2 = ADDRESS OF CSR ( CURRENT DHU11 )
                                ;)                               R3 = RX INTERRUPT ROUTINE ADDRESS
                                ;)                               R4 = TX INTERRUPT ROUTINE ADDRESS
                                ;)
                                ;)   REGISTERS USED,UNRESTORED: R1 - 5 PUSHED ONTO STACK
                                ;)
                                ;)   SUBROUTINES USED:          NONE.
                                ;)
                                ;)   RESTRICTIONS:              NONE.
                                ;)
                                ;) --
                                ;
004452                          SETUP1:
004452  016701  173564                  MOV     DHU.LIVE,R1             ;  ANY DHU11'S TO TEST?
004456  001734                        BEQ     DROP                    ;  NO, DROP THE MODULE.
004460  012767  000001  173562        MOV     #1,POINT                ;  YES, SETUP DEVICE POINTER TO POINT TO 1ST DEVICE.
004466  016767  173314  173552        MOV     ADDR,DHUCSR             ;  GET THE CSR ADDRESS FOR FIRST DEVICE.
004474  016767  173310  176164        MOV     VECTOR,DHUVECT          ;  GET THE DEVICE VECTOR.
004502  012703  007566'                MOV     #DHRO,R3                ;  SETUP ADDRESS OF RECEIVER INTERRUPT ROUTINE.
004506  012704  007260'                MOV     #DHXO,R4                ;  SETUP ADDRESS OF TRANSMIT INTERRUPT ROUTINE.
004512  012701  002702'                MOV     #ADDRO,R1               ;  SETUP DEVICE ADDRESS TABLE FOR 1ST DEVICE.

                                ;  DETERMINE WHICH UNITS REQUIRE TESTING.

004516                          SETUP2:
004516  036767  173526  173516        BIT     POINT,DHU.LIVE          ;  DOES OPERATOR WANT THIS DHU11 TESTED?
004524  001002                        BNE     1$                      ;  YES, GO TEST IT.
004526  000167  001162                JMP     NEXTDHU                 ;  NO, GO GET ANOTHER DHU11.
004532                          1$:
004532  016702  173510                MOV     DHUCSR,R2               ;  YES, SAVE THE CSR ADDRESS OF THIS DHU11.


004536  010146                        MOV     R1,-(SP)                ;  SAVE REGISTERS.
```

INITIAL SETUP

```
     841 004540  010246                          MOV     R2, (SP)
     842 004542  010346                          MOV     R3, (SP)
     843 004544  010446                          MOV     R4, (SP)
     844 004546  010546                          MOV     R5, (SP)
     845
     846                                .SBTTL  TEST 1
```

```
                                ;) **
                                ;)  NAME:                          DHUT01
                                ;)
                                ;)  FUNCTIONAL DESCRIPTION:        THIS SECTION RESETS THE BOARD AND TESTS
                                ;)                                 THAT THE BOARD COMES OUT OF RESET
                                ;)
                                ;)
                                ;)  CALLING SEQUENCE:              IN LINE.
                                ;)
                                ;)  INPUTS REQUIRED:               R2 = CSR ADDRESS OF CURRENT DHU11
                                ;)
                                ;)  OUTPUTS GIVEN:                 "DHU.LIVE" WILL BE MODIFIED TO DROP A UNIT
                                ;)                                 ON A FAIL
                                ;)
                                ;)  REGISTERS USED,UNRESTORED:     R1 CORRUPTED
                                ;)
                                ;)  SUBROUTINES USED:              NONE.
                                ;)
                                ;)  RESTRICTIONS.                  NONE.
                                ;)
                                ;) --
                                :

004550                          DHUT01:                                 ;  FIRST WE INITIALIZE THE DEVICE.

004550  012701  004000                        MOV     #4000,R1          ;  SET A WATCHDOG TIMER
004554  032762  000040  000000  5$:     BIT     #MASTER.RESET,CSR(R2)   ;  SEE IF DHU11 IS IN RESET
004562  001407                        BEQ     8$                        ;  NO THEN WE CAN RESET IT

004564  104407  000000'                       BREAK$,BEGIN              ;TEMPORARY RETURN TO MONITOR....
004570  104407  000000'                       BREAK$,BEGIN              ;THEN CONTINUE AT NEXT INSTRUCTION.

004574  005301                        DEC     R1                        ;  HAVE WE TIMED OUT
004576  001366                        BNE     5$                        ;  NO TRY AGAIN
004600  000417                        BR      15$                       ;  YES GO TO THE ERROR REPORT
004602  012701  004000          8$:   MOV     #4000,R1                  ;  SET-UP WATCHDOG TIMER.
004606  012762  000040  000000        MOV     #MASTER.RESET,CSR(R2)     ;  SET MASTER RESET IN CSR.
004614                          10$:
004614  032762  000040  000000        BIT     #MASTER.RESET,CSR(R2)     ;  IS MASTER RESET STILL SET?
004622  001442                        BEQ     20$                       ;  NO, SELF-TEST IS DONE.

004624  104407  000000'                       BREAK$,BEGIN              ;TEMPORARY RETURN TO MONITOR....
004630  104407  000000'                       BREAK$,BEGIN              ;THEN CONTINUE AT NEXT INSTRUCTION.

004634  005301                        DEC     R1                        ;  DECREMENT WATCHDOG TIMER.
004636  001366                        BNE     10$                       ;  YES, WAIT FOR MASTER RESET TO CLEAR.



                                ;  SELF-TEST TIMED OUT!!  FATAL ERROR!!  DROP THE UNIT!!

004640                          15$:
004640  016767  173402  173232        MOV     DHUCSR,CSRA               ;  YES, MASTER RESET DIDN'T SET. REPORT AN ERROR.
004646  016267  000000  173226        MOV     CSR(R2),ACSR              ;  PRINTOUT THE CSR ADDRESS (CSRA), THE CONTENTS
004654  016267  000002  173222        MOV     RBUF(R2),ASTAT            ;  OF THE CSR (ACSR), AND RBUF (ASTAT).
004662  012767  000034  173216        MOV     #34,ERRTYP
```

```
903
904 004670  104401  000000  003120'         MSG$,BEGIN,ERR.C        ;ASCII MESSAGE CALL
905                                          ;*************************************************************
    004676  104405  000000' 000000          HRDER$,BEGIN,NULL       ;
                                             ;*************************************************************
906
907 004704  046767  173340  173330          BIC     POINT,DHU.LIVE          ;  FATAL ERROR!! DROP THIS DHU11 UNIT!!
908
909 004712  012605                           MOV     (SP)+,R5                ;  RESTORE REGISTERS.
910 004714  012604                           MOV     (SP)+,R4
911 004716  012603                           MOV     (SP)+,R3
912 004720  012602                           MOV     (SP)+,R2
913 004722  012601                           MOV     (SP)+,R1
914
915 004724  000167  000764                   JMP     NEXTDHU                 ;  GO BACK FOR ANOTHER DHU11.
916
917                                  .SBTTL  TEST 2
```

```
                        ;) --
                        ;)
                        ;)   NAME:                        DHUT02
                        ;)
                        ;)   FUNCTIONAL DESCRIPTION:      SEND ONE CHARACTER AROUND THE LOOP
                        ;)                                AND TEST THE RX AND TX INTERRUPTS
                        ;)
                        ;)
                        ;)   CALLING SEQUENCE:           IN LINE.
                        ;)
                        ;)   INPUTS REQUIRED:            R2 = ADDRESS OF DHU11 CSR TO BE TESTED
                        ;)                                'DHUVECT" , 'DHUCSR"
                        ;)
                        ;)   OUTPUTS GIVEN:              "DHU.LIVE' WILL BE MODIFIED TO DROP A UNIT
                        ;)                                ON A FAIL
                        ;)
                        ;)   REGISTERS USED,UNRESTORED: R1   5 PULLED FROM STACK
                        ;)
                        ;)   SUBROUTINES USED:           NONE.
                        ;)
                        ;)   RESTRICTIONS:               NONE.
                        ;)
                        ;) --

004730                  20$:

004730                  DHUT02:

004730                  30$:
004730  016200  000002            MOV     RBUF(R2),R0             ; CLEAR OUT THE RECEIVER FIFO.
004734  100001                    BPL     40$
004736  000774                    BR      30$

004740                  40$:
004740  016705  175722            MOV     DHUVECT,R5              ; POINT TO VECTOR ADDRESS.

004744  012725  005314'           MOV     @RCV,(R5)+              ; LOAD INTERRUPT RECEIVE ROUTINE ADDRESS.
004750  116725  173036            MOVB    BR1,(R5)+               ; LOAD PRIORITY.
004754  105025                    CLRB    (R5)+
004756  012725  005274'           MOV     @TX,(R5)+               ; LOAD INTERRUPT TRANSMIT ROUTINE ADDRESS.
004762  116725  173024            MOVB    BR1,(R5)+               ; LOAD PRIORITY.
004766  105015                    CLRB    (R5)

004770  005000                    CLR     R0                      ; START WITH LINE ZERO.
004772  012701  000001            MOV     #1,R1                   ; POINT TO IT.
004776                  50$:
004776  005067  175710            CLR     FLAG                    ; INIT. INTERRUPT INDICATOR.
005002  030167  173214            BIT     R1,LINESELECT           ; IS THIS LINE ON THE DHU11 SELECTED FOR TESTING?
005006  001525                    BEQ     100$                    ; NO, DON'T SET IT UP.

005010  010062  000000            MOV     R0,CSR(R2)              ; SELECT LINE NUMBER IN CSR.
                                                                  ; NOTE THE INITAL RESET SELECTED A
                                                                  ; XMIT AND RECEIVE BAUD OF 9600, NO PARITY,
                                                                  ; 1 STOP BIT, NORMAL OPERATION.  WE'RE
                                                                  ; NOT GOING TO CHANGE THAT HERE.
```

```
    976 005014  030167  173002                BIT     R1,SR3                  ;  DOES THE OPERATOR WANT TO TEST IN LOCAL LOOPBACK?
    977 005020  001404                        BEQ     60$                     ;  YES, SET UP FOR LOCAL LOOPBACK.
    978                                                                       ;  NO, SET UP IN NORMAL OPERATION.
    979
    980 005022  012762  000026  000010        MOV     #RX.ENABLE!AUTO1!AUTO0,LNCTRL(R2)
    981                                                                       ;  SET RX ENABLE & NORMAL
    982                                                                       ;  OPERATION IN LNCTRL.
    983 005030  000403                        BR      65$
    984
    985 005032                          60$:
    986 005032  012762  000226  000010        MOV     #LOCAL.LOOP!RX.ENABLE!AUTO1!AUTO0,LNCTRL(R2)
    987                                                                       ;  SET RX ENABLE & LOCAL
    988                                                                       ;  LOOPBACK IN LNCTRL.
    989
    990 005040                          65$:
    991
    992 005040  052762  100000  000014        BIS     #TX.ENABLE,TX2BUFADR(R2) ;  SET TX ENABLE
    993
    994 005046  010004                        MOV     R0,R4
    995 005050  052704  040100                BIS     #TXINTR.ENABLE!RCVINTR.ENABLE,R4
    996                                                                       ;  SET TRANSMIT & RECEIVE
    997                                                                       ;  INTERRUPT ENABLES.
    998 005054  010462  000000                MOV     R4,CSR(R2)
    999
   1000
   1001 005060  012703  020000                MOV     #20000,R3               ;  SET WATCHDOG TIMER.  WE WILL
   1002                                                                       ;  TIME OUT IF IT TAKES TOO LONG
   1003                                                                       ;  TO INTERRUPT SUCCESSFULLY.
   1004 005064  112762  000224  000006        MOVB    #TXDATA,TXCHAR(R2)      ;  PUT TXDATA IN THE TX FIFO
   1005                                                                       ;  TO SEND A CHARACTER.
   1006
   1007 005072                          70$:
   1008 005072  104407  000000'              BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
        005076  104407  000000'              BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
   1009
   1010 005102  005303                        DEC     R3                      ;  UPDATE COUNTDOWN.
   1011 005104  001052                        BNE     90$                     ;  HAS THE TIMER TIMED OUT?
   1012
   1013
   1014 005106  016767  173134  172764        MOV     DMUCSR,CSRA             ;  YES, THE DEVICE DID NOT INTERRUPT PROPERLY!
   1015                                                                       ;  REPORT AN ERROR.
   1016 005114  016267  000000  172760        MOV     CSR(R2),ACSR            ;  PRINTOUT THE CSR ADDRESS (CSRA), THE CONTENTS
   1017 005122  016267  000002  172754        MOV     RBUF(R2),ASTAT          ;  OF THE CSR (ACSR), AND RBUF (ASTAT).
   1018
   1019 005130                          75$:
   1020 005130  005762  000002                TST     RBUF(R2)                ;  FLUSH OUT RECEIVER SILO.
   1021 005134  100775                        BMI     75$
   1022 005136  012762  000000  000000        MOV     #0,CSR(R2)              ;  DISABLE TRANSMIT & RECEIVER INTERRUPTS.
   1023
   1024 005144  022767  000002  175540        CMP     #BIT1,FLAG              ;  DID TRANSMITTER INTERRUPT?
   1025 005152  001007                        BNE     80$                     ;  NO IT DIDN'T, REPORT INTERRUPT ERROR.
   1026 005154  030167  172642                BIT     R1,SR3                  ;  EXTERNAL LOOPBACK MODE (NORMAL)?
   1027 005160  001404                        BEQ     80$                     ;  NO, REPORT INTERRUPT ERROR.
   1028 005162  104401  000000'  004150'      MSG$,BEGIN,ERR.M        ;ASCII MESSAGE CALL
   1029 005170  000403                        BR      85$
   1030 005172                          80$:
   1031 005172  104401  000000'  004042'      MSG$,BEGIN,ERR.L        ;ASCII MESSAGE CALL
```

```
005200                              85$:
                                         ;*****************************************************
      005200  104405  000000' 000000     HRDER$,BEGIN,NULL          ;
                                         ;*****************************************************

005206  046767  173036  173026           BIC    POINT,DHU.LIVE          ; FATAL ERROR!! DROP THIS DHU11 UNIT!!

005214  012605                           MOV    (SP)+,R5                ; RESTORE REGISTERS.
005216  012604                           MOV    (SP)+,R4
005220  012603                           MOV    (SP)+,R3
005222  012602                           MOV    (SP)+,R2
005224  012601                           MOV    (SP)+,R1

005226  000167  000462                   JMP    NEXTDHU                 ; GO BACK FOR ANOTHER DHU11.

005232                              90$:
005232  022767  000006  175452           CMP    #6,FLAG                 ; HAVE WE INTERRUPTED SUCCESSFULLY?
005240  001314                           BNE    70$                     ; NO, WAIT FOR FLAG TO SET.

005242  012762  000000  000000           MOV    #0,CSR(R2)              ; DISABLE TRANSMIT & RECEIVER INTERRUPTS.
005250                              95$:
005250  005762  000002                   TST    RBUF(R2)                ; FLUSH OUT RECEIVER SILO.
005254  100775                           BMI    95$                     ; EMPTY YET
005256  005762  000000                   TST    CSR(R2)                 ; READ CSR TO CLEAR TRANSMIT INTERRUPT

005262                              100$:
005262  005200                           INC    R0                      ; UPDATE LINE NUMBER FOR CSR.
005264  006301                           ASL    R1                      ; POINT TO NEXT LINE.
005266  001243                           BNE    50$                     ; HAVE WE DONE 16 LINES?

005270  000167  000040                   JMP    EX                      ; FINISHED TESTING INTERRUPT CAPABILITIES.

005274                              TX:                                 ; TRANSMIT ROUTINE.
                                         ;-----------------------------------------------------
      005274  000004  000000' 005302'    PIRQ$,BEGIN,1$          ; QUEUE UP TO CONTINUE AT 1$ AND RTI
                                         ;-----------------------------------------------------
005302                              1$:
005302  052767  000002  175402           BIS    #BIT1,FLAG              ; SET FLAG INDICATING SUCCESSFUL TRANSMIT INTERRUPT

005310  104400  000000'                  EXIT$,BEGIN                    ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.

005314                              RCV:                                ; RECEIVE ROUTINE.
                                         ;-----------------------------------------------------
      005314  000004  000000' 005322'    PIRQ$,BEGIN,1$          ; QUEUE UP TO CONTINUE AT 1$ AND RTI
                                         ;-----------------------------------------------------
005322                              1$:
005322  052767  000004  175362           BIS    #BIT2,FLAG              ; SET FLAG INDICATING SUCCESSFUL RECEIVE INTERRUPT.
005330  104400  000000'                  EXIT$,BEGIN                    ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.

005334                              EX:
005334  012605                           MOV    (SP)+,R5                ; RESTORE REGISTERS.
005336  012604                           MOV    (SP)+,R4
005340  012603                           MOV    (SP)+,R3
005342  012602                           MOV    (SP)+,R2
005344  012601                           MOV    (SP)+,R1
```

```
                        ;) **
                        ;)
                        ;)   NAME:                      CHECKLINES:
                        ;)
                        ;)   FUNCTIONAL DESCRIPTION:    CHECKS THAT WE HAVE LINES SELECED AND
                        ;)                              NOT TOO MANY LINES IF ON HIGH BAUD
                        ;)                              RATES, CHECK FOR DEFAULT ( SR2 = 0 )
                        ;)                              SET UP THE BAUDRATE
                        ;)
                        ;)   CALLING SEQUENCE:          IN LINE.
                        ;)
                        ;)   INPUTS REQUIRED:           R2 = CURRENT DHU11 CSR
                        ;)                              "DHU.LIVE" AND THE MODULE HEADER
                        ;)
                        ;)   OUTPUTS GIVEN:             NONE.
                        ;)
                        ;)   REGISTERS USED,UNRESTORED: ALL
                        ;)
                        ;)   SUBROUTINES USED:          BAUDRATE
                        ;)
                        ;)   RESTRICTIONS:              NONE.
                        ;)
                        ;) --

005346                  CHECKLINES:

005346  022767  000000  172646          CMP     #0,LINESELECT           ;  MAKE SURE THERE ARE LINES SELECTED.
005354  001024                          BNE     3$                      ;  YES, THERE ARE.  GO TEST THEM.

005356  016767  172664  172514          MOV     DHUCSR,CSRA             ;  NO, THERE ARE NO LINES SELECTED.  REPORT AN ERROR

005364  016267  000000  172510          MOV     CSR(R2),ACSR            ;  PRINTOUT THE CSR ADDRESS (CSRA), THE CONTENTS
005372  016267  000002  172504          MOV     RBUF(R2),ASTAT          ;  OF THE CSR (ACSR), AND RBUF (ASTAT).

005400  104401  000000' 004003'          MSG$,BEGIN,ERR.K        ;ASCII MESSAGE CALL
                                        ;*************************************************************\***
005406  104405  000000' 000000          MRDER$,BEGIN,NULL       ;
                                        ;*************************************************************\*

005414  046767  172630  172620          BIC     POINT,DHU.LIVE          ;  FATAL ERROR!! DROP THIS DHU11 'NIT!!
005422  000167  000266                  JMP     NEXTDHU                 ;  GO BACK FOR ANOTHER DHU11.

005426                  3$:
005426  012700  000020                  MOV     #16.,R0                 ;  SET-UP BIT COUNTER.
005432  005067  172576                  CLR     LINECOUNT               ;  INITIALIZE THE "LINES SELECTED INDICATOR".
005436  022767  000000  172354          CMP     #0,SR2                  ;  DOES OPERATOR WANT TO RUN AT 9600 BAUD?
005444  001005                          BNE     5$                      ;  NO, DETERMINE THE BAUD RATE DESIRED.

005446  012767  000015  172342          MOV     #15,SR1                 ;  YES, SET-UP BAUD USING 9600.
005454  000167  000200                  JMP     12$                     ;  GO DO THE SET-UP.

005460                  5$:                                             ;  FIRST, HOWEVER, WE MUST DETERMINE HOW MANY
                                                                        ;  LINES HAVE BEEN SELECTED FOR TESTING.

005460  006167  172536                  ROL     LINESELECT              ;  PLACE THE LINESELECT BIT INTO THE CARRY BIT.
005464  103002                          BCC     7$                      ;  IS THIS LINE SELECTED?
```

```
1139 005466  005267  172542                INC     LINECOUNT               ;  YES, INDICATE THAT IT IS.
1140
1141
1142
1143
1144 005472                          7$:
1145 005472  077006                       SOB     R0,5$                   ;  UPDATE COUNT-DOWN UNTIL ALL BITS ARE CHECKED.
1146
1147 005474  006167  172522                ROL     LINESELECT              ;  RESTORE LINESELECT.
1148 005500  022767  000016  172310        CMP     #16,SR1                 ;  IS SELECTED BAUD RATE 19.2K?
1149 005506  001030                       BNE     10$                     ;  NO, SEE IF IT IS SET FOR 38.4K.
1150 005510  022767  000004  172516        CMP     #4,LINECOUNT            ;  YES.  ARE THERE MORE THAN 4 LINES SELECTED?
1151 005516  002060                       BGE     12$                     ;  NO, THAT'S GOOD.  GO SET-UP THE BAUD RATE.
1152
1153 005520  016767  172522  172352        MOV     DHUCSR,CSRA             ;  YES, TOO MANY LINES SELECTED.  REPORT AN ERROR.
1154 005526  016267  000000  172346        MOV     CSR(R2),ACSR            ;  PRINTOUT THE CSR ADDRESS (CSRA), THE CONTENTS
1155 005534  016267  000002  172342        MOV     RBUF(R2),ASTAT          ;  OF THE CSR (ACSR), AND RBUF (ASTAT).
1156
1157 005542  104401  000000' 002720'        MSG$,BEGIN,ERR.A        ;ASCII MESSAGE CALL
1158                                                                  ;  RUNNING CONCURRENTLY AT 19.2K BAUD!
1159                                        ;*****************************************************************
     005550  104405  000000' 000000        MRDER$,BEGIN,NULL       ;
                                            ;*****************************************************************
1160
1161 005556  046767  172466  172456        BIC     POINT,DHU.LIVE          ;  FATAL ERROR!! DROP THIS DHU11 UNIT!!
1162 005564  000167  000124                JMP     NEXTDHU                 ;  GO BACK FOR ANOTHER DHU11.
1163
1164 005570                          10$:
1165
1166 005570  022767  000017  172220        CMP     #17,SR1                 ;  IS SELECTED BAUD RATE 38.4K?
1167 005576  001030                       BNE     12$                     ;  NO, GO SETUP ITS BAUD RATE.
1168 005600  022767  000002  172426        CMP     #2,LINECOUNT            ;  YES.  ARE THERE MORE THAN 2 LINES SELECTED?
1169 005606  002024                       BGE     12$                     ;  NO, GO SET-UP THE BAUD RATE.
1170
1171 005610  016767  172432  172262        MOV     DHUCSR,CSRA             ;  YES, TOO MANY LINES SELECTED.  REPORT AN ERROR.
1172 005616  016267  000000  172256        MOV     CSR(R2),ACSR            ;  PRINTOUT THE CSR ADDRESS (CSRA), THE CONTENTS
1173 005624  016267  000002  172252        MOV     RBUF(R2),ASTAT          ;  OF THE CSR (ACSR), AND RBUF (ASTAT).
1174
1175 005632  104401  000000' 003020'        MSG$,BEGIN,ERR.B        ;ASCII MESSAGE CALL
1176                                                                  ;  RUNNING CONCURRENTLY AT 38.4K BAUD!
1177
1178                                        ;*****************************************************************
     005640  104405  000000' 000000        MRDER$,BEGIN,NULL       ;
                                            ;*****************************************************************
1179
1180 005646  046767  172376  172366        BIC     POINT,DHU.LIVE          ;  FATAL ERROR!! DROP THIS DHU11 UNIT!!
1181 005654  000167  000034                JMP     NEXTDHU                 ;  GO BACK FOR ANOTHER DHU11.
1182
1183 005660                          12$:
1184 005660  004767  002704                JSR     PC,BAUDRATE             ;  GO SET-UP THE BAUDRATE.
1185
1186
1187
1188
```

```
1190                                    ;) **
1191                                    ;)
1192                                    ;)   NAME:                        VECT
1193                                    ;)
1194                                    ;)   FUNCTIONAL DESCRIPTION:      SET UP VECTORS FOR EACH BOARD
1195                                    ;)
1196                                    ;)
1197                                    ;)   CALLING SEQUENCE:            IN LINE.
1198                                    ;)
1199                                    ;)   INPUTS REQUIRED:             MODULE HEADER,
1200                                    ;)                                "DHUCSR" AND "DHUVECT
1201                                    ;)
1202                                    ;)   OUTPUTS GIVEN:               VECTOR TABLE AT CURRENT DHU11 VECTOR ADDRESS
1203                                    ;)
1204                                    ;)   REGISTERS USED,UNRESTORED:   R1 , 3 , 4 , 5
1205                                    ;)
1206                                    ;)   SUBROUTINES USED:            NONE.
1207                                    ;)
1208                                    ;)   RESTRICTIONS:                NONE.
1209                                    ;)
1210                                    ;) --
1211
1212
1213
1214 005664                             VECT:
1215
1216 005664  016705  174776                    MOV     DHUVECT,R5          ;  POINT TO THE VECTOR.
1217 005670  016711  172352                    MOV     DHUCSR,(R1)         ;  MOVE THE CSR ADDRESS INTO THE ADDRESS TABLE.
1218 005674  010325                            MOV     R3,(R5)+            ;  MOVE THE ADDRESS OF THE RECEIVER INTERRUPT
1219                                                                       ;  ROUTINE & THE PRIORITY OF THIS DHU11 INTO
1220 005676  116725  172110                    MOVB    BR1,(R5)+           ;  THE VECTOR ADDRESS FOR THIS DHU11.
1221 005702  105025                            CLRB    (R5)+
1222 005704  010425                            MOV     R4,(R5)+            ;  MOVE THE ADDRESS OF THE TRANSMIT INTERRUPT
1223                                                                       ;  ROUTINE & THE PRIORITY OF THIS DHU11 INTO
1224 005706  116725  172100                    MOVB    BR1,(R5)+           ;  THE VECTOR ADDRESS FOR THIS DHU11.
1225 005712  105015                            CLRB    (R5)
1226
```

```
1228                                    ;) **
1229                                    ;)
1230                                    ;)   NAME:                        NEXTDHU
1231                                    ;)
1232                                    ;)   FUNCTIONAL DESCRIPTION:      SET UP POINTERS FOR THE NEXT DHU11
1233                                    ;)                                GOTO NEXT TEST IF ALL DHU11'S DONE
1234                                    ;)
1235                                    ;)   CALLING SEQUENCE:            IN LINE.
1236                                    ;)
1237                                    ;)   INPUTS REQUIRED:             "POINT" , "DHUCSR" AND "DHUVECT"
1238                                    ;)
1239                                    ;)   OUTPUTS GIVEN:               "POINT" , "DHUCSR" AND "DHUVECT"
1240                                    ;)
1241                                    ;)   REGISTERS USED,UNRESTORED:   R1 , 3 , 4
1242                                    ;)
1243                                    ;)   SUBROUTINES USED:            NONE.
1244                                    ;)
1245                                    ;)   RESTRICTIONS:                NONE.
1246                                    ;)
1247                                    ;) -
1248
1249
1250 005714                            NEXTDHU:
1251
1252 005714  006367  172330                    ASL     POINT                   ;  POINT TO NEXT DHU11.
1253 005720  062701  000002                    ADD     #2,R1                   ;  POINT TO NEXT LOC. IN ADDRESS TABLE.
1254 005724  062703  000034                    ADD     #DHR1-DHR0,R3           ;  POINT TO NEXT DHU11'S RX INTERRUPT ROUTINE.
1255 005730  062704  000020                    ADD     #DHX1-DHX0,R4           ;  POINT TO NEXT DHU11'S TX INTERRUPT ROUTINE.
1256
1257 005734  032767  000020  172306           BIT     #20,POINT               ;  HAVE WE SETUP ALL DHU11S?
1258 005742  001010                            BNE     DHUT03                  ;  YES, LET'S GO TEST THEM!
1259
1260 005744  062767  000020  172274           ADD     #20,DHUCSR              ;  NO, GO SETUP ANOTHER DHU11.
1261 005752  062767  000010  174706           ADD     #10,DHUVECT
1262 005760  000167  176532                    JMP     SETUP2
1263
1264
1265
1266                                    .SBTTL  TEST 3
```

```
                                ;) **
                                ;)
                                ;)   NAME:                        DHUT03
                                ;)
                                ;)   FUNCTIONAL DESCRIPTION:     RESET BOARD AND TEST STATE OF
                                ;)                               DIAG.FAIL BIT AND SELF TEST CODES
                                ;)
                                ;)
                                ;)   CALLING SEQUENCE:           IN LINE.
                                ;)
                                ;)   INPUTS REQUIRED:            "DHUCSR" , "DHU.LIVE"
                                ;)
                                ;)   OUTPUTS GIVEN:              "DHU.LIVE" WILL BE MODIFIED TO DROP A UNIT
                                ;)                               ON A FAIL
                                ;)
                                ;)   REGISTERS USED,UNRESTORED: ALL
                                ;)
                                ;)   SUBROUTINES USED:          NONE.
                                ;)
                                ;)   RESTRICTIONS:              NONE.
                                ;)
                                ;) --


005764                          DHUT03:

005764  012767  000001  172256          MOV     #1,POINT                ;  SETUP DEVICE POINTER TO POINT TO 1ST DEVICE.
005772  016767  172010  172246          MOV     ADDR,DHUCSR             ;  SETUP ADDRESS OF 1ST DEVICE.

006000                          DHU.L1:

006000  036767  172244  172234          BIT     POINT,DHU.LIVE          ;  DOES THE OPERATOR WANT TO TEST THIS DEVICE?
006006  001002                          BNE     3$                      ;  YES, GO TEST IT.
006010  000167  000534                  JMP     DHU.N1                  ;  NO, GO LOOK AT THE NEXT DHU11.
006014                          3$:
006014  016702  172226                  MOV     DHUCSR,R2               ;  SETUP THIS DHU11'S DEVICE ADDRESS.



                                ;  THE ONBOARD SELF-TEST DIAGNOSTIC WILL NOW BE RUN.
                                ;  THIS IS ACCOMPLISHED BY SETTING THE MASTER RESET IN THE CSR.  THE
                                ;  CSR MASTER RESET BIT IS CLEARED WHEN THE SELF TEST IS COMPLETED.

006020  012701  004000                  MOV     #4000,R1                ;  SET A WATCHDOG TIMER
006024  032762  000040  000000  5$:     BIT     #MASTER.RESET,CSR(R2)   ;  SEE IF DHU11 IS IN RESET
006032  001407                          BEQ     8$                      ;  NO THEN WE CAN RESET IT

006034  104407  000000'                 BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
006040  104407  000000'                 BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.

006044  005301                          DEC     R1                      ;  HAVE WE TIMED OUT
006046  001366                          BNE     5$                      ;  NO TRY AGAIN
006050  000417                          BR      15$                     ;  YES GO TO THE ERROR REPORT
```

```
1324 006052  012701  004000          8$:     MOV     #4000,R1                ;  SET UP WATCHDOG TIMER.
1325 006056  012762  000040  000000          MOV     #MASTER.RESET,CSR(R2)   ;  SET MASTER RESET IN CSR.
1326 006064                          10$:
1327 006064  032762  000040  000000          BIT     #MASTER.RESET,CSR(R2)   ;  IS MASTER RESET STILL SET?
1328 006072  001435                          BEQ     20$                     ;  NO, SELF-TEST IS DONE.
1329
1330 006074  104407  000000'                 BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
     006100  104407  000000'                 BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
1331
1332 006104  005301                          DEC     R1                      ;  DECREMENT WATCHDOG TIMER.
1333 006106  001366                          BNE     10$                     ;  YES, WAIT FOR MASTER RESET TO CLEAR.
1334 006110                          15$:
1335
1336
1337
1338                                 ;  SELF-TEST TIMED OUT!!  FATAL ERROR!!  DROP THE UNIT!!
1339
1340
1341 006110  016767  172132  171762          MOV     DHUCSR,CSRA             ;  YES, MASTER RESET DIDN'T SET. REPORT AN ERROR.
1342 006116  016267  000000  171756          MOV     CSR(R2),ACSR            ;  PRINTOUT THE CSR ADDRESS (CSRA), THE CONTENTS
1343 006124  016267  000002  171752          MOV     RBUF(R2),ASTAT          ;  OF THE CSR (ACSR), AND RBUF (ASTAT).
1344 006132  012767  000034  171746          MOV     #34,ERRTYP
1345
1346 006140  104401  000000' 003120'         MSG$,BEGIN,ERR.C        ;ASCII MESSAGE CALL
1347                                         ;*************************************************************
     006146  104405  000000' 000000          HRDER$,BEGIN,NULL       ;
                                             ;*************************************************************
1348
1349 006154  046767  172070  172060          BIC     POINT,DHU.LIVE          ;  FATAL ERROR!! DROP THIS DHU11 UNIT!!
1350 006162  000167  000362                  JMP     DHU.N1                  ;  GO BACK FOR ANOTHER DHU11.
1351
1352
1353
1354                                 ;  THE SELF-TEST DID COMPLETE FOR THIS DHU11.  BUT, WERE THERE ANY FAILURES?
1355
1356
1357 006166                          20$:
1358 006166  032762  020000  000000          BIT     #DIAG.FAIL,CSR(R2)      ;  DID SELF-TEST DIAGNOSTICS PASS?
1359 006174  001426                          BEQ     30$                     ;  YES, IF "DIAGNOSTICS FAILURE" BIT CLEAR.
1360
1361
1362
1363                                 ;  SELF-TEST DIAGNOSTICS FAILED!!  FATAL ERROR!!  DROP THE UNIT!!
1364
1365
1366 006176  016767  172044  171674          MOV     DHUCSR,CSRA             ;  REPORT CSR.
1367 006204  016267  000000  171670          MOV     CSR(R2),ACSR            ;  REPORT CSR CONTENTS.
1368 006212  010567  171666                  MOV     R5,ASTAT                ;  REPORT CONTENTS OF RBUF.
1369 006216  012767  000034  171662          MOV     #34,ERRTYP              ;  REPORT ERROR TYPE.
1370
1371 006224  104401  000000' 003210'         MSG$,BEGIN,ERR.D        ;ASCII MESSAGE CALL
1372                                         ;*************************************************************
     006232  104405  000000' 000000          HRDER$,BEGIN,NULL       ;
                                             ;*************************************************************
1373
1374 006240  046767  172004  171774          BIC     POINT,DHU.LIVE          ;  REMOVE THIS FAULTY DEVICE FROM TESTING.
1375 006246  000167  000276                  JMP     DHU.N1                  ;  GO GET NEXT DHU11.
```

```
                                ;  SELF TEST PASSED O.K.

006252                          30$:
006252  012704  000010                MOV     #8.,R4                  ;  SETUP TO READ THE 8 CHAR. CODES FROM RBUF.
006256  012701  007640                MOV     #4000.,R1               ;  SETUP WATCHDOG TIMER.
006262                          40$:
006262  016203  000002                MOV     RBUF(R2),R3             ;  READ RBUF FIFO.  A VALID CHAR. WILL SET BIT
006266  100434                      BMI     50$                     ;  #15 (FIFO NOT EMPTY).

006270  104407  000000'              BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
006274  104407  000000'              BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.

006300  005301                      DEC     R1                      ;  DECREMENT WATCHDOG TIMER.
006302  001367                      BNE     40$                     ;  WAIT. WE WANT TO READ RBUF FIFO.



                                ;  DATA VALID DID NOT SET!!  FATAL ERROR!!  DROP THIS DHU11!!

                                                                        ;  BIT #15 DIDN'T SET!  REPORT AN ERROR!
006304  016767  171736  171566        MOV     DHUCSR,CSRA             ;  REPORT CSR.
006312  016267  000000  171562        MOV     CSR(R2),ACSR            ;  REPORT CSR CONTENTS.
006320  010367  171560                MOV     R3,ASTAT                ;  REPORT CONTENTS OF RBUF.
006324  012767  000034  171554        MOV     #34,ERRTYP              ;  REPORT ERROR TYPE.

006332  104401  000000' 003311'      MSG$,BEGIN,ERR.E        ;ASCII MESSAGE CALL
                                ;*****************************************************************
006340  104405  000000' 000000      MRDER$,BEGIN,NULL       ;
                                ;*****************************************************************

006346  046767  171676  171666        BIC     POINT,DHU.LIVE          ;  REMOVE THIS FAULTY DHU11 FROM TESTING.
006354  000167  000170                JMP     DHU.N1                  ;  GO GET NEXT DHU11.



                                ;  IT IS A VALID CHARACTER.  CHECK BITS 12-14 FOR A POSSIBLE ERROR.

006360                          50$:
006360  032703  070000                BIT     #OVERRUN.ERROR!FRAMING.ERROR!PARITY.ERROR,R3
                                                                        ;  IS IT MODEM STATUS,
                                                                        ;  SELF-TEST ERROR CODE,
                                                                        ;  OR DATA?
006364  001736                      BEQ     40$                     ;  IF NONE SET THEN IT IS DATA.  IT DOESN'T COUNT.
006366  062703  010000                ADD     #BIT12,R3               ;  MAKE SURE ALL THE BITS WERE SET!
006372  100026                      BPL     60$                     ;  THEY WERE.  THAT'S GOOD.



                                ;  OVERRUN, PARITY OR FRAMING ERROR!!  FATAL ERROR!'  DROP THIS UNIT!!
```

```
1430                                                                    ;  IF OVERFLOW TO BIT 15 DID NOT OCCUR,
1431                                                                    ;  THEN WE KNOW ALL 3 BITS WERE NOT SET!
1432                                                                    ;  REPORT AN ERROR.
1433 006374  016767  171646  171476          MOV     DHUCSR,CSRA       ;  REPORT CSR.
1434 006402  016267  000000  171472          MOV     CSR(R2),ACSR      ;  REPORT CSR CONTENTS.
1435 006410  010367  171470                  MOV     R3,ASTAT          ;  REPORT RBUF CONTENTS.
1436 006414  012767  000034  171464          MOV     #34,ERRTYP        ;  REPORT ERROR TYPE.
1437
1438 006422  104401  000000' 003404'         MSG$,BEGIN,ERR.F        ;ASCII MESSAGE CALL
1439                                          ;*****************************************************************
     006430  104405  000000' 000000          HRDER$,BEGIN,NULL       ;
                                              ;*****************************************************************
1440
1441 006436  046767  171606  171576          BIC     POINT,DHU.LIVE    ;  REMOVE THIS FAULTY DHU11 FROM TESTING.
1442 006444  000167  000100                  JMP     DHU.N1            ;  GO GET NEXT DHU11.
1443
1444
1445
1446                                  ;  THE "ALL ONES" CODE FOR BITS 12-14 IN RBUF INDICATES THAT MODEM
1447                                  ;  STATUS OR SELF-TEST DIAGNOSTIC INFO.  IS BEING HELD IN RBUF BITS 0-7.
1448                                  ;  WE WANT TO LOOK AT THE SELF-TEST CODE TO ASSURE THAT THE SELF-TEST
1449                                  ;  COMPLETED O.K.
1450
1451
1452 006450                           60$:
1453 006450  032703  000001                  BIT     #BIT0,R3          ;  IS THIS MODEM STATUS OR SELF-TEST ERROR CODE?
1454 006454  001702                          BEQ     40$               ;  IGNOR THE MODEM STATUS. WE JUST WANT
1455                                                                    ;  SELF-TEST ERROR CODES.
1456
1457 006456  105703                          TSTB    R3                ;  LOOK TO SEE IF ROM VERSION OR SELF-TEST ERROR COD
E.
1458 006460  100031                          BPL     70$               ;  IF BIT7=0 THEN ROM VERSION. READ AND IGNOR.
1459 006462  120327  000201                  CMPB    R3,#201           ;  DID WE GET SUCCESSFUL COMPLETION CODE?
1460 006466  001426                          BEQ     70$               ;  IF SO, CODE 201 RETURNED.  ANY OTHER CODE IS
1461                                                                    ;  AN ERROR!
1462
1463
1464
1465                                  ;  SELF-TEST ERROR CODE INDICATES AN ERROR!!  FATAL ERROR!!  DROP THIS DHU11'!
1466
1467                                                                    ;  REPORT AN ERROR.
1468 006470  016767  171552  171402          MOV     DHUCSR,CSRA       ;  REPORT CSR.
1469 006476  016267  000000  171376          MOV     CSR(R2),ACSR      ;  REPORT CSR CONTENTS.
1470 006504  010367  171374                  MOV     R3,ASTAT          ;  REPORT RBUF CONTENTS.
1471 006510  012767  000034  171370          MOV     #34,ERRTYP        ;  REPORT ERROR TYPE.
1472
1473 006516  104401  000000' 003511'         MSG$,BEGIN,ERR.G        ;ASCII MESSAGE CALL
1474                                          ;*****************************************************************
     006524  104405  000000' 000000          HRDER$,BEGIN,NULL       ;
                                              ;*****************************************************************
1475
1476 006532  046767  171512  171502          BIC     POINT,DHU.LIVE    ;  REMOVE THIS FAULTY DHU11 FROM TESTING.
1477 006540  000167  000004                  JMP     DHU.N1            ;  GO GET NEXT DHU11.
1478
1479
1480
1481
1482
```

```
                                ;  THESE ARE THE SELF TEST ERROR CODES
                                ;
                                ;
                                ; CODE                    MEANING
                                ;
                                ; 201                SELF TEST NULL CODE (OK)
                                ; 203                SELF TEST CODE SKIPPED
                                ; 211                BASIC DATA PATH ERROR FRO PROC2
                                ; 213                UNDEFINED UART ERROR
                                ; 225                PROC1 TO EXTERNAL RAM ERROR
                                ; 227                PROC2 TO EXTERNAL RAM ERROR
                                ; 231                PROC1 TO INTERNAL RAM ERROR
                                ; 233                PROC2 TO INTERNAL RAM ERROR
                                ; 235                PROC1 ROM ERROR
                                ; 237                PROC2 ROM ERROR
                                ; 241-277            UART ACCESS OR FUNCTION
                                ;
                                ; BIT 7=0 AND BIT0=1 ROM VERSION
                                ; ALL OTHER CODES ARE UNDEFINED ERRORS.


006544                          70$:
006544  005304                          DEC     R4                      ;  NEXT CHARACTER.
006546  001245                          BNE     40$                     ;  HAVE WE DONE ALL 16 CHARACTERS?
                                                                        ;  IF NOT, GO BACK & DO THEM.
```

```
                                        ;) **
                                        ;)
                                        ;)   NAME:                          DHU.N1
                                        ;)
                                        ;)   FUNCTIONAL DESCRIPTION:       SET UP POINTERS FOR THE NEXT DHU11
                                        ;)                                 GOTO NEXT TEST IF ALL DHU11 S DONE
                                        ;)
                                        ;)   CALLING SEQUENCE:             IN LINE.
                                        ;)
                                        ;)   INPUTS REQUIRED:              "POINT" AND 'DHUCSR
                                        ;)
                                        ;)   OUTPUTS GIVEN:                "POINT" AND "DHUCSR"
                                        ;)
                                        ;)   REGISTERS USED,UNRESTORED:    NONE
                                        ;)
                                        ;)   SUBROUTINES USED:             NONE.
                                        ;)
                                        ;)   RESTRICTIONS:                 NONE.
                                        ;)
                                        ;) --
006550                                  DHU.N1:
006550  005767  171466                          TST     DHU.LIVE                ;  ANY DHU11 S TO TEST?
006554  001002                                BNE     10$
006556  000167  175566                        JMP     DROP
006562                                  10$:
006562  106367  171462                        ASLB    POINT                   ;  LOOK AT NEXT DHU11.
006566  032767  000020  171454                BIT     #20,POINT               ;  HAVE WE TESTED ALL DHU11'S.
006574  001005                                BNE     DHUT04                  ;  YES, GOTO NEXT TEST SECTION.
006576  062767  000020  171442                ADD     #20,DHUCSR              ;  NO, GET NEXT DHU11 ADDRESS.
006604  000167  177170                        JMP     DHU.L1                  ;  GO BACK AND TEST.
```

```
                                        .SBTTL  TEST 4
```

```
                                ;
                                ;) ..
                                ;)
                                ;)   NAME:                          DMUT04
                                ;)
                                ;)   FUNCTIONAL DESCRIPTION:       THIS SECTION SETS UP THE DMU11 TO PASS DATA IN
                                ;)                                 INTERNAL LOOPBACK MODE OR TO LOOPBACK DATA
                                ;)                                 EXTERNALLY IN NORMAL MODE.
                                ;)
                                ;)
                                ;)   CALLING SEQUENCE:             IN LINE.
                                ;)
                                ;)   INPUTS REQUIRED:              MODULE HEADER
                                ;)
                                ;)   OUTPUTS GIVEN:                'LINESELECT' , 'DMUSLEFT
                                ;)
                                ;)   REGISTERS USED,UNRESTORED:    ALL
                                ;)
                                ;)   SUBROUTINES USED:             NONE.
                                ;)
                                ;)   RESTRICTIONS:                 NONE.
                                ;)
                                ;)  .


006610                          DMUT04:

006610  005067  171444                  CLR     LCKOUT                  ;  INIT. LOCKOUT.
006614  005067  171404                  CLR     LINESDONE               ;  INIT. LINE DONE INDICATOR.
006620  005067  171402                  CLR     LINESDONE+2             ;  INIT. LINE DONE INDICATOR.
006624  005067  171400                  CLR     LINESDONE+4             ;  INIT. LINE DONE INDICATOR.
006630  005067  171376                  CLR     LINESDONE+6             ;  INIT. LINE DONE INDICATOR.

006634  005067  171404                  CLR     DMUNOW                  ;  START WITH DEVICE #0.
006640  016767  171142  171400          MOV     ADDR,DMUCSR             ;  GET ADDRESS OF FIRST DMU11.
006646  016767  171136  174012          MOV     VECTOR,DMUVECT          ;  GET VECTOR OF FIRST DMU11.

006654  012767  000001  171366          MOV     #1,POINT                ;  RECORD WORKING ON 1ST DMU11.
006662  005067  171350                  CLR     DMUSDONE                ;  CLEAR THE INDICATOR OF HOW MANY DMU11'S
                                                                        ;  ARE DONE,
006666  005067  171346                  CLR     DMUSLEFT                ;  AND HOW MANY ARE LEFT TO DO.

006672                          DMU.L2:

006672  036767  171352  171342          BIT     POINT,DMU.LIVE          ;  DOES OPERATOR WANT THIS DMU11 TESTED?
006700  001503                          BEQ     DMU.N2                  ;  NO, GO LOOK AT THE NEXT DMU11.

006702  016702  171340                  MOV     DMUCSR,R2               ;  SAVE DMU11 ADDRESS FOR SUBROUTINES.



                                ;  SETUP THE COUNT PATTERN TO SEND VIA THE TRANSMITTER.
                                ;  THE COUNT PATTERN IS 224-377 (148-255 DECIMAL).


006706                          DMU.BUF:
```

```
     006706 012700 000266          MOV     #TXBUF,R0               ; NOW WRITE COUNT PATTERN INTO XMIT BUFFER.
     006712 012701 000224          MOV     #224,R1                 ; START WITH 224 UP TO 377 (148-255 DECIMAL).
     006716                   10$:
     006716 110120                 MOVB    R1,(R0)+                ; STORE PATTERN.
     006720 105201                 INCB    R1                      ; UPDATE PATTERN UNTIL PATTERN 377 IS
                                                                   ; STORED.
     006722 001375                 BNE     10$                     ; HAVE WE STORED UP TO 377 YET?

                              ; INITIALISE THE RX BUFFERS IN USE WITH THE FIRST 16 BYTES AS THE FIRST
                              ; EXPECTED CHARACTER AND THE REST CLEARED

     006724                   20$:
     006724 012700 000666'         MOV     #DHR0BUF,R0             ; GET RECEIVER BUFFER ADDRESS.
     006730 016701 171310          MOV     DHUNOW,R1
     006734                   30$:
     006734 005301                 DEC     R1                      ; IS THIS DHU11 #0?
     006736 100403                 BMI     40$                     ; YES THEN R0 IS CORRECT
     006740 062700 000400          ADD     #DHR1BUF-DHR0BUF,R0     ; NO UPDATE R0 FOR NEXT BUFFER ADDRESS
     006744 000773                 BR      30$                     ; DO MORE UNTIL CORRECT BUFFER ADDRESS OBTAINED.

     006746                   40$:
     006746 005267 171266          INC     DHUSLEFT                ; KEEP TRACK OF HOW MANY DHU11'S HAVE TO BE DONE.

     006752 012701 000020          MOV     #16.,R1                 ; SETUP COUNT FOR 16 LINES.
     006756                   45$:
     006756 112720 000224          MOVB    #224,(R0)+              ; DEPOSIT 1ST EXPECTED DATA VALUE INTO RECEIVER BIT
                                                                   ; MAP FOR THIS UNIT.
     006762 005301                 DEC     R1                      ; NEXT LINE.
     006764 001374                 BNE     45$                     ; HAVE WE FINISHED 16 LINES YET?

     006766 012701 000170          MOV     #120.,R1                ; SETUP COUNT FOR 240 BYTE SIZE LOCATIONS.
     006772                   47$:
     006772 005020                 CLR     (R0)+                   ; CLEAR REMAINDER OF RECEIVER BIT MAP FOR THIS UNIT

     006774 005301                 DEC     R1                      ; HAVE WE FINISHED 240 LOCATIONS YET?
     006776 001375                 BNE     47$





                              ; LINE PARAMETER ROUTINE.


     007000 005000                 CLR     R0                      ; START WITH LINE ZERO.
     007002 012701 000001          MOV     #1,R1                   ; POINT TO IT.
     007006                   50$:
     007006 030167 171210          BIT     R1,LINESELECT           ; IS THIS LINE ON THE DHU11 SELECTED FOR TESTING?
     007012 001433                 BEQ     70$                     ; NO, DON'T SET IT UP.

     007014 010062 000000          MOV     R0,CSR(R2)              ; SELECT LINE NUMBER IN CSR.
                                                                   ; NOTE THE INITAL RESET SELECTED A
                                                                   ; XMIT AND RECEIVE BAUD OF 9600, NO PARITY,
                                                                   ; 1 STOP BIT, NORMAL OPERATION.  WE'RE
                                                                   ; NOT GOING TO CHANGE THAT HERE.

     007020 030167 170776          BIT     R1,SR3                  ; DOES THE OPERATOR WANT TO TEST IN LOCAL LOOPBACK?
```

```
1665 007024  001404                          BEQ     52$                     ;  YES, SET UP FOR LOCAL LOOPBACK.
1666
1667                                                                         ;  NO, SET-UP IN NORMAL OPERATION.
1668
1669 007026  012762  000026  000010          MOV     #RX.ENABLE!AUTO1!AUTO0,LNCTRL(R2)
1670                                                                         ;  SET RX ENABLE & NORMAL
1671                                                                         ;  OPERATION IN LNCTRL.
1672 007034  000403                          BR      54$
1673
1674 007036                           52$:
1675 007036  012762  000226  000010          MOV     #LOCAL.LOOP!RX.ENABLE!AUTO1!AUTO0,LNCTRL(R2)
1676                                                                         ;  SET RX ENABLE & LOCAL
1677                                                                         ;  LOOPBACK IN LNCTRL.
1678
1679 007044                           54$:
1680 007044  016762  173626  000012          MOV     PA22,TX1BUFADR(R2)      ;  PUT XMIT BUF ADDRESS IN TBUFFAD1.
1681                                                                         ;  IF EXTENEDED ADDRESS, MUST SET UP +14.
1682 007052  116762  173622  000014          MOVB    EA22,TX2BUFADR(R2)
1683 007060  012762  000154  000016          MOV     #108.,TXBUFCNTR(R2)     ;  SET UP BYTE COUNT.
1684
1685
1686
1687                                  ;  SETUP THE BAUD RATES.
1688
1689
1690 007066  016762  173624  000004          MOV     LPRSELECT,LPR(R2)       ;  SETUP THE BAUD RATES.
1691
1692
1693
1694                                  ;  TURN ON THE TRANSMITTER
1695                                  ;  NOTE THE INTERRUPTS ARE NOT ENABLED AT THIS POINT BUT THE AUTO FLOW
1696                                  ;  WILL STOP TX WHEN THE FIFO STARTS TO GET FULL SO DATA WILL NOT BE LOST
1697
1698 007074  052762  100200  000014          BIS     #TX.ENABLE!DMA.START,TX2BUFADR(R2)
1699                                                                         ;  ENABLE DMA START AND TXMIT ENABLE.
1700 007102                           70$:
1701 007102  005200                          INC     R0                      ;  UPDATE LINE NUMBER FOR CSR.
1702 007104  006301                          ASL     R1                      ;  POINT TO NEXT LINE.
1703 007106  001337                          BNE     50$                     ;  HAVE WE DONE 16 LINES?
1704
1705
1706
1707
```

```
                                ;) **
                                ;)
                                ;)   NAME:                          DHU.N2
                                ;)
                                ;)   FUNCTIONAL DESCRIPTION:       SET UP POINTERS FOR THE NEXT DHU11
                                ;)                                 AND INCREMENT DHUNOW
                                ;)                                 WHEN ALL BOARDS SET TURN ON INTERRUPT ENABLES
                                ;)
                                ;)   CALLING SEQUENCE:             IN LINE.
                                ;)
                                ;)   INPUTS REQUIRED:              "POINT' , "DHUCSR" , 'DHUVECT'
                                ;)
                                ;)   OUTPUTS GIVEN:                'POINT" , "DHUNOW" , "DHUCSR" , "DHUVECT"
                                ;)
                                ;)   REGISTERS USED,UNRESTORED:  R1 , 3 , 4
                                ;)
                                ;)   SUBROUTINES USED:            NONE.
                                ;)
                                ;)   RESTRICTIONS:                NONE.
                                ;)
                                ;) --

007110                          DHU.N2:

007110  062767  000020  171130          ADD     #20,DHUCSR              ;  NEXT DHU11 ADDRESS.
007116  062767  000010  173542          ADD     #10,DHUVECT             ;  NEXT VECTOR ADDRESS.
007124  005267  171114                  INC     DHUNOW                  ;  UPDATE FOR ERROR TYPECOUNT IF ANY.

007130  106367  171114                  ASLB    POINT                   ;  LOOK TO NEXT DHU11 BITMASK POSITION.
007134  032767  000020  171106          BIT     #20,POINT               ;  HAVE WE TESTED ALL DHU11'S.
007142  001653                          BEQ     DHU.L2                  ;  NO GO BACK AND TEST IF ANY OTHERS SELECTED

                                ;  YES ALL BOARDS SET SO TURN ON THE TRANSMITTER
                                ;  AND RECEIVER INTERRUPT ENABLES

007144  032767  000001  171070          BIT     #1,DHU.LIVE             ; CHECK FOR DHU11
007152  001405                          BEQ     10$                     ; IF NOT TRY THE NEXT ONE
007154  016702  173522                  MOV     ADDR0,R2                ; SET UP R2 FOR FIRST CSR
                                ;
007160  012762  040100  000000          MOV     #TXINTR.ENABLE!RCVINTR.ENABLE,CSR(R2)
                                                                        ; SET XMIT-INTERPT-ENABLE AND
                                                                        ; RECEIVE DATA INTERRUPT ENABLE.
007166                          10$:
007166  032767  000002  171046          BIT     #2,DHU.LIVE             ; CHECK FOR DHU11
007174  001405                          BEQ     20$                     ; IF NOT TRY THE NEXT ONE
007176  016702  173502                  MOV     ADDR1,R2                ; SET UP R2 FOR SECOUND CSR
                                ;
007202  012762  040100  000000          MOV     #TXINTR.ENABLE!RCVINTR.ENABLE,CSR(R2)
                                                                        ; SET XMIT-INTERPT-ENABLE AND
                                                                        ; RECEIVE DATA INTERRUPT ENABLE.
007210                          20$:
007210  032767  000004  171024          BIT     #4,DHU.LIVE             ; CHECK FOR DHU11
007216  001405                          BEQ     30$                     ; IF NOT TRY THE NEXT ONE
007220  016702  173462                  MOV     ADDR2,R2                ; SET UP R2 FOR THIRD CSR
                                ;
007224  012762  040100  000000          MOV     #TXINTR.ENABLE!RCVINTR.ENABLE,CSR(R2)
                                                                        ; SET XMIT-INTERPT-ENABLE AND
```

```
1766                                                    ; RECEIVE DATA INTERRUPT ENABLE.
1767 007232                            30$:
1768 007232  032767  000010  171002          BIT     #10,DHU.LIVE            ; CHECK FOR DHU11
1769 007240  001405                          BEQ     40$                     ; IF NOT TRY THE NEXT ONE
1770 007242  016702  173442                  MOV     ADDR3,R2                ; SET UP R2 FOR FORTH CSR
1771                                         ;
1772 007246  012762  040100  000000          MOV     #TXINTR.ENABLE!RCVINTR.ENABLE,CSR(R2)
1773                                                    ; SET XMIT INTERPT-ENABLE AND
1774                                                    ; RECEIVE DATA INTERRUPT ENABLE.
1775 007254                            40$:
1776                                         ;
1777                                         ; AND NOW WE WAIT FOR THOSE INTERRUPTS
1778                                         ;
1779 007254  104400  000000'                EXIT$,BEGIN                     ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
1780
1781
1782
1783
1784
1785                                                .SBTTL          DHU11 XMITTER INTERRUPT SERVICE ROUTINES.
```

```
1787                                    ;) ++
1788                                    ;)
1789                                    ;)   NAME:                        DHX0-X3,DHXSRV
1790                                    ;)
1791                                    ;)   FUNCTIONAL DESCRIPTION:      PROCESS THE INTERRUPTS OF XMITTERS.
1792                                    ;)                                CHECK FOR VALID INTERRUPT AND REPORT
1793                                    ;)                                ERROR AS REQUIRED.
1794                                    ;)
1795                                    ;)   CALLING SEQUENCE:            THESE ROUTINES ARE INTERRUPTED INTO.
1796                                    ;)
1797                                    ;)   INPUTS REQUIRED:             ADDRESS OF ROUTINES MUST BE SET UP.
1798                                    ;)                                "ADDR0" , 1 , 2 , 3 POINT TO DHU11 CSR S
1799                                    ;)
1800                                    ;)   OUTPUTS GIVEN:               NONE
1801                                    ;)
1802                                    ;)   REGISTERS USED,UNRESTORED:   R1 , 2  ALL RESTORED
1803                                    ;)
1804                                    ;)   SUBROUTINE ROUTINES USED:    NONE.
1805                                    ;)
1806                                    ;)   RESTRICTIONS:                NONE.
1807                                    ;)
1808                                    ;) --
1809
1810
1811                                    ;  ENTRANCE POINT FOR DHU11'S DURING TRANSMITTER INTERRUPTS.
1812
1813
1814 007260                             DHX0:
1815 007260  010246                            MOV     R2,-(SP)                ;  SAVE THE REGISTERS.
1816 007262  010146                            MOV     R1,-(SP)
1817 007264  012701  000766'                   MOV     #DHR0BUF+100,R1         ;  SAVE BUFFER AREA FOR ERROR REPORTS.
1818
1819 007270  016702  173406                    MOV     ADDR0,R2                ;  CSR ADDRESS FOR DHU11 #1.
1820 007274  000167  000060                    JMP     TXSERV                  ;  GOTO COMMON TX INTERRUPT ROUTINE.
1821
1822 007300                             DHX1:
1823 007300  010246                            MOV     R2,-(SP)                ;  SAVE THE REGISTERS.
1824 007302  010146                            MOV     R1,-(SP)
1825 007304  012701  001366'                   MOV     #DHR1BUF+100,R1         ;  SAVE BUFFER AREA FOR ERROR REPORTS.
1826
1827 007310  016702  173370                    MOV     ADDR1,R2                ;  CSR ADDRESS FOR DHU11 #2.
1828 007314  000167  000040                    JMP     TXSERV                  ;  GOTO COMMON TX INTERRUPT ROUTINE.
1829
1830 007320                             DHX2:
1831 007320  010246                            MOV     R2,-(SP)                ;  SAVE REGISTERS.
1832 007322  010146                            MOV     R1,-(SP)
1833 007324  012701  001766'                   MOV     #DHR2BUF+100,R1         ;  SAVE BUFFER AREA FOR ERROR REPORTS.
1834
1835 007330  016702  173352                    MOV     ADDR2,R2                ;  CSR ADDRESS FOR DHU11 #3.
1836 007334  000167  000020                    JMP     TXSERV                  ;  GOTO COMMON TX INTERRUPT ROUTINE.
1837
1838
1839
1840
1841 007340                             DHX3:
1842 007340  010246                            MOV     R2,-(SP)                ;  SAVE REGISTERS.
1843 007342  010146                            MOV     R1,-(SP)
```

```
   1844 007344  012701  002366'                MOV     #DHR3BUF+100,R1         ;  SAVE BUFFER AREA FOR ERROR REPORTS
   1845
   1846 007350  016702  173334                MOV     ADDR3,R2                ;  CSR ADDRESS FOR DHU11 #4.
   1847 007354  000167  000000                JMP     TXSERV                  ;  GOTO COMMON TX INTERRUPT ROUTINE.
   1848
   1849
   1850
   1851                                ;  COMMON TRANSMITTER SERVICE ROUTINE.
   1852
   1853
   1854 007360                         TXSERV:
   1855
   1856 007360  016267  000000  170670          MOV     CSR(R2),CDHUCSR         ;  GET CSR.
   1857 007366  100474                         BMI     100$                    ;  REPORT ERROR IF TRANSMITTER ACTION BIT (15) NOT S
ET.
   1858
   1859
   1860
   1861                                ;  ILLEGAL TRANSMIT INTERRUPT!!  ERROR!!
   1862
   1863
   1864                                                                        ;  REPORT "ILLEGAL TRANSMIT INTERRUPT ERROR."
   1865 007370  010221                         MOV     R2,(R1)+                ;  STORE CSR REPORT.
   1866 007372  016721  170660                MOV     CDHUCSR,(R1)+           ;  STORE CSR CONTENTS.
   1867 007376  016221  000002                MOV     RBUF(R2),(R1)+          ;  STORE RBUF CONTENTS.
   1868
   1869 007402  012762  000000  000000        MOV     #0,CSR(R2)              ;  DISABLE TRANSMIT &
   1870                                                                        ;  RECEIVER INTERRUPTS.
   1871
   1872 007410  012601                         MOV     (SP)+,R1                ;  RESTORE REGISTERS.
   1873 007412  012602                         MOV     (SP)+,R2
   1874                                        ;-----------------------------------------------------------------
        007414  000004  000000' 007422'        PIRQ$,BEGIN,10$         ; QUEUE UP TO CONTINUE AT 10$ AND RTI
                                                ;-----------------------------------------------------------------
   1875
   1876 007422                         10$:
   1877 007422  005767  170632                TST     LCKOUT                  ;  DO NOT TYPEOUT ERROR REPORT IF ANOTHER REPORT
   1878 007426  001405                         BEQ     15$                     ;  IS IN PROGRESS.
   1879 007430  104407  000000'                BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
        007434  104407  000000'                BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
   1880 007440  000770                         BR      10$
   1881 007442                         15$:
   1882 007442  005267  170612                INC     LCKOUT                  ;  DURING ERROR TYPEOUT, WE MUST NOT ALLOW ANY OTHER
   1883 007446  012704  000766'                MOV     #DHR0BUF+100,R4         ;  SECTION TO REPORT AN ERROR.
   1884 007452                         20$:
   1885 007452  005714                         TST     (R4)                    ;  DOES THIS LINE HAVE A SAVED ERROR?  (R4<>0).
   1886 007454  001003                         BNE     30$                     ;  YES, REPORT IT.
   1887 007456  062704  000400                ADD     #DHR1BUF-DHR0BUF,R4     ;  NO, THEN LOOK AT NEXT LINE FOR ERROR.
   1888 007462  000773                         BR      20$
   1889
   1890
   1891
   1892
   1893 007464                         30$:
   1894 007464  011467  170410                MOV     (R4),CSRA               ;  REPORT CSR ADDRESS.
   1895 007470  005024                         CLR     (R4)+                   ;  CLEAR ERROR INDICATION.
   1896
   1897 007472  012467  170404                MOV     (R4)+,ACSR              ;  REPORT CSR CONTENTS.
```

```
1898 007476  012467  170402                MOV     (R4)+,ASTAT             ;  REPORT RBUF CONTENTS.
1899 007502  012767  000025  170376         MOV     #25,ERRTYP              ;  REPORT ERROR TYPE.
1900
1901 007510  104401  000000' 003625'       MSG$,BEGIN,ERR.H        ;ASCII MESSAGE CALL
1902                                        ;*************************************************************
     007516  104405  000000' 000000        HRDER$,BEGIN,NULL       ;
                                            ;*************************************************************
1903
1904 007524  005067  170530                CLR     LCKOUT                  ;  ALLOW ANOTHER DHU11 TO RUN THIS DIAGNOSTIC.
1905 007530  005267  170502                INC     DHUSDONE                ;  ANOTHER DHU11 IS DONE.
1906 007534  026767  170476  170476        CMP     DHUSDONE,DHUSLEFT       ;  ANY DHU11'S LEFT TO TEST?
1907 007542  001402                        BEQ     35$                     ;  YES,  GO DO THEM.
1908
1909 007544  104400  000000'               EXIT$,BEGIN                     ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
1910
1911 007550                        35$:
1912 007550  104413  000000'               ENDIT$,BEGIN            ;SIGNAL END OF ITERATION.
                                                                    ;MONITOR SHALL TEST END OF PASS
1913 007554  000167  174614                JMP     RESTRT                  ;  GO START TESTING ANOTHER DHU11.
1914
1915 007560                        100$:
1916 007560  012601                        MOV     (SP)+,R1                ;  RESTORE REGISTERS.
1917 007562  012602                        MOV     (SP)+,R2
1918 007564  000002                        RTI
1919
1920
1921
1922
1923                                        .SBTTL          DHU11 RECEIVER INTERRUPT SERVICE ROUTINES.
```

```
1925                                    ;) **
1926                                    ;)
1927                                    ;)   NAME:                          DHR0-R3,DHRSRV
1928                                    ;)
1929                                    ;)   FUNCTIONAL DESCRIPTION:        PROCESS THE INTERRUPTS OF RECEIVERS.
1930                                    ;)                                  CHECK DATA RECIEVED AND REPORT ANY ERRORS
1931                                    ;)                                  ON GETTING A LAST CHARACTER ( 377 )
1932                                    ;)                                  SET THE APPROPRIATE BIT IN LINESDONE
1933                                    ;)                                  TEST FOR ALL BOARDS FINISHED AND IF
1934                                    ;)                                  SO DROP TO NORMAL PRIORITY AND DECLARE
1935                                    ;)                                  AN END OF ITERATION. RETURN TO RESTART
1936                                    ;)                                  POINT.
1937                                    ;)
1938                                    ;)
1939                                    ;)   CALLING SEQUENCE:              THESE ROUTINES ARE INTERRUPTED INTO.
1940                                    ;)
1941                                    ;)   INPUTS REQUIRED:               ADDRESS OF ROUTINES MUST BE SET UP.
1942                                    ;)                                  "ADDR0" , 1 , 2 AND 3 CONTAIN CSR ADDRESSES
1943                                    ;)                                  "LINESELECT" CONTAINS BIT MAP OF SELECTED LINES
1944                                    ;)                                  "DHUSNOW" CONTAINS NUMBER OF DHU11'S SELECTED
1945                                    ;)
1946                                    ;)   OUTPUTS GIVEN:                 "LINESDONE" AND "DHUSDONE" ARE UPDATED
1947                                    ;)
1948                                    ;)   REGISTERS USED,UNRESTORED:     R1 , 2 , 3 , 4 ALL RESTORED
1949                                    ;)
1950                                    ;)   SUBROUTINE ROUTINES USED:      NONE.
1951                                    ;)
1952                                    ;)   RESTRICTIONS:                  NONE.
1953                                    ;)
1954                                    ;) --
1955
1956
1957                                    ;  ENTRANCE POINT FOR DHU11'S DURING RECEIVER INTERRUPTS.
1958
1959
1960 007566                             DHR0:
1961 007566  010146                            MOV     R1,-(SP)                ;  SAVE REGISTERS ON STACK.
1962 007570  010246                            MOV     R2,-(SP)
1963 007572  010346                            MOV     R3,-(SP)
1964 007574  010446                            MOV     R4,-(SP)
1965
1966 007576  012767  000000  170446            MOV     #0,OFFSET               ;  INDICATE THIS IS DEVICE #1.
1967 007604  012767  000666' 170442            MOV     #DHR0BUF,BUFADR         ;  GET BUFFER TO RECEIVE CHARACTERS.
1968 007612  016702  173064                    MOV     ADDR0,R2                ;  CSR ADDRESS FOR DHU11 #1.
1969 007616  000167  000124                    JMP     RXSERV                  ;  GOTO COMMON RECEIVER INTERRUPT SERVICE ROUTINE.
1970
1971 007622                             DHR1:
1972 007622  010146                            MOV     R1,-(SP)                ;  SAVE REGISTERS ON STACK.
1973 007624  010246                            MOV     R2,-(SP)
1974 007626  010346                            MOV     R3,-(SP)
1975 007630  010446                            MOV     R4,-(SP)
1976
1977 007632  012767  000002  170412            MOV     #2,OFFSET               ;  INDICATE THIS IS DEVICE #2.
1978 007640  012767  001266' 170406            MOV     #DHR1BUF,BUFADR         ;  GET BUFFER TO RECEIVE CHARACTERS.
1979 007646  016702  173032                    MOV     ADDR1,R2                ;  CSR ADDRESS FOR DHU11 #2.
1980 007652  000167  000070                    JMP     RXSERV                  ;  GOTO COMMON RECEIVER INTERRUPT SERVICE ROUTINE.
1981
```

DHU11 RECEIVER INTERRUPT SERVICE ROUTINES.

```
                                        DHR2:
007656  010146                                MOV     R1,-(SP)                ;  SAVE REGISTERS ON STACK.
007660  010246                                MOV     R2,-(SP)
007662  010346                                MOV     R3,-(SP)
007664  010446                                MOV     R4,-(SP)

007666  012767  000004  170356                MOV     #4,OFFSET               ;  INDICATE THIS IS DEVICE #3.
007674  012767  001666' 170352                MOV     #DHR2BUF,BUFADR         ;  GET BUFFER TO RECEIVE CHARACTERS.
007702  016702  173000                        MOV     ADDR2,R2                ;  CSR ADDRESS FOR DHU11 #3.
007706  000167  000034                        JMP     RXSERV                  ;  GOTO COMMON RECEIVER INTERRUPT SERVICE ROUTINE.

007712                                  DHR3:
007712  010146                                MOV     R1,-(SP)                ;  SAVE REGISTERS ON STACK.
007714  010246                                MOV     R2,-(SP)
007716  010346                                MOV     R3,-(SP)
007720  010446                                MOV     R4,-(SP)

007722  012767  000006  170322                MOV     #6,OFFSET               ;  INDICATE THIS IS DEVICE #4.
007730  012767  002266' 170316                MOV     #DHR3BUF,BUFADR         ;  GET BUFFER TO RECEIVE CHARACTERS.
007736  016702  172746                        MOV     ADDR3,R2                ;  CSR ADDRESS FOR DHU11 #4.
007742  000167  000000                        JMP     RXSERV                  ;  GOTO COMMON RECEIVER INTERRUPT SERVICE ROUTINE.


                                        ;  COMMON RECEIVER INTERRUPT SERVICE ROUTINE.


007746                                  RXSERV:

007746                                  20$:
007746  016201  000002                        MOV     RBUF(R2),R1             ;  PULL OUT RECEIVED CHARACTER.
007752  100402                                BMI     25$                     ;  ARE THERE ANY CHARACTERS?
007754  000167  000576                        JMP     100$                    ;  THERE ARE NO MORE CHARACTERS TO PROCESS.
007760                                  25$:
007760  032701  070000                        BIT     #OVERRUN.ERROR!FRAMING.ERROR!PARITY.ERROR,R1
                                                                                ;  YES.  ARE THERE
                                                                                ;  ANY ERRORS?
007764  001510                                BEQ     40$                     ;  NO.

007766  010103                                MOV     R1,R3                   ;  THERE COULD BE AN ERROR.
007770  062703  010000                        ADD     #BIT12,R3               ;  ARE BITS 12,13, & 14 SET?  WE HOPE SO.
007774  103764                                BCS     20$                     ;  YES THEY ARE.  NO PROBLEM.


                                        ;  ERROR!!  OVERRUN,PARITY, OR FRAMING ERROR!!


007776  016704  170252                        MOV     BUFADR,R4               ;  GET BUFFER TO STORE ERROR REPORT
010002  062704  000040                        ADD     #40,R4
```

```
2039 010006  010224                          MOV     R2,(R4)+                ;  STORE THE CSR.
2040 010010  016224  000000                  MOV     CSR(R2),(R4)+           ;  STORE THE CSR CONTENTS.
2041 010014  010124                          MOV     R1,(R4)+                ;  STORE THE RECEIVE BUFFER CONTENTS.
2042
2043 010016  012762  000000  000000          MOV     #0,CSR(R2)              ;  DISABLE TX & RECEIVE
2044                                                                         ;  INTERRUPTS.
2045
2046 010024  012604                          MOV     (SP)+,R4                ;  RESTORE THE REGISTERS.
2047 010026  012603                          MOV     (SP)+,R3
2048 010030  012602                          MOV     (SP)+,R2
2049 010032  012601                          MOV     (SP)+,R1
2050
2051                                         ;-------------------- ---- ----------------------------------------
     010034  000004  000000' 010042'         PIRQ$,BEGIN,26$         ; QUEUE UP TO CONTINUE AT 26$ AND RTI
                                             ;--------------------- - ----------------------------------------
2052
2053 010042                          26$:
2054 010042  005767  170212                  TST     LCKOUT                  ;  DO NOT TYPEOUT ERROR REPORT IF ANOTHER ERROR
2055 010046  001405                          BEQ     27$                     ;  REPORT IS IN PROGRESS.
2056 010050  104407  000000'                BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
     010054  104407  000000'                BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
2057 010060  000770                          BR      26$
2058 010062                          27$:
2059 010062  005267  170172                  INC     LCKOUT                  ;  DURING ERROR TYPEOUT, WE MUST NOT ALLOW ANY OTHER
2060 010066  012704  000726'                MOV     #DHR0BUF+40,R4          ;  SECTION TO REPORT AN ERROR.
2061 010072                          28$:
2062 010072  005714                          TST     (R4)                    ;  DOES THIS LINE HAVE A SAVED ERROR?  (R1<>0).
2063 010074  001003                          BNE     29$                     ;  YES, REPORT IT.
2064 010076  062704  000400                  ADD     #DHR1BUF-DHR0BUF,R4     ;  NO, THEN LOOK AT NEXT LINE FOR ERROR.
2065 010102  000773                          BR      28$
2066
2067
2068 010104                          29$:
2069 010104  011467  167770                  MOV     (R4),CSRA               ;  REPORT THE CSR ADDRESS.
2070 010110  005024                          CLR     (R4)+                   ;  CLEAR ERROR INDICATION.
2071
2072 010112  012467  167764                  MOV     (R4)+,ACSR              ;  REPORT THE CSR CONTENTS.
2073 010116  012467  167762                  MOV     (R4)+,ASTAT             ;  REPORT THE CONTENTS OF RBUF.
2074
2075 010122  012767  000034  167756          MOV     #34,ERRTYP              ;  REPORT ERROR TYPE.
2076 010130  104401  000000' 003675'         MSG$,BEGIN,ERR.I        ;ASCII MESSAGE CALL
2077 010136                          30$:                                    ;  ERROR OCCURRED.
2078                                         ;*****************************************************************
     010136  104405  000000' 000000          HRDER$,BEGIN,NULL       ;
                                             ;*****************************************************************
2079
2080 010144                          32$:
2081 010144  005067  170110                  CLR     LCKOUT
2082 010150  005267  170062                  INC     DHUSDONE                ;  THIS DHU11 IS DONE!
2083 010154  026767  170056  170056          CMP     DHUSDONE,DHUSLEFT       ;  ANY MORE DHU11'S TO DO?
2084 010162  001405                          BEQ     36$
2085
2086 010164                          33$:
2087 010164  104400  000000'                EXIT$,BEGIN                     ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
2088
2089
2090
```

```
     010170                        35$:
                                             ;-----------------------------------------------------------
     010170  000004  000000' 010176'         PIRQ$,BEGIN,36$          ; QUEUE UP TO CONTINUE AT 36$ AND RTI
                                             ;-----------------------------------------------------------
     010176                        36$:
     010176  104413  000000'                 ENDIT$,BEGIN             ;SIGNAL END OF ITERATION.
                                                                      ;MONITOR SHALL TEST END OF PASS

     010202  000167  174166                  JMP     RESTRT

     010206                        40$:
     010206  010103                          MOV     R1,R3                    ;  GET LINE #.
     010210  000303                          SWAB    R3                       ;  PLACE LINE # IN 1ST FOUR BITS.
     010212  042703  177760                  BIC     #^C<17>,R3               ;  CLEAR OUT EVERYTHING BUT THE LINE #.

     010216  066703  170032                  ADD     BUFADR,R3                ;  SET-UP LOCATION FOR LINE #.

     010222                        60$:
     010222  042701  177400                  BIC     #^C<377>,R1              ;  CLEAR OUT EVERYTHING BUT THE CHARACTER RECEIVED.
     010226  120113                          CMPB    R1,(R3)                  ;  DOES THE EXPECTED (R3) MATCH THE RECEIVED R1.
     010230  001504                          BEQ     70$                      ;  BRANCH IF EQUAL.  THIS IS WHAT WE WANT.

     010232  020127  000021                  CMP     R1,#21                   ;  XON?  IF SO, WE MUST IGNORE.
     010236  001643                          BEQ     20$

     010240  020127  000023                  CMP     R1,#23                   ;  XOFF?  IF SO, WE MUST IGNORE.  IF NOT, REPORT
     010244  001640                          BEQ     20$                      ;  A DATA ERROR.  IF CHARACTER DOES NOT MATCH
                                                                              ;  EXPECTED, IT MUST BE EITHER XOFF OR XON.



                                   ;  ERROR!!  DATA ERROR!!


     010246                        62$:
     010246  016704  170002                  MOV     BUFADR,R4
     010252  062704  000020                  ADD     #20,R4

     010256  010124                          MOV     R1,(R4)+                 ;  ACTUAL DATA.
     010260  011324                          MOV     (R3),(R4)+               ;  EXPECTED DATA.
     010262  010224                          MOV     R2,(R4)+                 ;  CSR ADDRESS.
     010264  010314                          MOV     R3,(R4)                  ;  LINE # + BUFADR.
     010266  166724  167762                  SUB     BUFADR,(R4)+             ;  LINE #.

     010272  012762  000000  000000          MOV     #0,CSR(R2)               ;  DISABLE TX & RECEIVE
                                                                              ;  INTERRUPTS.

     010300  012604                          MOV     (SP)+,R4                 ;  RESTORE REGISTERS.
     010302  012603                          MOV     (SP)+,R3
     010304  012602                          MOV     (SP)+,R2
     010306  012601                          MOV     (SP)+,R1

                                             ;-----------------------------------------------------------
     010310  000004  000000' 010316'         PIRQ$,BEGIN,66$          ; QUEUE UP TO CONTINUE AT 66$ AND RTI
                                             ;-----------------------------------------------------------
```

```
2143 010316                                66$:
2144 010316  005767  167736                       TST     LCKOUT                  ;  DON'T TYPEOUT ERROR REPORT IF ANOTHER ERROR
2145 010322  001405                               BEQ     67$                     ;  REPORT IS IN PROGRESS.
2146 010324  104407  000000'                      BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
     010330  104407  000000'                      BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
2147 010334  000770                               BR      66$
2148
2149
2150
2151
2152 010336                                67$:
2153 010336  005267  167716                       INC     LCKOUT                  ;  DURING ERROR TYPEOUT, WE MUST NOT ALLOW ANY OTHER
2154 010342  012701  000706'                      MOV     @DHR0BUF+20,R1          ;  SECTION TO REPORT AN ERROR.
2155
2156 010346                                68$:
2157 010346  005711                               TST     (R1)                    ;  DOES THIS LINE HAVE A SAVED ERROR?  (R1<>0)
2158 010350  001003                               BNE     69$                     ;  YES, REPORT IT.
2159 010352  062701  000400                       ADD     @DHR1BUF DHR0BUF,R1     ;  NO, THEN LOOK AT NEXT LINE FOR ERROR.
2160 010356  000773                               BR      68$
2161
2162
2163 010360                                69$:
2164 010360  011167  167524                       MOV     (R1),AWAS               ;  REPORT ACTUAL DATA.
2165 010364  005021                               CLR     (P1)+                   ;  CLEAR ERROR INDICATION.
2166
2167 010366  112167  167514                       MOVB    (R1)+,ASB               ;  REPORT EXPECTED DATA.
2168 010372  105021                               CLRB    (R1)+
2169 010374  012167  167500                       MOV     (R1)+,CSRA              ;  REPORT CSR ADDRESS.
2170 010400  012167  167476                       MOV     (R1)+,SBADR             ;  REPORT LINE #.
2171
2172 010404  104401  000000' 003753'              MSG$,BEGIN,ERR.J        ;ASCII MESSAGE CALL
2173
2174                                              ;*****************************************************************
     010412  104404  000000'                      DATER$,BEGIN            ;DATA ERROR!!!
                                                  ;*****************************************************************
2175
2176 010416  005067  167636                       CLR     LCKOUT                  ;  DONE REPORTING THIS ERROR?
2177 010422  005267  167610                       INC     DHUSDONE                ;  MARK THIS DHU11 AS BEING DONE.
2178 010426  026767  167604  167604               CMP     DHUSDONE,DHUSLEFT       ;  ARE ALL THE DHU11'S DONE?
2179 010434  001660                               BEQ     36$                     ;  YES, THEN END PASS.  ELSE NORMAL EXIT.
2180
2181 010436  104400  000000'                      EXIT$,BEGIN                     ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
2182
2183 010442                                70$:
2184 010442  105213                               INCB    (R3)                    ;  INCREMENT CHARACTER.
2185 010444  001402                               BEQ     75$                     ;  SEE IF DID LAST CHARACTER (377), BR IF NOT.
2186 010446  000167  177274                       JMP     20$                     ;  NOT THE LAST CHARACTER.  GO SEE IF ANOTHER
2187 010452                                75$:                                   ;  CHARACTER IS AVAILABLE.
2188 010452  166703  167576                       SUB     BUFADR,R3               ;  R3 = LINE# + BUFFER ADDRESS.  GET LINE#.
2189 010456  012704  000001                       MOV     #1,R4                   ;  MAKE BITMASK OF LINE NUMBER.
2190 010462                                80$:
2191 010462  005303                               DEC     R3                      ;  DECREMENT UNTIL 0.
2192 010464  100402                               BMI     90$
2193 010466  006304                               ASL     R4                      ;  SHIFT UNTIL # GOES TO ZERO FOR BITMASK.
2194 010470  000774                               BR      80$                     ;  BITMASK IN R4.
2195 010472                                90$:
2196 010472  016701  167554                       MOV     OFFSET,R1
```

```
2197 010476  050461  000224                BIS     R4,LINESDONE(R1)        ;  RECORD THIS LINE DONE.
2198
2199 010502  026761  167514  000224        CMP     LINESELECT,LINESDONE(R1)
2200                                                                         ;  SEE IF ALL LINES DONE ON THIS UNIT
2201 010510  001402                        BEQ     95$
2202 010512  000167  177230                JMP     20$                     ;  BR IF ALL LINES NOT DONE.
2203
2204
2205 010516                          95$:
2206 010516  012762  000000  000000        MOV     #0,CSR(R2)              ;  DISABLE TX & RECEIVE
2207                                                                         ;  INTERRUPTS.
2208
2209 010524  012604                        MOV     (SP)+,R4                ;  RESTORE REGISTERS.
2210 010526  012603                        MOV     (SP)+,R3
2211 010530  012602                        MOV     (SP)+,R2
2212 010532  012601                        MOV     (SP)+,R1
2213
2214 010534  005267  167476                INC     DMUSDONE                ;  RECORD THIS DMU11 AS BEING DONE.
2215 010540  026767  167472  167472        CMP     DMUSDONE,DMUSLEFT       ;  ARE ALL DMU11'S DONE?
2216 010546  001002                        BNE     97$                     ;  NO, EXIT.
2217 010550  000167  177414                JMP     35$                     ;  YES, REPORT END OF PASS.
2218 010554                          97$:
2219 010554  000002                        RTI                             ;  EXIT, JUST THIS DMU11 IS DONE.
2220
2221 010556                          100$:
2222 010556  012604                        MOV     (SP)+,R4                ;  RESTORE REGISTERS.
2223 010560  012603                        MOV     (SP)+,R3
2224 010562  012602                        MOV     (SP)+,R2
2225 010564  012601                        MOV     (SP)+,R1
2226 010566  000002                        RTI
2227
2228
2229
2230
```

```
                                        ;) ..
                                        ;)
                                        ;)   NAME:                          BAUDRATE
                                        ;)
                                        ;)   FUNCTIONAL DESCRIPTION:       SUBROUTINE TO SET UP THE BAUDRATE
                                        ;)                                  AND STORE IT IN TXSPEED AND RCVSPEED
                                        ;)
                                        ;)   CALLING SEQUENCE:             JSR PC,BAUDRATE
                                        ;)
                                        ;)   INPUTS REQUIRED:              "SR1" AND "2" IN MODULE HEADER AND "DMULPR"
                                        ;)
                                        ;)   OUTPUTS GIVEN:                "TXSPEED", "RCVSPEED" AND "LPRSELECT"
                                        ;)
                                        ;)   REGISTERS USED,UNRESTORED:    R0
                                        ;)
                                        ;)   SUBROUTINE ROUTINES USED:     NONE.
                                        ;)
                                        ;)   RESTRICTIONS:                 "DMULPR" IS SET UP WHEN DECLARED.
                                        ;)                                  IT SHOULD NOT BE CHANGED.
                                        ;) --
010570                                  BAUDRATE:

010570  016767  167222  167464          MOV     SR1,TXSPEED             ;  GET SELECTED TRANSMIT SPEED.
010576  016767  167214  167460          MOV     SR1,RCVSPEED            ;  GET SELECTED RECEIVE SPEED.

010604  012700  000004                  MOV     #4,R0                   ;  SET UP SHIFT COUNTER.
010610                                  10$:
010610  006367  167446                  ASL     TXSPEED                 ;  LINE-UP TRANSMIT SPEED SELECTOR W/LPR.
010614  077003                          SOB     R0,10$

010616  056767  167440  167440          BIS     TXSPEED,RCVSPEED        ;  PUT TRANSMIT & SPEED SELECTION TOGETHER.
010624  000367  167434                  SWAB    RCVSPEED

010630  016767  172060  172060          MOV     DMULPR,LPRSELECT        ;  INITIALIZE THE LINE PARAMETER SELECTION.
010636  056767  167422  172052          BIS     RCVSPEED,LPRSELECT      ;  SAVE THE TRANSMIT & RECEIVE SPEEDS.

010644  000207                          RTS     PC                      ;  RETURN TO MAIN PROGRAM.

        000001                          .END
```

DHUA DEC/X11 SYSTEM EXERCISER M MACRO M1200 12 DEC 83 09:01 PAGE 19 1 SEQ 55

SYMBOL TABLE

```
ACSR    000102R     DATCK$- 104411     ERR.C   003120R     OFFSET  000252R     SETUP1  004452R
ADDR    000006R     DATER$- 104404     ERR.D   003210R     OPEN  - 000000      SETUP2  004516R
ADDR0   002702R     DHR0    007566R    ERR.E   003311R     OTOA$ - 104420      SOFCNT  000042R
ADDR1   002704R     DHR0BU  000666R    ERR.F   003404R     OVERRU- 040000      SOFER$- 104406
ADDR2   002706R     DHR1    007622R    ERR.G   003511R     PA      002672R     SOFPAS  000046R
ADDR22- 001000      DHR1BU  001266R    ERR.H   003625R     PARITY- 010000      SPOINT  000032R
ADDR3   002710R     DHR2    007656R    ERR.I   003675R     PASCNT  000034R     SPSIZ - 000040
ASB     000106R     DHR2BU  001666R    ERR.J   003753R     PA22    002676R     SR1     000016R
ASTAT   000104R     DHR3    007712R    ERR.K   004003R     PIRQ$ - 000004      SR2     000020R
AUTO0 - 000020      DHR3BU  002266R    ERR.L   004042R     POINT   000250R     SR3     000022R
AUTO1 - 000002      DHUCSR  000246R    ERR.M   004150R     POPSP - 005726      SR4     000024R
AWAS    000110R     DHULPR  002714R    EX      005334R     POPSP2- 022626      START   004340R
BAUDRA  010570R     DHUNOW  000244R    EXIT$ - 104400      PRTY  - 000000      STAT    000026R
BEGIN   000000R     DHUSDO  000236R    FLAG    002712R     PRTY0 - 000000      SVR0    000062R
BIT0  - 000001      DHUSLE  000240R    FRAMIN- 020000      PRTY1 - 000040      SVR1    000064R
BIT1  - 000002      DHUT01  004550R    GETPA$- 104415      PRTY2 - 000100      SVR2    000066R
BIT10 - 002000      DHUT02  004730R    GWBUF$- 104414      PRTY3 - 000140      SVR3    000070R
BIT11 - 004000      DHUT03  005764R    HRDCNT  000044R     PRTY4 - 000200      SVR4    000072R
BIT12 - 010000      DHUT04  006610R    HRDER$- 104405      PRTY5 - 000240      SVR5    000074R
BIT13 - 020000      DHUVEC  002666R    HRDPAS  000050R     PRTY6 - 000300      SVR6    000076R
BIT14 - 040000      DHU.BU  006706R    ICONT   000036R     PRTY7 - 000340      SYSCNT  000052R
BIT15 - 100000      DHU.LI  000242R    ICOUNT  000040R     PS    - 177776      TRPDFD- 000022
BIT2  - 000004      DHU.L1  006000R    IDNUM   000122R     PSW   - 177776      TX      005274R
BIT3  - 000010      DHU.L2  006672R    INIT    000030R     PUSH  - 005746      TXBUF   000266R
BIT4  - 000020      DHU.N1  006550R    INTR    000120R     PUSH2 - 024646      TXBUFC- 000016
BIT5  - 000040      DHU.N2  007110R    LCKOUT  000260R     PWRFLG- 000002      TXCHAR- 000006
BIT6  - 000100      DHX0    007260R    LINECO  000234R     RAND$ - 104417      TXDATA- 000224
BIT7  - 000200      DHX1    007300R    LINESD  000224R     RANNUM  000054R     TXDMA.- 010000
BIT8  - 000400      DHX2    007320R    LINESE  000222R     RBUF  - 000002      TXINTR- 040000
BIT9  - 001000      DHX3    007340R    LNCTRL- 000010      RCV     005314R     TXSERV  007360R
BREAK$- 104407      DIAG.F- 020000     LOCAL.- 000200      RCVDAT- 000200      TXSPEE  000262R
BR1     000012R     DMA.ST- 000200     LPR   - 000004      RCVINT- 000100      TX.ACT- 100000
BR2     000013R     DROP    004350R    LPRSEL  002716R     RCVSPE  000264R     TX.ENA- 100000
BTOD$ - 104421      DROP1   004354R    MAP22$- 104416      RESTRT  004374R     TX1BUF- 000012
BUFADR  000254R     DVID1   000014R    MASTER- 000040      RES1    000056R     TX2BUF- 000014
CDATA$- 104412      EA      002674R    MODNAM  000000R     RES2    000060R     VA      002670R
CDHUCS  000256R     EA22    002700R    MODSP   000222R     RSTRT   000112R     VECT    005664R
CHECKL  005346R     ENDIT$- 104413     MSGN$ - 104403      RXSERV  007746R     VECTOR  000010R
CONFIG  000056R     END$  - 104410     MSGS$ - 104402      RX.ENA- 000004      WASADR  000104R
CSR   - 000000      ERRTYP  000106R    MSG$  - 104401      R6    =%000006      WDFR    000116R
CSRA    000100R     ERR.A   002720R    NEXTDH  005714R     R7    =%000007      WDTO    000114R
DATA.V- 100000      ERR.B   003020R    NULL  - 000000      SBADR   000102R     XFLAG   000005R

. ABS.  000000      000
        010646      001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  12513 WORDS  ( 49 PAGES)
DYNAMIC MEMORY:  20060 WORDS  ( 77 PAGES)
ELAPSED TIME:  00:00:31
CXDHUA0.OBJ,CXDHUA0.LST/-SP=DDXCOM.MAC,CXDHUA0.P11
```