.REM   _

IDENTIFICATION

PRODUCT CODE: AC-E956B-MC

PRODUCT NAME: CXVTBB0 DH11/VT20 MODULE

PRODUCT DATE: SEPTEMBER 1978

MAINTAINER: DEC/X11 SUPPORT GROUP

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

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED TO THE
PURCHASER UNDER A LICENSE FOR USE ON A SINGLE COMPUTER
SYSTEM AND CAN BE COPIED (WITH INCLUSION OF DIGITALS
COPYRIGHT NOTICE) ONLY FOR USE IN SUCH SYSTEM, EXCEPT AS MAY
OTHERWISE BE PROVIDED IN WRITING BY DIGITAL.

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

1.   ABSTRACT

        VTB IS AN IOMODX THAT EXERCISES UP TO FOUR VT20'S (DH11 LINES).
        IT IS INTENDED TO BE A DATA HANDLING ROUTINE USED IN CONJUNCTION
        WITH TEST 21 OF MAINDEC-11-DBVTA (PREVIOUSLY LOADED AND RUNNING
        IN THE VT20'S PDP11-05). DATA IS ENTERED AT EACH SELECTED TUBE
        AND SET INTO THE CONTINUOUS TRANSMIT MODE. THIS DATA IS THEN
        RECEIVED AND RETRANSMITTED BY THE VT20 HOST COMPUTER (THIS
        DEC/X11 MODULE). ALL LINES SELECTED FOR TEST CAN BE ACTIVATED AND
        RUN CONCURRENTLY. ALL TRANSMIT AND RECEIVE ERRORS ARE REPORTED
        ON THE CONSOLE TTY. NO DATA ERRORS ARE REPORTED BY THIS MODULE.

2.   REQUIREMENTS

        HARDWARE: AT LEAST ONE VT20 CONNECTED VIA A DH11

        STORAGE:: VTB REQUIRES:
                           1. DECIMAL WORDS: 4002
                           2. OCTAL WORDS: 07642
                           3. OCTAL BYTES: 17504

3.   PASS DEFINITION

        ONE PASS OF THE VTAB MODULE CONSISTS OF CONTINUOUSLY RECEIVING
        AND TRANSMITTING THE DATA ENTERED ON ALL SELECTED LINES FOR
        THE PERIOD DEFINED BELOW.

4.   EXECUTION TIME

        EXECUTION TIME VARIES WITH THE NUMBER OF JOBS (MODULES)
        ACTIVE, THE BAUD RATE AND THE NUMBER OF TUBES BEING
        EXERCISED. HOWEVER, THIS MODULE RUNNING ALONE WILL
        TAKE NO MORE THAN 3 MINUTES WITH 16 TUBES AT 110 BAUD

5.   CONFIGURATION PARAMETERS

        DEFAULT PARAMETERS:

                DVA:160020, VCT:350, BR1:5, BR2:0, DVC:1

                L0BR-L17BR:IF ANY OF THE DH11 LINES IS NOT A 9600 BAUD LINE
                THE WORD ASSOCIATED WITH THAT LINE MUST BE MODIFIED
                BEFORE RUNNING

        REQUIRED PARAMETERS:

                DVC: NO OF VT20'S IF GREATER THAN 1

```
                        DVID1:  1 BIT SET FOR EACH DH11 LINE. ITS POSITION SHOULD
                                CORROSPOND WITH THE LINE #. E.G. IF DH11 LINE 6
                                IS USED "DVID1" BIT 6 SHOULD BE SET.
```

6.   DEVICE SETUP

A. THE USER MUST LOAD AND START TEST 21 OF MAINDEC-11-DBVTA IN THE VT20
    PDP11/05 IN ORDER FOR THIS MODULE TO EXERCISE. CONSULT THE ABOVE
    DOCUMENT AND COMPLY WITH THE OPERATING INSTRUCTIONS FOR TEST 21
    (SECTION 26). THIS DEC/X11 MODULE EXPECTS THE USER TO ENTER DATA
    ON EACH SELECTED TUBE AND SET EACH TUBE IN THE CONTINUOUS
    TRANSMIT MODE. THIS STEP IS TAKEN AFTER THE DEC/X11 EXERCISER
    HAS BEEN STARTED BY THE "RUN" COMMAND. TYPICAL USER ACTION
    ON EACH SELECTED TUBE WILL BE AS FOLLOWS:

    KEY   FUNCTION
    ---   --------

    CTRL E   CLEAR SCREEN
    CTRL W   GENERATE WORST CASE CHARACTER PATTERN ON TOP OF SCREEN
    CTRL T   CONTINUOUS TRANSMIT TO DEC/X11 MODULE (DEC/X11 MODULE
                    WILL RECEIVE DATA AND TRANSMIT IT BACK TO BOTTOM OF SCREEN)

    NOTE: IF THE CHARACTER PATTERN FAILS TO RETURN ON THE
            BOTTOM OF THE SCREEN AFTER ONE "CTRL T", THEN RETRY AFTER
            "END PASS" IS REPORTED FOR THIS DEC/X11 MODULE (DL11 RECEIVERS
            ARE TURNED OFF SECONDS BEFORE "END PASS" MSG). IF DATA IS STILL
            NOT RETURNED FROM HOST COMPUTER (DEC/X11 SYSTEM) THEN VERIFY
            THE VT20 HOST COMPUTER BY RUNNING MAINDEC-11-DZVTE.

B. IF LINES WITH BAUD RATES OTHER THAT 9600 ARE TO BE USED, THEN THE
    VALUE OF THE CORRESPONDING WORD IN THE BAUD RATE TABLE (16
    WORDS STARTING AT LOC "LOBR") MUST BE MODIFIED REFER TO THE
    PDP-11 PERIPHERALS AND INTERFACING HANDBOOK FOR THE EXACT
    VALUES NEEDED

8.   OPERATOR OPTIONS

A. THE USER CAN MODIFY (VTA 14) "DVID1" TO SELECT OR
    DESELECT INDIVIDUAL VT20'S. THIS MODULE IS QUITE
    ABLE TO HANDLE VT20'S THAT DO NOT HAPPEN TO HAVE ADJACENT
    DH11 LINES.

B. THE USER CAN USE THE "MOD" COMMAND TO DUMP THE TABLES
    OR BUFFERS DESCRIBED IN 7.2 TO OBTAIN MORE DETAILED
    ERROR INFORMATION.

9.   ERROR PRINTOUTS

9.1   ERROR FORMAT - RECEIVE

CSRA = CSR ADDRESS
CSRC = NRC WORD AS FOLLOWS:

THE # PRINTED OUT LABELED AS `STATC` IS THE NEXT RECIEVED CHARACTER
BIT 15   = DATA PRESENT
BIT 14   = OVERRUN
BIT 13   = FRAMING
BIT 12   = PARITY
BIT 11-8 = LINE #
BIT 7-0 = DATA RECEIVED

WITH SOME ERRORS SUCH AS "NO DH11 LINES REMAIN SELECTED"
THE CONTENTS OF THE DH11 REGISTERS ARE IRRELEVENT.
IN SUCH CASES THEY ARE PRINTED ANYWAYS.

9.2   ERROR FORMAT - TRANSMIT

CSRA = CSR ADDRESS
CSRC = CSR CONTENTS AS FOLLOWS:

BIT 7 = XMITR READY
BIT 6 = XMITR INTERRUPT ENABLED

188   .LIST   SEQ,BIN,TOC

```
 189                                          ;SOME DEFINITIONS
 190  000000'                          IOMODX  <VTBB >,160020,350,5,,,1,126    ;BR LEVEL 5, UP TO 16 DEVICES
 191  000000'                          MODULE  150000,VTBB ,160020,350,5,,,1,126
 192                                    .TITLE  VTBB DEC/X11 SYSTEM EXERCISER MODULE
 193                                  , DDXCOM  VERSION  6    23-MAY-78
 194                                    .LIST   BIN
 195                                  ;***************************************************************
 196                          BEGIN:
 197  000000' 052126  041102   040    MODNAM: .ASCII  /VTBB / ;MODULE NAME.
 198  000005'    000            XFLAG:  .BYTE   OPEN     ;USED TO KEEP TRACK OF WBUFF USAGE
 199  000006' 160020            ADDR:   160020+0         ;1ST DEVICE ADDR.
 200  000010' 000350            VECTOR: 350+0            ;1ST DEVICE VECTOR.
 201  000012'    240            BR1:    .BYTE   PRTY5+0  ;1ST BR LEVEL.
 202  000013'    000            BR2:    .BYTE   PRTY+0   ;2ND BR LEVEL.
 203  000014' 000001            DVID1:  +1               ;DEVICE INDICATOR 1.
 204  000016' 000000            SR1:    OPEN             ;SWITCH REGISTER 1
 205  000020' 000000            SR2:    OPEN             ;SWITCH REGISTER 2
 206  000022' 000000            SR3:    OPEN             ;SWITCH REGISTER 3
 207  000024' 000000            SR4:    OPEN             ;SWITCH REGISTER 4
 208                                  ;***************************************************************
 209  000026' 150000            STAT:   150000           ;STATUS WORD.
 210  000030' 000554'           INIT:   START            ;MODULE START ADDR.
 211  000032' 000224'           SPOINT: MODSP            ;MODULE STACK POINTER.
 212  000034' 000000            PASCNT: 0                ;PASS COUNTER.
 213  000036' 000001            ICONT:  1                ;# OF ITERATIONS PER PASS=1
 214  000040' 000000            ICOUNT: 0                ;LOC TO COUNT ITERATIONS
 215  000042' 000000            SOFCNT: 0                ;LOC TO SAVE TOTAL SOFT ERRORS
 216  000044' 000000            HRDCNT: 0                ;LOC TO SAVE TOTAL HARD ERRORS
 217  000046' 000000            SOFPAS: 0                ;LOC TO SAVE SOFT ERRORS PER PASS
 218  000050' 000000            HRDPAS: 0                ;LOC TO SAVE HARD ERRORS PER PASS
 219  000052' 000000            SYSCNT: 0                ;# OF SYS ERRORS ACCUMULATED
 220  000054' 000000            RANNUM: 0                ;HOLDS RANDOM # WHEN RAND MACRO IS CALLED
 221  000056'                   CONFIG:                  ;RESERVED FOR MONITOR USE
 222  000056' 000000            RES1:   0                ;RESERVED FOR MONITOR USE
 223  000060' 000000            RES2:   0                ;RESERVED FOR MONITOR USE
 224  000062' 000000            SVR0:   OPEN             ;LOC TO SAVE R0.
 225  000064' 000000            SVR1:   OPEN             ;LOC TO SAVE R1.
 226  000066' 000000            SVR2:   OPEN             ;LOC TO SAVE R2.
 227  000070' 000000            SVR3:   OPEN             ;LOC TO SAVE R3.
 228  000072' 000000            SVR4:   OPEN             ;LOC TO SAVE R4.
 229  000074' 000000            SVR5:   OPEN             ;LOC TO SAVE R5.
 230  000076' 000000            SVR6:   OPEN             ;LOC TO SAVE R6.
 231  000100' 000000            CSRA:   OPEN             ;ADDR OF CURRENT CSR.
 232  000102'                   SBADR:                   ;ADDR OF GOOD DATA, OR
 233  000102' 000000            ACSR:   OPEN             ;CONTENTS OF CSR.
 234  000104'                   WASADR:                  ;ADDR OF BAD DATA, OR
 235  000104' 000000            ASTAT:  OPEN             ;STATUS REG CONTENTS.
 236                            ERRTYP:                  ;TYPE OF ERROR
 237  000106' 000000            ASB:    OPEN             ;EXPECTED DATA.
 238  000110' 000000            AWAS:   OPEN             ;ACTUAL DATA.
 239  000112' 000652'           RSTRT:  RESTRT           ;RESTART ADDRESS AFTER END OF PASS
 240  000114' 000000            WDTO:   OPEN             ;WORDS TO MEMORY PER ITERATION
 241  000116' 000000            WDFR:   OPEN             ;WORDS FROM MEMORY PER ITERATION
 242  000120' 000000            INTR:   OPEN             ;# OF INTERRUPTS PER ITERATION
 243  000122' 000126            IDNUM:  126              ;MODULE IDENTIFICATION NUMBER=126
 244       000040                        .REPT   SPSIZ   ;MODULE STACK STARTS HERE.
```

```
 245                                    .NLIST
 246                                    .WORD   0
 247                                    .LIST
 248                                    .ENDR
 249  000224'                   MODSP:
 250                                  ;***************************************************************
 251                                  ;SOME POINTERS AND VARIABLES AND CONSTANTS UNIQUE TO THIS MODULE
 252  000224' 000000            SELCNT: 0                ;PROGRAM SETS THIS LOC TO = # OF VT20'S SELECTED
 253  000226' 000000            SAVR0:  000000           ;PLACE TO STUFF R0 WHILE SERVICING INTERRUPTS
 254  000230' 000000            WAITIM: 000000           ;USED AS A COUNTER FOR THE INTERRUPT WAIT LOOP
 255  000232' 000000            RELOC:  000000           ;A RELOCATION COUNT FOR INDEX REGISTERS IF NEEDED
 256  000234' 000000            SVT0:   000000
 257  000236' 000000            SVT1:   000000
 258  000240' 000000            XCNT:   000000           ;COUNT FOR # OF XFERS TO DO EACH PASS
 259  000242' 000000            VIRTAD: 000000           ;VIRTUAL ADDRESS
 260  000244' 000000            PHYSAD: 000000           ;PHYSICAL ADDRESS
 261  000246' 000000            EXTAD:  000000           ;EXTENDED ADDRESS BITS
 262  000250' 000000            DUNCNT: 000000           ;# OF DEVICES FINISHED
 263  000252' 000000            WAITIM: 000000           ;TIMER COUNT
 264  000254' 000000            TEMP:   000000           ;TEMPORARY LOCATION
 265  000256' 000000            TEMP0:  0                ;GENERAL PURPOSE TEMPORARY STORAGE
 266  000260' 000000            TEMP1:  0                ;GENERAL PURPOSE TEMPORARY STORAGE
 267  000262' 000000            COUNT1: 000000           ;SET BY BY USER AT CONFIGURATION TIME
 268                                                     ;TO A 3 IF LINES OF 1200 BAUD OR LESS ARE USED
 269  000264' 000000            INTSW:  000000           ;SWITCH=1 IF WE SHOULD RETURN FROM INTERRUPT
 270                                                     ;SERVICE USING AN "RTI" INSTRUCTION
 271  000266' 000000            FATERR: 000000           ;SWITCH=1 TO SIGNAL A FATAL ERROR.
 272  000270' 000000            ERPT1:  000000           ;POINTS TO LAST WORD SERVICED FROM ERROR QUEUE
 273  000272' 000000            ERPT2:  000000           ;POINTS TO LAST WORD ENTERED IN ERROR QUEUE
 274
 275  000274' 002542'           BPLIST: VTBF0            ;POINTS TO ONE OF THE 16 BUFFERS
 276  000276' 003350'                   VTBF1            ;POINTS TO 1 OF THE 16 BUFFERS
 277  000300' 004156'                   VTBF2            ;POINTS TO 1 OF THE 16 BUFFERS
 278  000302' 004764'                   VTBF3            ;POINTS TO ONE OF THE 16 BUFFERS
 279  000304' 005572'                   VTBF4            ;POINTS TO ONE OF THE 16 BUFFERS
 280  000306' 006400'                   VTBF5            ;POINTS TO ONE OF THE 16 BUFFERS
 281  000310' 007206'                   VTBF6            ;POINTS TO ONE OF THE 16 BUFFERS
 282  000312' 010014'                   VTBF7            ;POINTS TO ONE OF THE 16 BUFFERS
 283  000314' 010622'                   VTBF10           ;POINTS TO ONE OF THE 16 BUFFERS
 284  000316' 011430'                   VTBF11           ;POINTS TO ONE OF THE 16 BUFFERS
 285  000320' 012236'                   VTBF12           ;POINTS TO ONE OF THE 16 BUFFERS
 286  000322' 013044'                   VTBF13           ;POINTS TO ONE OF THE 16 BUFFERS
 287  000324' 013652'                   VTBF14           ;POINTS TO ONE OF THE 16 BUFFERS
 288  000326' 014460'                   VTBF15           ;POINTS TO ONE OF THE 16 BUFFERS
 289  000330' 015266'                   VTBF16           ;POINTS TO ONE OF THE 16 BUFFERS
 290  000332' 016074'                   VTBF17           ;POINTS TO ONE OF THE 16 BUFFERS
 291                                  ;EACH OF THE NEXT 8 WORDS IS SETUP BY THIS MODULE TO POINT TO A DH11 REGISTER
 292  000334' 000000            X00:    0                ;POINTS TO THE SYSTEM CONTROL REGISTER
 293  000336' 000000            X02:    0                ;POINTS TO THE NEXT CHARACTER RECEIVE REGISTER
 294  000340' 000000            X04:    0                ;POINTS TO THE LINE PARAMETER REGISTER
 295  000342' 000000            X06:    0                ;POINTS TO THE CURRENT ADDRESS REGISTER
 296  000344' 000000            X10:    0                ;POINTS TO THE BYTE COUNT REGISTER
 297  000346' 000000            X12:    0                ;POINTS TO THE BUFFER ACTIVE REGISTER
 298  000350' 000000            X14:    0                ;POINTS TO THE BREAK CONTROL REGISTER
 299  000352' 000000            X16:    0                ;POINTS TO THE SILO STATUS REGISTER
 300
```

```
301                          ;TABLE OF BUFFER POINTERS FOR DH11 LINES
302   000354'  000000       L0SW:   0                          ;BUFFER POINTER FOR DH11 LINE 0
303   000356'  000000       L1SW:   0                          ;BUFFER POINTER FOR DH11 LINE 1
304   000360'  000000       L2SW:   0                          ;BUFFER POINTER FOR DH11 LINE 2
305   000362'  000000       L3SW:   0                          ;BUFFER POINTER FOR DH11 LINE 3
306   000364'  000000       L4SW:   0                          ;BUFFER POINTER FOR DH11 LINE 4
307   000366'  000000       L5SW:   0                          ;BUFFER POINTER FOR DH11 LINE 5
308   000370'  000000       L6SW:   0                          ;BUFFER POINTER FOR DH11 LINE 6
309   000372'  000000       L7SW:   0                          ;BUFFER POINTER FOR DH11 LINE 7
310   000374'  000000       L10SW:  0                          ;BUFFER POINTER FOR DH11 LINE 10
311   000376'  000000       L11SW:  0                          ;BUFFER POINTER FOR DH11 LINE 11
312   000400'  000000       L12SW:  0                          ;BUFFER POINTER FOR DH11 LINE 12
313   000402'  000000       L13SW:  0                          ;BUFFER POINTER FOR DH11 LINE 13
314   000404'  000000       L14SW:  0                          ;BUFFER POINTER FOR DH11 LINE 14
315   000406'  000000       L15SW:  0                          ;BUFFER POINTER FOR DH11 LINE 15
316   000410'  000000       L16SW:  0                          ;BUFFER POINTER FOR DH11 LINE 16
317   000412'  000000       L17SW:  0                          ;BUFFER POINTER FOR DH11 LINE 17
318
319                          ;TABLE OF BAUD RATES FOR DH11 LINES
320   000414'  033500       L0BR:   033500                     ;DEFAULT BAUD RATE (9600) FOR DH11 LINE 0
321   000416'  033500       L1BR:   033500                     ;DEFAULT BAUD RATE (9600) FOR DH11 LINE 1
322   000420'  033500       L2BR:   033500                     ;DEFAULT BAUD RATE (9600) FOR DH11 LINE 2
323   000422'  033500       L3BR:   033500                     ;DEFAULT BAUD RATE (9600) FOR DH11 LINE 3
324   000424'  033500       L4BR:   033500                     ;DEFAULT BAUD RATE (9600) FOR DH11 LINE 4
325   000426'  033500       L5BR:   033500                     ;DEFAULT BAUD RATE (9600) FOR DH11 LINE 5
326   000430'  033500       L6BR:   033500                     ;DEFAULT BAUD RATE (9600) FOR DH11 LINE 6
327   000432'  033500       L7BR:   033500                     ;DEFAULT BAUD RATE (9600) FOR DH11 LINE 7
328   000434'  033500       L10BR:  033500                     ;DEFAULT BAUD RATE (9600) FOR DH11 LINE 10
329   000436'  033500       L11BR:  033500                     ;DEFAULT BAUD RATE (9600) FOR DH11 LINE 11
330   000440'  033500       L12BR:  033500                     ;DEFAULT BAUD RATE (9600) FOR DH11 LINE 12
331   000442'  033500       L13BR:  033500                     ;DEFAULT BAUD RATE (9600) FOR DH11 LINE 13
332   000444'  033500       L14BR:  033500                     ;DEFAULT BAUD RATE (9600) FOR DH11 LINE 14
333   000446'  033500       L15BR:  033500                     ;DEFAULT BAUD RATE (9600) FOR DH11 LINE 15
334   000450'  033500       L16BR:  033500                     ;DEFAULT BAUD RATE (9600) FOR DH11 LINE 16
335   000452'  033500       L17BR:  033500                     ;DEFAULT BAUD RATE (9600) FOR DH11 LINE 17
336
337
338                          ;EACH OF THE FOLLOWING 16 WORDS CONTAINS THE ERROR AND STATUS INFORMATION
339                          ;FOR 1 DH11 LINE. THE HIGH ORDER BYTE HOLDS THE # OF ERRORS IN THAT LINE
340                          ;AND THE LOW ORDER BYTE CONTAINS THE LINES CURRENT STATUS
341                          ;LOW BYTE        MODE
342                          ;   0            UNSELECTED OR DROPPED
343                          ;   3            ACTIVE MODE
344                          ;   4            DONE RECIEVING
345
346   000454'  000000       L0EC:   0                          ;ERROR COUNT/LINE STATUS FOR DH11 LINE 0
347   000456'  000000       L1EC:   0                          ;ERROR COUNT/LINE STATUS FOR DH11 LINE 1
348   000460'  000000       L2EC:   0                          ;ERROR COUNT/LINE STATUS FOR DH11 LINE 2
349   000462'  000000       L3EC:   0                          ;ERROR COUNT/LINE STATUS FOR DH11 LINE 3
350   000464'  000000       L4EC:   0                          ;ERROR COUNT/LINE STATUS FOR DH11 LINE 4
351   000466'  000000       L5EC:   0                          ;ERROR COUNT/LINE STATUS FOR DH11 LINE 5
352   000470'  000000       L6EC:   0                          ;ERROR COUNT/LINE STATUS FOR DH11 LINE 6
353   000472'  000000       L7EC:   0                          ;ERROR COUNT/LINE STATUS FOR DH11 LINE 7
354   000474'  000000       L10EC:  0                          ;ERROR COUNT/LINE STATUS FOR DH11 LINE 10
355   000476'  000000       L11EC:  0                          ;ERROR COUNT/LINE STATUS FOR DH11 LINE 11
356   000500'  000000       L12EC:  0                          ;ERROR COUNT/LINE STATUS FOR DH11 LINE 12
```

```
357   000502'  000000       L13EC:  0                          ;ERROR COUNT/LINE STATUS FOR DH11 LINE 13
358   000504'  000000       L14EC:  0                          ;ERROR COUNT/LINE STATUS FOR DH11 LINE 14
359   000506'  000000       L15EC:  0                          ;ERROR COUNT/LINE STATUS FOR DH11 LINE 15
360   000510'  000000       L16EC:  0                          ;ERROR COUNT/LINE STATUS FOR DH11 LINE 16
361   000512'  000000       L17EC:  0                          ;ERROR COUNT/LINE STATUS FOR DH11 LINE 17
362
363
364                          ;TABLE OF CHARACTER COUNT LOCATIONS FOR DH11 LINES
365   000514'  000000       L0CC:   0                          ;CHARACTER COUNT FOR DH11 LINE 0
366   000516'  000000       L1CC:   0                          ;CHARACTER COUNT FOR DH11 LINE 1
367   000520'  000000       L2CC:   0                          ;CHARACTER COUNT FOR DH11 LINE 2
368   000522'  000000       L3CC:   0                          ;CHARACTER COUNT FOR DH11 LINE 3
369   000524'  000000       L4CC:   0                          ;CHARACTER COUNT FOR DH11 LINE 4
370   000526'  000000       L5CC:   0                          ;CHARACTER COUNT FOR DH11 LINE 5
371   000530'  000000       L6CC:   0                          ;CHARACTER COUNT FOR DH11 LINE 6
372   000532'  000000       L7CC:   0                          ;CHARACTER COUNT FOR DH11 LINE 7
373   000534'  000000       L10CC:  0                          ;CHARACTER COUNT FOR DH11 LINE 10
374   000536'  000000       L11CC:  0                          ;CHARACTER COUNT FOR DH11 LINE 11
375   000540'  000000       L12CC:  0                          ;CHARACTER COUNT FOR DH11 LINE 12
376   000542'  000000       L13CC:  0                          ;CHARACTER COUNT FOR DH11 LINE 13
377   000544'  000000       L14CC:  0                          ;CHARACTER COUNT FOR DH11 LINE 14
378   000546'  000000       L15CC:  0                          ;CHARACTER COUNT FOR DH11 LINE 15
379   000550'  000000       L16CC:  0                          ;CHARACTER COUNT FOR DH11 LINE 16
380   000552'  000000       L17CC:  0                          ;CHARACTER COUNT FOR DH11 LINE 17
381
382
383                          ;DETERMINE IF ANY DH'S ARE SELECTED. IF SO SETUP DEVICE REGISTER
384                          ;ADDRESSES, INTERRUPT VECTORS, AND START THE
385                          ;MODULE PROCESSING... IF NOT, DROP THE MODULE FROM THE RUN
386
387   000554'  012767  000004  177336   START:  MOV   #4,INTR          ;4 INTERRUPTS / ITERATION
388   000562'  012767  000020  177324           MOV   #20,WDTO         ;20 WORDS TO MEM/ITERATION
389   000570'  012767  000020  177320           MOV   #20,WDFR         ;20 WORDS FROM MEM/ITERATION
390   000576'  012777  004400  177202           MOV   #004400,@ADDR    ;INIT THE DH11
391   000604'  012767  000004  177440           MOV   #4,WAIT1M        ;MAKE A LONG 1ST PASS TO ALLOW FOR STARTING VT20S
392   000612'  016706  177214                   MOV   SPOINT,SP        ;SETUP THE MODULE STACK
393   000616'  005767  177172                   TST   DVID1            ;CHECK FOR SELLECTED VT20'S
394   000622'  001155                            BNE   MODROP           ;ANY SELLECTED?
395                          ;WE END UP AT 'DROP' IF NO VT20 IS SELECTED FOR TEST (DVID1=0)
396                          ;WE WILL NOW DROP THIS MODULE
397   000624'  016767  177156  177502   DROP:   MOV   ADDR,X00         ;SETUP ADDRESS OF THE DH11 CONTROL/STATUS REG
398   000632'  012777  004000  177474           MOV   #4000,@X00       ;NO. PREVENT DH11 INTERRUPTS
399   000640'  104403  017424'           MSGN$,BEGIN,ERRNS            ;ASCII MESSAGE CALL WITH COMMON HEADER
400   000646'  104410  000000'           END$,BEGIN                  ;DROP THIS MODULE
401
402                          ;THIS IS THE RESTART ROUTINE. IT RESTARTS IN TRANSMIT MODE, ALL LINES THAT
403                          ;FINISHED RECIEVING ON THE LAST PASS. ALL OTHER LINES IF IGNCRS
404   000652'  116767  177537  177360   RESTRT: MOVB  L0BR+1,XCNT      ;GET BAUD RATE INFO
405   000660'  042767  177774  177352           BIC   #177774,XCNT     ;A ROUGH GUESS
406   000666'  005267  177346                   INC   XCNT             ;AT HOW MANY XFERS WE SHOULD DO PER PASS
407   000672'  006167  177342                   ROL   XCNT             ;MULT XCNT BY 16
408   000676'  006167  177336                   ROL   XCNT             ;TO GET THE NUMBER
409   000702'  006167  177332                   ROL   XCNT             ;OF TRANSFERS TO DO
410   000706'  006167  177326                   ROL   XCNT             ;EACH PASS
411   000712'  005467  177322                   NEG   XCNT             ;MAKE IT NEGATIVE
412   000716'  005000                   NXTFER: CLR   R0               ;START WITH LINE 0
```

```
413  000720' 012767  000003  177324          MOV   #3,WAIT1M       ;REGULAR PASS TIMEOUT
414  000726' 005067  177324                  CLR   TEMPO           ;INIT WORD WE ARE GOING TO SEND TO X12
415  000732' 005067  177312                  CLR   DUNCNT          ;CLEAR OUT COUNT FOR # OF LINES
416  000736' 126027  000454' 000004  1$:     CMPB  LOEC(0),#4      ;IS LINE READY TO XMIT?
417  000744' 001054                          BNE   2$              ;IF NOT, GO SEE ABOUT NEXT LINE
418  000746' 012760  000003  000454'         MOV   #3,LOEC(0)      ;IF IT IS READY, SET IT TO ACTIVE MODE
419  000754' 000261                          SEC                   ;SET THE C BIT AND
420  000756' 006067  177274                  ROR   TEMPO           ;ROTATE IT INTO THE NEW "BAR" REGISTER
421  000762' 005367  177262                  DEC   DUNCNT          ;ADD 1 TO LINE COUNT
422  000766' 010067  177262                  MOV   R0,TEMP         ;GRAB THE LINE #
423  000772' 006067  177250                  ROR   TEMP            ;DIVIDE IT BY2 TO MAKE IT REAL
424  001004' 042767  177244  177252          BIC   #177760,TEMP    ;MAKE SURE ONLY THE LINE # IS SET
425  001012' 016067  000514' 177234          MOV   TEMP,@X00       ;PUT IT INTO THE STATUS REGISTER
426  001020' 005467  177230          MOV   LOCC(0),TEMP    ;NOW GET THE CHARACTER COUNT
427  001024' 016777  177224  177312          NEG   TEMP            ;USE ITS 2'S COMPLIMENT
428  001032' 016067  000354' 177202          MOV   TEMP,@X10       ;AND SETUP THE BYTE COUNT REGISTER WITH IT
429          104415  000000' 000243'         MOV   LOSW(0),VIRTAD  ;GET VIRTUAL ADDRESS OF THE BUFFER
430  001040' 104415  000000' 000243' GETPAS,BEGIN, VIRTAD  ;GET PHYSICAL ADDRESS FROM 16-BIT VIRTAD
431  001044' 042767  177177  177252          BIC   #177177,EXTAD   ;CLEAR ALL BUT THE EXTEND BITS
432  001054' 050777  177166  177252          BIS   EXTAD,@X00      ;SET ADDRESS EXTEND BITS IN STATUS REG
433  001062' 016777  177156  177252          MOV   PHYSAD,@X06     ;SETUP CURRENT ADDRESS FOR THIS LINE
434  001070' 005060  000514'                 CLR   LOCC(0)         ;AND REINIT ITS CHARACTER COUNT
435  001074' 000403                          BR    3$              ;GO SEE IF WE HAVE CHECKED ALL LINES YET
436  001076' 000241          2$:             CLC                   ;CLEAR C BIT TO
437  001100' 006067  177152          3$:     ROR   TEMPO           ;MAKE SURE THAT THIS LINES "BAR" REGISTER BIT IS CLEAR
438  001104' 062700  000002                  ADD   #2,R0           ;GO ON TO THE NEXT LINE
439  001110' 020027  000040                  CMP   R0,#40          ;HAVE WE DONE ALL LINES YET?
440  001114' 001310                          BNE   1$              ;IF NOT, GO DO ANOTHER
441  001116' 005767  177134                  TST   TEMPO           ;ALL DONE, ARE ANY LINES STILL ACTIVE?
442  001122' 001010                          BNE   4$              ;IF SO, GO START EM UP XMITING
443  001124' 012777  004000  177202          MOV   #4000,@X00      ;NO LINES REMAIN, STOP THE DH11
444  001132' 104403  000000' 017432'         MSGN$,BEGIN,ERRNR     ;ASCII MESSAGE CALL WITH COMMON HEADER
445          104440  000000'                 END$,BEGIN            ;AND DROP THIS MODULE
446  001144' 016777  177106  177174  4$:     MOV   TEMPO,@X12      ;SET THE BAR REGISTER
447  001152' 000167  000004                  JMP   MAIN            ;GO WAIT FOR SOMTHING TO HAPPEN
448  001156' 004767  001060          NODROP: JSR   PC,SETUP        ;DO A BUNCH OF INITIALIZATION
449
450                                  ;THIS IS THE MAIN PROGRAM LOOP
451                                  ;NOTHING IS DONE HERE EXCEPT TO CONTINUOUSLY TEST TO SEE IF ENOUGH TIME HAS PASSED
452                                  ;FOR AN END OF PASS, CHECK FOR ERRORS, AND CALL THE
453                                  ;INTERRUPT SERVICE ROUTINE TO CLEAR OUT THE SILO IN CASE IT HAS STUFF
454                                  ;IN IT, BUT NOT ENOUGH STUFF TO CAUSE AN INTERRUPT
455
456  001162' 012777  000040  177162  MAIN:   MOV   #40,@X16        ;SET SILO TO INTERRUPT ON 40(OCTAL) CHARS
457  001170' 012706  000000'                 MOV   SPOINT,SP       ;MAKE SURE THAT THE STACK POINTER IS RIGHT
458  001174' 012777  030100  177132          MOV   #030100,@X00    ;ENABLE INTERRUPTS
459  001202' 012746  000000          MLOOP:  MOV   #0,-(SP)        ;PUT FAKE PS ONTO STACK
460  001206' 012746  001216'                 MOV   #BRAKE,-(SP)    ;AND A FAKE PC
461  001212' 000167  000464                  JMP   INTSRV          ;PRETEND THAT WE JUST HAD AN INTERRUPT
462  001216'                         BRAKE:
463  001216' 104407  000000'                 BREAK$,BEGIN          ;TEMPORARY RETURN TO MONITOR....
464          104407  000000'                 BREAK$,BEGIN          ;THEN CONTINUE AT NEXT INSTRUCTION.
465  001226' 026767  177036  177036  TSTRE:  CMP   ERPT1,ERPT2     ;ANY RECEIVE ERRORS?
466  001234' 001430                          BEQ   1$              ;IF NOT GO CHECK FOR FATAL ERRORS
467  001236' 016701  177026                  MOV   ERPT1,R1        ;IF SO, MAKE THE ERROR QUEUE PCINTER MORE ACCESSABLE
468  001242' 016767  177066  176630          MOV   X00,CSRA        ;SETUP THE CONTROL REG ADDRESS FOR PRINTING
```

```
469  001250' 012167  176626                  MOV   (R1)+,ACSR      ;SETUP THE CONTROL REG CONTENTS FOR PRINTING
470  001254' 012167  176624                  MOV   (R1)+,ASTAT     ;SETUP THE STATUS REG CONTENTS FOR PRINTING
471  001260' 104403  000000' 017464'         MSGN$,BEGIN,ERRRE     ;ASCII MESSAGE CALL WITH COMMON HEADER
472  001266' 012767  000017  176612          MOV   #17,ERRTYP      ;RECEIVER ERROR
473                                  ;**************************************************
474  001274' 104405  000000' 000000          HRDER$,BEGIN,NULL     ;RECEIVE ERROR - NON FATAL
475                                  ;**************************************************
476  001302' 010167  176762                  MOV   R1,ERPT1        ;UPDATE THE ERROR QUEUE POINTER
477  001306' 026767  176756  176756          CMP   ERPT1,ERPT2     ;IS THE ERROR QUEUE EMPTY?
478  001314' 001006                          BNE   TSTFE           ;IF IT IS NOT EMPTY, IGNORE IT FOR A MOMENT
479  001316' 012767  016702' 176744  1$:     MOV   #ERQUEU,ERPT1   ;ERROR QUEUE IS EMPTY. REINITIALIZE THE
480  001324' 012767  016702' 176740          MOV   #ERQUEU,ERPT2   ;ERROR QUEUE POINTERS
481  001332' 005767  176730          TSTFE:  TST   FATERR          ;HAVE WE HAD ANY FATAL ERRORS?
482  001336' 001132                          BNE   ERRPT           ;IF SO, GO REPORT IT AND DROP THIS MODULE
483  001340' 005767  176704                  TST   DUNCNT          ;HAVE ALL ACTIVE LINES FINISHED RECIEVING?
484  001344' 001406                          BEQ   ALDONE          ;IF SO GO CLEAN UP AND END THIS PASS
485  001346' 005267  176656                  INC   WAIT1M          ;NO, TICK GOES THE WAIT TIMER
486  001352' 001313                          BNE   MLOOP           ;HAVE ENOUGH DEVICES HAD ENOUGH TIME TO FINISH?
487  001354' 005367  176672                  DEC   WAIT1M          ;HAD ENOUGH TIME TO FINNISH
488  001360' 001310                          BNE   MLOOP           ;IF NOT.
489  001362' 042777  030100  176744  ALDONE: BIC   #030100,@X00    ;ENOUGH TIME! PREVENT FURTHER DH11 INTERRUPTS
490  001372' 005000                          CLR   R0              ;SETUP A LINE INDEX REGISTER TO 0
491  001374' 105760  000454'         1$:     TSTB  LOEC(0)         ;FIND OUT IF THE LINE IS SELECTED
492  001400' 001442                          BEQ   2$              ;IF NOT GO TEST FOR ERRORS ON IT
493  001402' 126027  000454' 000004          CMPB  LOEC(0),#4      ;IF IT IS CHECK THAT IT HAS FINISHED RECIEVING
494  001410' 001472                          BEQ   3$              ;IF IT DID AT LEAST 1 TRANSFER, DONT PRINT AN ERROR
495  001412' 012760  000000  000454'         MOV   #0,LOEC(0)      ;DIDNT FINISH, SET THE LINE TO DROPPED MODE
496  001416' 016767  176712  176454          MOV   X00,CSRA        ;SETUP FOR PRINTING THE ADRESS OF THE CONTROL/STATUS REG
497  001424' 017767  176704  176450          MOV   @X00,ACSR       ;SETUP FOR PRINTING THE CONTENTS OF THE CONTROL/STATUS R
498  001432' 016767  176620  176444          MOV   TEMPO,ASTAT     ;SETUP TO PRINT CONTENTS OF THE NEXT CHAR RECIEVED REG
499  001440' 010067  001070                  MOV   R0,NUMBA1       ;SAVE IT
500  001444' 006067  001064                  ROR   NUMBA1
501                                  ;**************************************************
502                                  ;CONVERT NUMBA1 TO ASCII AND
503                                  ;STORE AT MESNMO
504  001450' 104420  000000' 002534'         OTOA$,BEGIN,NUMBA1,MESNMO
505          017405'
506                                  ;**************************************************
507  001460' 104403  000000' 017440'         MSGN$,BEGIN,ERRDH     ;ASCII MESSAGE CALL WITH COMMON HEADER
508  001466' 012767  000023  176412          MOV   #23,ERRTYP      ;FAILED TO INTERRUPT
509                                  ;**************************************************
510  001474' 104405  000000' 000000          HRDER$,BEGIN,NULL     ; A DH11 LINE HUNG
511                                  ;**************************************************
512  001502' 000434                          BR    3$              ;GO TRY THE NEXT LINE FOR ERRORS
513                                  ;**************************************************
514  001504' 105760  000455'         2$:     TSTB  LOEC+1(0)       ;ANY ERRORS ON THIS UNSELECTED LINE?
515  001510' 001431                          BEQ   3$              ;IF NOT TRY THE NEXT LINE
516  001512' 010046                          MOV   R0,-(SP)        ;PUT THE LINE INDEX ONTO THE STACK
517  001514' 006016                          ROR   (SP)            ;DIVIDE BY 2 TO GET THE LINE #
518  001516' 005046                          CLR   -(SP)           ;CLEAR OUT THE FIRST WORD ON THE STACK
519  001520' 116016  000455'                 MOVB  LOEC+1(0),(SP)  ;SO THAT WE CAN PUT THE # OF ERRORS INTO LOW BYTE
520  001524' 011667  001006                  MOV   (SP),NUMBA2     ;SAVE
521                                  ;**************************************************
522                                  ;CONVERT NUMBA2 TO ASCII AND
523                                  ;STORE AT MESNMO
524  001530' 104420  000000' 002536'         OTOA$,BEGIN,NUMBA2,MESNMO
```

```
 525   001536'  017405'
 526                                              ;*********************************************************
 527   001540'  005726                     TST     (SP)+               ;CLEAN UP STACK
 528   001542'  010046                     MOV     R0,-(SP)            ;PUT LINE # (*2) ONTO THE STACK
 529   001544'  000241                     CLC                         ;MAKE SURE THE C BIT IS CLEAR
 530   001546'  006016                     ROR     (SP)                ;MAKE IT THE LINE #
 531   001550'  011667   000764            MOV     (SP),NUMBA3         ;SAVE IT
 532                                              ;*********************************************************
 533                                              ;CONVERT NUMBA3 TO ASCII AND
 534                                              ;STORE AT MESNM1
 535   001554'  104420   000000' 002540'   OTOA$,BEGIN,NUMBA3,MESNM1
 536   001562'  017414'
 537                                              ;*********************************************************
 538   001564'  005726                     TST     (SP)+
 539   001566'  104403   000000' 017472'   MSGN$,BEGIN,ERRUS           ;ASCII MESSAGE CALL WITH COMMON HEADER
 540   001574'  062700   000002     3$:    ADD     #2,R0               ;TRY THE NEXT LINE
 541   001600'  022700   000040            CMP     #40,R0              ;DONE 'EM ALL YET?
 542   001604'  001272                     BNE     1$                  ;IF NOT GO TRY ANOTHER
 543   001606'  005267   176426            INC     XCNT                ;DONE ENOUGH XFERS THISS PASS?
 544   001612'  001402                     BEQ     4$                  ;IF SO, GO END THE PASS
 545   001614'  000167   177076            JMP     NXTFER              ;IF NOT, GO START ANOTHER XFER
 546   001620'                       4$:
 547   001620'  104413   000000'            ENDIT$,BEGIN               ;SIGNAL END OF ITERATION.
 548                                                                   ;MONITOR SHALL TEST END OF PASS
 549                                                                   ;I'V GOT FAITH
 550                                              ;FATAL ERROR WILL BRING US HERE, WHERE WE REPORT IT AND DROP
 551                                              ;THIS MODULE
 552   001624'  016767   176504 176246    ERRPT: MOV     X00,CSRA      ;IT IS AN ERROR. SETUP CONTROL REG ADDR
 553   001632'  017767   176476 176242           MOV     @X00,ACSR     ;SETUP CONTROL REG DATA
 554   001640'  017767   176506 176236           MOV     @X16,ASTAT    ;SETUP SILO STATUS REGISTER DATA
 555   001646'  012777   004000 176460           MOV     #4000,@X00    ;PREVENT ANY MORE DH11 INTERRUPTS
 556   001654'  104403   000000' 017456'         MSGN$,BEGIN,ERRSO     ;ASCII MESSAGE CALL WITH COMMON HEADER
 557   001662'  012767   000017 176216           MOV     #17,ERRTYP    ;RECEIVER ERROR
 558                                              ;*********************************************************
 559   001670'  104405   000000' 000000    HRDER$,BEGIN,NULL           ;RECEIVE ERROR - SILO STORAGE OVERFLOW
 560                                              ;*********************************************************
 561   001676'  104410   000000'           END$,BEGIN                  ;
 562
 563                                       ;THIS IS THE RECEIVER INTERRUPT SERVICE ROUTINE
 564                                       ;ONCE WE GET IN HERE WE STAY UNTILL 4 CHARS ARE PULLED FROM THE  SILO
 565                                       ;SEQUENCE OF EVENTS IN THIS SUBROUTINE IS AS FOLLOWS
 566                                       ;FIRST WE  TEST TO SEE IF IT WAS REALLY A RECEIVER INTERRUPT OF IF IT
 567                                       ;WAS A SILO OVERFLOW INTERRUPT.
 568                                       ;IF IT WAS A SILO OVERFLOW INTERRUPT WE CLEAR THE DH11, SET
 569                                       ;A FATAL ERROR SWITCH, AND IMMEDIATLY RETURN, VIA A 'RTI' INSTRUCTION
 570                                       ;THE SWITCH WILL BE EXAMINED IN THE 'MAIN' PROGRAM LOOP AND PROPER ACTIONS
 571                                       ;WILL BE TAKEN IF IT IS FOUND SET
 572                                       ;IF THE INTERRUPT WAS REALLY A RECEIVER INTERRUPT, A CHARACTER IS TAKEN
 573                                       ;FROM THE SILO. THE WORD TAKEN FROM THE SILO IS CHECKED FOR
 574                                       ;ITS VALID DATA BIT (BIT15) SET. IF IT IS NOT SET A 'RTI' IS PERFORMED.
 575                                       ;IF THERE IS VALID DATA, THE CHARACTER IS PLACED INTO THE PROPER OUTPUT BUFFER.
 576                                       ;THE WORD TAKEN FROM THE SILO IS ALSO CHECKED FOR ERROR BITS, AND IF
 577                                       ;ANY ARE SET, THE CONTROL REGISTER AND STATUS REGISTER ARE PLACED
 578                                       ;INTO THE ERROR QUEUE BUFFER, SO THAT THE ERROR CAN BE REPORTED LATER, AT PRIORITY 0
 579                                       ;IN ANY CASE THE CHARACTER IS THEN CHECKED TO SEE IF IT IS A '14' CODE
 580                                       ;WHICH SIGNIFIES 'END OF MESSAGE'.
```

```
 581                                       ;IF IT IS A '14' CODE THE 'XMIT' ROUTINE IS ENTERED. THE 'XMIT' ROUTINE
 582                                       ;RESETS THE LINES BUFFER POINTER
 583                                       ;CODE '14' OR NOT, THE PROGRAM LOOPS BACK AND TAKES ANOTHER CHARACTER
 584                                       ;FROM THE SILO.
 585                                       ;IF THIS ROUTINE USES R0, IT RESTORES IT AFTERWORD
 586   001702'  032777   040000 176424    INTSRV: BIT     #040000,@X00  ;CHECK FOR RECEIVER ERRORS
 587   001710'  001412                     BEQ     RINT                 ;IF ITS A RECEIVER INTERRUPT
 588   001712'  017777   004000 176414     MOV     #4000,@X00           ;FATAL ERROR INTERRUPT!!! STOP THE DH11
 589   001720'  012767   000001 176340     MOV     #1,FATERR            ;SET THE FATAL ERROR SWITCH SO WE KNOW TO REPORT IT
 590   001726'  012767   000004 176330     MOV     #4,INTSW             ;SETUP FOR 4 CHARS PULLED OFF THE SILO
 591   001734'  000002                     RTI                          ;RETURN
 592   001736'                       RINT:
 593                                        ;------------------------------------------------------------------
 594   001736'  000004   000000' 001744'   PIRQ$,BEGIN,XXXX             ; QUEUE UP TO CONTINUE AT XXXX AND RTI
 595                                        ;------------------------------------------------------------------
 596   001744'  010067   176112    XXXX:   MOV     R0,SVR0             ;MUST SAVE R0. IT MAY BELONG TO ANOTHER MODULE
 597   001750'  012767   000020 176306     MOV     #20,INTSW           ;SERVICE 20 CHARS
 598   001756'  017767   176354 176272     YYYY:   MOV     @X02,TEMPO  ;GET FIRST CHAR IN THE SILO
 599   001764'  100077                     BPL     RETRN               ;IF SILO IS EMPTY, QUIT TAKING STUFF FROM IT
 600   001766'  116265   176264            MOVB    TEMPO+1,TEMP1       ;GET ERROR BITS AND LINE #
 601   001774'  016700   176260            MOV     TEMP1,R0            ;MAKE LINE # MORE ACCESSABLE
 602   002000'  006100                     ROL     R0                  ;IF NOT MULTIPLY THE LINE # *2
 603   002002'  042700   177741            BIC     #177741,R0          ;AND ISOLATE THE LINE # IN BITS 1-4
 604   002006'  005760   000454'    1$:    TST     LOEC(0)             ;IS THE LINE SELECTED?
 605   002012'  001011                     BNE     RCVMOD              ;IF IT IS
 606   002014'  105269   000455'           INCB    LOEC+1(0)           ;IF NOT SELECTED ITS AN ERROR. INC ERROR COUNT
 607   002020'  005367   176240            DEC     INTSW               ;4 CHARS PULLED YET?
 608   002024'  001361                     BNE     YYYY                ;IF NOT GO BACK AND DO ANOTHER
 609   002026'  016700   176030            MOV     SVR0,R0             ;YES, DONE 4 ALREADY. RESTORE R0
 610   002032'  104400   000000'           EXIT$,BEGIN                 ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
 611   002036'  116770   176214 000354'   RCVMOD: MOVB    TEMPO,@LOSW(0) ;PUT THE CHAR INTO THE PROPER BUFFER
 612   002044'  005260   000354'           INC     LOSW(0)             ;BUMP BUFFER POINTER
 613   002050'  005269   000514'           INC     LOCC(0)             ;INCREMENT THE CHARACTER COUNT BY +1
 614   002054'  032761   000140 176176     BIT     #000140,TEMP1       ;TEST FOR ERRORS
 615   002062'  001426                     BEQ     NOERR               ;ERRORS?
 616   002064'  105269   000455'           INCB    LOEC+1(0)           ;YES, ADD 1 TO THE ERROR COUNT FOR THAT LINE
 617   002070'  022767   176176 017072'    CMP     ERPT2,#ERQOFL       ;IS THE ERROR QUEUE FULL?
 618   002076'  001003                     BNE     1$                  ;IF NOT.
 619   002100'  016702   176164            MOV     #ERQUEU,ERPT2       ;YES ITS FULL. RESET IT
 620   002106'  017777   176222 176156    1$:    MOV     @X00,@ERPT2   ;SAVE CONTENTS OF CONTROL REG
 621   002114'  062767   176130 176142     ADD     #2,ERPT2            ;UPDATE ERROR QUEUE POINTER
 622   002122'  017767   176136 176142     MOV     @X16,@ERPT2         ;SAVE CONTENTS OF STATUS REG
 623   002130'  062767   000002 176134     ADD     #2,ERPT2            ;UPDATE ERROR QUEUE POINTER
 624   002136'  122767   000014 176112    NOERR: CMPB    #14,TEMPO     ;CHECK FOR AN END OF PARAGRAPH CHAR
 625   002144'  001417                     BEQ     XMIT                ;IF EOP FOUND, GOTO RECIEVE DONE MODE
 626   002146'  005367   176112            DEC     INTSW               ;20 CHARS PULLED YET?
 627   002152'  001301                     BNE     YYYY                ;IF NOT GO BACK AND DO ANOTHER
 628   002154'  016700   175702            MOV     SVR0,R0             ;YES, DONE 4 ALREADY. RESTORE R0
 629   002160'  104400   000000'           EXIT$,BEGIN                 ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
 630   002164'  016700   175672   RETRN:   MOV     SVR0,R0             ;RESTORE R0
 631   002170'  104400   000000'           EXIT$,BEGIN                 ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
 632                                       ;THIS ROUTINE SERVICES TRANSMITER INTERRUPTS
 633                                       ;IT DOES ABSOLUTLY NOTHING !!!
 634   002174'  042777   100000 176132    XINT:  BIC     #100000,@X00   ;WE DONT NEED ANOTHER INTERRUPT
 635   002202'  000002                     RTI                          ;RETURN
 636
```

```
 637                                        ;THIS ROUTINE IS ENERED WHEN A LINE FINISHES RECEIVING
 638   002204' 112760 000004 000454'  XMIT:   MOVB    #4,LOEC(0)        ;SET LINE TO "RECIEVE DONE" MODE
 639   002212' 166060 000514' 000354'          SUB     LOCC(0),LOSW(0)  ;RESET THE LINES BUFFER POINTER
 640   002220' 005267 176024                    INC     DUNCNT           ;ALSO TALLY UP 1 MORE FINISHED DEVICE
 641   002224' 005367 176034                    DEC     INTSW            ;20 CHARS PULLED YET?
 642   002230' 001452                           BNE     YYYY             ;IF NOT GO BACK AND DO ANOTHER
 643   002232' 016700 175624'                   MOV     SVR0,R0          ;YES, DONE 4 ALREADY. RESTORE R0
 644   002236' 104400 000000'                   EXIT$,BEGIN              ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
 645
 646                                        ;THIS IS THE INITIALIZATION ROUTINE.
 647                                        ;FIRST IT CLEARS OUT THE FATAL ERROR SWITCH
 648                                        ;THEN IT SETS UP POINTERS TO ALL OF THE DH11 REGISTERS
 649                                        ;THEN IT ACCESSES EACH DH11 REGISTER WITH A "TST" INSTRUCTION TO
 650                                        ;MAKE SURE IT IS REALLY THERE.
 651                                        ;THEN IT ACCESSES EACH DH11 REGISTER USING A "TST" INSTRUCTION SO THAT IF
 652                                        ;THE REGISTER DOES NOT EXIST OR HAS MAJOR PROBLEMS, WE FIND OUT RIGHT AWAY
 653                                        ;THEN IT SETS UP DH11 VECTORS
 654                                        ;THEN IT INITIALIZES THE ERROR QUEUE POINTERS
 655                                        ;THEN IT ASSIGNS BUFFER POINTERS TO ALL SELECTED LINES
 656   002242' 005067 176020          SETUP:  CLR     FATERR           ;INIT FATAL ERROR SWITCH = NO FATAL ERRORS
 657   002246' 016700 175534                   MOV     ADDR,R0          ;MAKE THE DEVICE REGISTER ADDRESS EASY TO ACCESS
 658   002252' 010067 175724                   MOV     R0,X00           ;SETUP ADDRESS OF THE SYSTEM CONTROL REGISTER
 659   002256' 005720                           TST     (R0)+            ;TRAP IF IT DOES NOT EXIST
 660   002260' 010067 176052                   MOV     R0,X02           ;SETUP ADDRESS OF NEXT CHARACTER RECEIVED REGISTER
 661   002264' 005720                           TST     (R0)+            ;TRAP IF IT DOES NOT EXIST
 662   002266' 010067 176046                   MOV     R0,X04           ;SETUP ADDRESS OF LINE PARAMETER REGISTER
 663   002272' 005720                           TST     (R0)+            ;TRAP IF IT DOES NOT EXIST
 664   002274' 010067 176042                   MOV     R0,X06           ;SETUP ADDRESS OF CURRENT ADDRESS REGISTER
 665   002300' 005720                           TST     (R0)+            ;TRAP IF IT DOES NOT EXIST
 666   002302' 010067 176036                   MOV     R0,X10           ;SETUP ADDRESS OF BYTE COUNT REGISTER
 667   002306' 005720                           TST     (R0)+            ;TRAP IF IT DOES NOT EXIST
 668   002310' 010067 176032                   MOV     R0,X12           ;SETUP ADDRESS OF BUFFER ACTIVE REGISTER
 669   002314' 005720                           TST     (R0)+            ;TRAP IF IT DOES NOT EXIST
 670   002316' 010067 176026                   MOV     R0,X14           ;SETUP ADDRESS OF BREAK CONTROL REGISTER
 671   002322' 005720                           TST     (R0)+            ;TRAP IF IT DOES NOT EXIST
 672   002324' 010067 176022                   MOV     R0,X16           ;SETUP THE ADDRESS OF THE SILO STATUS REGISTER
 673   002330' 005720                           TST     (R0)+            ;TRAP IF IT DOES NOT EXIST
 674                                        ;NOW SETUP DH11 INTERRUPT VECTORS. THERE ARE 2 OF THEM, ONE
 675                                        ;FOR RECEIVER INTERRUPTS, AND 1 FOR TRANSMITER INTERRUPTS
 676   002332' 016700 175452'                   MOV     VECTOR,R0        ;GET THE VECTOR ADDRESS
 677   002336' 012720 001702'                   MOV     #INTSRV,(R0)+    ;POINT VECTOR TO RECEIVER INTERRUPT SERVICE
 678   002342' 116720 175444                     MOVB    BR1,(R0)+        ;SET PRIORITY LEVEL FOR THE INTERRUPT ROUTINE
 679   002346' 105020                           CLRB    (R0)+            ;SET R0 TO AN EVEN ADDRESS
 680   002350' 012720 002174'                   MOV     #XINT,(R0)+      ;POINT VECTOR AT TRANSMITER SERVICE ROUTINE
 681   002354' 116720 175432                     MOVB    BR1,(R0)+        ;PRIORITY LEVEL FOR THE TRANSMITER INTERRUPT ROUTINE
 682   002360' 105020                           CLRB    (R0)+            ;SET R0 TO AN EVEN ADDRESS
 683                                        ;NOW SETUP ERROR QUEUE POINTERS
 684   002362' 012767 016702' 175700          MOV     #ERQUEU,ERPT1    ;SET THE "WHERE WE'VE SERVICED TO" ERROR QUEUE
 685                                                                      ;POINTER TO THE BEGINNING OF THE QUEUE AREA
 686   002370' 012767 016702' 175674          MOV     #ERQUEU,ERPT2    ;SET THE "WHERE WE'VE PUT STUFF" ERROR QUEUE
 687                                                                      ;POINTER TO THE BEGINING OF THE QUEUE AREA
 688                                        ;NOW ASSIGN BUFFER POINTERS TO SELECTED LINES (DONE ONLY ONCE AT THE BEGINING)
 689   002376' 005067 175622                   CLR     SELCNT           ;INIT DEVICE SELECT COUNT TO 0 BEFORE COUNTING VT20'S
 690   002402' 005067 175642                   CLR     DUNCNT           ;AND OF COURSE NONE HAVE FINISHED YET
 691   002406' 016701 175402                   MOV     DVID1,R1         ;MAKE BITMAP OF SELECTED VT20 LINES MORE ACCESSABLE
 692   002412' 005000                           CLR     R0               ;INIT FOR 16 POSSIBLE LINES
```

```
 693   002414' 005067 175636                   CLR     TEMPO            ;INIT THE LINE #
 694   002420' 005060 000454'          1$:      CLR     LOEC(0)          ;ASSUME LINE IS UNSELECTED UNTIL PROVEN OTHERWISE
 695   002424' 006001                           ROR     R1               ;PUT A BIT INTO THE C-BIT
 696   002426' 103031                           BCC     2$               ;IS THE LINE TO BE USED ?
 697   002430' 016060 000274' 000354'           MOV     BPLIST(0),LOSW(0);YES, ASSIGN A BUFFER POINTER TO THAT LINE
 698   002436' 042777 000017 175670             BIC     #17,@X00         ;CLEAR OUT LINE SELECTION BITS
 699   002444' 005367 175600                     DEC     DUNCNT           ;ADD 1 TO COUNT OF ACTIVE DEVICES
 700   002450' 000240                           NOP
 701   002452' 056777 175600 175654             BISB    TEMPO,@X00       ;SET LINE SELECTION BITS FOR THIS LINE
 702   002460' 016077 000414' 175652           MOV     LOBR(0),@X04     ;SETUP BAUD RATES FOR THIS LINE
 703   002466' 052777 000023 175644             BIS     #23,@X04         ;PARITY-FULL DUPLEX-8 BIT
 704   002474' 005060 000514'                   CLR     LOCC(0)          ;CLEAR OUT THIS LINES CHARACTER COUNT
 705   002500' 012760 000003 000454'           MOV     #3,LOEC(0)       ;START OUT THE LINE IN RECIEVE MODE
 706   002506' 005267 175512                     INC     SELCNT           ;UPDATE COUNT OF SELECTED VT20'S
 707   002512' 062700 000002          2$:      ADD     #2,R0            ;GO ON TO THE NEXT LINE
 708   002516' 005267 175534                     INC     TEMPO            ;NEXT LINE
 709   002522' 026727 175530 000017             CMP     TEMPO,#17        ;CHECKED ALL LINES FOR SELECTION YET ?
 710   002530' 001333                           BNE     1$               ;IF NOT, GO BACK AND CHECK THE NEXT ONE
 711   002532' 000207                           RTS     PC               ;RETURN
 712
 713
 714
 715   002534' 000000          NUMBA1:  .WORD   0
 716   002536' 000000          NUMBA2:  .WORD   0
 717   002540' 000000          NUMBA3:  .WORD   0
 718
 719   002542' 000303          VTBF0:   .BLKW   195.                  ;BUFFER SPACE FOR 1ST SELECTED DH11 LINE
 720   003350' 000303          VTBF1:   .BLKW   195.                  ;BUFFER SPACE FOR 2ND SELECTED DH11 LINE
 721   004156' 000303          VTBF2:   .BLKW   195.                  ;BUFFER SPACE FOR 3RD SELECTED DH11 LINE
 722   004764' 000303          VTBF3:   .BLKW   195.                  ;BUFFER SPACE FOR 4TH SELECTED DH11 LINE
 723   005572' 000303          VTBF4:   .BLKW   195.                  ;BUFFER SPACE FOR A SELLECTED DH11 LINE
 724   006400' 000303          VTBF5:   .BLKW   195.                  ;BUFFER SPACE FOR A SELLECTED DH11 LINE
 725   007206' 000303          VTBF6:   .BLKW   195.                  ;BUFFER SPACE FOR A SELLECTED DH11 LINE
 726   010014' 000303          VTBF7:   .BLKW   195.                  ;BUFFER SPACE FOR A SELLECTED DH11 LINE
 727   010622' 000303          VTBF10:  .BLKW   195.                  ;BUFFER SPACE FOR A SELLECTED DH11 LINE
 728   011430' 000303          VTBF11:  .BLKW   195.                  ;BUFFER SPACE FOR A SELLECTED DH11 LINE
 729   012236' 000303          VTBF12:  .BLKW   195.                  ;BUFFER SPACE FOR A SELLECTED DH11 LINE
 730   013044' 000303          VTBF13:  .BLKW   195.                  ;BUFFER SPACE FOR A SELLECTED DH11 LINE
 731   013652' 000303          VTBF14:  .BLKW   195.                  ;BUFFER SPACE FOR A SELLECTED DH11 LINE
 732   014460' 000303          VTBF15:  .BLKW   195.                  ;BUFFER SPACE FOR A SELLECTED DH11 LINE
 733   015266' 000303          VTBF16:  .BLKW   195.                  ;BUFFER SPACE FOR A SELLECTED DH11 LINE
 734   016074' 000303          VTBF17:  .BLKW   195.                  ;BUFFER SPACE FOR A SELLECTED DH11 LINE
 735   016702' 000074          ERQUEU:  .BLKW   60.                   ;RECEIVER ERROR QUEUE SPACE
 736   017072' 000000          ERQOFL:  .WORD   0                     ;ERROR QUEUE OVERFLOW BOUNDARY
 737
 738                                        ;FOLLOWING IS THE TEXT FOR ALL ASCII ERROR MESSAGES
 739   017074' 047516 046040 047111  MESNR:  .ASCIZ /NO LINES REMAIN ACTIVE/
 740   017102' 051505 051040 046505
 741   017110' 044501 020116 041501
 742   017116' 044524 042529 000
 743   017123' 040 047516 053040  MESNS:  .ASCIZ / NO VT20'S SELECTED/
 744   017130' 031124 023460 020123
 745   017136' 042524 042505 052103
 746   017144' 042105 000
 747   017147' 040 044514 042516  MESDH:  .ASCIZ / LINE /
 748   017154' 000040
```

```
 749   017156· 042040  042111  047040   MESDH1: .ASCIZ  / DID NOT COMPLETE 1 TRANSFER IN TIME/
 750   017164· 052117  041440  046517
 751   017172· 046120  052105  020105
 752   017200· 020061  051124  047101
 753   017206· 043123  051105  044440
 754   017214· 020116  044524  042515
 755   017223·    000
 756          123   020117  042527   MESDH2: .ASCIZ  /SO WE ARE DROPING THE LINE/
 757   017230· 040440  042522  042040
 758   017236· 047522  044520  043516
 759   017244· 052040  042510  046040
 760   017252· 047111  000105
 761   017256· 042440  051122  051117   MESUS:  .ASCIZ  / ERRORS ON UNSELECTED LINE/
 762   017263· 020123  047117  052440
 763   017272· 051413  046105  041505
 764   017300· 042524  020104  044514
 765   017306· 042516    000
 766   017311·    040   044523  047514   MESSO:  .ASCIZ  / SILO OVERFLOW... ITS FATAL/
 767   017316· 047440  042526  043122
 768   017324· 047514  027127  027056
 769   017332· 044440  051524  043040
 770   017340· 052101    000
 771   017345·    040   020101  042522   MESRE:  .ASCIZ  / A RECEIVER ERROR BIT WAS SET/
 772   017352· 042503  053111  051105
 773   017360· 042440  051122  051117
 774   017366· 041040· 052111  053440
 775   017374· 051501  051440  052105
 776   017402·    000
 777   017403·    045
 778   017405·    060   030060  030060   MESBE:  .ASCIZ  /%/
                                         MESNM0: .ASCIZ  /000000/
 779   017412· 000060                    
 780   017414· 030060  030060  030060   MESNM1: .ASCIZ  /000000/
 781   017422·    000
 782   017424·                           .EVEN                        ;IN CASE THERE IS AN ODD # OF BYTES IN THE ABOVE MESSAGE
 783
 784                                     ;TABLE OF ERROR MESSAGE POINTERS & TERMINATORS (COMMENTED!)
 785   017424· 017403·          ERRNS:   MESBE                        ;POINTS TO A CARRIAGE RETURN-LINE FEED
 786   017426· 017123·                   MESNS                        ;POINTS TO AN ASCIZ ERROR MESSAGE
 787   017430· 177777                    177777                       ;TERMINATOR. ONLY 1 MESSAGE THIS TIME
 788   017432· 017403·          ERRNR:   MESBE                        ;POINTS TO A CARRIAGE RETURN-LINE FEED
 789   017434· 017074·                   MESNR                        ;POINTS TO "NO LINES REMAIN ACTIVE" MESSAGE
 790   017436· 177777                    177777                       ;TERMINATE MESSAGE
 791   017440· 017403·          ERRDH:   MESBE                        ;POINTS TO A CARRIAGE RETURN-LINE FEED
 792   017442· 017147·                   MESDH                        ;POINTS TO 1ST PART OF "HUNG" ERROR MESSAGE ASCIZ
 793   017444· 017405·                   MESNM0                       ;POINTS TO LINE # ASCIZ
 794   017446· 017156·                   MESDH1                       ;POINTS TO PART OF MESSAGE
 795   017450· 017403·                   MESBE                        ;RETURN
 796   017452· 017223·                   MESDH2                       ;DROPING A LINE MESSAGE
 797   017454· 177777                    177777                       ;END OF MESSAGE
 798   017456· 017403·          ERRSO:   MESBE                        ;POINTS TO A CARRIAGE RETURN-LINE FEED
 799   017460· 017311·                   MESSO                        ;POINTS TO ASCIZ TEXT OF FATAL ERROR MESSAGE
 800   017462· 177777                    177777                       ;THIS INDICATES THAT NO MORE MESSAGES FOLLOW
 801   017464· 017403·          ERRRE:   MESBE                        ;POINTS TO A CARRIAGE RETURN-LINE FEED
 802   017466· 017345·                   MESRE                        ;POINTS TO ASCIZ TEXT OF THE RECEIVER ERROR MESSAGE
 803   017470· 177777                    177777                       ;TERMINATOR
 804   017472· 017403·          ERRUS:   MESBE                        ;POINTS TO A CARRIAGE RETURN-LINE FEED
```

```
 805   017474· 017405·                   MESNM0        ;POINTS TO ASCIZ FOR # OF ERRORS
 806   017476· 017256·                   MESUS         ;POINTS TO TEXT PART OF MESSAGE
 807   017500· 017414·                   MESNM1        ;POINTS TO ASCIZ FOR LINE #
 808   017502· 177777                    177777        ;SIGNALS END OF MESSAGE TO DECX/11
 809                                     ;END OF MESSAGE TABLE
 810          000001                     .END
```

```
ACSR     000102R     233#    469*    497*    553*
ADDR     000006R     199#    390*    397     657
ADDR22=  001000      151#
ALDONE   001362R     484#    489#
ASB      000106R     237#
ASTAT    000104R     235#    470*    498*    554*
AWAS     000110R     238#
BEGIN    000000R     196#    399     400     439     444     445     463     461     471     474     504     507     510
                     561#    535     539     547     556     559     561     594     610     629     631     644
BIT0  =  000001      561#
BIT1  =  000002      251#
BIT10 =  002000      251#
BIT11 =  004000      251#
BIT12 =  010000      251#
BIT13 =  020000      251#
BIT14 =  040000      251#
BIT15 =  100000      251#
BIT2  =  000004      251#
BIT3  =  000010      251#
BIT4  =  000020      251#
BIT5  =  000040      251#
BIT6  =  000100      251#
BIT7  =  000200      251#
BIT8  =  000400      251#
BIT9  =  001000      251#
BPLIST   000274R     275#    697#
BRAKE    001216R     460     462#
BREAK$=  104407      251#    463#    464
BR1      000012R     201#    678     681
BR2      000013R     202#
BTOD$ =  104421      251#
CDATA$=  104412      251#
CONFIG   000056R     221#
COUNTI   000262R     267#
CSRA     000100R     231#    468*    496*    552*
DATCK$=  104411      251#
DATER$=  104404      251#
DROP     000624R     397#
DUNCNT   000250R     262#    415*    421*    483     640*    690*    699*
DVID1    000014R     203#    393     691
ENDIT$=  104413      251#    547
END$  =  104410      251#    400     445     561
ERPT1    000270R     272#    467     476*    477     479*    684*
ERPT2    000272R     273#    465     477     480*    617     619*    620*    621*    622*    623*    686*
ERQUEL   016702R     616#    736#    619     684     686     735#
ERQUEU   016702R     479#    480     791#
ERRDH    017440R     507#    788#
ERRMR    017432R     444#    785#
ERRNS    017424R     399#    552#
ERRPT    001624R     482#    801#
ERRRE    017464R     471#    792#
ERRSO    017456R     556#    508*    557*
ERRTYP   000106R     236#    804#
ERRUS    017472R     539#
EXIT$ =  104400      251#    610     629     631     644
EXTAD    000246R     261#    431*    432
```

```
FATERR   000266R     271#    481     589*    656*
GETPA$=  104415      251#    430
GMBUF$=  104414      251#
HRDCNT   000044R     216#    474     510     559
HRDER$=  104405      251#
HRDPA$   000050R     218#
ICONT    000036R     213#
ICOUNT   000040R     214#
IDNUM    000142R     210#
INTT     000036R     242#    387*
INTR     000120R             586#    677
INTSRV   001702R     461#    590*    597*    607*    626*    641*
INTSW    000264R     269#    702
LOBR     000414R     321#    404     434*    613*    639     704*
LOCC     000514R     365#    426     418*    491     493     495*    514     519     604     606*    616*    638*    694*
LOEC     000454R     416#    702#
LOSW     000354R     302#    429     611*    612*    639*    697*
L1BR     000416R     322#
L1CC     000516R     366#
L1EC     000456R     347#
L1SW     000356R     303#
L10BR    000434R     330#
L10CC    000534R     343#
L10EC    000474R     354#
L10SW    000374R     310#
L11BR    000436R     330#
L11CC    000536R     374#
L11EC    000476R     357#
L11SW    000376R     311#
L12BR    000440R     331#
L12CC    000500R     356#
L12EC    000500R     358#
L12SW    000400R     312#
L13BR    000442R     332#
L13CC    000542R     379#
L13EC    000502R     313#
L13SW    000402R     313#
L14BR    000444R     333#
L14CC    000544R     377#
L14EC    000504R     358#
L14SW    000404R     314#
L15BR    000446R     334#
L15CC    000546R     359#
L15EC    000506R     315#
L15SW    000406R     315#
L16BR    000450R     335#
L16CC    000550R     379#
L16EC    000510R     360#
L16SW    000410R     316#
L17BR    000452R     336#
L17CC    000552R     360#
L17EC    000512R     361#
L17SW    000412R     317#
L2BR     000420R     323#
L2CC     000520R     367#
```

```
L2EC    000460R    348#
L2SW    000360R    304#
L3BR    000422R    324#
L3CC    000522R    368#
L3EC    000462R    305#
L3SW    000365R    325#
L4BR    000424R    369#
L4CC    000524R    350#
L4EC    000464R    306#
L4SW    000364R    346#
L5BR    000426R    371#
L5CC    000526R    351#
L5EC    000466R    307#
L5SW    000366R    327#
L6BR    000430R    371#
L6CC    000530R    371#
L6EC    000470R    352#
L6SW    000370R    308#
L7BR    000432R    371#
L7CC    000532R    340#
L7EC    000474R    353#
L7SW    000372R    309#
MAIN    001162R    447#   456#
MAP22$= 104416     754#   785    788    791    795    798    801    804
MESBR   017403R    747#   782
MESDH1  017156R    749#   794
MESDH2  017223R    756#   796
MESNMO  017405R    504#   524    778#   793    805
MESNM1  017414R    535#   780#   807#
MESNR   017074R    739#   789
MESNS   017123R    747#   786
MESRR   017315R    802#   902
MESSO   017311R    766#   999
MESUS   017256R    761#   806
MLOOP   001202R    459#   486    488
MODNAM  000000R    197#
MODSP   000224R    211#   249#
MSCM$ = 104403     251#   399    444    471    507    539    556
MSGS$ = 104402     251#
MSG$  = 104401     251#
MODROP  001156R    594#   448#
MOERR   002136R    615#   624#
NULL  = 000000     251#   474    510    559
NUMBA1  002534R    499#   500*   504    715#
NUMBA2  002536R    520*   524    716#
NUMBA3  002540R    545#   717#
NXTFER  000716R    471*   204    205    206    207    224    225    226#   227    228    229    230    231
OPEN  = 000000     198#   235    237    238    240    241    242    251#
                          504    524    535
OTOA$ = 104420     251#
PASCNT  000034R    212#
PHYSAD  000244R    460#   433
PIRQS   000004     251#
POPSP   005726     251#
POPSP2= 022626     251#
```

```
PRTY  = 000000     202#   251#
PRTY0 = 000000     251#
PRTY1 = 000040     251#
PRTY2 = 000100     251#
PRTY3 = 000140     251#
PRTY4 = 000200     251#
PRTY5 = 000240     201#   251#
PRTY6 = 000300     251#
PRTY7 = 000340     251#
PS    = 177776     251#
PSW   = 177776     251#
PUSH  = 005746     251#
PUSH2 = 024646     251#
RANDS = 104417     251#
RANNUM  000054R    220#
RCVMOD  002036R    605#   611#
RBLOC   000032R    235*   404#
RESTRT  000652R    239#
RES1    000056R    222#
RES2    000060R    223#
RETRN   007164R    599#   630#
RINT    001736R    587#   592#
RSTRT   001112R    589#
SAVR0   000226R    253#
SBADR   000102R    232#
SELCNT  000224R    252#   689*   706*
SETUP   002242R    448#   656#
SOFCNT  000042R    215#
SOFERS= 104406     251#
SOFPAS  000046R    211#   392    457
SPOINT  000032R    244#
SPSIZ = 000040       1#
SR1     000016R    204#
SR2     000020R    205#
SR3     000022R    206#
SR4     000024R    207#
START   000064R    210#   387#
STAT    000026R    209#
SVR0    000062R    224#   596*   609    628    630    643
SVR1    000064R    225#
SVR2    000066R    226#
SVR3    000070R    227#
SVR4    000072R    228#
SVR5    000074R    229#
SVR6    000076R    230#
SVR7    000234R    231#
SVT1    000236R    257#
SYSCNT  000052R    219#
TEMP    000254R    264#   422*   423*   424*   425    426*   427*   428*   600    611    622    624    693*
TEMP0   000256R    265#   414*   420*   437*   441    446    498    598*
                          708*   709
TEMP1   000260R    266#   600*   601    614
TRPDFD= 000022     251#
TSTFE   001332R    478#   481#
TSTRE   001226R    465#
VECTOR  000010R    200#   676
```

```
VIRTAD   000242R           259#   429*    430
VTBFO    002542R           275    719*
VTBF1    003350R           276    720#
VTBF10   010522R           283    728#
VTBF11   011430R           284    729#
VTBF12   012236R           285    729#
VTBF13   013044R           286    730#
VTBF14   013652R           287    731#
VTBF15   014460R           288    733#
VTBF16   015266R           290    734#
VTBF17   016074R           290    734#
VTBF2    004156R           277    721#
VTBF3    004764R           278    722#
VTBF4    005572R           279    723#
VTBF5    006400R           280    724#
VTBF6    007206R           281    725#
VTBF7    010014R           282    726#
WAITIM   000230R           465#
WAIT1M   000252R           263#   391*    413*    487*
WASADR   000104R           234#
WDPR     000116R           241#   389*
WDTO     000114R           240#   388*
XCNT     000240R           258#   404*    405*    406*    407*    408*    409*    410*    411*    543*
XFLAG    000005R           199
XINT     002174R           134#   680
XMIT     002404R           625    638#
XXXX     001744R           594    596#
X00      000334R           292#   397*    398*    425*    432*    443*    458*    468    489*    496     497     552     553
                           555#   586    588*    620     634*    658*    698*    701*
X02      000336R           293#   598    660*    702*    703*
X04      000340R           294#   662*
X06      000342R           295#   702*    664*
X10      000344R           296#   428*    666*
X12      000346R           297#   446*    668*
X14      000350R           298#   670*
X16      000352R           299#   456*    554    672*
YYYY     001756R           598    608    927    942
.    =   017504R           719#   720#   721#   722#   723#   724#   725#   726#   727#   728#   729#   730#   731#
                           732#   733#   734#   735#   782#
```

```
. ABS.   000000      000
         017504      001
```

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

XVTBBO,XVTBBO/SOL/CRF:SYM=DDXCOM,XVTBBO
RUN-TIME: 1 2 .4 SECONDS
RUN-TIME RATIO: 20/5=4.0
CORE USED: 7K (13 PAGES)