

.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.

COPYRIGHT (C) 1976, 1978 DIGITAL EQUIPMENT CORPORATION

1.

ABSTRACT

VTB IS AN IOMODX THAT EXERCISES UP TO FOUR VT20'S (DH11 LINES) IN
VTB IS INTENDED TO BE A DATA HANDLING ROUTINE USED IN CONNECTION
WITH TEST 21 OF MAINDEC-11-D8VIA (PREVIOUSLY LCH SELECTED AND RUNNING
IN THE VT20'S PDP11-05). DATA IS ENTERED AT THIS DATA POINT THEN
AND SET INTO THE CONTINUOUS TRANSMIT MODE AT THIS DATA POINT THEN
RECEIVED AND RETRANSMITTED BY THE VT20 HOST COMPUTER (THIS THEN
DEC/X11 MODULE). ALL LINES SELECTED FOR TEST CAN BE ACTIVATED AND
RUN CONCURRENTLY. ALL TRANSMIT AND RECEIVE ERRORS ARE REPORTED AND
CN THE CONSOLE ITY. 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

CNE PASS OF THE VTB 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, BRI:5, BR2:0, DVC:1
LORR-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
CORRESPOND 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-DBVIA 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 CONTINUOUS
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 REPTRY AFTER
END PASS IS REPORTED FOR THIS DEC/X11 MODULE (DH11 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 THAN 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 RECEIVED CHARACTER
BIT 15 = DATA PRESENT
BIT 14 = OVERFLOW
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 IRRELEVANT.
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
-LIST SEQ,BIN,TOC

301
000354* 000000
000356* 000000
000360* 000000
000364* 000000
000368* 000000
000372* 000000
000376* 000000
000380* 000000
000384* 000000
000388* 000000
000392* 000000
000396* 000000
000400* 000000
000404* 000000
000408* 000000
000412* 000000

TABLE OF BUFFER POINTERS FOR DH11 LINES
L0SM: 000000
L1SM: 000000
L2SM: 000000
L3SM: 000000
L4SM: 000000
L5SM: 000000
L6SM: 000000
L7SM: 000000
L8SM: 000000
L9SM: 000000
L10SM: 000000
L11SM: 000000
L12SM: 000000
L13SM: 000000
L14SM: 000000
L15SM: 000000
L16SM: 000000
L17SM: 000000

000414* 033500
000418* 033500
000422* 033500
000426* 033500
000430* 033500
000434* 033500
000438* 033500
000442* 033500
000446* 033500
000450* 033500
000454* 033500

TABLE OF BAUD RATES FOR DH11 LINES
L0BR: 033500
L1BR: 033500
L2BR: 033500
L3BR: 033500
L4BR: 033500
L5BR: 033500
L6BR: 033500
L7BR: 033500
L8BR: 033500
L9BR: 033500
L10BR: 033500
L11BR: 033500
L12BR: 033500
L13BR: 033500
L14BR: 033500
L15BR: 033500
L16BR: 033500
L17BR: 033500

000454* 000000
000456* 000000
000460* 000000
000464* 000000
000468* 000000
000472* 000000
000476* 000000
000480* 000000
000484* 000000
000488* 000000
000492* 000000
000496* 000000
000500* 000000

EACH OF THE FOLLOWING 16 WORDS CONTAINS THE ERROR AND STATUS INFORMATION FOR DH11 LINE. THE HIGH ORDER BYTE HOLDS THE # OF ERRORS IN THAT LINE AND THE LOW ORDER BYTE CONTAINS THE LINES CURRENT STATUS
L0EC: 0
L1EC: 0
L2EC: 0
L3EC: 0
L4EC: 0
L5EC: 0
L6EC: 0
L7EC: 0
L8EC: 0
L9EC: 0
L10EC: 0
L11EC: 0
L12EC: 0
L13EC: 0
L14EC: 0
L15EC: 0
L16EC: 0
L17EC: 0

000502* 000000
000504* 000000
000506* 000000
000510* 000000
000514* 000000
000518* 000000
000522* 000000
000526* 000000
000530* 000000
000534* 000000
000538* 000000
000542* 000000
000546* 000000
000550* 000000
000554* 000000

TABLE OF CHARACTER COUNT LOCATIONS FOR DH11 LINES
L0CC: 0
L1CC: 0
L2CC: 0
L3CC: 0
L4CC: 0
L5CC: 0
L6CC: 0
L7CC: 0
L8CC: 0
L9CC: 0
L10CC: 0
L11CC: 0
L12CC: 0
L13CC: 0
L14CC: 0
L15CC: 0
L16CC: 0
L17CC: 0

000554* 012767 000004 177336
000556* 012767 000020 177324
000576* 012777 000020 177200
000576* 012777 004000 177200
000604* 012767 000004 177240
000616* 016796 177214
000622* 001155 177172

DETERMINE IF ANY DH'S ARE SELECTED. IF SO SETUP DEVICE REGISTER ADDRESSES, INTERRUPT VECTORS, AND START THE MODULE PROCESSING... IF NOT, DROP THE MODULE FROM THE RUN
START: MOV #4,INTR ;4 INTERRUPTS / ITERATION
MOV #20,NDTO ;20 WORDS TO MEM/ITERATION
MOV #20,WDFR ;20 WORDS FROM MEM/ITERATION
MOV #004400,SADDR ;INIT THE DH11
TEST SP,SP ;MAKE A LONG 1ST PASS TO ALLOW FOR STARTING VT20S
DVID1 ;CHECK FOR SELECTED VT20'S
BNE DVID1 ;ANY SELECTED?
;IS SELECTED FOR TEST (DVID1=0)
;SETUP ADDRESS OF THE DH11 CONTRL/STATUS REG
;NO. PREVENT DH11 INTERRUPTS
;ASCII MESSAGE CALL WITH COMMON HEADER
;DROP THIS MODULE

000624* 016767 177156 177502
000632* 012777 004000 177474
000640* 104403 000000 017424
000646* 104410 000000
000652* 116767 177537 177360
000660* 042767 177174 177352
000666* 005267 177346
000672* 006167 177342
000678* 006167 177336
000709* 006167 177326
000706* 006167 177326
000712* 005467 177322
000716* 005000

THIS IS THE RESTART ROUTINE. IT RESTARTS IN TRANSMIT MODE ALL LINES THAT FINISHED RECEIVING ON THE LAST PASS. ALL OTHER LINES IF IGCNRS RESTRT: MOVB LBR,LCNT ;GET BAUD RATE INFO
BIC #177774,XCNT ;A ROUGH GUESS
ROL XCNT ;AT HOW MANY XFERS WE SHOULD DC PER PASS
ROL XCNT ;MULT XCNT BY 16
ROL XCNT ;TO GET THE NUMBER
ROL XCNT ;OF TRANSFERS TO DO
NEG XCNT ;EACH PASS
CLR XCNT ;MAKE IT NEGATIVE
RO ;START WITH LINE 0

```

413 000720 012767 000003 177324 MOV #3,WAIT1M ;REGULAR PASS TIMEOUT
414 000726 012767 177312 CLR #0 ;INIT WORD WE ARE GOING TO SENT TO X12
415 000733 005067 177312 CLR #0 ;CLEAR OUT COUNT FOR # OF LINES
416 000736 126027 000004 1S: CMPB LOEC(0),#4 ;IS LINE READY TO XMIT?
417 000741 012767 000003 000454* BNE #3,LOEC(0) ;IF NOT, GO SEE ABOUT NEXT LINE
418 000744 012767 000003 000454* SEC ;IF IT IS READY, SET IT TO ACTIVE MODE
419 000750 006067 177274 ROR #1 ;RETATE I BIT AND THE NEW "BAR" REGISTER
420 000756 006067 177274 ROR #1 ;RETATE I BIT AND THE NEW "BAR" REGISTER
421 000762 012767 177274 DEC DUNCT ;ADD 1 TO LINE COUNT
422 000768 012767 177274 MOV #0,TEMP ;GRAB THE LINE #
423 000777 006067 177250 BIC #177760,TEMP ;DIVIDE IT BY 2 TO MAKE IT REAL
424 000783 012767 177250 MOV #0,TEMP ;MAKE SURE ONLY THE STATUS REGISTER SET
425 000790 012767 177234 MOV #0,TEMP ;PUT IT INTO THE STATUS REGISTER
426 000796 042767 177230 BIC #0,TEMP ;NOW GET THE CHARACTER COUNT
427 000803 005467 177230 NEG #0,TEMP ;USE ITS 2'S COMPLIMENT
428 000809 012767 177224 MOV #0,TEMP ;AND SETUP THE BYTE COUNT REGISTER WITH IT
429 000816 012767 177224 MOV #0,TEMP ;GET VIRTUAL ADDRESS OF THE BUFFER
430 000823 012767 177202 GETPAS,BEGIN,VIRTAD ;GET PHYSICAL ADDRESS FROM 16-BIT VIRTAD
431 000830 012767 177202 BIC #17777,EXTAD ;CLEAR ALL BUT THE EXTEND BITS
432 000836 012767 177172 MOV #0,TEMP ;SET ADDRESS EXTEND BITS IN STATUS REG
433 000843 012767 177172 MOV #0,TEMP ;SETUP CURRENT ADDRESS FOR THIS LINE
434 000850 012767 177152 MOV #0,TEMP ;AND REINIT ITS CHARACTER COUNT
435 000856 012767 177152 CLR #0,TEMP ;CLEAR BIT TO
436 000863 000231 2S: CLC ;MAKE SURE THAT THIS LINES "BAR" REGISTER BIT IS CLEAR
437 000870 012767 000002 ROR #0 ;GO ON TO THE NEXT LINE
438 000876 012767 000004 3S: ADD #0,#4 ;HAVE WE DONE ALL LINES YET?
439 000883 012767 177134 BNE #15 ;NO, GO TO NEXT LINE
440 000890 012767 177134 TST #0,TEMP ;ALL DONE, ARE ANY LINES STILL ACTIVE?
441 000896 012767 177134 BNE #0,TEMP ;IF SO, GO START EM UP XMITTING
442 000903 012767 177202 MSGS,BEGIN,ERRNH ;RESTARTS AGAIN, STOP THE DH11
443 000910 012767 177172 ENDS,BEGIN ;AND DROP THIS MODULE WITH COMMON HEADER
444 000916 012767 177174 4S: MOV #0,TEMP ;SET THE BAR REGISTER
445 000923 012767 177152 MOV #0,TEMP ;GO WAIT FOR SOMETHING TO HAPPEN
446 000930 012767 177152 WODROP: JSR PC,SETUP ;DO A BUNCH OF INITIALIZATION
447 000936 012767 177152
448 000943 012767 177152
449 000950 012767 177152
450 000956 012767 177152
451 000963 012767 177152 ;THIS IS THE MAIN PROGRAM LOOP
452 000970 012767 177152 ;NOTHING IS DOWN HERE EXCEPT TO CONTINUOUSLY TEST TO SEE IF ENOUGH TIME HAS PASSED
453 000976 012767 177152 ;FOR AN END OF PASS, CHECK FOR ERRORS AND CALL THE
454 000983 012767 177152 ;INTERRUPT SERVICE ROUTINE TO CLEAN OUT THE SILO IN CASE IT HAS STUFF
455 000990 012767 177152 ;IN IT, BUT NOT ENOUGH STUFF TO CAUSE AN INTERRUPT
456 001162 012777 000040 177162 MAIN: MOV #40,EX16 ;SET SILO TO INTERRUPT ON 40(OCTAL) CHARS
457 001174 012777 036536 MOV #0,SP ;MAKE SURE THAT THE STACK POINTER IS RIGHT
458 001181 012777 177132 MLOOP: MOV #030100,EX00 ;ENABLE INTERRUPTS
459 001206 012746 000000 MOV #0,-(SP) ;PUT FAKE PS ONTO STACK
460 001206 012746 001216* MOV #0,INTSRV ;PRETEND THAT WE JUST HAD AN INTERRUPT
461 001216 000167 000464 JMP INTSRV
462 001216 104407 000000* BRAKE: BREAKS,BEGIN ;TEMPORARY RETURN TO MONITOR...
463 001222 104407 000000* BREAKS,BEGIN ;THEN CONTINUE AT NEXT INSTRUCTION.
464 001228 104407 177036 TSTRE: CMP #0,ERRPT1,ERRPT2 ;ANY RECEIVE ERRORS?
465 001234 012767 177036 BEQ #0,ERRPT1,ERRPT2 ;IF NOT GO CHECK FOR FATAL ERRORS
466 001236 016701 177026 MOV #0,ERRPT1,R1 ;IF SO, MAKE THE ERROR QUEUE PCITER MORE ACCESSABLE
467 001242 016767 177066 MOV #0,ERRPT1,R1 ;SETUP THE CONTROL REG ADDRESS FOR PRINTING
468 001242 016767 177066
  
```

```

469 001250 012167 176626 MOV #0,ACSR ;SETUP THE CONTROL REG CONTENTS FOR PRINTING
470 001254 012167 176626 MSGS,BEGIN,ERRSTAT ;SETUP THE STATUS REG CONTENTS FOR PRINTING
471 001260 104403 000000* 017464* MOV #0,ERRMSG ;RECEIVE MESSAGE CALL WITH COMMON HEADER
472 001266 012767 000017 176612 MOV #17,ERRTYP ;RECEIVE ERROR
473 001274 104405 000000* 000000 ;*****
474 001302 012167 176762 HDRS$,BEGIN,NULL ;RECEIVE ERROR - NON FATAL
475 001302 012167 176756 BNE #0,ERRPT1,ERRPT2 ;*****
476 001314 001006 MOV #0,ERRPT1,ERRPT2 ;UPDATE THE ERROR QUEUE POINTER
477 001316 012767 016702* 176744 1S: MOV #0,ERRPT1,ERRPT2 ;IS THE ERROR QUEUE EMPTY?
478 001324 012767 016702* 176740 TSTFE: MOV #0,ERRPT1,ERRPT2 ;IF IT IS NOT EMPTY, IGNORE IT FOR A MOMENT
479 001336 001132 TSTFE: TST #0,ERRPT1,ERRPT2 ;ERROR QUEUE IS EMPTY. REINITIALIZE THE
480 001344 005767 176704 BNE #0,ERRPT1,ERRPT2 ;ERROR QUEUE POINTER
481 001344 005767 176704 BEQ #0,ERRPT1,ERRPT2 ;HAVE WE HAD ANY FATAL ERRORS?
482 001344 005767 176656 BEQ #0,ERRPT1,ERRPT2 ;IF SO, GO REPORT IT AND DROP THIS MODULE
483 001352 005767 176656 INC #0,ERRPT1,ERRPT2 ;HAVE ALL ACTIVE LINES FINISHED RECEIVING?
484 001352 005767 176656 BEQ #0,ERRPT1,ERRPT2 ;IF SO GO CLEAN UP AND END THIS PASS
485 001354 005367 176672 MOV #0,ERRPT1,ERRPT2 ;NO, TICK GOES THE WAIT TIMER
486 001360 005367 176672 DEC #0,ERRPT1,ERRPT2 ;HAVE THESE DEVICES HAD ENOUGH TIME TO FINISH?
487 001360 005367 176672 BNE #0,ERRPT1,ERRPT2 ;HAD ENOUGH TIME TO FINISH?
488 001370 005000 030100 176744 ALDONE: BIC #30100,EX00 ;IF NOT, TIME! PREVENT FURTHER DH11 INTERRUPTS
489 001372 105760 000454* 1S: CLR #0 ;SETUP A LINE INDEX REGISTER TO 0
490 001376 001442 MOV #0,LOEC(0) ;FIND OUT IF THE LINE IS SELECTED
491 001382 126027 000454* 000004 BEQ #0,LOEC(0) ;IF NOT GO TEST FOR ERRORS ON IT
492 001406 012767 000000* 000454* CMPB #0,LOEC(0),#4 ;IF IT IS CHECK THAT IT HAS FINISHED RECEIVING
493 001410 012767 000000* 000454* BEQ #0,LOEC(0) ;IF IT DID AT LEAST 1 TRANSFER, DONT PRINT AN ERROR
494 001416 012767 176712 176454 MOV #0,CSRA ;DIDNT FINISH, SET THE LINE TO DROPPED MODE
495 001422 012767 176712 176454 MOV #0,ACSR ;SETUP FOR PRINTING THE ADDRESS OF THE CONTROL/STATUS REG
496 001428 012767 176712 176454 MOV #0,TEMP,ASTAT ;SETUP FOR PRINTING THE CONTENTS OF THE CONTROL/STATUS REG
497 001434 012767 176712 176454 MOV #0,NUMBA1 ;SAVE IT
498 001440 010067 001070 ROR #0,NUMBA1 ;SAVE IT
499 001444 006067 001064 ROR #0,NUMBA1 ;SAVE IT
500 001450 104420 000000* 002534* ;*****
501 001456 017403 OTOAS,BEGIN,NUMBA1,MESNM0 ;CCNVERT NUMBA1 TO ASCII AND STORE AT MESNM0
502 001460 104403 000000* 017440* ;*****
503 001466 012767 000023 176412 MSGS,BEGIN,ERRDH ;ASCII MESSAGE CALL WITH COMMON HEADER
504 001474 104405 000000* 000000 MOV #23,ERRTYP ;RECEIVE ERROR
505 001482 104405 000000* 000000 ;*****
506 001490 000434 HDRS$,BEGIN,NULL ;A DH11 LINE HUNG
507 001502 000434 BR #3 ;GO TRY THE NEXT LINE FOR ERRORS
508 001504 105760 000455* 2S: TSTB LOEC+1(0) ;ANY ERRORS IN THIS UNSELECTED LINE?
509 001510 001431 BEQ #0,-(SP) ;IF NOT TRY THE NEXT LINE
510 001512 001046 MOV #0,-(SP) ;PUT THE LINE # INDEX ONTO THE STACK
511 001514 006016 ROR #0,-(SP) ;DIVIDE BY 2 TO GET THE LINE #
512 001516 005046 CLR #0,-(SP) ;CLEAR OUT THE FIRST WORD ON THE STACK
513 001520 116016 MOV #0,LOEC+1(0),SP ;SO THAT WE CAN PUT THE # OF ERRORS INTO LOW BYTE
514 001524 011687 001006 ;*****
515 001530 104420 000000* 002536* ;*****
516 001536 104420 000000* 002536* OTOAS,BEGIN,NUMBA2,MESNM0 ;CCNVERT NUMBA2 TO ASCII AND STORE AT MESNM0
  
```



```

637      002204- 112760 000004- 000454- }THIS ROUTINE IS ENTERED WHEN A LINE FINISHES RECEIVING
638      002210- 168060 000514- 000354- }AMIT:  MOV# #4,LOEC(0) }SET LINE TO "RECEIVED DONE" MODE
639      002217- 168060 000514- 000354- }SUB   LOCC(0),LOSW(0) }RESET THE LINES BUFFER POINTER
640      002220- 001252 178034- }DUMCN }ALSO TALLY UP 1 MORE FINISHED DEVICE
641      002230- 001252 178034- }DNC   }TO CLEAR FULL-DUPLEX MODE
642      002230- 001252 178034- }BNE   YVVY }IF NOT GO BACK AND DO ANOTHER
643      002232- 016700 175624- }MOV   SVR0,RO }YES, DONE 4 ALREADY. RESTORE RO
644      002236- 016700 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 }SETUP: CLR  FERRN }INIT FATAL ERROR SWITCH = NO FATAL ERRORS
657      MOV  ADDR,RO }MAKE THE DEVICE REGISTER ADDRESS EASY TO ACCESS
658      MOV  RO,R00 }SETUP ADDRESS OF THE SYSTEM CONTROL REGISTER
659      TST  (R0)+ }TRAP IF IT DOES NOT EXIST
660      MOV  RO,R02 }SETUP ADDRESS OF NEXT CHARACTER RECEIVED REGISTER
661      TST  (R0)+ }TRAP IF IT DOES NOT EXIST
662      MOV  RO,R04 }SETUP ADDRESS OF LINE PARAMETER REGISTER
663      TST  (R0)+ }TRAP IF IT DOES NOT EXIST
664      MOV  RO,R06 }SETUP ADDRESS OF CURRENT ADDRESS REGISTER
665      TST  (R0)+ }TRAP IF IT DOES NOT EXIST
666      MOV  RO,R10 }SETUP ADDRESS OF BYE COUNT REGISTER
667      TST  (R0)+ }TRAP IF IT DOES NOT EXIST
668      MOV  RO,R12 }SETUP ADDRESS OF BUFFER ACTIVE REGISTER
669      TST  (R0)+ }TRAP IF IT DOES NOT EXIST
670      MOV  RO,R14 }SETUP ADDRESS OF BREAK CONTROL REGISTER
671      TST  (R0)+ }TRAP IF IT DOES NOT EXIST
672      MOV  RO,R16 }SETUP THE ADDRESS OF THE SILO STATUS REGISTER
673      TST  (R0)+ }TRAP IF IT DOES NOT EXIST
674      MOV  R0,R0 }THERE ARE 2 OF THEM
675 }NOW SETUP DH11 INTERRUPT VECTORS
676 }FOR RECEIVER INTERRUPTS AND 1 FOR TRANSMITTER INTERRUPTS
677      MOV  VECTOR,R0 }GET THE VECTOR ADDRESS
678      MOV# B1,(R0)+ }SET VECTOR TO RECEIVER INTERRUPT SERVICE
679      CLRB (R0)+ }SET PRIORITY LEVEL FOR THE INTERRUPT ROUTINE
680      MOV  R0,R0 }SET RO TO AN EVEN ADDRESS
681      MOV  R0,R0 }POINT VECTOR AT TRANSMITTER SERVICE ROUTINE
682      CLRB (R0)+ }PRIORITY LEVEL FOR THE TRANSMITTER INTERRUPT ROUTINE
683      MOV  R0,R0 }SET RO TO AN EVEN ADDRESS
684 }NOW SETUP ERROR QUEUE POINTERS
685      MOV  #ERQUEU,ERPT1 }SET THE "WHERE WE'VE SERVICED TO" ERROR QUEUE
686      MOV  #ERQUEU,ERPT2 }SET THE "WHERE BEGINNING OF THE QUEUE AREA
687      MOV  #ERQUEU,ERPT2 }SET THE "WHERE WE PUT STUFF" ERROR QUEUE
688      MOV  #ERQUEU,ERPT2 }POINTER TO THE BEGINNING OF THE QUEUE AREA
689 }NOW ASSIGN BUFFER POINTERS TO SELECTED LINES (DONE ONLY ONCE AT THE BEGINNING)
690      CLR  DUMCN }INIT DEVICE SELECT COUNT TO 0 BEFORE COUNTING VT20'S
691      MOV  DVID1,R1 }MAKE BITMAP OF SELECTED VT20 LINES MORE ACCESSABLE
692      CLR  RO }INIT FOR 16 PCISIBLE LINES

```

```

693      002414- 005067 175636- }
694      002420- 005060 000454- }
695      002424- 006001 }
696      002426- 103031 }
697      002430- 018959 }
698      002430- 018959 }
699      002444- 005367 175600- }
700      002450- 000440 }
701      002460- 175600 175654- }
702      002460- 175600 175654- }
703      002466- 052777 000023- }
704      002474- 005060 000514- }
705      002480- 018959 000803- }
706      002506- 018959 000000- }
707      002512- 062700 000002- }
708      002516- 095267 175634- }
709      002522- 026427 175630- }
710      002530- 081333 000017- }
711      002532- 000207 }
712
713 }
714 }
715      002534- 000000 }
716      002536- 000000 }
717      002540- 000000 }
718 }
719      002542- 000303 }
720      003350- 000303 }
721      004156- 000303 }
722      004784- 000303 }
723      005572- 000303 }
724      006400- 000303 }
725      007206- 000303 }
726      010014- 000303 }
727      010622- 000303 }
728      011430- 000303 }
729      013024- 000303 }
730      013652- 000303 }
731      014460- 000303 }
732      016096- 000303 }
733      016702- 000074 }
734      017072- 000000 }
735      017072- 000074 }
736      017072- 000000 }
737 }
738 }
739      017074- 047516 046040 047111 }
740      017102- 051505 051040 046505 }
741      017110- 044504 020116 041501 }
742      017118- 044524 042526 000 }
743      017123- 040 047516 053040 }
744      017130- 031124 023460 020123 }
745      017149- 042162 042514 052103 }
746      017149- 042162 000 }
747      017147- 040 044514 042516 }
748      017154- 000040 }

```

```

15: CLR  TEMPO }INIT THE LINE #
    LOCC(0) }RESUME LINE IS UNSELECTED UNTIL PROVEN OTHERWISE
    ROR  R1 }PUT A BIT INTO THE C-BIT
    BCC  2$ }IS THE LINE TO BE USED?
    MOV  BPLIST(0),LOSW(0) }YES, ASSIGN A BUFFER POINTER TO THAT LINE
    R0,R0 }CLEAR OUT LINE SELECTION BITS
    DUMCN }ADD 1 TO COUNT OF ACTIVE DEVICES
    NOP
    BISB TEMPO,#X00 }SET LINE SELECTION BITS FOR THIS LINE
    LOBR(0),#X04 }SETUP BAUD RATES FOR THIS LINE
    BIS  #25,#X04 }PARTY-FULL DUPLEX-8 BIT
    CLR  LOCC(0) }CLEAR OUT THIS LINES CHARACTER COUNT
    INC  #3,LOCC(0) }START OUT THE LINE IN RECEIVE MODE
    INC  #2,R0 }UPDATE COUNT OF SELECTED VT20'S
    ADD  #2,R0 }GO ON TO THE NEXT LINE
    INC  TEMPO }NEXT LINE
    CMP  TEMPO,#17 }CHECKED ALL LINES FOR SELECTION YET?
    BNE  1$ }IF NOT, GO BACK AND CHECK THE NEXT ONE
    RTS  1$ }RETURN

```

```

NUMBA1: -WORD 0
NUMBA2: -WORD 0
NUMBA3: -WORD 0

```

```

VTBF0: -BLKW 195- }BUFFER SPACE FOR 1ST SELECTED DH11 LINE
VTBF1: -BLKW 195- }BUFFER SPACE FOR 2ND SELECTED DH11 LINE
VTBF2: -BLKW 195- }BUFFER SPACE FOR 3RD SELECTED DH11 LINE
VTBF3: -BLKW 195- }BUFFER SPACE FOR 4TH SELECTED DH11 LINE
VTBF4: -BLKW 195- }BUFFER SPACE FOR A SELECTED DH11 LINE
VTBF5: -BLKW 195- }BUFFER SPACE FOR A SELECTED DH11 LINE
VTBF6: -BLKW 195- }BUFFER SPACE FOR A SELECTED DH11 LINE
VTBF7: -BLKW 195- }BUFFER SPACE FOR A SELECTED DH11 LINE
VTBF8: -BLKW 195- }BUFFER SPACE FOR A SELECTED DH11 LINE
VTBF9: -BLKW 195- }BUFFER SPACE FOR A SELECTED DH11 LINE
VTBF10: -BLKW 195- }BUFFER SPACE FOR A SELECTED DH11 LINE
VTBF11: -BLKW 195- }BUFFER SPACE FOR A SELECTED DH11 LINE
VTBF12: -BLKW 195- }BUFFER SPACE FOR A SELECTED DH11 LINE
VTBF13: -BLKW 195- }BUFFER SPACE FOR A SELECTED DH11 LINE
VTBF14: -BLKW 195- }BUFFER SPACE FOR A SELECTED DH11 LINE
VTBF15: -BLKW 195- }BUFFER SPACE FOR A SELECTED DH11 LINE
VTBF16: -BLKW 195- }BUFFER SPACE FOR A SELECTED DH11 LINE
VTBF17: -BLKW 195- }BUFFER SPACE FOR A SELECTED DH11 LINE
ERQDEU: -BLKW 60- }RECEIVER ERROR QUEUE SPACE
ERQOFL: -WORD 0- }RECEIVER ERROR QUEUE OVERFLOW BOUNDARY

```

```

}FOLLOWING IS THE TEXT FOR ALL ASCII ERROR MESSAGES
MESNR: -ASCIIZ /NO LINES REMAIN ACTIVE/
MESNS: -ASCIIZ / NO VT20'S SELECTED/
MESDH: -ASCIIZ / LINE /

```

749 017156 042040 042111 047040
 750 017164 052117 041440 046517
 751 017172 046120 052105 020105
 752 017200 020061 052105 047101
 753 017206 043123 051105 044440
 754 017214 020116 044524 042515
 755 017222 000 000 000
 756 017230 040240 042522 042040
 757 017236 047520 044520 043516
 758 017244 052040 042510 046040
 759 017252 047111 000105 000
 760 017256 042440 051122 051117
 761 017264 020122 047117 052440
 762 017272 050116 046105 041505
 763 017300 042524 020104 044514
 764 017306 042516 000 000
 765 017314 040 044524 047514
 766 017322 047514 042522 043122
 767 017330 047514 042522 043122
 768 017336 044440 051524 043040
 769 017344 052100 046101 000
 770 017352 042503 051101 042522
 771 017360 042440 051122 051117
 772 017366 041040 052111 053440
 773 017374 051505 051440 052105
 774 017402 000 000 000
 775 017408 045 000 000
 776 017414 060 030060 030060
 777 017420 000 030060 030060
 778 017426 000 030060 030060
 779 017432 000 030060 030060
 780 017438 000 030060 030060
 781 017444 000 030060 030060
 782 017450 000 030060 030060
 783 017456 000 030060 030060
 784 017462 000 030060 030060
 785 017468 000 030060 030060
 786 017474 000 030060 030060
 787 017480 000 030060 030060
 788 017486 000 030060 030060
 789 017492 000 030060 030060
 790 017498 000 030060 030060
 791 017504 000 030060 030060
 792 017510 000 030060 030060
 793 017516 000 030060 030060
 794 017522 000 030060 030060
 795 017528 000 030060 030060
 796 017534 000 030060 030060
 797 017540 000 030060 030060
 798 017546 000 030060 030060
 799 017552 000 030060 030060
 800 017558 000 030060 030060
 801 017564 000 030060 030060
 802 017570 000 030060 030060
 803 017576 000 030060 030060
 804 017582 000 030060 030060

MESDH1: .ASCIZ / DID NOT COMPLETE 1 TRANSFER IN TIME/
 MESDH2: .ASCIZ /SO WE ARE DROPPING THE LINE/
 MESUS: .ASCIZ / ERRORS ON UNSELECTED LINE/
 MESSO: .ASCIZ / SILO OVERFLOW... ITS FATAL/
 MESRE: .ASCIZ / A RECEIVER ERROR BIT WAS SET/
 MESBE: .ASCIZ /%/
 MESNM0: .ASCIZ /000000/
 MESNM1: .ASCIZ /000000/
 -EVEN

;IN CASE THERE IS AN ODD # OF BYTES IN THE ABOVE MESSAGE
 ;TABLE OF ERROR MESSAGE POINTERS & TERMINATORS (COMMENTED!)
 ERRNS: MESBE ;POINTS TO A CARRIAGE RETURN-LINE FEED
 ;POINTS TO AN ASCIZ ERROR MESSAGE
 ERRNR: 17777 ;TERMINATOR, ONLY 1 MESSAGE THIS TIME
 ;POINTS TO A CARRIAGE RETURN-LINE FEED
 ;POINTS TO "NO LINES REMAIN ACTIVE" MESSAGE
 ERRDH: 17777 ;TERMINATE MESSAGE
 ;POINTS TO A CARRIAGE RETURN-LINE FEED
 ;POINTS TO 1ST PART OF "HUNG" ERROR MESSAGE ASCIZ
 MESDH1 ;POINTS TO LINE # ASCIZ
 ;POINTS TO PART OF MESSAGE
 ;RETURN
 ;DROPPING A LINE MESSAGE
 ;END OF MESSAGE
 ERRSO: MESBE ;POINTS TO A CARRIAGE RETURN-LINE FEED
 ;POINTS TO ASCIZ TEXT OF FATAL ERROR MESSAGE
 ;THIS INDICATES THAT NO MORE MESSAGES FOLLOW
 ;POINTS TO A CARRIAGE RETURN-LINE FEED
 ERRRE: 17777 ;POINTS TO ASCIZ TEXT OF THE RECEIVER ERROR MESSAGE
 ;TERMINATOR
 ERRUS: MESBE ;POINTS TO A CARRIAGE RETURN-LINE FEED

805 017474 017405
 806 017476 017256
 807 017500 017474
 808 017502 177777
 809 17777
 810 000001

MESNM0 ;POINTS TO ASCIZ FOR # OF ERRORS
 MESUS ;POINTS TO TEXT PART OF MESSAGE
 MESNM1 ;POINTS TO ASCIZ FOR LINE #
 ;END OF MESSAGE TABLE
 ;END ;SIGNALS END OF MESSAGE TO DECX/11

| | | | | | | | | | | | | | | | | | | | | |
|--------|---------|------|------|------|------|------|------|------|------|------|------|------|------|-----|--|--|--|--|--|--|
| VIRAD | 000242R | 259# | 429* | | | | | | | | | | | | | | | | | |
| VTBF0 | 002542R | 475 | 719* | 430 | | | | | | | | | | | | | | | | |
| VTBF1 | 003350R | 276 | 720* | | | | | | | | | | | | | | | | | |
| VTBF10 | 010627R | 483 | 721* | | | | | | | | | | | | | | | | | |
| VTBF11 | 011430R | 483 | 722* | | | | | | | | | | | | | | | | | |
| VTBF12 | 012236R | 485 | 723* | | | | | | | | | | | | | | | | | |
| VTBF13 | 013044R | 286 | 730* | | | | | | | | | | | | | | | | | |
| VTBF14 | 013852R | 488 | 731* | | | | | | | | | | | | | | | | | |
| VTBF15 | 014660R | 488 | 732* | | | | | | | | | | | | | | | | | |
| VTBF16 | 015266R | 489 | 733* | | | | | | | | | | | | | | | | | |
| VTBF17 | 016074R | 290 | 734* | | | | | | | | | | | | | | | | | |
| VTBF2 | 04196R | 277 | 721* | | | | | | | | | | | | | | | | | |
| VTBF3 | 004766R | 478 | 722* | | | | | | | | | | | | | | | | | |
| VTBF4 | 005572R | 478 | 723* | | | | | | | | | | | | | | | | | |
| VTBF5 | 006400R | 280 | 724* | | | | | | | | | | | | | | | | | |
| VTBF6 | 007206R | 281 | 725* | | | | | | | | | | | | | | | | | |
| VTBF7 | 010014R | 281 | 726* | | | | | | | | | | | | | | | | | |
| WATTIM | 000230R | 484 | 465* | | | | | | | | | | | | | | | | | |
| WATTIM | 000252R | 263 | 391* | 413* | 487* | | | | | | | | | | | | | | | |
| WASADR | 000104R | 234 | 391* | | | | | | | | | | | | | | | | | |
| WDFR | 000116R | 33 | 389* | | | | | | | | | | | | | | | | | |
| WDT0 | 000114R | 40 | 388* | | | | | | | | | | | | | | | | | |
| XCNT | 000240R | 258 | 404* | 405* | 406* | 407* | 408* | 409* | 410* | 411* | 543* | | | | | | | | | |
| XFLAG | 000005R | 134 | 398* | | | | | | | | | | | | | | | | | |
| XINT | 001174R | 634 | 680 | | | | | | | | | | | | | | | | | |
| XMIT | 002204R | 635 | 638 | | | | | | | | | | | | | | | | | |
| XXXX | 001744R | 634 | 596 | | | | | | | | | | | | | | | | | |
| X00 | 000334R | 422 | 594 | | | | | | | | | | | | | | | | | |
| X02 | 000336R | 293 | 397 | 398* | 425* | 432* | 443* | 458* | 466 | 489* | 496 | 497 | 552 | 553 | | | | | | |
| X04 | 000340R | 293 | 398 | 398* | 620 | 634* | 658* | 698* | 701* | | | | | | | | | | | |
| X06 | 000342R | 294 | 586 | 398* | | | | | | | | | | | | | | | | |
| X10 | 000344R | 295 | 598 | 398* | 703* | | | | | | | | | | | | | | | |
| X12 | 000346R | 297 | 662* | 398* | | | | | | | | | | | | | | | | |
| X14 | 000350R | 298 | 433* | 398* | | | | | | | | | | | | | | | | |
| X16 | 000352R | 299 | 429* | 398* | | | | | | | | | | | | | | | | |
| YVYV | 001756R | 698 | 670* | | | | | | | | | | | | | | | | | |
| . | 017504R | 719 | 554 | 554 | 672* | | | | | | | | | | | | | | | |
| | | 732 | 527 | 527 | 642 | | | | | | | | | | | | | | | |
| | | 733 | 720 | 720 | 723* | 724* | 725* | 726* | 727* | 728* | 729* | 730* | 731* | | | | | | | |
| | | | 734 | 734 | 735* | 782* | | | | | | | | | | | | | | |

. ABS. 000000 000
 017504 001

ERRORS DETECTED: 0
 DEFAULT GLOBALS GENERATED: 0
 XVTBBO,XVTBBO/SOL/CRF:SYM=DDXCON,XVTBBO
 RUN-TIME: 1 2 .4 SECONDS
 RUN-TIME RATIO: 20/5=4.0
 CORE USED: 7K (13 PAGES)