```
.REM _
```

IDENTIFICATION

PRODUCT CODE: AC-E836E-MC

PRODUCT NAME: CXCBBE0 CB11 DISTRIBUTE MOD

PRODUCT DATE: SEPTEMBER 1978

MAINTAINER: DEC/X11 SUPPORT GROUP

1. ABSTRACT

CBB IS A BKMOD THAT EXERCISES UP TO "N" CB11 DISTRIBUTOR MODULES HAVING CONTIGUOUS UNIBUS ADDRESSES. THE MAXIMUM VALUE IF "N" IS THE SAME AS THE MAXIMUM NO. OF DISTRIBUTORS MODULES ALLOWED FOR A SINGLE CB11. NON-CONTIGUOUS GROUPS OF DISTRIBUTOR MODULES MAY BE EXERCISED BY CONFIGURATING THE CBB MODULE FOR EACH GROUP. THE MODULE SIMPLY TESTS THE ABILITY TO SET AND CLEAR ALL BITS IN ALL AVAILABLE DISTRIBUTE REGISTERS SELECTED FOR TEST. IF ANY BIT FAILS TO SET OR CLEAR PROPERLY THE ERROR IS REPORTED ON THE CONSOLE TTY.

2. REQUIREMENTS:

HARDWARE: A CB11 INTERFACE WITH AT LEAST ONE DISTRIBUTE MODULE.

STORAGE:: CBB REQUIRES:
1. DECIMAL WORDS: 126
2. OCTAL WORDS: 0176
3. OCTAL BYTES: 374

3. PASS DEFINITION

ONE PASS OF THE CBB MODULE RESULTS IN 100. ITERATIONS OF THE BASIC TEST SEQUENCE WHICH CLEARS AND SETS ALL DISTRIBUTOR REGISTERS SELECTED FOR TEST

4. EXECUTION TIME

CBB RUNNING ALONE ON A PDP 11/05 SYSTEM WITH ONE DISTRIBUTE MODULE SELECTED TAKES APPROXIMATELY 10 SECONDS.

5. CONFIGURATION PARAMETERS

DEFAULT PARAMETERS:

DVADR: 0,VECTOR:0,BR1:0,BR2:0,DVECNT:1,SR1:0

REQUIRED PARAMETERS:

FOR EACH COPY OF CBB CONFIGURED THE
USER MUST SPECIFY THE FOLLOWING PARAMETERS:

DEVADR:	EQUAL TO THE FIRST ADDRESS IN A CONTIGUOUS GROUP
SR1:	NUMBER OF MODULES (SEE "OPERATION OPTIONS")

6. DEVICE OPTION SETUP:

NONE REQUIRED

7. MODULE OPERATION

TEST SEQUENCE.

A. SET UP THE PASS COUNTER FOR 100. ITERATIONS
B. GET THE CONTENTS OF SR1 TO FIND OUT HOW
   MANY REGISTERS TO TEST
C. CLEAR A DISTRIBUTE REGISTER
D. COUNT IT
E. REPORT ANY ERROR
F. GENERATE NEXT ADDRESS
G. REPEAT B-F UNTIL ALL REGISTERS TESTED
   FOR ALL ZEROES
H. REPEAT B-G BUT SET ALL REGISTERS TO
   ALL ONES AND TEST
I. COUNT ONE ITERATION
J. IF NOT 100. REPEAT B-I
K. REPEAT END PAS,RESTART AT A.

8. OPERATION OPTIONS

A. USER CAN MODIFY "ADDR" AND "SR1" TO
   SELECT ANY GROUP OF DISTRIBUTE MODULES

SR1:	THE NUMBER (OCTAL) OF DISTRIBUTE MODULES TO BE TESTED (MUST BE CONTIGUOUS). THIS NUMBER MUST BE A STRAIGHT OCTAL NUMBER, NOT A BIT MAP AS IS NORMALLY USED IN DEVCNT. DEVCNT WAS NOT USED BECAUSE IT CAN HOLD A MAXIMUN OF 16 DEVICES AND CBB CAN RUN UP TO 256 DEVICES. IF SR1 IS LEFT AT ZERO OR OTHERWISE IMPROPERLY SET UP, THE SYSTEM WILL EVENTUALLY CRASH.

9. NON-STANDARD PRINTOUTS

NONE-ALL PRINTOUTS HAVE THE STANDARD
DEC/X11 FORMATS.

```
                        ;CB11 DISTRIBUTE MODULE - DEC/X11 EXERCISER MODULE

000000'                 BKMOD   <CBBE >,,,100.,34
000000'                 MODULE  40020,CBBE,,,,100.,,34
                        .TITLE  CBBE DEC/X11 SYSTEM EXERCISER MODULE
                ;       DOXCOM  VERSION  6      23-MAY-78
                                .LIST   BIN
                        ;*****************************************************************************
000000'                 BEGIN:
000000' 041103  042502      040 MODNAM: .ASCII  /CBBE / ;MODULE NAME.
000005'     000 XFLAG:  .BYTE   OPEN            ;USED TO KEEP TRACK OF WBUFF USAGE
000006' 000000 ADDR:    +0                      ;1ST DEVICE ADDR.
000010' 000000 VECTOR:  +0                      ;1ST DEVICE VECTOR.
000012'     000 BR1:    .BYTE   PRTY+0          ;1ST BR LEVEL.
000013'     000 BR2:    .BYTE   PRTY+0          ;2ND BR LEVEL.
000014' 000001 DVID1:   +1                      ;DEVICE INDICATOR 1.
000016' 000000 SR1:     OPEN                    ;SWITCH REGISTER 1
000020' 000000 SR2:     OPEN                    ;SWITCH REGISTER 2
000022' 000000 SR3:     OPEN                    ;SWITCH REGISTER 3
000024' 000000 SR4:     OPEN                    ;SWITCH REGISTER 4
                        ;*****************************************************************************
000026' 040020 STAT:    40020                   ;STATUS WORD.
000030' 000224' INIT:   START                   ;MODULE START ADDR.
000032' 000224' SPOINT: MODSP                   ;MODULE STACK POINTER.
000034' 000000 PASCNT:  0                       ;PASS COUNTER.
000036' 000144 ICONT:   100.                    ;# OF ITERATIONS PER PASS=100.
000040' 000000 ICOUNT:  0                       ;LOC TO COUNT ITERATIONS
000042' 000000 SOFCNT:  0                       ;LOC TO SAVE TOTAL SOFT ERRORS
000044' 000000 HRDCNT:  0                       ;LOC TO SAVE TOTAL HARD ERRORS
000046' 000000 SOFPAS:  0                       ;LOC TO SAVE SOFT ERRORS PER PASS
000050' 000000 HRDPAS:  0                       ;LOC TO SAVE HARD ERRORS PER PASS
000052' 000000 SYSCNT:  0                       ;# OF SYS ERRORS ACCUMULATED
000054' 000000 RANNUM:  0                       ;HOLDS RANDOM # WHEN RAND MACRO IS CALLED
000056'                 CONFIG:
000056' 000000 RES1:    0                       ;RESERVED FOR MONITOR USE
000060' 000000 RES2:    0                       ;RESERVED FOR MONITOR USE
000062' 000000 SVR0:    OPEN                    ;RESERVED FOR MONITOR USE
000064' 000000 SVR1:    OPEN                    ;LOC TO SAVE R0.
000066' 000000 SVR2:    OPEN                    ;LOC TO SAVE R1.
000070' 000000 SVR3:    OPEN                    ;LOC TO SAVE R2.
000072' 000000 SVR4:    OPEN                    ;LOC TO SAVE R3.
000074' 000000 SVR5:    OPEN                    ;LOC TO SAVE R4.
000076' 000000 SVR6:    OPEN                    ;LOC TO SAVE R5.
000100' 000000 CSRA:    OPEN                    ;LOC TO SAVE R6.
000102'                 SBADR:                  ;ADDR OF CURRENT CSR.
000102' 000000 ACSR:    OPEN                    ;ADDR OF GOOD DATA, OR
000104'                 WASADR:                 ;CONTENTS OF CSR.
000104' 000000 ASTAT:   OPEN                    ;ADDR OF BAD DATA, OR
000106'                 ERRTYP:                 ;STATUS REG CONTENTS.
000106' 000000 ASB:     OPEN                    ;TYPE OF ERROR
000110' 000000 AWAS:    OPEN                    ;EXPECTED DATA.
000112' 000232' RSTRT:  RESTRT                  ;ACTUAL DATA.
000114' 000000 WDTO:    OPEN                    ;RESTART ADDRESS AFTER END OF PASS
000116' 000000 WDFR:    OPEN                    ;WORDS TO MEMORY PER ITERATION
000120' 000000 INTR:    OPEN                    ;WORDS FROM MEMORY PER ITERATION
                                                ;# OF INTERRUPTS PER ITERATION
```



```
000122' 000034          IDNUM:  34
        000040                  .REPT   SPSIZ           ;MODULE IDENTIFICATION NUMBER=34
                                .NLIST                  ;MODULE STACK STARTS HERE.
                                .WORD   0
                                .LIST
000224'                         .ENDR
                        MODSP:
                        ;*****************************************************************************
  223
  224 000224' 012767  000002  177664 START:  MOV     #2,WDFR             ;2 WORDS FROM MEM/ITERATION
  225 000232' 016700  177550 RESTRT: MOV     ADDR,R0             ;GET THE 1ST REG. ADDRESS
  226 000236' 016702  177554         MOV     SR1,R2              ;GET THE DISTR. MODULE COUNT
  227 000242' 006302                 ASL     R2                  ;MAKE IT A REG. COUNT
  228 000244' 005010                 CLR     (R0)                ;CLEAR THE REGISTER
  229 000246' 001024 1$:     CLR     (R0)
  230 000250' 005302                 DEC     R2                  ;BR IF IT DIDN'T CLEAR ALL BITS
  231 000252' 001402 4$:     BNE     5$                  ;COUNT ONE REG.
  232 000254' 005720                 BEQ     3$                  ;BR IF ALL REGS CLEARED
  233 000256' 000772                 TST     (R0)+               ;GENERATE NXT ADDRESS
  234 000260' 016700  177522         BR      1$                  ;GO CLEAR THE NEXT ONE
  235 000264' 016702  177526 3$:     MOV     ADDR,R0             ;GET THE 1ST REG. ADDRESS
  236 000270' 006302                 MOV     SR1,R2              ;GET THE DISTR. MODULE COUNT
  237 000272' 005110                 ASL     R2                  ;MAKE IT A REG. COUNT
  238 000274' 022710  177777 8$:     COM     (R0)                ;SET ALL ONES
  239 000300' 001022                 CMP     #177777,(R0)        ;DID ALL BITS SET ?
  240 000302' 005302                 BNE     6$                  ;BR IF NOT
  241 000304' 001402 7$:     DEC     R2                  ;COUNT ONE REG.
  242 000306' 005720                 BEQ     2$                  ;BR IF ALL REGS SET
  243 000310' 000770                 TST     (R0)+               ;GENERATE THE NXT ADDRESS
  244 000312'                        BR      8$                  ;GO DO ANOTHER ONE
  245 000312' 104413  000000' 2$:
                                ENDITS,BEGIN            ;SIGNAL END OF ITERATION.
  246                                                   ;MONITOR SHALL TEST END OF PASS
  247 000316' 000745                 BR      RESTRT              ;BR TIL 100. TIMES THROUGH
  248 000320' 010067  177554 5$:     MOV     R0,CSRA             ;SAVE THE ADDRESS OF THE REG.
  249 000324' 011067  177552         MOV     (R0),ACSR           ;SAVE THE CONTENTS
  250 000330' 012767  000025  177550 MOV     #25,ERRTYP          ;BIT STUCK IN REG.
  251                                ;*****************************************************************
  252 000336' 104405  000000' 000000 HRDERR$,BEGIN,NULL   ;FAILED TO CLEAR ALL BITS
  253                                ;*****************************************************************
  254 000344' 000741                 BR      4$                  ;GO TRY THE NEXT GUY
  255 000346' 010067  177526 6$:     MOV     R0,CSRA             ;SAVE THE ADDRESS OF THE REG
  256 000352' 011067  177524         MOV     (R0),ACSR           ;SAVE THE CONTENTS
  257 000356' 012767  000025  177522 MOV     #25,ERRTYP          ;BIT STUCK IN REG.
  258                                ;*****************************************************************
  259 000364' 104405  000000' 000000 HRDERR$,BEGIN,NULL   ;FAILED TO SET ALL BITS
  260                                ;*****************************************************************
  261 000372' 000743                 BR      7$                  ;GO TRY THE NEXT GUY
  262
  263
  264         000001          .END
```

```
ACSR    000102R          205#    249*    256*
ADDR    000006R          171#    225     234
ADDR22= 001000           223#
ASB     000106R          209#
ASTAT   000104R          207#
AWAS    000110R          210#
BEGIN   000000R          168#    245     252     259
BIT0    = 000001         223#
BIT1    = 000002         223#
BIT10   = 002000         223#
BIT11   = 004000         223#
BIT12   = 010000         223#
BIT13   = 020000         223#
BIT14   = 040000         223#
BIT15   = 100000         223#
BIT2    = 000004         223#
BIT3    = 000010         223#
BIT4    = 000020         223#
BIT5    = 000040         223#
BIT6    = 000100         223#
BIT7    = 000200         223#
BIT8    = 000400         223#
BIT9    = 001000         223#
BREAK$= 104407           223#
BR1     000012R          173#
BR2     000013R          174#
BTOD$ = 104421           223#
CDATA$= 104412           223#
CONFIG  000056R          193#
CSRA    000100R          203#    248*    255*
DATCK$= 104411           223#
DATE$= 104404            223#
DVID1   000014R          175#
ENDIT$= 104413           223#    245
END$    = 104410         223#
ERRTYP  000106R          208#    250*    257*
EXIT$ = 104400           223#
GETPA$= 104415           223#
GWBUF$= 104414           223#
HRDCNT  000044R          188#
HRDER$= 104405           223#    252     259
HRDPAS  000050R          190#
ICONT   000036R          185#
ICOUNT  000040R          186#
IDNUM   000122R          215#
INIT    000030R          182#
INTR    000120R          214#
MAP22$= 104416           223#
MODNAM  000000R          169#
MODSP   000224R          183     221#
MSGN$ = 104403           223#
MSGS$ = 104402           223#
MSG$  = 104401           223#
NULL  = 000000           223#    252     259
OPEN  = 000000           170     176     177     178     179     196     197     198     199     200     201     202     203
                         205     207     209     210     212     213     214     223#
```



```
OTOA$ = 104420           223#
PASCNT  000034R          184#
PIRQ$ = 000004           223#
POPSP = 005726           223#
POPSP2= 022626           223#
PRTY  = 000000           173     174     223#
PRTY0 = 000000           223#
PRTY1 = 000040           223#
PRTY2 = 000100           223#
PRTY3 = 000140           223#
PRTY4 = 000200           223#
PRTY5 = 000240           223#
PRTY6 = 000300           223#
PRTY7 = 000340           223#
PS    = 177776           223#
PSW   = 177776           223#
PUSH  = 005746           223#
PUSH2 = 024646           223#
RAND$ = 104417           223#
RANNUM  000054R          192#
RESTRT  000232R          211     225#    247
RES1    000056R          194#
RES2    000060R          195#
RSTRT   000112R          211#
SBADR   000102R          204#
SOFCNT  000042R          187#
SOFER$= 104406           223#
SOFPAS  000046R          189#
SPOINT  000032R          183#
SPSIZ = 000040           1#      216
SR1     000016R          176#    226     235
SR2     000020R          177#
SR3     000022R          178#
SR4     000024R          179#
START   000224R          182     224#
STAT    000026R          181#
SVR0    000052R          196#
SVR1    000064R          197#
SVR2    000066R          198#
SVR3    000070R          199#
SVR4    000072R          200#
SVR5    000074R          201#
SVR6    000076R          202#
SVSCNT  000052R          191#
TRPDFD= 000022           223#
VECTOR  000010R          172#
WASADR  000104R          206#
WDFR    000116R          213#    224*
WDTO    000114R          212#
XFLAG   000005R          170#

. ABS.  000000      000
        000374      001
```

SEQ 0008

```
 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 XCBBE0,XCBBE0/SOL/CRF:SYM=DDXCOM,XCBBE0
 RUN-TIME:  1  1  .2 SECONDS
 RUN-TIME RATIO:  8/2=3.3
 CORE USED:  7K  (13 PAGES)
```