KW11-W

LOGIC TEST

MD-11-DDKWB-A

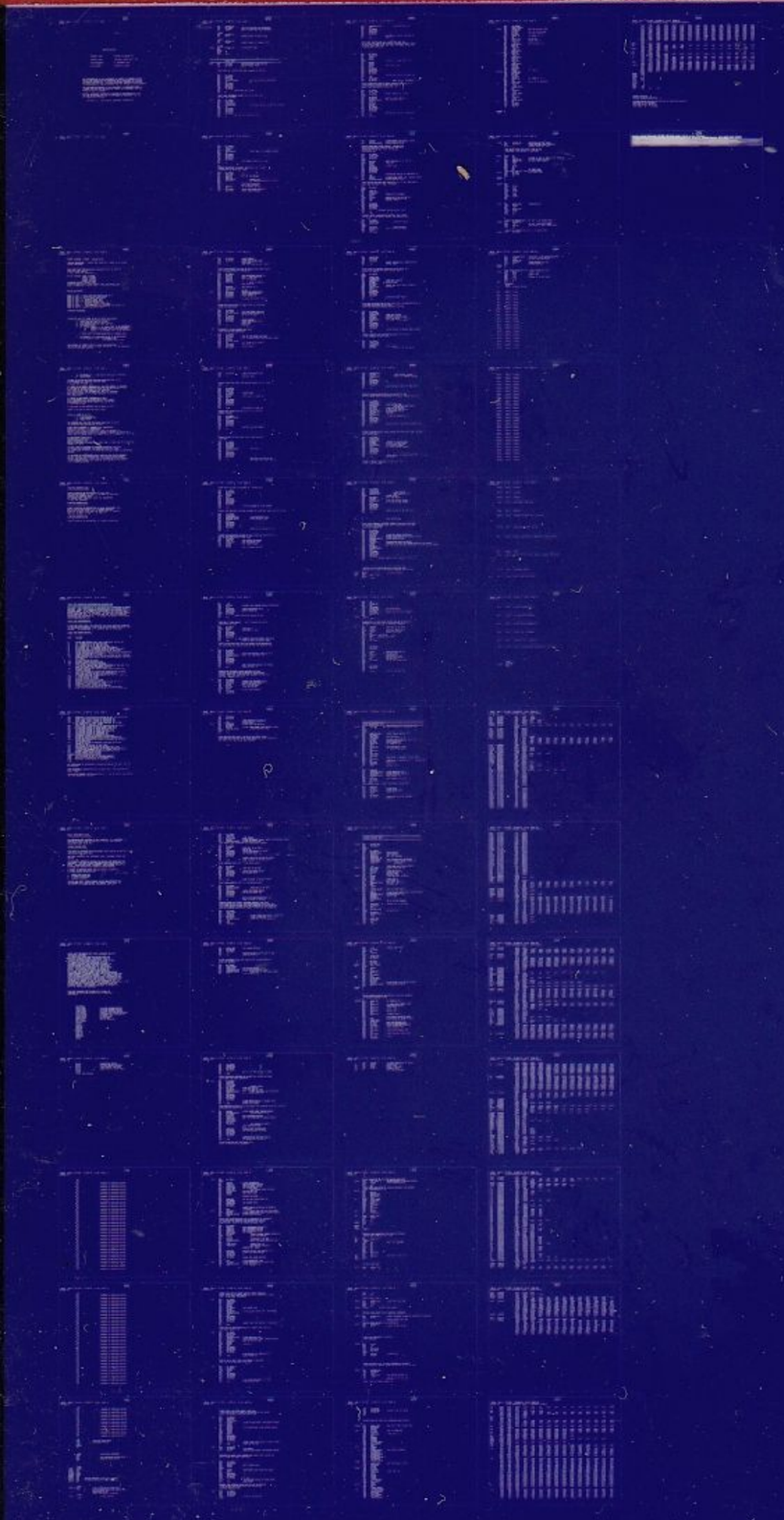
EP-DDKWB-A-DL-A

OCT 1976

FICHE 1 OF 1

digital

Made in U.S.A.



```
.REM }
```

IDENTIFICATION

PRODUCT CODE:	MAINDEC-11-DDKWB-A-D
PRODUCT NAME:	WATCHDOG TIMER LOGIC TEST
DATE RELEASED:	21 DECEMBER 1975
MAINTAINER:	DIAGNOSTIC GROUP

MEMORY STORAGE - 00000 - 012000 OCTAL

LOADING PROCEDURE - (PAPER TAPE SUPPLIED) LOADED USING NORAML
BINARY PROCEDURE.

MAKE SURE THAT THE FOLLOWING PARAMETERS ARE THE SAME AS
THOSE OF YOUR SYSTEM.
IF NOT, PATCH THE TAGS IN ():

DEVICE ADDRESS (CSR) =172400
(CINT) =172402
(ECSR) =172404
(SWBU) =172406
INTERRUPT VECTOR (VADRS) =320
DEVICE PRIORITY INT. LEVEL(DPRIV) =265 (IN ASCII) (=5)
MODEL OF COMPUTER- (PDP11) =301 (=A) (05 OR 10 OR 20)

SWITCH SETTINGS:

SW00 =1 (UP) --- WILL EXIT ANY PARTICULAR TEST
SW01 =1 (UP) --- NO TTY WITH THE SYSTEM
SW02 =1 (UP) ---- SELECTS LOGIC TEST
SW03 =1 (UP) ---- SELECTS DELAY TEST
SW04 =1 (UP) ---- SELECTS DYNAMIC TEST
SW05 =1 (UP) ---- SELECTS DELAY:(1) WATCHDOG
SW05 =0 (DOWN) --- SELECTS DELAY:(2) WARNING AND SHORT LOOP
SW06 =1 (UP) --- HALTS ON END OF PASS

STARTING PROCEDURE:

EITHER A) ONLY IF THERE IS NO TTY WITH THE SYSTEM:

1.1 PATCH THE TAG TYSR TO 177570
1.2 LOAD ADDRESS 200 AND KEEP SW07 UP.
1.3 SET SW01=1 (UP)
1.4 SET EITHER SW02=1 --- IF LOGIC TEST TO BE PERFORMED
OR SW03=1---IF DELAY TEST TO BE PERFORMED
OR SW04=1 ---IF DYNAMIC TEST TO BE PERFORMED

IF DELAY TEST HAS BEEN SELECTED I.E. SW03=1,THEN

1.5 SET SW05=1 ---IF WATCHDOG DELAY TO BE PERFORMED
OR SET SW05=0 (DOWN)---FOR SHORT LOOP
AND WARNING DELAY TEST

FOR EXAMPLE,IF THERE IS NO TTY WITH THE SYSTEM,AND
ONE OF THE TWO DELAY TESTS ---WATCH DOG DELAY ---TO BE PERFORMED,
THEN SET SW01,SW03,SW05 UP.

1.6 SET SW06=1 -----FOR END OF PASS HALT (OPTIONAL)
1.7 PRESS START

IF SW01 IS UP AND NEITHER SW02 NOR SW03 NOR SW04 IS UP
THEN PROGRAM WILL HALT AT LOCATION 7256
IF SO,GO BACK TO STEP 1.2

IF THERE IS NO ERROR, PROGRAM WILL RUN FOR APPROX. 2 MINUTES
AT THE END OF WHICH THERE WILL BE A "5" ON THE DISPLAY LIGHTS
FOR COUPLE OF SECONDS (ONLY IF THE COMPUTER
IS A 1105/10/20) AND PROGRAM WILL HALT AT LOCATION
252 (ONLY IF SW06 IS UP).PRESS CONTINUE TO CONTINUE
WITH THIS TEST.

IF THERE IS ANY ERROR, PROGRAM WILL HALT.
PC OF THE ERROR WILL BE DISPLAYED ON THE LIGHTS.
ALSO,ADDRESS OF THE ERROR MESSAGE WILL BE STORED
AT LOCATION 017400.

IF YOU WISH TO RUN ANOTHER TEST,GO BACK TO STEP 1.2.

THERE IS NO END OF PASS FOR DELAY TESTS.

OR B) IF THERE IS A TTY:

1.1 LOAD ADDRESS 200
1.2 PRESS START

THE PROGRAM WILL ASK THE FOLLOWING QUESTIONS TO WHICH
THE OPERATOR MUST RESPOND BY TYPING:

FIRST DEVICE ADDRESS = (ANSWER IN 6 CHARACTERS)
FIRST INT. VECTOR = (3 CHARACTERS)
PRIORITY INT. LEVEL = (INPUT INT. LEVEL OF DEVICE 4-7)
PDP-11 (A)=05,10,20 (B)=35,40 (C)=45,50 ---ANSWER IS A OR B OR C
LOGIC TEST (1),DELAY TEST (2),DYNAMIC TEST(3) ---ANSWER IS 1 OR 2 OR 3

IF THE ANSWER TYPED WAS 2
PROGRAM WILL TYPE OUT ---
DELAY ADJUSTMENT TEST
DELAY:(1)WATCHDOG.(2)WARNING & SHORT LOOP ---THIS LINE IS A QUESTION.
ANSWER IS 1 OR 2

YOU WILL HAVE 15 SECONDS TO ANSWER A QUESTION (VIA TTY).
IF YOU ARE NOT ANSWERING THE QUESTIONS VIA TTY, THEN
YOU MUST START AT STEP 1.1 OF PART A) OR START DIRECT AS DESCRIBED
BELLOW.

IF AN ERROR IS ENCOUNTERED WHILE INPUTING THE DEVICE ADDRESS
OR INT. VECTOR, THE OPERATOR MAY TYPE A RUB OUT AND REPEAT
ENTIRE LINE INPUTING. WHEN LINE IS COMPLETE OR IF NO CHANGES
(FROM STANDARD DEVICE) THE OPERATOR MUST TERMINATE THE LINE
WITH A CARRIAGE RETURN.

```
        STARTING ADDRESS 1000:
        ++++++++++++++++++++++

        DIRECT STARTING OR RESTARTING OF LOGIC TEST.
        YOU CAN SKIP STEPS 1.2 THRU 1.5 OF PART A) AND START
        DIRECTLY AT ADDRESS 1000.
        BUT BEFORE THAT  YOU MUST CHECK THE PARAMETERS
        IF THEY ARE STANDARD.

        STARTING ADDRESS 220:
        +++++++++++++++++++++

        DIRECT STARTING OR RESTARTING OF DELAY ADJUSTMENT TEST.
        BUT IF YOU DO NOT HAVE A TTY TO ANSWER ANOTHER
        QUESTION IN DELAY TEST,THEN YOU MUST START AT ADDRESS
        200 I.E. GO BACK TO STEP 1.1 OF PART A).

        STARTING ADDRESS 240:
        +++++++++++++++++++++

        DIRECT STARTING OR RESTARTING OF DYNAMIC SYSTEM TEST.
```

LOGIC TEST OPERATING PROCEDURE AND DESCRIPTION

THE LOGIC TEST CONSISTS OF 44 SUB TESTS EACH CONTAINING AN ERROR ROUTINE. LOGIC TEST SHOULD BE STARTED WITH ALL CONSOLE (PDP-11) SWITCHES DOWN. IF AN ERROR IS ENCOUNTERED, THE TELETYPE WILL REPORT WITH FAILING TEST AND HALT. TO LOOP ON A FAILURE CONSOLE SWITCH REGISTER, BIT 14 IS RAISED AND THE TEST IS LOOPED BY DEPRESSING THE CONTINUE KEY. TO INHIBIT THE ERROR TYPING RAISE SWITCH REGISTER BIT 13.

LOGIC TEST REQUIREMENTS:

TO RUN THE LOGIC TEST, THE SUPPLIED TEST PLUG MUST BE CONNECTED TO THE MODULE CONNECTOR. THE T3 DELAY MUST EXCEED T2 BY 50 MICRO-SECONDS. THE ADJUSTMENT OF T1 MUST BE SET AT A MINIMUM.

LOGIC TEST ERROR REPORTS.

TEST	FAILURE
T1	STATUS AND CONTROL REG. NOT CLEAR UPON INITIALIZE.
T2	INT. ENABLE BIT 6 OF CSR NOT SET.
T3	INT. ENABLE BIT 6 OF CSR NOT CLEAR.
T4	AN ILLEGAL INT. IS PRESENT FROM THE MODULE.
T5	AN ILLEGAL INT. IS PRESENT AFTER A DELAY TIME OUT.
T6	STATUS REGISTER NOT CLEAR AFTER DELAY TIME OUT.
T7	STATUS REGISTER WAS AFFECTED BY A CLEAR FLAGS INSTRUCTION.
T8	FAILED TO MONITOR A T2 FLAG.
T9	ALARM RECEIVED BIT 14 OF CSR SET WITHOUT AN EXT. ENALBE.
T10	ALARM RECIEVE FLAG FAILED TO SET AFTER ENABLING THE EXTERNAL WORLD.
T11	T2 FLAG FAILED TO CLEAR.
T12	RECIEVE FLAG FAILED TO CLEAR.
T13	SHORT LOOP (T1) FAILED TO CLEAR.
T14	T2 FLAG SET SHOULD HAVE BEEN TEMINATED BY THE LOCK FLOP UPON THE SHORT LOOP GENERATED IN T13.
T15	RECEIVE FLAG WAS SET UPON A SHORT LOOP WITHOUT THE EXTERNAL ENABLE.
T16	LOCK FLOP FAILED TO CLEAR UPON ENABLING THE EXTERNAL WORLD AND WAITING 5M.SEC. FOR CLEAR ERROR.
T17	NO ERROR CHECKING, CONDITIONING FOR T18, T19, AND T20.
T18	T1 FLAG FAILED TO CLEAR.
T19	T2 FLAG FAILED TO CLEAR.
T20	RECEIVE FLAG FAILED TO CLEAR.
T21	RECEIVE FLAG FAILED TO SET UPON A TIME OUT OF T3.
T22	ALARM REC. FLAG FAILED TO CLEAR.
T23	ENABLE FLOP NOT CLEARED AN ALARM OUT.
T24	A T2 FLAG FAILED TO RAISE AN INT. AT VECTOR 350.
T25	SHORT LOOP FLAG FAILED TO RAISE AN INT. AT VECTOR 350.

```
        T26     ALARM REC. FLAG FAILED TO RAISE AN INT. AT VECTOR 350.
        T27     SHORT LOOP "LOCK" FAILED TO TERMINATE T3.
        T28     SHORT LOOP "LOCK" FAILED TO TERMINATE T2.
        T29     THE CLEARING OF "LOCK" DIDNOT RELEASE T2 TO THE
                BUS (REFER TO T26,T27, &T28).
        T30     SHORT LOOP " LOCK"FAILED TO LOCK PULSING OF TIMER.
        T31     INITIALIZE FAILED TO CLEAR SHORT LOOP FLAG.
        T32     INITIALIZE FAILED TO CLEAR RECEIVE FLAG.
        T33     INITIALIZE FAILED TO CLEAR THE LOCK FLOP.
        T34     INITIALIZE FAILED TO TERMINATE T3.
        T35     INITIALIZE FAILED TO TERMINATE T2.
        T36     INCORRECT STATUS IN EXTERNAL STATUS REG.
        T37     STATUS OF ECSR CHANGED UPON A RESET INSTRUCTION.
        T38     SWITCH BUS INSTRUCTION FAILED TO CHANGE THE EXTERNAL
                STATUS REGISTER.
        T39     THE LATCH FAILED TO PREVENT A 2ND BUS SWITCHING
                (REFER TO T38).
        T40     BUS FAILED TO SWITCH TO 2ND LATCH.
        T41     TERMINATION OF T3 FAILED TO RELEASE THE LATCH.
        T42     A SWITCH BUS COMMAND FAILED TO CHANGE THE BUS STATUS.
        T43     A RESET INSTRUCTION FAILED TO CLEAR THE LATCH.
        T44     MANUAL ENABLE INPUT FAILED.
        ************************************
        T38S    TEST FOLLOWED BY AN "S" DENOTES SUBTEST.
        T40S    WHICH IS A FAILURE OF THE EXTERNAL STATUS
        T41S    REGISTER (STATUS READ IS INCORRECT).
        T43S
```

ALL TESTS MUST BE SUCCESSFULLY COMPLETED BEFORE THE NEXT TEST MAY BE EXECUTED.

UPON SUCCESSFUL COMPLETION OF THE LOGIC TEST, THE TELETYPE WILL PRINT "PASS".

FOR SCOPING ERRORS, SWITCH REGISTER BIT 7 MAY BE RAISED TO INCREASE THE TEST REPEATIBILITY RATE.

DELAY ADJUSTMENT TESTS.
++++++++++++++++++++++++++

THE OPERATOR MUST RESPOND TO THE QUESTION, "(1) WATCHDOG, (2)
WARNING & SHORT LOOP" BY TYPING A 1 OR A 2 FOR DESIRED
ADJUSTMENT LOOP TO BE RUN.

DYNAMIC SYSTEM TEST.
++++++++++++++++++++

THIS TEST IS INTENDED AS AN EXCERSER WHICH SHOULD BE RUN ONLY AFTER
COMPLETING THE LOGIC TEST.

THE TEST EXERCISES THE INTERRUPT LOGIC, EXTERNAL STATUS AND
DELAYS.

THE TIMER IS PRIMED BY PULSING THE DELAYS AND ENABLING THE
INTERRUPT. THE ONLY LEGAL INTERRUPTS ARE THOSE FROM THE T2
DELAY. AFTER EACH LEGAL INTERRUPT, THE EXTERNAL STATUS IS
SWITCHED AND TESTED FOR A CORRECT STATUS READING.

A "PASS" IS PRINTED AFTER 100 SUCCESSFUL INTERRUPTS.
POSSIBLE ERRORS REPORTED ARE:

1. SHORT LOOP FLAG SET
2. EXTERNAL STATUS BAD
3. ILLEGAL INTERRUPT

TO EXIT ANY TEST, RAISE CONSOLE SWITCH REGISTER,BIT 00,
THE PROGRAM WILL THEN OUTPUT THE HEADER. MAKE SURE THAT
THE SWITCH IS LOWERED BEFORE ENTERING A NEW TEST.

```
        LOGIC TEST FOR WATCH DOG TIMER (HARDWARE MONITOR)
        FOR PDP-11 SYSTEMS.
        THE WATCH DOG TIMER MUST BE PULSED FROM THE
        COMPUTER PROGRAM WITHIN A SPECIFIED TIME TO
        INHIBIT THE TIMER FROM TIMING OUT.  UPON A TIME
        OUT CONDITION THE WATCH DOG WILL POST AN
        INTERRUPT ALLOWING THE PROCESSOR A CHANCE TO
        RECOVER BEFORE A SECOND TIME OUT CONDITION
        OCCURS ISSUING AN "ALARM OUT" TO THE EXTERNAL
        WORLD.  THE WARNING INTERRUPT IS SENSED AT
        BIT 07 OF THE CONTROLL AND STATUS REGISTER.
        THE TIMER ALSO HAS A "SHORT LOOP" PROTECTION WHICH
        GUARDS AGAINST A PROGRAM FAILURE THAT MAYBE PULSING THE
        TIMER BEFORE A SPECIFIED INTERVAL.  THE SHORT LOOP WILL
        POST AN INTERRUPT ALSO AT VECTOR (XXX) WHENEVER PULSED
        TWICE WITHIN A SPECIFIED TIME INTERVAL.  THE SHORT LOOP
        INTERRUPT IS SENSED IN THE STATUS REGISTER BY BIT 14.
        THE RECEIVE CIRCUIT ALLOWS FOR AN INTERRUPT (VECTOR XXX) TO BE
        POSTED WHEN AN "ALARM SIGNAL" IS RECEIVED FROM THE
        EXTERNAL WORLD.  THE RECEIVE INTERRUPT IS SENSED AT
        BIT 7 OF THE STATUS REGISTER.


        FOR TEST PURPOSES THE "ALARM OUT" SIGNAL IS
        JUMPERED TO THE ALARM RECEIVE WITH A SUPPLIED
        TEST PATCH.

                                ]


        PS=177776
        TKB=177562              ;TELETYPE READER BUFFER
        TKS=177560              ;TELETYPE READER STATUS
        TYDB=177566             ;TELETYPE PRINTER BUFFER
        CSR=172400              ;CONTROLL AND STATUS-WATCHDOG
        CINT=172402             ;INTERRUPT CLR INSTRUCTION
        ECSR=172404             ;EXTERNAL CSR
        SWBU=172406             ;SWITCH BUS
        SR=177570               ;SWITCH REGISTER
        NULL=104000
        NOP=240
        HLT=0
        SW00=01
        SW01=02
        SW02=4
        SW03=10
        SW04=20
        SW05=40
        SW06=100
        R0=%0
        R1=%1
```

```
                R2=%2                   ;SUBROUTINE RETAIN
                R3=%3                   ;TELTYPE ROUTINE RETAIN
                R4=%4                   ;TIME CONTROL ROUTINES
                R5=%5                   ;ERROR PRINT & SCOPE MODE
                SP=%6
                PC=%7
                ;LOAD TRAP CATCHER
```

```
        .=0
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
```

```
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                 ;TRAPPED TO PREVIOUS VECTOR
```

```
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
        .+2
        HLT                     ;TRAPPED TO PREVIOUS VECTOR
                ;
        .=4
        TOBE            ;TIME OUT BUSS ERROR
        340             ;PROCESSOR STATUS
        .=30
        NULPRT
        340
                ;
                ;
                ;
        .=350                   ;WATCH DOG INTERRUPT
        WAOG
        340                     ;SET PROCESSOR STATUS BACK TO 7
        RECF                    ;RECEIVE FLAG VECTOR
        340
                ;
        ;
        ;
        .=500
TYSR:   177564
XCSR:   CSR
XCINT:  CINT
XECSR:  ECSR
XSWBU:  SWBU
VADRS:  320
DPRIV:  265     ;DEVICE PRIORITY LEVEL (5) IN ASCII
PPRIV:  200     ;PROCESSOR PRIORITY LEVEL [DPRIV-1]
PDP11:  301
        ;
        ;
        .=250
EOPHLT: HALT            ;THIS IS AN END OF PASS HALT
        RTS     PC      ;NOT AN ERROR HALT.THIS HAPPENS ONLY
                        ;WHEN SW06 IS UP.
                        ;PRESS CONTINUE TO CONTINUE
        .=600
STACK:  0                       ;STACK LOCATOR
        .=200
```

```
                MOV     #STACK,SP
        1$:     JSR     R5,ADDMO        ;ASK FOR VECTORS AND ADDRESSES
                JMP     HEADER          ;PRINT HEADER AND TAKE COMMANDS
                ;
                .=220
                MOV     #STACK,SP
                JMP     DELAY           ;DIRECT ENTRY TO DELAY TEST
                ;
                .=240
                MOV     #STACK,SP
                JMP     DYMAN           ;DIRECT ENTRY TO DYMANIC TEST
                ;
                .=1000
        LOGIC:  MOV     #STACK,SP       ;INITIALIZE STACK
                NULL
                RESET
                NULL
                JMP     TEST            ;
                ;
                ;
        ;*************************************************************************
        ;               LOGIC TEST
        ;*************************************************************************(
        TEST:   MOV     #LT,MES         ;SET REGISTER 3 FOR OUTCODE
                JSR     R3,PRINT        ;PRINT "LOGIC TEST"
                ;
                ;
                ;TEST THAT ALL STATUS BITS ARE CLEARED IN THE CSR
                ;
        T1:     JSR     R1,CHECK
                MOV     #340,PS
                MOV     #T1,LOOP
                BIT     #177777,@XCSR           ;TEST ALL BITS OF STATUS REG
                BEQ     T2
                JSR     R5,ERROR
                MOV     #EE1,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT             ;STATUS REG NOT CLEAR
                JMP     T1
                ;
                ;TEST THAT INTERRUPT ENABLE (BIT 6 OF CSR) CAN
                ;BE SET AND CLEARED .
                ;
        T2:     JSR     R1,CHECK
                MOV     #T2,LOOP
                MOV     #100,@XCSR
                BIT     #100,@XCSR              ;TEST THAT INT.EN. IS SET IN THE CSR
                BNE     T3
                JSR     R5,ERROR
                MOV     #EE2,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT             ;INT. EN. BIT FAILED TO SET
                JMP     T2
```

```
                ;
                ;
                ;
T3:             JSR     R1,CHECK
                MOV     #T3,LOOP
                CLR     @XCSR
                BIT     #100,@XCSR              ;CHECK THAT INT ENABLE CLEARED
                BEQ     T4
                JSR     R5,ERROR
                MOV     #EE3,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                             ;INT.ENABLE FAILED TO CLEAR
                JMP     T3
                ;
                ;EX
                ;ENABLE PROCESSOR INTERRUPT AND CHECK FOR ILLEGAL
                ;INTERRUPTS OF WATCH DOG TIMER
                ;
T4:             JSR     R1,CHECK
                MOV     #T4,LOOP
                MOV     #AA1,R4                 ;SET UP INT RETURN
                MOV     #AA1,R5
                BIS     #100,@XCSR                      ;ENABLE INT
                MOV     PPRIV,PS                        ;SET PROCESSOR STATUS TO PPRIV
                NOP
                NOP                             ;WAIT FOR INTERRUPT
                MOV     #340,PS                 ;NO INTERRUPT SO GO
                JMP     T5
AA1:            JSR     R5,ERROR                ;CHECK FOR ERROR PRINTOUT
                MOV     #EE4,MES                ;PRINT ERROR MESSAGE
```

```
                JSR     R3,PRINT        ;PRINT ERROR 1
                JSR     R5,SCOPE        ;CHECK FOR SCOPE LOOP
                HLT                     ;STAY HERE
                JMP     T4              ;RUN TEST OVER ON KEY "COUNT"
                ;
                ;
                ;WAIT FOR POSSIBLE TIME OUT OF DELAYS AND CHECK AGAIN
                ;FOR ILLEGAL WATCHDOG INTERRUPTS
                ;
T5:             JSR     R1,CHECK
                MOV     #340,PS         ;SET PROCESSOR STATUS TO 7
                JSR     R2,TOUT         ;WAIT . FOR TIME OUTS
                MOV     #AA2,R4         ;SET INT RETURN
                MOV     #AA2,R5
                MOV     PPRIV,PS                ;P.S. TO PPRIV
                NOP                     ;WAIT FOR INT
                NOP
                MOV     #340,PS         ;SET STATUS TO 7
                BIC     #100,@XCSR
                JMP     T6              ;GO ON
AA2:            JSR     R5,ERROR        ;CHECK FOR ERROR PRINT
                MOV     #EE5,MES        ;PRINT ERROR MESSAGE
                JSR     R3,PRINT        ;PRINT ERROR 1A
                JSR     R5,SCOPE
                HLT                     ;ILLEGAL INT AFTER TIME OUT
                JMP     T5
                ;
                ;READ THE STATUS WORD TO CHECK THAT ALL BITS ARE
                ;CLEAR AT THIS TIME.
                ;
T6:             JSR     R1,CHECK
                MOV     #T6,LOOP        ;SET FOR SCOPE LOOP RUN
                BIT     #177777,@XCSR   ;CHECK STATUS REGISTER
                BEQ     T7              ;IF 0 GO TO TEST 7
                JSR     R5,ERROR
                MOV     #EE6,MES
                JSR     R3,PRINT        ;PRINT ERROR2
                JSR     R5,SCOPE        ;CHECK FOR SCOPE LOOP
                HLT                     ;STAY HERE
                JMP     T6              ;RECYCLE
                ;
                ;GENERATE A CLEAR COMMAND AND CHECK
                ;THAT STATUS WAS UNAFFECTED
                ;
T7:             JSR     R1,CHECK
                MOV     #T7,LOOP        ;SET UP FOR SCOPE LOOP RUN
                TST     @XCINT          ;CLR TO FLAGS SHORT LOOP,SECOND CHANCE
                JSR     R2,TOUT
                BIT     #177777,@XCSR
                BEQ     T8              ;ALL CLEAR GO TO TEST 8
                JSR     R5,ERROR
                MOV     #EE7,MES        ;PRINT ERROR 3
                JSR     R3,PRINT
```

```
                JSR     R5,SCOPE        ;CHECK FOR SCOPE LOOP
                HLT
                JMP     T7              ;RECYCLE TEST
                ;
                ;
                ;
                ;PULSE TIMER AND CHECK FOR SECOND CHANCE (T2) TO TIME
                ;OUT
                ;
T8:             JSR     R1,CHECK
                MOV     #T8,LOOP
                MOV     #340,PS
                MOV     #1,@XCSR        ;PULSE TIMER
                JSR     R2,TOUT
                TSTB    @XCSR           ;TEST THAT T2 TIMED OUT
                BMI     T9
                JSR     R5,ERROR
                MOV     #EE8,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT
                JMP     T8              ;T2 FAILED TO TIME OUT
                ;
                ;CHECK THAT ALARM RECEIVE (BIT 14) IS NOT SET
                ;ENABLE FLOP=0
                ;
T9:             JSR     R1,CHECK
                BIT     #040000,@XCSR
                BEQ     T10
                JSR     R5,ERROR
                MOV     #EE9,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                     ;REC.SET WITHOUT ENABLE
                JMP     T9
                ;
                ;
                ;
                ;
                ;ENABLE ALARM OUT AND CHECK THAT ALARM RECEIVE
                ;WILL SET
                ;
T10:            JSR     R1,CHECK
                MOV     #T8,LOOP
                MOV     #2,@XCSR
                JSR     R2,WAITX
                BIT     #040000,@XCSR           ;TEST BIT 14
                BNE     T11
                JSR     R5,ERROR
                MOV     #EE10,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                             ;RECEIVE FLAG FAILED TO
                JMP     T8                      ;SET,CHECK THAT T3 TIMED OUT
                ;
                ;
```

```
                ;TEST THAT T2 FLAG IS CLEARED BY (164002)CINT
                ;
        T11:    JSR     R1,CHECK
                MOV     #T11,LOOP
                TST     @XCINT
                TSTB    @XCSR
                BPL     T12
                JSR     R5,ERROR
                MOV     #EE11,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                             ;T2 NOT CLEARED BY CINT(164002)
                JMP     T11
                ;
                ;CHECK THAT RECIEVE FLAG WAS CLEARED BY CLEAR REC FLAG (BIT 8 OF CSR)
                ;
        T12:    JSR     R1,CHECK
                MOV     #T11,LOOP
                MOV     #000400,@XCSR           ;CLEAR RECIEVE FLAG
                BIT     #040000,@XCSR           ;TEST RECIEVE FLAG
                BEQ     T13
                JSR     R5,ERROR
                MOV     #EE12,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                             ;RECEIVE FLG FAILED TO CLR
                JMP     T12
                ;
                ;ISSUE TWO SUCCESSIVE PULSES TO THE TIMER AND MONITOR BIT 15
                ;OF THE CSR FOR A SHORT LOOP FLAG
        T13:    JSR     R1,CHECK
                MOV     #340,PS         ;SET STATUS TO SEVEN
                MOV     #THW,LOOP       ;SET FOR SCOPE LOOP
                INC     @XCSR           ;HIT TIMER TWICE
                INC     @XCSR
                TST     @XCSR           ;BIT 15 SHOULD BE SET
```

```
                BPL     .+12
                JSR     R2,TOUT         ;GO WAIT FOR TIMEOUT BRFORE PROCEEDING
                JMP     T14
                JSR     R5,ERROR        ;CHECK FOR ERROR PRINT
                MOV     #EE13,MES       ;PRINT ERROR 13
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT             ;SHORT LOOP FLAG FAILED TO SET
                JMP     THW
                ;
                ;
                ;TEST BIT 7 (2ND CHANCE) .  BIT 7 SHOULD NOT BE SET
                ;TERMINATED BY LOCK
                ;
        T14:    JSR     R1,CHECK
                TSTB    @XCSR           ;TEST BIT 7
                BPL     T15             ;CONTINUE TO TEST
                JSR     R5,ERROR
                MOV     #EE14,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT             ;T2 (2ND CHANCE) FLAG SET,CHECK LOCK FLOP
                JMP     THW             ;LOOP TO TEST 13 ON KEY "CONT"
                ;
                ;WAIT 5 M.SEC AND TEST THAT THE ENABLE FLOP PREVRENTED
                ;THE WATCH DOG OUT SIGNAL FROM SETTING THE RECEIVE FLAG
                ;
                ;
        T15:    JSR     R1,CHECK
                MOV     #340,PS
                JSR     R2,WAITX        ;WAIT FOR POSSIBLE TIME OUT OFT4
                BIT     #040000,@XCSR   ;TEST THAT REC.DIDNT SET
                BEQ     T16
                JSR     R5,ERROR
                MOV     #EE15,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                     ;REC. FLAG SET WITHOUT AN ENABLE
                JMP     THW             ;LOOP ON TEST 13
                ;
                ;
                ;ENABLE THE EXTERNAL WORLD AND WAIT 5 M.SEC
                ;FOR THE LOCK FLOP TO BE CLEARED BY CLEAR ERROR
                ;SIGNAL. TEST THAT LOCK FLOP WAS CLEARED BY RESETTING
                ;THE 2ND. CHANCE FLAG (T2).
                ;
        T16:    JSR     R1,CHECK
                MOV     #2,@XCSR        ;ENABLE THE EXTERNAL WORLD
                JSR     R2,WAITX        ;WAIT FOR CLEAR ERROR
                INC     @XCSR           ;SET T2 FLAG AGAIN
                JSR     R2,TOUT         ;TIME OUT DELAYS
                TSTB    @XCSR
                BMI     T17             ;RE CLR AND PROCEED
                JSR     R5,ERROR
                MOV     #EE16,MES
```

```
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                     ;LOCK FLOP NOT CLEARED BY
                JMP     THW             ;CLEAR ERROR
        THW:    MOV     #2,@XCSR        ;CLEAR LOCK FOR RECYCLE
                JSR     R2,WAITX
                TST     @XCINT          ;CLEAR FLAGS SHORT LOOP,SECOND CHANCE
                MOV     #000400,@XCSR           ;CLEAR RECIEVE FLAG
                RESET
                JMP     T13
                ;
                ;
                ;WITH RECEIVE AND SHORT LOOP FLAG SET PULSE TIMER
                ;AND WAIT FOR TIME OUT OF T2 THEN ISSUE AN CLEAR INSTRUCTION
                ;AND CHECK THAT ALL FLAGS ARE CLEARED.
```

```
                ;
T17:    JSR     R1,CHECK
        MOV     #340,PS
        MOV     #T17,LOOP       ;LOOP TESTS
        TST     @XCINT          ;CLEAR INTERRUPT FLAGS. SHORT LOOP,SECOND CHANCE
        MOV     #000400,@XCSR   ;CLEAR RECEIVE FLAG
        ;TEST SHORT LOOP INTERRUPT FLAG FOR BEING CLEARED
                ;
T18:    JSR     R1,CHECK
        TST     @XCSR           ;TEST BIT 15 OF CSR
        BPL     T19             ;OK GO ON
        JSR     R5,ERROR        ;CHECK FOR REPORT OF ERROR
        MOV     #EE18,MES       ;SET ERROR PRINT
        JSR     R3,PRINT
        JSR     R5,SCOPE
        HLT                     ;SHORT LOOP FLAG FAILED TO CLEAR
        JMP     T17             ;LOOP TO TEST 17 ON KEY "CONT"
        ;
        ;TEST WARNING FLAG (BIT 7) FOR BEING CLEAR
                ;
T19:    JSR     R1,CHECK
        TSTB    @XCSR           ;TEST BIT 07 OF CSR
        BPL     T20
        JSR     R5,ERROR        ;CHECK FOR ERROR REPORT
        MOV     #EE19,MES       ;SET ERROR PRINT
        JSR     R3,PRINT
        JSR     R5,SCOPE
        HLT
        JMP     T17             ;LOOP TO TEST 17 ON KEY "CONT"
        ;
        ;TEST ALARM RECEIVE (BIT 14) FOR BEING CLEAR
                ;
T20:    JSR     R1,CHECK
        BIT     #040000,@XCSR           ;TEST BIT 14 OF CSR
        BEQ     T21
        JSR     R5,ERROR        ;CHECK FOR ERROR REPORT
        MOV     #EE20,MES       ;SET PRINT ERROR T20
        JSR     R3,PRINT
        JSR     R5,SCOPE
        HLT                     ;BIT 14 OF CSR FAILED TO A CLR
        JMP     17              ;LOOP TO TEST 17 ON CONT
        ;
        ;ENABLE ALARM OUT SIGNAL (PROGRAM CONTROL) AND TEST
        ;THAT TIME OUT OF T3 WILL SET ALARM RECEIVE FLOP (BIT 14)
        ;THE ALARM RECEIVE SIGNAL MUST BE CONNECTED TO THE ALARM OUT
        ;AND THE PROGRAM CONTROLLED "RESET" STRAP MUST BE CONNECTED
                ;
T21:    JSR     R1,CHECK
        MOV     #T21,LOOP
        MOV     #340,PS
        MOV     #3,@XCSR                ;START TIMER AND ENABLE "ALARM OUT"
        JSR     R2,TOUT                 ;TIME OUT DELAYS
        BIT     #040000,@XCSR           ;CHECK BIT 14 FOR BEING SET
        BEQ     .+6
        JMP     T22
        JSR     R5,ERROR
```

```
                MOV     #EE21,MES       ;SET ERROR PRINTOUT
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                     ;RECEIVE ALARM FLOP FAILED TO SET
                JMP     T21             ;LOOP ON TEST 21
                ;
                ;
                ;CLEAR INTERRUPT FLAGS AND CHECK THAT "ALARM RECEIVE"
                ;FLAG WAS CLEARED
                ;
        T22:    JSR     R1,CHECK
                MOV     #340,@XCSR
                MOV     #T22,LOOP       ;SET SCOPE RETURN
                MOV     #000400,@XCSR   ;CLEAR ALARM RECEIVE FLAG
                BIT     #040000,@XCSR           ;TEST BIT 14 FOR BEING CLEAR
                BEQ     T23             ;FLAG CLEAR GO ON
```

```
                JSR     R5,ERROR
                MOV     #EE22,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                             ;BIT 14 OF CSR FAILED TO CLEAR
                JMP     T22
                ;
                ;CHECK "LOCKOUT" FEATURE OF "ALARM OUT" AFTER AN ALARM
                ;OUT SIGNAL HAS BEEN GENERATED.
                ;
T23:            JSR     R1,CHECK
                MOV     #340,PS
                MOV     #T23,LOOP
                MOV     #1,@XCSR                        ;START TIMING
                JSR     R2,TOUT                 ;TIME OUT DELAYS
                JSR     R2,WAITX                ;WAIT FOR REC TO -SET
                BIT     #040000,@XCSR                   ;TEST THAT LOCK OUT PREVENTED
                BEQ     T24                     ;BIT 14 FROM BEING SET
                JSR     R5,ERROR
                MOV     #EE23,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                             ;CLEAR ERROR FAILED TO ENABLE FLOP
                JMP     T23                     ;ALARM OUT SIGNAL
                ;
                ;DROP PROCESSOR STATUS TO 6 AND CHECK THAT WARNING FLAG WILL GIVE AN
                ;INTERRUPT AT VECTOR 170
                ;
T24:            TST     @XCINT                  ;CLEAR SHORT LOOP, SECOND CHANCE
                MOV     #000400,@XCSR                   ;CLEAR RECEIVE FLAG
                JSR     R1,CHECK
                MOV     #T25,R4                 ;SET INTERRUPT RETURN
                MOV     #WRNG,R5                ;SET FOR INTERRUPT AT WRONG VECTOR
                MOV     #T24,LOOP
                MOV     #101,@XCSR
                JSR     R2,TOUT
                MOV     PPRIV,PS                        ;SET STATUS TO PPRIV
                NOP                             ;WAIT FOR INTERRUPT
                NOP                             ;
                MOV     #340,PS                 ;SHOULD HAVE INTERRUPTED
                JSR     R5,ERROR                ;SET STATUS BACK TO SEVEN
                MOV     #EE24,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                             ;WARNING FLAG (BIT 07) FAILED
                JMP     T24                     ;TO PRODUCE AN INTERRUPT
                ;
                ;CLEAR FLAGS AND SET THE SHORT LOOP
                ;FLAG TO CHECK FOR AN INTERRUPT
                ;
                ;
```

```
                ;
T25:    JSR     R1,CHECK
        RESET
        TST     @XCINT          ;CLEAR INTERRUPTS
        MOV     #AA4,R4         ;SET INTERRUPT RETURN
        MOV     #WRNG,R5        ;SETFOR WRONG VECTOR
        MOV     #T25,LOOP       ;SET SCOPE MODE
        MOV     PPRIV,PS                ;GET STATUS DOWN TO PPRIV
        BIS     #102,@XCSR      ;INT.EN +EXT.EN
        INC     @XCSR           ;SET SHORT LOOP
        INC     @XCSR
        NOP                     ;SHOULD INT HERE
        NOP
        MOV     #340,PS         ;NO INT.GET STATUS BACK UP
        JSR     R5,ERROR
        MOV     #EE25,MES       ;SET ERROR PRINT
        JSR     R3,PRINT
        JSR     R5,SCOPE
        HLT                     ;SHORT LOOP FLAG FAILED TO RAISE A
        JMP     T25             ;INTERRUPT
AA4:    JSR     R2,WAITX                ;WAIT FOR TIME OUT CLR ERROR
        TST     @XCINT          ;CLEAR FLAGS SHORT LOOP, SECOND CHANCE
        MOV     #000400,@XCSR   ;CLEAR RECEIVE FLAG
        JMP     T26
        ;
        ;CHECK FOR ALARM RECEIVE FLAG TO GENERATE AN INTERRUPT
        ;BY CLEARING THE WARNING FLAG BEFORE THE RECEIVE
        ;FLAG IS SET. T3>T2 BY AT LEAST 50 MIRCO. SEC.
        ;
T26:    JSR     R1,CHECK
        MOV     #340,PS         ;SET PROCESSOR STATUS TO 7
        MOV     #AA5,R5         ;SET INTERRUPT RETURN
        MOV     #WRNG,R4        ;SET FOR WRONG VECTOR
        MOV     #AA4,LOOP       ;SET SCOPE
        MOV     #3,@XCSR                ;START TIMING, ENABLE ALARM OUT
        TSTB    @XCSR                   ;WAIT FOR WARNING
        BPL     .-4
        TST     @XCINT                  ;CLEAR BIT 7 OF CSR
        BIT     #040000,@XCSR           ;WAIT FOR ALARM RECEIVE FLAG
        BEQ     .-6
        BIS     #100,@XCSR              ;ENABLE INT.-2L
        MOV     PPRIV,PS                ;SET STATUS TO PPRIV
        NOP                     ;SHOULD INT. HERE
        NOP
        MOV     #340,PS         ;FAILED TO INT. SET STATUS TO 7
        JSR     R5,ERROR        ;CHECK FOR ERROR REPORT
        MOV     #EE26,MES       ;
        JSR     R3,PRINT
        JSR     R5,SCOPE        ;CHECK FOR SCOPE ROUTINE
        HLT
        JMP     AA4             ;CLR FLAGSBEFORE LOOP
AA5:    JSR     R2,TOUT         ;CLEAR POSSIBILITY OF SHORT LOOP
        MOV     #000400,@XCSR   ;CLEAR RECEIVE FLAG
        JMP     T27                     ;T 27
        ;
        ;
```

```
                ;CHECK LOCK OUT OF SHORT LOOP BY FIRST CAUSING A
                ;SHORT LOOP FLAG THEN READING STATUS OF RECEIVE TO CHECK
                ;THAT T3 HAS BEEN TERMINATED
                ;
        T27:    JSR     R1,CHECK
                MOV     #AA5,LOOP
                MOV     #340,PS
                TST     @XCINT
                CLR     @XCSR
                MOV     #3,@XCSR        ;SET SHORT LOOP
                INC     @XCSR
                JSR     R2,WAITX        ;5.MILLI.SEC.CLEAR LOCK  AND ENABLE
                BIT     #040000,@XCSR
                BNE     T28
                JSR     R5,ERROR
                MOV     #EE27,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                     ;SHORT LOOP LOCK FAILED TO TERMINATE T3
                JMP     AA5
                ;
                ;TEST THAT T2 WAS TERMINATED BY A SHORT LOOP LOCK BY
                ;TESTING STATUS OF T2
                ;
        T28:    JSR     R1,CHECK
                MOV     #T28,LOOP
                JSR     R2,TOUT
                MOV     #000400,@XCSR   ;CLEAR RECEIVE FLAG
                TST     @XCINT          ;CLR FLAGS SHORT LOOP, SECOND CHANCE
                MOV     #3,@XCSR        ;EXT EN. SET S.LOOP
                INC     @XCSR
                TSTB    @XCSR           ;TEST BIT 7
                BPL     T29
                JSR     R5,ERROR
                MOV     #EE28,MES
                JSR     R3,PRINT
                JSR     R3,SCOPE
                HLT                     ;LOCK FAILED TO TERMINATE T2
                JMP     T28             ;LOOP ONTEST 28
                ;
                ;
                ;WAIT 5 MILLI SEC. FOR CLEAR ERROR TO RELEASE
                ;THE LOCK FLOP AND PASS T2 TO BUS
                ;
                ;
        T29:    JSR     R1,CHECK
                JSR     R2,WAITX
                TSTB    @XCSR
                BMI     T30
                JSR     R5,ERROR
                MOV     #EE29,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                     ;LOCK DIDNT RELEASE T2 TO BUS
                JMP     T28             ;LOOP ON TEST 28
                ;
```

```
                        ;
                        ;
                        ;CHECK THAT LOCK FROM SHORT LOOP FLAG
                        ;LOCKS PULSE ING OF TIMER BY ATTEMPING TO SET SHORT
                        ;LOOP FLAG AFTER A PREVIOUS SHORT LOOP
                        ;
T30:    JSR     R1,CHECK
        MOV     #340,PS
        MOV     #T30,LOOP
        TST     @XCINT          ;CLEAR FLAGS SHORT LOOP,SECOND CHANCE
        INC     @XCSR
        INC     @XCSR
        TST     @XCINT          ;CLR FLAGS SHORT LOOP,SECOND CHANCE
        INC     @XCSR
        INC     @XCSR
        TST     @XCSR
        BPL     PTR
        JSR     R5,ERROR
        MOV     #EE30,MES
        JSR     R3,PRINT
        JSR     R5,SCOPE
        HLT                     ;SHORT LOOP LOCK FAILED TO LOCK TIMER
        JMP     T30             ;LOOP ON TEST 30
        ;
PTR:    MOV     #2,@XCSR
        JSR     R2,WAITX        ;CLR ERROR
        TST     @XCINT          ;CLEAR FLAGS SHORT LOOP,SECOND CHANCE
                        ;
                        ;
                        ;GENERATE A SHORT LOOP COMMAND AND CHECK THAT INITIALIZE
                        ;WILL CLEAR THE SHORT LOOP FLAG.
                        ;
                        ;
T31:    JSR     R1,CHECK
        MOV     #T31,LOOP
        INC     @XCSR           ;SET A SHORT LOOP
        INC     @XCSR
        RESET
        TST     @XCSR           ;TEST SHORT LOOP FLAG FOR CLEAR
        BPL     T32
        JSR     R5,ERROR
        MOV     #EE31,MES
        JSR     R3,PRINT
        JSR     R5,SCOPE
        HLT                     ;INITIALIZE FAILED TO CLEAR SHORT
        JMP     T31             ;LOOP FLAG
        ;
        ;ENABLE THE EXTERNAL WORLD AND ALLOW THE RECIEVE
        ;FLAG TO SET THEN CHECK THAT THE INITIALIZE
        ;CLEARED RECEIVE FLAG.
        ;
T32:    JSR     R1,CHECK
        MOV     #T32,LOOP
        MOV     #2,@XCSR
        JSR     R2,WAITX        ;5.MILLI SEC SET REC
        RESET
```

```
                BIT     #040000,@XCSR           ;TEST RECIEVE FLAG
                BEQ     T33
                JSR     R5,ERROR
                MOV     #EE32,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                             ;INITIALIZE FAILED TO CLEAR REC
                JMP     T32                     ;FLAG
                ;
                ;
                ;TEST THAT INITIALIZE WILL CLEAR THE LOCK FLOP
                ;BY SETTING A SHORT LOOP FLAG AND ISSUEING A RESET INSTR
                ;-UCTION. TEST THAT THE LOCK FLOP IS CLEARED BY SETTING
                ;A 2ND SHORT LOOP FLAG.
                ;
                ;
                ;
T33:            JSR     R1,CHECK
                MOV     #T33,LOOP
                INC     @XCSR
                INC     @XCSR
                RESET                           ;ATTEMPT TO CLEAR SHORT LOOP
                INC     @XCSR
                INC     @XCSR
                TST     @XCSR                   ;TEST SHOT LOOP
                BMI     PTX1
                JSR     R5,ERROR
                MOV     #EE33,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                             ;INITIALIZE FAILED TO CLEAR LOCK
                JMP     T33
PTX1:           MOV     #2,@XCSR
                JSR     R2,WAITX                        ;WAIT 5.M.S. FOR ERROR TO CLR
                ;
                ;PULSE TIMER AND ISSUE A RESET INSTRUCTION, WAIT 5.MILLI
                ;WITH EXTERNAL ENABLE SETAND TEST RECEIVE FLAG TO
                ;CHECK THAT INITIALIZE TERMINATED T3.
                ;
T34:            JSR     R1,CHECK
                MOV     #T34,LOOP
                MOV     #1,@XCSR                ;PULSE TIMER & EX. EN SET
                RESET
                MOV     #2,@XCSR
                JSR     R2,WAITX                ;WAIT 5. MILLI SEC FOR T4
                BIT     #040000,@XCSR           ;TEST RECEIVE FLAG
                BNE     PTX2
                JSR     R2,TOUT
                TST     @XCINT
                JSR     R5,ERROR
                MOV     #EE34,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                             ;INITIALIZE FAILED TO TERMINATE
                JMP     T34                     ;T3
PTX2:           JSR     R2,TOUT
```

```
                TST     @XCINT          ;CLEAR SHORT LOOP,SECOND CHANCE
                MOV     #000400,@XCSR   ;CLEAR RECEIVE FLAG
                ;
                ;PULSE TIMER AND ISSUE A RESET  INSTRUCTION
                ;WAIT FOR POSSIBLE TIME-OUT OF T2 THEN READ
                ;STATUS OF T2 TO TEST THAT INITIALIZE
                ;TERMINATED T2.
                ;
T35:            JSR     R1,CHECK
                MOV     #T35,LOOP
                MOV     #3,@XCSR
                RESET
                JSR     R2,TOUT         ;WAIT FOR DELAYS TO TIME OUT
                TSTB    @XCSR           ;TEST BIT 7
                BPL     PTX3
                TST     @XCINT          ;CLEAR FLAGS
                JSR     R5,ERROR
                MOV     #EE35,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                     ;INITIALIZE FAILED TO TERMINATE T2
                JMP     T35
PTX3:           TST     @XCINT          ;CLEAR FLAGS SHORT LOOP, SECOND CHANCE
                MOV     #000400,@XCSR   ;CLEAR RECEIVE FLAG
                ;
                ;TEST STATUS A&B AND TEST THAT BITS 7 & 15 ARE COMPLEMENTS
                ;15=1 AND 7=0 FROM A RESET CONDITION
                ;
T36:            JSR     R1,CHECK
                RESET
                MOV     #340,PS
                MOV     #T36,LOOP
                TST     @XECSR          ;CHECK BIT 15 OFECSR
                BPL     TILT
                TSTB    @XECSR          ;CHECK BIT 7 OF EXT CSR 15=1
                BMI     TILT            ;REPORT ERROR BOTH SAME
                JMP     T37             ;OK 15=1  7=0
TILT:           NOP
                JSR     R5,ERROR
                MOV     #EE36,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT             ;INCORECT STATUS IN EXT. CSR
                JMP     T36
                ;
                ;ISSUE A RESET INSTRUCTION AND TEST THAT STATUS
                ;OF THE EXT. CSR DOESNT CHANGE (NEUTRAL-POSITION):
                ;
T37:            JSR     R1,CHECK
                MOV     @XECSR,PPL              ;SAVE EXT STATUS
                MOV     #T37,LOOP
                RESET
                JSR     R2,WAITX                ;WAIT FO RELAY
                ADD     @XECSR,PPL      ;TEST FOR STATUS CHANGE
                BPL     T38
                JSR     R5,ERROR
```

```
                MOV     #EE37,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                     ;RESET INSTRUCTION CHANGED STATUS
                JMP     T37             ;OF EXT CSR
                ;
                ;
                ;WITH LATCH IN NEUTRAL POSITION SWITCH BUSS AND
                ;CHECK FOR CHANGE OF STATUS IN EXT. CSR
                ;
        T38:    JSR     R1,CHECK
                RESET
                MOV     #T38,LOOP
                MOV     #EE38S,MES
                MOV     @XECSR,PPL      ;SAVE EXT. STATUS
                INC     @XSWBU          ;SWITCH BUSS
                JSR     R2,WAITX
                JSR     R3,OPP          ;CHECK THAT BITS ARE OF OPPOSITE
                ADD     @XECSR,PPL      ;POLARITY
                BMI     T39
                JSR     R5,ERROR
                MOV     #EE38,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                     ;SWITCHING BUSS FAILED
                JMP     T38
                ;
                ;
                ;WITH BUS SWITCHED CHECK THAT LATCH WILL DE-ENERGIZE RELAY ON
                ;A SECOND SWITCHING OF THE BUS
                ;TEST THAT STATUS CHANGED (NEUTRAL POSITION)
                ;
                ;
        T39:    JSR     R1,CHECK
                MOV     #T39,LOOP
                MOV     @XECSR,PPL      ;SAVE EXT STATUS
                INC     @XSWBU          ;ATTEMPT TO SWITCH BUSS
                JSR     R2,WAITX        ;WAIT FOR RELAY
                ADD     @XECSR,PPL      ;WAS BUSS STATUS CHANGED
                BMI     T40
                JSR     R5,ERROR
                MOV     #EE39,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                     ;LATCH FAILED TO PREVENT BUSS CHANGE
                JMP     T39
                ;
                ;ISSUE A SWITCH BUS COMMAND AND CHECK THAT STATUS
                ;A & B CHANGED TO NEXT LATCH
                ;
                ;
        T40:    JSR     R1,CHECK
                MOV     #T40,LOOP
                MOV     @XECSR.PPL
                INC     @XSWBU          ;SWITCH BUS
                JSR     R2,WAITX                ;5 MILLI SEC FOR RELAY
```

```
                MOV     #EE40S,MES
                JSR     R3,OPP                  ;TEST FOR POL.CHANGE
                ADD     @XECSR,PPL              ;TEST THAT STATUS CHANGED
                BMI     T41
                JSR     R5,ERROR
                MOV     #EE40,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                             ;BUS FAILED TO SWITCH TO 2ND LATCH
                JMP     T40
                ;
                ;
                ;
                ;ENABLE EXTERNALWORLD AND ALLOW DELAYS TO TIME
                ;OUT TO CHECK THAT TERMINATION OF T3 WILL RELEASE THE LATCH
                ;
                ;
        T41:    JSR     R1,CHECK
                MOV     #T40,LOOP
                MOV     @XECSR,PPL              ;SAVE STATUS A&B
                MOV     #3,@XCSR                        ;ALLOW FOR CLEAR ERROR
                JSR     R2,TOUT                 ;TIME OUT ALLDELAYS
                JSR     R2,WAITX                ;5 MILLI.SEC FOR RELAYS
                MOV     #EE41S,MES              ;SET SUBTEST PRINT
                JSR     R3,OPP                  ;SUBTEST
                ADD     @XECSR,PPL
                BMI     T42                     ;IF STATUS CHANGED GO ON
                JSR     R5,ERROR                ;
                MOV     #EE41,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                             ;BUSS FAILED TO SWITCH TO SECOND LATCH
                JMP     T41
                ;
                ;
                ;
                ;ISSUE A SECOND SWITCH BUSS COMAND AND CHECK THAT SWBU (164006)
                ;WILL SWITCH BUS
                ;
        T42:    JSR     R1,CHECK
                MOV     #T42,LOOP
                MOV     @XECSR,PPL
                INC     @XSWBU                  ;ATTEMPT TO SWITCH BUSS
                JSR     R2,WAITX                ;5 MILLI SEC FOR RELAY
                ADD     @XECSR,PPL              ;DID STATUS CHANGE
                BMI     T43                     ;IF MINUS STATUS CHANGED
                JSR     R5,ERROR
                MOV     #EE42,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                             ;LATCH FAILED TO PREVENT BUSS FROM
                JMP     T42                     ;SWITCHING
                ;
                ;
                ;ISSUE A "RESET" INSTRUCTION AND TEST THAT
                ;LATCH  GETS CLEARED
```

```
                ;
        T43:    JSR     R1,CHECK
                MOV     #T43,LOOP
                MOV     @XECSR,PPL              ;SAVE STATUS
                MOV     #EE43S,MES              ;SET SUBTEST PRINT
                RESET                   ;CLEAR LATCH
                JSR     R2,WAITX        ;WAIT FOR RELAY
                JSR     R3,OPP
                ADD     @XECSR,PPL      ;TEST FOR STATUS CHANGE
                BMI     T44             ;IF MINUS STATUS CHANGED
                JSR     R5,ERROR
                MOV     #EE43,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT                     ;RESET FAILED TO RELEASE LATCH
                JMP     T43
                ;
                ;
                ;
        PPL:    0                       ;TEST DATA POINT
                ;
                ;
                ;TEST THAT "MANUAL" EXTERNAL ENABLE CIRCUITRY FUNCTIONS
                ;BY SETTING THE ENABLE FLOP FROM A SWITCHING OF THE BUS
                ;(VIA TEST CONNECTOR)
                ;
        T44:    JSR     R1,CHECK
                MOV     #T44,LOOP
                MOV     #3,@XCSR
                JSR     R2,TOUT         ;CLEAR ANY ERROR CONDITIONS
                TST     @XCINT          ;CLEAR FLAGS SHORT LOOP,SECOND CHANCE
                MOV     #000400,@XCSR   ;CLEAR RECEIVE FLAG
                INC     @XSWBU
                JSR     R2,WAITX        ;SWITCH BUS& WAIT FOR RELAY
                INC     @XCSR           ;GENERATE A SHORT LOOP ERROR,RECEIVE FLAG SHOULD
                INC     @XCSR           ;GET SET WITH EXTERNAL ENABLE SET BY SWITCH BUS COMMAND
                JSR     R2,WAITX
                BIT     #40000,@XCSR
                BNE     INIT
                JSR     R5,ERROR
                MOV     #EE44,MES
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT             ;MANUAL ENABLE INPUT CIRCUITRY FAILED
                JMP     T44
                ;
                ;
                ;
                ;INITIALIZE THE WORLD AND PRINT PASS FOR EACH
                ;PASS OF TEST CHECK TELETYPE FOR COMMAND INSTRUCTIONS
                ;
        INIT:   RESET                   ;INITIALIZE WORLD
        EOP:    MOV     #50.,TEMP2
        1$:     RESET
                DEC     TEMP2
```

```
                BNE     1$
                BIT     #SW06,SR
                BEQ     2$
                JSR     PC,EOPHLT
        2$:     MOV     #PASS,MES       ;SET PASS PRINT
                JSR     R3,PRINT        ;
                JSR     R1,CHECK        ;CHECK FOR TELTYPE COMMANDS
                MOV     #STACK,SP
                JMP     T1
        TEMP2:  .WORD 0
                ;
                ;
                ;SUBROUTINE TO TEST THAT STATUS A&B ARE OF ADJACENT
                ;POLARITY. THIS ROUTINE IS USED AS A SUBTEST.
                ;
        OPP:    TST     @XECSR          ;TEST BIT 15 OF ESCR
                BPL     .+12
                TSTB    @XECSR          ;BIT 15=1 TEST BIT 7
                BMI     .+14            ;REPORT A&B SAME
                RTS     R3
                TSTB    @XECSR  ;15=1 7=0
                BPL     .+4     ;REPORT BOTH SAME
                RTS     R3
                JSR     R5,ERROR
                NOP
                NOP
                JSR     R3,PRINT
                JSR     R5,SCOPE
                HLT
                JMP     @LOOP           ;LOOP ON FAILING TEST
                RTS     R3              ;OK 15=1 7=0
                TSTB    @XECSR          ;15=0 TEST 7
                BPL     .+4             ;REPORT BOTH SAME
                RTS     R3              ;OK 15=0 7=1
                JSR     R5,ERROR        ;CHECK FOR ERROR REPORT
                NOP
                NOP
                NOP
                JSR     R2,PRINT
                JSR     R5,SCOPE
                HLT
                JMP     @LOOP           ;LOOP ON FAILING TEST
                ;
                ;
                ;
```

```
                ;
                ;
                ;*********************************************************************
                ;DELAY ADJUSTMENT TEST
                ;*********************************************************************
DELAY:  MOV     #340,PS         ;KEEP STATUS AT 7 FOR ENTIRE TEST
                NULL
                RESET
                NULL
                MOV     #DELTST,MES     ;PRINT "DELAY ADJUST TEST"
                JSR     R3,PRINT
LREP:   MOV     #ASK,MES        ;ASK FOR DELAY 1, 2
                JSR     R3,PRINT
                JSR     R1,REC          ;RECIEVE ONE CHARACTER FROM TELE
                JSR     R3,ECHO         ;ECHO CHARACTER
                BIT     #SW01,SR        ;IS THERE A TTY?
                BEQ     2$
                BIT     #SW05,SR        ;ASK FOR DELAY 1 OR 2
                BNE     1$              ;IF SW05 UP,RUN DELAY 1
                MOV     #262,HLD
                BR      2$
1$:     MOV     #261,HLD
2$:     CMP     HLD,#261
                BEQ     D1              ;RUN DELAY 1 (RECEIVE) TIMING TEST
                CMP     HLD,#262
                BEQ     D2              ;RUN DELAY 2 (WARNING & S.L.) TIMING TEST
                MOV     #QU,MES         ;PRINT A ?
                JSR     R3,PRINT
                JMP     LREP            ;REPEAT LINE
                ;
                ;
D1:     MOV     #340,PS
                MOV     #000400,@XCSR   ;CLEAR RECEIVE FLAG
                MOV     #3,@XCSR        ;PULSE WORNING FLAG DELAY
                BIT     #040000,@XCSR   ;MONITOR BIT 14
                BEQ     .-6
                JSR     R2,TOUT         ;WAIT APPROX 10.-6 SEC
                JSR     R1,CHECK
                JMP     D1
                ;RUN WARNING FLAG & SHORT LOOP DELAY TIMING TESTS
                ;
D2:     MOV     #340,PS
                TST     @XCINT          ;CLEAR FLAG
                MOV     #1,@XCSR        ;PULSE ALARM DELAY
                TSTB    @XCSR           ;TEST BIT 7
                BPL     .-4
                JSR     R2,WAITX
                JSR     R1,CHECK        ;CHECK FOR TELETYPE COMMANDS
                JMP     D2              ;RECYCLE
                ;
                ;
```

```
                ;
                ;
                ;*****************************************************************
                ;DYNAMIC SYSTEM TEST
                ;*****************************************************************
                ;
DYMAN:  MOV     #DYTST,MES
        JSR     R3,PRINT
        NULL
        RESET
DYX:    JSR     R1,CHECK
        MOV     @XECSR,SIRE     ;SAVE STATUS
        MOV     #340,PS         ;SET STATUS
        MOV     #HORN,R4        ;SET INT. RETURN
        MOV     #HORN,R5
        MOV     #120,COUNT      ;SET COUNTER FOR 100 PASSES
PT:     MOV     #103,@XCSR      ;START TIMING, ENABLE ALARM OUT
        MOV     PPRIV,PS                ;STATUS=6
        NOP
        JMP     .-2             ;STAY HERE WAIT FOR INTERRUPT
HERE:   MOV     #SLF,MES        ;SHORT LOOP FLAG SET
        JSR     R3,PRINT
        JSR     R2,TOUT         ;TIME OUT DELAYS
        TST     @XCINT          ;CLEAR FLAGS
        JMP     PT              ;RECYCLE TEST
HORN:   TST     @XCSR           ;FIND WHO GAVE INTERRUPT
        BMI     HERE
        TSTB    @XCSR           ;CHECK BIT 7
        BMI     OKK             ;ALL RIGHT GO
        BIT     #040000,@XCSR   ;NOT BIT 7 SO TRY 14
        BEQ     ERR             ;ILLEGAL INT.
        TSTB    @XCSR
        BPL     .-4
OKK:    BIT     #40000,@XCSR    ;BIT 7=1 WAIT FOR 14
        BEQ     .-6
        TST     @XCINT          ;CLEAR FLAGS SHORT LOOP, SECOND CHANCE
        MOV     #000400,@XCSR   ;CLEAR RECEIVE FLAG
        INC     @XSWBU          ;SWITCH BUS
        JSR     R2,WAITX
        ADD     @XECSR,SIRE
        BMI     CLST            ;BR IF STATUS CHANGED
        NULL
        MOV     #BBS,MES        ;REPORT EXT. STATUS BAD
        JSR     R3,PRINT
        JMP     DYX
CLST:   MOV     @XECSR,SIRE
        RESET                   ;CLEAR LATCH
        JSR     R2,WAITX
        ADD     @XECSR,SIRE
        BMI     TIKE
        NULL
        MOV     #BBS,MES
        JSR     R3,PRINT
        JMP     DYX
TIKE:   DEC     COUNT
```

```
                BEQ     .+12                    ;BRANCH WHEN DONE
                JSR     R1,CHECK
                JMP     PT
                MOV     #340,PS
                NULL
                MOV     #PASS,MES
                JSR     R3,PRINT
                NULL
                JSR     R2,TOUT
                JSR     R2,TOUT
EOP2:           MOV     #50.,TEMP2
1$:             RESET
                DEC     TEMP2
                BNE     1$
                BIT     #SW06,SR
                BEQ     2$
                JSR     PC,EOPHLT
2$:             JMP     DYX
ERR:            MOV     #IINT,MES
                JSR     R3,PRINT
                JSR     R2,TOUT
                TST     @XCINT                  ;CLEAR SHORT LOOP,SECOND CHANCE
                MOV     #000400,@XCSR           ;CLEAR RECEIVE FLAG
                JMP     PT
COUNT:          0
SIRE:           0
                ;
                ;
                ;
                ;PRINT HEADER ROUTINE AND RECIEVE TELETYPE COMMANDS
                ;TO SPECIFY DESIRED TESTS
                ;
HEADER:         BIT     #SW01,SR                ;IS THERE A TTY?
                BEQ     1$
                BIT     #SW00,SR                ;IF CAME FROM "CHECK" SUBR
                BNE     1$                      ;THEN PRINT HEADER
                BIT     #SW02,SR                ;LOGIC TEST ?
                BNE     TA
                BIT     #SW03,SR                ;DELAY TEST ?
                BNE     TB
                BIT     #SW04,SR                ;DYNAMIC TEST ?
                BNE     TC
                HALT                            ;YOU SHOULD NEVER GET HERE
1$:             CLR     TKB                     ;CLEAR DONE OF READER STATUS
                MOV     #HEAD,MES
                JSR     R3,PRINT                ;ASK FOR DESIRED TEST
                JSR     R1,REC                  ;RECIEVE ONE CHARACTER
                JSR     R3,ECHO                 ;ECHO TELEREADER CHARACTER
                CMP     HLD,#261                ;TEST A=LOGIC TEST
                BEQ     TA
                CMP     HLD,#262                ;TEST B=TIME DELAY TEST
                BEQ     TB
                CMP     HLD,#263                ;TEST C=SYSTEM STATUS TEST
                BEQ     TC
                MOV     #QU,MES                 ;TYPE ?
                JSR     R3,PRINT
```

```
                JMP     HEADER          ;ILLEGAL CHARACTER TRY AGAIN
        TA:     JMP     LOGIC           ;LOGIC TEST
        TB:     JMP     DELAY           ;DELAY TIME LOOP
        TC:     JMP     DYMAN           ;DYNAMIC TEST
        HLD:    0                       ;HOLD TEST DESIRED
```

```
                ;
                ;ECHO CHARACTER IN TELETYPE BUFFER REGISTER
ECHO:   MOV     TKB,TYDB        ;READER BUFFER TO PRINTER BUFFER
        TSTB    @TYSR           ;TEST PRINTER STATUS
        BPL     .-4
        RTS     R3
        ;PRINT DATA SPECIFIED BY LEADING ADDRESS IN REGISTER 4
        ;UNTIL TERMINATED BY (@)
PRINT:  NULL
        MOV     MES,R4
        MOV     #ERRST,R2
        MOV     MES,(R2)+
        CLR     (R2)
1$:     MOVB    (R4)+,DBUF
        CMPB    #100,DBUF
        BEQ     DONE
        CMPB    #45,DBUF
        BNE     OUT
        JSR     R2,LFCR
        JMP     1$
OUT:    JSR     R5,PRT
        BR      PRINT+16
        ;
PRT:    MOVB    DBUF,TYDB
        TSTB    @TYSR
        BPL     .-4
        RTS     R5
DONE:   NULL
        RTS     R3
DBUF:   0
MES:    0
TEMP1:  .WORD 0
        ;
        ;
        ;RECEIVE ONE CHARACTER FROM TELETYPE READER
        ;BUFFER AND STORE IN LOC. "HLD"
        ;
REC:    MOV     HOLD,(PC)+
1$:     .WORD 0
2$:     TSTB    TKS             ;WAIT FOR KEYBOARD
        BMI     3$
        INC     #0
        BNE     2$
        DEC     1$
        BNE     2$
        MOVB    #215,HLD
        BR      4$
3$:     MOV     TKB,HLD         ;MOV DATA TO LOC. HLD
4$:     RTS     R1
HOLD:   20
        ;
        ;
        ;5 MILLI-SEC LOOP

WAIT:   MOV     #1750,TIMI
        DEC     TIMI
```

```
                BNE     .-4
                RTS     R2
                ;
                ;WAIT 5 MILLI SEC.
                ;
WAITX:          MOV     SPEED,TIMI      ;LOAD FOR 5 MILLISEC
                DEC     TIMI
                BNE     .-4
                RTS     R2
SPEED:          177777          ;5 MILLI SEC CONSTT
                ;
                ;
                ;RETURN FROM SHORT LOOP & WARNING INTERRUPTS
                ;
WAOG:           MOV     #STACK,SP       ;FORCE INTERRUPT RETURN OF SHORTLOOP & WARNING
                MOV     R4,PC
                HLT                     ;SHOULD NEVER GET HERE
                ;
                ;RETURN FROM RECIEVE FLAG INTERRUPTS
RECF:           MOV     #STACK,SP
                MOV     R5,PC           ;FORCE INT RET OF REC
                HLT
                ;
                ;
                ;
                ;
                ;
                ;CHECK FOR COMMANDS TO MONITOR
                ;TEST REQUIRED
                ;
                ;
CHECK:          BIT     #1,SR
                BNE     .+4
                RTS     R1
                MOV     #3,@XCSR
                JSR     R2,TOUT1
                RESET
                TST     @XCINT          ;CLEAR FLAGS
                JMP     HEADER
                ;
                ;
                ;
                ;
                ;WHEN VECTORS 170 & 174 ARE INCORRECTLY ASSERTED
                ;ENTER ROUTINE, REPORT ERROR AND RETURN TO FAILING TEST
                ;
                ;
WRNG:           MOV     #WRONG,MES
                JSR     R3,PRINT
                TST     (SP)+
                MOV     #340,@(SP)      ;SET RETURN STATUS TO 7
                TST     -(SP)
                RTI                     ;RETURN TO FAILING TEST
                ;
                ;BUSS TIME-OUT ERROR SUBROUTINE
                ;
```

```
                ;
TOBE:   JSR     R5,ERROR
        MOV     #TOER,MES       ;REPORT TIME OUT ERROR
        JSR     R3,PRINT
        NOP
        RTI
        ;
        ;
        ;SUBROUTINE FOR INPUTTING ADDRESSES AND VECTORS
        ;
        ;
ADDMO:  MOV     #ASK1,MES       ;ASK FOR "FIRST DEVICE ADD"
        JSR     R3,PRINT
        CLR     ADDRES
MOD:    JSR     R1,REC          ;REC A CHARACTER
        JSR     R3,ECHO
        CMP     #215,HLD        ;TEST FOR A C.R.
        BEQ     SET
        ASL     ADDRES
        ASL     ADDRES
        ASL     ADDRES
        BIC     #177770,HLD
        ADD     HLD,ADDRES
        JMP     MOD
SET:    TST     ADDRES
        BEQ     STRB
        MOV     ADDRES,XCSR     ;SET DEVICE ADDRESS
        ADD     #2,ADDRES
        MOV     ADDRES,XCINT
        ADD     #2,ADDRES
        MOV     ADDRES,XECSR
        ADD     #2,ADDRES
        MOV     ADDRES,XSWBU
STRB:   M
        MO.     #ASK2,MES       ;ASK FIRST INT. VECTOR
        JSR     R3,PRINT
        CLR     VADDR
MODA:   JSR     R1,REC
        JSR     R3,ECHO
        CMP     #215,HLD        ;TEST FOR C.R.
        BEQ     SETA
        ASL     VADDR
        ASL     VADDR
        ASL     VADDR
        BIC     #177770,HLD
        ADD     HLD,VADDR
        JMP     MODA
SETA:   TST     VADDR
        BNE     2$
        MOV     VADRS,VADDR
2$:     BIC     #177000,VADDR
        MOV     #WAOG,@VADDR
        ADD     #2,VADDR
        MOV     #340,@VADDR
        ADD     #2,VADDR
        MOV     #RECF,@VADDR
```

```
                ADD     #2,VADDR
                MOV     #340,@VADDR
        STRBA:  MOV     #ASK3,MES       ;ASK FOR VECTOR LEVEL
                JSR     R3,PRINT
                JSR     R1,REC          ;REC ONE CHARACTERS
                JSR     R3,ECHO
                BIT     #SW01,SR        ;IS THERE A TTY?
                BEQ     1$
                MOV     DPRIV,HLD       ;IF NO TTY
        1$:     CMP     #264,HLD        ;CHECK FOR A 4
                BNE     .+12
                MOV     #140,PPRIV      ;SET PRIORITY AT 3
                BR      PROTIM
                CMP     #265,HLD
                BNE     .+12
                MOV     #200,PPRIV
                BR      PROTIM
                CMP     #266,HLD
                BNE     .+12
                MOV     #240,PPRIV
                BR      PROTIM
                CMP     #267,HLD
                BNE     .+12
                MOV     #300,PPRIV
                BR      PROTIM
                MOV     #QU,MES
                JSR     R3,PRINT
                JMP     STRBA           ;TRY AGAIN
        PROTIM: MOV     #ASK4,MES
                JSR     R3,PRINT
                JSR     R1,REC
                JSR     R3,ECHO
                BIT     #SW01,SR        ;IS THERE A TTY ?
                BEQ     1$
                MOV     PDP11,HLD       ;PDP11-OS OR WHATEVER
        1$:     CMP     #301,HLD
                BNE     .+12
                MOV     #1500,SPEED
                BR      KA
                CMP     #302,HLD
                BNE     .+12
                MOV     #3000,SPEED
                BR      KA
                CMP     #303,HLD
                BNE     .+12
                MOV     #6000,SPEED
        KA:     RTS     R5
                MOV     #QU,MES
                JSR     R3,PRINT
                JMP     PROTIM

                .
        ADDRES: 0
        VADDR:  0
```

```
                ;
TIMEX:  MOV     #STACK,SP       ;REINITIALIZE THE STACK
        MOV     (R4),PC         ;FORCE RETURN FROM INTERRUPT
        HLT                     ;SHOULD NEVER GET HERE
        ;
        ;RUN DELAY TIME OUTS,TOTAL TIME AS PER
        ;PROCESSOR TIME 05=4SEC.+ 40=3 SEC+
        ;
        ;
TOUT:   TSTB    SR
        BR      TOUT1           ;CHANGE TO BPL TO RUN A FAST MODE FOR
        MOV     #12,CNTIM       ;T1,T2,T3<50 MILLI-SEC
RA:     MOV     SPEED,TIMI      ;50 MILLI SEC. WORTH
        DEC     TIMI
        BNE     .-4
        DEC     CNTIM
        BNE     RA
        RTS     R2              ;GO WHEN DONE
        ;RUN LONG TIME OUT FOR ALL DELAYS HERE
TOUT1:  MOV     #5,TIMI         ;SET COUNT TIME
        JSR     R4,TIME
        RTS     R2
CNTIM:  0
TIMI:   0
        ;
        ;
NULPRT: NOP
        TSTB    @TYSR
        BPL     .-4
        CLR     TYDB
        TSTB    @TYSR
        BPL     .-4
        CLR     TYDB
        TSTB    @TYSR
        BPL     .-4
        RTI
        ;
        ;
LFCR:   MOV     #215,TYDB       ;CARRIAGE RET.
        TSTB    @TYSR
        BPL     .-4
        MOV     #212,TYDB
        TSTB    @TYSR
        BPL     .-4
        RTS     R2
        ;
        ;
        ;
        ;CHECK FOR ERROR PRINT OUT (BIT 13 OF SWITCH REG.)
ERROR:  BIT     #20000,SR
        BEQ     .+6             ;IF BIT 13=0 PRINT ERROR
        ADD     #12,R5          ;ADD VALUE TO OMIT ERROR PRINT
        RTS     R5
        ;
        ;CHECK FOR SCOPE LOOP (BIT 14 OF SWITCH REG.)
SCOPE:  MOV     SR,ESTAT
```

```
                BIT     #04C000,ESTAT   ;TEST BIT 14 OF SWITCH REGISTER
                BEQ     .+12            ;IF BIT 14=1 RUN SCOPE
                MOV     #STACK,SP
                MOV     LOOP,PC         ;FORCE RETURN TO TEST
                RTS     R5              ;RETURN AND HLT
        ESTAT:  0
        LOOP:   0
                ;
                ;RUN TIME INCREMENTS IN VALUES STORED IN TIMI
        TIME:   CLR     XES
                INC     XES             ;COUNT LOOP
                BNE     .-4             ; ↑
                DEC     TIMI            ;NUMBER OF COUNT LOOPS
                BNE     .-12
                RTS     R4
        XES:    0
                ;MESSAGES
                .BYTE
        LT:     .ASCII  /%LOGIC TEST%@/

        EE1:    .ASCII  /T1%@/
        EE2:    .ASCII  /T2%@/
        EE3:    .ASCII  /T3%@/
        EE4:    .ASCII  /T4%@/
        EE5:    .ASCII  /T5%@/
        EE6:    .ASCII  /T6%@/
        EE7:    .ASCII  /T7%@/
        EE8:    .ASCII  /T8%@/
        EE9:    .ASCII  /T9%@/
        EE10:   .ASCII  /T10%@/
        EE11:   .ASCII  /T11%@/
        EE12:   .ASCII  /T12%@/
        EE13:   .ASCII  /T13%@/
        EE14:   .ASCII  /T14%@/
        EE15:   .ASCII  /T15%@/
        EE16:   .ASCII  /T16%@/
        EE17:   .ASCII  /T17%@/
        EE18:   .ASCII  /T18%@/
```

```
EE19:   .ASCII  /T19%@/
EE20:   .ASCII  /T20%@/
EE21:   .ASCII  /T21%@/
EE22:   .ASCII  /T22%@/
EE23:   .ASCII  /T23%@/
EE24:   .ASCII  /T24%@/
EE25:   .ASCII  /T25%@/
EE26:   .ASCII  /T26%@/
EE27:   .ASCII  /T27%@/
EE28:   .ASCII  /T28%@/
EE29:   .ASCII  /T29%@/
EE30:   .ASCII  /T30%@/
EE31:   .ASCII  /T31%@/
EE32:   .ASCII  /T32%@/
EE33:   .ASCII  /T33%@/
EE34:   .ASCII  /T34%@/
EE35:   .ASCII  /T35%@/
EE36:   .ASCII  /T36%@/
EE37:   .ASCII  /T37%@/
EE38:   .ASCII  /T38%@/
EE39:   .ASCII  /T39%@/
EE40:   .ASCII  /T40%@/
EE41:   .ASCII  /T41%@/
EE42:   .ASCII  /T42%@/
EE43:   .ASCII  /T43%@/
EE44:   .ASCII  /T44%@/
EE38S:  .ASCII  /T38S%@/
```

```
        EE40S:  .ASCII  /T40S%@/

        EE41S:  .ASCII  /T41S%@/

        EE43S:  .ASCII  /T41S%@/

        TOER:   .ASCII  /BUS ERROR TIME-OUT%@/


        PASS:   .ASCII  /PASS%@/

        WRONG:  .ASCII  /TRAPPED TO WRONG INT. VECTOR%@/



        DELTST: .ASCII  /%DELAY ADJUSTMENT TEST%@/

        ASK:    .ASCII  /DELAY:(1)WATCHDOG.(2)WARNING & SHORT LOOP%@/



        QU:     .ASCII  /?%@/
        HEAD:   .ASCII  /%LOGIC TEST(1),DELAY TEST(2),DYNAMIC TEST(3)%@/



        DYTST:  .ASCII  /%DYNAMIC TEST%@/

        SLF:    .ASCII  /SHORT LOOP FLAG SET%@/
```

```
IINT:   .ASCII  /ILLEGAL INTERRUPT%@/

BBS:    .ASCII  /EXT.BUS STATUS BAD%@/

ASK1:   .ASCII  /%FIRST DEVICE ADDRESS=@/

ASK2:   .ASCII  /%FIRST INT. VECTOR=@/

ASK3:   .ASCII  /%PRIORITY INT. LEVEL=@/

ASK4:   .ASCII  /%PDP-11 (A)=05,10,20 (B)=35,40 (C)=45,50 @/

        .EVEN
        .=017400
ERRST:  .BLKW 5
        .END
```

```
AA1     001322              627     628     635#
AA2     001432              650     651     658#
AA4     003446              985    1001#   1014    1031
AA5     003626             1012    1032#   1042    1056
ADDMO   007756              551    1767#
ADDRES  010634             1769*   1774*   1775*   1776*   1778*   1780    1782    1783*   1784    1785*   1786    1787*   1788
                           1864#
ASK     011627             1487    2064#
ASK1    012076             1767    2103#
ASK2    012125             1790    2108#
ASK3    012151             1814    2113#
ASK4    012177             1840    2117#
BBS     012052             1570    1579    2098#
CHECK   007650              580     595     609     625     647     669     684     702     719     735     751     765     780
                            801     815     833     858     865     877     889     904     923     940     960     982    1010
                           1041    1061    1082    1099    1126    1144    1166    1188    1213    1233    1253    1271    1294
                           1312    1334    1356    1374    1399    1435    1513    1523    1538    1584    1731#
CINT  = 172402              359#    533
CLST    006770             1568    1573#
CNTIM   010724             1877*   1881*   1888#
COUNT   007200             1543*   1582*   1607#
CSR   = 172400              358#    532
DBUF    007512             1657*   1658    1660    1667    1673#
DELAY   006254              556    1481#   1641
DELTST  011577             1485    2058#
DONE    007506             1659    1671#
DPRIV   000514              537#   1820
DYMAN   006530              560    1534#   1642
DYTST   011763             1534    2084#
DYX     006546             1538#   1572    1581    1600
Di      006420             1499    1507#   1514
D2      006466             1501    1517#   1524
ECHO    007370             1490    1630    1646#   1771    1794    1817    1843
ECSR  = 172404              360#    534
EE1     011134              586    1943#
EE10    011200              742    1961#
EE11    011205              757    1963#
EE12    011212              771    1965#
EE13    011217              791    1967#
EE14    011224              805    1969#
EE15    011231              821    1971#
EE16    011236              841    1973#
EE17    011243             1975#
EE18    011250              869    1977#
EE19    011255              881    1979#
EE2     011140              601    1945#
EE20    011262              893    1981#
EE21    011267              913    1983#
EE22    011274              931    1985#
EE23    011301              949    1987#
EE24    011306              971    1989#
EE25    011313              996    1991#
EE26    011320             1027    1993#
EE27    011325             1052    1995#
EE28    011332             1071    1997#
EE29    011337             1087    1999#
```

DDKWBA.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

```
EE3     011144            615    1947#
EE30    011344           1111    2001#
EE31    011351           1134    2003#
EE32    011356           1152    2005#
EE33    011363           1176    2007#
EE34    011370           1199    2009#
EE35    011375           1222    2011#
EE36    011402           1244    2013#
EE37    011407           1261    2015#
EE38    011414           1282    2017#
EE38S   011457           1274    2031#
EE39    011421           1302    2019#
EE4     011150            636    1949#
EE40    011426           1322    2021#
EE40S   011465           1317    2034#
EE41    011433           1345    2023#
EE41S   011473           1340    2037#
EE42    011440           1364    2025#
EE43    011445           1384    2027#
EE43S   011501           1377    2040#
EE44    011452           1413    2029#
EE5     011154            659    1951#
EE6     011160            674    1953#
EE7     011164            691    1955#
EE8     011170            710    1957#
EE9     011174            723    1959#
EOP     006052           1426#
EOPHLT  000250            543#   1432    1599
EOP2    007106           1593#
ERR     007144           1558    1601#
ERROR   011020            585     600     614     635     658     673     690     709     722     741     756     770     790
                          804     820     840     868     880     892     912     930     948     970     995    1026    1051
                         1070    1086    1110    1133    1151    1175    1198    1221    1243    1260    1281    1301    1321
                         1344    1363    1383    1412    1452    1464    1757    1915#
ERRST   017400           1654    2128#
ESTAT   011066           1921*   1922    1927#
HEAD    011705           1627    2075#
HEADER  007204            552    1615#   1639    1738
HERE    006626           1548#   1554
HLD     007366           1495*   1497*   1498    1500    1631    1633    1635    1643#   1689*   1691*   1772    1777*   1778
                         1795    1800*   1801    1820*   1821    1825    1829    1833    1846*   1847    1851    1855
HLT   = 000000            365#    385     387     389     391     393     395     397     399     401     403     405     407
                          409     411     413     415     417     419     421     423     425     427     429     431     433
                          435     437     439     441     443     445     447     449     451     453     455     457     459
                          461     463     465     467     469     471     473     475     477     479     481     483     485
                          487     489     491     493     495     497     499     501     503     505     507     509     511
                          589     604     618     640     662     677     694     713     726     745     760     774     794
                          808     824     844     872     894     896     916     934     952     974     999    1030    1055
                         1074    1090    1114    1137    1155    1179    1202    1225    1247    1264    1285    1305    1325
                         1348    1367    1387    1416    1458    1470    1716    1721    1869
HOLD    007570           1681    1693#
HORN    006654           1541    1542    1553#
IINT    012027           1601    2093#
INIT    006050           1411    1425#
KA      010614           1850    1854    1858#
LFCR    010766           1662    1904#
```

```
LOGIC    001000            563#  1640
LOOP     011070            582*   596*   610*   626*   670*   685*   703*   736*   752*   766*   782*   860*   905*
                           925*   942*   963*   987*  1014*  1042*  1062*  1101*  1127*  1145*  1167*  1189*  1214*
                          1236*  1255*  1273*  1295*  1313*  1335*  1357*  1375*  1400*  1459   1471   1925   1928#
LREP     006302           1487#  1504
LT       011117            573   1940#
MES      007514            573*   586*   601*   615*   636*   659*   674*   691*   710*   723*   742*   757*   771*
                           791*   805*   821*   841*   869*   881*   893*   913*   931*   949*   971*   996*  1027*
                          1052*  1071*  1087*  1111*  1134*  1152*  1176*  1199*  1222*  1244*  1261*  1274*  1282*
                          1302*  1317*  1322*  1340*  1345*  1364*  1377*  1384*  1413*  1433*  1485*  1487*  1502*
                          1534*  1548*  1570*  1579*  1588*  1601*  1627*  1637*  1653   1655   1674#  1747*  1758*
                          1767*  1790*  1814*  1837*  1840*  1859*
MOD      007774           1770#  1779
MODA     010150           1793#  1802
NOP    = 000240            364#
NULL   = 104000            363#   564    566    1482   1484   1536   1569   1578   1587   1590   1652   1671
NULPRT   010730            517   1892#
OKK      006706           1556   1561#
OPP      006134           1278   1318   1341   1380   1444#
OUT      007462           1661   1664#
PASS     011533           1433   1588   2048#
PC     =%000007            380#   544*  1432*  1599*  1681*  1715*  1720*  1868*  1925*
PDP11    000520            539#  1846
PPL      005724           1254*  1258*  1275*  1279*  1296*  1299*  1314*  1319*  1336*  1342*  1358*  1361*  1376*
                          1381*  1392#
PPRIV    000516            538#   630    652    966    988   1022   1545   1823*  1827*  1831*  1835*
PRINT    007406            574    587    602    616    638    660    675    692    711    724    743    758    772
                           792    806    822    842    870    882    894    914    932    950    972    997   1028
                          1053   1072   1088   1112   1135   1153   1177   1200   1223   1245   1262   1283   1303
                          1323   1346   1365   1385   1414   1434   1456   1468   1486   1488   1503   1535   1549
                          1571   1580   1589   1602   1628   1638   1652#  1665   1748   1759   1768   1791   1815
                          1838   1841   1860
PROTIM   010476           1824   1828   1832   1836   1840#  1861
PRT      007470           1664   1667#
PS     = 177776            354#   581*   630*   633*   648*   652*   655*   704*   781*   816*   859*   906*   941*
                           966*   969*   988*   994*  1011*  1022*  1025*  1043*  1100*  1235*  1481*  1507*  1517*
                          1540*  1545*  1586*
PT       006604           1544#  1552   1585   1606
PTR      004226           1109   1117#
PTX1     004506           1174   1181#
PTX2     004624           1195   1204#
PTX3     004732           1219   1227#
QU       011702           1502   1637   1837   1859   2073#
RA       010664           1878#  1882
REC      007520           1489   1629   1681#  1770   1793   1816   1842
RECF     007640            525   1719#  1811
R0     =%000000            373#
R1     =%000001            374#   580*   595*   609*   625*   647*   669*   684*   702*   719*   735*   751*   765*
                           780*   801*   815*   833*   858*   865*   877*   889*   904*   923*   940*   960*   982*
                          1010*  1041*  1061*  1082*  1099*  1126*  1144*  1166*  1188*  1213*  1233*  1253*  1271*
                          1294*  1312*  1334*  1356*  1374*  1399*  1435*  1489*  1513*  1523*  1538*  1584*  1629*
                          1692*  1733*  1770*  1793*  1816*  1842*
R2     =%000002            375#   649*   687*   706*   738*   788*   817*   835*   837*   847*   908*   944*   945*
                           965*  1001*  1032*  1048*  1063*  1083*  1118*  1147*  1182*  1193*  1196*  1204*  1217*
                          1257*  1277*  1298*  1316*  1338*  1339*  1360*  1379*  1402*  1406*  1409*  1468*  1512*
                          1522*  1550*  1566*  1575*  1591*  1592*  1603*  1654*  1655*  1656*  1662*  1701*  1708*
```

DDKWBA.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

```
                    1735*  1883*  1887*  1910*
R3      =%000003     376#   574*   587*   602*   616*   638*   660*   675*   692*   711*   724*   743*   758*
                     772*   792*   806*   822*   842*   870*   882*   894*   914*   932*   950*   972*   997*
                    1028*  1053*  1072*  1073*  1088*  1112*  1135*  1153*  1177*  1200*  1223*  1245*  1262*
                    1278*  1283*  1303*  1318*  1323*  1341*  1346*  1365*  1380*  1385*  1414*  1434*  1448*
                    1451*  1456*  1460*  1463*  1486*  1488*  1490*  1503*  1535*  1549*  1571*  1580*  1589*
                    1602*  1628*  1630*  1638*  1649*  1672*  1748*  1759*  1768*  1771*  1791*  1794*  1815*
                    1817*  1838*  1841*  1843*  1860*
R4      =%000004     377#   627*   650*   961*   985*  1013*  1541*  1653*  1657   1715   1868   1886*  1936*
R5      =%000005     378#   551*   585*   588*   600*   603*   614*   617*   628*   635*   639*   651*   658*
                     661*   673*   676*   690*   693*   709*   712*   722*   725*   741*   744*   756*   759*
                     770*   773*   790*   793*   804*   807*   820*   823*   840*   843*   868*   871*   880*
                     883*   892*   895*   912*   915*   930*   933*   948*   951*   962*   970*   973*   986*
                     995*   998*  1012*  1026*  1029*  1051*  1054*  1070*  1086*  1089*  1110*  1113*  1133*
                    1136*  1151*  1154*  1175*  1178*  1198*  1201*  1221*  1224*  1243*  1246*  1260*  1263*
                    1281*  1284*  1301*  1304*  1321*  1324*  1344*  1347*  1363*  1366*  1383*  1386*  1412*
                    1415*  1452*  1457*  1464*  1469*  1542*  1664*  1670*  1720   1757*  1858*  1917*  1918*
                    1926*
SCOPE   011036       588    603    617    639    661    676    693    712    725    744    759    773    793
                     807    823    843    871    883    895    915    933    951    973    998   1029   1054
                    1073   1089   1113   1136   1154   1178   1201   1224   1246   1263   1284   1304   1324
                    1347   1366   1386   1415   1457   1469   1921#
SET     010050      1773   1780#
SETA    010224      1796   1803#
SIRE    007202      1539*  1567*  1573*  1576*  1608#
SLF     012002      1548   2088#
SP      =%000006     379#   550*   555*   559*   563*  1436*  1714*  1719*  1749   1750*  1751   1867*  1924*
SPEED   007626      1705   1709#  1849*  1853*  1857*  1878
SR      = 177570     362#  1430   1491   1493   1597   1615   1617   1619   1621   1623   1731   1818   1844
                    1875   1915   1921
STACK   000600       548#   550    555    559    563   1436   1714   1719   1867   1924
STRB    010130      1781   1789#
STRBA   010320      1814#  1839
SWBU    = 172406     361#   535
SWOO    = 000001     366#  1617
SWO1    = 000002     367#  1491   1615   1818   1844
SWO2    = 000004     368#  1619
SWO3    = 000010     369#  1621
SWO4    = 000020     370#  1623
SWO5    = 000040     371#  1493
SWO6    = 000100     372#  1430   1597
TA      007352      1620   1632   1640#
TB      007356      1622   1634   1641#
TC      007362      1624   1636   1642#
TEMP1   007516      1675#
TEMP2   006132      1426*  1428*  1438#  1593*  1595*
TEST    001016       567    573#
THW     002474       782    795    809    825    845    846#
TIKE    007034      1577   1582#
TILT    005006      1238   1240   1242#
TIME    011072      1886   1931#
TIMEX   010640      1867#
TIMI    010726      1698*  1699*  1705*  1706*  1878*  1879*  1885*  1889#  1934*
TKB     = 177562     355#  1626*  1646   1691
TKS     = 177560     356#  1683
TOBE    007734       514   1757#
```

```
TOER    011507          1758    2043#
TOUT    010650           649     687     706     788     837     908     944     965    1032    1063    1196    1204    1217
                        1338    1402    1512    1550    1591    1592    1603    1875#
TOUT1   010710          1735    1876    1885#
TYDB  = 177566           357#   1646*   1667*   1895*   1898*   1904*   1907*
TYSR    000500           531#   1647    1668    1893    1896    1899    1905    1908
T1      001030           580#    582     590    1437
T10     001752           721     735#
T11     002036           740     751#    752     761     766
T12     002112           755     765#    775
T13     002172           769     780#    851
T14     002270           789     801#
T15     002[illegible]   803     815#
T16     002410           819     833#
T17     002526           839     858#    860     873     885
T18     002560           865#
T19     002622           867     877#
T2      001110           584     595#    596     605
T20     002664           879     889#
T21     002730           891     904#    905     917
T22     003026           911     923#    925     935
T23     003114           928     940#    942     953
T24     003212           947     958#    963     975
T25     003330           961     982#    987    1000
T26     003470          1004    1010#
T27     003644          1034    1041#
T28     003752          1050    1061#   1062    1075    1091
T29     004052          1069    1082#
T3      001170           599     609#    610     619
T30     004120          1085    1099#   1101    1115
T31     004244          1126#   1127    1138
T32     004326          1132    1144#   1145    1156
T33     004414          1150    1166#   1167    1180
T34     004520          1188#   1189    1203
T35     004642          1213#   1214    1226
T36     004744          1233#   1236    1248
T37     005040          1241    1253#   1255    1265
T38     005126          1259    1271#   1273    1286
T39     005232          1280    1294#   1295    1306
T4      001246           613     625#    626     641
T40     005322          1300    1312#   1313    1326    1335
T41     005424          1320    1334#   1349
T42     005534          1343    1356#   1357    1368
T43     005624          1362    1374#   1375    1388
T44     005726          1382    1399#   1400    1417
T5      001352           634     647#    663
T6      001462           657     669#    670     678
T7      001534           672     684#    685     695
T8      001616           689     702#    703     714     736     746
T9      001706           708     719#    727
VADDR   010636          1792*   1797*   1798*   1799*   1801*   1803    1805*   1806*   1807*   1808*   1809*   1810*   1811*
                        1812*   1813*   1865#
VADRS   000512           536#   1805
WAIT    007572          1698#
WAITX   007610           738     817     835     847     945    1001    1048    1083    1118    1147    1182    1193    1257
                        1277    1298    1316    1339    1360    1379    1406    1409    1522    1566    1575    1705#
```

M04


```
WAOG    007630          523    1714#  1807
WRNG    007706          962     986   1013   1747#
WRONG   011541         1747    2051#
XCINT   000504          533#    686    753    848    861    958    984   1002   1018   1044   1065   1102   1105
                       1119    1197   1205   1220   1227   1403   1518   1551   1563   1604   1737   1784#
XCSR    000502          532#    583    597#   598    611#   612    629#   656#   671    688    705#   707    720
                        737#    739    754    767#   768    783#   784#   785    802    818    834#   836#   838
                        846#    849#   862#   866    878    890    907#   909    924#   926#   927    943#   946
                        959#    964#   989#   990#   991#  1003#  1015#  1016   1019   1021#  1033#  1045#  1046#
                       1047#   1049   1064#  1066#  1067#  1068   1084   1103#  1104#  1106#  1107#  1108   1117#
                       1128#   1129#  1131   1146#  1149   1168#  1169#  1171#  1172#  1173   1181#  1190#  1192#
                       1194    1206#  1215#  1218   1228#  1337#  1401#  1404#  1407#  1408#  1410   1508#  1509#
                       1510    1519#  1520   1544#  1553   1555   1557   1559   1561   1564#  1605#  1734#  1782#
XECSR   000506          534#   1237   1239   1254   1258   1275   1279   1296   1299   1314   1319   1336   1342
                       1358    1361   1376   1381   1444   1446   1449   1461   1539   1567   1573   1576   1786#
XES     011114         1931#   1932#  1937#
XSWBU   000510          535#   1276#  1297#  1315#  1359#  1405#  1565#  1788#
.     = 017412          383#    384    386    388    390    392    394    396    398    400    402    404    406
                        408     410    412    414    416    418    420    422    424    426    428    430    432
                        434     436    438    440    442    444    446    448    450    452    454    456    458
                        460     462    464    466    468    470    472    474    476    478    480    482    484
                        486     488    490    492    494    496    498    500    502    504    506    508    510
                        513#    516#   522#   530#   542#   547#   549#   554#   558#   562#   787    910   1017
                       1020    1445   1447   1450   1462   1511   1521   1547   1560   1562   1583   1648   1669
                       1700    1707   1732   1822   1826   1830   1834   1848   1852   1856   1880   1894   1897
                       1900    1906   1909   1916   1923   1933   1935   2126#  2127#  2128#
```

DDKWBA.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

```
ADD     1258  1279  1299  1319  1342  1361  1381  1567  1576  1778  1783  1785  1787  1801  1808
        1810  1812  1917
ASL     1774  1775  1776  1797  1798  1799
BEQ      584   613   672   689   721   769   819   891   910   928   947  1020  1150  1431  1492
        1499  1501  1511  1558  1562  1583  1598  1616  1632  1634  1636  1659  1773  1781  1796
        1819  1845  1916  1923
BIC      656  1777  1800  1806
BIS      629   989  1021
BIT      583   598   612   671   688   720   739   768   818   890   909   927   946  1019  1049
        1149  1194  1410  1430  1491  1493  1510  1557  1561  1597  1615  1617  1619  1621  1623
        1731  1818  1844  1915  1922
BMI      708   839  1085  1174  1240  1280  1300  1320  1343  1362  1382  1447  1554  1556  1568
        1577  1684
BNE      599   740  1050  1195  1411  1429  1494  1596  1618  1620  1622  1624  1661  1686  1688
        1700  1707  1732  1804  1822  1826  1830  1834  1848  1852  1856  1880  1882  1933  1935
BPL      755   787   803   867   879  1017  1069  1109  1132  1219  1238  1259  1445  1450  1462
        1521  1560  1648  1669  1894  1897  1900  1906  1909
BR      1496  1665  1690  1824  1828  1832  1836  1850  1854  1876
CLR      611  1045  1626  1656  1769  1792  1895  1898  1931
CMP     1498  1500  1631  1633  1635  1772  1795  1821  1825  1829  1833  1847  1851  1855
CMPB    1658  1660
DEC     1428  1582  1595  1687  1699  1706  1879  1881  1934
HALT     543  1625
INC      783   784   836   990   991  1047  1067  1103  1104  1106  1107  1128  1129  1168  1169
        1171  1172  1276  1297  1315  1359  1405  1407  1408  1565  1685  1932
JMP      552   556   560   567   590   605   619   634   641   657   663   678   695   714   727
         746   761   775   789   795   809   825   845   851   873   885   897   911   917   935
         953   975  1000  1004  1031  1034  1056  1075  1091  1115  1138  1156  1180  1203  1226
        1241  1248  1265  1286  1306  1326  1349  1368  1388  1417  1437  1459  1471  1504  1514
        1524  1547  1552  1572  1581  1585  1600  1606  1639  1640  1641  1642  1663  1738  1779
        1802  1839  1861
JSR      551   574   580   585   587   588   595   600   602   603   609   614   616   617   625
         635   638   639   647   649   658   660   661   669   673   675   676   684   687   690
         692   693   702   706   709   711   712   719   722   724   725   735   738   741   743
         744   751   756   758   759   765   770   772   773   780   788   790   792   793   801
         804   806   807   815   817   820   822   823   833   835   837   840   842   843   847
         858   865   868   870   871   877   880   882   883   889   892   894   895   904   908
         912   914   915   923   930   932   933   940   944   945   948   950   951   960   965
         970   972   973   982   995   997   998  1001  1010  1026  1028  1029  1032  1041  1048
        1051  1053  1054  1061  1063  1070  1072  1073  1082  1083  1086  1088  1089  1099  1110
        1112  1113  1118  1126  1133  1135  1136  1144  1147  1151  1153  1154  1166  1175  1177
        1178  1182  1188  1193  1196  1198  1200  1201  1204  1213  1217  1221  1223  1224  1233
        1243  1245  1246  1253  1257  1260  1262  1263  1271  1277  1278  1281  1283  1284  1294
        1298  1301  1303  1304  1312  1316  1318  1321  1323  1324  1334  1338  1339  1341  1344
        1346  1347  1356  1360  1363  1365  1366  1374  1379  1380  1383  1385  1386  1399  1402
        1406  1409  1412  1414  1415  1432  1434  1435  1452  1456  1457  1464  1468  1469  1486
        1488  1489  1490  1503  1512  1513  1522  1523  1535  1538  1549  1550  1566  1571  1575
        1580  1584  1589  1591  1592  1599  1602  1603  1628  1629  1630  1638  1662  1664  1735
        1748  1757  1759  1768  1770  1771  1791  1793  1794  1815  1816  1817  1838  1841  1842
        1843  1860  1886
MOV      550   555   559   563   573   581   582   586   596   597   601   610   615   626   627
         628   630   633   636   648   650   651   652   655   659   670   674   685   691   703
         704   705   710   723   736   737   742   752   757   766   767   771   781   782   791
         805   816   821   834   841   846   849   859   860   862   869   881   893   905   906
         907   913   924   925   926   931   941   942   943   949   959   961   962   963   964
         966   969   971   985   986   987   988   994   996  1003  1011  1012  1013  1014  1015
```

BOS


```
            1022    1025    1027    1033    1042    1043    1046    1052    1062    1064    1066    1071    1087    1100    1101
            1111    1117    1127    1134    1145    1146    1152    1167    1176    1181    1189    1190    1192    1199    1206
            1214    1215    1222    1228    1235    1236    1244    1254    1255    1261    1273    1274    1275    1282    1295
            1296    1302    1313    1314    1317    1322    1335    1336    1337    1340    1345    1357    1358    1364    1375
            1376    1377    1384    1400    1401    1404    1413    1426    1433    1436    1481    1485    1487    1495    1497
            1502    1507    1508    1509    1517    1519    1534    1539    1540    1541    1542    1543    1544    1545    1548
            1564    1570    1573    1579    1586    1588    1593    1601    1605    1627    1637    1646    1653    1654    1655
            1681    1691    1698    1705    1714    1715    1719    1720    1734    1747    1750    1758    1767    1782    1784
            1786    1788    1790    1805    1807    1809    1811    1813    1814    1820    1823    1827    1831    1835    1837
            1840    1846    1849    1853    1857    1859    1867    1868    1877    1878    1885    1904    1907    1921    1924
            1925
MOVB        1657    1667    1689
NOP          631     632     653     654     967     968     992     993    1023    1024    1242    1453    1454    1455    1465
            1466    1467    1546    1760    1789    1892
RESET        565     850     983    1130    1148    1170    1191    1216    1234    1256    1272    1378    1425    1427    1483
            1537    1574    1594    1736
RTI         1752    1761    1901
RTS          544    1448    1451    1460    1463    1649    1670    1672    1692    1701    1708    1733    1858    1883    1887
            1910    1918    1926    1936
TST          686     753     785     848     861     866     958     984    1002    1018    1044    1065    1102    1105    1108
            1119    1131    1173    1197    1205    1220    1227    1237    1403    1444    1518    1551    1553    1563    1604
            1737    1749    1751    1780    1803
TSTB         707     754     802     838     878    1016    1068    1084    1218    1239    1446    1449    1461    1520    1555
            1559    1647    1668    1683    1875    1893    1896    1899    1905    1908
.ASCII      1940    1943    1945    1947    1949    1951    1953    1955    1957    1959    1961    1963    1965    1967    1969
            1971    1973    1975    1977    1979    1981    1983    1985    1987    1989    1991    1993    1995    1997    1999
            2001    2003    2005    2007    2009    2011    2013    2015    2017    2019    2021    2023    2025    2027    2029
            2031    2034    2037    2040    2043    2048    2051    2058    2064    2073    2075    2084    2088    2093    2098
            2103    2108    2113    2117
.BLKW       2128
.BYTE       1939
.ENABL         4
.END        2129
.EVEN       2126
.LIST          4
.NLIST         4
.REM           5
.REPT        384
.WORD       1438    1675    1682


ERRORS DETECTED:  0
DEFAULT GLOBALS GENERATED:  0

*DDKWBA,DDKWBA.SEQ/SOL/CRF/DS:ERFZ/EN:ABS=DSKM:DDKWBA.P11
RUN-TIME:  5  9  3 SECONDS
RUN-TIME RATIO: 30/18=1.6
CORE USED:  8K  (15 PAGES)
```

C05