PDP11/45

PIRQ INTERRUPT TEST

MD-11-DCKBN-B

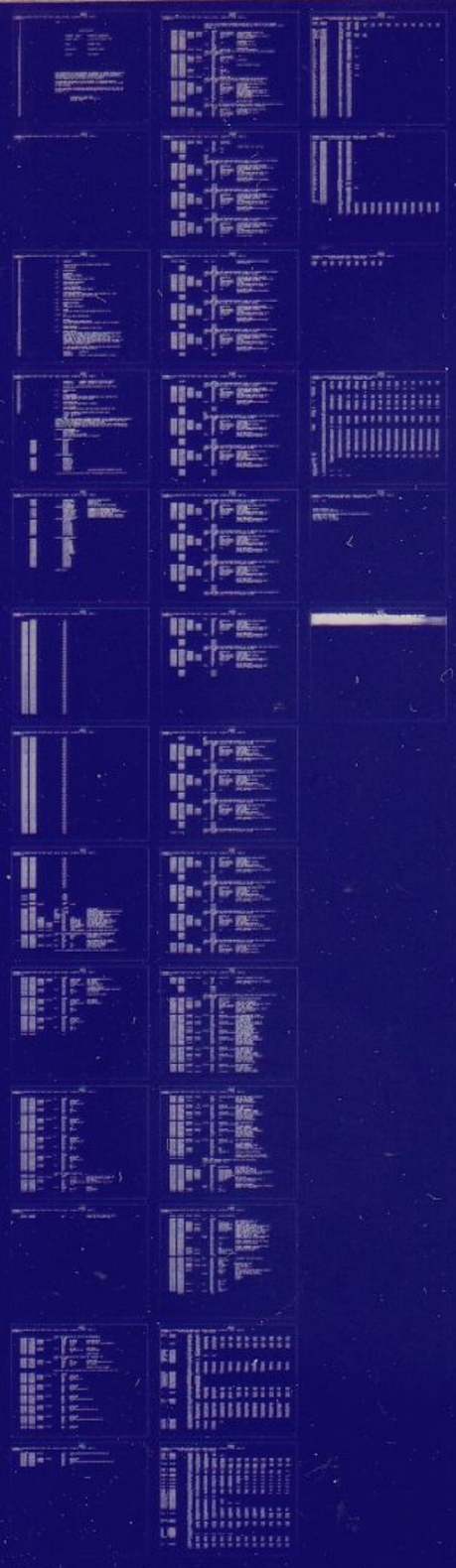
EP-DCKBN-B-DL-A

OCT 1976

digital

Made in U.S.A.

FICHE 1 OF 1



```
        .REM    %

                        IDENTIFICATION

                PRODUCT   CODE:         MAINDEC-11-DCKBN-B-D

                PRODUCT NAME:           11/45 PIRQ INTERUPT TEST

                DATE:                   JANUARY 1976

                MAINTAINER:             DIAGNOSTIC GROUP

                AUTHOR:                 JOHN ADAMS
```

```
    57
```

```
                                1.0     ABSTRACT

                                        THIS IS A TEST OF THE PROGRAM INTERUPT REQUEST
                                        (PIRQ) LOGIC.

                                2.0     REQUIREMENTS

                                2.1     EQUIPMENT
                                        BASIC 11/45 SYSTEM
                                2.2     STORAGE
                                        THIS PROGRAM USES 0 THRU 17500

                                2.3     PRELIMINARY PROGRAMS
                                        DOAA THRU DOMA

                                3.0     LOADING PROCEDURE
                                        LOAD PROGRAM USING ABS LOADER

                                4.0     STARTING PROCEDURE
                                        LOAD ADDRESS 200. PRESS START. THE PROGRAM WILL LOOP
                                        AND RING BELL ON PASS COMPLETION.

                                5.0     OPERATING PROCEDURE

                                5.1     SWITCH SETTINGS
                                        NONE
                                5.2     SUBROUTINE ABSTRACTS

                                5.2.1   SCOPE
                                        SCOPE IS A MOVE PC,R1 AND STORES THE PC+2 IN R1.

                                5.2.2   HLT
                                        HLT IS A HALT INSTRUCTION.

                                6.0     ERRORS
                                        ALL ERRORS WILL CAUSE A HALT
                                        TRAP AND INTERRUPT ERRORS WILL CAUSE A HALT AT VECTOR+2.

                                6.1     ERROR RECOVERY
                                        PRESS CONTINUE TO PROCEDE TO NEXT TEST

                                6.2     ERROR LOOPING
                                        TO LOOP ON AN ERROR, PLACE A BRANCH TO THE PREVIOUS SCOPE
                                        INSTRUCTION IN PLACE OF THE HALT INSTRUCTION.
                                        NOTE THAT IF THE ERROR IS INTERMITTANT THAT THE TEST WILL
                                        DROP THRU THE HALT AND PROCEED TO THE NEXT TEST.
                                        THEREFORE, TO LOOP THE TEST CONTINUOSLY REPLACE THE BEQ .+4
                                        INSTRUCTION IMMEDIATLY PRECEEDING THE HALT WITH A BRANCH
                                        BACK TO THE PREVIOUS SCOPE.

                                        TO LOOP ON TRAP FAILURES, PATCH IN THE FOLLOWING ROUTINE
                                        AT THE ADDRESS OF THE TRAP VECTOR.

                                        TRAPVEC:        TRAPVEC+4
                                        TRAPVEC+2:      0
                                        TRAPVEC+4:      012716  ;MOVE SCOPE ADDRESS TO STACK
```

```
                                        TRAPVEC+6:      ADDRESS ;ADDRESS OF PREVIOUS SCOPE
                                        TRAPVEC+10:     000006  ;RETURN TO TEST AT SCOPE

                                        RESTORE ALL LOCATIONS BEFORE PROCEEDING TO NEXT TEST.

                                7.0     RESTRICTIONS
                                        NONE

                                8.0     MISCELLANEOUS
                                        ON TRAP ERRORS THE STACK POINTER(R6) WILL CONTAIN THE
                                        ADDRESS WHERE THE TRAP OCCURED.

                                8.1     EXECUTION TIME
                                        THIS PROGRAM TAKES ABOUT 1 MINUTE.

                                8.2     STACK POINTER
                                        THIS PROGRAM INITIALY SETS THE STACK POINTER AT 500.
                                        %
                                        .TITLE  MAINDEC-11-DCKBN-B PROG INT RQST LOGIC
                                        .NLIST  MC,MD,SEQ
                                        .LIST   ME
                                        .ABS
                                ;DCKBNA- TESTS THAT PROGRAM INTERRUPT REQUEST (PIRQ) HARDWARE WORKS PROPERLY.
                                ;THE PIRQ LOGIC 'BOOKS' AN INTERRUPT AND WHEN THE PRIORITY LEVEL FALLS
                                ;BELOW THAT OF THE REQUEST THE INTERRUPT IS TAKEN
                                ;NOTE: IT IS NOT SPECIFIED AS TO WHETHER OR NOT THE NEXT INSTRUCTION IS
                                ;EXECUTED IF THE 'BOOKED' REQUEST IS GREATER THAN THAT OF THE PROCESSER.

                                ;STARTING PROCEEDURE
                                ;       LOAD ADDRESS=200
                                ;       PRESS START
                                ;       STACK POINTER IS AT 500
                                ;       BELL WILL RING WHEN TEST IS COMPLETE

                                ;EQUATE STATEMENTS
000000                                  R0=%0
000001                                  R1=%1
000002                                  R2=%2
000003                                  R3=%3
000004                                  R4=%4
000005                                  R5=%5
000006                                  SP=%6
000007                                  PC=%7
                                ;VECTOR ADDRESSES

000004                                  ERRVEC=4
000010                                  RESVEC=10
000014                                  TBITVEC=14
000020                                  IOTVEC=20
000024                                  PFVEC=24
000030                                  EMTVEC=30
000034                                  TRPVEC=34
000064                                  TPVEC=64                ;TELETYPE PRINTER INTERRUPT VECTOR

                                ;**********************INITIAL STACK POINTER=500*************************
```

```
000500                          STKPTR=500              ;INITIAL STACK POINTER
010701                          SCOPE=010701            ;MOVE PC TO R1
000000                          HLT=HALT                ;ERROR HALT
 22626                          POP2=[illegible]26      ;POPS TWO WORD OFF THE STACK
                        ;RESISTER ADDRESSES
177776                          PSW=177776              ;ADDRESS OF PROCESSOR STATUS
177770                          [illegible]K=177770     ;ADDRESS OF MICRO BREAK REGISTER
177564                          TPCSR=177564            ;ADDRESS OF TELEPRINTER CONTROL STATUS
177566                          TPBUF=177566            ;AND DATA BUFFER REGISTER
177570                          SWR=177570              ;ADDRESS OF CONSOLE SWITCH REGISTER
177570                          DISPLAY=177570          ;ADDRESS OF CONSOLE DISPLAY REGISTER
                        ;PSW BIT ASSIGNMENTS
000000                          PRTY0=0
000040                          PRTY1=40
000100                          PRTY2=100
000140                          PRTY3=140
000200                          PRTY4=200
000240                          PRTY5=240
000300                          PRTY6=300
000340                          PRTY7=340
000340                          PRTY10=340

                        ;PIRQ BIT ASSIGNMENTS
000000                          PIRQ=0
001000                          PIR1=1000
002000                          PIR2=2000
004000                          PIR3=4000
010000                          PIR4=10000
020000                          PIR5=20000
040000                          PIR6=40000
100000                          PIR7=100000
100000                          PIR10=100000
000000                          PIA0=0
000042                          PIA1=42
000104                          PIA2=104
000146                          PIA3=146
000210                          PIA4=210
000252                          PIA5=252
000314                          PIA6=314
000356                          PIA7=356

                        ;MACRO CALLS
```

```
000000  000000          .=0
000000  000002          HALT
000002  000000          .+2
000004  000006          HALT
000006  000010          .+2
000010  000012          HALT
000012  000014          .+2
000014  000016          HALT
000016  000020          .+2
000020  000022          HALT
000022  000024          .+2
000024  000026          HALT
000026  000030          .+2
000030  000032          HALT
000032  000034          .+2
000034  000036          HALT
000036  000040          .+2
000040  000042          HALT
000042  000044          .+2
000044  000046          HALT
000046  000050          .+2
000050  000052          HALT
000052  000054          .+2
000054  000056          HALT
000056  000060          .+2
000060  000062          HALT
000062  000064          .+2
000064  000066          HALT
000066  000070          .+2
000070  000072          HALT
000072  000074          .+2
000074  000076          HALT
000076  000100          .+2
000100  000102          HALT
000102  000104          .+2
000104  000106          HALT
000106  000110          .+2
000110  000112          HALT
000112  000114          .+2
000114  000116          HALT
000116  000120          .+2
000120  000122          HALT
000122  000124          .+2
000124  000126          HALT
000126  000130          .+2
000130  000132          HALT
000132  000134          .+2
000134  000136          HALT
000136  000140          .+2
000140  000142          HALT
000142  000144          .+2
000144  000146          HALT
000146  000150          .+2
000150  000152          HALT
000152  000000          HALT
```

```
000154  000156                  .+2
000156  000000                  HALT
000160  000162                  .+2
000162  000000                  HALT
000164  000166                  .+2
000166  000000                  HALT
000170  000172                  .+2
000172  000000                  HALT
000174  000176                  .+2
000176  000000                  HALT
000200  000202                  .+2
000202  000000                  HALT
000204  000206                  .+2
000206  000000                  HALT
000210  000212                  .+2
000212  000000                  HALT
000214  000216                  .+2
000216  000000                  HALT
000220  000222                  .+2
000222  000000                  HALT
000224  000226                  .+2
000226  000000                  HALT
000230  000232                  .+2
000232  000000                  HALT
000234  000236                  .+2
000236  000000                  HALT
000240  000242                  .+2
000242  000000                  HALT
000244  000246                  .+2
000246  000000                  HALT
000250  000252                  .+2
000252  000000                  HALT
000254  000256                  .+2
000256  000000                  HALT
000260  000262                  .+2
000262  000000                  HALT
000264  000266                  .+2
000266  000000                  HALT
000270  000272                  .+2
000272  000000                  HALT
000274  000276                  .+2
000276  000000                  HALT
000300  000302                  .+2
000302  000000                  HALT
000304  000306                  .+2
000306  000000                  HALT
000310  000312                  .+2
000312  000000                  HALT
000314  000316                  .+2
000316  000000                  HALT
000320  000322                  .+2
000322  000000                  HALT
000324  000326                  .+2
000326  000000                  HALT
000330  000332                  .+2
000332  000000                  HALT
```

```
000334  000336                          .+2
000336  000000                          HALT
000340  000342                          .+2
000342  000000                          HALT
000344  000346                          .+2
000346  000000                          HALT
000350  000352                          .+2
000352  000000                          HALT
000354  000356                          .+2
000356  000000                          HALT
000360  000362                          .+2
000362  000000                          HALT
000364  000366                          .+2
000366  000000                          HALT
000370  000372                          .+2
000372  000000                          HALT
000374  000376                          .+2
000376  000000                          HALT


        000046                          .=46
000046  004722                          ENDAD
        000052                          .=52
000052  040000                          40000

        000200                          .=200
000200  000167  000606                  JMP     START

        001000                          .=1000
001000  000000              ICNT:       0                       ;PASS COUNT
001002  177772              PI:         177772                  ;PROGRAM INTERRUPT REQUEST REGISTER
001004  177773              PIH:        177773                  ;HIGH (000) BYTE
001006  000240              PIRVEC:     240                     ;PROGRAM INTERRUPT REQUEST INTER-
001010  000242              PIRLVL:     242                     ;RUPT VECTOR AND STATUS
001012  005067  177762      START:      CLR     ICNT            ;CLEAR PASS COUNT
001016  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK PTR
001022  012706  000500      BEGIN:      MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001026  012767  000340  176742          MOV     #PRTY7,PSW      ;SET PROCESSER PRIORITY=7
001034  016737  177740  177570          MOV     ICNT,@#DISPLAY  ;DISPLAY PASS COUNT
001042  032737  000400  177570          BIT     #400,@#SWR      ;LOAD MICRO BREAK REGISTER
001050  001403                          BEQ     .+10
001052  113737  177570  177770          MOVB    @#SWR,@#UBREAK  ;LOAD MICRO BREAK REG WITH SR0-7

                                ;TEST THAT PROGRAM INTERRUPT REGISTER (PIRQ) CAN BE ACCESSED
001060  012767  001074  176716  T0:     MOV     #T0A,4          ;LOAD ERROR TRAP VECTOR
001066  005777  177710                  TST     @PI             ;REFERENCE PI
001072  000403                          BR      T0B
001074  022626              T0A:        POP2                    ;POP 2 WORDS OFF THE STACK
001076  000000                          HLT                     ;ERROR CP FAILED TO ACCESS PI
001100  000767                          BR      T0              ;LOOP TEST IF ERROR
001102  012767  000006  176674  T0B:     MOV     #6,4            ;RESTORE ERROR TRAP
001110  000400                          BR      T1              ;GO TO NEXT TEST

                                ;TEST THAT PROGRAM INTERRUPT REGISTER CAN BE LOADED AND CLEARED (T1-T7)
```

```
001112  010701                          T1:     SCOPE
001114  012767  000340  176654                  MOV     #PRIY7,PSW      ;SET PROCESSER PRIORITY=7
001122  012777  001000  177652                  MOV     #PIR1,@PI       ;SET PIR=1
001130  017700  177646                          MOV     @PI,R0          ;GET RESULT
001134  022700  001042                          CMP     #PIR1+PIA1,R0   ;CORRECT RESULT?
001140  001401                                  BEQ     T1A
001142  000000                                  HLT                     ;INCORRECT RESULT
001144  005077  177632          T1A:    CLR     @PI                     ;CLEAR PROGRAM INTERRUPT REGISTER
001150  017700  177626                          MOV     @PI,R0          ;GET RESULTS
001154  001401                                  BEQ     T2
001156  000000                                  HLT                     ;PIR0 DID NOT CLEAR

001160  010701                          T2:     SCOPE
001162  012777  002000  177612                  MOV     #PIR2,@PI       ;SET PIR=2
001170  017700  177606                          MOV     @PI,R0          ;GET RESULT
001174  022700  002104                          CMP     #PIR2+PIA2,R0   ;CORRECT RESULT
001200  001401                                  BEQ     T2A
001202  000000                                  HLT
001204  005077  177572          T2A:    CLR     @PI
001210  017700  177566                          MOV     @PI,R0
001214  001401                                  BEQ     T3
001216  000000                                  HLT

001220  010701                          T3:     SCOPE
001222  012777  004000  177552                  MOV     #PIR3,@PI
001230  017700  177546                          MOV     @PI,R0
001234  022700  004146                          CMP     #PIR3+PIA3,R0
001240  001401                                  BEQ     T3A
001242  000000                                  HLT
001244  005077  177532          T3A:    CLR     @PI
001250  017700  177526                          MOV     @PI,R0
001254  001401                                  BEQ     T4

001256  000000                                  HLT
```

```
001260  010701                          T4:     SCOPE
001262  012777  010000  177512                  MOV     #PIR4,@PI
001270  017700  177506                          MOV     @PI,R0
001274  022700  010210                          CMP     #PIR4+PIA4,R0
001300  001401                                  BEQ     T4A
001302  000000                                  HLT
001304  005077  177472          T4A:    CLR     @PI
001310  017700  177466                          MOV     @PI,R0
001314  001401                                  BEQ     T5

001316  000000                                  HLT
001320  010701                          T5:     SCOPE
001322  012777  020000  177452                  MOV     #PIR5,@PI
001330  017700  177446                          MOV     @PI,R0
001334  022700  020252                          CMP     #PIR5+PIA5,R0
001340  001401                                  BEQ     T5A
001342  000000                                  HLT
001344  005077  177432          T5A:    CLR     @PI
001350  017700  177426                          MOV     @PI,R0
001354  001401                                  BEQ     T6

001356  000000                                  HLT
001360  010701                          T6:     SCOPE
001362  012777  040000  177412                  MOV     #PIR6,@PI
001370  017700  177406                          MOV     @PI,R0
001374  022700  040314                          CMP     #PIR6+PIA6,R0
001400  001401                                  BEQ     T6A
001402  000000                                  HLT
001404  005077  177372          T6A:    CLR     @PI
001410  017700  177366                          MOV     @PI,R0
001414  001401                                  BEQ     T7

001416  000000                                  HLT
001420  010701                          T7:     SCOPE
001422  012777  100000  177352                  MOV     #PIR7,@PI
001430  017700  177346                          MOV     @PI,R0
001434  022700  100356                          CMP     #PIR7+PIA7,R0
001440  001401                                  BEQ     T7A
001442  000000                                  HLT
001444  005C77  177332          T7A:    CLR     @PI
001450  017700  177326                          MOV     @PI,R0
001454  001401                                  BEQ     T10
001456  000000                                  HLT

                                ;TEST THAT RESET CLEARS PIR0
001460  010701                          T10:    SCOPE
001462  012777  177777  177312                  MOV     #-1,@PI         ;SET ALL PIR BITS IN MSH AND
                                                                        ;PIA BITS = TO 7 IN LSH
001470  017700  177306                          MOV     @PI,R0          ;GET RESULT
001474  022700  177356                          CMP     #177000+PIA7,R0 ;DID ALL CORRECT BITS SET
001500  001401                                  BEQ     T10A
001502  000000                                  HLT
001504  000005          T10A:   RESET                           ;RESET
001506  017700  177270                          MOV     @PI,R0          ;GET RESULT
001512  001402                                  BEQ     T11             ;BRANCH IF 0
```

```
001514  000000                          HLT                     ;RESET DID NOT CLEAR ALL BITS
001516  000760                          BR      T10             ;LOOP TEST IF RESET FAILS
```

```
                                        ;TEST THAT ODD BYTE OF PIRQ CAN BE REFERENCED.
001520  010701                  T11:    SCOPE
001522  112777  000002  177254          MOVB    #2,@PIH         ;LOAD ODD BYTE
001530  117700  177250                  MOVB    @PIH,R0         ;GET ODD BYTE
001534  122700  000002                  CMPB    #2,R0           ;DID ODD BYTE LOAD CORRECTLY
001540  001401                          BEQ     T11A
001542  000000                          HLT
001544  017700  177232          T11A:   MOV     @PI,R0          ;GET PIRQ
001550  022700  001042                  CMP     #PIR1+PIA1,R0   ;COMPARE WORD
001554  001401                          BEQ     T12
001556  000000                          HLT                     ;WORD FAILED

                                        ;TEST THAT EVEN BYTE BITS CANNOT BE 'PROGRAM' SET.
001560  010701                  T12:    SCOPE
001562  005077  177214                  CLR     @PI             ;CLEAR PIRQ
001566  112777  177777  177206          MOVB    #-1,@PI         ;TRY TO SET EVEN BYTE BITS
001574  017700  177202                  MOV     @PI,R0          ;GET RESULT
001600  001401                          BEQ     T13
001602  000000                          HLT                     ;ERROR! PIRQ GOT LOADED

                                        ;TEST THAT 'PIA' BITS DECODE ONLY THE MOST SIGNIFICANT 'SET' BIT

001604  010701                  T13:    SCOPE
001606  012777  001000  177166          MOV     #PIR1,@PI
001614  017700  177162                  MOV     @PI,R0
001620  022700  001042                  CMP     #PIR1+PIA1,R0
001624  001401                          BEQ     .+4
001626  000000                          HLT

001630  052777  002000  177144          BIS     #PIR2,@PI
001636  017700  177140                  MOV     @PI,R0
001642  022700  003104                  CMP     #PIR1+PIR2+PIA2,R0
001646  001401                          BEQ     .+4
001650  000000                          HLT

001652  052777  004000  177122          BIS     #PIR3,@PI
001660  017700  177116                  MOV     @PI,R0
001664  022700  007146                  CMP     #PIR1+PIR2+PIR3+PIA3,R0
001670  001401                          BEQ     .+4
001672  000000                          HLT

001674  052777  010000  177100          BIS     #PIR4,@PI
001702  017700  177074                  MOV     @PI,R0
001706  022700  017210                  CMP     #PIR1+PIR2+PIR3+PIR4+PIA4,R0
001712  001401                          BEQ     .+4
001714  000000                          HLT

001716  052777  020000  177056          BIS     #PIR5,@PI
001724  017700  177052                  MOV     @PI,R0
001730  022700  037252                  CMP     #PIR1+PIR2+PIR3+PIR4+PIR5+PIA5,R0
001734  001401                          BEQ     .+4
001736  000000                          HLT

001740  052777  040000  177034          BIS     #PIR6,@PI
001746  017700  177030                  MOV     @PI,R0
```

```
001752  022700  077314                  CMP     #PIR1+PIR2+PIR3+PIR4+PIR5+PIR6+PIA6,R0
001756  001401                          BEQ     .+4
001760  000000                          HLT
001762  052777  100000  177012          BIS     #PIR7,@PI
001770  017700  177006                  MOV     @PI,R0
001774  022700  177356                  CMP     #PIR1+PIR2+PIR3+PIR4+PIR5+PIR6+PIR7+PIA7,R0
002000  001401                          BEQ     T14
002002  000000                          HLT
```

```
                                        ;TEST THAT WHEN PROGRAM INTERRUPT OCCURS THAT IT DOES SO AT THE CORRECT
                                        ;VECTOR.  IF THE VECTOR IS INCORRECT PROGRAM WILL HALT AT THE INCORRECT VECTOR
                                        ;ADDRESS +2.

002004  010701                  T14:    SCOPE
002006  012777  002042  176772          MOV     #T14A,@PIRVEC   ;LOAD INTERRUPT VECTOR AND
002014  012777  000340  176766          MOV     #PRTY7,@PIRLVL  ;PRIORITY LEVEL
002022  012777  001000  176752          MOV     #PIR1,@PI       ;REQUEST AN INTERRUPT AT LEVEL 1
002030  005067  175742                  CLR     PSW             ;ALLOW INTERRUPTS
002034  000240                          NOP
002036  000000                          HLT                     ;INTERRUPT FAILED
002040  000661                          BR      T13             ;LOOP TEST
002042  000240                  T14A:   NOP                     ;INTERRUPTS TO T14A
002044  022626                          POP2
002046  000005                          RESET                   ;CLEAR PIRQ
002050  012767  000340  175720          MOV     #PRTY7,PSW      ;LOCK OUT INTERRUPTS
                                        ;TEST THAT PSW GETS LOADED WITH THE WORD FOLLOWING THE VECTOR ADDRESS
                                        ;(VECTOR ADDRESS+2).
002056  010701                  T15:    SCOPE
002060  012777  002114  176720          MOV     #T15A,@PIRVEC
002066  012777  000340  176714          MOV     #PRTY7,@PIRLVL
002074  005067  175676                  CLR     PSW             ;CLEAR PSW
002100  012777  001000  176674          MOV     #PIR1,@PI
002106  000240                          NOP
002110  000000                          HLT                     ;ERROR INTERRUPT FAILED
002112  000412                          BR      T16
002114  022626                  T15A:   POP2
002116  016700  175654                  MOV     PSW,R0
002122  022700  000340                  CMP     #PRTY7,R0
002126  001404                          BEQ     T16
002130  000000                          HLT
002132  012767  000340  175636          MOV     #PRTY7,PSW
                                        ;TEST THAN AN INTERRUPT OCCURS IF INSTRUCTION FOLLOWING REQUEST
                                        ;LOWERS REQUEST LEVEL BELOW CURRENT PROCESSER STATUS LEVEL.
002140  010701                  T16:    SCOPE
002142  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002146  005077  176630                  CLR     @PI             ;CLEAR PIRQ
002152  012777  002212  176626          MOV     #T16A,@PIRVEC   ;LOAD INTERRUPT VECTOR
002160  012777  000340  176622          MOV     #PRTY7,@PIRLVL  ;AND STATUS
002166  012767  000200  175602          MOV     #PRTY4,PSW      ;SET PROCESSER STATUS EQUAL TO 4
002174  012777  020000  176600          MOV     #PIR5,@PI       ;REQUEST INTERRUPT AT LEVEL 5
002202  005077  176574                  CLR     @PI             ;DISABLE REQUEST
002206  000000                          HLT                     ;ERROR! PROGRAM DID NOT INTERRUPT
002210  000402                          BR      T17             ;GO TO NEXT TEST
002212  022626                  T16A:   POP2
002214  000400                          BR      T17             ;GO TO NEXT TEST

                                        ;TEST THAT IF THE INSTRUCTION FOLLOWING A REQUEST RAISES THE PROCESSER PRIORITY
                                        ;ABOVE THAT OF THE REQUEST THAT THE REQUEST STILL INTERRUPTS.
002216  010701                  T17:    SCOPE
002220  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002224  005077  176552                  CLR     @PI             ;CLEAR PIRQ
002230  012777  002274  176550          MOV     #T17A,@PIRVEC   ;LOAD INTERRUPT VECTOR
002236  012777  000340  176544          MOV     #PRTY7,@PIRLVL  ;AND STATUS
002244  012767  000100  175524          MOV     #PRTY2,PSW
```

```
002252  012777  004000  176522          MOV     #PIR3,@PI
002260  012767  000140  175510          MOV     #PRTY3,PSW
002266  000240                          NOP
002270  000000                          HLT                     ;ERROR! PROG. INT. NOT ACK.
002272  000402                          BR      T20
002274  022626                  T17A:   POP2
002276  000400                          BR      T20

        000020                  C=20
        000021                  D=21
        000001                  N=1
                                ;TEST THAT WHEN THE PROCESSER PRIORITY LEVEL (1) IS EQUAL TO THE PROGRAM
                                ;INTERRUPT REQUEST LEVEL (1) THAT NO INTERRUPT OCCURS.
002300  010701                  T20:    SCOPE
002302  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002306  005077  176470                  CLR     @PI             ;CLEAR PROGRAM INTERRUPT REQUEST REG.
002312  012777  002346  176466          MOV     #T20A,@PIRVEC   ;LOAD INTERRUPT VECTOR &
002320  012777  000340  176462          MOV     #PRTY7,@PIRLVL  ;STATUS
002326  012767  000040  175442          MOV     #PRTY1,PSW      ;SET PROCESSER STATUS EQUAL TO 1
002334  012777  001000  176440          MOV     #PIR1,@PI       ;REQUEST INTERRUPT AT LEVEL 1
002342  000240                          NOP
002344  000401                          BR      T21             ;GO TO NEXT TEST
002346  000000                  T20A:   HLT                     ;PROGRAM INTERRUPTED.STATUS=REQ-
                                                                ;UEST LEVEL=1.

        000002                          N=N+1
        000021                          C=C+1
        000022                          D=D+1
                                ;TEST THAT WHEN THE PROCESSER PRIORITY LEVEL (2) IS EQUAL TO THE PROGRAM
                                ;INTERRUPT REQUEST LEVEL (2) THAT NO INTERRUPT OCCURS.
002350  010701                  T21:    SCOPE
002352  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002356  005077  176420                  CLR     @PI             ;CLEAR PROGRAM INTERRUPT REQUEST REG.
002362  012777  002416  176416          MOV     #T21A,@PIRVEC   ;LOAD INTERRUPT VECTOR &
002370  012777  000340  176412          MOV     #PRTY7,@PIRLVL  ;STATUS
002376  012767  000100  175372          MOV     #PRTY2,PSW      ;SET PROCESSER STATUS EQUAL TO 2
002404  012777  002000  176370          MOV     #PIR2,@PI       ;REQUEST INTERRUPT AT LEVEL 2
002412  000240                          NOP
002414  000401                          BR      T22             ;GO TO NEXT TEST
002416  000000                  T21A:   HLT                     ;PROGRAM INTERRUPTED.STATUS=REQ-
                                                                ;UEST LEVEL=2.

        000003                          N=N+1
        000022                          C=C+1
        000023                          D=D+1
                                ;TEST THAT WHEN THE PROCESSER PRIORITY LEVEL (3) IS EQUAL TO THE PROGRAM
                                ;INTERRUPT REQUEST LEVEL (3) THAT NO INTERRUPT OCCURS.
002420  010701                  T22:    SCOPE
002422  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002426  005077  176350                  CLR     @PI             ;CLEAR PROGRAM INTERRUPT REQUEST REG.
002432  012777  002466  176346          MOV     #T22A,@PIRVEC   ;LOAD INTERRUPT VECTOR &
002440  012777  000340  176342          MOV     #PRTY7,@PIRLVL  ;STATUS
002446  012767  000140  175322          MOV     #PRTY3,PSW      ;SET PROCESSER STATUS EQUAL TO 3
002454  012777  004000  176320          MOV     #PIR3,@PI       ;REQUEST INTERRUPT AT LEVEL 3
002462  000240                          NOP
002464  000401                          BR      T23             ;GO TO NEXT TEST
```

```
002466  000000                  T22A:   HLT                     ;PROGRAM INTERRUPTED.STATUS=REQ-
                                                                ;UEST LEVEL=3.

        000004                          N=N+1
        000023                          C=C+1
        000024                          D=D+1
                                ;TEST THAT WHEN THE PROCESSER PRIORITY LEVEL (4) IS EQUAL TO THE PROGRAM
                                ;INTERRUPT REQUEST LEVEL (4) THAT NO INTERRUPT OCCURS.
002470  010701                  T23:    SCOPE
002472  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002476  005077  176230                  CLR     @PI             ;CLEAR PROGRAM INTERRUPT REQUEST REG.
002502  012777  002536  176276          MOV     #T23A,@PIRVEC   ;LOAD INTERRUPT VECTOR &
002510  012777  000340  176272          MOV     #PRTY7,@PIRLVL  ;STATUS
002516  012767  000240  175752          MOV     #PRTY5,PSW      ;SET PROCESSER STATUS EQUAL TO 4
002524  012777  010000  176250          MOV     #PIR4,@PI       ;REQUEST INTERRUPT AT LEVEL 4
002532  000240                          NOP
002534  000401                          BR      T24             ;GO TO NEXT TEST
002536  000000                  T23A:   HLT                     ;PROGRAM INTERRUPTED.STATUS=REQ-
                                                                ;UEST LEVEL=4.

        000005                          N=N+1
        000024                          C=C+1
        000025                          D=D+1
                                ;TEST THAT WHEN THE PROCESSER PRIORITY LEVEL (5) IS EQUAL TO THE PROGRAM
                                ;INTERRUPT REQUEST LEVEL (5) THAT NO INTERRUPT OCCURS.
002540  010701                  T24:    SCOPE
002542  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002546  005077  176230                  CLR     @PI             ;CLEAR PROGRAM INTERRUPT REQUEST REG.
002552  012777  002606  176226          MOV     #T24A,@PIRVEC   ;LOAD INTERRUPT VECTOR &
002560  012777  000340  176222          MOV     #PRTY7,@PIRLVL  ;STATUS
002566  012767  000240  175202          MOV     #PRTY5,PSW      ;SET PROCESSER STATUS EQUAL TO 5
002574  012777  020000  176200          MOV     #PIR5,@PI       ;REQUEST INTERRUPT AT LEVEL 5
002602  000240                          NOP
002604  000401                          BR      T25             ;GO TO NEXT TEST
002606  000000                  T24A:   HLT                     ;PROGRAM INTERRUPTED.STATUS=REQ-
                                                                ;UEST LEVEL=5.

        000006                          N=N+1
        000025                          C=C+1
        000026                          D=D+1
                                ;TEST THAT WHEN THE PROCESSER PRIORITY LEVEL (6) IS EQUAL TO THE PROGRAM
                                ;INTERRUPT REQUEST LEVEL (6) THAT NO INTERRUPT OCCURS.
002610  010701                  T25:    SCOPE
002612  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002616  005077  176160                  CLR     @PI             ;CLEAR PROGRAM INTERRUPT REQUEST REG.
002622  012777  002656  176156          MOV     #T25A,@PIRVEC   ;LOAD INTERRUPT VECTOR &
002630  012777  000340  176152          MOV     #PRTY7,@PIRLVL  ;STATUS
002636  012767  000300  175132          MOV     #PRTY6,PSW      ;SET PROCESSER STATUS EQUAL TO 6
002644  012777  040000  176130          MOV     #PIR6,@PI       ;REQUEST INTERRUPT AT LEVEL 6
002652  000240                          NOP
002654  000401                          BR      T26             ;GO TO NEXT TEST
002656  000000                  T25A:   HLT                     ;PROGRAM INTERRUPTED.STATUS=REQ-
                                                                ;UEST LEVEL=6.

        000007                          N=N+1
        000026                          C=C+1
```

```
                000027                          D=D+1
                                        ;TEST THAT WHEN THE PROCESSER PRIORITY LEVEL (7) IS EQUAL TO THE PROGRAM
                                        ;INTERRUPT REQUEST LEVEL (7) THAT NO INTERRUPT OCCURS.
002460  010701                          T26:    SCOPE
002462  012706  000500                          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002466  005077  176110                          CLR     @PI             ;CLEAR PROGRAM INTERRUPT REQUEST REG.
002472  012777  002726  176106                  MOV     #T26A,@PIRVEC   ;LOAD INTERRUPT VECTOR &
002500  012777  000340  176102                  MOV     #PRTY7,@PIRLVL  ;STATUS
002506  012767  000340  175062                  MOV     #PRTY7,PSW      ;SET PROCESSER STATUS EQUAL TO 7
002514  012777  100000  176060                  MOV     #PIR7,@PI       ;REQUEST INTERRUPT AT LEVEL 7
002522  000240                                  NOP
002524  000[illegible]  01                      BR      T27             ;GO TO NEXT TEST
002526  000000                          T26A:   HLT                     ;PROGRAM INTERRUPTED.STATUS=REQ-
                                                                        ;UEST LEVEL=7.

                000010                          N=N+1
                00[illegible]27                 C=C+1
                00[illegible]30                 D=D+1
                00[illegible]01         N=1
                00[illegible]00         M=0
                                        ;TEST THAT WHEN THE PROCESSER PRIORITY (1) IS GREATER THAN THE PROGRAM INT-
                                        ;ERRUPT REQUEST LEVEL (0) THAT NO INTERRUPT OCCURS.
002730  010701                          T27:    SCOPE
002732  012706  000500                          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002736  005077  176040                          CLR     @PI             ;CLEAR PIRQ
002742  012777  002776  176036                  MOV     #T27A,@PIRVEC;  ;LOAD INTERRUPT VECTOR
002750  012777  000340  176032                  MOV     #PRTY7,@PIRLVL  ;AND STATUS
002756  012767  000040  175012                  MOV     #PRTY1,PSW      ;SET PROCESSER STATUS EQUAL TO 1
002764  012777  000000  176010                  MOV     #PIR0,@PI       ;REQUEST INTERRUPT AT LEVEL 0
002772  000240                                  NOP
002774  000401                                  BR      T30             ;GO TO NEXT TEST
002776  000000                          T27A:   HLT                     ;ERROR PROGRAM INTERRUPTED WHEN
                                                                        ;STATUS=1, REQUEST LEVEL=0

                000030                          C=C+1
                000031                          D=D+1
                000001                          M=M+1
                000002                          N=N+1
                                        ;TEST THAT WHEN THE PROCESSER PRIORITY (2) IS GREATER THAN THE PROGRAM INT-
                                        ;ERRUPT REQUEST LEVEL (1) THAT NO INTERRUPT OCCURS.
003000  010701                          T30:    SCOPE
003002  012706  000500                          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003006  000077  175770                          CLR     @PI             ;CLEAR PIRQ
003012  012777  003046  175766                  MOV     #T30A,@PIRVEC;  ;LOAD INTERRUPT VECTOR
003020  012777  000340  175762                  MOV     #PRTY7,@PIRLVL  ;AND STATUS
003026  012767  000100  174742                  MOV     #PRTY2,PSW      ;SET PROCESSOR STATUS EQUAL TO 2
003034  012777  001000  175740                  MOV     #PIR1,@PI       ;REQUEST INTERRUPT AT LEVEL 1
003042  000240                                  NOP
003044  000401                                  BR      T31             ;GO TO NEXT TEST
003046  000000                          T30A:   HLT                     ;ERROR PROGRAM INTERRUPTED WHEN
                                                                        ;STATUS=2, REQUEST LEVEL=1

                000031                          C=C+1
                000032                          D=D+1
                000002                          M=M+1
                000003                          N=N+1
```

```
                                        ;TEST THAT WHEN THE PROCESSER PRIORITY (3) IS GREATER THAN THE PROGRAM INT-
                                        ;ERRUPT REQUEST LEVEL (2) THAT NO INTERRUPT OCCURS.
003050  010701                  T31:    SCOPE
003052  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003056  005077  175720                  CLR     @PI             ;CLEAR PIRQ
003062  012777  003116  175716          MOV     #T31A,@PIRVEC;  ;LOAD INTERRUPT VECTOR
003070  012777  000340  175712          MOV     #PRTY7,@PIRLVL  ;AND STATUS
003076  012767  000140  174672          MOV     #PRTY3,PSW      ;SET PROCESSER STATUS EQUAL TO 3
003104  012777  002000  175670          MOV     #PIR2,@PI       ;REQUEST INTERRUPT AT LEVEL 2
003112  000240                          NOP
003114  000401                          BR      T32             ;GO TO NEXT TEST
003116  000000                  T31A:   HLT                     ;ERROR PROGRAM INTERRUPTED WHEN
                                                                ;STATUS=3, REQUEST LEVEL=2

        000032                          C=C+1
        000033                          D=D+1
        000003                          M=M+1
        000004                          N=N+1
                                        ;TEST THAT WHEN THE PROCESSER PRIORITY (4) IS GREATER THAN THE PROGRAM INT-
                                        ;ERRUPT REQUEST LEVEL (3) THAT NO INTERRUPT OCCURS.
003120  010701                  T32:    SCOPE
003122  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003126  005077  175650                  CLR     @PI             ;CLEAR PIRQ
003132  012777  003166  175646          MOV     #T32A,@PIRVEC;  ;LOAD INTERRUPT VECTOR
003140  012777  000340  175642          MOV     #PRTY7,@PIRLVL  ;AND STATUS
003146  012767  000240  174622          MOV     #PRTY4,PSW      ;SET PROCESSER STATUS EQUAL TO 4
003154  012777  004000  175620          MOV     #PIR3,@PI       ;REQUEST INTERRUPT AT LEVEL 3
003162  000240                          NOP
003164  000401                          BR      T33             ;GO TO NEXT TEST
003166  000000                  T32A:   HLT                     ;ERROR PROGRAM INTERRUPTED WHEN
                                                                ;STATUS=4, REQUEST LEVEL=3

        000033                          C=C+1
        000034                          D=D+1
        000004                          M=M+1
        000005                          N=N+1
                                        ;TEST THAT WHEN THE PROCESSER PRIORITY (5) IS GREATER THAN THE PROGRAM INT-
                                        ;ERRUPT REQUEST LEVEL (4) THAT NO INTERRUPT OCCURS.
003170  010701                  T33:    SCOPE
003172  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003176  005077  175600                  CLR     @PI             ;CLEAR PIRQ
003202  012777  003236  175576          MOV     #T33A,@PIRVEC;  ;LOAD INTERRUPT VECTOR
003210  012777  000340  175572          MOV     #PRTY7,@PIRLVL  ;AND STATUS
003216  012767  000240  174552          MOV     #PRTY5,PSW      ;SET PROCESSER STATUS EQUAL TO 5
003224  012777  010000  175550          MOV     #PIR4,@PI       ;REQUEST INTERRUPT AT LEVEL 4
003232  000240                          NOP
003234  000401                          BR      T34             ;GO TO NEXT TEST
003236  000000                  T33A:   HLT                     ;ERROR PROGRAM INTERRUPTED WHEN
                                                                ;STATUS=5, REQUEST LEVEL=4

        000034                          C=C+1
        000035                          D=D+1
        000005                          M=M+1
        000006                          N=N+1
                                        ;TEST THAT WHEN THE PROCESSER PRIORITY (6) IS GREATER THAN THE PROGRAM INT-
                                        ;ERRUPT REQUEST LEVEL (5) THAT NO INTERRUPT OCCURS.
```

```
003240  010701                          T34:    SCOPE
003242  012706  000500                          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003246  005077  175530                          CLR     @PI             ;CLEAR PIRQ
003252  012777  003306  175526                  MOV     #T34A,@PIRVEC;  ;LOAD INTERRUPT VECTOR
003260  012777  000340  175522                  MOV     #PRTY7,@PIRLVL  ;AND STATUS
003266  012767  000300  174502                  MOV     #PRTY6,PSW      ;SET PROCESSOR STATUS EQUAL TO 6
003274  012777  020000  175500                  MOV     #PIR5,@PI       ;REQUEST INTERRUPT AT LEVEL 5
003302  000240                                  NOP
003304  000401                                  BR      T35             ;GO TO NEXT TEST
003306  000000                          T34A:   HLT                     ;ERROR PROGRAM INTERRUPTED WHEN
                                                                        ;STATUS=6, REQUEST LEVEL=5

        000035                                  C=C+1
        000036                                  D=D+1
        000006                                  M=M+1
        000007                                  N=N+1
                                        ;TEST THAT WHEN THE PROCESSER PRIORITY (7) IS GREATER THAN THE PROGRAM INT-
                                        ;ERRUPT REQUEST LEVEL (6) THAT NO INTERRUPT OCCURS.
003310  010701                          T35:    SCOPE
003312  012706  000500                          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003316  005077  175460                          CLR     @PI             ;CLEAR PIRQ
003322  012777  003336  175456                  MOV     #T35A,@PIRVEC;  ;LOAD INTERRUPT VECTOR
003330  012777  000340  175452                  MOV     #PRTY7,@PIRLVL  ;AND STATUS
003336  012767  000340  174432                  MOV     #PRTY7,PSW      ;SET PROCESSOR STATUS EQUAL TO 7
003344  012777  040000  175430                  MOV     #PIR6,@PI       ;REQUEST INTERRUPT AT LEVEL 6
003352  000240                                  NOP
003354  000401                                  BR      T36             ;GO TO NEXT TEST
003356  000000                          T35A:   HLT                     ;ERROR PROGRAM INTERRUPTED WHEN
                                                                        ;STATUS=7, REQUEST LEVEL=6

        000036                                  C=C+1
        000037                                  D=D+1
        000007                                  M=M+1
        000010                                  N=N+1
```

```
                    000000                  N=0
                    000001                  M=1
                                            ;TEST THAT WHEN THE PROCESSER PRIORITY (0) IS LESS THAN THE PROGRAM INT-
                                            ;ERRUPT REQUEST LEVEL(1) THAT AN INTERRUPT OCCURS.
003360  010701                            T36:    SCOPE
003362  012706  000500                            MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003366  005077  175410                            CLR     @PI             ;CLEAR PIRQ
003372  012777  003426  175406                    MOV     #T36A,@PIRVEC   ;LOAD INTERRUPT VECTOR
003400  012777  000340  175402                    MOV     #PRTY7,@PIRLVL  ;AND STATUS
003406  012767  000000  174362                    MOV     #PRTY0,PSW      ;SET PROCESSER STATUS=0
003414  012777  000200  175360                    MOV     #PIR1,@PI       ;REQUEST INTERRUPT AT LEVEL1
003422  000240                                    NOP
003424  000000                                    HLT                     ;ERROR! PROGRAM FAILED TO INTERRUPT
003426  000400                            T36A:   BR      T37             ;GO TO NEXT TEST

                    000001                          N=N+1
                    000002                          M=M+1
                    000037                          C=C+1
                    000040                          D=D+1
                                            ;TEST THAT WHEN THE PROCESSER PRIORITY (1) IS LESS THAN THE PROGRAM INT-
                                            ;ERRUPT REQUEST LEVEL(2) THAT AN INTERRUPT OCCURS.
003430  010701                            T37:    SCOPE
003432  012706  000500                            MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003436  005077  175340                            CLR     @PI             ;CLEAR PIRQ
003442  012777  003476  175336                    MOV     #T37A,@PIRVEC   ;LOAD INTERRUPT VECTOR
003450  012777  000340  175332                    MOV     #PRTY7,@PIRLVL  ;AND STATUS
003456  012767  000040  174312                    MOV     #PRTY1,PSW      ;SET PROCESSER STATUS=1
003464  012777  002000  175310                    MOV     #PIR2,@PI       ;REQUEST INTERRUPT AT LEVEL2
003472  000240                                    NOP
003474  000000                                    HLT                     ;ERROR! PROGRAM FAILED TO INTERRUPT
003476  000400                            T37A:   BR      T40             ;GO TO NEXT TEST

                    000002                          N=N+1
                    000003                          M=M+1
                    000040                          C=C+1
                    000041                          D=D+1
                                            ;TEST THAT WHEN THE PROCESSER PRIORITY (2) IS LESS THAN THE PROGRAM INT-
                                            ;ERRUPT REQUEST LEVEL(3) THAT AN INTERRUPT OCCURS.
003500  010701                            T40:    SCOPE
003502  012706  000500                            MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003506  005077  175270                            CLR     @PI             ;CLEAR PIRQ
003512  012777  003546  175266                    MOV     #T40A,@PIRVEC   ;LOAD INTERRUPT VECTOR
003520  012777  000340  175262                    MOV     #PRTY7,@PIRLVL  ;AND STATUS
003526  012767  000100  174242                    MOV     #PRTY2,PSW      ;SET PROCESSER STATUS=2
003534  012777  004000  175240                    MOV     #PIR3,@PI       ;REQUEST INTERRUPT AT LEVEL3
003542  000240                                    NOP
003544  000000                                    HLT                     ;ERROR! PROGRAM FAILED TO INTERRUPT
003546  000400                            T40A:   BR      T41             ;GO TO NEXT TEST

                    000003                          N=N+1
                    000004                          M=M+1
                    000041                          C=C+1
                    000042                          D=D+1
                                            ;TEST THAT WHEN THE PROCESSER PRIORITY (3) IS LESS THAN THE PROGRAM INT-
                                            ;ERRUPT REQUEST LEVEL(4) THAT AN INTERRUPT OCCURS.
003550  010701                            T41:    SCOPE
```

```
                                                MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
                                                CLR     @PI             ;CLEAR PIRQ
                                                MOV     #T41A,@PIRVEC   ;LOAD INTERRUPT VECTOR
                                                MOV     #PRTY7,@PIRLVL  ;AND STATUS
                                                MOV     #PRTY3,PSW      ;SET PROCESSER STATUS=3
                                                MOV     #PIR4,@PI       ;REQUEST INTERRUPT AT LEVEL4
                                                NOP
                                                HLT                     ;ERROR! PROGRAM FAILED TO INTERRUPT
                                T41A:           BR      T42             ;GO TO NEXT TEST

                                                N=N+1
                                                M=M+1
                                                C=C+1
                                                D=D+1
                                ;TEST THAT WHEN THE PROCESSER PRIORITY (4) IS LESS THAN THE PROGRAM INT-
                                ;ERRUPT REQUEST LEVEL(5) THAT AN INTERRUPT OCCURS.
                                T42:            SCOPE
                                                MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
                                                CLR     @PI             ;CLEAR PIRQ
                                                MOV     #T42A,@PIRVEC   ;LOAD INTERRUPT VECTOR
                                                MOV     #PRTY7,@PIRLVL  ;AND STATUS
                                                MOV     #PRTY4,PSW      ;SET PROCESSER STATUS=4
                                                MOV     #PIR5,@PI       ;REQUEST INTERRUPT AT LEVEL5
                                                NOP
                                                HLT                     ;ERROR! PROGRAM FAILED TO INTERRUPT
                                T42A:           BR      T43             ;GO TO NEXT TEST

                                                N=N+1
                                                M=M+1
                                                C=C+1
                                                D=D+1
                                ;TEST THAT WHEN THE PROCESSER PRIORITY (5) IS LESS THAN THE PROGRAM INT-
                                ;ERRUPT REQUEST LEVEL(6) THAT AN INTERRUPT OCCURS.
                                T43:            SCOPE
                                                MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
                                                CLR     @PI             ;CLEAR PIRQ
                                                MOV     #T43A,@PIRVEC   ;LOAD INTERRUPT VECTOR
                                                MOV     #PRTY7,@PIRLVL  ;AND STATUS
                                                MOV     #PRTY5,PSW      ;SET PROCESSER STATUS=5
                                                MOV     #PIR6,@PI       ;REQUEST INTERRUPT AT LEVEL6
                                                NOP
                                                HLT                     ;ERROR! PROGRAM FAILED TO INTERRUPT
                                T43A:           BR      T44             ;GO TO NEXT TEST

                                                N=N+1
                                                M=M+1
                                                C=C+1
                                                D=D+1
                                ;TEST THAT WHEN THE PROCESSER PRIORITY (6) IS LESS THAN THE PROGRAM INT-
                                ;ERRUPT REQUEST LEVEL(7) THAT AN INTERRUPT OCCURS.
                                T44:            SCOPE
                                                MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
                                                CLR     @PI             ;CLEAR PIRQ
                                                MOV     #T44A,@PIRVEC   ;LOAD INTERRUPT VECTOR
                                                MOV     #PRTY7,@PIRLVL  ;AND STATUS
                                                MOV     #PRTY6,PSW      ;SET PROCESSER STATUS=6
```

```
003774  012777  100000  175000          MOV     #PIR7,@PI       ;REQUEST INTERRUPT AT LEVEL7
004002  000240                          NOP
004004  000000                          HLT                     ;ERROR! PROGRAM FAILED TO INTERRUPT
004006  000400                  T44A:   BR      T45             ;GO TO NEXT TEST

                                        N=N+1
                                        M=M+1
                                        C=C+1
                                        D=D+1

                                ;TEST THAT PIRQ REQUESTS AN INTERRUPT AT EACH LEVEL AS THE REQUEST LEVEL
                                ;IS DECREASED.
004010  012706  000500          T45:    MOV     #STKPTR,SP      ;INITIALIZE STACK POINTER
004014  005000                          CLR     X0
004016  012767  000340  173752          MOV     #PRTY7,PSW      ;LOCK OUT INTERRUPTS
004024  012777  177777  174750          MOV     #-1,@PI         ;REQUEST AN INTERRUPT AT ALL LEVELS
004032  012777  004056  174746          MOV     #LEVEL7,@PIRVEC ;LOAD INT. RQST. VECTOR
004040  012777  000340  174742          MOV     #PRTY7,@PIRLVL  ;AND 'NEW' STATUS
004046  005067  173724                  CLR     PSW             ;ALLOW INTERRUPTS AT ALL LEVELS
004052  000000                          HLT                     ;ERROR! NO INTERRUPT
004054  000566                          BR      T45EX           ;EXIT TEST

004056  012700  100000          LEVEL7: MOV     #PIR7,X0        ;SET INDICATOR BIT IN R0
004062  022777  177356  174712          CMP     #PIR7+PIR6+PIR5+PIR4+PIR3+PIR2+PIR1+PIA7,@PI
004070  001401                          BEQ     .+4             ;IS PIRQ CORRECT?
004072  000000                          HLT                     ;INCORRECT PIRQ
004074  042777  100000  174700          BIC     #PIR7,@PI       ;DELETE REQUEST AT LEVEL 7
004102  012777  004120  174676          MOV     #LEVEL6,@PIRVEC ;SET UP FOR LEVEL 6 REQUEST
004110  022626                          POP2                    ;RESTORE THE STACK
004112  005067  173660                  CLR     PSW             ;ALLOW INTERRUPTS
004116  000000                          HLT                     ;ERROR! NO INTERRUPT
004120  052700  040000          LEVEL6: BIS     #PIR6,X0        ;SET INDICATOR BIT
004124  022777  077314  174650          CMP     #PIR6+PIR5+PIR4+PIR3+PIR2+PIR1+PIA6,@PI
004132  001401                          BEQ     .+4             ;IS PIRQ CORRECT?
004134  000000                          HLT                     ;ERROR! INCORRECT PIRQ
004136  042777  040000  174636          BIC     #PIR6,@PI       ;DELETE LEVEL 6 REQUEST
004144  012777  004162  174634          MOV     #LEVEL5,@PIRVEC ;SET UP FOR LEVEL 5 REQUEST
004152  022626                          POP2                    ;RESTORE THE STACK
004154  005067  173616                  CLR     PSW             ;ALLOW INTERRUPTS
004160  000000                          HLT                     ;ERROR! NO INTERRUPT

004162  052700  020000          LEVEL5: BIS     #PIR5,X0        ;SET INDICATOR BIT
004166  022777  037252  174606          CMP     #PIR5+PIR4+PIR3+PIR2+PIR1+PIA5,@PI
004174  001401                          BEQ     .+4             ;IS PIRQ CORRECT
004176  000000                          HLT                     ;ERROR! INCORRECT PIRQ
004200  042777  020000  174574          BIC     #PIR5,@PI       ;DELETE LEVEL 5 REQUEST
004206  012777  004224  174572          MOV     #LEVEL4,@PIRVEC ;SET UP FOR LEVEL 4 REQUEST
004214  022626                          POP2                    ;RESTORE THE STACK
004216  005067  173554                  CLR     PSW             ;ALLOW INTERRUPTS
004222  000000                          HLT                     ;ERROR! NO INTERRUPT

004224  052700  010000          LEVEL4: BIS     #PIR4,X0        ;SET INDICATOR BIT
004230  022777  017210  174544          CMP     #PIR4+PIR3+PIR2+PIR1+PIA4,@PI
004236  001401                          BEQ     .+4             ;IS PIRQ CORRECT?
004240  000000                          HLT                     ;ERROR! INCORRECT PIRQ
004242  042777  010000  174532          BIC     #PIR4,@PI       ;DELETE LEVEL 4 REQUEST
```

```
004250  012777  004266  174530          MOV     #LEVEL3,@PIRVEC ;SET UP FOR LEVEL 3 REQUEST
004256  [illegible]                     POP2                    ;RESTORE THE STACK
004260  [illegible]     173512          CLR     PSW             ;ALLOW INTERRUPTS
004264  000000                          HLT                     ;ERROR! NO INTERRUPT

004266  052700  004000          LEVEL3: BIS     #PIR3,%0        ;SET INDICATOR BIT
004272  012777  007146  174502          CMP     #PIR3+PIR2+PIR1+PIA3,@PI
004300  001401                          BEQ     .+4             ;IS PIRQ CORRECT?
004302  000000                          HLT                     ;ERROR! INCORRECT PIRQ
004304  042777  004000  174470          BIC     #PIR3,@PI       ;DELETE LEVEL 3 REQUEST
004312  012777  004330  174466          MOV     #LEVEL2,@PIRVEC ;SET UP FOR LEVEL 2 REQUEST
004320  [illegible]                     POP2                    ;RESTORE THE STACK
004322  [illegible]     173450          CLR     PSW             ;ALLOW INTERRUPTS
004326  [illegible]                     HLT                     ;ERROR! NO INTERRUPT

004330  052700  002000          LEVEL2: BIS     #PIR2,%0        ;SET INDICATOR BIT
004334  042777  003104  174440          CMP     #PIR2+PIR1+PIA2,@PI
004342  001401                          BEQ     .+4             ;IS PIRQ CORRECT?
004344  000000                          HLT                     ;ERROR! INCORRECT PIRQ
004346  042777  002000  174426          BIC     #PIR2,@PI       ;DELETE LEVEL 2 REQUEST
004354  012777  004372  174424          MOV     #LEVEL1,@PIRVEC ;SET UP FOR LEVEL 1 REQUEST
004362  [illegible]                     POP2                    ;RESTORE THE STACK
004364  [illegible]     173406          CLR     PSW             ;ALLOW INTERRUPTS
004370  [illegible]                     HLT                     ;ERROR! NO INTERRUPT

004372  052700  001000          LEVEL1: BIS     #PIR1,%0        ;SET INDICATOR BIT
004376  042777  001042  174376          CMP     #PIR1+PIA1,@PI
004404  001401                          BEQ     .+4             ;IS PIRQ CORRECT?
004406  000000                          HLT                     ;ERROR! INCORRECT PIRQ
004410  [illegible]                     POP2                    ;RESTORE THE STACK
004412  [illegible]     174364          CLR     @PI             ;CLEAR PROGRAM INT.RQST.REG.
004416  [illegible]     173354          CLR     PSW
004422  [illegible]     177000          CMP     #177000,%0      ;WERE ALL LEVELS SERVICED
004426  001401                          BEQ     .+4
004430  000000                          HLT                     ;ERROR! A LEVEL(S) NOT SERVICED
                                                                ;MISSINGBIT(S) GIVE LEVEL MISSED
004432  000240          T45EX:  NOP
                ;CHECK THAT PROGRAM INTERRUPT REQUEST TAKE PRECEDENCE
                ;OVER BUS INTERRUPT (TTY)

004434  010701          T46:    SCOPE
004436  012706  000500                  MOV     #STKPTR,%6      ;SET STACK PTR
004442  000237                          SPL     7               ;SET PRIOTITY LEVEL 7
004444  012777  004520  174334          MOV     #T46A,@PIRVEC   ;LOAD PIRQ INT VECTOR
004452  012777  000200  174330          MOV     #PRTY4,@PIRLVL  ;ASSUME PRIORITY LEVEL 4 ON INTERRUPT
004460  012737  004522  000064          MOV     #T46B,@#TPVEC   ;LOAD TTY PRINTER INTERRUPT
004466  012737  000200  000066          MOV     #PRTY4,@#TPVEC+2
004474  012737  000100  177564          MOV     #100,@#TPCSR    ;SET IE BIT
004502  012777  010000  174272          MOV     #PIR4,@PI       ;REQUEST INTERRUPT AT LEVEL 4
004510  005037  177776                  CLR     @#PSW           ;ALLOW PIRQ INTERRUPT
004514  000240                          NOP
004516  000000                          HLT                     ;ERROR!NO INTERRUPT
004520  000401          T46A:   BR      .+4                     ;BRANCH OVER HALT WHEN PIRQ INTERRUPTS
004522  000000          T46B:   HLT                             ;ERROR! TTY INTERRUPTED
004524  005077  174252                  CLR     @PI
004530  005037  177564                  CLR     @#TPCSR
```

```
004534  012737  000066  000064          MOV     #TPVEC+2,@#TPVEC

004542  010701                  T47:    SCOPE
004544  000237                          SPL     7                       ;SET PRIORITY LEVEL =7
004546  012706  000500                    MOV     #STKPTR,%6              ;SET STACK PTR
004552  012737  004612  000030          MOV     #T47A,@#EMTVEC          ;LOAD EMT TRAP VECTOR
004560  005037  000032                  CLR     @#EMTVEC+2              ;ASSUME PRIORITY LEVEL 0 AFTER EMT
004564  012777  004614  174214          MOV     #T47B,@PIRVEC           ;LOAD PIRQ INTERRUPT VECTOR
004572  012777  000340  174210          MOV     #PRTY7,@PIRLVL          ;ASSUME PRIORITY LEVEL 7 AFTER INT
004600  012777  010000  174174          MOV     #PIR4,@PI               ;BOOK INTERRUPT RQST AT LEVEL 4
004606  104000                          EMT                             ;TRAP ASSUME LEVEL 0 AFTER TRAP
004610  000000                          HLT                             ;ERROR! FAILED TO TRAP
004612  000000                  T47A:   HLT                             ;ERROR! PIRQ INT. FAILED AFTER EMT
004614  022726  004612          T47B:   CMP     #T47A,(6)+              ;CHECK RETURN PC ON THE STACK
004620  001401                          BEQ     .+4
004622  000000                          HLT                             ;ERROR! INCORRECT PC ON THE STACK
004624  005726                          TST     (6)+                    ;CHECK STATUS ON STACK
004626  001401                          BEQ     .+4
004630  000000                          HLT                             ;ERROR! INCORRECT STATUS ON STACK
004632  005077  174144                  CLR     @PI                     ;CLEAR INTERRUPT REQUEST
004636  016777  174146  174142          MOV     PIRLVL,@PIRVEC          ;RESTORE INT VECTOR
004644  005077  174140                  CLR     @PIRLVL
004650  012737  000032  000030          MOV     #EMTVEC+2,@#EMTVEC
004656  010701                          SCOPE

004660  005267  174114          END:    INC     ICNT                    ;INCREMENT THE PASS COUNTER
004664  026727  174110  001000          CMP     ICNT,#1000
004672  001402                          BEQ     DONE
004674  000167  174122                  JMP     BEGIN                   ;RESTART THE TEST
004700  012767  000007  172660  DONE:   MOV     #7,TPBUF                ;RING THE BELL
004706  105767  172652                  TSTB    TPCSR                   ;WAIT FOR BELL TO
004712  100375                          BPL     .-4                     ;RING
004714  013702  000042                  MOV     @#42,%2                 ;GET DECTAPE MONITOR RETURN ADDRESS
004720  001413                          BEQ     DONE1                   ;DO NOT RETURN IF (42)=0
004722  004712                  ENDAD:  JSR     7,(2)                   ;RETURN TO DECTAPE MONITOR
004724  000240                          NOP                             ;ACT11
004726  000240                          NOP                             ;OVERLAY
004730  000240                          NOP                             ;AREA
004732  010446                          MOV     R4,-(SP)
004734  005004                          CLR     R4
004736  012704  177777          1$:     MOV     #-1,R4
004742  005304                          DEC     R4
004744  001374                          BNE     1$
004746  012604                          MOV     (SP)+,R4
004750  000167  174036          DONE1:  JMP     START
        000001                          .END
```

```
BEGIN   001022   363#  1104
C     = 000045   602#   605   620#   622   637#   639   654#   656   671#   673   688#   690   705#
                 707    722#  726    740#  744    758#  762    776#  780    794#  798    812#  816
                 830#   834   848#   854   869#   871   886#   888   903#   905   920#   922   937#
                 939    954#  956    971#
D     = 000046   603#   605   621#   622   638#   639   655#   656   672#   673   689#   690   706#
                 707    723#  726    741#  744    759#  762    777#  780    795#  798    813#  816
                 831#   834   849#   854   870#   871   887#   888   904#   905   921#   922   938#
                 939    955#  956    972#
DISPLA= 177570   180#  365*
DONE    004700  1103  1105#
DONE1   004750  1109  1120#
EMTVEC= 000030   164#  1081*  1082*  1098*
END     004660  1101#
ENDAD   004722   348  1110#
ERRVEC= 000004   159#
HLT   = 000000   172#   375   398   392   399   403   410   415   422   427   433   438   444
                 449    455   459   468   472   481   485   493   502   508   514   520   526
                 542    537   549   563   569   581   597   616   633   650   667   684   701
                 718    737   755   773   791   809   827   845   864   881   898   915   932
                 949    966   983   989   994   998  1003  1008  1013  1018  1023  1028  1033
                1038   1043  1048  1054  1071  1073  1087  1088  1091  1094
ICNT    001000   356#  361*   365  1101*  1102
IOTVEC= 000020   162#
LEVEL1  004372  1040  1045#
LEVEL2  004330  1030  1035#
LEVEL3  004266  1020  1025#
LEVEL4  004224  1010  1015#
LEVEL5  004162  1000  1005#
LEVEL6  004120   991   995#
LEVEL7  004056   980   986#
M     = 000010   725#   726   742#   744   760#   762   778#   780   796#   798   814#   816   832#
                 834    850#  853#   854   868#   871   885#   888   902#   905   919#   922   936#
                 939    953#  956    970*
N     = 000007   604#   605   619#   622   636#   639   653#   656   670#   673   687#   690   704#
                 707    721#  724#   726   743#   744   761#   762   779#   780   797#   798   815#
                 816    833#  834    851#  852#   854   867#   871   884#   888   901#   905   918#
                 922    935#  939    952#  956    969#
PC    =%000007   156#
PFVEC = 000024   163#
PI      001002   357#   372   384*   385   389*   390   395*   396   400*   401   406*   407   411*
                 412    418*  419    423*  424    429*  430    434*  435    440*  441    445*  446
                 451*   452   456*   457   463*   465   470    482   489*   490*  491    498*  499
                 504*   505   510*   511   516*   517   522*   523   528*   529   533*   534   546*
                 561*   575*  579*   580*  590*   594*  609*   613*  626*   630*  643*   647*  660*
                 664*   677*  681*   694*  698*   711*  715*   730*  734*   748*  752*   766*  770*
                 784*   788*  802*   806*  820*   824*  838*   842*  858*   862*  875*   879*  892*
                 896*   909*  913*   926*  930*   943*  947*   960*  964*   979*  987    990*  996
                 999*  1006  1009*  1016  1019*  1026  1029*  1036  1039*  1046  1050*  1068*  1074*
                1085*  1095*
PIA0  = 000000   202#
PIA1  = 000042   203#   386   483   500  1046
PIA2  = 000104   204#   397   506  1036
PIA3  = 000146   205#   408   512  1026
PIA4  = 000210   206#   420   518  1016
PIA5  = 000252   207#   431   524  1006
```

DCKBNB.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

```
PIA6  = 000314   208#  442   530   996
PIA7  = 000356   209#  453   466   535   987
PIH     001004   358#  477*  478
PIRLVL  001010   360#  545*  559*  577*  592*  611*  628*  645*  662*  679*  696*  713*  732*
                 750*  768*  786*  804*  822*  840*  860*  877*  894*  911*  928*  945*  962*
                 981*  1064* 1084* 1096  1097*
PIRVEC  001006   359#  544*  558*  576*  591*  610*  627*  644*  661*  678*  695*  712*  731*
                 749*  767*  785*  803*  821*  839*  859*  876*  893*  910*  927*  944*  961*
                 980*  991*  1000* 1010* 1020* 1030* 1040* 1063* 1083* 1096*
PIR0  = 000000   193#  734
PIR1  = 001000   194#  384   386   483   498   500   506   512   518   524   530   535   546
                 561   613   752   862   987   996   1006  1016  1026  1036  1045  1046
PIR10 = 100000   201#
PIR2  = 002000   195#  395   397   504   506   512   518   524   530   535   630   770   879
                 987   996   1006  1016  1026  1035  1036  1039
PIR3  = 004000   196#  406   408   510   512   518   524   530   535   594   647   788   896
                 987   996   1006  1016  1025  1026  1029
PIR4  = 010000   197#  418   420   516   518   524   530   535   664   806   913   987   996
                 1006  1015  1016  1019  1068  1085
PIR5  = 020000   198#  429   431   522   524   530   535   579   681   824   930   987   996
                 1005  1006  1009
PIR6  = 040000   199#  440   442   528   530   535   698   842   947   987   995   996   999
PIR7  = 100000   200#  451   453   533   535   715   964   986   987   990
POP2  = 022626   173#  374   552   565   583   599   992   1001  1011  1021  1031  1041  1049
PRTY0 = 000000   182#  861
PRTY1 = 000040   183#  612   733   878
PRTY10= 000340   190#
PRTY2 = 000100   184#  593   629   751   895
PRTY3 = 000140   185#  595   646   769   912
PRTY4 = 000200   186#  578   663   787   929   1064  1066
PRTY5 = 000240   187#  680   805   946
PRTY6 = 000300   188#  697   823   963
PRTY7 = 000340   189#  364   383   545   554   559   567   570   577   592   611   628   645
                 662   679   696   713   714   732   750   768   786   804   822   840   841
                 860   877   894   911   928   945   962   978   981   1084
PSW   = 177776   175#  364*  383*  547*  554*  560*  566   570*  578*  593*  595*  612*  629*
                 646*  663*  680*  697*  714*  733*  751*  769*  787*  805*  823*  841*  861*
                 878*  895*  912*  929*  946*  963*  978*  982*  993*  1002* 1012* 1022* 1032*
                 1042* 1051* 1069*
RESVEC= 000010   160#
R0    =%000000   149#  385*  386   390*  396*  397   401*  407*  408   412*  419*  420   424*
                 430*  431   435*  441*  442   446*  452*  453   457*  465*  466   470*  478*
                 479   482*  483   491*  499*  500   505*  506   511*  512   517*  518   523*
                 524   529*  530   534*  535   566*  567
R1    =%000001   150#
R2    =%000002   151#
R3    =%000003   152#
R4    =%000004   153#  1114  1115* 1116* 1117* 1119*
R5    =%000005   154#
SCOPE = 010701   171#  382   394   405   417   428   439   450   462   476   488   497   543
                 557   573   588   607   624   641   658   675   692   709   728   746   764
                 [illegible]  800   818   836   856   873   890   907   924   941   958   1060  1078
                 1099
SP    =%000006   155#  362*  363*  574*  589*  608*  625*  642*  659*  676*  693*  710*  729*
                 747*  765*  783*  801*  819*  837*  857*  874*  891*  908*  925*  942*  959*
                 976*  1114* 1119
```

```
START   001012            353     361#   1120
STKPTR= 000500            170#    362    363     574     589     608     625     642     659     676     693     710     729
                          747     765    783     801     819     837     857     874     891     908     925     942     959
                          976    1061   1080
SWR    = 177570           179#    366    368
TBITVE= 000014            161#
TPBUF  = 177566           178#   1105*
TPCSR  = 177564           177#   1067*  1075*   1106
TPVEC  = 000064           166#   1065*  1066*   1076*
TRPVEC= 000034            165#
T0      001060            371#   376
T0A     001074            371    374#
T0B     001102            373    377#
T1      001112            378    3[illegible]#
T1A     001144            387    3[illegible]#
T10     001460            458    462#   473
T10A    001504            467    469#
T11     001520            471    476#
T11A    001544            480    48[illegible]#
T12     001560            484    48[illegible]#
T13     001604            492    497#   550
T14     002004            536    543#
T14A    002042            544    551#
T15     002056            557#
T15A    002114            558    565#
T16     002140            564    568    573#
T16A    002212            576    583#
T17     002216            582    584    588#
T17A    002274            591    5[illegible]#
T2      001160            391    394#
T2A     001204            398    400#
T20     002300            598    600    607#
T20A    002346            610    616#
T21     002350            615    624#
T21A    002416            627    633#
T22     002420            632    641#
T22A    002466            644    650#
T23     002470            649    658#
T23A    002536            661    667#
T24     002540            666    675#
T24A    002606            678    684#
T25     002610            683    692#
T25A    002656            695    701#
T26     002660            700    709#
T26A    002726            712    718#
T27     002730            717    728#
T27A    002776            731    737#
T3      001220            402    405#
T3A     001244            409    411#
T30     003000            736    746#
T30A    003046            749    755#
T31     003050            754    764#
T31A    003116            767    773#
T32     003120            772    782#
T32A    003166            785    791#
T33     003170            790    800#
```

PIRA	212#	605	622	639	656	673	690	707
PIRB	213#	726	744	762	780	798	816	834
PIRC	214#	854	871	888	905	922	939	956

```
BEQ      367   387   391   398   402   409   413   421   425   432   436   443   447   454   458
         467   471   479   484   492   501   507   513   519   525   531   536   568   988   997
        1007  1017  1027  1037  1047  1053  1090  1093  1103  1109
BIC      930   939  1009  1019  1029  1039
BIS      504   510   516   522   528   533   995  1005  1015  1025  1035  1045
BIT      366
BPT     1118
BR      1107
BVC      373   376   378   473   550   564   582   584   598   600   615   632   649   666   683
         700   717   736   754   772   790   808   826   844   865   882   899   916   933   950
         967   984  1072
CLR      361   389   400   411   423   434   445   456   489   547   560   575   580   590   609
         635   643   660   677   694   711   730   748   766   784   802   820   838   858   875
         892   909   926   943   960   977   982   993  1002  1012  1022  1032  1042  1050  1051
        1069  1074  1075  1082  1095  1097  1115
CMP      386   397   408   420   431   442   453   466   483   500   506   512   518   524   530
         555   567   987   996  1006  1016  1026  1036  1046  1052  1089  1102
CMPB     479
DEC     1117
EMT     1086
HALT     172   217   219   221   223   225   227   229   231   233   235   237   239   241   243
         245   247   249   251   253   255   257   259   261   263   265   267   269   271   273
         275   277   279   281   283   285   287   289   291   293   295   297   299   301   303
         305   307   309   311   313   315   317   319   321   323   325   327   329   331   333
         335   337   339   341   343
INC     1101
JMP      353  1104  1120
JSR     1110
MOV      362   363   364   365   371   377   383   384   385   390   395   396   401   406   407
         412   418   419   424   429   430   435   440   441   446   451   452   457   463   475
         470   482   491   498   499   505   511   517   523   529   534   544   545   546   554
         558   559   561   566   570   574   576   577   578   579   589   591   592   593   594
         595   608   610   611   612   513   625   627   628   629   630   642   644   645   646
         647   659   661   662   663   664   676   678   679   680   681   693   695   696   697
         698   710   712   713   714   715   729   731   732   733   734   747   749   750   751
         752   765   767   768   769   770   783   785   786   787   788   801   803   804   805
         806   819   821   822   823   824   837   839   840   841   842   857   859   860   861
         862   874   876   877   878   879   891   893   894   895   896   909   910   911   912
         913   925   927   928   929   930   942   944   945   946   947   959   961   962   963
         964   976   978   979   980   981   986   991  1000  1010  1020  1030  1040  1061  1063
        1064  1065  1066  1067  1068  1076  1080  1081  1083  1084  1085  1096  1098  1105  1108
        1114  1116  1119
MOVB     368   477   478   490
NOP      548   551   562   596   614   631   648   665   682   699   716   735   753   771   789
         807   825   843   863   880   897   914   931   948   965  1056  1070  1111  1112  1113
RESET    469   553
SPL     1062  1079
TST      372  1092
TSTB    1106
.ABS     135
.END    1121
.LIST    134
.MACR    212   213   214
.NLIST   133
.REM       1
.REPT    216   605   726   854
```

F03

```
MAINDEC-11-DCKBN-B PROG INT ROST LOGIC  MACY11 27(732)  10-SEP-76  10:01  PAGE 34
DCKBNB.P11      CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

.TITLE    132



 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 *DCKBNB,DCKBNB.SEQ/SOL/CRF/DS:ERFZ/EN:ABS=DSKM:DCKBNB.P11
 RUN-TIME:  3 6 1 SECONDS
 RUN-TIME RATIO: 21/13=1.6
 CORE USED:  7K  (13 PAGES)
```

G03

[illegible]