

PDP11/45

INTERRUPT TEST
MD-11-DCKBN-C

EP-DCKBN-C-DL-B
COPYRIGHT © 1976
FICHE 1 OF 1

DEC 1976
digital
MADE IN USA

MAINDEC-11-DCKBN-C PROG INT RQST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 4
DCKBNC.F11

IDENTIFICATION

PRODUCT CODE:	MAINDEC-11-DCKBN-C-D
PRODUCT NAME:	11/45 PIRQ INTERRUPT TEST
DATE:	DECEMBER 1976
MAINTAINER:	DIAGNOSTIC GROUP

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

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED UNDER A
LICENSE AND MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH THE TERMS
OF SUCH LICENSE.

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

COPYRIGHT (C) 1972, 1976
DIGITAL EQUIPMENT CORPORATION
MAYNARD, MASS

ՀԱՅԿԱՍՏԱՆԻ ՀԱՆՐԱՊԵՏՈՒԹՅԱՆ ՎԵՐԱԴԱՐՄԱՆ ԳԼԽԱՎՈՐ ԲԵՐԱՌԱՆՔԱՅԻՆ ԿԵՆՏՐՈՆԸ
ՀԱՅԿԱՍՏԱՆԻ ՀԱՆՐԱՊԵՏՈՒԹՅԱՆ ԱՇԽԱՏԱՆԿԱՅԻՆ ԿԵՆՏՐՈՆԸ

- 1.0 ABSTRACT
THIS IS A TEST OF THE PROGRAM INTERRUPT 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,P1 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 PROCEED 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 CONTINUOUSLY REPLACE THE BEQ .+4 INSTRUCTION IMMEDIATELY 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

100
101
102
103
104
105
106
107
108
109
110
111
112

001

MAINDEC-11-DCKBN-C PROG INT RQST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 6
DBKBN.C.F11

113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131

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-C 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
R0=%0
R1=%1
R2=%2
R3=%3
R4=%4
R5=%5
SP=%6
PC=%7

;VECTOR ADDRESSES

000000
000001
000002
000003
000004
000005
000006
000007

000004
000010
000014
000020
000024
000030
000034
000064

ERRVEC=4
RESVEC=10
TBITVEC=14
IOTVEC=20
PFVEC=24
EMTVEC=30
TRPVEC=34
TPVEC=64

;TELETYPE PRINTER INTERRUPT VECTOR

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

E01

MAINDEC-11-DCKBN-C PROG INT RQST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 7
DBKBNC.P11

000500
010701
000000
022626

177776
177770
177564
177566
177570
177570

000000
000040
000100
000140
000200
000240
000300
000340
000340

000000
001000
002000
004000
010000
020000
040000
100000
100000
000000
000042
000104
000146
000210
000252
000314
000356

STKPTR=500
SCOPE=010701
HLT=HALT
POP2=022626
;RESISTER ADDRESSES
PSW=177776
UBREAK=177770
TPCSR=177564
TPBUF=177566
SWR=177570
DISPLAY=177570

;PSW BIT ASSIGNMENTS
PRTY0=0
PRTY1=40
PRTY2=100
PRTY3=140
PRTY4=200
PRTY5=240
PRTY6=300
PRTY7=340
PRTY10=340

;PIRQ BIT ASSIGNMENTS
PIR0=0
PIR1=1000
PIR2=2000
PIR3=4000
PIR4=10000
PIR5=20000
PIR6=40000
PIR7=100000
PIR10=100000
PIA0=0
PIA1=42
PIA2=104
PIA3=146
PIA4=210
PIA5=252
PIA6=314
PIA7=356

;MACRO CALLS

;INITIAL STACK POINTER
;MOVE PC TO R1
;ERROR HALT
;POPS TWO WORD OFF THE STACK

;ADDRESS OF PROCESSOR STATUS
;ADDRESS OF MICRO BREAK REGISTER
;ADDRESS OF TELEPRINTER CONTROL STATUS
;AND DATA BUFFER REGISTER
;ADDRESS OF CONSOLE SWITCH REGISTER
;ADDRESS OF CONSOLE DISPLAY REGISTER

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

GO1

MAINDEC-11-DCKBN-C PROG INT RQST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 9
DBKBNC.P11

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

H01

MAINDEC-11-DCKBN-C PROG INT RQST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 10
DBKBNC.P11

```

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

000200 000200      .+200
000167 000167 000606 JMP      START

001000 001000      .+1000
001000 000000      ICNT: 0      ;PASS COUNT
001002 177772      PI: 177772 ;PROGRAM INTERRUPT REQUEST REGISTER
001004 177773      PIH: 177773 ;HIGH (ODD) 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      #PTY7,PSW ;SET PROCESSOR PRIORITY=7
001034 016737 177740 177570 MOV      ICNT,2#DISPLAY ;DISPLAY PASS COUNT
001042 032737 000400 177570 BIT      #400,2#SWR ;LOAD MICRO BREAK REGISTER
001050 001403      BEQ      .+10
001052 113737 177570 177770 MOVB     2#SWR,2#UBREAK ;LOAD MICRO BREAK REG WITH SR0-7

001060 012767 001074 176716 ;TEST THAT PROGRAM INTERRUPT REGISTER (PIRQ) CAN BE ACCESSED
001066 005777 177710 TO:      MOV      #TOA,4      ;LOAD ERROR TRAP VECTOR
001072 000403      TST      2#PI      ;REFERENCE PI
001074 022626      BR       TOB
001076 000000      TOA:     POP2      ;POP 2 WORDS OFF THE STACK
001100 000767      HLT      ;ERROR CP FAILED TO ACCESS PI
001102 012767 000006 176674 TOB: BR       TO      ;LOOP TEST IF ERROR
001110 000400      BR       T1      ;RESTORE ERROR TRAP
                                ;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	#PTY7,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		;PIRQ 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,API
001270	017700	177506			MOV	API,RO
001274	022700	010210			CMP	#PIR4+PIA4,RO
001300	001401				BEQ	T4A
001302	000000				HLT	
001304	005077	177472		T4A:	CLR	API
001310	017700	177466			MOV	API,RO
001314	001401				BEQ	T5
001316	000000				HLT	
001320	010701			T5:	SCOPE	
001322	012777	020000	177452		MOV	#PIR5,API
001330	017700	177446			MOV	API,RO
001334	022700	020252			CMP	#PIR5+PIA5,RO
001340	001401				BEQ	T5A
001342	000000				HLT	
001344	005077	177432		T5A:	CLR	API
001350	017700	177426			MOV	API,RO
001354	001401				BEQ	T6
001356	000000				HLT	
001360	010701			T6:	SCOPE	
001362	012777	040000	177412		MOV	#PIR6,API
001370	017700	177406			MOV	API,RO
001374	022700	040314			CMP	#PIR6+PIA6,RO
001400	001401				BEQ	T6A
001402	000000				HLT	
001404	005077	177372		T6A:	CLR	API
001410	017700	177366			MOV	API,RO
001414	001401				BEQ	T7
001416	000000				HLT	
001420	010701			T7:	SCOPE	
001422	012777	100000	177352		MOV	#PIR7,API
001430	017700	177346			MOV	API,RO
001434	022700	100356			CMP	#PIR7+PIA7,RO
001440	001401				BEQ	T7A
001442	000000				HLT	
001444	005077	177332		T7A:	CLR	API
001450	017700	177326			MOV	API,RO
001454	001401				BEQ	T10
001456	000000				HLT	
:TEST THAT RESET CLEARS PIRQ						
001460	010701			T10:	SCOPE	
001462	012777	177777	177312		MOV	#-1,API
001470	017700	177306			MOV	API,RO
001474	022700	177356			CMP	#177000+PIA7,RO
001500	001401				BEQ	T10A
001502	000000				HLT	
001504	000005			T10A:	RESET	
001506	017700	177270			MOV	API,RO
001512	001402				BEQ	T11
						:SET ALL PIR BITS IN MSH AND
						:PIA BITS = TO 7 IN LSH
						:GET RESULT
						:DID ALL CORRECT BITS SET
						:RESET
						:GET RESULT
						:BRANCH IF 0

K01

MAINDEC-11-DCKBN-C PROG INT RQST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 13
DBKBNC.P11

001514 000000
001516 000760

HLT
BR T10

:RESET DID NOT CLEAR ALL BITS
:LOOP TEST IF RESET FAILS

L01

MAINDEC-11-DCKBN-C PROG INT RQST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 14
DBKBNC.P11

```

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

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

;TEST THAT 'PIA' BITS DECODE ONLY THE MOST SIGNIFICANT 'SET' BIT
T13:  SCOPE
001604 010701
001606 012777 001000 177166  MOV     #PIR1,API
001614 017700 177162      MOV     API,RO
001620 022700 001042      CMP     #PIR1+PIA1,RO
001624 001401      BEQ     .+4
001626 000000      HLT

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

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

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

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

001740 052777 040000 177034  BIS     #PIR6,API
001746 017700 177030      MOV     API,RO

```


MO1

MAINDEC-11-DCKBN-C PROG INT RQST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 15
DBKBNC.P11

001752	022700	077314	CMP	#PIR1+PIR2+PIR3+PIR4+PIR5+PIR6+PIA6,R0
001756	001401		BEQ	.+4
001760	000000		HLT	
001762	052777	100000 177012	BIS	#PIR7,3PI
001770	017700	177006	MOV	3PI,R0
001774	022700	177356	CMP	#PIR1+PIR2+PIR3+PIR4+PIR5+PIR6+PIR7+PIA7,R0
002000	001401		BEQ	T14
002002	000000		HLT	

NO1

MAINDEC-11-DCKBN-C PROG INT RQST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 16
DBKBNC.P11

: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
002006 012777 002042 176772
002014 012777 000340 176766
002022 012777 001000 176752
002030 005067 175742
002034 000240
002036 000000
002040 000661
002042 000240
002044 022626
002046 000005
002050 012767 000340 175720

T14: SCOPE
MOV #T14A, @PIRVEC ;LOAD INTERRUPT VECTOR AND
MOV #PTY7, @PIRLVL ;PRIORITY LEVEL
MOV #PIR1, @PI ;REQUEST AN INTERRUPT AT LEVEL 1
CLR PSW ;ALLOW INTERRUPTS
NOP
HLT ;INTERRUPT FAILED
BR T13 ;LOOP TEST
T14A: NOP ;INTERRUPTS TO T14A
POP2
RESET ;CLEAR PIRQ
MOV #PTY7, PSW ;LOCK OUT INTERRUPTS

:TEST THAT PSW GETS LOADED WITH THE WORD FOLLOWING THE VECTOR ADDRESS
:(VECTOR ADDRESS+2).

002056 010701
002060 012777 002114 176720
002066 012777 000340 176714
002074 005067 175676
002100 012777 001000 176674
002106 000240
002110 000000
002112 000412
002114 022626
002116 016700 175654
002122 022700 000340
002126 001404
002130 000000
002132 012767 000340 175636

T15: SCOPE
MOV #T15A, @PIRVEC
MOV #PTY7, @PIRLVL
CLR PSW ;CLEAR PSW
MOV #PIR1, @PI
NOP
HLT ;ERROR INTERRUPT FAILED
BR T16
T15A: POP2
MOV PSW, R0
CMP #PTY7, R0
BEQ T16
HLT
MOV #PTY7, PSW

:TEST THAT AN INTERRUPT OCCURS IF INSTRUCTION FOLLOWING REQUEST
:LOWERS REQUEST LEVEL BELOW CURRENT PROCESSOR STATUS LEVEL.

002140 010701
002142 012706 000500
002146 005077 176630
002152 012777 002212 176626
002160 012777 000340 176622
002166 012767 000200 175602
002174 012777 020000 176600
002202 005077 176574
002206 000000
002210 000402
002212 022626
002214 000400

T16: SCOPE
MOV #STKPTR, SP ;INITIALIZE THE STACK POINTER
CLR @PI ;CLEAR PIRQ
MOV #T16A, @PIRVEC ;LOAD INTERRUPT VECTOR
MOV #PTY7, @PIRLVL ;AND STATUS
MOV #PTY4, PSW ;SET PROCESSOR STATUS EQUAL TO 4
MOV #PIR5, @PI ;REQUEST INTERRUPT AT LEVEL 5
CLR @PI ;DISABLE REQUEST
HLT ;ERROR! PROGRAM DID NOT INTERRUPT
BR T17 ;GO TO NEXT TEST
T16A: POP2
BR T17 ;GO TO NEXT TEST

:TEST THAT IF THE INSTRUCTION FOLLOWING A REQUEST RAISES THE PROCESSOR PRIORITY
:ABOVE THAT OF THE REQUEST THAT THE REQUEST STILL INTERRUPTS.

002216 010701
002220 012706 000500
002224 005077 176552
002230 012777 002274 176550
002236 012777 000340 176544
002244 012767 000100 175524

T17: SCOPE
MOV #STKPTR, SP ;INITIALIZE THE STACK POINTER
CLR @PI ;CLEAR PIRQ
MOV #T17A, @PIRVEC ;LOAD INTERRUPT VECTOR
MOV #PTY7, @PIRLVL ;AND STATUS
MOV #PTY2, PSW

002252 012777 004000 176522
 002260 012767 000140 175510
 002266 000240
 002270 000000
 002272 000402
 002274 022626
 002276 000400

MOV #PIR3,API
 MOV #PRTY3,PSW
 NOP
 HLT
 BR T20
 T17A: POP2
 BR T20

;ERROR! PROG. INT. NOT ACK.

000020
 000021
 000001

C=20
 D=21
 N=1

;TEST THAT WHEN THE PROCESSOR PRIORITY LEVEL (1) IS EQUAL TO THE PROGRAM
 ;INTERRUPT REQUEST LEVEL (1) THAT NO INTERRUPT OCCURS.

002300 010701
 002302 012706 000500
 002306 005077 176470
 002312 012777 002346 176466
 002320 012777 000340 176462
 002326 012767 000040 175442
 002334 012777 001000 176440
 002342 000240
 002344 000401
 002346 000000

T20: SCOPE
 MOV #STKPTR,SP
 CLR API
 MOV #T20A,APIRVEC
 MOV #PRTY7,APIRLVL
 MOV #PRTY1,PSW
 MOV #PIR1,API
 NOP
 BR T21
 T20A: HLT

;INITIALIZE THE STACK POINTER
 ;CLEAR PROGRAM INTERRUPT REQUEST REG.
 ;LOAD INTERRUPT VECTOR &
 ;STATUS
 ;SET PROCESSOR STATUS EQUAL TO 1
 ;REQUEST INTERRUPT AT LEVEL 1

;GO TO NEXT TEST
 ;PROGRAM INTERRUPTED.STATUS=REQ-
 ;UEST LEVEL=1.

000002
 000021
 000022

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

;TEST THAT WHEN THE PROCESSOR PRIORITY LEVEL (2) IS EQUAL TO THE PROGRAM
 ;INTERRUPT REQUEST LEVEL (2) THAT NO INTERRUPT OCCURS.

002350 010701
 002352 012706 000500
 002356 005077 176420
 002362 012777 002416 176416
 002370 012777 000340 176412
 002376 012767 000100 175372
 002404 012777 002000 176370
 002412 000240
 002414 000401
 002416 000000

T21: SCOPE
 MOV #STKPTR,SP
 CLR API
 MOV #T21A,APIRVEC
 MOV #PRTY7,APIRLVL
 MOV #PRTY2,PSW
 MOV #PIR2,API
 NOP
 BR T22
 T21A: HLT

;INITIALIZE THE STACK POINTER
 ;CLEAR PROGRAM INTERRUPT REQUEST REG.
 ;LOAD INTERRUPT VECTOR &
 ;STATUS
 ;SET PROCESSOR STATUS EQUAL TO 2
 ;REQUEST INTERRUPT AT LEVEL 2

;GO TO NEXT TEST
 ;PROGRAM INTERRUPTED.STATUS=REQ-
 ;UEST LEVEL=2.

000003
 000022
 000023

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

;TEST THAT WHEN THE PROCESSOR PRIORITY LEVEL (3) IS EQUAL TO THE PROGRAM
 ;INTERRUPT REQUEST LEVEL (3) THAT NO INTERRUPT OCCURS.

002420 010701
 002422 012706 000500
 002426 005077 176350
 002432 012777 002466 176346
 002440 012777 000340 176342
 002446 012767 000140 175322
 002454 012777 004000 176320
 002462 000240
 002464 000401

T22: SCOPE
 MOV #STKPTR,SP
 CLR API
 MOV #T22A,APIRVEC
 MOV #PRTY7,APIRLVL
 MOV #PRTY3,PSW
 MOV #PIR3,API
 NOP
 BR T23

;INITIALIZE THE STACK POINTER
 ;CLEAR PROGRAM INTERRUPT REQUEST REG.
 ;LOAD INTERRUPT VECTOR &
 ;STATUS
 ;SET PROCESSOR STATUS EQUAL TO 3
 ;REQUEST INTERRUPT AT LEVEL 3

;GO TO NEXT TEST


```

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

                                N=N+1
                                C=C+1
                                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      MOV      #STKPTR,SP              ;INITIALIZE THE STACK POINTER
002476 005077      CLR      @PI                     ;CLEAR PROGRAM INTERRUPT REQUEST REG.
002502 012777      MOV      #T23A,@PIRVEC          ;LOAD INTERRUPT VECTOR &
002510 012777      MOV      #PRTY7,@PIRLVL         ;STATUS
002516 012767      MOV      #PRTY4,PSW             ;SET PROCESSER STATUS EQUAL TO 4
002524 012777      MOV      #PIR4,@PI              ;REQUEST INTERRUPT AT LEVEL 4
002532 000240      NOP
002534 000401      BR       T24
002536 000000      T23A:  HLT                      ;GO TO NEXT TEST
                                           ;PROGRAM INTERRUPTED.STATUS=REQ-
                                           ;UEST LEVEL=4.

                                N=N+1
                                C=C+1
                                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      MOV      #STKPTR,SP              ;INITIALIZE THE STACK POINTER
002546 005077      CLR      @PI                     ;CLEAR PROGRAM INTERRUPT REQUEST REG.
002552 012777      MOV      #T24A,@PIRVEC          ;LOAD INTERRUPT VECTOR &
002560 012777      MOV      #PRTY7,@PIRLVL         ;STATUS
002566 012767      MOV      #PRTY5,PSW             ;SET PROCESSER STATUS EQUAL TO 5
002574 012777      MOV      #PIR5,@PI              ;REQUEST INTERRUPT AT LEVEL 5
002602 000240      NOP
002604 000401      BR       T25
002606 000000      T24A:  HLT                      ;GO TO NEXT TEST
                                           ;PROGRAM INTERRUPTED.STATUS=REQ-
                                           ;UEST LEVEL=5.

                                N=N+1
                                C=C+1
                                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      MOV      #STKPTR,SP              ;INITIALIZE THE STACK POINTER
002616 005077      CLR      @PI                     ;CLEAR PROGRAM INTERRUPT REQUEST REG.
002622 012777      MOV      #T25A,@PIRVEC          ;LOAD INTERRUPT VECTOR &
002630 012777      MOV      #PRTY7,@PIRLVL         ;STATUS
002636 012767      MOV      #PRTY6,PSW             ;SET PROCESSER STATUS EQUAL TO 6
002644 012777      MOV      #PIR6,@PI              ;REQUEST INTERRUPT AT LEVEL 6
002652 000240      NOP
002654 000401      BR       T26
002656 000000      T25A:  HLT                      ;GO TO NEXT TEST
                                           ;PROGRAM INTERRUPTED.STATUS=REQ-
                                           ;UEST LEVEL=6.

                                N=N+1
                                C=C+1

```


002

MAINDEC-11-DCKBN-C PROG INT RQST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 19
DBKBN.CP11

```

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.
T26:        SCOPE
            MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
            CLR     @PI             ;CLEAR PROGRAM INTERRUPT REQUEST REG.
            MOV     #T26A,@PIRVEC  ;LOAD INTERRUPT VECTOR &
            MOV     #PRTY7,@PIRLVL ;STATUS
            MOV     #PRTY7,PSW      ;SET PROCESSER STATUS EQUAL TO 7
            MOV     #PIR7,@PI      ;REQUEST INTERRUPT AT LEVEL 7
            NOP
            BR      T27             ;GO TO NEXT TEST
T26A:        HLT                   ;PROGRAM INTERRUPTED.STATUS=REQ-
                                   ;UEST LEVEL=7.

000010      N=N+1
000027      C=C+1
000030      D=D+1
000001      N=1
000000      M=0
            :TEST THAT WHEN THE PROCESSER PRIORITY (1) IS GREATER THAN THE PROGRAM INT-
            :ERRUPT REQUEST LEVEL (0) THAT NO INTERRUPT OCCURS.
T27:        SCOPE
            MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
            CLR     @PI             ;CLEAR PIRQ
            MOV     #T27A,@PIRVEC;  ;LOAD INTERRUPT VECTOR
            MOV     #PRTY7,@PIRLVL ;AND STATUS
            MOV     #PRTY1,PSW      ;SET PROCESSER STATUS EQUAL TO 1
            MOV     #PIR0,@PI      ;REQUEST INTERRUPT AT LEVEL 0
            NOP
            BR      T30             ;GO TO NEXT TEST
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.
T30:        SCOPE
            MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
            CLR     @PI             ;CLEAR PIRQ
            MOV     #T30A,@PIRVEC;  ;LOAD INTERRUPT VECTOR
            MOV     #PRTY7,@PIRLVL ;AND STATUS
            MOV     #PRTY2,PSW      ;SET PROCESSER STATUS EQUAL TO 2
            MOV     #PIR1,@PI      ;REQUEST INTERRUPT AT LEVEL 1
            NOP
            BR      T31             ;GO TO NEXT TEST
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

```


E02

MAINDEC-11-DCKBN-C PROG INT RQST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 20
DBKBNC.P11

```

;TEST THAT WHEN THE PROCESSER PRIORITY (3) IS GREATER THAN THE PROGRAM INT-
;ERRUPT REQUEST LEVEL (2) THAT NO INTERRUPT OCCURS.
003050 010701 000500 T31: SCOPE
003052 012706 175720 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 000500 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 000200 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 000500 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.

```


F02

MAINDEC-11-DCKBN-C PROG INT RQST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 21
DBKBNC.P11

```

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  #PTY7,@PIRLVL ;AND STATUS
003266 012767 000300 174502  MOV  #PTY6,PSW    ;SET PROCESSER STATUS EQUAL TO 6
003274 012777 020000 175500  MOV  #PIR5,@PI    ;REQUEST INTERRUPT AT LEVEL 5
003302 000240      NOP
003304 000401      BR   T35
003306 000000      T34A: HLT

```

```

;GO TO NEXT TEST
;ERROR PROGRAM INTERRUPTED WHEN
;STATUS=6, REQUEST LEVEL=5

```

```

000035
000036
000006
000007

```

```

C=C+1
D=D+1
M=M+1
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 003356 175456  MOV  #T35A,@PIRVEC; ;LOAD INTERRUPT VECTOR
003330 012777 000340 175452  MOV  #PTY7,@PIRLVL ;AND STATUS
003336 012767 000340 174432  MOV  #PTY7,PSW    ;SET PROCESSER STATUS EQUAL TO 7
003344 012777 040000 175430  MOV  #PIR6,@PI    ;REQUEST INTERRUPT AT LEVEL 6
003352 000240      NOP
003354 000401      BR   T36
003356 000000      T35A: HLT

```

```

;GO TO NEXT TEST
;ERROR PROGRAM INTERRUPTED WHEN
;STATUS=7, REQUEST LEVEL=6

```

```

000036
000037
000007
000010

```

```

C=C+1
D=D+1
M=M+1
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 001000 175360      MOV      #PIR1,@PI      ;REQUEST INTERRUPT AT LEVEL1
003422 000240      NOP
003424 000000      HLT
003426 000400      T36A: BR      T37      ;ERROR! PROGRAM FAILED TO INTERRUPT
;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
003476 000400      T37A: BR      T40      ;ERROR! PROGRAM FAILED TO INTERRUPT
;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
003546 000400      T40A: BR      T41      ;ERROR! PROGRAM FAILED TO INTERRUPT
;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

```


H02

MAINDEC-11-DCKBN-C PROG INT RQST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 23
DBKBN.C.P11

```

003552 012706 000500      MOV      #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003556 005077 175220      CLR      @PI          ;CLEAR PIRQ
003562 012777 003616 175216  MOV      #T41A,@PIRVEC      ;LOAD INTERRUPT VECTOR
003570 012777 000340 175212  MOV      #PRTY7,@PIRLVL      ;AND STATUS
003576 012767 000140 174172  MOV      #PRTY3,PSW        ;SET PROCESSER STATUS=3
003604 012777 010000 175170  MOV      #PIR4,@PI        ;REQUEST INTERRUPT AT LEVEL4
003612 000240      NOP
003614 000000      HLT
003616 000400      BR      T42          ;ERROR! PROGRAM FAILED TO INTERRUPT
                                         ;GO TO NEXT TEST

      000004      N=N+1
      000005      M=M+1
      000042      C=C+1
      000043      D=D+1
;TEST THAT WHEN THE PROCESSER PRIORITY (4) IS LESS THAN THE PROGRAM INT-
;ERRUPT REQUEST LEVEL(5) THAT AN INTERRUPT OCCURS.
003620 010701 000500      MOV      #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003622 012706 175150      CLR      @PI          ;CLEAR PIRQ
003626 005077 175150      MOV      #T42A,@PIRVEC      ;LOAD INTERRUPT VECTOR
003632 012777 003666 175146  MOV      #PRTY7,@PIRLVL      ;AND STATUS
003640 012777 000340 175142  MOV      #PRTY4,PSW        ;SET PROCESSER STATUS=4
003646 012767 000200 174122  MOV      #PIR5,@PI        ;REQUEST INTERRUPT AT LEVEL5
003654 012777 020000 175120  NOP
003662 000240      HLT
003664 000000      BR      T43          ;ERROR! PROGRAM FAILED TO INTERRUPT
003666 000400      BR      T43          ;GO TO NEXT TEST

      000005      N=N+1
      000006      M=M+1
      000043      C=C+1
      000044      D=D+1
;TEST THAT WHEN THE PROCESSER PRIORITY (5) IS LESS THAN THE PROGRAM INT-
;ERRUPT REQUEST LEVEL(6) THAT AN INTERRUPT OCCURS.
003670 010701 000500      MOV      #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003672 012706 175100      CLR      @PI          ;CLEAR PIRQ
003676 005077 175100      MOV      #T43A,@PIRVEC      ;LOAD INTERRUPT VECTOR
003702 012777 003736 175076  MOV      #PRTY7,@PIRLVL      ;AND STATUS
003710 012777 000340 175072  MOV      #PRTY5,PSW        ;SET PROCESSER STATUS=5
003716 012767 000240 174052  MOV      #PIR6,@PI        ;REQUEST INTERRUPT AT LEVEL6
003724 012777 040000 175050  NOP
003732 000240      HLT
003734 000000      BR      T44          ;ERROR! PROGRAM FAILED TO INTERRUPT
003736 000400      BR      T44          ;GO TO NEXT TEST

      000006      N=N+1
      000007      M=M+1
      000044      C=C+1
      000045      D=D+1
;TEST THAT WHEN THE PROCESSER PRIORITY (6) IS LESS THAN THE PROGRAM INT-
;ERRUPT REQUEST LEVEL(7) THAT AN INTERRUPT OCCURS.
003740 010701 000500      MOV      #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003742 012706 175030      CLR      @PI          ;CLEAR PIRQ
003746 005077 175030      MOV      #T44A,@PIRVEC      ;LOAD INTERRUPT VECTOR
003752 012777 004006 175026  MOV      #PRTY7,@PIRLVL      ;AND STATUS
003760 012777 000340 175022  MOV      #PRTY6,PSW        ;SET PROCESSER STATUS=6
003766 012767 000300 174002  MOV

```



```

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

000007      N=N+1
000010      M=M+1
000045      C=C+1
000046      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      %0
004016 012767 000340 173752      MOV      #PRTY7,PSW      ;LOCK OUT INTERRUPTS
004024 012777 177777 174750      MOV      #-1,API      ;REQUEST AN INTERRUPT AT ALL LEVELS
004032 012777 004056 174746      MOV      #LEVEL7,APIRVEC      ;LOAD INT. RQST. VECTOR
004040 012777 000340 174742      MOV      #PRTY7,APIRLVL      ;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,%0      ;SET INDICATOR BIT IN RO
004062 022777 177356 174712      CMP      #PIR7+PIR6+PIR5+PIR4+PIR3+PIR2+PIR1+PIA7,API
004070 001401                      BEQ      .+4      ;IS PIRQ CORRECT?
004072 000000                      HLT      ;INCORRECT PIRQ
004074 042777 100000 174700      BIC      #PIR7,API      ;DELETE REQUEST AT LEVEL 7
004102 012777 004120 174676      MOV      #LEVEL6,APIRVEC      ;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,%0      ;SET INDICATOR BIT
004124 022777 077314 174650      CMP      #PIR6+PIR5+PIR4+PIR3+PIR2+PIR1+PIA6,API
004132 001401                      BEQ      .+4      ;IS PIRQ CORRECT?
004134 000000                      HLT      ;ERROR! INCORRECT PIRQ
004136 042777 040000 174636      BIC      #PIR6,API      ;DELETE LEVEL 6 REQUEST
004144 012777 004162 174634      MOV      #LEVEL5,APIRVEC      ;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,%0      ;SET INDICATOR BIT
004166 022777 037252 174606      CMP      #PIR5+PIR4+PIR3+PIR2+PIR1+PIA5,API
004174 001401                      BEQ      .+4      ;IS PIRQ CORRECT?
004176 000000                      HLT      ;ERROR! INCORRECT PIRQ
004200 042777 020000 174574      BIC      #PIR5,API      ;DELETE LEVEL 5 REQUEST
004206 012777 004224 174572      MOV      #LEVEL4,APIRVEC      ;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,%0      ;SET INDICATOR BIT
004230 022777 017210 174544      CMP      #PIR4+PIR3+PIR2+PIR1+PIA4,API
004236 001401                      BEQ      .+4      ;IS PIRQ CORRECT?
004240 000000                      HLT      ;ERROR! INCORRECT PIRQ
004242 042777 010000 174532      BIC      #PIR4,API      ;DELETE LEVEL 4 REQUEST

```


J02

MAINDEC-11-DCKBN-C PROG INT RQST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 25
DBKBN.C.P11

```

004250 012777 004266 174530      MOV      #LEVEL3, @PIRVEC ;SET UP FOR LEVEL 3 REQUEST
004256 022626      POP2     ;RESTORE THE STACK
004260 005067 173512      CLR      PSW      ;ALLOW INTERRUPTS
004264 000000      HLT       ;ERROR! NO INTERRUPT

004266 052700 004000      LEVEL3: BIS      #PIR3, %0      ;SET INDICATOR BIT
004272 022777 007146 174502      CMP      #PIR3+PIR2+PIR1+PIA3, @PI ;IS PIRQ CORRECT?
004300 001401      BEQ       .+4      ;ERROR! INCORRECT PIRQ
004302 000000      HLT       ;DELETE LEVEL 3 REQUEST
004304 042777 004000 174470      BIC      #PIR3, @PI      ;SET UP FOR LEVEL 2 REQUEST
004312 012777 004330 174466      MOV      #LEVEL2, @PIRVEC ;RESTORE THE STACK
004320 022626      POP2     ;ALLOW INTERRUPTS
004322 005067 173450      CLR      PSW      ;ERROR! NO INTERRUPT
004326 000000      HLT

004330 052700 002000      LEVEL2: BIS      #PIR2, %0      ;SET INDICATOR BIT
004334 022777 003104 174440      CMP      #PIR2+PIR1+PIA2, @PI ;IS PIRQ CORRECT?
004342 001401      BEQ       .+4      ;ERROR! INCORRECT PIRQ
004344 000000      HLT       ;DELETE LEVEL 2 REQUEST
004346 042777 002000 174426      BIC      #PIR2, @PI      ;SET UP FOR LEVEL 1 REQUEST
004354 012777 004372 174424      MOV      #LEVEL1, @PIRVEC ;RESTORE THE STACK
004362 022626      POP2     ;ALLOW INTERRUPTS
004364 005067 173406      CLR      PSW      ;ERROR! NO INTERRUPT
004370 000000      HLT

004372 052700 001000      LEVEL1: BIS      #PIR1, %0      ;SET INDICATOR BIT
004376 022777 001042 174376      CMP      #PIR1+PIA1, @PI ;IS PIRQ CORRECT?
004404 001401      BEQ       .+4      ;ERROR! INCORRECT PIRQ
004406 000000      HLT       ;RESTORE THE STACK
004410 022626      POP2     ;CLEAR PROGRAM INT.RQST.REG.
004412 005077 174364      CLR      @PI
004416 005067 173354      CLR      PSW
004422 022700 177000      CMP      #177000, %0 ;WERE ALL LEVELS SERVICED
004426 001401      BEQ       .+4      ;ERROR! A LEVEL(S) NOT SERVICED
004430 000000      HLT       ;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

```


K02

MAINDEC-11-DCKBN-C PROG INT RQST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 26
DBKBN.C.P11

```

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 ASUME 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)+
004626 001401      BEQ      .+4          ;CHECK STATUS ON STACK
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      JSR      7, (2)          ;RETURN TO DECTAPE MONITOR
004724 000240      ENDAD: NOP
004726 000240      NOP
004730 000240      NOP
004732 010446      MOV      R4, -(SP)
004734 005004      CLR      R4
004736 012704 177777      MOV      #-1, R4
004742 005304      1$:  DEC      R4
004744 001376      BNE      1$
004746 012604      MOV      (SP)+, R4
004750 000167 174036      DONE1: JMP      START
000001      .END

```



```
PIA0 = 000000
PIA1 = 000042
PIA2 = 000104
PIA3 = 000146
PIA4 = 000210
PIA5 = 000252
```

362#	1103											
601#	604	619#	621	636#	638	653#	655	670#	672	687#	689	704#
706	721#	725	739#	743	757#	761	775#	779	793#	797	811#	815
829#	833	847#	853	868#	870	885#	887	902#	904	919#	921	936#
938	953#	955	970#									
602#	604	620#	621	637#	638	654#	655	671#	672	688#	689	705#
706	722#	725	740#	743	758#	761	776#	779	794#	797	812#	815
830#	833	848#	853	869#	870	886#	887	903#	904	920#	921	937#
938	954#	955	971#									
179#	364#											
1102	1104#											
1108	1119#											
163#	1080#	1081#	1097#									
1100#												
347	1109#											
158#												
171#	374	387	391	398	402	409	414	421	426	432	437	443
448	454	458	467	471	480	484	492	501	507	513	519	525
531	536	548	562	568	580	596	615	632	649	666	683	700
717	736	754	772	790	808	826	844	863	880	897	914	931
948	965	982	988	993	997	1002	1007	1012	1017	1022	1027	1032
1037	1042	1047	1053	1070	1072	1086	1087	1090	1093			
355#	360#	364	1100#	1101								
161#												
1039	1044#											
1029	1034#											
1019	1024#											
1009	1014#											
999	1004#											
990	994#											
979	985#											
724#	725	741#	743	759#	761	777#	779	795#	797	813#	815	831#
833	849#	852#	853	867#	870	884#	887	901#	904	918#	921	935#
938	952#	955	969#									
603#	604	618#	621	635#	638	652#	655	669#	672	686#	689	703#
706	720#	723#	725	742#	743	760#	761	778#	779	796#	797	814#
815	832#	833	850#	851#	853	866#	870	883#	887	900#	904	917#
921	934#	938	951#	955	968#							
155#												
162#												
356#	371	383#	384	388#	389	394#	395	399#	400	405#	406	410#
411	417#	418	422#	423	428#	429	433#	434	439#	440	444#	445
450#	451	455#	456	462#	464	469	481	488#	489#	490	497#	498
503#	504	509#	510	515#	516	521#	522	527#	528	532#	533	545#
560#	574#	578#	579#	589#	593#	608#	612#	625#	629#	642#	646#	659#
663#	676#	680#										

PIA6 = 000314
PIA7 = 000356
PIH 001004
PIRLVL 001010

PIRVEC 001006

```
PIRD = 000000
PIR1 = 001000
```

```
PIR10 = 100000
PIR2   = 002000
```

PIR3 = 004000

PIR4 = 010000

P1R5 = 020000

PIR6 = 040000

POP2 = 022626

PRTY0 = 000000
PRTY1 = 000040

PRTY10= 000340
PRTY2 = 000100

PRTY3 = 000140
PRTY4 = 000200

```
PRTYS = 000240
PRTYL = 000300
```

PRY7 = 000340

PSW = 177776

```
RESVEC= 000010
RO      =%000000
```

R1 =%000001

```
R2      =%.000000E
R3      =%.0000003
```

```
R4      =%.000004
R5      =%.000005
```

SCOPE = 010701

SP =%000006

207#	441	529	995									
208#	452	465	534	986								
357#	476*	477										
359#	544*	558*	576*	591*	610*	627*	644*	661*	678*	695*	712*	731*
743#	767*	785*	803*	821*	839*	859*	876*	893*	910*	927*	944*	961*
980#	1063*	1083*	1095	1096*								
358#	543*	557*	575*	590*	609*	626*	643*	660*	677*	694*	711*	730*
748#	766*	784*	802*	820*	838*	858*	875*	892*	909*	926*	943*	960*
979#	990*	999*	1009*	1019*	1029*	1039*	1062*	1082*	1095*			
192#	733											
193#	383	385	482	497	499	505	511	517	523	529	534	545
560	612	751	861	986	995	1005	1015	1025	1035	1044	1045	
200#												
194#	394	396	503	505	511	517	523	529	534	629	769	878
986	995	1005	1015	1025	1034	1035	1038					
195#	405	407	509	511	517	523	529	534	593	646	787	895
986	995	1005	1015	1024	1025	1028						
196#	417	419	515	517	523	529	534	663	805	912	986	995
1005	1014	1015	1018	1067	1084							
197#	428	430	521	523	529	534	578	680	823	929	986	995
1004	1005	1008										
198#	439	441	527	529	534	697	841	946	986	994	995	998
199#	450	452	532	534	714	963	985	986	989			
172#	373	551	564	582	598	991	1000	1010	1020	1030	1040	1048
181#	860											
182#	611	732	877									
189#												
183#	592	628	750	894								
184#	594	645	768	911								
185#	577	662	786	928	1063	1065						
186#	679	804	945									
187#	696	822	962									
188#	363	382	544	553	558	566	569	576	591	610	627	644
661	678	695	712	713	731	749	767	785	803	821	839	840
859	876	893	910	927	944	961	977	980	1083			
174#	363*	382*	546*	553*	559*	565	569*	577*	592*	594*	611*	628*
645#	662*	679*	696*	713*	732*	750*	768*	786*	804*	822*	840*	860*
877#	894*	911*	928*	945*	962*	977*	981*	992*	1001*	1011*	1021*	1031*
1041#	1050*	1068*										
159#												
148#	384*	385	389*	395*	396	400*	406*	407	411*	418*	419	423*
429#	430	434*	440*	441	445*	451*	452	456*	464*	465	469*	477*
478	481*	482	490*	498*	499	504*	505	510*	511	516*	517	522*
523	528*	529	533*	534	565*	566				</		

[illegible]

[illegible]

COMMEN	18							
ENDCOM	18							
ESCAPE	18							
GETPRI	18							
GETSWR	18							
MULT	18							
NEWTST	18							
PIRA	2118	604	621	638	655	672	689	706
PIRB	2128	725	743	761	779	797	815	833
PIRC	2138	852	870	887	904	921	938	955
POP	18							
PUSH	18							
REPORT	18							
SETPRI	18							
SETUP	18							
SKIP	18							
SLASH	18							
STARS	18							
SWRSU	18							
TYPBIN	18							
TYPDEC	18							
TYPNAM	18							
TYPNUM	18							
TYPOCS	18							
TYPOCT	18							
TYPTXT	18							
SSESCA	18							
SSNEWT	18							
SSSKIP	18							
.EQUAT	18							
.HEADE	18							
.KT11	18							
.SETUP	18							
.SWRHI	18							
.SACT1	18							
.SAPTB	18							
.SAPTH	18							
.SAPTY	18							
.SASTA	18							
.SCATC	18							
.SCMTA	18							
.SDB2D	18							
.SDB2O	18							
.SDIV	18							
.SEOP	18							
.SERRO	18							
.SERRT	18							
.SMULT	18							
.SPOWE	18							
.SRAND	18							
.SRDDE	18							
.SRDOC	18							
.SREAO	18							
.SR2AZ	18							
.SSAVE	18							
.SSB2D	18							

D03

MAINDEC-11-DCKBN-C PROG INT ROST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 34
DBKBNC.P11 CROSS REFERENCE TABLE -- MACRO NAMES

.SSB20	1*
.SSCOP	1*
.SSIZE	1*
.SSUPR	1*
.STRAP	1*
.STYB	1*
.STYD	1*
.STYDNT	1*
.STYPO	1*
.400A	1*
.1170	1*

MAINDEC-11-DCKBN-C PROG INT RQST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 36
DBKBN.C.F11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

Instruction	366	386	390	397	401	408	412	420	424	431	435	442	446	453	457
BEG	366	386	390	397	401	408	412	420	424	431	435	442	446	453	457
BIC	466	470	479	483	491	500	506	512	518	524	530	535	546	553	557
BIT	1006	1016	1026	1036	1046	1052	1069	1092	1102	1108					
BITT	989	998	1008	1018	1028	1038									
BNE	503	509	515	521	527	532	994	1004	1014	1024	1034	1044			
BPL	365														
BR	1117														
	1106														
	372	375	377	472	549	563	581	583	597	599	614	631	648	665	682
	699	715	735	753	771	789	807	825	843	864	881	898	915	932	949
	966	983	1071												
CLR	360	389	399	410	422	433	444	455	488	546	559	574	579	589	608
	625	642	659	676	693	710	729	747	765	783	801	819	837	857	874
	891	908	925	942	959	976	981	992	1001	1011	1021	1031	1041	1049	1050
CMP	1068	1073	1074	1081	1094	1096	1114								
	385	396	407	419	430	441	452	465	482	499	505	511	517	523	529
	534	566	986	995	1005	1015	1025	1035	1045	1051	1088	1101			
CMPB	478														
DEC	1116														
EMT	1085														
HALT	171	216	218	220	222	224	226	228	230	232	234	236	238	240	242
	244	246	248	250	252	254	256	258	260	262	264	266	268	270	272
	274	276	278	280	282	284	286	288	290	292	294	296	298	300	302
	304	306	308	310	312	314	316	318	320	322	324	326	328	330	332
	334	336	338	340	342										
INC	1100														
JMP	352	1103	1119												
JSR	1109														
MOV	361	362	363	364	370	376	382	383	384	389	394	395	400	405	406
	411	417	418	423	428	429	434	439	440	445	450	451	456	462	464
	469	481	490	497	498	504	510	516	522	528	533	543	544	5	

F03

MAINDEC-11-DCKBN-C PROG-INT RQST LOGIC MACY11 27(732) 27-SEP-76 10:40 PAGE 37
DBKBNC.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

.REM	1			
.REPT	215	604	725	853
.TITLE	131			

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

*DBKBNC,DBKBNC/SOL/CRF/PAGNUM/NL:TOC=SYSMAC.SML(400,10661,DBKBNC(400,45711
RUN-TIME: 23 25 2 SECONDS
RUN-TIME RATIO: 219/51=4.2
CORE USED: 33K (65 PAGES)

G03

Spooler runtime 4 Seconds, 24 KCS, 103 disk reads, 3 disk writes, 31 pages.

Shooter Firing 4 Seconds, 24 RCS, 103 disk Reads, 3 disk Writes, 31 pages
 /usr/bin/monitord, built on 2011-07-21 09:00:04.7% Date 01-Dec-76 17:26:10 Monitor IPC-0 602 (600) #END

[illegible]