

IDENTIFICATION

Product Code: / MAINDEC-15-D0EA-D (D)
Product Name: JMP Y - Interrupt Test
Date: / January 5, 1970
Maintainer: Diagnostic Group
Author: Edward P. Steinberger

27

1. ABSTRACT

The JMP Y - Interrupt Test determines if the PDP-15 will complete a JMP Y (where Y is some random value) instruction before it goes into program interrupt. This is done by setting a I/O flag and then transferring control to an ION/JMP Y instruction group (located at a random place in memory). The computer should complete the JMP Y instruction before the computer goes into program interrupt. If no error occurs, the ION/JMP Y instruction group is moved to other random memory locations and the test is repeated. Errors are indicated to the operator via the Teletype or error halts.

2. REQUIREMENTS

2.1 Equipment

Standard PDP-15 computer.

2.2 Storage

The program uses all of 4K memory for the program or as a test area. The program occupies memory from location 07400 to 07746 and tests all locations below 07400.

2.3 Preliminary Programs

Basic Instruction Tests

3. LOADING PROCEDURE

- a. Put HRI tape of program in reader (high speed if available).
- b. Set ADDRESS SWITCHES to 07400; the BANK MODE switch on a 1.
- c. Depress and release READ-IN key.

4. STARTING PROCEDURE

4.1 Control Switch Settings

The following is a table of accumulator switch settings and their action on the program:

<u>ACSwitch</u>	<u>Set As</u>	<u>Action</u>
0	1	Halt on error
	0	Don't halt on error
1	1	Don't print errors
	0	Print errors

<u>AC Switch</u>	<u>Set As</u>	<u>Action</u>
2	1	Ring bell on error
	0	Ring bell after N passes
3	1	Loop on current Y
	0	Don't loop on current Y
4	1	Loop on current location
	0	Don't loop on current location

N is an arbitrary number (initially 100₈) which is controlled by the LAW-N instruction in location 07400 and may be changed at the operator's discretion.

4.2 Starting Address

The starting address of the program is 07400.

4.3 Program and/or Operator Action

- a. ADDRESS SWITCHES to 07400.
- b. Set ACCUMULATOR SWITCHES to desired positions (see 4.1).
Normal setting is 500000.
- c. Depress I/O RESET
- d. Depress START

5. OPERATING PROCEDURE

5.1 Operational Switch Settings (see 4.1)

5.2 Subroutine Abstracts

None

5.3 Program and/or Operator Action

To put the program in the scope mode, the ACCUMULATOR SWITCH REGISTER should be set to 260000(don't halt, don't print, bell after N passes, loop on current Y, loop on current locations).

6. ERRORS

Unless AC switch 1 is a 1, errors will be printed on the teletype.

6.1 Error Halts and Description

There is one error halt inside the program at location 07546. Any program diagnosed errors will cause a halt at this location if AC switch 0 is a 1. The program stores HALT in all locations of the test area memory. If the computer does not go into program interrupt immediately after executing the JMP Y, the computer will halt at location Y.

6.2 Error Recovery

6.2.1 Program Diagnosed Error

If AC switch 0 is a 1, the computer will halt on a program diagnosed error. To recover from this type of error, reset AC switches 0 to 4 as necessary (see Section 4.1) and then depress CONTINUE.

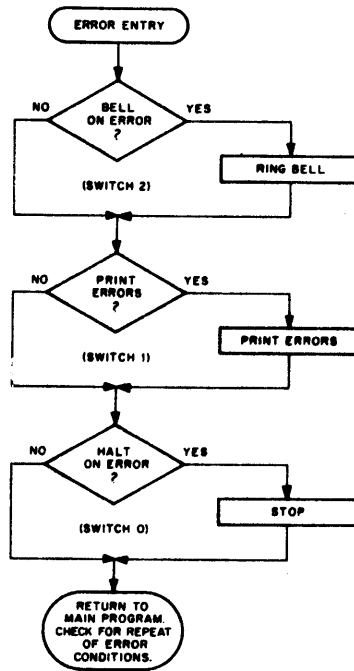
6.2.2 Interrupt Failures

Interrupt failures will cause a halt at location Y. To recover, reset AC switches 0 to 4 as necessary (see Section 4.1) and then start the computer at location 07400 (BEGIN) after depressing I/O RESET.

6.2.3 Test for ION, JMP Y, and Y

To test particular memory locations for the ION, JMP Y and/or Y, store the address of the ION in location 07733 (POINT1), that address + 1 in location 07734 (POINT2), the address Y in location 07735 (POINT3). Then set AC switches 3 and 4 to 1, depress I/O RESET, and start the computer at location 07400 (BEGIN). All addresses must be less than 07400 and not 00001.

6.3 Error Switch Hierarchy



6.4 Error Timeout Example

```
ION -JMP Y
JMP AT "Y" C(0)
001234 007654 001235
```

The above example shows that a JMP 7654 instruction was stored in location 1234 (it is implied that the ION is in 1233). The 1235 stored in location 00000 indicates the JMP was not completed before the computer went into program interrupt.

7. RESTRICTIONS

7.1 Starting Restrictions

(None)

7.2 Operating Restrictions

(None)

8. MISCELLANEOUS

8.1 Execution Time

Approximately 96 ms per ION/JMP Y instruction group.

9. PROGRAM DESCRIPTION

- a. The first function that is performed is that of initialization. A register to count loops and a location to assure timeout of the error message header are initialized, and the bell on the Teletype is run to raise the teleprinter flag to assure a flag for program interrupt.
- b. Then a check is made to see if the locations of the ION and JMP Y instructions should be changed (switch 4). If they are not changed, the program proceeds to c. If they are, a number is obtained from a random number generator, made into an address, and checked that it is below the program, not equal to Y, not equal to 00000 or 000001, and stored in POINT1 and incremented and stored in POINT2.
- c. Then a check is made to see if the number Y should be changed (switch 3). If it is not changed, the program proceeds to d. If it is, a number is obtained from a different random number generator than was used in b., made into an address, checked to see that it was below the program, not equal to location of ION or JMP Y instructions, not equal to 00001, and stored in POINT3.
- d. Then HALT is stored in all memory locations in the test area of memory. The ION instruction is stored, as well as the JMP Y instruction after it has been formed from Y and JMP. The AC and Link are then cleared and control is transferred to the ION/JMP Y instruction group.
- e. Upon return from the program interrupt, the contents of location 00000 are checked to make sure the proper number was stored. If not, the error subroutine is called.
- f. A check is then made to see if the scope mode (AC switches 3 and 4 a 1) has been requested and if so, control is immediately transferred back to the instruction group.
- g. If the instruction group is not being scoped, a check is made on ringing the bell (switch 2), after which control goes back to b.

10. LISTINGS

```

          .TITLE IONJMP
/
/JMP Y-INTERRUPT TEST
          .FULL
07400    .LOC 7400
07400    777700   BEGIN   LAW 17700
07401    047717   DAC COUNT           /SET UP TO COUNT LOOPS
07402    760207   LAW 207
07403    107560   JMS TYPE           /RING BELL TO SET I/O FLAG
07404    207746   LAC ZZZ+1
07405    047525   DAC ERROR1+13   /INITIALIZE ERROR TIMEOUT ROUTINE
07406    750004   HERE1  LAS
07407    507725   AND MASK2
07410    740200   SZA           /VARY CURRENT LOCATION
07411    607437   JMP HERE2     /NO
07412    107610   JMS RANDOM   /YES, GENERATE RANDOM ADDRESS
07413    507723   AND MASK
07414    047733   DAC POINT1   /STORE IN "ION" POINTER
07415    047734   DAC POINT2   /STORE IN "JMP Y" POINTER
07416    447734   ISZ POINT2   /AND INCREMENT
07417    741200   SNA           /IS "ION"=0?
07420    607412   JMP HERE1+4  /YES
07421    547730   SAD ONE      /HOW ABOUT 1?
07422    607412   JMP HERE1+4  /YES
07423    347743   TAD UPLIM    /IS THE "ION" POINTER
07424    740100   S#A          /INSIDE THIS PROGRAM?
07425    607412   JMP HERE1+4  /YES, GENERATE ANOTHER
07426    207734   LAC POINT2   /NO, NOW HOW ABOUT
07427    347743   TAD UPLIM    /THE "JMP Y" POINTER?
07430    740100   SMA          /IS IT OK?
07431    607412   JMP HERE1+4  /NO, TRY AGAIN
07432    207735   LAC POINT3   /OK SO FAR, NOW IS "Y" POINTER
07433    547733   SAD POINT1   /EQUAL TO "ION" POINTER
07434    607412   JMP HERE1+4  /YES
07435    547734   SAD POINT2   /NO, EQUAL TO "JMP Y" POINTER
07436    607412   JMP HERE1+4  /YES
07437    750004   HERE2  LAS
07440    507724   AND MASK1
07441    740200   SZA           /VARY "Y" POINTER?
07442    607460   JMP HERE3     /NO
07443    107621   JMS RANDUM   /YES, GENERATE RANDOM ADDRESS
07444    507723   AND MASK
07445    047735   DAC POINT3   /AND STORE IN POINT3
07446    547730   SAD ONE      /IS "Y"=1?
07447    607443   JMP HERE2+4  /YES
07450    347743   TAD UPLIM    /IS "Y" INSIDE THE PROGRAM?
07451    740100   SMA
07452    607443   JMP HERE2+4  /YES
07453    207735   LAC POINT3   /NO, CHECK IT AGAINST
07454    547733   SAD POINT1   /POINT1
07455    607443   JMP HERE2+4  /AND
07456    547734   SAD POINT2   /POINT2
07457    607443   JMP HERE2+4
          .EJECT

```

```

07460 107574   HERE3  JMS HALT           /STORE H&L IN MEMORY
07461 207721   LAC IONCON        /THEN STORE THE ION
07462 067733   DAC* POINT1      /VIA THE "ION" POINTER
07463 207735   LAC POINT3       /GET "Y"
07464 247722   XOR JMPCON       /FORM JMP "Y"
07465 067734   DAC* POINT2      /STORE VIA "JMP Y" POINTER
07466 754000   CLA!CLL          /CLEAR AC AND L
07467 627733   JMP* POINT1      /EXECUTE ION-JMP Y

/
07470 207735   RETURN LAC POINT3 /GET "Y"
07471 200000   LAC 0
07472 507723   AND MASK
07473 547735   SAD POINT3       /DOES C(0)="Y+1"
07474 741000   SKP              /YES
07475 107512   JMS ERROR1       /NO, ERROR
07476 750004   LAS
07477 742010   RTL
07500 742010   RTL              /MOVE BITS 3 + 4 INTO LINK AND AC
07501 740400   SNL              /LOOP ON CURRENT "Y"?
07502 607505   JMP .+3          /NO
07503 755100   SPA!CLA!CLL     /YES, LOOP ON CURRENT LOCATION?
07504 627733   JMP* POINT1     /YES, RETURN TO ION-JMP Y
07505 750004   LAS              /NO, SEE ABOUT RINGING BELL
07506 742010   RTL
07507 740100   SMA              /RING BELL?
07510 107550   JMS BELL         /YES
07511 607406   JMP HERE1
          .EJECT

```

```

/ERROR TYPEOUT SUBROUTINE
/
ERROR1  0
07512  000000  LAS
07513  750004  RTL
07514  742010  SMA
07515  740100  JMP .+3      /RING BELL?
07516  607521  JMP .+3      /NO
07517  760207  LAW 207
07520  107560  JMS TYPE
07521  750004  LAS
07522  740010  RAL
07523  741100  SPA          /PRINT ERRORS?
07524  607544  JMP .+20     /NO
07525  207745  LAC ZZZ
07526  107652  JMS MPRINT   /PRINT HEADER
07527  207727  LAC NEWINS
07530  047525  DAC ERROR1+13 /CHANGE SO THAT HEADER PRINTS ONLY ONCE
07531  207734  LAC POINT2
07532  107632  JMS PRINT    /PRINT LOCATION OF JMP Y
07533  760240  LAW 240
07534  107560  JMS TYPE    /1 SPACE
07535  207735  LAC POINT3
07536  107632  JMS PRINT    /PRINT "Y"
07537  760240  LAW 240
07540  107560  JMS TYPE    /1 SPACE
07541  200000  LAC 0
07542  107632  JMS PRINT    /PRINT C(0)
07543  107566  JMS CRLF    /CR-LF
07544  750004  LAS
07545  741100  SPA          /HALT ON ERROR?
07546  740040  XX          /YES
07547  627512  JMP* ERROR1 /EXIT
.EJECT

```

/USEFUL SUBROUTINES

```

/
07550 000000 BELL 0
07551 447717 ISZ COUNT
07552 627550 JMP* BELL
07553 407400 XCT BEGIN
07554 047717 DAC COUNT
07555 760207 LAW 207
07556 107560 JMS TYPE
07557 627550 JMP* BELL

/
07560 000000 TYPE 0
07561 507736 AND RUBOUT
07562 700406 TLS
07563 700401 TSF
07564 607563 JMP .-1
07565 627560 JMP* TYPE

/
07566 000000 CRLF 0
07567 760215 LAW 215
07570 107560 JMS TYPE
07571 760212 LAW 212
07572 107560 JMS TYPE
07573 627566 JMP* CRLF
.EJECT
```

```

/SUBROUTINE TO STORE HALTS IN MEMORY
/
HALT 0
      DZM PNTR
      LAC HLTCN
      DAC* PNTR
      ISZ PNTR
      LAC PNTR
      SAD UPLIM1
      SKP
      JMP HALT+2
      LAC CON1
      DAC 1
      JMP* HALT
/
/RANDOM NUMBER GENERATORS
/
RANDOM 0
      LAC RAND1
      RAL:CLL
      SZL
      TAD RAND1+1
      DAC RAND1
      JMP* RANDOM
/
RAND1 137
      3
/
RANDUM 0
      LAC RAND2
      RAL:CLL
      SZL
      TAD RAND2+1
      DAC RAND2
      JMP* RANDUM
/
RAND2 65
      3
      .EJECT

```

07574	000000
07575	147731
07576	207720
07577	067731
07600	447731
07601	207731
07602	547744
07603	741000
07604	607576
07605	207716
07606	040001
07607	627574
07610	000000
07611	207617
07612	744010
07613	741400
07614	347620
07615	047617
07616	627610
07617	000137
07620	000003
07621	000000
07622	207630
07623	744010
07624	741400
07625	347631
07626	047630
07627	627621
07630	000065
07631	000003

```

/OCTAL PRINT SUBROUTINE
/
PRINT 0
07632 000000 DAC TEMP
07633 047741 LAW 17772
07634 777772 DAC TALLY
07635 047740 LAC TEMP
07636 207741 RAL:CLL
07637 744010 RAL
07640 740010 RTL
07641 742010 DAC TEMP
07642 047741 AND SEVEN
07643 507737 TAD ASKII
07644 347715 JMS TYPE
07645 107560 LAC TEMP
07646 207741 ISZ TALLY
07647 447740 JMP .-10
07650 607640 JMP* PRINT
07651 627632

/
/MESSAGE PRINT SUBROUTINE
/
MPRINT 0
07652 000000 DAC PNTR1
07653 047732 LAC* PNTR1
07654 227732 RTR; RTR; RTR;
07655 742020
07656 742020
07657 742020
07660 742020 RTR; RAR
07661 740020
07662 107560 JMS TYPE
07663 547736 SAD RUBOUT
07664 627652 JMP* MPRINT
07665 227732 LAC* PNTR1
07666 107560 JMS TYPE
07667 547736 SAD RUBOUT
07670 627652 JMP* MPRINT
07671 447732 ISZ PNTR1
07672 607654 JMP MPRINT+2

/
/ERROR MESSAGE HEADER
MESS1 215212 /CR,LF
07674 311317 /I,0
07675 316255 /N,-
07676 312315 /J,M
07677 320240 /P,SP
07700 331215 /Y,CR
07701 212312 /LF,J
07702 315320 /M,P
07703 240301 /SP,A
07704 324240 /T,SP
07705 240240 /SP,SP
07706 242331 /",Y
07707 242240 /",SP
07710 240240 /SP,SP
07711 303250 /C,(

```

PAGE 7

JMP-Y

IONJMP

07712 260251
07713 215212
07714 377000

260251
215212
377000
.EJECT

/0,)
/CR,LF
/END

```
                /CONSTANTS AND VARIABLES
                /
07715  000260  ASCII  260
07716  607470  CON1   JMP RETURN
07717  000000  COUNT  0
07720  740040  HLTC0N HLT
07721  700042  IONCON ION
07722  600000  JMPCON JMP
07723  017777  MASK   17777
07724  040000  MASK1  40000
07725  020000  MASK2  20000
07726  010000  MASK3  10000
07727  607531  NEWINS JMP ERROR1+17
07730  000001  ONE    1
07731  000000  PNTR   0
07732  000000  PNTR1  0
07733  000002  POINT1 2
07734  000003  POINT2 3
07735  000004  POINT3 4
07736  000377  RUBOUT 377
07737  000007  SEVEN  7
07740  000000  TALLY  0
07741  000000  TEMP   0
07742  000002  TWO    2
07743  770400  UPLIM  -BEGIN
07744  007400  UPLIM1 BEGIN
07745  007673  ZZZ    MESS1
07746  207745  LAC ZZZ

                /
000000  .END
        SIZE=07747  NO ERROR LINES
```

ASKII	07715
REGIN	07400
BELL	07550
CLOF	700004
CLON	700044
CLSF	700001
CON1	07716
COUNT	07717
CRLF	07566
EEM	707702
ERROR1	07512
HALT	07574
HERE1	07406
HERE2	07437
HERE3	07460
HLTCON	07720
IONCON	07721
JMPCON	07722
KRB	700312
KSF	700301
LEM	707704
MASK	07723
MASK1	07724
MASK2	07725
MASK3	07726
MESS1	07673
MPRINT	07652
NEWINS	07727
ONE	07730
PCF	700202
PNTR	07731
PNTR1	07732
POINT1	07733
POINT2	07734
POINT3	07735
PRINT	07632
PSA	700204
PSB	700244
PSF	700201
RANDOM	07610
RANDUM	07621
RAND1	07617
RAND2	07630
RCF	700102
RETURN	07470
RRB	700112
RSA	700104
RSB	700144
RSF	700101
RUROUT	07736
SEVEN	07737
TALLY	07740
TCF	700402
TEMP	07741
TLS	700406

TSF	700401
TWO	07742
TYPE	07560
UPLIM	07743
UPLIM1	07744
ZZZ	07745
.EOT	00000

.EOT	00000
REGIN	07400
HERE1	07406
HERE2	07437
HERE3	07460
RETURN	07470
ERROR1	07512
BELL	07550
TYPE	07560
CRLF	07566
HALT	07574
RANDOM	07610
RAND1	07617
RANDUM	07621
RAND2	07630
PRINT	07632
MPRINT	07652
MESS1	07673
ASKII	07715
CON1	07716
COUNT	07717
HLTCON	07720
IONCON	07721
JMPCON	07722
MASK	07723
MASK1	07724
MASK2	07725
MASK3	07726
NEWINS	07727
ONE	07730
PNTR	07731
PNTR1	07732
POINT1	07733
POINT2	07734
POINT3	07735
RUBOUT	07736
SEVEN	07737
TALLY	07740
TEMP	07741
TWO	07742
UPLIM	07743
UPLIM1	07744
ZZZ	07745
CLSF	700001
CLOF	700004
CLON	700044
RSF	700101
RCF	700102
RSA	700104
RRB	700112
RSB	700144
PSF	700201
PCF	700202
PSA	700204
PSB	700244

KSF	700301
KRR	700312
TSF	700401
TCF	700402
TLS	700406
EEM	707702
LEM	707704