IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DCKBA TO DCKBE-B-D

PRODUCT NAME: PDP11/45-11/40 BASIC CP TESTS

DATE CREATED: 15 NOV 1972

MAINTAINER: DIAGNOSTIC GROUP

AUTHORS: JOHN ADAMS

THIS GROUP OF TESTS CONSIST OF:

MAINDEC NO.	TEST FUNCTION	PROCESSOR
DCKBA-A	SXT INSTRUCTION	11/45,11/40
DCKBB-A	SOR INSTRUCTION	11/45,11/40
DCKBC-A	XOR INSTRUCTION	11/45,11/40
DCKBD-B	MARK INSTRUCTION	11/45,11/40
DCKBE-B	RTT INSTRUCTION	11/45,11/40

```
1                                       .TITLE  MAINDEC-11-DCKBA-A PDP11/25-11/45 SXT INST TEST
                                        .NLIST  MC,MD,SEQ
                                        .LIST   ME
                                        .ABS
                                ;TEST DCKBAA- TEST OF THE SXT INSTRUCTION AND INCLUDES DATA CHECKS AND
                                ;CONDITION CODES CHECKS.

                                ;       THE SXT INSTRUCTION EXTENDS THE SIGN (AS DETERMINED BY THE 'N'
                                ;BIT IN THE STATUS WORD) INTO THE DESTINATION ADDRESS, IE; IF 'N' IS
                                ;= TO 1 THEN THE DESTINATION RESULT WILL BE -1 (ALL 1'S), AND IF 'N' IS
                                ;= TO 0 THEN THE DESTINATION RESULT WILL BE 0 (ALL 0'S). THE CONDITION
                                ;CODES ARE AFFECTED AS FOLLOWS;
                                ;       'Z' IS SET IF THE RESULT IS 0,OTHERWISE 'Z' IS CLEARED
                                ;       'N' IS SET IF THE RESULT IS NEGATIVE,CLEARED IF POSITIVE
                                ;       'C' IS UNCHANGED
                                ;       'V' IS CLEARED

                                ;ITEMS NOT TESTED BY THIS TEST
                                ;       1) THAT SXT WILL SET ALL BITS IN THE PSW WHEN 1'S ARE EXTENDED (EX-
                                ;       CLUSIVE OF THE 'T' BIT)
                                ;STARTING PROCEEDURE
                                ;       LOAD ADDRESS =200
                                ;       PRESS START
                                ;       STACK POINTER IS SET AT 500
                                ;       BELL WILL RING WHEN TEST IS COMPLETE

            000022                      HLT=HALT
            010701                      SCOPE=010701                    ;MOV PC,R1
            177776                      PSW=177776                  ;ADDRESS OF PROCESSER STATUS WORD
            177770                      UBREAK=177770               ;ADDRESS OF MICRO BREAK REGISTER
            177564                      TPCSR=177564                ;ADDRESS OF TELEPRINTER CSR
            177566                      TPBUF=177566                ;ADDRESS OF TELEPRINTER BUFFER
            177570                      SWR=177570                  ;ADDRESS OF CONSOLE SWITCH REGISTER
            177570                      DISPLAY=177570              ;ADDRESS OF DISPLAY REGISTER
                                ;****************INITIAL STACK POINTER=0500**********************
            000500              STKPTR=0500                         ;INITIAL STACK POINTER
            000002                      .=0
000000      000002                      .+2
000002      000000                      HALT
000004      000026                      .+2
000006      000023                      HALT
000010      000012                      .+2
000012      000000                      HALT
000014      000016                      .+2
000016      000022                      HALT
000020      000022                      .+2
000022      000000                      HALT
000024      000026                      .+2
000026      000022                      HALT
000030      000032                      .+2
000032      000000                      HALT
000034      000036                      .+2
000036      000000                      HALT
000040      000042                      .+2
```

1.0 ABSTRACT

THIS IS THE FIRST 5 OF 15 TESTS THAT INCREMENTALLY TEST AND ISOLATE SIMPLE MALFUNCTIONS IN THE PDP 11/45,11/40. THE TESTS SHOULD BE RUN IN THE INDICATED ALPHABETIC SEQUENCE. THERE ARE ADDITIONAL TESTS FOR MORE COMPLEX MALFUNCTIONS. ALL TESTS ARE EXECUTED IN KERNEL MODE ONLY EXCEPT FOR TEST CCKBE (11/45 ONLY).

2.0 REQUIREMENTS

2.1 EQUIPMENT

PDP11/45 OR PDP 11/40

2.2 STORAGE

PROGRAM STORAGE = THE PROGRAMS USE ALL OF A 4KW MEMORY WITH THE EXCEPTION OF 17500 TO 17776 (WHICH IS RESERVED FOR THE BOOT AND ABSOLUTE LOADER).

2.3 PRELIMINARY PROGRAMS

TESTS T0-T13(D0AA-D0MA)

3.0 LOADING PROCEDURE

LOAD PROGRAM INTO MEMORY USING ABS LOADER.

4.0 STARTING PROCEDURE

LOAD ADDRESS 200, PRESS START, THE PROGRAM WILL LOOP, AND RING BELL ON COMPLETION. PASS COUNT MAY BE MONITORED IN THE DISPLAY REGISTER (11/45 ONLY), AND IS STORED IN ADDRESS 1000.

5.0 OPERATING PROCEDURE

5.1 OPERATIONAL SWITCH SETTINGS

SW<08>=1 LOAD PDP11/45 MICRO BREAK REGISTER WITH VALUE IN SW<00-07>. (AT START OF TEST ONLY).

5.2 SUBROUTINE ABSTRACTS

5.2.1 SCOPE

SCOPE IS A MOVE PC, R1(010701) AND STORES THE PC+2 IN R1 (OR R11 IF THE REGISTER SET BIT IS SET IN THE PDP11/45 PROCESSOR STATUS WORD.) THUS, R1 (R11) MAY BE USED AS A TAG TO DETERMINE THE LAST TEST SUCESSFULLY COMPLETED.

5.2.2 HLT

HLT IS A HALT INSTRUCTION AND IS EXECUTED WHENEVER A HARDWARE MALFUNCTION IS DETECTED BY THE DIAGNOSTIC, THE ADDRESS LIGHTS DISPLAY THE PC=PC+2 OF THE HALT INSTRUCTION IN AN 11/40 OR 11/45.

5.3 PROGRAM AND/OR OPERATOR ACTION

5.3.1 PASS COUNT (ICNT)

THE NUMBER OF PROGRAM PASSES COMPLETED IS CONTAINED IN ADDRESS ICNT (1020). THIS ADDRESS MAY BE EXAMINED TO DETERMINE IN WHICH PASS THE ERROR OCCURED. NOTE, THE PASS COUNT IS DISPLAYED IN THE DISPLAY REGISTER IN THE PDP11/45.

6.0 ERRORS

6.1 TEST ERRORS WILL CAUSE A HALT

FALSE TRAP/INTERRUPT ERRORS = THE PROGRAM WILL HALT AT THE TRAP VECTOR ADDRESS +2.

6.2 ERROR RECOVERY

TEST ERRORS = PRESS CONTINUE OR LOOP TEST (SEE SEC 6.3)

TRAP ERRORS = DETERMINE WHERE ERROR OCCURRED (SEE SEC 8)

6.3 ERROR LOOPING

TO LOOP ON AN ERROR REPLACE THE HLT INSTRUCTION WITH A BRANCH BACK TO THE PREVIOUS SCOPE INSTRUCTION. NOTE THAT IF THE ERROR IS INTERMITTENT THAT THE TEST WILL DROP THROUGH THE HLT AND PROCEED TO THE NEXT TEST, THEREFORE TO LOOP THE TEST CONTINUOSLY REPLACE THE BEQ .+4 INSTRUCTION IMMEDIATLY

PRECEEDING THE HLT WITH THE BRANCH BACK TO THE PREVIOUS SCOPE.

TO LOOP 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 INST,
TRAPVEC+10:     000006  ;RETURN TO TEST AT 'SCOPE'
```

RESTORE ALL LOCATIONS BEFORE PROCEEDING TO NEXT TEST(S),

7.? RESTRICTIONS

THESE PROGRAMS MUST BE LOADED IN THE LOWER 4K OF MEMORY,

7.1 STARTING RESTRICTION

ALL PROGRAMS MUST BE INITIALLY STARTED AT 200 AND MAY BE STARTED AT A SCOPE INSTRUCTION THEREAFTER,

7.2 OPERATIONAL RESTRICTION

NONE

8.? MISCELLANEOUS

IF A HALT OCCURES IN THE TRAP OR INTERRUPT VECTOR AREA, EXAMINE REGISTER 6 (THE STACK POINTER), THE CONTENTS OF R6 CONTAINS THE ADDRESS WHERE THE PC OF THE INSTRUCTION THAT CAUSED THE TRAP IS STORED,

8.1 EXECUTION TIME

ALL TESTS TAKE APPROXIMATELY 1 MINUTE EACH ON AN 11/45 WITH CORE MEMORY,

8.2 STACK POINTER

ALL PROGRAMS INITIALLY SET THE STACK POINTER AT 500

9.0 PROGRAM DESCRIPTION

DCKBA THIS IS A TEST OF THE SXT INSTRUCTION AND INSURES CORRECT RESULTS AND CONDITION CODE OPERATION. THE SXT INSTRUCTION IS TESTED IN ALL ADDRESS MODES IN A GENERAL REGISTER AND THE PC.

DCKBB THIS IS A TEST OF THE SOS INSTRUCTION AND INSURES CORRECT BRANCHING AND CONDITION CODE OPERATION.

DCKBC THIS IS A TEST OF THE XOR INSTRUCTION AND INSURES CORRECT RESULTS AND CONDITION CODE OPERATION. THE XOR INSTRUCTION IS EXECUTED USING VARIOUS OPERANDS AND IS EXECUTED USING ALL ADDRESS MODES USING A GENERAL REGISTER AND THE PC.

DCKBD THIS IS A TEST OF THE MARK INSTRUCTION. THE TEST EXECUTES THE MARK INSTRUCTION USING ALL VALUES OF 'N' AND CHECKS RESULTS. CORRECT CONDITION CODE OPERATION IS ALSO TESTED.

DCKBE THIS IS A TEST OF THE RTT AND RTI INSTRUCTIONS AND USES 'T' BIT TRAPS IN THE TEST. PROPER STACK OPERATION IS TESTED AND ALSO PROPER STATUS CHANGES ARE TESTED.

```
000242  000000          HALT
000244  000046          .+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
```



```
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  000002                  HALT

        000000                  .=0
000000  000167  001034          JMP     SXT07


        000000          DEST=%0
        000200                  .=200
000200  000167  000576          JMP     START
        001000                  .=1000
001000  000000          ICNT:   0                       ;PASS COUNT
001002  005067  177772  START:  CLR     ICNT            ;CLEAR PASS COUNT
001006  012706  000500  BEGIN:  MOV     #STKPTR,%6      ;SET STACK POINTER
001012  016737  177762  177570  MOV     ICNT,@#DISPLAY  ;DISPLAY PASS COUNT
001022  032737  000400  177570  BIT     #400,@#SWR      ;LOAD MICRO BREAK REGISTER?
001026  001403                  BEQ     .+10            ;BRANCH IF NOT
001030  113737  177570  177770  MOVB    @#SWR,@#UBREAK  ;LOAD MICRO BREAK WITH SR0=7

                        ;TEST THAT WHEN N IS CLEAR SXT CLEARS THE DESTINATION
001036  010701                  SCOPE
001040  012700  177777          MOV     #-1,DEST
001044  000277                  SCC                     ;SET ALL CONDITION CODES
001046  000250                  CLN                     ;CLEAR N
001050  006700                  SXT     DEST            ;EXTEND N BIT (0) INTO DESTINATION
001052  005700                  TST     DEST            ;DESTINATION=0?
001054  001401                  BEQ     .+4
001056  000000                  HLT                     ;DESTINATION OTHER THAN 0

                        ;TEST THAT WHEN N IS SET THAT SXT SETS ALL BITS IN THE DESTINATION
001060  010701                  SCOPE
001062  005000                  CLR     DEST
001064  000257                  CCC                     ;CLEAR ALL CONDTION CODES
001066  000270                  SEN                     ;SET N
001070  006700                  SXT     DEST            ;EXTEND N BIT (1) INTO DESTINATION
001072  020027  177777          CMP     DEST,#-1        ;DID ALL BITS SET IN DESTINATION
001076  001401                  BEQ     .+4
001100  000000                  HLT                     ;DESTINATION OTHER THAN 177777

                        ;TEST THAT N IS CLEAR AFTER SXT EXTENDS 0'S INTO THE DESTINATION
001102  010701                  SCOPE
001104  012700  177777          MOV     #-1,DEST
001110  000250                  CLN
001112  006700                  SXT     DEST
001114  100001                  BPL     .+4
001116  000000                  HLT

                        ;TEST THAT N IS SET AFTER SXT EXTENDS 1'S INTO THE DESTINATION
001120  010701                  SCOPE
001122  005000                  CLR     DEST
001124  000270                  SEN
001126  006700                  SXT     DEST
001130  100401                  BMI     .+4
```



```
001132  000000                  HLT

                        ;TEST THAT SIGN EXTENDS PROPERLY AFTER A MOVE INSTRUCTION (N SET)
001134  010701                  SCOPE
001136  005000                  CLR     DEST
001140  012767  177777  177774  MOV     #-1,,.+2
001146  006700                  SXT     DEST
001150  020027  177777          CMP     DEST,#-1
001154  001401                  BEQ     .+4
001156  000000                  HLT
```

```
                                ;TEST THAT SIGN EXTENDS PROPERLY AFTER A MOVE INSTRUCTION (N CLEAR)
001160  010701                        SCOPE
001162  012700  177777                MOV     #-1,DEST
001166  012767  000000  177774        MOV     #0,,+2
001174  006700                        SXT     DEST
001176  020027  000000                CMP     DEST,#0
001202  001401                        BEQ     .+4
001204  000000                        HLT

                                ;TEST THAT SIGN EXTENDS PROPERLY AFTER VARIOUS BYTE INSTRUCTIONS
                                ;NOTE: THESE TESTS MUST BE RUN SEQUENTIALLY OR TEMP & TEMP+2 MUST
                                ;BE MANUALLY LOADED BEFORE STARTING THE TEST,THE COMMENT AT THE BEG-
                                ;INNING OF THE TEST SHOWS CONTENTS OF TEMP & TEMP+2,
                                                        ;(TEMP+1),(TEMP)/(TEMP+3),(TEMP+2)
001206  005000                        CLR     DEST            ;?,?/?,?
001210  005067  000226                CLR     TEMP            ;0,0/?,?
001214  105167  000223                COMB    TEMP+1          ;-1,0/?,?
001220  006700                        SXT     DEST
001222  022700  177777                CMP     #-1,DEST
001226  001401                        BEQ     .+4
001230  000000                        HLT                     ;ERROR! SIGN EXTEND FAILED (N=1)
001232  010701                        SCOPE

001234  005000                        CLR     DEST            ;-1,0/?,?
001236  105367  000200                DECB    TEMP            ;-1,-1/?,?
001242  006700                        SXT     DEST
001244  022700  177777                CMP     #-1,DEST
001250  001401                        BEQ     .+4
001252  000000                        HLT                     ;ERROR! SIGN EXTEND FAILED (N=1)
001254  010701                        SCOPE

001256  005070                        CLR     DEST            ;-1,-1/?,?
001260  156767  000157  000156        BISB    TEMP+1,TEMP+2   ;-1,-1/?,-1
001266  006700                        SXT     DEST
001270  022700  177777                CMP     #-1,DEST
001274  001401                        BEQ     .+4
001276  000000                        HLT                     ;ERROR! SIGN EXTEND FAILED (N=1)
001300  010701                        SCOPE

001302  005000                        CLR     DEST            ;-1,-1/?,-1
001304  116767  000134  000133        MOVB    TEMP+2,TEMP+3   ;-1,-1/-1,-1
001312  006700                        SXT     DEST
001314  022700  177777                CMP     #-1,DEST
001320  001401                        BEQ     .+4
001322  000000                        HLT                     ;ERROR! SIGN EXTEND FAILED (N=1)
001324  010701                        SCOPE

001326  012700  177777                MOV     #-1,DEST        ;-1,-1/-1,-1
001332  146767  000105  000105        BICB    TEMP+1,TEMP+3   ;-1,-1/0,-1
001340  006700                        SXT     DEST
001342  005700                        TST     DEST
001344  001401                        BEQ     .+4
001346  000000                        HLT                     ;ERROR! SIGN EXTEND FAILED (N=0)
```



```
001350  010701                        SCOPE

001352  012700  177777                MOV     #-1,DEST        ;-1,-1/0,-1
001356  000261                        SEC                     ;SET CARRY
001360  105567  000097                ADCB    TEMP+1          ;0,-1/?,-1
001364  006700                        SXT     DEST
001366  005700                        TST     DEST
001370  001401                        BEQ     .+4
001372  000000                        HLT                     ;ERROR! SIGN EXTEND FAILED (N=0)
001374  010701                        SCOPE

001376  012700  177777                MOV     #-1,DEST        ;0,-1/0,-1
001402  105267  000034                INCB    TEMP            ;0,0/0,-1
001406  006700                        SXT     DEST
001410  005700                        TST     DEST
001412  001401                        BEQ     .+4
001414  000000                        HLT                     ;ERROR! SIGN EXTEND FAILED (N=0)
001416  010701                        SCOPE

001420  012700  177777                MOV     #-1,DEST        ;0,0/0,-1
001424  105167  000014                COMB    TEMP+2          ;0,0/0,0
001430  006700                        SXT     DEST
001432  005700                        TST     DEST
001434  001404                        BEQ     TZ
001436  000000                        HLT                     ;ERROR! SIGN EXTEN FAILED (N=0)
001440  010701                        SCOPE
001442  000000          TEMP10
                        001446          .+2,+2

                                ;TEST THAT Z BIT IS SET AFTER SXT EXTENDS 0'S
001446  010701          TZ)           SCOPE
001450  012700  177777                MOV     #-1,DEST
001454  000262                        SEN                     ;CLEAR N
001456  006700                        SXT     DEST            ;EXTEND N BIT (0) INTO DESTINATION
001460  001401                        BEQ     .+4             ;BRANCH IF Z IS SET
001462  000000                        HLT                     ;Z NOT SET AFTER 0'S WERE EXTENDED

                                ;TEST THAT Z BIT IS CLEAR AFTER SXT EXTENDS 1'S
001464  010701                        SCOPE
001466  005000                        CLR     DEST
001470  000272                        SEN
001472  006700                        SXT     DEST
001474  001001                        BNE     .+4
001476  000000                        HLT


                                ;TEST THAT THE CARRY BIT (C) IS UNCHANGED BY SXT (C=0,N=0)
001500  010701                        SCOPE
001502  000277                        CLC                     ;CLEAR ALL CONDITION CODES
001504  006700                        SXT     DEST            ;EXTEND SIGN
001506  103001                        BCC     .+4             ;BRANCH IF CARRY IS CLEAR
001510  000000                        HLT                     ;ERROR, CARRY SET

                                ;TEST THAT THE CARRY BIT IS UNCHANGED BY SXT (C=0,N=1)
```

```
001512  010701                  SCOPE
001514  000277                  SCC
001516  000270                  SEN                     ;C=0,N=1
001520  006700                  SXT     DEST            ;EXTEND SIGN
001522  103001                  BCC     .+4             ;BRANCH IF CARRY IS CLEAR
001524  000000                  HLT                     ;ERROR! CARRY SET

                                ;TEST THAT THE CARRY BIT IS UNCHANGED BY SXT (C=1,N=0)
001526  010701                  SCOPE
001530  000257                  CCC
001532  000261                  SEC                     ;C=1,N=0
001534  006700                  SXT     DEST            ;EXTEND SIGN
001536  103401                  BCS     .+4             ;BRANCH IF CARRY IS SET
001540  000000                  HLT                     ;ERROR! CARRY CLEARED

                                ;TEST THAT CARRY IS UNCHANGED BY SXT (C=1,N=1)
001542  010701                  SCOPE
001544  000277                  SCC                     ;C=1,N=1
001546  006700                  SXT     DEST            ;EXTEND SIGN
001550  103401                  BCS     .+4             ;BRANCH IF CARRY SET
001552  000000                  HLT                     ;ERROR! CARRY CLEARED

                                ;TEST THAT THE V BIT IS CLEARED BY SXT (V=0,N=0)
001554  010701                  SCOPE
001556  000257                  CCC                     ;V=0,N=0
001560  006700                  SXT     DEST            ;EXTEND SIGN
001562  102001                  BVC     .+4             ;BRANCH IF V IS CLEAR
001564  000000                  HLT                     ;ERROR! V SET

                                ;TEST THAT V IS CLEARED BY SXT (V=0,N=1)
001566  010701                  SCOPE
001570  000257                  CCC
001572  000270                  SEN                     ;V=0,N=1
001574  006700                  SXT     DEST            ;EXTEND SIGN
001576  102001                  BVC     .+4             ;BRANCH IF V IS CLEAR
001600  000000                  HLT                     ;ERROR! V SET

                                ;TEST THAT V IS CLEARED BY SXT (V=1,N=0)
001602  010701                  SCOPE
001604  000277                  SCC
001606  000250                  CLN                     ;C=1,N=0
001610  006700                  SXT     DEST            ;EXTEND SIGN
001612  102001                  BVC     .+4             ;BRANCH IF V IS CLEAR
001614  000000                  HLT                     ;ERROR! V SET
```



```
                                ;TEST THAT V IS CLEARED BY SXT (C=1,N=1)
001616  010701                  SCOPE
001620  000277                  SCC                     ;V=1,N=1
001622  006700                  SXT     DEST            ;EXTEND SIGN
001624  102001                  BVC     .+4     ;BRANCH IF V IS CLEAR
001626  000000                  HLT                     ;ERROR! V REMAINED SET

        000007          DEST=%7
                        ;TEST THAT SIGN EXTENDS INTO R7 (N=0),
001630  010701                  SCOPE
001632  000257                  CCC                     ;N=0
001634  006707                  SXT     DEST            ;EXTEND 0'S INTO THE PC
001636  000000                  HLT                     ;ERROR! PC SHOULD'VE GONE TO 0
001640  000402          SXT07:  BR      MODE67

        001642          DEST=.
        001644                  .=.+2
001644  000000          IDEST:  0
                        ;TEST DESTINATION MODE 67
001646  010701          MODE67: SCOPE
001650  005067  177766          CLR     DEST
001654  000270                  SEN                     ;SET N
001656  006767  177760          SXT     DEST            ;EXTEND 1'S INTO DEST
001662  026727  177754  177777  CMP     DEST,#-1        ;DID 1'S EXTEND
001670  001401                  BEQ     .+4
001672  000000                  HLT                     ;1'S FAILED TO EXTEND

                                ;TEST DESTINATION MODE 27
001674  010701                  SCOPE
001676  005067  000004          CLR     MODE27
001702  000270                  SEN                     ;SET N
001704  006727                  SXT     (7)+            ;EXTEND 1'S INTO NEXT LOCATION
001706  000002          MODE27: 2
001710  026727  177772  177777  CMP     MODE27,#-1      ;DID 1'S EXTEND
001716  001401                  BEQ     .+4
001720  000000                  HLT

                                ;TEST DESTINATION MODE 37
001722  010701                  SCOPE
001724  012737  177777  001642  MOV     #-1,@#DEST
001732  000250                  CLN                     ;CLEAR N
001734  006737  001642          SXT     @#DEST          ;EXTEND 0'S
001740  023727  001642  000000  CMP     @#DEST,#0       ;DID 0'S EXTEND
001746  001401                  BEQ     .+4
001750  000000                  HLT

                                ;TEST DESTINATION MODE 77
001752  010701                  SCOPE
001754  012767  001644  177660  MOV     #IDEST,DEST
001762  012777  177777  177652  MOV     #-1,@DEST
001770  000250                  CLN
001772  006777  177644          SXT     @DEST
001776  027727  177640  000000  CMP     @DEST,#0
002004  001401                  BEQ     .+4
```

```
002206  000000                          HLT

                                ;TEST DESTINATION MODE 1
002210  010701                          SCOPE
002212  012700  002320                  MOV     #MODE1,%0
002216  005002                          CLR     %2
002220  000261                          SEC
002222  005622                          SBC     %2
002224  006710                          SXT     (0)
002226  022767  177777  000264          CMP     #-1,MODE1
002234  001401                          BEQ     .+4
002236  000000                          HLT                     ;ERROR: INCORRECT RESULT

                                ;TEST DESTINATION MODE2
002240  010701                          SCOPE
002242  005246                          CLR     -(6)            ;SET UP STACK WITH 'NEW' STATUS
002244  012746  002256                  MOV     #MODE2,-(6)     ;AND 'NEW' PC
002250  012702  002320                  MOV     #MODE1,%2
002254  000002                          RTI
002256  006722                  MODE2:  SXT     (2)+
002260  005767  000234                  TST     MODE1
002264  001401                          BEQ     .+4
002266  000000                          HLT                     ;ERROR: INCORRECT RESULT
002270  022702  002322                  CMP     #MODE1+2,%2     ;CHECK AUTO-INCREMENT
002274  001401                          BEQ     .+4
002276  000000                          HLT                     ;ERROR: AUTO INCREMENT FAILED

                                ;TEST DESTINATION MODE 3
002100  010701                          SCOPE
002102  012767  002322  000210          MOV     #MODE3,MODE1
002110  012702  002320                  MOV     #MODE1,%2
002114  000270                          SEN
002116  006732                          SXT     @(2)+
002120  022767  177777  000174          CMP     #-1,MODE3
002126  001401                          BEQ     .+4
002130  000000                          HLT                     ;ERROR: INCORRECT RESULT

                                ;TEST DESTINATION MODE 4
002132  010701                          SCOPE
002134  012703  002322                  MOV     #MODE3,%3
002140  012767  177777  000152          MOV     #-1,MODE1
002146  005067  000150                  CLR     MODE3
002152  006743                          SXT     -(3)
002154  005767  000140                  TST     MODE1
002160  001401                          BEQ     .+4
002162  000000                          HLT                     ;ERROR: INCORRECT RESULT
002164  022703  002320                  CMP     #MODE3-2,%3     ;CHECK AUTO-DECREMENT
002170  001401                          BEQ     .+4
002172  000000                          HLT                     ;ERROR: AUTO-DECREMENT FAILED

                                ;TEST DESTINATION MODE 5
002174  010701                          SCOPE
002176  012767  002320  000110          MOV     #MODE1,MODE3
002204  012704  002324                  MOV     #MODE3+2,%4
```



```
002210  000270                          SEN
002212  006754                          SXT     @-(4)
002214  022767  177777  000070          CMP     #-1,MODE1
002222  001401                          BEQ     .+4
002224  000000                          HLT                     ;ERROR: INCORRECT RESULT
002226  022704  002322                  CMP     #MODE3,%4       ;CHECK AUTO-DECREMENT
002232  001401                          BEQ     .+4
002234  000000                          HLT                     ;ERROR: AUTO-DECREMENT DEFERRED FAILED

                                ;TEST DESTINATION MODE 6
002236  010701                          SCOPE
002240  012705  002322                  MOV     #MODE3,%5
002244  006765  177776                  SXT     -2(5)
002250  005767  000044                  TST     MODE1
002254  001401                          BEQ     .+4
002256  000000                          HLT                     ;ERROR: INCORRECT RESULT

                                ;TEST DESTINATION MODE 7
002260  010701                          SCOPE
002262  012767  177777  000030          MOV     #-1,MODE1
002270  012704  177770                  MOV     #-10,%4
002274  012767  002320  000020          MOV     #MODE1,MODE3
002302  006774  002332                  SXT     @MODE3+10(4)
002306  005767  000006                  TST     MODE1
002312  001401                          BEQ     .+4
002314  000000                          HLT                     ;ERROR INCORRECT RESULT
002316  000422                          BR      .+6             ;GO TO NEST TEST
002320  000000                  MODE1:  0
002322  000000                  MODE3:  0

                                ;TEST CONDITION CODES WHEN THE DESTINATION MODE IS NOT 0
                                ;N=0,Z=1,V=0,C=0
002324  010701                          SCOPE
002326  012767  177777  177754          MOV     #-1,MODE1
002334  000257                          CCC
002336  006767  177                   SXT     MODE1
002342  012700  000004                  MOV     PSW,%0
002346  022700  000004                  CMP     #4,%0
002352  001401                          BEQ     .+4
002354  000000                          HLT                     ;ERROR: INCORRECT CONDITION CODES

                                ;N=0,Z=1,V=1,C=1
002356  010701                          SCOPE
002360  012767  177777  177732          MOV     #-1,MODE1
002366  012702  002320                  MOV     #MODE1,%2
002372  012767  000003  175376          MOV     #3,PSW
002400  006712                          SXT     (2)
002402  016700  175370                  MOV     PSW,%0
002406  022700  000005                  CMP     #5,%0
002412  001401                          BEQ     .+4
002414  000000                          HLT                     ;ERROR: INCORRECT CONDITION CODES

                                ;N=1,Z=0,V=0,C=0
002416  010701                          SCOPE
```

```
002420  012702  002320          MOV     #MODE1,X2
002424  005067  177670          CLR     MODE1
002430  012767  000216  175340  MOV     #16,PSW
002436  006722                  SXT     (2)+
002440  016700  175332          MOV     PSW,X0
002444  022700  000010          CMP     #10,X0
002450  001401                  BEQ     .+4
002452  000000                  HLT                     ;ERROR: INCORRECT CONDITION CODES

                                ;N=1,Z=0,V=0,C=1
002454  010701                  SCOPE
002456  012702  002322          MOV     #MODE1+2,X2
002462  005067  177632          CLR     MODE1
002466  012767  000017  175302  MOV     #17,PSW
002474  006742                  SXT     -(2)
002476  016700  175274          MOV     PSW,X0
002502  022700  000011          CMP     #11,X0
002506  001401                  BEQ     .+4
002510  000000                  HLT                     ;ERROR: INCORRECT CONDITION CODES


                                ;TEST THAT SXT EXTENDS 0'S INTO THE PSW
002512  010701                  SCOPE
002514  012767  000357  175254  MOV     #357,PSW
002522  000250                  CLN
002524  006767  175246          SXT     PSW
002530  016700  175242          MOV     PSW,X0          ;GET & TEST PSW CONTENTS
002534  001401                  BEQ     .+4
002536  000000                  HLT

002540  005267  176234          INC     ICNT            ;INCREMENT PASS COUNT
002544  026727  176230  177777  CMP     ICNT,#-1
002552  001402                  BEQ     DONE            ;GO TO DONE IF 128. PASSES COMPLETED
002554  000167  176226          JMP     BEGIN
002560  012767  000007  175078  DONE:   MOV     #7,TPBUF        ;RING BELL
002566  105767  174772          TSTB    TPCSR           ;WAIT FOR THE
002572  100375                  BPL     .+4             ;BELL TO RING
002574  013702  000042          MOV     @#42,X2         ;GET DECTAPE MONITOR ADDRESS
002600  001424                  BEQ     DONE1           ;DO NOT RETURN IF (42)=0
002602  004712                  JSR     7,(2)           ;RETURN TO DECTAPE MONITOR
002604  000240                  NOP                     ;ACT11
002606  000240                  NOP                     ;OVERLAY
002610  000240                  NOP                     ;AREA
002612  000167  176164  DONE1: JMP     START           ;AND REPEAT TEST

        000001                  .END
```



```
BEGIN   002000  DEST    = 001442  DISPLA = 177570  DONE    002560
DONE1   002616  HLT     = 000000  ICNT    001100  IDEST   001444
MODE1   002320  MODE2   002322  MODE7   001700  MODE7   002322
MODE67  001646  PSW     = 177776  SCOPE   = 010701  START   002002
STKPTR  = 000500  SWR     = 177570  SXT07   001640  TEMP    001442
TPBUF   = 177566  TPCSR   = 177564  TZ      001446  UBREAK  = 177770
.       = 002616
```

ERRORS DETECTED: 0

```
    1                                   .TITLE  MAINDEC-11-DCKBB-A PDP11/25-11/45 SOB INST TEST
                                        .NLIST  MC,MD,SEQ
                                        .LIST   ME
                                        .ABS
                                    ;TEST DCKBBA- TEST OF THE SOB INSTRUCTION;
                                    ;       THE SOB INSTRUCTION SUBTRACTS ONE (1) FROM THE REGISTER SPACIFIED
                                    ;AND IF THE RESULT IS NOT ZERO (0) THEN A BRANCH IS TAKEN TO THE ADDRESS
                                    ;(OR NUMBER OF WORDS) SPECIFIED. IF THE RESULT IS 0 THEN THE NEXT SEQUENTIAL
                                    ;INSTRUCTION IS EXECUTED.

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

            000000                          .=0
000000  000002                          .+2
000002  000000                          HALT
000004  000006                          .+2
000006  000000                          HALT
000010  000012                          .+2
000012  000000                          HALT
000014  000016                          .+2
000016  000000                          HALT
000020  000022                          .+2
000022  000000                          HALT
000024  000026                          .+2
000026  000000                          HALT
000030  000032                          .+2
000032  000000                          HALT
000034  000036                          .+2
000036  000000                          HALT
000040  000042                          .+2
000042  000000                          HALT
000044  000046                          .+2
000046  000000                          HALT
000050  000052                          .+2
000052  000000                          HALT
000054  000056                          .+2
000056  000000                          HALT
000060  000062                          .+2
000062  000000                          HALT
000064  000066                          .+2
000066  000000                          HALT
000070  000072                          .+2
000072  000000                          HALT
000074  000076                          .+2
000076  000000                          HALT
000100  000102                          .+2
000102  000000                          HALT
000104  000106                          .+2
000106  000000                          HALT
000110  000112                          .+2
```



```
000112  000000                          HALT
000114  000116                          .+2
000116  000000                          HALT
000120  000122                          .+2
000122  000000                          HALT
000124  000126                          .+2
000126  000000                          HALT
000130  000132                          .+2
000132  000000                          HALT
000134  000136                          .+2
000136  000000                          HALT
000140  000142                          .+2
000142  000000                          HALT
000144  000146                          .+2
000146  000000                          HALT
000150  000152                          .+2
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



        010701                  SCOPE=010701            ;MOV  PC,R1
        000000                  HLT=HALT
        177776                  PSW=177776              ;ADDRESS OF PROCESSER STATUS WORD
        177770                  UBMAP=177770            ;ADDRESS OF MICRO BREAK REGISTER
        177564                  TPCSR=177564            ;ADDRESS OF TELEPRINTER CSR
        177566                  TPBUF=177566            ;ADDRESS OF TELEPRINTER BUFFER
        177570                  SWR=177570              ;ADDRESS OF CONSOLE SWITCH REGISTER
        177570                  DISPLAY=177570          ;ADDRESS OF CONSOLE DISPLAY REGISTER
                        ;**********************INITIAL STACK POINTER**********************
        000500                  STKPTR=500              ;INITIAL STACK SETTING
        000000                  .=0
        000200                  .=200
000200  000167  000376          JMP     START
```



```
        001000                  .=1000
001000  000000          ICNT1   0                       ;CONTAINS PASS COUNT
001002  005067  177772  START1  CLR     ICNT            ;INITIALIZE PASS COUNT
001006  012706  000500  INITIN  MOV     #STKPTR,SP      ;SET STACK POINTER
001012  016737  177762  177570  MOV     ICNT,@#DISPLAY  ;DISPLAY PASS COUNT
001020  032737  000400  177570  BIT     #400,@#SWR      ;LOAD MICRO BREAK REGISTER?
001026  001403                  BEQ     .+10
001030  113737  177570  177770  MOVB    @#SWR,@#UBMAP   ;LOAD MICRO BREAK REG WITH SP=7

                        ;TEST THAT SOB DOES NOT  BRANCH WHEN THE SPECIFIED REGISTER (R0) DECREMENTS TO 0,
001034  010701          T2A:    SCOPE
001040  012700  000001          MOV     #1,R0           ;LOAD R0=1
001044  000401                  BR      T0B             ;GO DO SOB INSTRUCTION
001046  000402                  BR      T0C
001050  077002          T0B1:   SOB     R0,.-2          ;SOB SHOULD NOT BRANCH
001052  000401                  BR      T1A             ;GO TO NEXT TEST
001054  000000          T0C:    HLT                     ;ERROR SOB BRANCHED

                        ;TEST THAT SOB BRANCHES WHEN R0 BECOMES NEGATIVE,
001056  010701          T1A:    SCOPE
001060  005000                  CLR     R0
001062  000405                  BR      T1B             ;GO DO SOB INSTRUCTION
001064  020027  177777  T1AA:   CMP     R0,#-1          ;WAS R0 DECREMENTED?
001070  001404                  BEQ     T2A
001072  000000                  HLT                     ;ERROR! SOB FAILED TO DECREMENT R0
001074  000402                  BR      T2A
001076  077000          T1B:    SOB     R0,T1AA         ;SOB SHOULD BRANCH
001100  000000          T1C:    HLT                     ;ERROR SOB FAILED TO BRANCH

                        ;TEST THAT SOB BRANCHES WHEN R0 DOES NOT DECREMENT TO 0
001102  010701          T2A:    SCOPE
001104  012700  000002          MOV     #2,R0           ;R0=2
001110  000401                  BR      T2B             ;GO DO SOB INSTRUCTION
001112  000402                  BR      T3A             ;GO TO NEXT TEST
001114  077002          T2B:    SOB     R0,.-2          ;SOB SHOULD BRANCH
001116  000000          T2C:    HLT                     ;ERROR SOB FAILED TO BRANCH

                        ;TEST THAT SOB DECREMENTS R0 TO 0 PROPERLY,
001120  010701          T3A:    SCOPE
001122  012700  000001          MOV     #1,R0           ;LOAD R0 =1
001126  077001          T3B1:   SOB     R0,T3A          ;SOB SHOULD NOT BRANCH
001130  020027  000000          CMP     R0,#0           ;DID SOB DECREMENT R0 TO 0?
001134  001421                  BEQ     T4A
001136  000000                  HLT                     ;SOB DID NOT DECREMENT R0 TO = 0

                        ;TEST THAT SOB BRANCHES WHEN R0 IS DECREMENTED AND BECOMES POSITIVE,
001140  010701          T4A:    SCOPE
001142  012700  100000          MOV     #100000,R0      ;LOAD R0 SUCH THAT WHEN DECREMENTED
                                                        ;IT BECOMES POSITIVE (077777)
001146  000401                  BR      T4B             ;GO DO SOB INSTRUCTION
001150  000402                  BR      T5A             ;GO TO NEXT TEST
001152  077002          T4B1:   SOB     R0,.-2          ;SOB SHOULD BRANCH
001154  000000          T4C:    HLT                     ;SOB FAILED TO BRANCH
```

```
                                ;TEST THAT SOB DECREMENTS R0 PROPERLY WHEN R0=100000
001156  010701                  T5A:    SCOPE
001160  012700  100000                  MOV     #100000,%0      ;LOAD R0=170000
001164  000405                          BR      T5B             ;GO DO SOB INSTRUCTION
001166  020027  077777          T5AA:   CMP     %0,#77777       ;DID SOB DECREMENT R0?
001172  001404                          BEQ     T6A             ;GO TO NEXT TEST
001174  000000                          HLT                     ;ERROR SOB DID NOT DECREMENT R0
001176  000402                          BR      T6A             ;GO TO NEXT TEST
001200  077006                  T5B:    SOB     %0,T5AA         ;SOB SHOULD BRANCH
001202  000000                          HLT                     ;SOB FAILED TO BRANCH

                                ;TEST THAT SOB INSTRUCTION DECREMENTS R0 PROPERLY
001204  010701                  T6A:    SCOPE
001206  012700  077777                  MOV     #77777,%0       ;LOAD R0 & R2 WITH THE
001212  010002                          MOV     %0,%2           ;LARGEST POSITIVE NUMBER
001214  000405                          BR      T6C             ;GO DO SOB INSTRUCTION
001216  005302                  T6B:    DEC     %2              ;DECREMENT R2
001220  020002                          CMP     %0,%2           ;R0=R2?
001222  001402                          BEQ     T6C             ;GO TO SOB INSTRUCTION
001224  000000                          HLT                     ;ERROR SOB DID NOT DECREMENT PROPERLY
001226  000401                          BR      T7A             ;GO TO NEXT TEST
001230  077006                  T6C:    SOB     %0,T6B          ;SOB

                                ;TEST THAT SOB DOES NOT BRANCH WHEN R0 DECREMENTS TO 0 (R0 WILL
                                ;INITIALLY BE LOADED TO THE LARGEST POSITIVE NUMBER).
001232  010701                  T7A:    SCOPE
001234  012700  077777                  MOV     #77777,%0       ;LOAD R0 & R2 WITH THE
001240  010002                          MOV     %0,%2           ;LARGEST POSITIVE NUMBER
001242  005302                          DEC     %2
001244  077002                  T7B:    SOB     %0,.-2
001246  060002                          ADD     %0,%2           ;R0=R2=0?
001250  001401                          BEQ     .+4
001252  000000                          HLT                     ;ERROR

        000010                  C=10
        000000                  N=0
                                ;TEST THAT SOB DOES NOT AFFECT THE CONDITION CODES WHEN SOB DOES NOT BRANCH

001254  010701                  T10A:   SCOPE
001256  012702  002001                  MOV     #1,%2
001262  012767  003000  176506          MOV     #0,PSW          ;LOAD PSW
001270  000402                          BR      T10B            ;GO DO SOB INSTRUCTION
001272  000000                  T10AA:  HLT                     ;ERROR SOB BRANCHED
001274  000407                          BR      T10D
001276  077203                  T10B:   SOB     %2,T10AA        ;SOB SHOULD NOT BRANCH
001300  016700  176472                  MOV     PSW,%0          ;GET STATUS CONTENTS
001304  022700  003000                  CMP     #0,%0           ;STATUS = 0
001310  001401                          BEQ     .+4
001312  000000                          HLT                     ;ERROR STATUS WORD CHANGED
001314  000240                  T10D:   NOP
        000011                          C=C+1
        000001                          N=N+1
```



```
001316  010701                  T11A:   SCOPE
001320  012702  002001                  MOV     #1,%2
001324  012767  003001  176444          MOV     #1,PSW          ;LOAD PSW
001332  000402                          BR      T11B            ;GO DO SOB INSTRUCTION
001334  000000                  T11AA:  HLT                     ;ERROR SOB BRANCHED
001336  000407                          BR      T11D
001340  077203                  T11B:   SOB     %2,T11AA        ;SOB SHOULD NOT BRANCH
001342  016700  176430                  MOV     PSW,%0          ;GET STATUS CONTENTS
001346  022700  003001                  CMP     #1,%0           ;STATUS = 1
001352  001401                          BEQ     .+4
001354  000000                          HLT                     ;ERROR STATUS WORD CHANGED
001356  000240                  T11D:   NOP
        000012                          C=C+1
        000002                          N=N+1

001360  010701                  T12A:   SCOPE
001362  012702  002001                  MOV     #1,%2
001366  012767  003002  176402          MOV     #2,PSW          ;LOAD PSW
001374  000402                          BR      T12B            ;GO DO SOB INSTRUCTION
001376  000000                  T12AA:  HLT                     ;ERROR SOB BRANCHED
001400  000407                          BR      T12D
001402  077203                  T12B:   SOB     %2,T12AA        ;SOB SHOULD NOT BRANCH
001404  016700  176366                  MOV     PSW,%0          ;GET STATUS CONTENTS
001410  022700  003002                  CMP     #2,%0           ;STATUS = 2
001414  001401                          BEQ     .+4
001416  000000                          HLT                     ;ERROR STATUS WORD CHANGED
001420  000240                  T12D:   NOP
        000013                          C=C+1
        000003                          N=N+1

001422  010701                  T13A:   SCOPE
001424  012702  002001                  MOV     #1,%2
001430  012767  003003  176340          MOV     #3,PSW          ;LOAD PSW
001436  000402                          BR      T13B            ;GO DO SOB INSTRUCTION
001440  000000                  T13AA:  HLT                     ;ERROR SOB BRANCHED
001442  000407                          BR      T13D
001444  077203                  T13B:   SOB     %2,T13AA        ;SOB SHOULD NOT BRANCH
001446  016700  176324                  MOV     PSW,%0          ;GET STATUS CONTENTS
001452  022700  003003                  CMP     #3,%0           ;STATUS = 3
001456  001401                          BEQ     .+4
001460  000000                          HLT                     ;ERROR STATUS WORD CHANGED
001462  000240                  T13D:   NOP
        000014                          C=C+1
        000004                          N=N+1

001464  010701                  T14A:   SCOPE
001466  012702  002001                  MOV     #1,%2
001472  012767  003004  176276          MOV     #4,PSW          ;LOAD PSW
001500  000402                          BR      T14B            ;GO DO SOB INSTRUCTION
001502  000000                  T14AA:  HLT                     ;ERROR SOB BRANCHED
001504  000407                          BR      T14D
001506  077203                  T14B:   SOB     %2,T14AA        ;SOB SHOULD NOT BRANCH
001510  016700  176262                  MOV     PSW,%0          ;GET STATUS CONTENTS
001514  022700  003004                  CMP     #4,%0           ;STATUS = 4
```

```
001520  001401                          BEQ     .+4
001522  000000                          HLT                     ;ERROR STATUS WORD CHANGED
001524  000240                  T14D:   NOP
        000015                          C=C+1
        000005                          N=N+1

001526  010701                  T15A:   SCOPE
001530  012702  000001                  MOV     #1,%2
001534  012767  000005  176234          MOV     #5,PSW          ;LOAD PSW
001542  000402                          BR      T15B            ;GO DO SOB INSTRUCTION
001544  000000                  T15AA:  HLT                     ;ERROR SOB BRANCHED
001546  000407                          BR      T15D
001550  077203                  T15B:   SOB     %2,T15AA        ;SOB SHOULD NOT BRANCH
001552  016700  176220                  MOV     PSW,%0          ;GET STATUS CONTENTS
001556  022700  000005                  CMP     #5,%0           ;STATUS = 5
001562  001401                          BEQ     .+4
001564  000000                          HLT                     ;ERROR STATUS WORD CHANGED
001566  000240                  T15D:   NOP
        000016                          C=C+1
        000006                          N=N+1

001570  010701                  T16A:   SCOPE
001572  012702  000001                  MOV     #1,%2
001576  012767  000006  176172          MOV     #6,PSW          ;LOAD PSW
001604  000402                          BR      T16B            ;GO DO SOB INSTRUCTION
001606  000000                  T16AA:  HLT                     ;ERROR SOB BRANCHED
001610  000407                          BR      T16D
001612  077203                  T16B:   SOB     %2,T16AA        ;SOB SHOULD NOT BRANCH
001614  016700  176156                  MOV     PSW,%0          ;GET STATUS CONTENTS
001620  022700  000006                  CMP     #6,%0           ;STATUS = 6
001624  001401                          BEQ     .+4
001626  000000                          HLT                     ;ERROR STATUS WORD CHANGED
001630  000240                  T16D:   NOP
        000017                          C=C+1
        000007                          N=N+1

001632  010701                  T17A:   SCOPE
001634  012702  000001                  MOV     #1,%2
001642  012767  000007  176130          MOV     #7,PSW          ;LOAD PSW
001646  000402                          BR      T17B            ;GO DO SOB INSTRUCTION
001650  000000                  T17AA:  HLT                     ;ERROR SOB BRANCHED
001652  000407                          BR      T17D
001654  077203                  T17B:   SOB     %2,T17AA        ;SOB SHOULD NOT BRANCH
001660  016700  176114                  MOV     PSW,%0          ;GET STATUS CONTENTS
001664  022700  000007                  CMP     #7,%0           ;STATUS = 7
001670  001401                          BEQ     .+4
001672  000000                          HLT                     ;ERROR STATUS WORD CHANGED
001674  000240                  T17D:   NOP
        000020                          C=C+1
        000010                          N=N+1

001676  010701                  T20A:   SCOPE
001700  012702  000001                  MOV     #1,%2
001704  012767  000010  176066          MOV     #10,PSW         ;LOAD PSW
```



```
001710  000402                          BR      T20B            ;GO DO SOB INSTRUCTION
001712  000000                  T20AA:  HLT                     ;ERROR SOB BRANCHED
001714  000407                          BR      T20D
001716  077203                  T20B:   SOB     %2,T20AA        ;SOB SHOULD NOT BRANCH
001720  016700  176052                  MOV     PSW,%0          ;GET STATUS CONTENTS
001724  022700  000010                  CMP     #10,%0          ;STATUS = 10
001730  001401                          BEQ     .+4
001732  000000                          HLT                     ;ERROR STATUS WORD CHANGED
001734  000240                  T20D:   NOP
        000021                          C=C+1
        000011                          N=N+1

001736  010701                  T21A:   SCOPE
001740  012702  000001                  MOV     #1,%2
001744  012767  000011  176024          MOV     #11,PSW         ;LOAD PSW
001752  000402                          BR      T21B            ;GO DO SOB INSTRUCTION
001754  000000                  T21AA:  HLT                     ;ERROR SOB BRANCHED
001756  000407                          BR      T21D
001760  077203                  T21B:   SOB     %2,T21AA        ;SOB SHOULD NOT BRANCH
001762  016700  176010                  MOV     PSW,%0          ;GET STATUS CONTENTS
001766  022700  000011                  CMP     #11,%0          ;STATUS = 11
001772  001401                          BEQ     .+4
001774  000000                          HLT                     ;ERROR STATUS WORD CHANGED
001776  000240                  T21D:   NOP
        000022                          C=C+1
        000012                          N=N+1

002000  010701                  T22A:   SCOPE
002002  012702  000001                  MOV     #1,%2
002006  012767  000012  175762          MOV     #12,PSW         ;LOAD PSW
002014  000402                          BR      T22B            ;GO DO SOB INSTRUCTION
002016  000000                  T22AA:  HLT                     ;ERROR SOB BRANCHED
002020  000407                          BR      T22D
002022  077203                  T22B:   SOB     %2,T22AA        ;SOB SHOULD NOT BRANCH
002024  016700  175746                  MOV     PSW,%0          ;GET STATUS CONTENTS
002030  022700  000012                  CMP     #12,%0          ;STATUS = 12
002034  001401                          BEQ     .+4
002036  000000                          HLT                     ;ERROR STATUS WORD CHANGED
002040  000240                  T22D:   NOP
        000023                          C=C+1
        000013                          N=N+1

002042  010701                  T23A:   SCOPE
002044  012702  000001                  MOV     #1,%2
002050  012767  000013  175720          MOV     #13,PSW         ;LOAD PSW
002056  000402                          BR      T23B            ;GO DO SOB INSTRUCTION
002060  000000                  T23AA:  HLT                     ;ERROR SOB BRANCHED
002062  000407                          BR      T23D
002064  077203                  T23B:   SOB     %2,T23AA        ;SOB SHOULD NOT BRANCH
002066  016700  175704                  MOV     PSW,%0          ;GET STATUS CONTENTS
002072  022700  000013                  CMP     #13,%0          ;STATUS = 13
002076  001401                          BEQ     .+4
002100  000000                          HLT                     ;ERROR STATUS WORD CHANGED
002102  000240                  T23D:   NOP
```

```
                000024                          C=C+1
                000014                          N=N+1

002124  010701                  T24A:   SCOPE
002126  012702  000021                  MOV     #1,%2
002132  012767  000014  175656          MOV     #14,PSW         ;LOAD PSW
002140  000402                          BR      T24B            ;GO DO SOB INSTRUCTION
002142  000000                  T24AA:  HLT                     ;ERROR SOB BRANCHED
002144  000407                          BR      T24D
002146  077203                  T24B:   SOB     %2,T24AA        ;SOB SHOULD NOT BRANCH
002150  016700  175642                  MOV     PSW,%0          ;GET STATUS CONTENTS
002154  022700  000014                  CMP     #14,%0          ;STATUS = 14
002160  001401                          BEQ     .+4
002162  000000                          HLT                     ;ERROR STATUS WORD CHANGED
002164  000240                  T24D:   NOP
                000025                  C=C+1
                000015                  N=N+1

002166  010701                  T25A:   SCOPE
002170  012702  000021                  MOV     #1,%2
002174  012767  000015  175614          MOV     #15,PSW         ;LOAD PSW
002202  000402                          BR      T25B            ;GO DO SOB INSTRUCTION
002204  000000                  T25AA:  HLT                     ;ERROR SOB BRANCHED
002206  000407                          BR      T25D
002210  077203                  T25B:   SOB     %2,T25AA        ;SOB SHOULD NOT BRANCH
002212  016700  175600                  MOV     PSW,%0          ;GET STATUS CONTENTS
002216  022700  000015                  CMP     #15,%0          ;STATUS = 15
002222  001401                          BEQ     .+4
002224  000000                          HLT                     ;ERROR STATUS WORD CHANGED
002226  000240                  T25D:   NOP
                000026                  C=C+1
                000016                  N=N+1

002230  010701                  T26A:   SCOPE
002232  012702  000021                  MOV     #1,%2
002236  012767  000016  175552          MOV     #16,PSW         ;LOAD PSW
002244  000402                          BR      T26B            ;GO DO SOB INSTRUCTION
002246  000000                  T26AA:  HLT                     ;ERROR SOB BRANCHED
002250  000407                          BR      T26D
002252  077203                  T26B:   SOB     %2,T26AA        ;SOB SHOULD NOT BRANCH
002254  016700  175536                  MOV     PSW,%0          ;GET STATUS CONTENTS
002260  022700  000016                  CMP     #16,%0          ;STATUS = 16
002264  001401                          BEQ     .+4
002266  000000                          HLT                     ;ERROR STATUS WORD CHANGED
002270  000240                  T26D:   NOP
                000027                  C=C+1
                000017                  N=N+1

002272  010701                  T27A:   SCOPE
002274  012702  000021                  MOV     #1,%2
002300  012767  000017  175510          MOV     #17,PSW         ;LOAD PSW
002306  000402                          BR      T27B            ;GO DO SOB INSTRUCTION
002310  000000                  T27AA:  HLT                     ;ERROR SOB BRANCHED
002312  000407                          BR      T27D
```



```
002314  077203                  T27B:   SOB     %2,T27AA        ;SOB SHOULD NOT BRANCH
002316  016700  175474                  MOV     PSW,%0          ;GET STATUS CONTENTS
002322  022700  000017                  CMP     #17,%0          ;STATUS = 17
002326  001401                          BEQ     .+4
002330  000000                          HLT                     ;ERROR STATUS WORD CHANGED
002332  000240                  T27D:   NOP
                000030                  C=C+1
                000020                  N=N+1

                000000                  N=0
                                ;TEST THAT SOB DOES NOT AFFECT THE CONDITION CODES WHEN SOB BRANCHES,

002334  010701                  T30A:   SCOPE
002336  012702  000002                  MOV     #2,%2           ;SET UP SOB TO BRANCH
002342  012767  000000  175446          MOV     #0,PSW          ;LOAD 0 INTO STATUS WORD
002350  000407                          BR      T30B            ;GO DO SOB INSTRUCTION
002352  016700  175440          T30AA:  MOV     PSW,%0          ;GET STATUS WORD CONTENTS
002356  022700  000000                  CMP     #0,%0           ;STATUS = 0?
002362  001404                          BEQ     T30D
002364  000000                          HLT                     ;ERROR STATUS WORD CHANGED
002366  000402                          BR      T30D
002370  077210                  T30B:   SOB     %2,T30AA        ;SOB SHOULD BRANCH
002372  000000                          HLT                     ;SOB FAILED TO BRANCH
002374  000240                  T30D:   NOP
                000031                  C=C+1
                000001                  N=N+1

002376  010701                  T31A:   SCOPE
002400  012702  000002                  MOV     #2,%2           ;SET UP SOB TO BRANCH
002404  012767  000001  175404          MOV     #1,PSW          ;LOAD 1 INTO STATUS WORD
002412  000407                          BR      T31B            ;GO DO SOB INSTRUCTION
002414  016700  175376          T31AA:  MOV     PSW,%0          ;GET STATUS WORD CONTENTS
002420  022700  000001                  CMP     #1,%0           ;STATUS =
002424  001404                          BEQ     T31D
002426  000000                          HLT                     ;ERROR STAT
002430  000402                          BR      T31D
002432  077210                  T31B:   SOB     %2,T31AA        ;SOB S
002434  000000                          HLT                     ;SOB FAILED TO
002436  000240                  T31D:   NOP
                000032                  C=C+1
                000002                  N=N+1

002440  010701                  T32:    SCOPE
002442  012702  000002                  MOV     #2,%2           ;SET UP SOB TO BRANCH
002446  012767  000002  175342          MOV     #2,PSW          ;LOAD 2 INTO STATUS WORD
002454  000407                          BR      T32B            ;GO DO SOB INSTRUCTION
002456  016700  175334          T32AA:  MOV     PSW,%0          ;GET STATUS WORD CONTENTS
002462  022700  000002                  CMP     #2,%0           ;STATUS = 2?
002466  001404                          BEQ     T32D
002470  000000                          HLT                     ;ERROR STATUS WORD CHANGED
002472  000402                          BR      T32D
002474  077210                  T32B:   SOB     %2,T32AA        ;SOB SHOULD BRANCH
002476  000000                          HLT                     ;SOB FAILED TO BRANCH
002500  000240                  T32D:   NOP
```

//

```
                000033                          C=C+1
                000003                          N=N+1

002462  010701                  T33A1:  $SCOPE
002464  012702  000002                  MOV     #2,X2           ;SET UP SOB TO BRANCH
002470  012767  000003  175100          MOV     #3,PSW          ;LOAD 3 INTO STATUS WORD
002476  000437                          BR      T33B            ;GO DO SOB INSTRUCTION
002500  016700  175272          T33AA1: MOV     PSW,X0          ;GET STATUS WORD CONTENTS
002504  022700  000003                  CMP     #3,X0           ;STATUS = 3?
002510  001404                          BEQ     T33D
002512  000000                          HLT                     ;ERROR STATUS WORD CHANGED
002514  000402                          BR      T33D
002516  077210          T33B1:  SOB     X2,T33AA        ;SOB SHOULD BRANCH
002520  000000                          HLT                     ;SOB FAILED TO BRANCH
002522  000240          T33D1:  NOP
                000034                  C=C+1
                000004                  N=N+1

002524  010701                  T34A1:  $SCOPE
002526  012702  000002                  MOV     #2,X2           ;SET UP SOB TO BRANCH
002532  012767  000004  175236          MOV     #4,PSW          ;LOAD 4 INTO STATUS WORD
002540  000407                          BR      T34B            ;GO DO SOB INSTRUCTION
002542  016700  175230          T34AA1: MOV     PSW,X0          ;GET STATUS WORD CONTENTS
002546  022700  000004                  CMP     #4,X0           ;STATUS = 4?
002552  001404                          BEQ     T34D
002554  000000                          HLT                     ;ERROR STATUS WORD CHANGED
002556  000402                          BR      T34D
002560  077210          T34B1:  SOB     X2,T34AA        ;SOB SHOULD BRANCH
002562  000000                          HLT                     ;SOB FAILED TO BRANCH
002564  000240          T34D1:  NOP
                000035                  C=C+1
                000005                  N=N+1

002566  010701                  T35A1:  $SCOPE
002570  012702  000002                  MOV     #2,X2           ;SET UP SOB TO BRANCH
002574  012767  000005  175174          MOV     #5,PSW          ;LOAD 5 INTO STATUS WORD
002602  000407                          BR      T35B            ;GO DO SOB INSTRUCTION
002604  016700  175166          T35AA1: MOV     PSW,X0          ;GET STATUS WORD CONTENTS
002610  022700  000005                  CMP     #5,X0           ;STATUS = 5?
002614  001404                          BEQ     T35D
002616  000000                          HLT                     ;ERROR STATUS WORD CHANGED
002620  000402                          BR      T35D
002622  077210          T35B1:  SOB     X2,T35AA        ;SOB SHOULD BRANCH
002624  000000                          HLT                     ;SOB FAILED TO BRANCH
002626  000240          T35D1:  NOP
                000036                  C=C+1
                000006                  N=N+1

002630  010701                  T36A1:  $SCOPE
002632  012702  000002                  MOV     #2,X2           ;SET UP SOB TO BRANCH
002636  012767  000006  175132          MOV     #6,PSW          ;LOAD 6 INTO STATUS WORD
002644  000407                          BR      T36B            ;GO DO SOB INSTRUCTION
002646  016700  175124          T36AA1: MOV     PSW,X0          ;GET STATUS WORD CONTENTS
002652  022700  000006                  CMP     #6,X0           ;STATUS = 6?
```



```
002656  001404                          BEQ     T36D
002660  000000                          HLT                     ;ERROR STATUS WORD CHANGED
002662  000402                          BR      T36D
002664  077210          T36B1:  SOB     X2,T36AA        ;SOB SHOULD BRANCH
002666  000000                          HLT                     ;SOB FAILED TO BRANCH
002670  000240          T36D1:  NOP
                000037                  C=C+1
                000007                  N=N+1

002672  010701                  T37A1:  $SCOPE
002674  012702  000002                  MOV     #2,X2           ;SET UP SOB TO BRANCH
002700  012767  000007  175070          MOV     #7,PSW          ;LOAD 7 INTO STATUS WORD
002706  000407                          BR      T37B            ;GO DO SOB INSTRUCTION
002710  016700  175062          T37AA1: MOV     PSW,X0          ;GET STATUS WORD CONTENTS
002714  022700  000007                  CMP     #7,X0           ;STATUS = 7?
002720  001404                          BEQ     T37D
002722  000000                          HLT                     ;ERROR STATUS WORD CHANGED
002724  000402                          BR      T37D
002726  077210          T37B1:  SOB     X2,T37AA        ;SOB SHOULD BRANCH
002730  000000                          HLT                     ;SOB FAILED TO BRANCH
002732  000240          T37D1:  NOP
                000040                  C=C+1
                000010                  N=N+1

002734  010701                  T40A1:  $SCOPE
002736  012702  000002                  MOV     #2,X2           ;SET UP SOB TO BRANCH
002742  012767  000010  175026          MOV     #10,PSW         ;LOAD 10 INTO STATUS WORD
002750  000407                          BR      T40B            ;GO DO SOB INSTRUCTION
002752  016700  175020          T40AA1: MOV     PSW,X0          ;GET STATUS WORD CONTENTS
002756  022700  000010                  CMP     #10,X0          ;STATUS = 10?
002762  001404                          BEQ     T40D
002764  000000                          HLT                     ;ERROR STATUS WORD CHANGED
002766  000402                          BR      T40D
002770  077210          T40B1:  SOB     X2,T40AA        ;SOB SHOULD BRANCH
002772  000000                          HLT                     ;SOB FAILED TO BRANCH
002774  000240          T40D1:  NOP
                000041                  C=C+1
                000011                  N=N+1

002776  010701                  T41A1:  $SCOPE
003000  012702  000002                  MOV     #2,X2           ;SET UP SOB TO BRANCH
003004  012767  000011  174764          MOV     #11,PSW         ;LOAD 11 INTO STATUS WORD
003012  000407                          BR      T41B            ;GO DO SOB INSTRUCTION
003014  016700  174756          T41AA1: MOV     PSW,X0          ;GET STATUS WORD CONTENTS
003020  022700  000011                  CMP     #11,X0          ;STATUS = 11?
003024  001404                          BEQ     T41D
003026  000000                          HLT                     ;ERROR STATUS WORD CHANGED
003030  000402                          BR      T41D
003032  077210          T41B1:  SOB     X2,T41AA        ;SOB SHOULD BRANCH
003034  000000                          HLT                     ;SOB FAILED TO BRANCH
003036  000240          T41D1:  NOP
                000042                  C=C+1
                000012                  N=N+1
```

```
003040  010701                  T42A1   SCOPE
003042  012702  000002                  MOV     #2,%2           ;SET UP SOB TO BRANCH
003046  012767  000012  174722          MOV     #12,PSW         ;LOAD 12 INTO STATUS WORD
003054  000407                          BR      T42B            ;GO DO SOB INSTRUCTION
003056  016700  174714          T42AA1  MOV     PSW,%0          ;GET STATUS WORD CONTENTS
003062  022700  000012                  CMP     #12,%0          ;STATUS = 12?
003066  001404                          BEQ     T42D
003070  000000                          HLT                     ;ERROR STATUS WORD CHANGED
003072  000402                          BR      T42D
003074  077210                  T42B1   SOB     %2,T42AA        ;SOB SHOULD BRANCH
003076  000000                          HLT                     ;SOB FAILED TO BRANCH
003100  000240                  T42D1   NOP
        000043                          C=C+1
        000013                          N=N+1

003102  010701                  T43A1   SCOPE
003104  012702  000002                  MOV     #2,%2           ;SET UP SOB TO BRANCH
003110  012767  000013  174660          MOV     #13,PSW         ;LOAD 13 INTO STATUS WORD
003116  000407                          BR      T43B            ;GO DO SOB INSTRUCTION
003120  016700  174652          T43AA1  MOV     PSW,%0          ;GET STATUS WORD CONTENTS
003124  022700  000013                  CMP     #13,%0          ;STATUS = 13?
003130  001404                          BEQ     T43D
003132  000000                          HLT                     ;ERROR STATUS WORD CHANGED
003134  000402                          BR      T43D
003136  077210                  T43B1   SOB     %2,T43AA        ;SOB SHOULD BRANCH
003140  000000                          HLT                     ;SOB FAILED TO BRANCH
003142  000240                  T43D1   NOP
        000044                          C=C+1
        000014                          N=N+1

003144  010701                  T44A1   SCOPE
003146  012702  000002                  MOV     #2,%2           ;SET UP SOB TO BRANCH
003152  012767  000014  174616          MOV     #14,PSW         ;LOAD 14 INTO STATUS WORD
003160  000407                          BR      T44B            ;GO DO SOB INSTRUCTION
003162  016700  174610          T44AA1  MOV     PSW,%0          ;GET STATUS WORD CONTENTS
003166  022700  000014                  CMP     #14,%0          ;STATUS = 14?
003172  001404                          BEQ     T44D
003174  000000                          HLT                     ;ERROR STATUS WORD CHANGED
003176  000402                          BR      T44D
003200  077210                  T44B1   SOB     %2,T44AA        ;SOB SHOULD BRANCH
003202  000000                          HLT                     ;SOB FAILED TO BRANCH
003204  000240                  T44D1   NOP
        000045                          C=C+1
        000015                          N=N+1

003206  010701                  T45A1   SCOPE
003210  012702  000002                  MOV     #2,%2           ;SET UP SOB TO BRANCH
003214  012767  000015  174554          MOV     #15,PSW         ;LOAD 15 INTO STATUS WORD
003222  000407                          BR      T45B            ;GO DO SOB INSTRUCTION
003224  016700  174546          T45AA1  MOV     PSW,%0          ;GET STATUS WORD CONTENTS
003230  022700  000015                  CMP     #15,%0          ;STATUS = 15?
003234  001404                          BEQ     T45D
003236  000000                          HLT                     ;ERROR STATUS WORD CHANGED
003240  000402                          BR      T45D
```



```
003242  077210                  T45B1   SOB     %2,T45AA        ;SOB SHOULD BRANCH
003244  000000                          HLT                     ;SOB FAILED TO BRANCH
003246  000240                  T45D1   NOP
        000046                          C=C+1
        000016                          N=N+1

003250  010701                  T46A1   SCOPE
003252  012702  000002                  MOV     #2,%2           ;SET UP SOB TO BRANCH
003256  012767  000016  174512          MOV     #16,PSW         ;LOAD 16 INTO STATUS WORD
003264  000407                          BR      T46B            ;GO DO SOB INSTRUCTION
003266  016700  174504          T46AA1  MOV     PSW,%0          ;GET STATUS WORD CONTENTS
003272  022700  000016                  CMP     #16,%0          ;STATUS = 16?
003276  001404                          BEQ     T46D
003300  000000                          HLT                     ;ERROR STATUS WORD CHANGED
003302  000402                          BR      T46D
003304  077210                  T46B1   SOB     %2,T46AA        ;SOB SHOULD BRANCH
003306  000000                          HLT                     ;SOB FAILED TO BRANCH
003310  000240                  T46D1   NOP
        000047                          C=C+1
        000017                          N=N+1

003312  010701                  T47A1   SCOPE
003314  012702  000002                  MOV     #2,%2           ;SET UP SOB TO BRANCH
003320  012767  000017  174450          MOV     #17,PSW         ;LOAD 17 INTO STATUS WORD
003326  000407                          BR      T47B            ;GO DO SOB INSTRUCTION
003330  016700  174442          T47AA1  MOV     PSW,%0          ;GET STATUS WORD CONTENTS
003334  022700  000017                  CMP     #17,%0          ;STATUS = 17?
003340  001404                          BEQ     T47D
003342  000000                          HLT                     ;ERROR STATUS WORD CHANGED
003344  000402                          BR      T47D
003346  077210                  T47B1   SOB     %2,T47AA        ;SOB SHOULD BRANCH
003350  000000                          HLT                     ;SOB FAILED TO BRANCH
003352  000240                  T47D1   NOP
        000050                          C=C+1
        000020                          N=N+1

                                ;TEST THAT SOB CAN BRANCH BACK 64. WORDS
003354  012702  000002                  MOV     #2,%2
003360  000476                          BR      SOB3            ;GO TO SOB INSTRUCTION
003362  000477                  SOBEND1 BR      END             ;WITH HALTS
003364  000000                          0                       ;WITH HALTS
003366  000000                          0                       ;WITH HALTS
003370  000000                          0                       ;WITH HALTS
003372  000000                          0                       ;WITH HALTS
003374  000000                          0                       ;WITH HALTS
003376  000000                          0                       ;WITH HALTS
003400  000000                          0                       ;WITH HALTS
003402  000000                          0                       ;WITH HALTS
003404  000000                          0                       ;WITH HALTS
003406  000000                          0                       ;WITH HALTS
003410  000000                          0                       ;WITH HALTS
003412  000000                          0                       ;WITH HALTS
003414  000000                          0                       ;WITH HALTS
003416  000000                          0                       ;WITH HALTS
```

```
003420  000000                          0                       ;WITH HALTS
003422  000000                          0                       ;WITH HALTS
003424  000000                          0                       ;WITH HALTS
003426  000000                          0                       ;WITH HALTS
003430  000000                          0                       ;WITH HALTS
003432  000000                          0                       ;WITH HALTS
003434  000000                          0                       ;WITH HALTS
003436  000000                          0                       ;WITH HALTS
003440  000000                          0                       ;WITH HALTS
003442  000000                          0                       ;WITH HALTS
003444  000000                          0                       ;WITH HALTS
003446  000000                          0                       ;WITH HALTS
003450  000000                          0                       ;WITH HALTS
003452  000000                          0                       ;WITH HALTS
003454  000000                          0                       ;WITH HALTS
003456  000000                          0                       ;WITH HALTS
003460  000000                          0                       ;WITH HALTS
003462  000000                          0                       ;WITH HALTS
003464  000000                          0                       ;WITH HALTS
003466  000000                          0                       ;WITH HALTS
003470  000000                          0                       ;WITH HALTS
003472  000000                          0                       ;WITH HALTS
003474  000000                          0                       ;WITH HALTS
003476  000000                          0                       ;WITH HALTS
003500  000000                          0                       ;WITH HALTS
003502  000000                          0                       ;WITH HALTS
003504  000000                          0                       ;WITH HALTS
003506  000000                          0                       ;WITH HALTS
003510  000000                          0                       ;WITH HALTS
003512  000000                          0                       ;WITH HALTS
003514  000000                          0                       ;WITH HALTS
003516  000000                          0                       ;WITH HALTS
003520  000000                          0                       ;WITH HALTS
003522  000000                          0                       ;WITH HALTS
003524  000000                          0                       ;WITH HALTS
003526  000000                          0                       ;WITH HALTS
003530  000000                          0                       ;WITH HALTS
003532  000000                          0                       ;WITH HALTS
003534  000000                          0                       ;WITH HALTS
003536  000000                          0                       ;WITH HALTS
003540  000000                          0                       ;WITH HALTS
003542  000000                          0                       ;WITH HALTS
003544  000000                          0                       ;WITH HALTS
003546  000000                          0                       ;WITH HALTS
003550  000000                          0                       ;WITH HALTS
003552  000000                          0                       ;WITH HALTS
003554  000000                          0                       ;WITH HALTS
003556  177777          SOBS1:  SOB     %2,SOBLNO               ;BRANCH BACK 64, WORDS
003560  000000                  HLT                             ;ERROR DID NOT BRANCH

003562  005267  175212          END:    INC     ICNT            ;INCREMENT PASS COUNT
003566  026727  175206  000370          CMP     ICNT,#370
003574  001422                  BEQ     DONE                    ;GO TO DONE IF 1200 PASSES COMPLETED
003576  000167  175204          JMP     BEGIN                   ;RESTART TEST
```



```
001602  012767  000007  173756  DONE1:  MOV     #7,TPBUF        ;RING BELL
001610  105767  173750          TSTB    TPCSR
001614  100375                  BPL     .-4
001616  013702  000042          MOV     @#42,%2                 ;GET DECTAPE MONITOR RETURN ADDRESS
001622  001404                  BEQ     DONE1                   ;DO NOT RETURN IF (42)=0
001624  004712                  JSR     7,(2)                   ;RETURN TO DECTAPE MONITOR
001626  000240                  NOP                             ;ACT11
001630  000240                  NOP                             ;OVERLAY
001632  000240                  NOP                             ;AREA
001634  000167  175142          JMP     START
        000001                  .END
```

```
BEGIN    001005        C        = 000050       DISPLA = 177570        DONE     003602
DONE1    003634        END        003562       HLT    = 000700        ICNT     001700
\      = 000020        PSW      = 177776       SCOPE  = 010701        SDPEND   003362
SOBS     003556        START      001002       STKPTR = 000500        SWR    = 177570
TPBUF  = 177566        TPCSR    = 177564       T0A      001036        T0B      001050
T0C      001054        T1A        001056       T1AA     001064        T1B      001076
T1C      001102        T10A       001254       T10AA    001272        T10B     001276
T10D     001314        T11A       001316       T11AA    001334        T11B     001340
T11D     001356        T12A       001360       T12AA    001376        T12B     001402
T12D     001420        T13A       001422       T13AA    001440        T13B     001444
T13D     001462        T14A       001464       T14AA    001502        T14B     001506
T14D     001524        T15A       001526       T15AA    001544        T15B     001550
T15D     001566        T16A       001570       T16AA    001606        T16B     001612
T16D     001630        T17A       001632       T17AA    001650        T17B     001654
T17D     001672        T2A        001102       T2B      001114        T20A     001674
T20AA    001712        T20B       001716       T20D     001734        T21A     001736
T21AA    001754        T21B       001760       T21D     001776        T22A     002000
T22AA    002016        T22B       002022       T22D     002040        T23A     002042
T23AA    002060        T23B       002064       T23D     002102        T24A     002104
T24AA    002122        T24B       002126       T24D     002144        T25A     002146
T25AA    002164        T25B       002170       T25D     002206        T26A     002210
T26AA    002226        T26B       002232       T26D     002250        T27A     002252
T27AA    002270        T27B       002274       T27D     002312        T3A      001120
T3B      001126        T30A       002314       T30AA    002332        T30B     002350
T30D     002354        T31A       002356       T31AA    002374        T31B     002412
T31D     002416        T32A       002420       T32AA    002436        T32B     002454
T32D     002460        T33A       002462       T33AA    002500        T33B     002516
T33D     002522        T34A       002524       T34AA    002542        T34B     002560
T34D     002564        T35A       002566       T35AA    002604        T35B     002622
T35D     002626        T36A       002630       T36AA    002646        T36B     002664
T36D     002670        T37A       002672       T37AA    002710        T37B     002726
T37D     002732        T4A        001140       T4B      001152        T40A     002734
T40AA    002752        T40B       002770       T40D     002774        T41A     002776
T41AA    003014        T41B       003032       T41D     003036        T42A     003040
T42AA    003056        T42B       003074       T42D     003100        T43A     003102
T43AA    003120        T43B       003136       T43D     003142        T44A     003144
T44AA    003162        T44B       003200       T44D     003204        T45A     003206
T45AA    003224        T45B       003242       T45D     003246        T46A     003250
T46AA    003266        T46B       003304       T46D     003310        T47A     003312
T47AA    003330        T47B       003346       T47D     003352        T5A      001156
T5AA     001166        T5B        001200       T6A      001204        T6B      001216
T6C      001230        T7A        001232       T7B      001244        UBREAK = 177770
.      = 003640
```

ERRORS DETECTED: 0


```
      1                                        .TITLE  MAINDEC-11-DCKBC-A PDP11/25-11/45 XOR INST TEST
                                               .NLIST  MD,MC,CND,SEQ
                                               .LIST   ME
                                               .ABS

                                       ;TEST DCKBCA- TEST OF THE XOR INSTRUCTION AND INCLUDES DATA CHECKS AND
                                       ;CONDITION CODES CHECKS
                                       ;   THE XOR INSTRUCTION IS ESSENTIALLY A 'HALF ADD' INSTRUCTION,IE, IF
                                       ;THE CORRESPONDING BIT IN THE SOURCE AND DESTINATION ADDRESS ARE THE
                                       ;SAME THE RESULT IN DESTINATION ADDRESS IS A '0' IN THAT BIT POSITION
                                       ;AND, IF THE CORRESPONDING BITS ARE DIFFERENT THEN THE RESULT IS A '1'
                                       ;IN THAT BIT POSITION, THE CONDITION CODES ARE AFFECTED AS FOLLOWS:
                                               ;'Z' IS SET IF THE RESULT IS = TO 0,OTHERWISE IT IS CLEARED
                                               ;'N' IS SET IF THE RESULT IS NEGATIVE,CLEARED IF POSITIVE
                                               ;'C' IS UNCHANGED
                                               ;'V' IS CLEARED

                                       ;FUNCTIONS NOT TESTED BY THIS PROGRAM
                                       ;1: THAT XOR WILL EXPLICITLY CHANGE THE USER/EXEC BIT IN THE STATUS WORD,
                                       ;STARTING PROCEEDURE
                                       ;       LOAD ADDRESS=200
                                       ;       PRESS START
                                       ;       STACK POINTER IS SET AT 500
                                       ;       BELL WILL RING WHEN TEST IS COMPLETE

                   010701                      SCOPE=010701            ;MOV    PC,R1
                   000000                      HLT=HALT
                   177776                      PSW=177776              ;ADDRESS OF PROCESSER STATUS WORD
                   177770                      UBREAK=177770           ;ADDRESS OF MICRO BREAK REGISTER
                   177564                      TPCSR=177564            ;ADDRESS OF TELEPRINTER CSR
                   177566                      TPBUF=177566            ;ADDRESS OF TELEPRINTER BUFFER
                   177570                      SWR=177570              ;ADDRESS OF CONSOLE SWITCH REGISTER
                   177570                      DISPLAY=177570          ;ADDRESS OF CONSOLE LIGHT REGISTER

                                       ;********************INITIAL STACK POINTER=0500************************
                   000500              STKPTR=0500                     ;INITIAL STACK SETTING


                   000000                      .=0
          000000   000002                      .+2
          000002   000000                      HALT
          000004   000006                      .+2
          000006   000000                      HALT
          000010   000012                      .+2
          000012   000000                      HALT
          000014   000016                      .+2
          000016   000000                      HALT
          000020   000022                      .+2
          000022   000000                      HALT
```

```
000024  000026                  ,+2
000026  000000                  HALT
000030  000032                  ,+2
000032  000000                  HALT
000034  000036                  ,+2
000036  000000                  HALT
000040  000042                  ,+2
000042  000000                  HALT
000044  000046                  ,+2
000046  000000                  HALT
000050  000052                  ,+2
000052  000000                  HALT
000054  000056                  ,+2
000056  000000                  HALT
000060  000062                  ,+2
000062  000000                  HALT
000064  000066                  ,+2
000066  000000                  HALT
000070  000072                  ,+2
000072  000000                  HALT
000074  000076                  ,+2
000076  000000                  HALT
000100  000102                  ,+2
000102  000000                  HALT
000104  000106                  ,+2
000106  000000                  HALT
000110  000112                  ,+2
000112  000000                  HALT
000114  000116                  ,+2
000116  000000                  HALT
000120  000122                  ,+2
000122  000000                  HALT
000124  000126                  ,+2
000126  000000                  HALT
000130  000132                  ,+2
000132  000000                  HALT
000134  000136                  ,+2
000136  000000                  HALT
000140  000142                  ,+2
000142  000000                  HALT
000144  000146                  ,+2
000146  000000                  HALT
000150  000152                  ,+2
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

        000200                          ,=200
000200  000167  000576                  JMP     START
        001000                          ,=1000
001000  000000              ICNT1       0                       ;CONTAINS PASS COUNT
001002  005067  177772      START1      CLR     ICNT            ;CLEAR PASS COUNT
001006  012706  003500      BEGIN1      MOV     #STKPTR,%6      ;INITIALIZE STACK POINTER
001012  016737  177762  177570          MOV     ICNT,@#DISPLAY  ;DISPLAY PASS COUNT
001020  032737  000400  177570          BIT     #400,@#SWR      ;LOAD MICRO BREAK REGISTER?
001026  001403                          BEQ     .+10
001030  113737  177570  177770          MOVB    @#SWR,@#UBREAK  ;LOAD MICRO BREAK REG WITH SW0-7
        000072                  SOURCE=:X2
        000000                  DEST=:X0
        000000                  Y=0
001036  012701                          SCOPE
001040  012702  000000                  MOV     #0,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
001044  012700  000000                  MOV     #0,DEST ;MOVE DESTINATION OPERAND TO DESTINATION
001050  000257                          CCC
001052  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
001054  103001                          BCC     .+4
001056  000000                          HLT                     ;ERROR! CARRY CHANGED
001060  102001                          BVC     .+4
001062  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
001064  020027  000000                  CMP     DEST,#0 ;DEST=0?
001070  001401                          BEQ     .+4
001072  000000                          HLT                     ;0 XOR 0 FAILED

        000001                          Y=Y+1
001074  012701                          SCOPE
001076  012702  125252                  MOV     #125252,SOURCE  ;MOVE SOURCE OPERAND TO SOURCE
001102  012700  125252                  MOV     #125252,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
001106  000277                          SCC
001110  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
001112  103401                          BCS     .+4
001114  000000                          HLT                     ;ERROR! CARRY CHANGED
001116  102001                          BVC     .+4
001120  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
001122  020027  000000                  CMP     DEST,#0 ;DEST=0?
001126  001401                          BEQ     .+4
001130  000000                          HLT                     ;125252 XOR 125252 FAILED

        000002                          Y=Y+1
001132  012701                          SCOPE
001134  012702  052525                  MOV     #052525,SOURCE  ;MOVE SOURCE OPERAND TO SOURCE
```



```
001140  012700  052525                  MOV     #052525,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
001144  000257                          CCC
001146  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
001150  103001                          BCC     .+4
001152  000000                          HLT                     ;ERROR! CARRY CHANGED
001154  102001                          BVC     .+4
001156  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
001160  020027  000000                  CMP     DEST,#0 ;DEST=0?
001164  001401                          BEQ     .+4
001166  000000                          HLT                     ;052525 XOR 052525 FAILED

        000003                          Y=Y+1
001170  012701                          SCOPE
001172  012702  177777                  MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
001176  012700  177777                  MOV     #-1,DEST        ;MOVE DESTINATION OPERAND TO DESTINATION
001202  000277                          SCC
001204  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
001206  103401                          BCS     .+4
001210  000000                          HLT                     ;ERROR! CARRY CHANGED
001212  102001                          BVC     .+4
001214  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
001216  020027  000000                  CMP     DEST,#0 ;DEST=0?
001222  001401                          BEQ     .+4
001224  000000                          HLT                     ;-1 XOR -1 FAILED

        000004                          Y=Y+1
001226  012701                          SCOPE
001232  012702  125252                  MOV     #125252,SOURCE  ;MOVE SOURCE OPERAND TO SOURCE
001234  012700  052525                  MOV     #052525,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
001240  000257                          CCC
001242  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
001244  103001                          BCC     .+4
001246  000000                          HLT                     ;ERROR! CARRY CHANGED
001250  102001                          BVC     .+4
001252  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
001254  020027  177777                  CMP     DEST,#-1        ;DEST=-1?
001260  001401                          BEQ     .+4
001262  000000                          HLT                     ;125252 XOR 052525 FAILED

        000005                          Y=Y+1
001264  012701                          SCOPE
001266  012702  052525                  MOV     #052525,SOURCE  ;MOVE SOURCE OPERAND TO SOURCE
001272  012700  125252                  MOV     #125252,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
001276  000277                          SCC
001300  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
001302  103401                          BCS     .+4
001304  000000                          HLT                     ;ERROR! CARRY CHANGED
001306  102001                          BVC     .+4
001310  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
001312  020027  177777                  CMP     DEST,#-1        ;DEST=-1?
001316  001401                          BEQ     .+4
001320  000000                          HLT                     ;052525 XOR 125252 FAILED

        000006                          Y=Y+1
```

```
001322  010701                  SCOPE
001324  012702  100000          MOV     #100000,SOURCE  ;MOVE SOURCE OPERAND TO SOURCE
001330  012700  100001          MOV     #100001,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
001334  000257                  CCC
001336  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
001340  103001                  BCC     .+4
001342  000000                  HLT                     ;ERROR! CARRY CHANGED
001344  102001                  BVC     .+4
001346  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
001350  020027  000001          CMP     DEST,#1 ;DEST=1?
001354  001401                  BEQ     .+4
001356  000000                  HLT                     ;100000 XOR 100001 FAILED

        000007                  Y=Y+1
001360  010701                  SCOPE
001362  012702  077777          MOV     #77777,SOURCE   ;MOVE SOURCE OPERAND TO SOURCE
001366  012700  100000          MOV     #100000,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
001372  000277                  SCC
001374  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
001376  103401                  BCS     .+4
001400  000000                  HLT                     ;ERROR! CARRY CHANGED
001402  102001                  BVC     .+4
001404  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
001406  020027  177777          CMP     DEST,#-1        ;DEST=-17
001412  001401                  BEQ     .+4
001414  000000                  HLT                     ;77777 XOR 100000 FAILED

        000010                  Y=Y+1

        000001          N=1
001416  010701                  SCOPE
001420  012702  000001          MOV     #1,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
001424  012700  000001          MOV     #1,DEST ;MOVE DESTINATION OPERAND TO DESTINATION
001430  000257                  CCC
001432  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
001434  103001                  BCC     .+4
001436  000000                  HLT                     ;ERROR! CARRY CHANGED
001440  102001                  BVC     .+4
001442  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
001444  020027  000000          CMP     DEST,#0 ;DEST=0?
001450  001401                  BEQ     .+4
001452  000000                  HLT                     ;1 XOR 1 FAILED

        000011                  Y=Y+1
        000002                  N=N+N
001454  010701                  SCOPE
001456  012702  000002          MOV     #2,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
001462  012700  000002          MOV     #2,DEST ;MOVE DESTINATION OPERAND TO DESTINATION
001466  000277                  SCC
001470  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
001472  103401                  BCS     .+4
001474  000000                  HLT                     ;ERROR! CARRY CHANGED
001476  102001                  BVC     .+4
001500  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
```



```
001502  020027  000000          CMP     DEST,#0 ;DEST=0?
001506  001401                  BEQ     .+4
001510  000000                  HLT                     ;2 XOR 2 FAILED

        000012                  Y=Y+1
        000004                  N=N+N
001512  010701                  SCOPE
001514  012702  000004          MOV     #4,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
001520  012700  000004          MOV     #4,DEST ;MOVE DESTINATION OPERAND TO DESTINATION
001524  000257                  CCC
001526  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
001530  103001                  BCC     .+4
001532  000000                  HLT                     ;ERROR! CARRY CHANGED
001534  102001                  BVC     .+4
001536  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
001540  020027  000000          CMP     DEST,#0 ;DEST=0?
001544  001401                  BEQ     .+4
001546  000000                  HLT                     ;4 XOR 4 FAILED

        000013                  Y=Y+1
        000010                  N=N+N
001550  010701                  SCOPE
001552  012702  000010          MOV     #10,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
001556  012700  000010          MOV     #10,DEST        ;MOVE DESTINATION OPERAND TO DESTINATION
001562  000277                  SCC
001564  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
001566  103401                  BCS     .+4
001570  000000                  HLT                     ;ERROR! CARRY CHANGED
001572  102001                  BVC     .+4
001574  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
001576  020027  000000          CMP     DEST,#0 ;DEST=0?
001602  001401                  BEQ     .+4
001604  000000                  HLT                     ;10 XOR 10 FAILED

        000014                  Y=Y+1
        000020                  N=N+N
001606  010701                  SCOPE
001610  012702  000020          MOV     #20,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
001614  012700  000020          MOV     #20,DEST        ;MOVE DESTINATION OPERAND TO DESTINATION
001620  000257                  CCC
001622  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
001624  103001                  BCC     .+4
001626  000000                  HLT                     ;ERROR! CARRY CHANGED
001630  102001                  BVC     .+4
001632  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
001634  020027  000020          CMP     DEST,#0 ;DEST=0?
001640  001401                  BEQ     .+4
001642  000000                  HLT                     ;20 XOR 20 FAILED

        000015                  Y=Y+1
        000040                  N=N+N
001644  010701                  SCOPE
001646  012702  000040          MOV     #40,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
001652  012700  000040          MOV     #40,DEST        ;MOVE DESTINATION OPERAND TO DESTINATION
```

```
001656  000277                  SCC
001660  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
001662  103401                  BCS     .+4
001664  000000                  HLT                     ;ERROR! CARRY CHANGED
001666  102001                  BVC     .+4
001670  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
001672  020027  000000          CMP     DEST,#0 ;DEST=0?
001676  001401                  BEQ     .+4
001700  000000                  HLT                     ;40 XOR 40 FAILED

        000016                  Y=Y+1
        000100                  N=N+N
001702  010701                  SCOPE
001704  012702  000100          MOV     #100,SOURCE     ;MOVE SOURCE OPERAND TO SOURCE
001710  012700  000100          MOV     #100,DEST       ;MOVE DESTINATION OPERAND TO DESTINATION
001714  000257                  CCC
001716  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
001720  103001                  BCC     .+4
001722  000020                  HLT                     ;ERROR! CARRY CHANGED
001724  102001                  BVC     .+4
001726  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
001730  020027  000000          CMP     DEST,#0 ;DEST=0?
001734  001401                  BEQ     .+4
001736  000000                  HLT                     ;100 XOR 100 FAILED

        000017                  Y=Y+1
        000200                  N=N+N
001740  010701                  SCOPE
001742  012702  000200          MOV     #200,SOURCE     ;MOVE SOURCE OPERAND TO SOURCE
001746  012700  000200          MOV     #200,DEST       ;MOVE DESTINATION OPERAND TO DESTINATION
001752  000277                  SCC
001754  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
001756  103401                  BCS     .+4
001760  000000                  HLT                     ;ERROR! CARRY CHANGED
001762  102001                  BVC     .+4
001764  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
001766  020027  000000          CMP     DEST,#0 ;DEST=0?
001772  001401                  BEQ     .+4
001774  000000                  HLT                     ;200 XOR 200 FAILED

        000020                  Y=Y+1
        000400                  N=N+N
001776  010701                  SCOPE
002000  012702  000400          MOV     #400,SOURCE     ;MOVE SOURCE OPERAND TO SOURCE
002004  012700  000400          MOV     #400,DEST       ;MOVE DESTINATION OPERAND TO DESTINATION
002010  000257                  CCC
002012  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002014  103001                  BCC     .+4
002016  000000                  HLT                     ;ERROR! CARRY CHANGED
002020  102001                  BVC     .+4
002022  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
002024  020027  000000          CMP     DEST,#0 ;DEST=0?
002032  001401                  BEQ     .+4
002032  000000                  HLT                     ;400 XOR 400 FAILED
```



```
        000021                  Y=Y+1
        001000                  N=N+N
002034  010701                  SCOPE
002036  012702  001000          MOV     #1000,SOURCE    ;MOVE SOURCE OPERAND TO SOURCE
002042  012700  001000          MOV     #1000,DEST      ;MOVE DESTINATION OPERAND TO DESTINATION
002046  000277                  SCC
002050  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002052  103401                  BCS     .+4
002054  000000                  HLT                     ;ERROR! CARRY CHANGED
002056  102001                  BVC     .+4
002060  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
002062  020027  000000          CMP     DEST,#0 ;DEST=0?
002066  001401                  BEQ     .+4
002070  000000                  HLT                     ;1000 XOR 1000 FAILED

        000022                  Y=Y+1
        002000                  N=N+N
002072  010701                  SCOPE
002074  012702  002000          MOV     #2000,SOURCE    ;MOVE SOURCE OPERAND TO SOURCE
002100  012700  002000          MOV     #2000,DEST      ;MOVE DESTINATION OPERAND TO DESTINATION
002104  000257                  CCC
002106  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002110  103001                  BCC     .+4
002112  000000                  HLT                     ;ERROR! CARRY CHANGED
002114  102001                  BVC     .+4
002116  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
002120  020027  000000          CMP     DEST,#0 ;DEST=0?
002124  001401                  BEQ     .+4
002126  000000                  HLT                     ;2000 XOR 2000 FAILED

        000023                  Y=Y+1
        004000                  N=N+N
002130  010701                  SCOPE
002132  012702  004000          MOV     #4000,SOURCE    ;MOVE SOURCE OPERAND TO SOURCE
002136  012700  004000          MOV     #4000,DEST      ;MOVE DESTINATION OPERAND TO DESTINATION
002142  000277                  SCC
002144  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002146  103401                  BCS     .+4
002150  000000                  HLT                     ;ERROR! CARRY CHANGED
002152  102001                  BVC     .+4
002154  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
002156  020027  000000          CMP     DEST,#0 ;DEST=0?
002162  001401                  BEQ     .+4
002164  000000                  HLT                     ;4000 XOR 4000 FAILED

        000024                  Y=Y+1
        010000                  N=N+N
002166  010701                  SCOPE
002170  012702  010000          MOV     #10000,SOURCE   ;MOVE SOURCE OPERAND TO SOURCE
002174  012700  010000          MOV     #10000,DEST     ;MOVE DESTINATION OPERAND TO DESTINATION
002200  000257                  CCC
002202  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002204  103001                  BCC     .+4
```

```
002236  000000                          HLT                     ;ERROR! CARRY CHANGED
002240  102001                          BVC     .+4
002242  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
002244  020027  000000                  CMP     DEST,#0 ;DEST=0?
002250  001401                          BEQ     .+4
002252  000000                          HLT                     ;20000 XOR 20000 FAILED

        000025                          Y=Y+1
        020000                          N=N+N
002254  012701                          SCOPE
002256  012702  020000                  MOV     #20000,SOURCE   ;MOVE SOURCE OPERAND TO SOURCE
002262  012700  020000                  MOV     #20000,DEST     ;MOVE DESTINATION OPERAND TO DESTINATION
002266  000277                          SCC
002270  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002272  103401                          BCS     .+4
002274  000000                          HLT                     ;ERROR! CARRY CHANGED
002276  102001                          BVC     .+4
002300  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
002302  020027  000000                  CMP     DEST,#0 ;DEST=0?
002306  001401                          BEQ     .+4
002310  000000                          HLT                     ;20000 XOR 20000 FAILED

        000026                          Y=Y+1
        040000                          N=N+N
002312  012701                          SCOPE
002316  012702  040000                  MOV     #40000,SOURCE   ;MOVE SOURCE OPERAND TO SOURCE
002322  012700  040000                  MOV     #40000,DEST     ;MOVE DESTINATION OPERAND TO DESTINATION
002326  000257                          CCC
002330  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002332  103001                          BCC     .+4
002334  000000                          HLT                     ;ERROR! CARRY CHANGED
002336  102001                          BVC     .+4
002340  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
002342  020027  000000                  CMP     DEST,#0 ;DEST=0?
002346  001401                          BEQ     .+4
002350  000000                          HLT                     ;40000 XOR 40000 FAILED

        000027                          Y=Y+1
        100000                          N=N+N
002352  012701                          SCOPE
002356  012702  100000                  MOV     #100000,SOURCE  ;MOVE SOURCE OPERAND TO SOURCE
002362  012700  100000                  MOV     #100000,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
002366  000277                          SCC
002370  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002372  103401                          BCS     .+4
002374  000000                          HLT                     ;ERROR! CARRY CHANGED
002376  102001                          BVC     .+4
002400  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
002402  020027  177776                  CMP     DEST,#177776    ;DEST=177776?
002406  001401                          BEQ     .+4
002410  000000                          HLT                     ;I=I XOR 1 FAILED

        000031                          Y=Y+1
        000002                          N=N+N
002412  012701                          SCOPE
002416  012702  177777                  MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
002422  012700  000002                  MOV     #2,DEST ;MOVE DESTINATION OPERAND TO DESTINATION
002426  000277                          SCC
002430  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002432  103401                          BCS     .+4
002434  000000                          HLT                     ;ERROR! CARRY CHANGED
002436  102001                          BVC     .+4
002440  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
002442  020027  177775                  CMP     DEST,#177775    ;DEST=177775?
002446  001401                          BEQ     .+4
002450  000000                          HLT                     ;-1 XOR 2 FAILED

        000032                          Y=Y+1
        000004                          N=N+N
002452  012701                          SCOPE
002456  012702  177777                  MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
002462  012700  000004                  MOV     #4,DEST ;MOVE DESTINATION OPERAND TO DESTINATION
002466  000257                          CCC
002470  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002472  103001                          BCC     .+4
002474  000000                          HLT                     ;ERROR! CARRY CHANGED
002476  102001                          BVC     .+4
002500  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
002502  020027  177773                  CMP     DEST,#177773    ;DEST=177773?
002506  001401                          BEQ     .+4
002510  000000                          HLT                     ;-I XOR 4 FAILED

        000033                          Y=Y+1
        000010                          N=N+N
002512  012701                          SCOPE
002516  012702  177777                  MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
002522  012700  000010                  MOV     #10,DEST        ;MOVE DESTINATION OPERAND TO DESTINATION
002526  000277                          SCC
002530  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002532  103401                          BCS     .+4
002534  000000                          HLT                     ;ERROR! CARRY CHANGED
```



```
        000001                  N=1
002536  012701                  SCOPE
002542  012702  177777          MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
002546  012700  000001          MOV     #1,DEST ;MOVE DESTINATION OPERAND TO DESTINATION
002552  000257                  CCC
002554  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002556  103001                  BCC     .+4
002560  000000                  HLT                     ;ERROR! CARRY CHANGED
002562  102001                  BVC     .+4
002564  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
002566  020027  177776          CMP     DEST,#177776    ;DEST=177776?
002572  001401                  BEQ     .+4
002574  000000                  HLT                     ;I=I XOR 1 FAILED

        000031                  Y=Y+1
        000002                  N=N+N
002576  012701                  SCOPE
002602  012702  177777          MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
002606  012700  000002          MOV     #2,DEST ;MOVE DESTINATION OPERAND TO DESTINATION
002612  000277                  SCC
002614  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002616  103401                  BCS     .+4
002620  000000                  HLT                     ;ERROR! CARRY CHANGED
002622  102001                  BVC     .+4
002624  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
002626  020027  177775          CMP     DEST,#177775    ;DEST=177775?
002632  001401                  BEQ     .+4
002634  000000                  HLT                     ;-1 XOR 2 FAILED

        000032                  Y=Y+1
        000004                  N=N+N
002636  012701                  SCOPE
002642  012702  177777          MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
002646  012700  000004          MOV     #4,DEST ;MOVE DESTINATION OPERAND TO DESTINATION
002652  000257                  CCC
002654  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002656  103001                  BCC     .+4
002660  000000                  HLT                     ;ERROR! CARRY CHANGED
002662  102001                  BVC     .+4
002664  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
002666  020027  177773          CMP     DEST,#177773    ;DEST=177773?
002672  001401                  BEQ     .+4
002674  000000                  HLT                     ;-I XOR 4 FAILED

        000033                  Y=Y+1
        000010                  N=N+N
002676  012701                  SCOPE
002702  012702  177777          MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
002706  012700  000010          MOV     #10,DEST        ;MOVE DESTINATION OPERAND TO DESTINATION
002712  000277                  SCC
002714  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002716  103401                  BCS     .+4
002720  000000                  HLT                     ;ERROR! CARRY CHANGED
```

```
002532  102001                    BVC     .+4
002534  000000                    HLT                     ;ERROR: V FAILED TO CLEAR
002536  022027  177767            CMP     DEST,#177767    ;DEST=177767?
002542  001401                    BEQ     .+4
002544  000000                    HLT                     ;-1 XOR 10 FAILED

        000034                    Y=Y+1
        000020                    N=N+N
002546  010701                    SCOPE
002550  012702  177777            MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
002554  012700  000020            MOV     #20,DEST        ;MOVE DESTINATION OPERAND TO DESTINATION
002560  000257                    CCC
002562  074200                    XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002564  103001                    BCC     .+4
002566  000000                    HLT                     ;ERROR: CARRY CHANGED
002570  102001                    BVC     .+4
002572  000000                    HLT                     ;ERROR: V FAILED TO CLEAR
002574  020027  177757            CMP     DEST,#177757    ;DEST=177757?
002600  001401                    BEQ     .+4
002602  000000                    HLT                     ;-1 XOR 20 FAILED

        000035                    Y=Y+1
        000040                    N=N+N
002604  010701                    SCOPE
002606  012702  177777            MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
002612  012700  000040            MOV     #40,DEST        ;MOVE DESTINATION OPERAND TO DESTINATION
002616  000277                    SCC
002620  074200                    XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002622  103401                    BCS     .+4
002624  000000                    HLT                     ;ERROR: CARRY CHANGED
002626  102001                    BVC     .+4
002630  000000                    HLT                     ;ERROR: V FAILED TO CLEAR
002632  020027  177737            CMP     DEST,#177737    ;DEST=177737?
002636  001401                    BEQ     .+4
002640  000000                    HLT                     ;-1 XOR 40 FAILED

        000036                    Y=Y+1
        000100                    N=N+N
002642  010701                    SCOPE
002644  012702  177777            MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
002650  012700  000100            MOV     #100,DEST       ;MOVE DESTINATION OPERAND TO DESTINATION
002654  000257                    CCC
002656  074200                    XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002660  103001                    BCC     .+4
002662  000000                    HLT                     ;ERROR: CARRY CHANGED
002664  102001                    BVC     .+4
002666  000000                    HLT                     ;ERROR: V FAILED TO CLEAR
002670  020027  177677            CMP     DEST,#177677    ;DEST=177677?
002674  001401                    BEQ     .+4
002676  000000                    HLT                     ;-1 XOR 100 FAILED

        000037                    Y=Y+1
        000200                    N=N+N
002700  010701                    SCOPE
```



```
002702  012702  177777            MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
002706  012700  000200            MOV     #200,DEST       ;MOVE DESTINATION OPERAND TO DESTINATION
002712  000277                    SCC
002714  074200                    XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002716  103401                    BCS     .+4
002720  000000                    HLT                     ;ERROR: CARRY CHANGED
002722  102001                    BVC     .+4
002724  000000                    HLT                     ;ERROR: V FAILED TO CLEAR
002726  020027  177577            CMP     DEST,#177577    ;DEST=177577?
002732  001401                    BEQ     .+4
002734  000000                    HLT                     ;-1 XOR 200 FAILED

        000040                    Y=Y+1
        000400                    N=N+N
002736  010701                    SCOPE
002740  012702  177777            MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
002744  012700  000400            MOV     #400,DEST       ;MOVE DESTINATION OPERAND TO DESTINATION
002750  000257                    CCC
002752  074200                    XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
002754  103001                    BCC     .+4
002756  000000                    HLT                     ;ERROR: CARRY CHANGED
002760  102001                    BVC     .+4
002762  000000                    HLT                     ;ERROR: V FAILED TO CLEAR
002764  020027  177377            CMP     DEST,#177377    ;DEST=177377?
002770  001401                    BEQ     .+4
002772  000000                    HLT                     ;-1 XOR 400 FAILED

        000041                    Y=Y+1
        001000                    N=N+N
002774  010701                    SCOPE
002776  012702  177777            MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
003002  012700  001000            MOV     #1000,DEST      ;MOVE DESTINATION OPERAND TO DESTINATION
003006  000277                    SCC
003010  074200                    XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
003012  103401                    BCS     .+4
003014  000000                    HLT                     ;ERROR: CARRY CHANGED
003016  102001                    BVC     .+4
003020  000000                    HLT                     ;ERROR: V FAILED TO CLEAR
003022  020027  176777            CMP     DEST,#176777    ;DEST=176777?
003026  001401                    BEQ     .+4
003030  000000                    HLT                     ;-1 XOR 1000 FAILED

        000042                    Y=Y+1
        002000                    N=N+N
003032  010701                    SCOPE
003034  012702  177777            MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
003040  012700  002000            MOV     #2000,DEST      ;MOVE DESTINATION OPERAND TO DESTINATION
003044  000257                    CCC
003046  074200                    XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
003050  103001                    BCC     .+4
003052  000000                    HLT                     ;ERROR: CARRY CHANGED
003054  102001                    BVC     .+4
003056  000000                    HLT                     ;ERROR: V FAILED TO CLEAR
003060  020027  175777            CMP     DEST,#175777    ;DEST=175777?
```

```
003064  001401                          BEQ     .+4
003066  000000                          HLT                     ;1=1 XOR 2000 FAILED

        000043                          Y=Y+1
        004000                          N=N+N
003070  000241                          SCOPE
003072  012702  177777                  MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
003076  012700  004000                  MOV     #4000,DEST      ;MOVE DESTINATION OPERAND TO DESTINATION
003102  000277                          SCC
003104  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
003106  103401                          BCS     .+4
003110  000000                          HLT                     ;ERROR! CARRY CHANGED
003112  102001                          BVC     .+4
003114  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
003116  020027  173777                  CMP     DEST,#173777    ;DEST=1737777
003122  001401                          BEQ     .+4
003124  000000                          HLT                     ;1=1 XOR 4000 FAILED

        000044                          Y=Y+1
        010000                          N=N+N
003126  010701                          SCOPE
003130  012702  177777                  MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
003134  012700  010000                  MOV     #10000,DEST     ;MOVE DESTINATION OPERAND TO DESTINATION
003140  000257                          CCC
003142  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
003144  103001                          BCC     .+4
003146  000000                          HLT                     ;ERROR! CARRY CHANGED
003150  102001                          BVC     .+4
003152  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
003154  020027  167777                  CMP     DEST,#167777    ;DEST=1677777
003160  001401                          BEQ     .+4
003162  000000                          HLT                     ;1=1 XOR 10000 FAILED

        000045                          Y=Y+1
        020000                          N=N+N
003164  010701                          SCOPE
003166  012702  177777                  MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
003172  012700  020000                  MOV     #20000,DEST     ;MOVE DESTINATION OPERAND TO DESTINATION
003176  000277                          SCC
003200  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
003202  103401                          BCS     .+4
003204  000000                          HLT                     ;ERROR! CARRY CHANGED
003206  102001                          BVC     .+4
003210  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
003212  020027  157777                  CMP     DEST,#157777    ;DEST=1577777
003216  001401                          BEQ     .+4
003220  000000                          HLT                     ;1=1 XOR 20000 FAILED

        000046                          Y=Y+1
        040000                          N=N+N
003222  010701                          SCOPE
003224  012702  177777                  MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
003230  012700  040000                  MOV     #40000,DEST     ;MOVE DESTINATION OPERAND TO DESTINATION
003234  000257                          CCC
```



```
003236  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
003240  103001                          BCC     .+4
003242  000000                          HLT                     ;ERROR! CARRY CHANGED
003244  102001                          BVC     .+4
003246  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
003250  020027  137777                  CMP     DEST,#137777    ;DEST=1377777
003254  001401                          BEQ     .+4
003256  000000                          HLT                     ;1=1 XOR 40000 FAILED

        000047                          Y=Y+1
        100000                          N=N+N
003260  010701                          SCOPE
003262  012702  177777                  MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
003266  012700  100000                  MOV     #100000,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
003272  000277                          SCC
003274  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
003276  103401                          BCS     .+4
003300  000000                          HLT                     ;ERROR! CARRY CHANGED
003302  102001                          BVC     .+4
003304  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
003306  020027  077777                  CMP     DEST,#77777     ;DEST=777777
003312  001401                          BEQ     .+4
003314  000000                          HLT                     ;1=1 XOR 100000 FAILED

        000050                          Y=Y+1
        000000                          N=N+N
```

```
                    000001                  N=1
                                            SCOPE
003316  010701                              MOV     #0,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
003320  012702  000000                      MOV     #1,DEST ;MOVE DESTINATION OPERAND TO DESTINATION
003324  012700  000001                      CCC
003330  000257                              XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
003332  074200                              BCC     .+4
003334  103001                              HLT                     ;ERROR! CARRY CHANGED
003336  000000                              BVC     .+4
003340  102001                              HLT                     ;ERROR! V FAILED TO CLEAR
003342  000000                              CMP     DEST,#1 ;DEST=1?
003344  020027  000001                      BEQ     .+4
003350  001401                              HLT                     ;0 XOR 1 FAILED
003352  000000

                    000051                  Y=Y+1
                    000002                  N=N+N
                                            SCOPE
003354  010701                              MOV     #0,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
003356  012702  000000                      MOV     #2,DEST ;MOVE DESTINATION OPERAND TO DESTINATION
003362  012700  000002                      SCC
003366  000277                              XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
003370  074200                              BCS     .+4
003372  103401                              HLT                     ;ERROR! CARRY CHANGED
003374  000000                              BVC     .+4
003376  102001                              HLT                     ;ERROR! V FAILED TO CLEAR
003400  000000                              CMP     DEST,#2 ;DEST=2?
003402  020027  000002                      BEQ     .+4
003406  001401                              HLT                     ;0 XOR 2 FAILED
003410  000000

                    000052                  Y=Y+1
                    000004                  N=N+N
                                            SCOPE
003412  010701                              MOV     #0,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
003414  012702  000000                      MOV     #4,DEST ;MOVE DESTINATION OPERAND TO DESTINATION
003420  012700  000004                      CCC
003424  000257                              XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
003426  074200                              BCC     .+4
003430  103001                              HLT                     ;ERROR! CARRY CHANGED
003432  000000                              BVC     .+4
003434  102001                              HLT                     ;ERROR! V FAILED TO CLEAR
003436  000000                              CMP     DEST,#4 ;DEST=4?
003440  020027  000004                      BEQ     .+4
003444  001401                              HLT                     ;0 XOR 4 FAILED
003446  000000

                    000053                  Y=Y+1
                    000010                  N=N+N
                                            SCOPE
003450  010701                              MOV     #0,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
003452  012702  000000                      MOV     #10,DEST        ;MOVE DESTINATION OPERAND TO DESTINATION
003456  012700  000010                      SCC
003462  000277                              XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
003464  074200                              BCS     .+4
003466  103401                              HLT                     ;ERROR! CARRY CHANGED
003470  000000
```



```
003472  102001                              BVC     .+4
003474  000000                              HLT                     ;ERROR! V FAILED TO CLEAR
003476  020027  000010                      CMP     DEST,#10        ;DEST=10?
003502  001401                              BEQ     .+4
003504  000000                              HLT                     ;0 XOR 10 FAILED

                    000054                  Y=Y+1
                    000020                  N=N+N
                                            SCOPE
003506  010701                              MOV     #0,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
003510  012702  000000                      MOV     #20,DEST        ;MOVE DESTINATION OPERAND TO DESTINATION
003514  012700  000020                      CCC
003520  000257                              XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
003522  074200                              BCC     .+4
003524  103001                              HLT                     ;ERROR! CARRY CHANGED
003526  000000                              BVC     .+4
003530  102001                              HLT                     ;ERROR! V FAILED TO CLEAR
003532  000000                              CMP     DEST,#20        ;DEST=20?
003534  020027  000020                      BEQ     .+4
003540  001401                              HLT                     ;0 XOR 20 FAILED
003542  000000

                    000055                  Y=Y+1
                    000040                  N=N+N
                                            SCOPE
003544  010701                              MOV     #0,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
003546  012702  000000                      MOV     #40,DEST        ;MOVE DESTINATION OPERAND TO DESTINATION
003552  012700  000040                      SCC
003556  000277                              XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
003560  074200                              BCS     .+4
003562  103401                              HLT                     ;ERROR! CARRY CHANGED
003564  000000                              BVC     .+4
003566  102001                              HLT                     ;ERROR! V FAILED TO CLEAR
003570  000000                              CMP     DEST,#40        ;DEST=40?
003572  020027  000040                      BEQ     .+4
003576  001401                              HLT                     ;0 XOR 40 FAILED
003600  000000

                    000056                  Y=Y+1
                    000100                  N=N+N
                                            SCOPE
003602  010701                              MOV     #0,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
003604  012702  000000                      MOV     #100,DEST       ;MOVE DESTINATION OPERAND TO DESTINATION
003610  012700  000100                      CCC
003614  000257                              XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
003616  074200                              BCC     .+4
003620  103001                              HLT                     ;ERROR! CARRY CHANGED
003622  000000                              BVC     .+4
003624  102001                              HLT                     ;ERROR! V FAILED TO CLEAR
003626  000000                              CMP     DEST,#100       ;DEST=100?
003630  020027  000100                      BEQ     .+4
003634  001401                              HLT                     ;0 XOR 100 FAILED
003636  000000

                    000057                  Y=Y+1
                    000200                  N=N+N
003640  010701                              SCOPE
```

```
003642  012702  000000          MOV     #0,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
003646  012700  002000          MOV     #200,DEST       ;MOVE DESTINATION OPERAND TO DESTINATION
003652  000277                  SCC
003654  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
003656  103401                  BCS     .+4
003660  000000                  HLT                     ;ERROR! CARRY CHANGED
003662  102001                  BVC     .+4
003664  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
003666  020027  002000          CMP     DEST,#200       ;DEST=200?
003672  001401                  BEQ     .+4
003674  000000                  HLT                     ;0 XOR 200 FAILED

        000060                  Y=Y+1
        000400                  N=N+N
003676  010701                  SCOPE
003700  012702  000000          MOV     #0,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
003704  012700  000400          MOV     #400,DEST       ;MOVE DESTINATION OPERAND TO DESTINATION
003710  000257                  CCC
003712  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
003714  103001                  BCC     .+4
003716  000000                  HLT                     ;ERROR! CARRY CHANGED
003720  102001                  BVC     .+4
003722  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
003724  020027  000400          CMP     DEST,#400       ;DEST=400?
003730  001401                  BEQ     .+4
003732  000000                  HLT                     ;0 XOR 400 FAILED

        000061                  Y=Y+1
        001000                  N=N+N
003734  010701                  SCOPE
003736  012702  003000          MOV     #0,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
003742  012700  001000          MOV     #1000,DEST      ;MOVE DESTINATION OPERAND TO DESTINATION
003746  000277                  SCC
003750  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
003752  103401                  BCS     .+4
003754  000000                  HLT                     ;ERROR! CARRY CHANGED
003756  102001                  BVC     .+4
003760  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
003762  020027  001000          CMP     DEST,#1000      ;DEST=1000?
003766  001401                  BEQ     .+4
003770  000000                  HLT                     ;0 XOR 1000 FAILED

        000062                  Y=Y+1
        002000                  N=N+N
003772  010701                  SCOPE
003774  012702  002000          MOV     #0,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
004000  012700  002000          MOV     #2000,DEST      ;MOVE DESTINATION OPERAND TO DESTINATION
004004  000257                  CCC
004006  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
004010  103001                  BCC     .+4
004012  000000                  HLT                     ;ERROR! CARRY CHANGED
004014  102001                  BVC     .+4
004016  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
004020  020027  002000          CMP     DEST,#2000      ;DEST=2000?
```



```
004024  001401                  BEQ     .+4
004026  000000                  HLT                     ;0 XOR 2000 FAILED

        000063                  Y=Y+1
        004000                  N=N+N
004030  010701                  SCOPE
004032  012702  000000          MOV     #0,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
004036  012700  004000          MOV     #4000,DEST      ;MOVE DESTINATION OPERAND TO DESTINATION
004042  000277                  SCC
004044  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
004046  103401                  BCS     .+4
004050  000000                  HLT                     ;ERROR! CARRY CHANGED
004052  102001                  BVC     .+4
004054  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
004056  020027  004000          CMP     DEST,#4000      ;DEST=4000?
004062  001401                  BEQ     .+4
004064  000000                  HLT                     ;0 XOR 4000 FAILED

        000064                  Y=Y+1
        010000                  N=N+N
004066  010701                  SCOPE
004070  012702  000000          MOV     #0,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
004074  012700  010000          MOV     #10000,DEST     ;MOVE DESTINATION OPERAND TO DESTINATION
004100  000257                  CCC
004102  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
004104  103001                  BCC     .+4
004106  000000                  HLT                     ;ERROR! CARRY CHANGED
004110  102001                  BVC     .+4
004112  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
004114  020027  010000          CMP     DEST,#10000     ;DEST=10000?
004120  001401                  BEQ     .+4
004122  000000                  HLT                     ;0 XOR 10000 FAILED

        000065                  Y=Y+1
        020000                  N=N+N
004124  010701                  SCOPE
004126  012702  000000          MOV     #0,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
004132  012700  020000          MOV     #20000,DEST     ;MOVE DESTINATION OPERAND TO DESTINATION
004136  000277                  SCC
004140  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
004142  103401                  BCS     .+4
004144  000000                  HLT                     ;ERROR! CARRY CHANGED
004146  102001                  BVC     .+4
004150  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
004152  020027  020000          CMP     DEST,#20000     ;DEST=20000?
004156  001401                  BEQ     .+4
004160  000000                  HLT                     ;0 XOR 20000 FAILED

        000066                  Y=Y+1
        040000                  N=N+N
004162  010701                  SCOPE
004164  012702  000000          MOV     #0,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
004170  012700  040000          MOV     #40000,DEST     ;MOVE DESTINATION OPERAND TO DESTINATION
004174  000257                  CCC
```

```
004176  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
004200  103001                          BCC     ,+4
004202  000000                          HLT                     ;ERROR! CARRY CHANGED
004204  102001                          BVC     ,+4
004206  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
004210  020027  040000                  CMP     DEST,#40000     ;DEST=40000?
004214  001401                          BEQ     ,+4
004216  000000                          HLT                     ;0 XOR 40000 FAILED

        000067                          Y=Y+1
        100000                          N=N+N
004220  010701                          SCOPE
004222  012702  000000                  MOV     #0,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
004226  012700  100000                  MOV     #100000,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
004232  000277                          SCC
004234  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
004236  103401                          BCS     ,+4
004240  000000                          HLT                     ;ERROR! CARRY CHANGED
004242  102001                          BVC     ,+4
004244  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
004246  020027  100000                  CMP     DEST,#100000    ;DEST=100000?
004252  001401                          BEQ     ,+4
004254  000000                          HLT                     ;0 XOR 100000 FAILED

        000070                          Y=Y+1
        000000                          N=N+N

        000001                  N=1
004256  010701                          SCOPE
004260  012702  000001                  MOV     #1,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
004264  012700  177776                  MOV     #177776,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
004270  000257                          CCC
004272  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
004274  103001                          BCC     ,+4
004276  000000                          HLT                     ;ERROR! CARRY CHANGED
004320  102001                          BVC     ,+4
004322  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
004324  020027  177777                  CMP     DEST,#177777    ;DEST=177777?
004330  001401                          BEQ     ,+4
004332  000000                          HLT                     ;1 XOR 177776 FAILED

        000071                          Y=Y+1
        000002                          N=N+N
004334  010701                          SCOPE
004336  012702  000002                  MOV     #2,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
004342  012700  177775                  MOV     #177775,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
004346  000277                          SCC
004350  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
004352  103401                          BCS     ,+4
004354  000000                          HLT                     ;ERROR! CARRY CHANGED
004356  102001                          BVC     ,+4
004360  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
004362  020027  177777                  CMP     DEST,#177777    ;DEST=177777?
004366  001401                          BEQ     ,+4
```



```
004370  000000                          HLT                     ;2 XOR 177775 FAILED

        000072                          Y=Y+1
        000004                          N=N+N
004372  010701                          SCOPE
004374  012702  000004                  MOV     #4,SOURCE       ;MOVE SOURCE OPERAND TO SOURCE
004400  012700  177773                  MOV     #177773,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
004404  000257                          CCC
004406  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
004410  103001                          BCC     ,+4
004412  000000                          HLT                     ;ERROR! CARRY CHANGED
004414  102001                          BVC     ,+4
004416  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
004420  020027  177777                  CMP     DEST,#177777    ;DEST=177777?
004424  001401                          BEQ     ,+4
004426  000000                          HLT                     ;4 XOR 177773 FAILED

        000073                          Y=Y+1
        000010                          N=N+N
004430  010701                          SCOPE
004432  012702  000010                  MOV     #10,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
004436  012700  177767                  MOV     #177767,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
004442  000277                          SCC
004444  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
004446  103401                          BCS     ,+4
004450  000000                          HLT                     ;ERROR! CARRY CHANGED
004452  102001                          BVC     ,+4
004454  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
004456  020027  177777                  CMP     DEST,#177777    ;DEST=177777?
004462  001401                          BEQ     ,+4
004464  000000                          HLT                     ;10 XOR 177767 FAILED

        000074                          Y=Y+1
        000020                          N=N+N
004466  010701                          SCOPE
004470  012702  000020                  MOV     #20,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
004474  012700  177757                  MOV     #177757,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
004500  000257                          CCC
004502  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
004504  103001                          BCC     ,+4
004506  000000                          HLT                     ;ERROR! CARRY CHANGED
004510  102001                          BVC     ,+4
004512  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
004514  020027  177777                  CMP     DEST,#177777    ;DEST=177777?
004520  001401                          BEQ     ,+4
004522  000000                          HLT                     ;20 XOR 177757 FAILED

        000075                          Y=Y+1
        000040                          N=N+N
004524  010701                          SCOPE
004526  012702  000040                  MOV     #40,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
004532  012700  177737                  MOV     #177737,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
004536  000277                          SCC
004540  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
```

```
004522  103401                          BCS     .+4
004524  000000                          HLT                     ;ERROR! CARRY CHANGED
004526  102001                          BVC     .+4
004530  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
004532  020027  177777                  CMP     DEST,#177777    ;DEST=177777
004536  001401                          BEQ     .+4
004540  000000                          HLT                     ;40 XOR 177737 FAILED

        000076                          Y=Y+1
        000100                          N=N+N
004542  010701                          SCOPE
004544  012702  000100                  MOV     #100,SOURCE     ;MOVE SOURCE OPERAND TO SOURCE
004550  012700  177677                  MOV     #177677,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
004554  000257                          CCC
004556  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
004560  103001                          BCC     .+4
004562  000000                          HLT                     ;ERROR! CARRY CHANGED
004564  102001                          BVC     .+4
004566  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
004570  020027  177777                  CMP     DEST,#177777    ;DEST=177777
004574  001401                          BEQ     .+4
004576  000000                          HLT                     ;100 XOR 177677 FAILED

        000077                          Y=Y+1
        000200                          N=N+N
004600  010701                          SCOPE
004602  012702  000200                  MOV     #200,SOURCE     ;MOVE SOURCE OPERAND TO SOURCE
004606  012700  177577                  MOV     #177577,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
004612  000257                          CCC
004614  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
004616  103401                          BCS     .+4
004620  000000                          HLT                     ;ERROR! CARRY CHANGED
004622  102001                          BVC     .+4
004624  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
004626  020027  177777                  CMP     DEST,#177777    ;DEST=177777
004632  001401                          BEQ     .+4
004634  000000                          HLT                     ;200 XOR 177577 FAILED

        000100                          Y=Y+1
        000400                          N=N+N
004636  010701                          SCOPE
004640  012702  000400                  MOV     #400,SOURCE     ;MOVE SOURCE OPERAND TO SOURCE
004644  012700  177377                  MOV     #177377,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
004650  000257                          CCC
004652  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
004654  103001                          BCC     .+4
004656  000000                          HLT                     ;ERROR! CARRY CHANGED
004660  102001                          BVC     .+4
004662  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
004664  020027  177777                  CMP     DEST,#177777    ;DEST=177777
004670  001401                          BEQ     .+4
004672  000000                          HLT                     ;400 XOR 177377 FAILED

        000101                          Y=Y+1
```



```
        010000                          N=N+N
004674  010701                          SCOPE
004676  012702  001000                  MOV     #1000,SOURCE    ;MOVE SOURCE OPERAND TO SOURCE
004702  012700  176777                  MOV     #176777,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
004706  000257                          CCC
004710  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
004712  103401                          BCS     .+4
004714  000000                          HLT                     ;ERROR! CARRY CHANGED
004716  102001                          BVC     .+4
004720  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
004722  020027  177777                  CMP     DEST,#177777    ;DEST=177777
004726  001401                          BEQ     .+4
004730  000000                          HLT                     ;1000 XOR 176777 FAILED

        001002                          Y=Y+1
        002000                          N=N+N
004732  010701                          SCOPE
004734  012702  002000                  MOV     #2000,SOURCE    ;MOVE SOURCE OPERAND TO SOURCE
004740  012700  175777                  MOV     #175777,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
004744  000257                          CCC
004746  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
004750  103001                          BCC     .+4
004752  000000                          HLT                     ;ERROR! CARRY CHANGED
004754  102001                          BVC     .+4
004756  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
004760  020027  177777                  CMP     DEST,#177777    ;DEST=177777
004764  001401                          BEQ     .+4
004766  000000                          HLT                     ;2000 XOR 175777 FAILED

        001003                          Y=Y+1
        004000                          N=N+N
004770  010701                          SCOPE
004772  012702  004000                  MOV     #4000,SOURCE    ;MOVE SOURCE OPERAND TO SOURCE
004776  012700  173777                  MOV     #173777,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
005002  000257                          CCC
005004  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
005006  103401                          BCS     .+4
005010  000000                          HLT                     ;ERROR! CARRY CHANGED
005012  102001                          BVC     .+4
005014  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
005016  020027  177777                  CMP     DEST,#177777    ;DEST=177777
005022  001401                          BEQ     .+4
005024  000000                          HLT                     ;4000 XOR 173777 FAILED

        001004                          Y=Y+1
        010000                          N=N+N
005026  010701                          SCOPE
005030  012702  010000                  MOV     #10000,SOURCE   ;MOVE SOURCE OPERAND TO SOURCE
005034  012700  167777                  MOV     #167777,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
005040  000257                          CCC
005042  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
005044  103001                          BCC     .+4
005046  000000                          HLT                     ;ERROR! CARRY CHANGED
005050  102001                          BVC     .+4
```

```
005052  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
005054  027027  177777                  CMP     DEST,#177777    ;DEST=177777?
005060  001401                          BEQ     .+4
005062  000000                          HLT                     ;12000 XOR 157777 FAILED

        000105                          Y=Y+1
        020000                          N=N+N
005064  010701                          SCOPE
005066  012702  020000                  MOV     #20000,SOURCE   ;MOVE SOURCE OPERAND TO SOURCE
005072  012720  157777                  MOV     #157777,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
005076  000277                          SCC
005100  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
005102  103401                          BCS     .+4
005104  000000                          HLT                     ;ERROR! CARRY CHANGED
005106  102001                          BVC     .+4
005110  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
005112  027027  177777                  CMP     DEST,#177777    ;DEST=177777?
005116  001401                          BEQ     .+4
005120  000000                          HLT                     ;20000 XOR 157777 FAILED

        000106                          Y=Y+1
        040000                          N=N+N
005122  010701                          SCOPE
005124  012702  040000                  MOV     #40000,SOURCE   ;MOVE SOURCE OPERAND TO SOURCE
005130  012720  137777                  MOV     #137777,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
005134  000277                          SCC
005136  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
005140  103001                          BCC     .+4
005142  000000                          HLT                     ;ERROR! CARRY CHANGED
005144  102001                          BVC     .+4
005146  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
005150  027027  177777                  CMP     DEST,#177777    ;DEST=177777?
005154  001401                          BEQ     .+4
005156  000000                          HLT                     ;40000 XOR 137777 FAILED

        000107                          Y=Y+1
        100000                          N=N+N
005160  010701                          SCOPE
005162  012702  100000                  MOV     #100000,SOURCE  ;MOVE SOURCE OPERAND TO SOURCE
005166  012720  077777                  MOV     #77777,DEST     ;MOVE DESTINATION OPERAND TO DESTINATION
005172  000277                          SCC
005174  074200                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
005176  103401                          BCS     .+4
005200  000000                          HLT                     ;ERROR! CARRY CHANGED
005202  102001                          BVC     .+4
005204  000000                          HLT                     ;ERROR! V FAILED TO CLEAR
005206  027027  177777                  CMP     DEST,#177777    ;DEST=177777?
005212  001401                          BEQ     .+4
005214  000000                          HLT                     ;100000 XOR 77777 FAILED

        000110                          Y=Y+1
        000000                          N=N+N

        000000                          SOURCE=%0
```



```
                000002                  DEST=%2
                000002                  RSLT=%2

        005216  010701                  SCOPE
        005220  012720  031463          MOV     #031463,SOURCE  ;MOVE SOURCE OPERAND TO SOURCE
        005224  012702  146314          MOV     #146314,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
        005230  000257                  CCC
        005232  074002                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
        005234  103001                  BCC     .+4
        005236  000000                  HLT                     ;ERROR! CARRY CHANGED
        005240  102001                  BVC     .+4
        005242  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
        005244  027027  177777          CMP     DEST,#-1        ;DEST=-1?
        005250  001401                  BEQ     .+4
        005252  000000                  HLT                     ;031463 XOR 146314 FAILED

                000111                  Y=Y+1

                000002                  SOURCE=%2
                000000                  DEST=%0
                000002                  RSLT=%2

        005254  010701                  SCOPE
        005256  012702  146314          MOV     #146314,SOURCE  ;MOVE SOURCE OPERAND TO SOURCE
        005262  012720  031463          MOV     #031463,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
        005266  000277                  SCC
        005270  074200                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
        005272  103401                  BCS     .+4
        005274  000000                  HLT                     ;ERROR! CARRY CHANGED
        005276  102001                  BVC     .+4
        005300  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
        005302  027027  177777          CMP     DEST,#-1        ;DEST=-1?
        005306  001401                  BEQ     .+4
        005310  000000                  HLT                     ;146314 XOR 031463 FAILED

                000112                  Y=Y+1
                000003                  SOURCE=%3

        005312  010701                  SCOPE
        005314  012703  016161          MOV     #016161,SOURCE  ;MOVE SOURCE OPERAND TO SOURCE
        005320  012700  161616          MOV     #161616,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
        005324  000257                  CCC
        005326  074300                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
        005330  103001                  BCC     .+4
        005332  000000                  HLT                     ;ERROR! CARRY CHANGED
        005334  102001                  BVC     .+4
        005336  000000                  HLT                     ;ERROR! V FAILED TO CLEAR
        005340  020027  177777          CMP     DEST,#-1        ;DEST=-1?
        005344  001401                  BEQ     .+4
        005346  000000                  HLT                     ;016161 XOR 161616 FAILED

                000113                  Y=Y+1

                000003                  DEST=%3
```

```
005350  010701                          SCOPE
005352  012703  161616                  MOV     #161616,SOURCE  ;MOVE SOURCE OPERAND TO SOURCE
005356  012705  016161                  MOV     #016161,DEST    ;MOVE DESTINATION OPERAND TO DESTINATION
005362  000277                          SCC
005364  074305                          XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
005366  103401                          BCS     .+4
005370  000000                          HLT                     ;ERROR1 CARRY CHANGED
005372  102001                          BVC     .+4
005374  000000                          HLT                     ;ERROR1 V FAILED TO CLEAR
005376  020527  177777                  CMP     DEST,#-1        ;DEST=-17
005402  001401                          BEQ     .+4
005404  000000                          HLT                     ;161616 XOR 016161 FAILED

        000114                          Y=Y+1
        000000                  DEST=X0

                                ;TEST THAT CONDITION CODES ARE OPERATED ON CORRECTLY
                                ;CARRY REMAINS UNCHANGED (CARRY SET)

005406  010701                          SCOPE
005410  012702  177777                  MOV     #-1,X2
005414  012700  177777                  MOV     #-1,X0
005420  000261                          SEC                     ;SET CARRY
005422  074200                          XOR     X2,X0           ;XOR
005424  103421                          BCS     .+4             ;BRANCH IF CARRY IS SET
005426  000000                          HLT

                                ;TEST /N/
005430  010701                          SCOPE
005432  012700  100000                  MOV     #100000,X0
005436  005002                          CLR     X2
005440  074200                          XOR     X2,X0
005442  100401                          BMI     .+4
005444  000000                          HLT                     ;ERROR1 /N/ FAILED TO SET

005446  010701                  SCOPE
005450  012702  100000                  MOV     #100000,X2
005454  012700  177777                  MOV     #-1,X0
005460  000270                          CLN
005462  074200                          XOR     X2,X0
005464  100001                          BPL     .+4
005466  000000                          HLT                     ;ERROR1 /N/ FAILED TO CLEAR


                                ;CARRY REMAINS UNCHANGED (CARRY CLEAR)
005470  010701                          SCOPE
005472  012702  177777                  MOV     #-1,X2
005476  012700  177777                  MOV     #-1,X0
005502  000241                          CLC                     ;CLEAR CARRY
005504  074200                          XOR     X2,X0           ;XOR
005506  103001                          BCC     .+4             ;BRANCH IF CARRY IS CLEAR
005510  000000                          HLT
        000001                  N=1
```



```
                                ;TEST THAT Z BIT IS CLEAR IF RESULT IS NOT ZERO
005512  010701                          SCOPE
005514  012700  000001                  MOV     #1,DEST
005520  000264                          SEZ                     ;SET Z BIT
005522  074300                          XOR     SOURCE,DEST     ;XOR
005524  001001                          BNE     .+4
005526  000000                          HLT                     ;0 XOR 1 FAILED TO CLEAR Z

        000002                          N=N+N
005530  010701                          SCOPE
005532  012700  000002                  MOV     #2,DEST
005536  000264                          SEZ                     ;SET Z BIT
005540  074300                          XOR     SOURCE,DEST     ;XOR
005542  001001                          BNE     .+4
005544  000000                          HLT                     ;0 XOR 2 FAILED TO CLEAR Z

        000004                          N=N+N
005546  010701                          SCOPE
005550  012700  000004                  MOV     #4,DEST
005554  000264                          SEZ                     ;SET Z BIT
005556  074300                          XOR     SOURCE,DEST     ;XOR
005560  001001                          BNE     .+4
005562  000000                          HLT                     ;0 XOR 4 FAILED TO CLEAR Z

        000010                          N=N+N
005564  010701                          SCOPE
005566  012700  000010                  MOV     #10,DEST
005572  000264                          SEZ                     ;SET Z BIT
005574  074300                          XOR     SOURCE,DEST     ;XOR
005576  001001                          BNE     .+4
005600  000000                          HLT                     ;0 XOR 10 FAILED TO CLEAR Z

        000020                          N=N+N
005602  010701                          SCOPE
005604  012700  000020                  MOV     #20,DEST
005610  000264                          SEZ                     ;SET Z BIT
005612  074300                          XOR     SOURCE,DEST     ;XOR
005614  001001                          BNE     .+4
005616  000000                          HLT                     ;0 XOR 20 FAILED TO CLEAR Z

        000040                          N=N+N
005620  010701                          SCOPE
005622  012700  000040                  MOV     #40,DEST
005626  000264                          SEZ                     ;SET Z BIT
005630  074300                          XOR     SOURCE,DEST     ;XOR
005632  001001                          BNE     .+4
005634  000000                          HLT                     ;0 XOR 40 FAILED TO CLEAR Z

        000100                          N=N+N
005636  010701                          SCOPE
005640  012700  000100                  MOV     #100,DEST
005644  000264                          SEZ                     ;SET Z BIT
005646  074300                          XOR     SOURCE,DEST     ;XOR
005650  001001                          BNE     .+4
```

```
025652  000000                          HLT                     ;0 XOR 100 FAILED TO CLEAR Z

                000200                  N=N+N
                                        SCOPE
025654  010701
025656  012700  000200                  MOV     #200,DEST
025662  000264                          SEZ                     ;SET Z BIT
025664  074300                          XOR     SOURCE,DEST     ;XOR
025666  001001                          BNE     .+4
025670  000000                          HLT                     ;0 XOR 200 FAILED TO CLEAR Z

                000400                  N=N+N
                                        SCOPE
025672  010701
025674  012700  000400                  MOV     #400,DEST
025700  000264                          SEZ                     ;SET Z BIT
025702  074300                          XOR     SOURCE,DEST     ;XOR
025704  001001                          BNE     .+4
025706  000000                          HLT                     ;0 XOR 400 FAILED TO CLEAR Z

                001000                  N=N+N
                                        SCOPE
025710  010701
025712  012700  001000                  MOV     #1000,DEST
025716  000264                          SEZ                     ;SET Z BIT
025720  074300                          XOR     SOURCE,DEST     ;XOR
025722  001001                          BNE     .+4
025724  000000                          HLT                     ;0 XOR 1000 FAILED TO CLEAR Z

                002000                  N=N+N
                                        SCOPE
025726  010701
025730  012700  002000                  MOV     #2000,DEST
025734  000264                          SEZ                     ;SET Z BIT
025736  074300                          XOR     SOURCE,DEST     ;XOR
025740  001001                          BNE     .+4
025742  000000                          HLT                     ;0 XOR 2000 FAILED TO CLEAR Z

                004000                  N=N+N
                                        SCOPE
025744  010701
025746  012700  004000                  MOV     #4000,DEST
025752  000264                          SEZ                     ;SET Z BIT
025754  074300                          XOR     SOURCE,DEST     ;XOR
025756  001001                          BNE     .+4
025760  000000                          HLT                     ;0 XOR 4000 FAILED TO CLEAR Z

                010000                  N=N+N
                                        SCOPE
025762  010701
025764  012700  010000                  MOV     #10000,DEST
025770  000264                          SEZ                     ;SET Z BIT
025772  074300                          XOR     SOURCE,DEST     ;XOR
025774  001001                          BNE     .+4
025776  000000                          HLT                     ;0 XOR 10000 FAILED TO CLEAR Z

                020000                  N=N+N
                                        SCOPE
026000  010701
026002  012700  020000                  MOV     #20000,DEST
026006  000264                          SEZ                     ;SET Z BIT
```



```
026010  074300                          XOR     SOURCE,DEST     ;XOR
026012  001001                          BNE     .+4
026014  000000                          HLT                     ;0 XOR 20000 FAILED TO CLEAR Z

                040000                  N=N+N
                                        SCOPE
026016  010701
026020  012700  040000                  MOV     #40000,DEST
026024  000264                          SEZ                     ;SET Z BIT
026026  074300                          XOR     SOURCE,DEST     ;XOR
026030  001001                          BNE     .+4
026032  000000                          HLT                     ;0 XOR 40000 FAILED TO CLEAR Z

                100000                  N=N+N
                                        SCOPE
026034  010701
026036  012700  100000                  MOV     #100000,DEST
026042  000264                          SEZ                     ;SET Z BIT
026044  074300                          XOR     SOURCE,DEST     ;XOR
026046  001001                          BNE     .+4
026050  000000                          HLT                     ;0 XOR 100000 FAILED TO CLEAR Z

                000000                  N=N+N
                000001          N=1
                                ;Z BIT IS SET IF RESULT=0
                                        SCOPE
026052  010701
026054  012702  125252                  MOV     #125252,%2
026060  012700  125252                  MOV     #125252,%0
026064  000244                          CLZ                     ;CLEAR Z
026066  074200                          XOR     %2,%0           ;XOR
026070  001401                          BEQ     .+4             ;BRANCH IF Z IS SET
026072  000000                          HLT

                                ;V IS CLEARED
                                        SCOPE
026074  010701
026076  012702  125252                  MOV     #125252,%2
026102  012700  025252                  MOV     #025252,%0
026106  000262                          SEV
026110  074200                          XOR     %2,%0   ;XOR
026112  102001                          BVC     .+4             ;BRANCH IF V IS CLEARED
026114  000000                          HLT

026116  010701                          SCOPE
026120  012702  052525                  MOV     #052525,%2
026124  012700  125252                  MOV     #125252,%0
026130  000262                          SEV                     ;SET V
026132  074200                          XOR     %2,%0   ;XOR
026134  102001                          BVC     .+4             ;BRANCH IF V IS CLEARED
026136  000000                          HLT

026140  010701                          SCOPE
026142  005002                          CLR     %2
026144  005000                          CLR     %0
026146  000262                          SEV
026150  074200                          XOR     %2,%0
026152  102001                          BVC     .+4
```

```
006154  ?00000                          HLT                     ;ERROR! V FAILED TO CLEAR
006156  ?10?01                          SCOPE

006160  ?12?02  177777                  MOV     #-1,%2
006164  ?12?00  177777                  MOV     #-1,%0
006170  ?00262                          SEV
006172  ?74200                          XOR     %2,%0
006174  102003                         BVC     .+10
006176  ?00000                          HLT                     ;ERROR! V FAILED TO CLEAR

        000002                          SOURCE=%2
        006200                          DEST=.
        006202                          .=.+2
006202  000000                  ;DEST:  0
                                ;TEST DESTINATION MODE=67
006204  ?10?01                          ?COPE
006210  ?12?02  177777                  MOV     #-1,SOURCE      ;MOVE SOURCE OPERAND TO SOURCE
006214  ?12767  177777  177760          MOV     #-1,DEST        ;MOVE DESTINATION OPERAND TO DESTINATION
006220  ?00257                          CCC
006222  ?74267  177752                  XOR     SOURCE,DEST     ;XOR SOURCE AND DESTINATION
006226  103001                         HCC     .+4
006230  ?00000                          HLT                     ;ERROR! CARRY CHANGED
006232  102001                         BVC     .+4
006234  ?00000                          HLT                     ;ERROR! V FAILED TO CLEAR
006236  ?26727  177736  000000          CMP     DEST,#0 ;DEST=?7
006244  ?01401                          BEQ     .+4
006246  ?00000                          HLT                     ;-1 XOR -1 FAILED

        000115                          Y=Y+1

                                ;TEST DESTINATION MODE=27
006250  ?10?01                          SCOPE
006252  ?12?02  177777                  MOV     #-1,%2
006256  ?12767  177777  000002          MOV     #-1,MODE27
006264  ?74227                          XOR     %2,(?)+
006266  000000                  MODE27: 0
006270  ?01401                          BEQ     .+4
006272  ?00000                          HLT

                                ;TEST DESTINATION MODE 37
006274  ?10?01                          SCOPE
006276  ?12?02  177777                  MOV     #-1,%2
006322  ?12737  177777  006200          MOV     #-1,@#DEST
006310  ?74237  006200                  XOR     %2,@#DEST
006314  ?01401                          BEQ     .+4
006316  ?00000                          HLT

                                ;TEST DESTINATION MODE 77
006320  ?10?01                          SCOPE
006322  ?12767  006202  177650          MOV     #IDEST,DEST
006330  ?12?02  177777                  MOV     #-1,%2
006334  ?12777  177777  177636          MOV     #-1,@DEST
006342  ?74277  177632                  XOR     %2,@DEST
006346  ?01401                          BEQ     .+4
```



```
006350  ?00000                          HLT


                                ;TEST DESTINATION MODE 1
006352  ?10?01                          ?COPE
006354  ?12?00  006750                  MOV     #MODE1,%0
006360  ?12?02  177777                  MOV     #-1,%2
006364  012710  177776                  MOV     #-2,(0)
006370  074210                         XOR     %2,(0)
006372  ?22767  000001  000350          CMP     #1,MODE1        ;CHECK RESULT
006400  ?01401                          BEQ     .+4
006402  ?00000                          HLT                     ;ERROR! INCORRECT RESULT
006404  ?22?00  006750                  CMP     #MODE1,%0       ;CHECK REGISTER
006410  ?01401                          BEQ     .+4
006412  ?00000                          HLT                     ;ERROR! R0 CHANGED
006414  ?22?02  177777                  CMP     #-1,%2          ;CHECK REGISTER
006420  ?01401                          BEQ     .+4
006422  ?00000                          HLT                     ;ERROR! R2 CHANGED

                                ;TEST DESTINATION MODE 2
006424  ?10?01                          SCOPE
006430  ?12?02  006750                  MOV     #MODE1,%2
006434  ?12?03  000002                  MOV     #2,%3
006440  ?05012                          CLR     (2)
006444  ?74122                          XOR     %3,(2)+
006446  ?22767  000002  000300          CMP     #2,MODE1        ;CHECK RESULT
006454  ?01401                          BEQ     .+4
006456  ?00000                          HLT                     ;ERROR! INCORRECT RESULT
006460  ?22?02  006752                  CMP     #MODE1+2,%2     ;TEST AUTO-INCREMENT
006464  ?01401                          BEQ     .+4
006466  ?00000                          HLT                     ;ERROR! AUTO-INCREMENT FAILED

                                ;TEST DESTINATION MODE 3
006470  ?10?01                          SCOPE
006472  ?12767  006752  000254          MOV     #MODE3,MODE1
006500  ?12?04  006750                  MOV     #MODE1,%4
006504  ?12?03  000003                  MOV     #3,%3
006510  005067  ?00242                  CLR     MODE3
006514  074334                         XOR     %3,@(4)+
006516  ?22767  ?00?03  000232          CMP     #3,MODE3
006524  ?01401                          BEQ     .+4
006526  ?00000                          HLT                     ;ERROR! INCORRECT RESULT
006530  ?22?04  006762                  CMP     #MODE1+2,%4     ;TEST AUTO-INCREMENT
006534  ?01401                          BEQ     .+4
006536  ?00000                          HLT                     ;ERROR! AUTO-INCREMENT DEFERRED FAILED

                                ;TEST DESTINATION MODE 4
006540  ?10?01                          SCOPE
006542  ?12726  ?00004                  MOV     #4,(6)+
006546  005005                         CLR     %5
006550  ?74546                          XOR     %5,-(6)
006552  ?22716  ?00004                  CMP     #4,(6)
006556  ?01401                          BEQ     .+4
006560  ?00000                          HLT                     ;ERROR! INCORRECT RESULT
```

```
006556  012706  000500                 MOV     #STKPTH,%6      ;RESTORE STACK PTR

                                ;TEST DESTINATION MODE 5
006562  010701                          SCOPE
006564  012767  006750  000160         MOV     #MODE1,MODE3
006572  012702  006754                 MOV     #MODE3+2,%2
006576  012767  000005  000144         MOV     #5,MODE1
006604  005000                         CLR     %0
006606  074052                         XOR     %0,0-(2)
006610  022767  000005  000132         CMP     #5,MODE1
006616  001401                         BEQ     .+4
006620  000000                         HLT                     ;ERROR! INCORRECT RESULT
006622  022702  006752                 CMP     #MODE3,%2       ;CHECK REGISTER
006626  001401                         BEQ     .+4
006630  000000                         HLT                     ;ERROR! REGISTER CONTENTS CHANGED

                                ;TEST DESTINATION MODE 6
006632  010701                          SCOPE
006634  005067  000110                 CLR     MODE1
006640  012702  000006                 MOV     #6,%2
006644  012703  006752                 MOV     #MODE3,%3
006650  074263  177776                 XOR     %2,-2(3)
006654  022767  000006  000066         CMP     #6,MODE1
006662  001401                         BEQ     .+4
006664  000000                         HLT                     ;ERROR! INCORRECT RESULT
006666  022703  006752                 CMP     #MODE3,%3       ;CHECK REGISTER
006672  001401                         BEQ     .+4
006674  000000                         HLT                     ;ERROR! REGISTER CONTENTS CHANGED


                                ;TEST DESTINATION MODE 7
006676  010701                          SCOPE
006700  012767  006752  000042         MOV     #MODE3,MODE1
006706  012705  177770                 MOV     #-10,%5
006712  012767  177777  000032         MOV     #-1,MODE3
006720  074575  006760                 XOR     %5,@MODE1+10(5)
006724  022767  000007  000020         CMP     #7,MODE3
006732  001401                         BEQ     .+4
006734  000000                         HLT                     ;ERROR! INCORRECT RESULT
006736  022705  177770                 CMP     #-10,%5
006742  001401                         BEQ     .+4
006744  000000                         HLT                     ;ERROR! REGISTER CONTENTS CHANGED
006746  000432                         BR      .+6             ;GO TO NEXT TEST
006750  000000                 MODE1:  0
006752  000000                 MODE3:  0

                                ;TEST CONDITION CODES WHEN DESTINATION MODE IS NOT = 0
                                ;N=0,Z=0,V=0,C=0
006754  010701                          SCOPE
006756  012700  000001                 MOV     #1,%0
006762  012767  000002  177760         MOV     #2,MODE1
006770  012702  006750                 MOV     #MODE1,%2
006774  000277                         SCC
006776  000241                         CLC
```



```
007000  074022                         XOR     %0,(2)+
007002  016700  170770                 MOV     PSW,%0
007006  022700  000000                 CMP     #0,%0
007012  001401                         BEQ     .+4
007014  000000                         HLT

                                ;N=0,Z=0,V=0,C=1
007016  010701                          SCOPE
007020  012700  000002                 MOV     #2,%0
007024  012767  000004  177710         MOV     #4,MODE1
007032  012703  006752                 MOV     #MODE3,%3
007036  000277                         SCC
007040  074043                         XOR     %0,-(3)
007042  016700  170730                 MOV     PSW,%0
007046  022700  000001                 CMP     #1,%0
007052  001401                         BEQ     .+4
007054  000000                         HLT

                                ;N=0,Z=1,V=0,C=0
007056  012701                          SCOPE
007060  012767  177777  177662         MOV     #-1,MODE1
007066  012702  177777                 MOV     #-1,%2
007072  012767  000012  170670         MOV     #12,PSW
007100  074267  177644                 XOR     %2,MODE1
007104  016700  170666                 MOV     PSW,%0
007110  022700  000004                 CMP     #4,%0
007114  001401                         BEQ     .+4
007116  000000                         HLT                     ;ERROR! INCORRECT CONDITION CODES

                                ;N=0,Z=1,V=0,C=1
007120  010701                          SCOPE
007122  012767  177777  177620         MOV     #-1,MODE1
007130  012704  177777                 MOV     #-1,%4
007134  012702  006750                 MOV     #MODE1,%2
007140  000277                         SCC
007142  000244                         CLZ
007144  074412                         XOR     %4,(2)
007146  016700  170624                 MOV     PSW,%0
007152  022700  000005                 CMP     #5,%0
007156  001401                         BEQ     .+4
007160  000000                         HLT                     ;ERROR! INCORRECT CONDITION CODES

                                ;N=1,Z=0,V=0,C=0
007162  010701                          SCOPE
007164  012700  177777                 MOV     #-1,%0
007170  012767  077777  177552         MOV     #77777,MODE1
007176  012767  000026  170572         MOV     #6,PSW
007204  074267  177540                 XOR     %0,MODE1
007210  016700  170562                 MOV     PSW,%0
007214  022700  000010                 CMP     #10,%0
007220  001401                         BEQ     .+4
007222  000000                         HLT                     ;ERROR! INCORRECT CONDITION CODES
```

```
                                ;N=1,Z=0,V=0,C=1
007224  010701                          SCOPE
007226  012700  077777                  MOV     #77777,%0
007232  012767  177777  177510          MOV     #-1,MODE1
007240  012767  000007  170530          MOV     #7,PSW
007246  074067  177476                  XOR     %0,MODE1
007252  016700  170520                  MOV     PSW,%0
007256  022700  000011                  CMP     #11,%0
007262  001401                          BEQ     .+4
007264  000000                          HLT                     ;ERROR! INCORRECT CONDITION CODES

        000000          N=0
        000002          SOURCE=%2
                        ;TEST THAT XOR OPERATES PROPERLY WHEN THE DESTINATION IS THE PSW,
007266  010701                          SCOPE
007270  012767  007322  170516          MOV     #TBITSET,14     ;SET UP TRACE TRAP VECTOR
007276  005067  170514                  CLR     16              ;TO TRAP TO 'TBITSET'
007302  012702  000377                  MOV     #377,%2
007306  005067  170464                  CLR     PSW             ;CLEAR THE PSW
007312  074267  170460                  XOR     %2,PSW          ;XOR THE #377 INTO A CLEAR PSW
007316  000240                          NOP
007320  000401                          BR      .+4
007322  000000          TBITSET:HLT                             ;T BIT SET
007324  022767  000357  170444          CMP     #357,PSW        ;DID ALL BUT THE T BIT SET?
007332  001401                          BEQ     .+4
007334  000000                          HLT                     ;PSW IS NOT CLEAR

007336  010701                          SCOPE
007340  012702  000357                  MOV     #357,%2
007344  012767  000357  170424          MOV     #357,PSW        ;PRE SET THE STATUS WORD
007352  074267  170420                  XOR     %2,PSW
007356  016700  170414                  MOV     PSW,%0          ;GET RESULT
007362  001401                          BEQ     .+4
007364  000000                          HLT                     ;ERROR XOR #357,#357 FAILED TO CLEAR PSW
007366  005267  171406          END:    INC     ICNT            ;INCREMENT THE PASS COUNT
007372  026727  171402  100000          CMP     ICNT,#100000
007400  001402                          BEQ     DONE            ;GO TO DONE IF 1000 PASSES COMPLETED
007402  000167  171400                  JMP     BEGIN           ;RESTART PROGRAM
007406  012767  000007  170152  DONE:   MOV     #7,TPBUF        ;RING BELL
007414  105767  170144          TSTB    TPCSR
007420  100375                          BPL     .-4
007422  013702  000042                  MOV     @#42,%2         ;GET DECTAPE MONITOR RETURN ADDRESS
007426  001404                          BEQ     DONE1           ;DO NOT RETURN IF (42)=0
007430  004712                          JSR     7,(2)           ;RETURN TO DECTAPE MONITOR
007432  000240                          NOP                     ;ACT11
007434  000240                          NOP                     ;OVERLAY
007436  000240                          NOP                     ;AREA
007440  000167  171336          DONE1:  JMP     START
        000001                          .END
```



```
BEGIN   001036          DEST    = 000202        DISPLA = 177570         DONE    007406
DONE1   007442          END     007366          HLT    = 000000         ICNT    001000
INDEX   006222          MODE1   000756          MODE27  000260          MODE7   000750
N       = 000000        PSW     = 177776        RSLT   =%000002         SCOPE   = 012701
SOURCE  =%000002        START   001002          STKPTR = 000500         SW      = 177572
TBITSE  007322          TPBUF   = 177566        TPCSR  = 177564         WDISPLAY= 177772
Y       = 000115        .       = 007444

ERRORS DETECTED:  0
```

```
                        .TITLE  MAINDEC-11-DCKBD-B 11/45,11/40 MARK TEST
                        .NLIST  MC,MD,SEQ
                        .LIST   ME
                        .ABS

                    ;TEST DCKBD- TEST OF THE MARK INSTRUCTION
                    ;OPERATIONS
                    ;       TEMP = PC+2+N
                    ;       PC   = R5
                    ;       R6   = TEMP
                    ;       R5   = (TEMP)           ;TEMP HOLDS A MEMORY ADDRESS
                    ;       R6   = TEMP+2

                    ;STARTING PROCEDURE
                    ;       LOAD ADDRESS=200
                    ;       PRESS START
                    ;       STACK POINTER IS SET 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                   ;STACK POINTER
000007                      PC=%7                   ;PROGRAM COUNTER

010701                      SCOPE=010701            ;MOVE PC TO R1
000000                      HLT=HALT                ;ERROR HALT
177776                      PSW=177776              ;PROCESSOR STATUS WORD ADDRESS
177770                      UBREAK=177770           ;ADDRESS OF POP11/45 MICRO BREAK REGISTER
177564                      TPCSR=177564            ;ADDRESS OF TELEPRINTER CSR
177566                      TPBUF=177566
177570                      SWR=177570              ;ADDRESS OF CONSOLE SWITCH REGISTER
177570                      DISPLAY=177570          ;ADDRESS OF CONSOLE DISPLAY REGISTER

                    ;*******************INITIAL STACK POINTER=0500**********************
000500              STKPTR=0500                     ;INITIAL STACK SETTING

                    ;MARK EQUATES
004400                      MARK0=MARK+0
004401                      MARK1=MARK+1
004402                      MARK2=MARK+2
```



```
                    000000                      .=0
                                                .NLIST  MC,MD,SEQ

002000  000002                                  .+2
000002  000000                                  HALT
000004  000006                                  .+2
000016  000000                                  HALT
000010  000012                                  .+2
000012  000000                                  HALT
000014  000016                                  .+2
000016  000000                                  HALT
000020  000022                                  .+2
000022  000000                                  HALT
000024  000026                                  .+2
000026  000000                                  HALT
000030  000032                                  .+2
000032  000000                                  HALT
000034  000036                                  .+2
000036  000000                                  HALT
000040  000042                                  .+2
000042  000000                                  HALT
000044  000046                                  .+2
000046  000000                                  HALT
000050  000052                                  .+2
000052  000000                                  HALT
000054  000056                                  .+2
000056  000000                                  HALT
000060  000062                                  .+2
000062  000000                                  HALT
000064  000066                                  .+2
000066  000000                                  HALT
000070  000072                                  .+2
000072  000000                                  HALT
000074  000076                                  .+2
000076  000000                                  HALT
000100  000102                                  .+2
000102  000000                                  HALT
000104  000106                                  .+2
000106  000000                                  HALT
000110  000112                                  .+2
000112  000000                                  HALT
000114  000116                                  .+2
000116  000000                                  HALT
000120  000122                                  .+2
000122  000000                                  HALT
000124  000126                                  .+2
000126  000000                                  HALT
000130  000132                                  .+2
000132  000000                                  HALT
000134  000136                                  .+2
000136  000000                                  HALT
000140  000142                                  .+2
```

```
000142  000000          HALT
000144  000140          .+2
000146  000000          HALT
000150  000152          .+2
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
.LIST   ME
```

```
                000200                          .=200
001200  000167  000576                          JMP     START
                001000                          .=1000
001000  000000                  ICNT:   0               ;CONTAINS PASS COUNT
001002  005067  177772          START:  CLR     ICNT            ;CLEAR PASS COUNT
001006  012706  005000          BEGIN:  MOV     #STKPTR,SP      ;SET STACK POINTER
001012  016737  177762  177570          MOV     ICNT,@#DISPLAY  ;DISPLAY PASS COUNT
001020  032737  000400  177570          BIT     #400,@#SWR      ;LOAD PDP11/45 MICRO BREAK REGISTER
001026  001403                          BEQ     .+10
001030  113737  177570  177770          MOVB    @#SWR,@#UBREAK  ;LOAD MICRO BREAK REG WITH SR0=7

                                ;TEST THAT THE OPERATION PC=R5 IS PERFORMED
001036  010701                          SCOPE
001040  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001044  012705  001054                  MOV     #T0A,R5         ;LOAD R5
001050  006400                          MARK    0               ;MARK N=0
001052  000000                          HALT                    ;PC=R5 FAILED
001054  000240                  T0A:    NOP                     ;PC=R5 OK

                                ;TEST THAT THE STACK POINTER (R6) CONTAINS THE PROPER VALUE
001056  010701                          SCOPE
001060  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001064  012705  001072                  MOV     #T1A,R5         ;LOAD R5
001070  006400                          MARK    0               ;MARK N=0

                                ;FUNCTION RESULTS  N=0
                                ;TEMP = PC+(2*N)        TEMP = #T1A+(2*N) = #T1A
                                ;PC   = R5              PC   = #T1A
                                ;SP   = TEMP            SP   = #T1A+(2*N) = #T1A
                                ;R5   = (TEMP)          R5   = (#T1A+(2*N) = (T1A)   NOT TESTED
                                ;SP   = TEMP+2          SP   = #T1A+(2*N)+2 = #T1A+2 TESTED

001072  010600                  T1A:    MOV     SP,R0           ;GET RESULTANT SP
001074  022706  001074                  CMP     #2+0+2+T1A,SP   ;SP = #T1A+(2*0)+2?  NOTE:
001100  001401                          BEQ     .+4             ;#2+0+2+T1A = #T1A+(2*0)+2
001102  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

                                ;TEST THAT R5 IS PROPERLY LOADED
001104  010701                          SCOPE
001106  012706  000500                  MOV     #STKPTR,SP
001112  012705  001120                  MOV     #T2A,R5
001116  006400                          MARK    0

                                ;FUNCTION RESULTS  N=0
                                ;TEMP = PC+(2*N)        TEMP = #T2A+(2*N) = #T2A
                                ;PC   = R5              PC   = #T2A
                                ;SP   = TEMP            SP   = #T2A+(2*N) = #T2A
                                ;R5   = (TEMP)          R5   = (#T2A+(2*N)) = (T2A) = 240 (NOP)
                                ;SP   = TEMP+2          SP   = #T2A+(2*N)+2 = #T2A+2

001120  000240                  T2A:    NOP
001122  010500                          MOV     R5,R0           ;GET CONTENTS OF R5
001124  022705  000240                  CMP     #NOP,R5         ;R5 = (#T2A+(2*N)) = (T2A)?
```



```
001130  001401                          BEQ     .+4
001132  000000                          HLT
                000003          C=3
                000000          M=0
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
001134  010701                          SCOPE
001136  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001142  012705  001150                  MOV     #T3A,R5 ;LOAD R5
001146  006400                          MARK    0               ;MARK N=0

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=0                                          *
                                ;TEMP = PC+(2*N)         TEMP= #T3A+(2*0)                      *
                                ;PC   = R5              PC   = T3A                             *
                                ;SP   = TEMP            SP   = #T3A+(2*0)                      *
                                ;R5   = (TEMP)          R5   = (#T3A+(2*0))                    *
                                ;SP   = TEMP+2          SP   = #T3A+(2*0)+2                    *
                                ;*****************************************************************

001150  010600                  T3A:    MOV     SP,R0           ;GET RESULTANT SP
001152  022706  001152                  CMP     #2+0+T3A+2,SP   ;SP = #T3A+(2*0)+2?
001156  001401                          BEQ     .+4
001160  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

                000004                  C=C+1
                000001                  M=M+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
001162  010701                          SCOPE
001164  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001170  012705  001176                  MOV     #T4A,R5 ;LOAD R5
001174  006401                          MARK    1               ;MARK N=1

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=1                                          *
                                ;TEMP = PC+(2*N)         TEMP= #T4A+(2*1)                      *
                                ;PC   = R5              PC   = T4A                             *
                                ;SP   = TEMP            SP   = #T4A+(2*1)                      *
                                ;R5   = (TEMP)          R5   = (#T4A+(2*1))                    *
                                ;SP   = TEMP+2          SP   = #T4A+(2*1)+2                    *
                                ;*****************************************************************

001176  010600                  T4A:    MOV     SP,R0           ;GET RESULTANT SP
001200  022706  001202                  CMP     #2+1+T4A+2,SP   ;SP = #T4A+(2*1)+2?
001204  001401                          BEQ     .+4
001206  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

                000005                  C=C+1
                000002                  M=M+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
001210  010701                          SCOPE
001212  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
```

```
001216  012705  001224          MOV     #T5A,R5 ;LOAD R5
001222  006402                  MARK    2               ;MARK N=2

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=2                                           *
                        ;TEMP = PC+(2*N)        TEMP= #T5A+(2*2)                        *
                        ;PC   = R5              PC  = T5A                               *
                        ;SP   = TEMP            SP  = #T5A+(2*2)                        *
                        ;R5   = (TEMP)          R5  = (#T5A+(2*2))                      *
                        ;SP   = TEMP+2          SP  = #T5A+(2*2)+2                      *
                        ;*****************************************************************

001224  010600          T5A:    MOV     SP,R0           ;GET RESULTANT SP
001226  022706  001232          CMP     #2*2+T5A+2,SP   ;SP = #T5A+(2*2)+2?
001232  001401                  BEQ     .+4
001234  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000006                  C=C+1
        000003                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
001236  010701                  SCOPE
001240  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001244  012705  001252          MOV     #T6A,R5 ;LOAD R5
001250  006403                  MARK    3               ;MARK N=3

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=3                                           *
                        ;TEMP = PC+(2*N)        TEMP= #T6A+(2*3)                        *
                        ;PC   = R5              PC  = T6A                               *
                        ;SP   = TEMP            SP  = #T6A+(2*3)                        *
                        ;R5   = (TEMP)          R5  = (#T6A+(2*3))                      *
                        ;SP   = TEMP+2          SP  = #T6A+(2*3)+2                      *
                        ;*****************************************************************

001252  010600          T6A:    MOV     SP,R0           ;GET RESULTANT SP
001254  022706  001262          CMP     #2*3+T6A+2,SP   ;SP = #T6A+(2*3)+2?
001260  001401                  BEQ     .+4
001262  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000007                  C=C+1
        000004                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
001264  010701                  SCOPE
001266  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001272  012705  001300          MOV     #T7A,R5 ;LOAD R5
001276  006404                  MARK    4               ;MARK N=4

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=4                                           *
                        ;TEMP = PC+(2*N)        TEMP= #T7A+(2*4)                        *
                        ;PC   = R5              PC  = T7A                               *
                        ;SP   = TEMP            SP  = #T7A+(2*4)                        *
```



```
                        ;R5   = (TEMP)          R5  = (#T7A+(2*4))                      *
                        ;SP   = TEMP+2          SP  = #T7A+(2*4)+2                      *
                        ;*****************************************************************

001300  010600          T7A:    MOV     SP,R0           ;GET RESULTANT SP
001302  022706  001312          CMP     #2*4+T7A+2,SP   ;SP = #T7A+(2*4)+2?
001306  001401                  BEQ     .+4
001310  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000010                  C=C+1
        000005                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
001312  010701                  SCOPE
001314  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001320  012705  001326          MOV     #T10A,R5        ;LOAD R5
001324  006405                  MARK    5               ;MARK N=5

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=5                                           *
                        ;TEMP = PC+(2*N)        TEMP= #T10A+(2*5)                       *
                        ;PC   = R5              PC  = T10A                              *
                        ;SP   = TEMP            SP  = #T10A+(2*5)                       *
                        ;R5   = (TEMP)          R5  = (#T10A+(2*5))                     *
                        ;SP   = TEMP+2          SP  = #T10A+(2*5)+2                     *
                        ;*****************************************************************

001326  010600          T10A:   MOV     SP,R0           ;GET RESULTANT SP
001330  022706  001342          CMP     #2*5+T10A+2,SP  ;SP = #T10A+(2*5)+2?
001334  001401                  BEQ     .+4
001336  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000011                  C=C+1
        000006                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
001340  010701                  SCOPE
001342  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001346  012705  001354          MOV     #T11A,R5        ;LOAD R5
001352  006406                  MARK    6               ;MARK N=6

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=6                                           *
                        ;TEMP = PC+(2*N)        TEMP= #T11A+(2*6)                       *
                        ;PC   = R5              PC  = T11A                              *
                        ;SP   = TEMP            SP  = #T11A+(2*6)                       *
                        ;R5   = (TEMP)          R5  = (#T11A+(2*6))                     *
                        ;SP   = TEMP+2          SP  = #T11A+(2*6)+2                     *
                        ;*****************************************************************

001354  010600          T11A:   MOV     SP,R0           ;GET RESULTANT SP
001356  022706  001372          CMP     #2*6+T11A+2,SP  ;SP = #T11A+(2*6)+2?
001362  001401                  BEQ     .+4
001364  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP
```

```
                000012                  C=C+1
                000007                  M=M+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
                                        SCOPE
001366  010701                          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001370  012706  000500                  MOV     #T12A,R5        ;LOAD R5
001374  012705  001402                  MARK    7               ;MARK N=7
001400  006407

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=7                                           *
                                ;TEMP = PC+(2*N)        TEMP= #T12A+(2*7)                       *
                                ;PC   = R5              PC  = T12A                              *
                                ;SP   = TEMP            SP  = #T12A+(2*7)                       *
                                ;R5   = (TEMP)          R5  = (#T12A+(2*7))                     *
                                ;SP   = TEMP+2          SP  = #T12A+(2*7)+2                     *
                                ;*****************************************************************

001402  010600                  T12A1   MOV     SP,R0           ;GET RESULTANT SP
001404  022706  001422                  CMP     #2*7+T12A+2,SP  ;SP = #T12A+(2*7)+2?
001410  001401                          BEQ     .+4
001412  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

                000013                  C=C+1
                000010                  M=M+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
001414  010701                          SCOPE
001416  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001422  012705  001430                  MOV     #T13A,R5        ;LOAD R5
001426  006410                          MARK    10              ;MARK N=10

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=10                                          *
                                ;TEMP = PC+(2*N)        TEMP= #T13A+(2*10)                      *
                                ;PC   = R5              PC  = T13A                              *
                                ;SP   = TEMP            SP  = #T13A+(2*10)                      *
                                ;R5   = (TEMP)          R5  = (#T13A+(2*10))                    *
                                ;SP   = TEMP+2          SP  = #T13A+(2*10)+2                    *
                                ;*****************************************************************

001430  010600                  T13A1   MOV     SP,R0           ;GET RESULTANT SP
001432  022706  001452                  CMP     #2*10+T13A+2,SP ;SP = #T13A+(2*10)+2?
001436  001401                          BEQ     .+4
001440  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

                000014                  C=C+1
                000011                  M=M+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
001442  010701                          SCOPE
001444  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001450  012705  001456                  MOV     #T14A,R5        ;LOAD R5
```



```
001454  006411                          MARK    11              ;MARK N=11

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=11                                          *
                                ;TEMP = PC+(2*N)        TEMP= #T14A+(2*11)                      *
                                ;PC   = R5              PC  = T14A                              *
                                ;SP   = TEMP            SP  = #T14A+(2*11)                      *
                                ;R5   = (TEMP)          R5  = (#T14A+(2*11))                    *
                                ;SP   = TEMP+2          SP  = #T14A+(2*11)+2                    *
                                ;*****************************************************************

001456  010600                  T14A1   MOV     SP,R0           ;GET RESULTANT SP
001460  022706  001502                  CMP     #2*11+T14A+2,SP ;SP = #T14A+(2*11)+2?
001464  001401                          BEQ     .+4
001466  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

                000015                  C=C+1
                000012                  M=M+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
001470  010701                          SCOPE
001472  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001476  012705  001504                  MOV     #T15A,R5        ;LOAD R5
001502  006412                          MARK    12              ;MARK N=12

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=12                                          *
                                ;TEMP = PC+(2*N)        TEMP= #T15A+(2*12)                      *
                                ;PC   = R5              PC  = T15A                              *
                                ;SP   = TEMP            SP  = #T15A+(2*12)                      *
                                ;R5   = (TEMP)          R5  = (#T15A+(2*12))                    *
                                ;SP   = TEMP+2          SP  = #T15A+(2*12)+2                    *
                                ;*****************************************************************

001504  010600                  T15A1   MOV     SP,R0           ;GET RESULTANT SP
001506  022706  001532                  CMP     #2*12+T15A+2,SP ;SP = #T15A+(2*12)+2?
001512  001401                          BEQ     .+4
001514  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

                000016                  C=C+1
                000013                  M=M+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
001516  010701                          SCOPE
001520  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001524  012705  001532                  MOV     #T16A,R5        ;LOAD R5
001530  006413                          MARK    13              ;MARK N=13

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=13                                          *
                                ;TEMP = PC+(2*N)        TEMP= #T16A+(2*13)                      *
                                ;PC   = R5              PC  = T16A                              *
                                ;SP   = TEMP            SP  = #T16A+(2*13)                      *
                                ;R5   = (TEMP)          R5  = (#T16A+(2*13))                    *
```

```
                                ;SP   = TEMP+2          SP   = #T16A+(2*13)+2
                                ;*****************************************************************

001532  010600          T16A1   MOV     SP,R0           ;GET RESULTANT SP
001534  022706  001562          CMP     #2*13+T16A+2,SP ;SP = #T16A+(2*13)+2?
001540  001401                  BEQ     .+4
001542  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000017                  C=C+1
        000014                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
                                SCOPE
001544  010701                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001546  012706  000500          MOV     #T17A,R5        ;LOAD R5
001552  012705  001560          MARK    14              ;MARK N=14
001556  006414

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=14                                          *
                                ;TEMP = PC+(2*N)          TEMP= #T17A+(2*14)                    *
                                ;PC   = R5                PC   = T17A                           *
                                ;SP   = TEMP              SP   = #T17A+(2*14)                   *
                                ;R5   = (TEMP)            R5   = (#T17A+(2*14))                 *
                                ;SP   = TEMP+2            SP   = #T17A+(2*14)+2                 *
                                ;*****************************************************************

001560  010600          T17A1   MOV     SP,R0           ;GET RESULTANT SP
001562  022706  001612          CMP     #2*14+T17A+2,SP ;SP = #T17A+(2*14)+2?
001566  001401                  BEQ     .+4
001570  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000020                  C=C+1
        000015                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
                                SCOPE
001572  010701                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001574  012706  000500          MOV     #T20A,R5        ;LOAD R5
001600  012705  001606          MARK    15              ;MARK N=15
001604  006415

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=15                                          *
                                ;TEMP = PC+(2*N)          TEMP= #T20A+(2*15)                    *
                                ;PC   = R5                PC   = T20A                           *
                                ;SP   = TEMP              SP   = #T20A+(2*15)                   *
                                ;R5   = (TEMP)            R5   = (#T20A+(2*15))                 *
                                ;SP   = TEMP+2            SP   = #T20A+(2*15)+2                 *
                                ;*****************************************************************

001606  010600          T20A1   MOV     SP,R0           ;GET RESULTANT SP
001610  022706  001642          CMP     #2*15+T20A+2,SP ;SP = #T20A+(2*15)+2?
001614  001401                  BEQ     .+4
001616  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP
```



```
        000021                  C=C+1
        000016                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
001620  010701                  SCOPE
001622  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001626  012705  001634          MOV     #T21A,R5        ;LOAD R5
001632  006416                  MARK    16              ;MARK N=16

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=16                                          *
                                ;TEMP = PC+(2*N)          TEMP= #T21A+(2*16)                    *
                                ;PC   = R5                PC   = T21A                           *
                                ;SP   = TEMP              SP   = #T21A+(2*16)                   *
                                ;R5   = (TEMP)            R5   = (#T21A+(2*16))                 *
                                ;SP   = TEMP+2            SP   = #T21A+(2*16)+2                 *
                                ;*****************************************************************

001634  010600          T21A1   MOV     SP,R0           ;GET RESULTANT SP
001636  022706  001672          CMP     #2*16+T21A+2,SP ;SP = #T21A+(2*16)+2?
001642  001401                  BEQ     .+4
001644  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000022                  C=C+1
        000017                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
001646  010701                  SCOPE
001650  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001654  012705  001662          MOV     #T22A,R5        ;LOAD R5
001660  006417                  MARK    17              ;MARK N=17

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=17                                          *
                                ;TEMP = PC+(2*N)          TEMP= #T22A+(2*17)                    *
                                ;PC   = R5                PC   = T22A                           *
                                ;SP   = TEMP              SP   = #T22A+(2*17)                   *
                                ;R5   = (TEMP)            R5   = (#T22A+(2*17))                 *
                                ;SP   = TEMP+2            SP   = #T22A+(2*17)+2                 *
                                ;*****************************************************************

001662  010600          T22A1   MOV     SP,R0           ;GET RESULTANT SP
001664  022706  001722          CMP     #2*17+T22A+2,SP ;SP = #T22A+(2*17)+2?
001670  001401                  BEQ     .+4
001672  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000023                  C=C+1
        000020                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
001674  010701                  SCOPE
001676  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001702  012705  001710          MOV     #T23A,R5        ;LOAD R5
001706  006420                  MARK    20              ;MARK N=20
```

```
                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=20                                          *
                        ;TEMP + PC+(2*N)          TEMP= #T23A+(2*20)                    *
                        ;PC   + R5                PC  = T23A                            *
                        ;SP   + TEMP              SP  = #T23A+(2*20)                    *
                        ;R5   + (TEMP)            R5  = (#T23A+(2*20))                  *
                        ;SP   + TEMP+2            SP  = #T23A+(2*20)+2                  *
                        ;*****************************************************************

001710  010600          T23A1:  MOV     SP,R0           ;GET RESULTANT SP
001712  022726  001752          CMP     #2*20+T23A+2,SP ;SP = #T23A+(2*20)+2?
001716  001401                  BEQ     .+4
001720  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000024                  C=C+1
        000021                  M=M+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
                                SCOPE
001722  012701                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001724  012706  000500          MOV     #T24A,R5        ;LOAD R5
001730  012705  001736          MARK    21              ;MARK N=21
001734  006421

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=21                                          *
                        ;TEMP + PC+(2*N)          TEMP= #T24A+(2*21)                    *
                        ;PC   + R5                PC  = T24A                            *
                        ;SP   + TEMP              SP  = #T24A+(2*21)                    *
                        ;R5   + (TEMP)            R5  = (#T24A+(2*21))                  *
                        ;SP   + TEMP+2            SP  = #T24A+(2*21)+2                  *
                        ;*****************************************************************

001736  010600          T24A1:  MOV     SP,R0           ;GET RESULTANT SP
001740  022726  002002          CMP     #2*21+T24A+2,SP ;SP = #T24A+(2*21)+2?
001744  001401                  BEQ     .+4
001746  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000025                  C=C+1
        000022                  M=M+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
001750  010701                  SCOPE
001752  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001756  012705  001764          MOV     #T25A,R5        ;LOAD R5
001762  006422                  MARK    22              ;MARK N=22

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=22                                          *
                        ;TEMP + PC+(2*N)          TEMP= #T25A+(2*22)                    *
                        ;PC   + R5                PC  = T25A                            *
                        ;SP   + TEMP              SP  = #T25A+(2*22)                    *
                        ;R5   + (TEMP)            R5  = (#T25A+(2*22))                  *
                        ;SP   + TEMP+2            SP  = #T25A+(2*22)+2                  *
                        ;*****************************************************************
```



```
                        ;*****************************************************************

001764  010600          T25A1:  MOV     SP,R0           ;GET RESULTANT SP
001766  022726  002032          CMP     #2*22+T25A+2,SP ;SP = #T25A+(2*22)+2?
001772  001401                  BEQ     .+4
001774  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000026                  C=C+1
        000023                  M=M+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
001776  010701                  SCOPE
002000  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002004  012705  002012          MOV     #T26A,R5        ;LOAD R5
002010  006423                  MARK    23              ;MARK N=23

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=23                                          *
                        ;TEMP + PC+(2*N)          TEMP= #T26A+(2*23)                    *
                        ;PC   + R5                PC  = T26A                            *
                        ;SP   + TEMP              SP  = #T26A+(2*23)                    *
                        ;R5   + (TEMP)            R5  = (#T26A+(2*23))                  *
                        ;SP   + TEMP+2            SP  = #T26A+(2*23)+2                  *
                        ;*****************************************************************

002012  010600          T26A1:  MOV     SP,R0           ;GET RESULTANT SP
002014  022726  002062          CMP     #2*23+T26A+2,SP ;SP = #T26A+(2*23)+2?
002020  001401                  BEQ     .+4
002022  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000027                  C=C+1
        000024                  M=M+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002024  010701                  SCOPE
002026  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002032  012705  002040          MOV     #T27A,R5        ;LOAD R5
002036  006424                  MARK    24              ;MARK N=24

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=24                                          *
                        ;TEMP + PC+(2*N)          TEMP= #T27A+(2*24)                    *
                        ;PC   + R5                PC  = T27A                            *
                        ;SP   + TEMP              SP  = #T27A+(2*24)                    *
                        ;R5   + (TEMP)            R5  = (#T27A+(2*24))                  *
                        ;SP   + TEMP+2            SP  = #T27A+(2*24)+2                  *
                        ;*****************************************************************

002040  010600          T27A1:  MOV     SP,R0           ;GET RESULTANT SP
002042  022726  002112          CMP     #2*24+T27A+2,SP ;SP = #T27A+(2*24)+2?
002046  001401                  BEQ     .+4
002050  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000030                  C=C+1
```

```
                    000025                      N=N+1
                                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002052  012701                              SCOPE
002054  012706  000500                      MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002060  012705  002066                      MOV     #T30A,R5        ;LOAD R5
002064  006425                              MARK    25              ;MARK N=25

                                        ;*****************************************************************
                                        ;FUNCTION RESULTS  N=25                                         *
                                        ;TEMP = PC+(2*N)        TEMP= #T30A+(2*25)                      *
                                        ;PC   = R5              PC  = T30A                              *
                                        ;SP   = TEMP            SP  = #T30A+(2*25)                      *
                                        ;R5   = (TEMP)          R5  = (#T30A+(2*25))                    *
                                        ;SP   = TEMP+2          SP  = #T30A+(2*25)+2                    *
                                        ;*****************************************************************

002066  012600                      T30A1:  MOV     SP,R0           ;GET RESULTANT SP
002070  022706  002142                      CMP     #2*25+T30A+2,SP ;SP = #T30A+(2*25)+2?
002074  001401                              BEQ     .+4
002076  000000                              HLT                     ;ERROR INCORRECT RESULTANT SP

                    000031                      C=C+1
                    000026                      N=N+1
                                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002100  012701                              SCOPE
002102  012706  000500                      MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002106  012705  002114                      MOV     #T31A,R5        ;LOAD R5
002112  006426                              MARK    26              ;MARK N=26

                                        ;*****************************************************************
                                        ;FUNCTION RESULTS  N=26                                         *
                                        ;TEMP = PC+(2*N)        TEMP= #T31A+(2*26)                      *
                                        ;PC   = R5              PC  = T31A                              *
                                        ;SP   = TEMP            SP  = #T31A+(2*26)                      *
                                        ;R5   = (TEMP)          R5  = (#T31A+(2*26))                    *
                                        ;SP   = TEMP+2          SP  = #T31A+(2*26)+2                    *
                                        ;*****************************************************************

002114  012600                      T31A1:  MOV     SP,R0           ;GET RESULTANT SP
002116  022706  002172                      CMP     #2*26+T31A+2,SP ;SP = #T31A+(2*26)+2?
002122  001401                              BEQ     .+4
002124  000000                              HLT                     ;ERROR INCORRECT RESULTANT SP

                    000032                      C=C+1
                    000027                      N=N+1
                                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002126  012701                              SCOPE
002132  012706  000500                      MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002134  012705  002142                      MOV     #T32A,R5        ;LOAD R5
002140  006427                              MARK    27              ;MARK N=27
```



```
                                        ;*****************************************************************
                                        ;FUNCTION RESULTS  N=27                                         *
                                        ;TEMP = PC+(2*N)        TEMP= #T32A+(2*27)                      *
                                        ;PC   = R5              PC  = T32A                              *
                                        ;SP   = TEMP            SP  = #T32A+(2*27)                      *
                                        ;R5   = (TEMP)          R5  = (#T32A+(2*27))                    *
                                        ;SP   = TEMP+2          SP  = #T32A+(2*27)+2                    *
                                        ;*****************************************************************

002142  012600                      T32A1:  MOV     SP,R0           ;GET RESULTANT SP
002144  022706  002222                      CMP     #2*27+T32A+2,SP ;SP = #T32A+(2*27)+2?
002150  001401                              BEQ     .+4
002152  000000                              HLT                     ;ERROR INCORRECT RESULTANT SP

                    000033                      C=C+1
                    000030                      N=N+1
                                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002154  012701                              SCOPE
002156  012706  000500                      MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002162  012705  002170                      MOV     #T33A,R5        ;LOAD R5
002166  006430                              MARK    30              ;MARK N=30

                                        ;*****************************************************************
                                        ;FUNCTION RESULTS  N=30                                         *
                                        ;TEMP = PC+(2*N)        TEMP= #T33A+(2*30)                      *
                                        ;PC   = R5              PC  = T33A                              *
                                        ;SP   = TEMP            SP  = #T33A+(2*30)                      *
                                        ;R5   = (TEMP)          R5  = (#T33A+(2*30))                    *
                                        ;SP   = TEMP+2          SP  = #T33A+(2*30)+2                    *
                                        ;*****************************************************************

002170  012600                      T33A1:  MOV     SP,R0           ;GET RESULTANT SP
002172  022706  002252                      CMP     #2*30+T33A+2,SP ;SP = #T33A+(2*30)+2?
002176  001401                              BEQ     .+4
002200  000000                              HLT                     ;ERROR INCORRECT RESULTANT SP

                    000034                      C=C+1
                    000031                      N=N+1
                                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002202  012701                              SCOPE
002204  012706  000500                      MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002210  012705  002216                      MOV     #T34A,R5        ;LOAD R5
002214  006431                              MARK    31              ;MARK N=31

                                        ;*****************************************************************
                                        ;FUNCTION RESULTS  N=31                                         *
                                        ;TEMP = PC+(2*N)        TEMP= #T34A+(2*31)                      *
                                        ;PC   = R5              PC  = T34A                              *
                                        ;SP   = TEMP            SP  = #T34A+(2*31)                      *
                                        ;R5   = (TEMP)          R5  = (#T34A+(2*31))                    *
                                        ;SP   = TEMP+2          SP  = #T34A+(2*31)+2                    *
                                        ;*****************************************************************
```

```
002216  010600              T34A1:  MOV     SP,R0           ;GET RESULTANT SP
002220  027706  002302              CMP     #2*31+T34A+2,SP ;SP = #T34A+(2*31)+2T
002224  001401                      BEQ     .+4
002226  000000                      HLT                     ;ERROR INCORRECT RESULTANT SP

        000035                      C=C+1
        000032                      N=N+1
                            ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                            ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002230  010701                      SCOPE
002232  012706  000500              MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002236  012705  002244              MOV     #T35A,R5        ;LOAD R5
002242  006432                      MARK    32              ;MARK N=32

                            ;*****************************************************************
                            ;FUNCTION RESULTS  N=32                                          *
                            ;TEMP = PC+(2*N)          TEMP= #T35A+(2*32)                    *
                            ;PC   = R5                PC  = T35A                            *
                            ;SP   = TEMP              SP  = #T35A+(2*32)                    *
                            ;R5   = (TEMP)            R5  = (#T35A+(2*32))                  *
                            ;SP   = TEMP+2            SP  = #T35A+(2*32)+2                  *
                            ;*****************************************************************

002244  010600              T35A1:  MOV     SP,R0           ;GET RESULTANT SP
002246  027706  002332              CMP     #2*32+T35A+2,SP ;SP = #T35A+(2*32)+2T
002252  001401                      BEQ     .+4
002254  000000                      HLT                     ;ERROR INCORRECT RESULTANT SP

        000036                      C=C+1
        000033                      N=N+1
                            ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                            ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002256  010701                      SCOPE
002260  012706  000500              MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002264  012705  002272              MOV     #T36A,R5        ;LOAD R5
002270  006433                      MARK    33              ;MARK N=33

                            ;*****************************************************************
                            ;FUNCTION RESULTS  N=33                                          *
                            ;TEMP = PC+(2*N)          TEMP= #T36A+(2*33)                    *
                            ;PC   = R5                PC  = T36A                            *
                            ;SP   = TEMP              SP  = #T36A+(2*33)                    *
                            ;R5   = (TEMP)            R5  = (#T36A+(2*33))                  *
                            ;SP   = TEMP+2            SP  = #T36A+(2*33)+2                  *
                            ;*****************************************************************

002272  010600              T36A1:  MOV     SP,R0           ;GET RESULTANT SP
002274  027706  002362              CMP     #2*33+T36A+2,SP ;SP = #T36A+(2*33)+2T
002300  001401                      BEQ     .+4
002302  000000                      HLT                     ;ERROR INCORRECT RESULTANT SP

        000037                      C=C+1
        000034                      N=N+1
```



```
                            ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                            ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002304  010701                      SCOPE
002306  012706  000500              MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002312  012705  002320              MOV     #T37A,R5        ;LOAD R5
002316  006434                      MARK    34              ;MARK N=34

                            ;*****************************************************************
                            ;FUNCTION RESULTS  N=34                                          *
                            ;TEMP = PC+(2*N)          TEMP= #T37A+(2*34)                    *
                            ;PC   = R5                PC  = T37A                            *
                            ;SP   = TEMP              SP  = #T37A+(2*34)                    *
                            ;R5   = (TEMP)            R5  = (#T37A+(2*34))                  *
                            ;SP   = TEMP+2            SP  = #T37A+(2*34)+2                  *
                            ;*****************************************************************

002320  010600              T37A1:  MOV     SP,R0           ;GET RESULTANT SP
002322  027706  002412              CMP     #2*34+T37A+2,SP ;SP = #T37A+(2*34)+2T
002326  001401                      BEQ     .+4
002330  000000                      HLT                     ;ERROR INCORRECT RESULTANT SP

        000040                      C=C+1
        000035                      N=N+1
                            ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                            ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002332  010701                      SCOPE
002334  012706  000500              MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002340  012705  002346              MOV     #T40A,R5        ;LOAD R5
002344  006435                      MARK    35              ;MARK N=35

                            ;*****************************************************************
                            ;FUNCTION RESULTS  N=35                                          *
                            ;TEMP = PC+(2*N)          TEMP= #T40A+(2*35)                    *
                            ;PC   = R5                PC  = T40A                            *
                            ;SP   = TEMP              SP  = #T40A+(2*35)                    *
                            ;R5   = (TEMP)            R5  = (#T40A+(2*35))                  *
                            ;SP   = TEMP+2            SP  = #T40A+(2*35)+2                  *
                            ;*****************************************************************

002346  010600              T40A1:  MOV     SP,R0           ;GET RESULTANT SP
002350  027706  002442              CMP     #2*35+T40A+2,SP ;SP = #T40A+(2*35)+2T
002354  001401                      BEQ     .+4
002356  000000                      HLT                     ;ERROR INCORRECT RESULTANT SP

        000041                      C=C+1
        000036                      N=N+1
                            ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                            ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002360  010701                      SCOPE
002362  012706  000500              MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002366  012705  002374              MOV     #T41A,R5        ;LOAD R5
002372  006436                      MARK    36              ;MARK N=36

                            ;*****************************************************************
```

```
                        ;FUNCTION RESULTS  N=36                                          *
                        ;TEMP = PC+(2*N)          TEMP= #T41A+(2*36)                     *
                        ;PC   = R5                PC  = T41A                             *
                        ;SP   = TEMP              SP  = #T41A+(2*36)                     *
                        ;R5   = (TEMP)            R5  = (#T41A+(2*36))                   *
                        ;SP   = TEMP+2            SP  = #T41A+(2*36)+2                   *
                        ;**************************************************************

002374  010600          T41A1   MOV     SP,R0           ;GET RESULTANT SP
002376  027706  002472          CMP     #2*36+T41A+2,SP ;SP = #T41A+(2*36)+2?
002402  001401                  BEQ     .+4
002404  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000042                  C=C+1
        000037                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002406  010701                  SCOPE
002410  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002414  012705  002422          MOV     #T42A,R5        ;LOAD R5
002420  006437                  MARK    37              ;MARK N=37

                        ;**************************************************************
                        ;FUNCTION RESULTS  N=37                                          *
                        ;TEMP = PC+(2*N)          TEMP= #T42A+(2*37)                     *
                        ;PC   = R5                PC  = T42A                             *
                        ;SP   = TEMP              SP  = #T42A+(2*37)                     *
                        ;R5   = (TEMP)            R5  = (#T42A+(2*37))                   *
                        ;SP   = TEMP+2            SP  = #T42A+(2*37)+2                   *
                        ;**************************************************************

002422  010600          T42A1   MOV     SP,R0           ;GET RESULTANT SP
002424  027706  002522          CMP     #2*37+T42A+2,SP ;SP = #T42A+(2*37)+2?
002430  001401                  BEQ     .+4
002432  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000043                  C=C+1
        000040                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002434  010701                  SCOPE
002436  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002442  012705  002450          MOV     #T43A,R5        ;LOAD R5
002446  006440                  MARK    40              ;MARK N=40

                        ;**************************************************************
                        ;FUNCTION RESULTS  N=40                                          *
                        ;TEMP = PC+(2*N)          TEMP= #T43A+(2*40)                     *
                        ;PC   = R5                PC  = T43A                             *
                        ;SP   = TEMP              SP  = #T43A+(2*40)                     *
                        ;R5   = (TEMP)            R5  = (#T43A+(2*40))                   *
                        ;SP   = TEMP+2            SP  = #T43A+(2*40)+2                   *
                        ;**************************************************************
```



```
002450  010600          T43A1   MOV     SP,R0           ;GET RESULTANT SP
002452  027706  002552          CMP     #2*40+T43A+2,SP ;SP = #T43A+(2*40)+2?
002456  001401                  BEQ     .+4
002460  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000044                  C=C+1
        000041                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002462  010701                  SCOPE
002464  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002470  012705  002476          MOV     #T44A,R5        ;LOAD R5
002474  006441                  MARK    41              ;MARK N=41

                        ;**************************************************************
                        ;FUNCTION RESULTS  N=41                                          *
                        ;TEMP = PC+(2*N)          TEMP= #T44A+(2*41)                     *
                        ;PC   = R5                PC  = T44A                             *
                        ;SP   = TEMP              SP  = #T44A+(2*41)                     *
                        ;R5   = (TEMP)            R5  = (#T44A+(2*41))                   *
                        ;SP   = TEMP+2            SP  = #T44A+(2*41)+2                   *
                        ;**************************************************************

002476  010600          T44A1   MOV     SP,R0           ;GET RESULTANT SP
002500  027706  002602          CMP     #2*41+T44A+2,SP ;SP = #T44A+(2*41)+2?
002504  001401                  BEQ     .+4
002506  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000045                  C=C+1
        000042                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002510  010701                  SCOPE
002512  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002516  012705  002524          MOV     #T45A,R5        ;LOAD R5
002522  006442                  MARK    42              ;MARK N=42

                        ;**************************************************************
                        ;FUNCTION RESULTS  N=42                                          *
                        ;TEMP = PC+(2*N)          TEMP= #T45A+(2*42)                     *
                        ;PC   = R5                PC  = T45A                             *
                        ;SP   = TEMP              SP  = #T45A+(2*42)                     *
                        ;R5   = (TEMP)            R5  = (#T45A+(2*42))                   *
                        ;SP   = TEMP+2            SP  = #T45A+(2*42)+2                   *
                        ;**************************************************************

002524  010600          T45A1   MOV     SP,R0           ;GET RESULTANT SP
002526  027706  002632          CMP     #2*42+T45A+2,SP ;SP = #T45A+(2*42)+2?
002532  001401                  BEQ     .+4
002534  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000046                  C=C+1
        000043                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
```

```
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
                                        SCOPE
002536  010701                          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002540  012706  000500                  MOV     #T46A,R5        ;LOAD R5
002544  012705  002552                  MARK    43              ;MARK N=43
002550  006443

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=43                                          *
                                ;TEMP = PC+(2*N)          TEMP= #T46A+(2*43)                    *
                                ;PC   = R5                PC  = T46A                            *
                                ;SP   = TEMP              SP  = #T46A+(2*43)                    *
                                ;R5   = (TEMP)            R5  = (#T46A+(2*43))                  *
                                ;SP   = TEMP+2            SP  = #T46A+(2*43)+2                  *
                                ;*****************************************************************

002552  010620                  T46A1   MOV     SP,R0           ;GET RESULTANT SP
002554  022706  002662                  CMP     #2*43+T46A+2,SP ;SP = #T46A+(2*43)+2?
002560  001401                          BEQ     .+4
002562  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

        000047                          C=C+1
        000044                          N=N+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002564  010701                          SCOPE
002566  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002572  012705  002600                  MOV     #T47A,R5        ;LOAD R5
002576  006444                          MARK    44              ;MARK N=44

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=44                                          *
                                ;TEMP = PC+(2*N)          TEMP= #T47A+(2*44)                    *
                                ;PC   = R5                PC  = T47A                            *
                                ;SP   = TEMP              SP  = #T47A+(2*44)                    *
                                ;R5   = (TEMP)            R5  = (#T47A+(2*44))                  *
                                ;SP   = TEMP+2            SP  = #T47A+(2*44)+2                  *
                                ;*****************************************************************

002600  010600                  T47A1   MOV     SP,R0           ;GET RESULTANT SP
002602  022706  002712                  CMP     #2*44+T47A+2,SP ;SP = #T47A+(2*44)+2?
002606  001401                          BEQ     .+4
002610  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

        000050                          C=C+1
        000045                          N=N+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002612  010701                          SCOPE
002614  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002620  012705  002626                  MOV     #T52A,R5        ;LOAD R5
002624  006445                          MARK    45              ;MARK N=45

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=45                                          *
```



```
                                ;TEMP = PC+(2*N)          TEMP= #T52A+(2*45)                    *
                                ;PC   = R5                PC  = T52A                            *
                                ;SP   = TEMP              SP  = #T52A+(2*45)                    *
                                ;R5   = (TEMP)            R5  = (#T52A+(2*45))                  *
                                ;SP   = TEMP+2            SP  = #T52A+(2*45)+2                  *
                                ;*****************************************************************

002626  010600                  T52A1   MOV     SP,R0           ;GET RESULTANT SP
002630  022706  002742                  CMP     #2*45+T52A+2,SP ;SP = #T52A+(2*45)+2?
002634  001401                          BEQ     .+4
002636  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

        000051                          C=C+1
        000046                          N=N+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002640  010701                          SCOPE
002642  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002646  012705  002654                  MOV     #T51A,R5        ;LOAD R5
002652  006446                          MARK    46              ;MARK N=46

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=46                                          *
                                ;TEMP = PC+(2*N)          TEMP= #T51A+(2*46)                    *
                                ;PC   = R5                PC  = T51A                            *
                                ;SP   = TEMP              SP  = #T51A+(2*46)                    *
                                ;R5   = (TEMP)            R5  = (#T51A+(2*46))                  *
                                ;SP   = TEMP+2            SP  = #T51A+(2*46)+2                  *
                                ;*****************************************************************

002654  010600                  T51A1   MOV     SP,R0           ;GET RESULTANT SP
002656  022706  002772                  CMP     #2*46+T51A+2,SP ;SP = #T51A+(2*46)+2?
002662  001401                          BEQ     .+4
002664  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

        000052                          C=C+1
        000047                          N=N+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002666  010701                          SCOPE
002670  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002674  012705  002702                  MOV     #T52A,R5        ;LOAD R5
002700  006447                          MARK    47              ;MARK N=47

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=47                                          *
                                ;TEMP = PC+(2*N)          TEMP= #T52A+(2*47)                    *
                                ;PC   = R5                PC  = T52A                            *
                                ;SP   = TEMP              SP  = #T52A+(2*47)                    *
                                ;R5   = (TEMP)            R5  = (#T52A+(2*47))                  *
                                ;SP   = TEMP+2            SP  = #T52A+(2*47)+2                  *
                                ;*****************************************************************

002702  010600                  T52A1   MOV     SP,R0           ;GET RESULTANT SP
```

```
002704  022706  003022          CMP     #2+47+T52A+2,SP ;SP = #T52A+(2+47)+2?
002710  001401                  BEQ     .+4
002712  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000053                  C=C+1
        000050                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002714  010701                  SCOPE
002716  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002722  012705  002730          MOV     #T53A,R5        ;LOAD R5
002726  006450                  MARK    50              ;MARK N=50

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=50                                          *
                        ;TEMP = PC+(2*N)        TEMP= #T53A+(2*50)                      *
                        ;PC   = R5              PC   = T53A                             *
                        ;SP   = TEMP            SP   = #T53A+(2*50)                     *
                        ;R5   = (TEMP)          R5   = (#T53A+(2*50))                   *
                        ;SP   = TEMP+2          SP   = #T53A+(2*50)+2                   *
                        ;*****************************************************************

002730  010600          T53A1:  MOV     SP,R0           ;GET RESULTANT SP
002732  022706  003052          CMP     #2+50+T53A+2,SP ;SP = #T53A+(2+50)+2?
002736  001401                  BEQ     .+4
002740  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000054                  C=C+1
        000051                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
002742  010701                  SCOPE
002744  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002750  012705  002756          MOV     #T54A,R5        ;LOAD R5
002754  006451                  MARK    51              ;MARK N=51

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=51                                          *
                        ;TEMP = PC+(2*N)        TEMP= #T54A+(2*51)                      *
                        ;PC   = R5              PC   = T54A                             *
                        ;SP   = TEMP            SP   = #T54A+(2*51)                     *
                        ;R5   = (TEMP)          R5   = (#T54A+(2*51))                   *
                        ;SP   = TEMP+2          SP   = #T54A+(2*51)+2                   *
                        ;*****************************************************************

002756  010600          T54A1:  MOV     SP,R0           ;GET RESULTANT SP
002760  022706  003102          CMP     #2+51+T54A+2,SP ;SP = #T54A+(2*51)+2?
002764  001401                  BEQ     .+4
002766  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000055                  C=C+1
        000052                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
```



```
002770  010701                  SCOPE
002772  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
002776  012705  003004          MOV     #T55A,R5        ;LOAD R5
003002  006452                  MARK    52              ;MARK N=52

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=52                                          *
                        ;TEMP = PC+(2*N)        TEMP= #T55A+(2*52)                      *
                        ;PC   = R5              PC   = T55A                             *
                        ;SP   = TEMP            SP   = #T55A+(2*52)                     *
                        ;R5   = (TEMP)          R5   = (#T55A+(2*52))                   *
                        ;SP   = TEMP+2          SP   = #T55A+(2*52)+2                   *
                        ;*****************************************************************

003004  010600          T55A1:  MOV     SP,R0           ;GET RESULTANT SP
003006  022706  003132          CMP     #2+52+T55A+2,SP ;SP = #T55A+(2*52)+2?
003012  001401                  BEQ     .+4
003014  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000056                  C=C+1
        000053                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
003016  010701                  SCOPE
003020  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003024  012705  003032          MOV     #T56A,R5        ;LOAD R5
003030  006453                  MARK    53              ;MARK N=53

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=53                                          *
                        ;TEMP = PC+(2*N)        TEMP= #T56A+(2*53)                      *
                        ;PC   = R5              PC   = T56A                             *
                        ;SP   = TEMP            SP   = #T56A+(2*53)                     *
                        ;R5   = (TEMP)          R5   = (#T56A+(2*53))                   *
                        ;SP   = TEMP+2          SP   = #T56A+(2*53)+2                   *
                        ;*****************************************************************

003032  010600          T56A1:  MOV     SP,R0           ;GET RESULTANT SP
003034  022706  003162          CMP     #2+53+T56A+2,SP ;SP = #T56A+(2*53)+2?
003040  001401                  BEQ     .+4
003042  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000057                  C=C+1
        000054                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
003044  010701                  SCOPE
003046  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003052  012705  003060          MOV     #T57A,R5        ;LOAD R5
003056  006454                  MARK    54              ;MARK N=54

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=54                                          *
                        ;TEMP = PC+(2*N)        TEMP= #T57A+(2*54)                      *
```

```
                                ;PC   = R5              PC   = T57A                 *
                                ;SP   = TEMP            SP   = #T57A+(2*54)         *
                                ;R5   = (TEMP)          R5   = (#T57A+(2*54))       *
                                ;SP   = TEMP+2          SP   = #T57A+(2*54)+2       *
                                ;***************************************************

003040  010600                  T57A1:  MOV     SP,R0           ;GET RESULTANT SP
003042  022706  003212                  CMP     #2*54+T57A+2,SP ;SP = #T57A+(2*54)+2?
003046  001401                          BEQ     .+4
003050  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

        000060                  C=C+1
        000055                  N=N+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
003052  010701                          SCOPE
003054  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003060  012705  003106                  MOV     #T62A,R5        ;LOAD R5
003064  006455                          MARK    55              ;MARK N=55

                                ;***************************************************
                                ;FUNCTION RESULTS  N=55                             *
                                ;TEMP = PC+(2*N)        TEMP= #T60A+(2*55)          *
                                ;PC   = R5              PC   = T60A                 *
                                ;SP   = TEMP            SP   = #T60A+(2*55)         *
                                ;R5   = (TEMP)          R5   = (#T60A+(2*55))       *
                                ;SP   = TEMP+2          SP   = #T60A+(2*55)+2       *
                                ;***************************************************

003066  010600                  T60A1:  MOV     SP,R0           ;GET RESULTANT SP
003070  022706  003242                  CMP     #2*55+T60A+2,SP ;SP = #T60A+(2*55)+2?
003074  001401                          BEQ     .+4
003076  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

        000061                  C=C+1
        000056                  N=N+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
003100  010701                          SCOPE
003102  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003106  012705  003134                  MOV     #T61A,R5        ;LOAD R5
003112  006456                          MARK    56              ;MARK N=56

                                ;***************************************************
                                ;FUNCTION RESULTS  N=56                             *
                                ;TEMP = PC+(2*N)        TEMP= #T61A+(2*56)          *
                                ;PC   = R5              PC   = T61A                 *
                                ;SP   = TEMP            SP   = #T61A+(2*56)         *
                                ;R5   = (TEMP)          R5   = (#T61A+(2*56))       *
                                ;SP   = TEMP+2          SP   = #T61A+(2*56)+2       *
                                ;***************************************************

003114  010600                  T61A1:  MOV     SP,R0           ;GET RESULTANT SP
003116  022706  003272                  CMP     #2*56+T61A+2,SP ;SP = #T61A+(2*56)+2?
```



```
003120  001401                          BEQ     .+4
003122  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

        000062                  C=C+1
        000057                  N=N+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
003124  010701                          SCOPE
003126  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003132  012705  003162                  MOV     #T62A,R5        ;LOAD R5
003136  006457                          MARK    57              ;MARK N=57

                                ;***************************************************
                                ;FUNCTION RESULTS  N=57                             *
                                ;TEMP = PC+(2*N)        TEMP= #T62A+(2*57)          *
                                ;PC   = R5              PC   = T62A                 *
                                ;SP   = TEMP            SP   = #T62A+(2*57)         *
                                ;R5   = (TEMP)          R5   = (#T62A+(2*57))       *
                                ;SP   = TEMP+2          SP   = #T62A+(2*57)+2       *
                                ;***************************************************

003140  010600                  T62A1:  MOV     SP,R0           ;GET RESULTANT SP
003142  022706  003322                  CMP     #2*57+T62A+2,SP ;SP = #T62A+(2*57)+2?
003146  001401                          BEQ     .+4
003150  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

        000063                  C=C+1
        000060                  N=N+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
003152  010701                          SCOPE
003154  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003160  012705  003210                  MOV     #T63A,R5        ;LOAD R5
003164  006460                          MARK    60              ;MARK N=60

                                ;***************************************************
                                ;FUNCTION RESULTS  N=60                             *
                                ;TEMP = PC+(2*N)        TEMP= #T63A+(2*60)          *
                                ;PC   = R5              PC   = T63A                 *
                                ;SP   = TEMP            SP   = #T63A+(2*60)         *
                                ;R5   = (TEMP)          R5   = (#T63A+(2*60))       *
                                ;SP   = TEMP+2          SP   = #T63A+(2*60)+2       *
                                ;***************************************************

003166  010600                  T63A1:  MOV     SP,R0           ;GET RESULTANT SP
003170  022706  003352                  CMP     #2*60+T63A+2,SP ;SP = #T63A+(2*60)+2?
003174  001401                          BEQ     .+4
003176  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

        000064                  C=C+1
        000061                  N=N+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
003200  010701                          SCOPE
```

```
003224  012706  003500        MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003230  012705  003236        MOV     #T64A,R5        ;LOAD R5
003234  006461                MARK    61              ;MARK N=61

                              ;*****************************************************************
                              ;FUNCTION RESULTS  N=61                                         *
                              ;TEMP = PC+(2*N)          TEMP= #T64A+(2*61)                    *
                              ;PC   = R5                PC  = T64A                            *
                              ;SP   = TEMP              SP  = #T64A+(2*61)                    *
                              ;R5   = (TEMP)            R5  = (#T64A+(2*61))                  *
                              ;SP   = TEMP+2            SP  = #T64A+(2*61)+2                  *
                              ;*****************************************************************

003236  010600        T64A1   MOV     SP,R0           ;GET RESULTANT SP
003240  022706  003402        CMP     #2*61+T64A+2,SP ;SP = #T64A+(2*61)+2?
003244  001401                BEQ     .+4
003246  000000                HLT                     ;ERROR INCORRECT RESULTANT SP

        000065                C=C+1
        000062                M=M+1
                              ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                              ;INSTRUCTION FOR ALL VALUES OF N (0-77)
                                      SCOPE
003250  010701                        MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003254  012706  003500                MOV     #T65A,R5        ;LOAD R5
003260  012705  003264                MARK    62              ;MARK N=62
003266  006462

                              ;*****************************************************************
                              ;FUNCTION RESULTS  N=62                                         *
                              ;TEMP = PC+(2*N)          TEMP= #T65A+(2*62)                    *
                              ;PC   = R5                PC  = T65A                            *
                              ;SP   = TEMP              SP  = #T65A+(2*62)                    *
                              ;R5   = (TEMP)            R5  = (#T65A+(2*62))                  *
                              ;SP   = TEMP+2            SP  = #T65A+(2*62)+2                  *
                              ;*****************************************************************

003264  010600        T65A1   MOV     SP,R0           ;GET RESULTANT SP
003266  022706  003432        CMP     #2*62+T65A+2,SP ;SP = #T65A+(2*62)+2?
003272  001401                BEQ     .+4
003274  000000                HLT                     ;ERROR INCORRECT RESULTANT SP

        000066                C=C+1
        000063                M=M+1
                              ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                              ;INSTRUCTION FOR ALL VALUES OF N (0-77)
                                      SCOPE
003276  010701                        MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003300  012706  003500                MOV     #T66A,R5        ;LOAD R5
003304  012705  003312                MARK    63              ;MARK N=63
003310  006463

                              ;*****************************************************************
                              ;FUNCTION RESULTS  N=63                                         *
                              ;TEMP = PC+(2*N)          TEMP= #T66A+(2*63)                    *
                              ;PC   = R5                PC  = T66A                            *
```



```
                              ;SP   = TEMP              SP  = #T66A+(2*63)                    *
                              ;R5   = (TEMP)            R5  = (#T66A+(2*63))                  *
                              ;SP   = TEMP+2            SP  = #T66A+(2*63)+2                  *
                              ;*****************************************************************

003312  010600        T66A1   MOV     SP,R0           ;GET RESULTANT SP
003314  022706  003462        CMP     #2*63+T66A+2,SP ;SP = #T66A+(2*63)+2?
003320  001401                BEQ     .+4
003322  000000                HLT                     ;ERROR INCORRECT RESULTANT SP

        000067                C=C+1
        000064                M=M+1
                              ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                              ;INSTRUCTION FOR ALL VALUES OF N (0-77)
003324  010701                        SCOPE
003326  012706  003500                MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003332  012705  003340                MOV     #T67A,R5        ;LOAD R5
003336  006464                MARK    64              ;MARK N=64

                              ;*****************************************************************
                              ;FUNCTION RESULTS  N=64                                         *
                              ;TEMP = PC+(2*N)          TEMP= #T67A+(2*64)                    *
                              ;PC   = R5                PC  = T67A                            *
                              ;SP   = TEMP              SP  = #T67A+(2*64)                    *
                              ;R5   = (TEMP)            R5  = (#T67A+(2*64))                  *
                              ;SP   = TEMP+2            SP  = #T67A+(2*64)+2                  *
                              ;*****************************************************************

003340  010600        T67A1   MOV     SP,R0           ;GET RESULTANT SP
003342  022706  003512        CMP     #2*64+T67A+2,SP ;SP = #T67A+(2*64)+2?
003346  001401                BEQ     .+4
003350  000000                HLT                     ;ERROR INCORRECT RESULTANT SP

        000070                C=C+1
        000065                M=M+1
                              ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                              ;INSTRUCTION FOR ALL VALUES OF N (0-77)
003352  010701                        SCOPE
003354  012706  003500                MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003360  012705  003366                MOV     #T70A,R5        ;LOAD R5
003364  006465                MARK    65              ;MARK N=65

                              ;*****************************************************************
                              ;FUNCTION RESULTS  N=65                                         *
                              ;TEMP = PC+(2*N)          TEMP= #T70A+(2*65)                    *
                              ;PC   = R5                PC  = T70A                            *
                              ;SP   = TEMP              SP  = #T70A+(2*65)                    *
                              ;R5   = (TEMP)            R5  = (#T70A+(2*65))                  *
                              ;SP   = TEMP+2            SP  = #T70A+(2*65)+2                  *
                              ;*****************************************************************

003366  010600        T70A1   MOV     SP,R0           ;GET RESULTANT SP
003370  022706  003542        CMP     #2*65+T70A+2,SP ;SP = #T70A+(2*65)+2?
003374  001401                BEQ     .+4
```

```
003376  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000071                  C=C+1
        000066                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
003400  010701                  SCOPE
003402  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003406  012705  003414          MOV     #T71A,R5        ;LOAD R5
003412  006466                  MARK    66              ;MARK N=66

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=66                                         *
                        ;TEMP = PC+(2*N)          TEMP= #T71A+(2*66)                    *
                        ;PC   = R5                PC  = T71A                            *
                        ;SP   = TEMP              SP  = #T71A+(2*66)                    *
                        ;R5   = (TEMP)            R5  = (#T71A+(2*66))                  *
                        ;SP   = TEMP+2            SP  = #T71A+(2*66)+2                  *
                        ;*****************************************************************

003414  010500          T71A1:  MOV     SP,R0           ;GET RESULTANT SP
003416  022706  003572          CMP     #2*66+T71A+2,SP ;SP = #T71A+(2*66)+2?
003422  001401                  BEQ     .+4
003424  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000072                  C=C+1
        000067                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
003426  010701                  SCOPE
003430  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003434  012705  003442          MOV     #T72A,R5        ;LOAD R5
003440  006467                  MARK    67              ;MARK N=67

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=67                                         *
                        ;TEMP = PC+(2*N)          TEMP= #T72A+(2*67)                    *
                        ;PC   = R5                PC  = T72A                            *
                        ;SP   = TEMP              SP  = #T72A+(2*67)                    *
                        ;R5   = (TEMP)            R5  = (#T72A+(2*67))                  *
                        ;SP   = TEMP+2            SP  = #T72A+(2*67)+2                  *
                        ;*****************************************************************

003442  010500          T72A1:  MOV     SP,R0           ;GET RESULTANT SP
003444  022706  003622          CMP     #2*67+T72A+2,SP ;SP = #T72A+(2*67)+2?
003450  001401                  BEQ     .+4
003452  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000073                  C=C+1
        000070                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
003454  010701                  SCOPE
003456  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
```



```
003462  012705  003470          MOV     #T73A,R5        ;LOAD R5
003466  006470                  MARK    70              ;MARK N=70

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=70                                         *
                        ;TEMP = PC+(2*N)          TEMP= #T73A+(2*70)                    *
                        ;PC   = R5                PC  = T73A                            *
                        ;SP   = TEMP              SP  = #T73A+(2*70)                    *
                        ;R5   = (TEMP)            R5  = (#T73A+(2*70))                  *
                        ;SP   = TEMP+2            SP  = #T73A+(2*70)+2                  *
                        ;*****************************************************************

003470  010600          T73A1:  MOV     SP,R0           ;GET RESULTANT SP
003472  022706  003652          CMP     #2*70+T73A+2,SP ;SP = #T73A+(2*70)+2?
003476  001401                  BEQ     .+4
003500  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000074                  C=C+1
        000071                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
003502  010701                  SCOPE
003504  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003510  012705  003516          MOV     #T74A,R5        ;LOAD R5
003514  006471                  MARK    71              ;MARK N=71

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=71                                         *
                        ;TEMP = PC+(2*N)          TEMP= #T74A+(2*71)                    *
                        ;PC   = R5                PC  = T74A                            *
                        ;SP   = TEMP              SP  = #T74A+(2*71)                    *
                        ;R5   = (TEMP)            R5  = (#T74A+(2*71))                  *
                        ;SP   = TEMP+2            SP  = #T74A+(2*71)+2                  *
                        ;*****************************************************************

003516  010600          T74A1:  MOV     SP,R0           ;GET RESULTANT SP
003520  022706  003672          CMP     #2*71+T74A+2,SP ;SP = #T74A+(2*71)+2?
003524  001401                  BEQ     .+4
003526  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000075                  C=C+1
        000072                  N=N+1
                        ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                        ;INSTRUCTION FOR ALL VALUES OF N (0-77)
003530  010701                  SCOPE
003532  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003536  012705  003544          MOV     #T75A,R5        ;LOAD R5
003542  006472                  MARK    72              ;MARK N=72

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=72                                         *
                        ;TEMP = PC+(2*N)          TEMP= #T75A+(2*72)                    *
                        ;PC   = R5                PC  = T75A                            *
                        ;SP   = TEMP              SP  = #T75A+(2*72)                    *
```

```
                                ;R5   = (TEMP)          R5   = (#T75A+(2*72))
                                ;SP   = TEMP+2          SP   = #T75A+(2*72)+2
                                ;*****************************************************************

003544  010600                  T75A1   MOV     SP,R0           ;GET RESULTANT SP
003546  022706  003732                  CMP     #2*72+T75A+2,SP ;SP = #T75A+(2*72)+2?
003552  001401                          BEQ     .+4
003554  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

        000076                  C=C+1
        000073                  M=M+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
                                        SCOPE
003556  010701                          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003560  012706  000500                  MOV     #T76A,R5        ;LOAD R5
003564  012705  003572                  MARK    73              ;MARK N=73
003570  006473

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=73
                                ;TEMP = PC+(2*N)         TEMP= #T76A+(2*73)
                                ;PC   = R5               PC   = T76A
                                ;SP   = TEMP             SP   = #T76A+(2*73)
                                ;R5   = (TEMP)           R5   = (#T76A+(2*73))
                                ;SP   = TEMP+2           SP   = #T76A+(2*73)+2
                                ;*****************************************************************

003572  010600                  T76A1   MOV     SP,R0           ;GET RESULTANT SP
003574  022706  003762                  CMP     #2*73+T76A+2,SP ;SP = #T76A+(2*73)+2?
003600  001401                          BEQ     .+4
003602  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

        000077                  C=C+1
        000074                  M=M+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
                                        SCOPE
003604  010701                          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003606  012706  000500                  MOV     #T77A,R5        ;LOAD R5
003612  012705  003620                  MARK    74              ;MARK N=74
003616  006474

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=74
                                ;TEMP = PC+(2*N)         TEMP= #T77A+(2*74)
                                ;PC   = R5               PC   = T77A
                                ;SP   = TEMP             SP   = #T77A+(2*74)
                                ;R5   = (TEMP)           R5   = (#T77A+(2*74))
                                ;SP   = TEMP+2           SP   = #T77A+(2*74)+2
                                ;*****************************************************************

003620  010600                  T77A1   MOV     SP,R0           ;GET RESULTANT SP
003622  022706  004012                  CMP     #2*74+T77A+2,SP ;SP = #T77A+(2*74)+2?
003626  001401                          BEQ     .+4
003630  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP
```



```
        000100                  C=C+1
        000075                  M=M+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
003632  010701                          SCOPE
003634  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003640  012705  003646                  MOV     #T100A,R5       ;LOAD R5
003644  006475                          MARK    75              ;MARK N=75

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=75
                                ;TEMP = PC+(2*N)         TEMP= #T100A+(2*75)
                                ;PC   = R5               PC   = T100A
                                ;SP   = TEMP             SP   = #T100A+(2*75)
                                ;R5   = (TEMP)           R5   = (#T100A+(2*75))
                                ;SP   = TEMP+2           SP   = #T100A+(2*75)+2
                                ;*****************************************************************

003646  010600                  T100A1  MOV     SP,R0           ;GET RESULTANT SP
003650  022706  004042                  CMP     #2*75+T100A+2,SP        ;SP = #T100A+(2*75)+2?
003654  001401                          BEQ     .+4
003656  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

        000101                  C=C+1
        000076                  M=M+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
003660  010701                          SCOPE
003662  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003666  012705  003674                  MOV     #T101A,R5       ;LOAD R5
003672  006476                          MARK    76              ;MARK N=76

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=76
                                ;TEMP = PC+(2*N)         TEMP= #T101A+(2*76)
                                ;PC   = R5               PC   = T101A
                                ;SP   = TEMP             SP   = #T101A+(2*76)
                                ;R5   = (TEMP)           R5   = (#T101A+(2*76))
                                ;SP   = TEMP+2           SP   = #T101A+(2*76)+2
                                ;*****************************************************************

003674  010600                  T101A1  MOV     SP,R0           ;GET RESULTANT SP
003676  022706  004072                  CMP     #2*76+T101A+2,SP        ;SP = #T101A+(2*76)+2?
003702  001401                          BEQ     .+4
003704  000000                          HLT                     ;ERROR INCORRECT RESULTANT SP

        000102                  C=C+1
        000077                  M=M+1
                                ;TEST THAT THE STACK POINTER CONTAINS THE PROPER VALUE AFTER THE MARK
                                ;INSTRUCTION FOR ALL VALUES OF N (0-77)
003706  010701                          SCOPE
003710  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003714  012705  003722                  MOV     #T102A,R5       ;LOAD R5
```

```
003720  006477                  MARK    77              ;MARK N=77

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=77                                          *
                        ;TEMP = PC+(2*N)        TEMP= #T102A+(2*77)                     *
                        ;PC   = R5              PC  = #T102A                            *
                        ;SP   = TEMP            SP  = #T102A+(2*77)                     *
                        ;R5   = (TEMP)          R5  = (#T102A+(2*77))                   *
                        ;SP   = TEMP+2          SP  = #T102A+(2*77)+2                   *
                        ;*****************************************************************

003722  010600          T102A:  MOV     SP,R0           ;GET RESULTANT SP
003724  027706  004122          CMP     #2*77+T102A+2,SP        ;SP = #T102A+(2*77)+2?
003730  001401                  BEQ     .+4
003732  000000                  HLT                     ;ERROR INCORRECT RESULTANT SP

        000103                  C=C+1
        000100                  H=H+1
        000000          H=0
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
003734  010701                  SCOPE
003736  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003742  012705  003750          MOV     #T103A,R5       ;LOAD R5
003746  006400                  MARK    0               ;MARK N=0

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=0                                           *
                        ;TEMP = PC+(2*0)        TEMP = #T103A+(2*0)                     *
                        ;PC   = R5              PC   = #T103A                           *
                        ;SP   = TEMP            SP   = #T103A+(2*0)                     *
                        ;R5   = (TEMP)          R5   = (#T103A+(2*0))                   *
                        ;SP   = TEMP+2          SP   = #T103A+(2*0)+2                   *
                        ;*****************************************************************

003750  000240          T103A:  NOP
003752  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
003754  023705  003750          CMP     @#2*0+T103A,R5  ;R5 = (#T103A+(2*0))?
003760  001401                  BEQ     .+4
003762  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000104                  C=C+1
        000001                  H=H+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
003764  010701                  SCOPE
003766  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
003772  012705  004000          MOV     #T104A,R5       ;LOAD R5
003776  006401                  MARK    1               ;MARK N=1

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=1                                           *
                        ;TEMP = PC+(2*1)        TEMP = #T104A+(2*1)                     *
                        ;PC   = R5              PC   = #T104A                           *
```



```
                        ;SP   = TEMP            SP   = #T104A+(2*1)                     *
                        ;R5   = (TEMP)          R5   = (#T104A+(2*1))                   *
                        ;SP   = TEMP+2          SP   = #T104A+(2*1)+2                   *
                        ;*****************************************************************

004000  000240          T104A:  NOP
004002  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004004  023705  004002          CMP     @#2*1+T104A,R5  ;R5 = (#T104A+(2*1))?
004010  001401                  BEQ     .+4
004012  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000105                  C=C+1
        000002                  H=H+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004014  010701                  SCOPE
004016  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004022  012705  004038          MOV     #T105A,R5       ;LOAD R5
004026  006402                  MARK    2               ;MARK N=2

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=2                                           *
                        ;TEMP = PC+(2*2)        TEMP = #T105A+(2*2)                     *
                        ;PC   = R5              PC   = #T105A                           *
                        ;SP   = TEMP            SP   = #T105A+(2*2)                     *
                        ;R5   = (TEMP)          R5   = (#T105A+(2*2))                   *
                        ;SP   = TEMP+2          SP   = #T105A+(2*2)+2                   *
                        ;*****************************************************************

004030  000240          T105A:  NOP
004032  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004034  023705  004034          CMP     @#2*2+T105A,R5  ;R5 = (#T105A+(2*2))?
004040  001401                  BEQ     .+4
004042  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000106                  C=C+1
        000003                  H=H+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004044  010701                  SCOPE
004046  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004052  012705  004068          MOV     #T106A,R5       ;LOAD R5
004056  006403                  MARK    3               ;MARK N=3

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=3                                           *
                        ;TEMP = PC+(2*3)        TEMP = #T106A+(2*3)                     *
                        ;PC   = R5              PC   = #T106A                           *
                        ;SP   = TEMP            SP   = #T106A+(2*3)                     *
                        ;R5   = (TEMP)          R5   = (#T106A+(2*3))                   *
                        ;SP   = TEMP+2          SP   = #T106A+(2*3)+2                   *
                        ;*****************************************************************

004060  000240          T106A:  NOP
```

```
004062  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004064  023705  004066          CMP     @#2+3+T106A,R5  ;R5 = (#T106A+(2+3))?
004070  001401                  BEQ     .+4
004072  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000107                  C=C+1
        000004                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004074  010701                  SCOPE
004076  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004102  012705  004110          MOV     #T107A,R5       ;LOAD R5
004106  006404                  MARK    4               ;MARK N=4

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=4                                           *
                        ;TEMP = PC+(2*4)        TEMP = #T107A+(2*4)                     *
                        ;PC   = R5              PC   = #T107A                           *
                        ;SP   = TEMP            SP   = #T107A+(2*4)                     *
                        ;R5   = (TEMP)          R5   = (#T107A+(2*4)                    *
                        ;SP   = TEMP+2          SP   = #T107A+(2*4)+2                   *
                        ;*****************************************************************

004110  000240          T107A:  NOP
004112  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004114  023705  004120          CMP     @#2+4+T107A,R5  ;R5 = (#T107A+(2*4))?
004120  001401                  BEQ     .+4
004122  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000110                  C=C+1
        000005                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004124  010701                  SCOPE
004126  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004132  012705  004140          MOV     #T110A,R5       ;LOAD R5
004136  006405                  MARK    5               ;MARK N=5

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=5                                           *
                        ;TEMP = PC+(2*5)        TEMP = #T110A+(2*5)                     *
                        ;PC   = R5              PC   = #T110A                           *
                        ;SP   = TEMP            SP   = #T110A+(2*5)                     *
                        ;R5   = (TEMP)          R5   = (#T110A+(2*5)                    *
                        ;SP   = TEMP+2          SP   = #T110A+(2*5)+2                   *
                        ;*****************************************************************

004140  000240          T110A:  NOP
004142  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004144  023705  004152          CMP     @#2+5+T110A,R5  ;R5 = (#T110A+(2*5))?
004150  001401                  BEQ     .+4
004152  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000111                  C=C+1
```



```
        000006                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004154  010701                  SCOPE
004156  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004162  012705  004170          MOV     #T111A,R5       ;LOAD R5
004166  006406                  MARK    6               ;MARK N=6

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=6                                           *
                        ;TEMP = PC+(2*6)        TEMP = #T111A+(2*6)                     *
                        ;PC   = R5              PC   = #T111A                           *
                        ;SP   = TEMP            SP   = #T111A+(2*6)                     *
                        ;R5   = (TEMP)          R5   = (#T111A+(2*6)                    *
                        ;SP   = TEMP+2          SP   = #T111A+(2*6)+2                   *
                        ;*****************************************************************

004170  000240          T111A:  NOP
004172  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004174  023705  004204          CMP     @#2+6+T111A,R5  ;R5 = (#T111A+(2*6))?
004200  001401                  BEQ     .+4
004202  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000112                  C=C+1
        000007                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004204  010701                  SCOPE
004206  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004212  012705  004220          MOV     #T112A,R5       ;LOAD R5
004216  006407                  MARK    7               ;MARK N=7

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=7                                           *
                        ;TEMP = PC+(2*7)        TEMP = #T112A+(2*7)                     *
                        ;PC   = R5              PC   = #T112A                           *
                        ;SP   = TEMP            SP   = #T112A+(2*7)                     *
                        ;R5   = (TEMP)          R5   = (#T112A+(2*7)                    *
                        ;SP   = TEMP+2          SP   = #T112A+(2*7)+2                   *
                        ;*****************************************************************

004220  000240          T112A:  NOP
004222  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004224  023705  004236          CMP     @#2+7+T112A,R5  ;R5 = (#T112A+(2*7))?
004230  001401                  BEQ     .+4
004232  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000113                  C=C+1
        000010                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004234  010701                  SCOPE
004236  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004242  012705  004250          MOV     #T113A,R5       ;LOAD R5
```

```
004246  006410                  MARK    10              ;MARK N=10

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=10                                          *
                        ;TEMP = PC+(2*10)       TEMP = #T113A+(2*10)                    *
                        ;PC   = R5              PC   = #T113A                           *
                        ;SP   = TEMP            SP   = #T113A+(2*10)                    *
                        ;R5   = (TEMP)          R5   = (#T113A+(2*10)                   *
                        ;SP   = TEMP+2          SP   = #T113A+(2*10)+2                  *
                        ;*****************************************************************

004250  000240          T113A:  NOP
004252  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004254  023705  004270          CMP     @#2*10+T113A,R5 ;R5 = (#T113A+(2*10))?
004260  001401                  BEQ     .+4
004262  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000114                  C=C+1
        000011                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004264  010701                  SCOPE
004266  012706  001500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004272  012705  004300          MOV     #T114A,R5       ;LOAD R5
004276  006411                  MARK    11              ;MARK N=11

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=11                                          *
                        ;TEMP = PC+(2*11)       TEMP = #T114A+(2*11)                    *
                        ;PC   = R5              PC   = #T114A                           *
                        ;SP   = TEMP            SP   = #T114A+(2*11)                    *
                        ;R5   = (TEMP)          R5   = (#T114A+(2*11)                   *
                        ;SP   = TEMP+2          SP   = #T114A+(2*11)+2                  *
                        ;*****************************************************************

004300  000240          T114A:  NOP
004302  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004304  023705  004322          CMP     @#2*11+T114A,R5 ;R5 = (#T114A+(2*11))?
004310  001401                  BEQ     .+4
004312  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000115                  C=C+1
        000012                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004314  010701                  SCOPE
004316  012706  001500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004322  012705  004330          MOV     #T115A,R5       ;LOAD R5
004326  006412                  MARK    12              ;MARK N=12

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=12                                          *
                        ;TEMP = PC+(2*12)       TEMP = #T115A+(2*12)                    *
                        ;PC   = R5              PC   = #T115A                           *
```



```
                        ;SP   = TEMP            SP   = #T115A+(2*12)                    *
                        ;R5   = (TEMP)          R5   = (#T115A+(2*12)                   *
                        ;SP   = TEMP+2          SP   = #T115A+(2*12)+2                  *
                        ;*****************************************************************

004330  000240          T115A:  NOP
004332  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004334  023705  004354          CMP     @#2*12+T115A,R5 ;R5 = (#T115A+(2*12))?
004340  001401                  BEQ     .+4
004342  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000116                  C=C+1
        000013                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004344  010701                  SCOPE
004346  012706  001500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004352  012705  004360          MOV     #T116A,R5       ;LOAD R5
004356  006413                  MARK    13              ;MARK N=13

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=13                                          *
                        ;TEMP = PC+(2*13)       TEMP = #T116A+(2*13)                    *
                        ;PC   = R5              PC   = #T116A                           *
                        ;SP   = TEMP            SP   = #T116A+(2*13)                    *
                        ;R5   = (TEMP)          R5   = (#T116A+(2*13)                   *
                        ;SP   = TEMP+2          SP   = #T116A+(2*13)+2                  *
                        ;*****************************************************************

004360  000240          T116A:  NOP
004362  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004364  023705  004406          CMP     @#2*13+T116A,R5 ;R5 = (#T116A+(2*13))?
004370  001401                  BEQ     .+4
004372  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000117                  C=C+1
        000014                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004374  010701                  SCOPE
004376  012706  001500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004402  012705  004410          MOV     #T117A,R5       ;LOAD R5
004406  006414                  MARK    14              ;MARK N=14

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=14                                          *
                        ;TEMP = PC+(2*14)       TEMP = #T117A+(2*14)                    *
                        ;PC   = R5              PC   = #T117A                           *
                        ;SP   = TEMP            SP   = #T117A+(2*14)                    *
                        ;R5   = (TEMP)          R5   = (#T117A+(2*14)                   *
                        ;SP   = TEMP+2          SP   = #T117A+(2*14)+2                  *
                        ;*****************************************************************

004410  000240          T117A:  NOP
```

```
004412  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004414  023705  004440          CMP     @#2+14+T117A,R5 ;R5 = (#T117A+(2+14))?
004420  001401                  BEQ     .+4
004422  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000128                  C=C+1
        000015                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004424  010701                  SCOPE
004426  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004432  012705  004440          MOV     #T120A,R5       ;LOAD R5
004436  006415                  MARK    15              ;MARK N=15

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=15                                          *
                        ;TEMP = PC+(2+15)       TEMP = #T120A+(2+15)                    *
                        ;PC   = R5              PC   = #T120A                           *
                        ;SP   = TEMP            SP   = #T120A+(2+15)                    *
                        ;R5   = (TEMP)          R5   = (#T120A+(2+15)                   *
                        ;SP   = TEMP+2          SP   = #T120A+(2+15)+2                  *
                        ;*****************************************************************

004440  000240          T120A:  NOP
004442  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004444  023705  004472          CMP     @#2+15+T120A,R5 ;R5 = (#T120A+(2+15))?
004450  001401                  BEQ     .+4
004452  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000121                  C=C+1
        000016                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004454  010701                  SCOPE
004456  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004462  012705  004470          MOV     #T121A,R5       ;LOAD R5
004466  006416                  MARK    16              ;MARK N=16

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=16                                          *
                        ;TEMP = PC+(2+16)       TEMP = #T121A+(2+16)                    *
                        ;PC   = R5              PC   = #T121A                           *
                        ;SP   = TEMP            SP   = #T121A+(2+16)                    *
                        ;R5   = (TEMP)          R5   = (#T121A+(2+16)                   *
                        ;SP   = TEMP+2          SP   = #T121A+(2+16)+2                  *
                        ;*****************************************************************

004470  000240          T121A:  NOP
004472  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004474  023705  004524          CMP     @#2+16+T121A,R5 ;R5 = (#T121A+(2+16))?
004500  001401                  BEQ     .+4
004502  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000122                  C=C+1
```



```
        000017                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004504  010701                  SCOPE
004506  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004512  012705  004520          MOV     #T122A,R5       ;LOAD R5
004516  006417                  MARK    17              ;MARK N=17

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=17                                          *
                        ;TEMP = PC+(2+17)       TEMP = #T122A+(2+17)                    *
                        ;PC   = R5              PC   = #T122A                           *
                        ;SP   = TEMP            SP   = #T122A+(2+17)                    *
                        ;R5   = (TEMP)          R5   = (#T122A+(2+17)                   *
                        ;SP   = TEMP+2          SP   = #T122A+(2+17)+2                  *
                        ;*****************************************************************

004520  000240          T122A:  NOP
004522  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004524  023705  004556          CMP     @#2+17+T122A,R5 ;R5 = (#T122A+(2+17))?
004530  001401                  BEQ     .+4
004532  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000123                  C=C+1
        000020                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004534  010701                  SCOPE
004536  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004542  012705  004550          MOV     #T123A,R5       ;LOAD R5
004546  006420                  MARK    20              ;MARK N=20

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=20                                          *
                        ;TEMP = PC+(2+20)       TEMP = #T123A+(2+20)                    *
                        ;PC   = R5              PC   = #T123A                           *
                        ;SP   = TEMP            SP   = #T123A+(2+20)                    *
                        ;R5   = (TEMP)          R5   = (#T123A+(2+20)                   *
                        ;SP   = TEMP+2          SP   = #T123A+(2+20)+2                  *
                        ;*****************************************************************

004550  000240          T123A:  NOP
004552  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004554  023705  004610          CMP     @#2+20+T123A,R5 ;R5 = (#T123A+(2+20))?
004560  001401                  BEQ     .+4
004562  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000124                  C=C+1
        000021                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004564  010701                  SCOPE
004566  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004572  012705  004620          MOV     #T124A,R5       ;LOAD R5
```

```
004576  006421                  MARK    21              ;MARK N=21

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=21                                         *
                        ;TEMP = PC+(2*21)       TEMP = #T124A+(2*21)                    *
                        ;PC   = R5              PC   = #T124A                           *
                        ;SP   = TEMP            SP   = #T124A+(2*21)                    *
                        ;R5   = (TEMP)          R5   = (#T124A+(2*21))                  *
                        ;SP   = TEMP+2          SP   = #T124A+(2*21)+2                  *
                        ;*****************************************************************

004600  000240          T124A:  NOP
004602  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004604  023705  004642          CMP     @#2+21+T124A,R5 ;R5 = (#T124A+(2*21))?
004610  001401                  BEQ     .+4
004612  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000125                  C=C+1
        000022                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004614  010701                  SCOPE
004616  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004622  012705  004630          MOV     #T125A,R5       ;LOAD R5
004626  006422                  MARK    22              ;MARK N=22

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=22                                         *
                        ;TEMP = PC+(2*22)       TEMP = #T125A+(2*22)                    *
                        ;PC   = R5              PC   = #T125A                           *
                        ;SP   = TEMP            SP   = #T125A+(2*22)                    *
                        ;R5   = (TEMP)          R5   = (#T125A+(2*22))                  *
                        ;SP   = TEMP+2          SP   = #T125A+(2*22)+2                  *
                        ;*****************************************************************

004630  000240          T125A:  NOP
004632  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004634  023705  004674          CMP     @#2+22+T125A,R5 ;R5 = (#T125A+(2*22))?
004640  001401                  BEQ     .+4
004642  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000126                  C=C+1
        000023                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004644  010701                  SCOPE
004646  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004652  012705  004660          MOV     #T126A,R5       ;LOAD R5
004656  006423                  MARK    23              ;MARK N=23

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=23                                         *
                        ;TEMP = PC+(2*23)       TEMP = #T126A+(2*23)                    *
                        ;PC   = R5              PC   = #T126A                           *
```



```
                        ;SP   = TEMP            SP   = #T126A+(2*23)                    *
                        ;R5   = (TEMP)          R5   = (#T126A+(2*23))                  *
                        ;SP   = TEMP+2          SP   = #T126A+(2*23)+2                  *
                        ;*****************************************************************

004660  000240          T126A:  NOP
004662  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004664  023705  004726          CMP     @#2+23+T126A,R5 ;R5 = (#T126A+(2*23))?
004670  001401                  BEQ     .+4
004672  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000127                  C=C+1
        000024                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004674  010701                  SCOPE
004676  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004702  012705  004710          MOV     #T127A,R5       ;LOAD R5
004706  006424                  MARK    24              ;MARK N=24

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=24                                         *
                        ;TEMP = PC+(2*24)       TEMP = #T127A+(2*24)                    *
                        ;PC   = R5              PC   = #T127A                           *
                        ;SP   = TEMP            SP   = #T127A+(2*24)                    *
                        ;R5   = (TEMP)          R5   = (#T127A+(2*24)                   *
                        ;SP   = TEMP+2          SP   = #T127A+(2*24)+2                  *
                        ;*****************************************************************

004710  000240          T127A:  NOP
004712  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004714  023705  004760          CMP     @#2+24+T127A,R5 ;R5 = (#T127A+(2*24))?
004720  001401                  BEQ     .+4
004722  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000130                  C=C+1
        000025                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004724  010701                  SCOPE
004726  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004732  012705  004740          MOV     #T130A,R5       ;LOAD R5
004736  006425                  MARK    25              ;MARK N=25

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=25                                         *
                        ;TEMP = PC+(2*25)       TEMP = #T130A+(2*25)                    *
                        ;PC   = R5              PC   = #T130A                           *
                        ;SP   = TEMP            SP   = #T130A+(2*25)                    *
                        ;R5   = (TEMP)          R5   = (#T130A+(2*25))                  *
                        ;SP   = TEMP+2          SP   = #T130A+(2*25)+2                  *
                        ;*****************************************************************

004740  000240          T130A:  NOP
```

```
004742  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004744  023705  005012          CMP     @#2+25+T130A,R5 ;R5 = (#T130A+(2+25))?
004750  001401                  BEQ     .+4
004752  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000131                  C=C+1
        000026                  N=N+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
004754  010701                  SCOPE
004756  012706  005000          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
004762  012705  004770          MOV     #T131A,R5       ;LOAD R5
004766  006426                  MARK    26              ;MARK N=26

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=26                                         *
                        ;TEMP = PC+(2+26)       TEMP = #T131A+(2+26)                    *
                        ;PC   = R5              PC   = #T131A                           *
                        ;SP   = TEMP            SP   = #T131A+(2+26)                    *
                        ;R5   = (TEMP)          R5   = (#T131A+(2+26)                   *
                        ;SP   = TEMP+2          SP   = #T131A+(2+26)+2                  *
                        ;*****************************************************************

004770  000240          T131A:  NOP
004772  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
004774  023705  005044          CMP     @#2+26+T131A,R5 ;R5 = (#T131A+(2+26))?
005000  001401                  BEQ     .+4
005002  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000132                  C=C+1
        000027                  N=N+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005004  010701                  SCOPE
005006  012706  005000          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005012  012705  005020          MOV     #T132A,R5       ;LOAD R5
005016  006427                  MARK    27              ;MARK N=27

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=27                                         *
                        ;TEMP = PC+(2+27)       TEMP = #T132A+(2+27)                    *
                        ;PC   = R5              PC   = #T132A                           *
                        ;SP   = TEMP            SP   = #T132A+(2+27)                    *
                        ;R5   = (TEMP)          R5   = (#T132A+(2+27)                   *
                        ;SP   = TEMP+2          SP   = #T132A+(2+27)+2                  *
                        ;*****************************************************************

005020  000240          T132A:  NOP
005022  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005024  023705  005076          CMP     @#2+27+T132A,R5 ;R5 = (#T132A+(2+27))?
005030  001401                  BEQ     .+4
005032  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000133                  C=C+1
```



```
        000030                  N=N+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005034  010701                  SCOPE
005036  012706  005000          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005042  012705  005050          MOV     #T133A,R5       ;LOAD R5
005046  006430                  MARK    30              ;MARK N=30

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=30                                         *
                        ;TEMP = PC+(2+30)       TEMP = #T133A+(2+30)                    *
                        ;PC   = R5              PC   = #T133A                           *
                        ;SP   = TEMP            SP   = #T133A+(2+30)                    *
                        ;R5   = (TEMP)          R5   = (#T133A+(2+30)                   *
                        ;SP   = TEMP+2          SP   = #T133A+(2+30)+2                  *
                        ;*****************************************************************

005050  000240          T133A:  NOP
005052  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005054  023705  005130          CMP     @#2+30+T133A,R5 ;R5 = (#T133A+(2+30))?
005060  001401                  BEQ     .+4
005062  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000134                  C=C+1
        000031                  N=N+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005064  010701                  SCOPE
005066  012706  005000          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005072  012705  005100          MOV     #T134A,R5       ;LOAD R5
005076  006431                  MARK    31              ;MARK N=31

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=31                                         *
                        ;TEMP = PC+(2+31)       TEMP = #T134A+(2+31)                    *
                        ;PC   = R5              PC   = #T134A                           *
                        ;SP   = TEMP            SP   = #T134A+(2+31)                    *
                        ;R5   = (TEMP)          R5   = (#T134A+(2+31)                   *
                        ;SP   = TEMP+2          SP   = #T134A+(2+31)+2                  *
                        ;*****************************************************************

005100  000240          T134A:  NOP
005102  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005104  023705  005162          CMP     @#2+31+T134A,R5 ;R5 = (#T134A+(2+31))?
005110  001401                  BEQ     .+4
005112  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000135                  C=C+1
        000032                  N=N+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005114  010701                  SCOPE
005116  012706  005000          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005122  012705  005130          MOV     #T135A,R5       ;LOAD R5
```

```
005126  006432                  MARK    32              ;MARK N=32

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=32                                         *
                        ;TEMP = PC+(2*32)       TEMP = #T135A+(2*32)                    *
                        ;PC   = R5              PC   = #T135A                           *
                        ;SP   = TEMP            SP   = #T135A+(2*32)                    *
                        ;R5   = (TEMP)          R5   = (#T135A+(2*32)                   *
                        ;SP   = TEMP+2          SP   = #T135A+(2*32)+2                  *
                        ;*****************************************************************

005130  000240          T135A:  NOP
005132  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005134  023705  005214          CMP     @#2+32+T135A,R5 ;R5 = (#T135A+(2*32))?
005140  001401                  BEQ     .+4
005142  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000136                  C=C+1
        000033                  N=N+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005144  010701                  SCOPE
005146  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005152  012705  005160          MOV     #T136A,R5       ;LOAD R5
005156  006433                  MARK    33              ;MARK N=33

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=33                                         *
                        ;TEMP = PC+(2*33)       TEMP = #T136A+(2*33)                    *
                        ;PC   = R5              PC   = #T136A                           *
                        ;SP   = TEMP            SP   = #T136A+(2*33)                    *
                        ;R5   = (TEMP)          R5   = (#T136A+(2*33)                   *
                        ;SP   = TEMP+2          SP   = #T136A+(2*33)+2                  *
                        ;*****************************************************************

005160  000240          T136A:  NOP
005162  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005164  023705  005246          CMP     @#2+33+T136A,R5 ;R5 = (#T136A+(2*33))?
005170  001401                  BEQ     .+4
005172  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000137                  C=C+1
        000034                  N=N+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005174  010701                  SCOPE
005176  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005202  012705  005210          MOV     #T137A,R5       ;LOAD R5
005206  006434                  MARK    34              ;MARK N=34

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=34                                         *
                        ;TEMP = PC+(2*34)       TEMP = #T137A+(2*34)                    *
                        ;PC   = R5              PC   = #T137A                           *
```



```
                        ;SP   = TEMP            SP   = #T137A+(2*34)                    *
                        ;R5   = (TEMP)          R5   = (#T137A+(2*34)                   *
                        ;SP   = TEMP+2          SP   = #T137A+(2*34)+2                  *
                        ;*****************************************************************

005210  000240          T137A:  NOP
005212  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005214  023705  005300          CMP     @#2+34+T137A,R5 ;R5 = (#T137A+(2*34))?
005220  001401                  BEQ     .+4
005222  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000140                  C=C+1
        000035                  N=N+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005224  010701                  SCOPE
005226  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005232  012705  005240          MOV     #T140A,R5       ;LOAD R5
005236  006435                  MARK    35              ;MARK N=35

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=35                                         *
                        ;TEMP = PC+(2*35)       TEMP = #T140A+(2*35)                    *
                        ;PC   = R5              PC   = #T140A                           *
                        ;SP   = TEMP            SP   = #T140A+(2*35)                    *
                        ;R5   = (TEMP)          R5   = (#T140A+(2*35)                   *
                        ;SP   = TEMP+2          SP   = #T140A+(2*35)+2                  *
                        ;*****************************************************************

005240  000240          T140A:  NOP
005242  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005244  023705  005332          CMP     @#2+35+T140A,R5 ;R5 = (#T140A+(2*35))?
005250  001401                  BEQ     .+4
005252  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000141                  C=C+1
        000036                  N=N+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005254  010701                  SCOPE
005256  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005262  012705  005270          MOV     #T141A,R5       ;LOAD R5
005266  006436                  MARK    36              ;MARK N=36

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=36                                         *
                        ;TEMP = PC+(2*36)       TEMP = #T141A+(2*36)                    *
                        ;PC   = R5              PC   = #T141A                           *
                        ;SP   = TEMP            SP   = #T141A+(2*36)                    *
                        ;R5   = (TEMP)          R5   = (#T141A+(2*36)                   *
                        ;SP   = TEMP+2          SP   = #T141A+(2*36)+2                  *
                        ;*****************************************************************

005270  000240          T141A:  NOP
```

```
005272  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005274  023705  005364          CMP     @#2+36+T141A,R5 ;R5 = (#T141A+(2+36))?
005300  001401                  BEQ     .+4
005302  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000142                  C+C+1
        000037                  M+M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005304  010701                  SCOPE
005306  012706  005500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005312  012705  005320          MOV     #T142A,R5       ;LOAD R5
005316  006437                  MARK    37              ;MARK N=37

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=37                                         *
                        ;TEMP = PC+(2+37)       TEMP = #T142A+(2+37)                    *
                        ;PC   = R5              PC   = #T142A                           *
                        ;SP   = TEMP            SP   = #T142A+(2+37)                    *
                        ;R5   = (TEMP)          R5   = (#T142A+(2+37)                   *
                        ;SP   = TEMP+2          SP   = #T142A+(2+37)+2                  *
                        ;*****************************************************************

005320  000240          T142A:  NOP
005322  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005324  023705  005416          CMP     @#2+37+T142A,R5 ;R5 = (#T142A+(2+37))?
005330  001401                  BEQ     .+4
005332  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000143                  C+C+1
        000040                  M+M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005334  010701                  SCOPE
005336  012706  005500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005342  012705  005350          MOV     #T143A,R5       ;LOAD R5
005346  006440                  MARK    40              ;MARK N=40

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=40                                         *
                        ;TEMP = PC+(2+40)       TEMP = #T143A+(2+40)                    *
                        ;PC   = R5              PC   = #T143A                           *
                        ;SP   = TEMP            SP   = #T143A+(2+40)                    *
                        ;R5   = (TEMP)          R5   = (#T143A+(2+40)                   *
                        ;SP   = TEMP+2          SP   = #T143A+(2+40)+2                  *
                        ;*****************************************************************

005350  000240          T143A:  NOP
005352  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005354  023705  005450          CMP     @#2+40+T143A,R5 ;R5 = (#T143A+(2+40))?
005360  001401                  BEQ     .+4
005362  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000144                  C+C+1
```



```
        000041                  M+M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005364  010701                  SCOPE
005366  012706  005500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005372  012705  005400          MOV     #T144A,R5       ;LOAD R5
005376  006441                  MARK    41              ;MARK N=41

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=41                                         *
                        ;TEMP = PC+(2+41)       TEMP = #T144A+(2+41)                    *
                        ;PC   = R5              PC   = #T144A                           *
                        ;SP   = TEMP            SP   = #T144A+(2+41)                    *
                        ;R5   = (TEMP)          R5   = (#T144A+(2+41)                   *
                        ;SP   = TEMP+2          SP   = #T144A+(2+41)+2                  *
                        ;*****************************************************************

005400  000240          T144A:  NOP
005402  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005404  023705  005502          CMP     @#2+41+T144A,R5 ;R5 = (#T144A+(2+41))?
005410  001401                  BEQ     .+4
005412  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000145                  C+C+1
        000042                  M+M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005414  010701                  SCOPE
005416  012706  005500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005422  012705  005430          MOV     #T145A,R5       ;LOAD R5
005426  006442                  MARK    42              ;MARK N=42

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=42                                         *
                        ;TEMP = PC+(2+42)       TEMP = #T145A+(2+42)                    *
                        ;PC   = R5              PC   = #T145A                           *
                        ;SP   = TEMP            SP   = #T145A+(2+42)                    *
                        ;R5   = (TEMP)          R5   = (#T145A+(2+42)                   *
                        ;SP   = TEMP+2          SP   = #T145A+(2+42)+2                  *
                        ;*****************************************************************

005430  000240          T145A:  NOP
005432  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005434  023705  005534          CMP     @#2+42+T145A,R5 ;R5 = (#T145A+(2+42))?
005440  001401                  BEQ     .+4
005442  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000146                  C+C+1
        000043                  M+M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005444  010701                  SCOPE
005446  012706  005500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005452  012705  005460          MOV     #T146A,R5       ;LOAD R5
```

```
005456  000443                  MARK    43              ;MARK N=43

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=43                                         *
                        ;TEMP = PC+(2*43)       TEMP = #T146A+(2*43)                    *
                        ;PC   = R5              PC   = #T146A                           *
                        ;SP   = TEMP            SP   = #T146A+(2*43)                    *
                        ;R5   = (TEMP)          R5   = (#T146A+(2*43)                   *
                        ;SP   = TEMP+2          SP   = #T146A+(2*43)+2                  *
                        ;*****************************************************************

005460  000240          T146A:  NOP
005462  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005464  023705  005566          CMP     @#2*43+T146A,R5 ;R5 = (#T146A+(2*43))?
005470  001401                  BEQ     .+4
005472  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000147                  C=C+1
        000044                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005474  010701                  SCOPE
005476  012706  005500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005502  012705  005510          MOV     #T147A,R5       ;LOAD R5
005506  000444                  MARK    44              ;MARK N=44

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=44                                         *
                        ;TEMP = PC+(2*44)       TEMP = #T147A+(2*44)                    *
                        ;PC   = R5              PC   = #T147A                           *
                        ;SP   = TEMP            SP   = #T147A+(2*44)                    *
                        ;R5   = (TEMP)          R5   = (#T147A+(2*44)                   *
                        ;SP   = TEMP+2          SP   = #T147A+(2*44)+2                  *
                        ;*****************************************************************

005510  000240          T147A:  NOP
005512  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005514  023705  005620          CMP     @#2*44+T147A,R5 ;R5 = (#T147A+(2*44))?
005520  001401                  BEQ     .+4
005522  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000150                  C=C+1
        000045                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005524  010701                  SCOPE
005526  012706  005500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005532  012705  005540          MOV     #T150A,R5       ;LOAD R5
005536  000445                  MARK    45              ;MARK N=45

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=45                                         *
                        ;TEMP = PC+(2*45)       TEMP = #T150A+(2*45)                    *
                        ;PC   = R5              PC   = #T150A                           *
```



```
                        ;SP   = TEMP            SP   = #T150A+(2*45)                    *
                        ;R5   = (TEMP)          R5   = (#T150A+(2*45)                   *
                        ;SP   = TEMP+2          SP   = #T150A+(2*45)+2                  *
                        ;*****************************************************************

005540  000240          T150A:  NOP
005542  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005544  023705  005652          CMP     @#2*45+T150A,R5 ;R5 = (#T150A+(2*45))?
005550  001401                  BEQ     .+4
005552  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000151                  C=C+1
        000046                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005554  010701                  SCOPE
005556  012706  005500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005562  012705  005570          MOV     #T151A,R5       ;LOAD R5
005566  000446                  MARK    46              ;MARK N=46

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=46                                         *
                        ;TEMP = PC+(2*46)       TEMP = #T151A+(2*46)                    *
                        ;PC   = R5              PC   = #T151A                           *
                        ;SP   = TEMP            SP   = #T151A+(2*46)                    *
                        ;R5   = (TEMP)          R5   = (#T151A+(2*46)                   *
                        ;SP   = TEMP+2          SP   = #T151A+(2*46)+2                  *
                        ;*****************************************************************

005570  000240          T151A:  NOP
005572  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005574  023705  005704          CMP     @#2*46+T151A,R5 ;R5 = (#T151A+(2*46))?
005600  001401                  BEQ     .+4
005602  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000152                  C=C+1
        000047                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005604  010701                  SCOPE
005606  012706  005500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005612  012705  005620          MOV     #T152A,R5       ;LOAD R5
005616  000447                  MARK    47              ;MARK N=47

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=47                                         *
                        ;TEMP = PC+(2*47)       TEMP = #T152A+(2*47)                    *
                        ;PC   = R5              PC   = #T152A                           *
                        ;SP   = TEMP            SP   = #T152A+(2*47)                    *
                        ;R5   = (TEMP)          R5   = (#T152A+(2*47)                   *
                        ;SP   = TEMP+2          SP   = #T152A+(2*47)+2                  *
                        ;*****************************************************************

005620  000240          T152A:  NOP
```

```
005632  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005634  023705  005736          CMP     @#2+47+T152A,R5 ;R5 = (#T152A+(2+47))?
005640  001401                  BEQ     .+4
005642  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000153                  C+C+1
        000050                  M+M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005644  010701                  SCOPE
005646  012706  005000          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005652  012705  005650          MOV     #T153A,R5       ;LOAD R5
005656  006450                  MARK    50              ;MARK N=50

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=50                                         *
                        ;TEMP + PC+(2+50)       TEMP = #T153A+(2+50)                    *
                        ;PC   + R5              PC   = #T153A                           *
                        ;SP   + TEMP            SP   = #T153A+(2+50)                    *
                        ;R5   + (TEMP)          R5   = (#T153A+(2+50))                  *
                        ;SP   + TEMP+2          SP   = #T153A+(2+50)+2                  *
                        ;*****************************************************************

005660  000240          T153A:  NOP
005662  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005664  023705  005770          CMP     @#2+50+T153A,R5 ;R5 = (#T153A+(2+50))?
005670  001401                  BEQ     .+4
005672  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000154                  C+C+1
        000051                  M+M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005674  010701                  SCOPE
005676  012706  005000          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005702  012705  005700          MOV     #T154A,R5       ;LOAD R5
005706  006451                  MARK    51              ;MARK N=51

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=51                                         *
                        ;TEMP + PC+(2+51)       TEMP = #T154A+(2+51)                    *
                        ;PC   + R5              PC   = #T154A                           *
                        ;SP   + TEMP            SP   = #T154A+(2+51)                    *
                        ;R5   + (TEMP)          R5   = (#T154A+(2+51))                  *
                        ;SP   + TEMP+2          SP   = #T154A+(2+51)+2                  *
                        ;*****************************************************************

005710  000240          T154A:  NOP
005712  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005714  023705  006022          CMP     @#2+51+T154A,R5 ;R5 = (#T154A+(2+51))?
005720  001401                  BEQ     .+4
005722  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000155                  C+C+1
```



```
        000052                  M+M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005724  010701                  SCOPE
005726  012706  005000          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005732  012705  005750          MOV     #T155A,R5       ;LOAD R5
005736  006452                  MARK    52              ;MARK N=52

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=52                                         *
                        ;TEMP + PC+(2+52)       TEMP = #T155A+(2+52)                    *
                        ;PC   + R5              PC   = #T155A                           *
                        ;SP   + TEMP            SP   = #T155A+(2+52)                    *
                        ;R5   + (TEMP)          R5   = (#T155A+(2+52))                  *
                        ;SP   + TEMP+2          SP   = #T155A+(2+52)+2                  *
                        ;*****************************************************************

005740  000240          T155A:  NOP
005742  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005744  023705  006054          CMP     @#2+52+T155A,R5 ;R5 = (#T155A+(2+52))?
005750  001401                  BEQ     .+4
005752  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000156                  C+C+1
        000053                  M+M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
005754  010701                  SCOPE
005756  012706  005000          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
005762  012705  005760          MOV     #T156A,R5       ;LOAD R5
005766  006453                  MARK    53              ;MARK N=53

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=53                                         *
                        ;TEMP + PC+(2+53)       TEMP = #T156A+(2+53)                    *
                        ;PC   + R5              PC   = #T156A                           *
                        ;SP   + TEMP            SP   = #T156A+(2+53)                    *
                        ;R5   + (TEMP)          R5   = (#T156A+(2+53))                  *
                        ;SP   + TEMP+2          SP   = #T156A+(2+53)+2                  *
                        ;*****************************************************************

005770  000240          T156A:  NOP
005772  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
005774  023705  006106          CMP     @#2+53+T156A,R5 ;R5 = (#T156A+(2+53))?
006000  001401                  BEQ     .+4
006002  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000157                  C+C+1
        000054                  M+M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
006004  010701                  SCOPE
006006  012706  005000          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
006012  012705  006010          MOV     #T157A,R5       ;LOAD R5
```

```
006006  006454                  MARK    54              ;MARK N=54

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=54                                          *
                        ;TEMP = PC+(2*54)       TEMP = #T157A+(2*54)                    *
                        ;PC   = R5              PC   = #T157A                           *
                        ;SP   = TEMP            SP   = #T157A+(2*54)                    *
                        ;R5   = (TEMP)          R5   = (#T157A+(2*54)                   *
                        ;SP   = TEMP+2          SP   = #T157A+(2*54)+2                  *
                        ;*****************************************************************

006010  000240          T157A:  NOP
006012  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
006014  023705  006140          CMP     @#2*54+T157A,R5 ;R5 = (#T157A+(2*54))?
006020  001401                  BEQ     .+4
006022  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000160                  C=C+1
        000055                  N=N+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
006024  010701                  SCOPE
006026  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
006032  012705  006040          MOV     #T160A,R5       ;LOAD R5
006036  006455                  MARK    55              ;MARK N=55

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=55                                          *
                        ;TEMP = PC+(2*55)       TEMP = #T160A+(2*55)                    *
                        ;PC   = R5              PC   = #T160A                           *
                        ;SP   = TEMP            SP   = #T160A+(2*55)                    *
                        ;R5   = (TEMP)          R5   = (#T160A+(2*55)                   *
                        ;SP   = TEMP+2          SP   = #T160A+(2*55)+2                  *
                        ;*****************************************************************

006040  000240          T160A:  NOP
006042  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
006044  023705  006172          CMP     @#2*55+T160A,R5 ;R5 = (#T160A+(2*55))?
006050  001401                  BEQ     .+4
006052  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000161                  C=C+1
        000056                  N=N+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
006054  010701                  SCOPE
006056  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
006062  012705  006070          MOV     #T161A,R5       ;LOAD R5
006066  006456                  MARK    56              ;MARK N=56

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=56                                          *
                        ;TEMP = PC+(2*56)       TEMP = #T161A+(2*56)                    *
                        ;PC   = R5              PC   = #T161A                           *
```



```
                        ;SP   = TEMP            SP   = #T161A+(2*56)                    *
                        ;R5   = (TEMP)          R5   = (#T161A+(2*56)                   *
                        ;SP   = TEMP+2          SP   = #T161A+(2*56)+2                  *
                        ;*****************************************************************

006070  000240          T161A:  NOP
006072  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
006074  023705  006224          CMP     @#2*56+T161A,R5 ;R5 = (#T161A+(2*56))?
006100  001401                  BEQ     .+4
006102  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000162                  C=C+1
        000057                  N=N+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
006104  010701                  SCOPE
006106  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
006112  012705  006120          MOV     #T162A,R5       ;LOAD R5
006116  006457                  MARK    57              ;MARK N=57

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=57                                          *
                        ;TEMP = PC+(2*57)       TEMP = #T162A+(2*57)                    *
                        ;PC   = R5              PC   = #T162A                           *
                        ;SP   = TEMP            SP   = #T162A+(2*57)                    *
                        ;R5   = (TEMP)          R5   = (#T162A+(2*57)                   *
                        ;SP   = TEMP+2          SP   = #T162A+(2*57)+2                  *
                        ;*****************************************************************

006120  000240          T162A:  NOP
006122  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
006124  023705  006256          CMP     @#2*57+T162A,R5 ;R5 = (#T162A+(2*57))?
006130  001401                  BEQ     .+4
006132  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000163                  C=C+1
        000060                  N=N+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
006134  010701                  SCOPE
006136  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
006142  012705  006150          MOV     #T163A,R5       ;LOAD R5
006146  006460                  MARK    60              ;MARK N=60

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=60                                          *
                        ;TEMP = PC+(2*60)       TEMP = #T163A+(2*60)                    *
                        ;PC   = R5              PC   = #T163A                           *
                        ;SP   = TEMP            SP   = #T163A+(2*60)                    *
                        ;R5   = (TEMP)          R5   = (#T163A+(2*60)                   *
                        ;SP   = TEMP+2          SP   = #T163A+(2*60)+2                  *
                        ;*****************************************************************

006150  000240          T163A:  NOP
```

```
        006152  010500                          MOV     R5,R0           ;GET CONTENTS OF R5
        006154  023705  006310                  CMP     @#2*60+T163A,R5 ;R5 = (#T163A+(2*60))?
        006160  001401                          BEQ     .+4
        006162  000000                          HLT                     ;R5 IMPROPERLY LOADED

                000164                          C=C+1
                000061                          M=M+1
                                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                                        ;OF N (0-77)
        006164  010701                          SCOPE
        006166  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
        006172  012705  006200                  MOV     #T164A,R5       ;LOAD R5
        006176  006461                          MARK    61              ;MARK N=61

                                        ;*************************************************************************
                                        ;FUNCTION RESULTS   N=61                                                 *
                                        ;TEMP ← PC+(2*61)          TEMP = #T164A+(2*61)                          *
                                        ;PC   ← R5                 PC   = #T164A                                 *
                                        ;SP   ← TEMP               SP   = #T164A+(2*61)                          *
                                        ;R5   ← (TEMP)             R5   = (#T164A+(2*61))                        *
                                        ;SP   ← TEMP+2             SP   = #T164A+(2*61)+2                        *
                                        ;*************************************************************************

        006200  000240                  T164A:  NOP
        006202  010500                          MOV     R5,R0           ;GET CONTENTS OF R5
        006204  023705  006342                  CMP     @#2*61+T164A,R5 ;R5 = (#T164A+(2*61))?
        006210  001401                          BEQ     .+4
        006212  000000                          HLT                     ;R5 IMPROPERLY LOADED

                000165                          C=C+1
                000062                          M=M+1
                                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                                        ;OF N (0-77)
        006214  010701                          SCOPE
        006216  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
        006222  012705  006230                  MOV     #T165A,R5       ;LOAD R5
        006226  006462                          MARK    62              ;MARK N=62

                                        ;*************************************************************************
                                        ;FUNCTION RESULTS   N=62                                                 *
                                        ;TEMP ← PC+(2*62)          TEMP = #T165A+(2*62)                          *
                                        ;PC   ← R5                 PC   = #T165A                                 *
                                        ;SP   ← TEMP               SP   = #T165A+(2*62)                          *
                                        ;R5   ← (TEMP)             R5   = (#T165A+(2*62))                        *
                                        ;SP   ← TEMP+2             SP   = #T165A+(2*62)+2                        *
                                        ;*************************************************************************

        006230  000240                  T165A:  NOP
        006232  010500                          MOV     R5,R0           ;GET CONTENTS OF R5
        006234  023705  006374                  CMP     @#2*62+T165A,R5 ;R5 = (#T165A+(2*62))?
        006240  001401                          BEQ     .+4
        006242  000000                          HLT                     ;R5 IMPROPERLY LOADED

                000166                          C=C+1
```



```
                000063                          M=M+1
                                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                                        ;OF N (0-77)
        006244  010701                          SCOPE
        006246  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
        006252  012705  006260                  MOV     #T166A,R5       ;LOAD R5
        006256  006463                          MARK    63              ;MARK N=63

                                        ;*************************************************************************
                                        ;FUNCTION RESULTS   N=63                                                 *
                                        ;TEMP ← PC+(2*63)          TEMP = #T166A+(2*63)                          *
                                        ;PC   ← R5                 PC   = #T166A                                 *
                                        ;SP   ← TEMP               SP   = #T166A+(2*63)                          *
                                        ;R5   ← (TEMP)             R5   = (#T166A+(2*63))                        *
                                        ;SP   ← TEMP+2             SP   = #T166A+(2*63)+2                        *
                                        ;*************************************************************************

        006260  000240                  T166A:  NOP
        006262  010500                          MOV     R5,R0           ;GET CONTENTS OF R5
        006264  023705  006426                  CMP     @#2*63+T166A,R5 ;R5 = (#T166A+(2*63))?
        006270  001401                          BEQ     .+4
        006272  000000                          HLT                     ;R5 IMPROPERLY LOADED

                000167                          C=C+1
                000064                          M=M+1
                                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                                        ;OF N (0-77)
        006274  010701                          SCOPE
        006276  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
        006302  012705  006310                  MOV     #T167A,R5       ;LOAD R5
        006306  006464                          MARK    64              ;MARK N=64

                                        ;*************************************************************************
                                        ;FUNCTION RESULTS   N=64                                                 *
                                        ;TEMP ← PC+(2*64)          TEMP = #T167A+(2*64)                          *
                                        ;PC   ← R5                 PC   = #T167A                                 *
                                        ;SP   ← TEMP               SP   = #T167A+(2*64)                          *
                                        ;R5   ← (TEMP)             R5   = (#T167A+(2*64))                        *
                                        ;SP   ← TEMP+2             SP   = #T167A+(2*64)+2                        *
                                        ;*************************************************************************

        006310  000240                  T167A:  NOP
        006312  010500                          MOV     R5,R0           ;GET CONTENTS OF R5
        006314  023705  006460                  CMP     @#2*64+T167A,R5 ;R5 = (#T167A+(2*64))?
        006320  001401                          BEQ     .+4
        006322  000000                          HLT                     ;R5 IMPROPERLY LOADED

                000170                          C=C+1
                000065                          M=M+1
                                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                                        ;OF N (0-77)
        006324  010701                          SCOPE
        006326  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
        006332  012705  006340                  MOV     #T170A,R5       ;LOAD R5
```

```
006336  006465                  MARK    65              ;MARK N=65

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=65                                         *
                        ;TEMP = PC+(2*65)       TEMP = #T170A+(2*65)                    *
                        ;PC   = R5              PC   = #T170A                           *
                        ;SP   = TEMP            SP   = #T170A+(2*65)                    *
                        ;R5   = (TEMP)          R5   = (#T170A+(2*65)                   *
                        ;SP   = TEMP+2          SP   = #T170A+(2*65)+2                  *
                        ;*****************************************************************

006340  000240          T170A:  NOP
006342  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
006344  027705  006512          CMP     #2*65+T170A,R5 ;R5 = (#T170A+(2*65))?
006350  001401                  BEQ     .+4
006352  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000171                  C=C+1
        000066                  N=N+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
006354  010701                  SCOPE
006356  012706  006500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
006362  012705  006370          MOV     #T171A,R5       ;LOAD R5
006366  006466                  MARK    66              ;MARK N=66

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=66                                         *
                        ;TEMP = PC+(2*66)       TEMP = #T171A+(2*66)                    *
                        ;PC   = R5              PC   = #T171A                           *
                        ;SP   = TEMP            SP   = #T171A+(2*66)                    *
                        ;R5   = (TEMP)          R5   = (#T171A+(2*66)                   *
                        ;SP   = TEMP+2          SP   = #T171A+(2*66)+2                  *
                        ;*****************************************************************

006370  000240          T171A:  NOP
006372  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
006374  027705  006544          CMP     #2*66+T171A,R5 ;R5 = (#T171A+(2*66))?
006400  001401                  BEQ     .+4
006402  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000172                  C=C+1
        000067                  N=N+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
006404  010701                  SCOPE
006406  012706  006500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
006412  012705  006420          MOV     #T172A,R5       ;LOAD R5
006416  006467                  MARK    67              ;MARK N=67

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=67                                         *
                        ;TEMP = PC+(2*67)       TEMP = #T172A+(2*67)                    *
                        ;PC   = R5              PC   = #T172A                           *
```



```
                        ;SP   = TEMP            SP   = #T172A+(2*67)                    *
                        ;R5   = (TEMP)          R5   = (#T172A+(2*67)                   *
                        ;SP   = TEMP+2          SP   = #T172A+(2*67)+2                  *
                        ;*****************************************************************

006420  000240          T172A:  NOP
006422  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
006424  027705  006576          CMP     #2*67+T172A,R5 ;R5 = (#T172A+(2*67))?
006430  001401                  BEQ     .+4
006432  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000173                  C=C+1
        000070                  N=N+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
006434  010701                  SCOPE
006436  012706  006500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
006442  012705  006450          MOV     #T173A,R5       ;LOAD R5
006446  006470                  MARK    70              ;MARK N=70

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=70                                         *
                        ;TEMP = PC+(2*70)       TEMP = #T173A+(2*70)                    *
                        ;PC   = R5              PC   = #T173A                           *
                        ;SP   = TEMP            SP   = #T173A+(2*70)                    *
                        ;R5   = (TEMP)          R5   = (#T173A+(2*70)                   *
                        ;SP   = TEMP+2          SP   = #T173A+(2*70)+2                  *
                        ;*****************************************************************

006450  000240          T173A:  NOP
006452  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
006454  027705  006630          CMP     #2*70+T173A,R5 ;R5 = (#T173A+(2*70))?
006460  001401                  BEQ     .+4
006462  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000174                  C=C+1
        000071                  N=N+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
006464  010701                  SCOPE
006466  012706  006500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
006472  012705  006500          MOV     #T174A,R5       ;LOAD R5
006476  006471                  MARK    71              ;MARK N=71

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=71                                         *
                        ;TEMP = PC+(2*71)       TEMP = #T174A+(2*71)                    *
                        ;PC   = R5              PC   = #T174A                           *
                        ;SP   = TEMP            SP   = #T174A+(2*71)                    *
                        ;R5   = (TEMP)          R5   = (#T174A+(2*71)                   *
                        ;SP   = TEMP+2          SP   = #T174A+(2*71)+2                  *
                        ;*****************************************************************

006500  000240          T174A:  NOP
```

```
006502  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
006504  023705  006662          CMP     @#2*71+T174A,R5 ;R5 = (#T174A+(2*71))?
006510  001401                  BEQ     .+4
006512  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000175                  C=C+1
        000072                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
                                SCOPE
006514  010701                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
006516  012706  000500          MOV     #T175A,R5       ;LOAD R5
006522  012705  006530          MARK    72              ;MARK N=72
006526  006472

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=72                                          *
                        ;TEMP = PC+(2*72)       TEMP = #T175A+(2*72)                    *
                        ;PC   = R5              PC   = #T175A                           *
                        ;SP   = TEMP            SP   = #T175A+(2*72)                    *
                        ;R5   = (TEMP)          R5   = (#T175A+(2*72)                   *
                        ;SP   = TEMP+2          SP   = #T175A+(2*72)+2                  *
                        ;*****************************************************************

006530  000240          T175A:  NOP
006532  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
006534  023705  006714          CMP     @#2*72+T175A,R5 ;R5 = (#T175A+(2*72))?
006540  001401                  BEQ     .+4
006542  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000176                  C=C+1
        000073                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
006544  010701                  SCOPE
006546  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
006552  012705  006560          MOV     #T176A,R5       ;LOAD R5
006556  006473                  MARK    73              ;MARK N=73

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=73                                          *
                        ;TEMP = PC+(2*73)       TEMP = #T176A+(2*73)                    *
                        ;PC   = R5              PC   = #T176A                           *
                        ;SP   = TEMP            SP   = #T176A+(2*73)                    *
                        ;R5   = (TEMP)          R5   = (#T176A+(2*73)                   *
                        ;SP   = TEMP+2          SP   = #T176A+(2*73)+2                  *
                        ;*****************************************************************

006560  000240          T176A:  NOP
006562  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
006564  023705  006746          CMP     @#2*73+T176A,R5 ;R5 = (#T176A+(2*73))?
006570  001401                  BEQ     .+4
006572  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000177                  C=C+1
```



```
        000074                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
006574  010701                  SCOPE
006576  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
006602  012705  006610          MOV     #T177A,R5       ;LOAD R5
006606  006474                  MARK    74              ;MARK N=74

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=74                                          *
                        ;TEMP = PC+(2*74)       TEMP = #T177A+(2*74)                    *
                        ;PC   = R5              PC   = #T177A                           *
                        ;SP   = TEMP            SP   = #T177A+(2*74)                    *
                        ;R5   = (TEMP)          R5   = (#T177A+(2*74)                   *
                        ;SP   = TEMP+2          SP   = #T177A+(2*74)+2                  *
                        ;*****************************************************************

006610  000240          T177A:  NOP
006612  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
006614  023705  007000          CMP     @#2*74+T177A,R5 ;R5 = (#T177A+(2*74))?
006620  001401                  BEQ     .+4
006622  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000200                  C=C+1
        000075                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
006624  010701                  SCOPE
006626  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
006632  012705  006640          MOV     #T200A,R5       ;LOAD R5
006636  006475                  MARK    75              ;MARK N=75

                        ;*****************************************************************
                        ;FUNCTION RESULTS  N=75                                          *
                        ;TEMP = PC+(2*75)       TEMP = #T200A+(2*75)                    *
                        ;PC   = R5              PC   = #T200A                           *
                        ;SP   = TEMP            SP   = #T200A+(2*75)                    *
                        ;R5   = (TEMP)          R5   = (#T200A+(2*75)                   *
                        ;SP   = TEMP+2          SP   = #T200A+(2*75)+2                  *
                        ;*****************************************************************

006640  000240          T200A:  NOP
006642  010500                  MOV     R5,R0           ;GET CONTENTS OF R5
006644  023705  007032          CMP     @#2*75+T200A,R5 ;R5 = (#T200A+(2*75))?
006650  001401                  BEQ     .+4
006652  000000                  HLT                     ;R5 IMPROPERLY LOADED

        000201                  C=C+1
        000076                  M=M+1
                        ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                        ;OF N (0-77)
006654  010701                  SCOPE
006656  012706  000500          MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
006662  012705  006670          MOV     #T201A,R5       ;LOAD R5
```

```
000666  000476                          MARK    76              ;MARK N=76

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=76                                         *
                                ;TEMP = PC+(2*76)       TEMP = #T201A+(2*76)                    *
                                ;PC   = R5              PC   = #T201A                           *
                                ;SP   = TEMP            SP   = #T201A+(2*76)                    *
                                ;R5   = (TEMP)          R5   = (#T201A+(2*76))                  *
                                ;SP   = TEMP+2          SP   = #T201A+(2*76)+2                  *
                                ;*****************************************************************

000670  000240                  T201A:  NOP
000672  010500                          MOV     R5,R0           ;GET CONTENTS OF R5
000674  023705  007064                  CMP     #2*76+T201A,R5  ;R5 = (#T201A+(2*76))?
000700  001401                          BEQ     .+4
000702  000000                          HLT                     ;R5 IMPROPERLY LOADED

        000202                          C=C+1
        000077                          N=N+1
                                ;TEST THAT R5 IS PROPERLY LOADED BY THE MARK INSTRUCTION FOR ALL VALUES
                                ;OF N (0-77)
000704  010701                          SCOPE
000706  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
000712  012705  006720                  MOV     #T202A,R5       ;LOAD R5
000716  006477                          MARK    77              ;MARK N=77

                                ;*****************************************************************
                                ;FUNCTION RESULTS  N=77                                         *
                                ;TEMP = PC+(2*77)       TEMP = #T202A+(2*77)                    *
                                ;PC   = R5              PC   = #T202A                           *
                                ;SP   = TEMP            SP   = #T202A+(2*77)                    *
                                ;R5   = (TEMP)          R5   = (#T202A+(2*77))                  *
                                ;SP   = TEMP+2          SP   = #T202A+(2*77)+2                  *
                                ;*****************************************************************

000720  000240                  T202A:  NOP
000722  010500                          MOV     R5,R0           ;GET CONTENTS OF R5
000724  023705  007116                  CMP     #2*77+T202A,R5  ;R5 = (#T202A+(2*77))?
000730  001401                          BEQ     .+4
000732  000000                          HLT                     ;R5 IMPROPERLY LOADED

        000203                          C=C+1
        003100                          N=N+1

                                ;TEST THAT MARK RETURNS FROM SUBROUTINE PROPERLY
000734  010701                  T203:   SCOPE
000736  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
000742  012705  000007                  MOV     #7,R5           ;LOAD R5
000746  010546                          MOV     R5,-(SP)        ;SAVE CURRENT R5
000750  012746  006400                  MOV     #MARK0,-(SP)    ;PUSH MARK ON THE STACK (N=0)
000754  010605                          MOV     SP,R5           ;SET UP PARAMETER POINTER
000756  004767  000002                  JSR     7,T203A         ;GO TO SUBROUTINE
000762  000401                          BR      T204            ;EXIT TEST
000764  000205                  T203A:  RTS     5               ;EXIT SUBROUTINE
```



```
                                ;TEST THAT SUBROUTINE LEAVES STACK POINTER POSITIONED PROPERLY
000766  010701                  T204:   SCOPE
000770  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
000774  012705  000007                  MOV     #7,R5           ;LOAD R5
001000  010546                          MOV     R5,-(SP)        ;SAVE CURRENT R5
001002  012746  006400                  MOV     #MARK0,-(SP)    ;PUSH MARK ON THE STACK (N=0)
001006  010605                          MOV     SP,R5           ;SET UP PARAMETER POINTER
001010  004767  000002                  JSR     7,T204A         ;GO TO SUBROUTINE
001014  000401                          BR      T204B
001016  000205                  T204A:  RTS     5               ;EXIT SUBROUTINE
001020  010600                  T204B:  MOV     SP,R0           ;GET STACK POINTER
001022  022700  000500                  CMP     #STKPTR,R0      ;STACK POINTER AT PROPER POSITION?
001026  001401                          BEQ     T205
001030  000000                          HLT                     ;ERROR INCORRECT STACK POINTER

                                ;TEST THAT SUBROUTINE LEAVES R5 UNDISTURBED
001032  010701                  T205:   SCOPE
001034  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001040  012705  000007                  MOV     #7,R5           ;LOAD R5
001044  010546                          MOV     R5,-(SP)        ;SAVE CURRENT R5
001046  012746  006400                  MOV     #MARK0,-(SP)    ;PUSH MARK ON THE STACK (N=0)
001052  010605                          MOV     SP,R5           ;SET UP PARAMETER POINTER
001054  004767  000002                  JSR     7,T205A         ;GO TO SUBROUTINE
001060  000401                          BR      T205B
001062  000205                  T205A:  RTS     5               ;EXIT SUBROUTINE
001064  010500                  T205B:  MOV     R5,R0           ;GET R5
001066  022700  000007                  CMP     #7,R0           ;WAS R5 = TO #7?
001072  001401                          BEQ     T206            ;GO TO NEXT TEST
001074  000000                          HLT                     ;ERROR CONTEN[illegible] ARE INCORRECT

                                ;TEST THAT SUBROUTINE CAN 'PICK' A PARAMETER
001076  010701                  T206:   SCOPE
001100  000000                          CLR     R0
001102  012705  000007                  MOV     #7,R5           ;LOAD R5
001106  010546                          MOV     R5,-(SP)        ;SAVE CURRENT R5
001110  012746  000001                  MOV     #1,-(SP)        ;PUSH A PAREMETER ON THE STACK
001114  012746  006401                  MOV     #MARK1,-(SP)    ;PUSH MARK ON THE STACK (N=1)
001120  010605                          MOV     SP,R5           ;SET UP PARAMETER POINTER
001122  004767  000002                  JSR     7,T206A         ;GO TO SUBROUTINE
001126  000403                          BR      T206B
001130  016500  100000          T206A:  MOV     2(5),R0         ;GET THE PARAMETER
001134  000205                          RTS     5               ;EXIT SUBROUTINE
001136  022700  000001          T206B:  CMP     #1,R0           ;DID SUBROUTINE GET THE PARAMETER
001142  001401                          BEQ     T207
001144  000000                          HLT

                                ;TEST THAT MARK DOES NOT CHANGE THE PSW
001146  010701                  T207:   SCOPE
001150  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001154  012705  007166                  MOV     #T207A,R5
001160  005067  170612                  CLR     PSW
001164  006400                          MARK    0
001166  016700  170604          T207A:  MOV     PSW,R0  ;GET PSW
```

```
007172  001401                          BEQ     T210
007174  000000                          HLT

                                ;TEST THAT MARK DOES NOT CHANGE THE PSW
007176  010701                  T210:   SCOPE
007200  012706  000500                  MOV     #STKPTR,SP      ;INITIALIZE TH ESTACK POINTER
007204  012705  007220                  MOV     #T210A,R5
007210  012767  000357  170560          MOV     #357,PSW
007216  006400                          MARK    0
007220  013700  177776          T210A:  MOV     @#PSW,R0        ;GET PSW RESULTS
007224  012706  000500                  MOV     #STKPTR,SP      ;RESET STACK PTR
007230  022700  000357                  CMP     #357,R0         ;DID PSW CHANGE
007234  001401                          BEQ     .+4
007236  000000                          HLT


007240  005267  171534          END:    INC     ICNT            ;INCREMENT PASS COUNT
007244  026727  171530  070000          CMP     ICNT,#70000
007252  001402                          BEQ     DONE            ;GO TO DONE IF 1000 PASSES COMPLETED
007254  000167  171526                  JMP     BEGIN           ;RESTART TEST
007260  112767  000007  170300  DONE:   MOVB    #7,TPBUF        ;RING BELL
007266  105767  170272                  TSTB    TPCSR
007272  100375                          BPL     .-4
007274  013702  000042                  MOV     @#42,X2         ;GET DECTAPE MONITOR RETURN ADDRESS
007300  001404                          BEQ     DONE1           ;DO NOT RETURN IF (42)=0
007302  004712                          JSR     7,(2)           ;RETURN TO DECTAPE MONITOR
007304  000240                          NOP                     ;ACT11
007306  000240                          NOP                     ;OVERLAY
007310  000240                          NOP                     ;AREA
007312  000167  171464          DONE1:  JMP     START
        000001                          .END
```



```
BEGIN    001006      C      = 000207      DISPLA = 177570      DONE     007260
DONE1    007312      END      007240      HLT    = 000000      ICNT     071000
H      = 000100      MARK0  = 006400      MARK1  = 006401      MARK2  = 006402
PC     =%000007      PSW    = 177776      R0     =%000000      R1     =%000001
R2     =%000002      R3     =%000003      R4     =%000004      R5     =%000005
SCOPE  = 010701      SP     =%000006      START    001002      STKPTR = 000500
SWR    = 177570      TPBUF  = 177566      TPCSR  = 177564      T0A      001054
T1A      001072      T10A     001326      T100A    003646      T101A    003674
T102A    003722      T103A    003750      T104A    004000      T105A    004030
T106A    004060      T107A    004110      T11A     001354      T110A    004140
T111A    004170      T112A    004220      T113A    004250      T114A    004300
T115A    004330      T116A    004360      T117A    004410      T12A     001402
T120A    004440      T121A    004470      T122A    004520      T123A    004550
T124A    004600      T125A    004630      T126A    004660      T127A    004710
T13A     001430      T130A    004740      T131A    004770      T132A    005020
T133A    005050      T134A    005100      T135A    005130      T136A    005160
T137A    005210      T14A     001456      T140A    005240      T141A    005270
T142A    005320      T143A    005350      T144A    005400      T145A    005430
T146A    005460      T147A    005510      T15A     001504      T150A    005540
T151A    005570      T152A    005620      T153A    005650      T154A    005700
T155A    005730      T156A    005760      T157A    006010      T16A     001532
T160A    006040      T161A    006070      T162A    006120      T163A    006150
T164A    006200      T165A    006230      T166A    006260      T167A    006310
T17A     001560      T170A    006340      T171A    006370      T172A    006420
T173A    006450      T174A    006500      T175A    006530      T176A    006560
T177A    006610      T2A      001120      T20A     001606      T200A    006640
T201A    006670      T202A    006720      T203     006734      T203A    006764
T204     006766      T204A    007016      T204B    007020      T205     007032
T205A    007062      T205B    007064      T206     007076      T206A    007130
T206A    007136      T207     007146      T207A    007166      T21A     001634
T210     007176      T210A    007220      T22A     001662      T23A     001710
T24A     001736      T25A     001764      T26A     002012      T27A     002040
T3A      001150      T30A     002066      T31A     002114      T32A     002142
T33A     002170      T34A     002216      T35A     002244      T36A     002272
T37A     002320      T4A      001176      T40A     002346      T41A     002374
T42A     002420      T43A     002450      T44A     002476      T45A     002524
T46A     002552      T47A     002600      T5A      001224      T50A     002626
T51A     002654      T52A     002702      T53A     002730      T54A     002754
T55A     003004      T56A     003032      T57A     003060      T6A      001252
T60A     003100      T61A     003134      T62A     003162      T63A     003210
T64A     003236      T65A     003264      T66A     003312      T67A     003340
T7A      001300      T70A     003366      T71A     003414      T72A     003442
T73A     003472      T74A     003516      T75A     003544      T76A     003572
T77A     003620      UBREAK = 177770      .      = 007316
```

ERRORS DETECTED: 0

```
                                .TITLE  MAINDEC-11-DCKBE-B PDP11/45 RTI/T TEST
                                .LIST   ME
                                .NLIST  MC,MD,SEQ
                                .ABS
                        ;TEST DCKBE- THIS IS A TEST OF THE RTT & RTI INSTRUCTIONS.
                        ;THE RTI INSTRUCTION ALLOWS A 'T' BIT TRAP IMMEDIATLY FOLLOWING THE
                        ;RTI, WHEREAS THE RTT WILL EXECUTE THE INSTRUCTION FOLLOWING THE RTT
                        ;BEFORE THE 'T' BIT TRAP IS ALLOWED. (LIKE 11/20 RTI)
                        ;NOTE: THIS TEST EXECUTES 'T' BIT TRAPS.
                        ;THE TEST TESTS RTT/RTI IN ALL THREE STATES (KERNEL,SUPERVISOR,AND USER).

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

000000                          HLT=HALT
010701                          SCOPE=010701            ;MOVE PC TO R1
177570                          SWR=177570              ;ADDRESS OF CONSOL SWITCH REGISTER
177570                          DISPLAY=177570          ;ADDRESS OF CONSOL DISPLAY REGISTER
177776                          PSW=177776              ;ADDRESS OF PROCESSER STATUS WORD
177770                          UBRLAK=177770           ;ADDRESS OF PDP11/45 MICRO BREAK REGISTER
177764                          TPCSR=177564
177766                          TPBUF=177566
                        ;*********************INITIAL STACK POINTER=0500**************************
000500                  STKPTR=0500                     ;INITIAL STACK SETTING


                        ;GENERAL REGISTERS
000000                          R0=%0
000001                          R1=%1
000002                          R2=%2
000003                          R3=%3
000004                          R4=%4
000005                          R5=%5
000006                          SP=%6
000006                          KSP=%6                  ;KERNEL'S STACK POINTER
000006                          SSP=%6                  ;SUPERVISORY STACK POINTER
000006                          USP=%6                  ;USER'S STACK POINTER
000007                          PC=%7
000020                          TBIT=20
000004                          ERRVEC=4                ;ADDRESS OF ERROR TRAP VECTOR
000010                          RESVEC=10               ;ADDRESS OF RESERVED INST. TRAP VECTOR
000014                          TBITVEC=14              ;ADDRESS OF 'T' BIT TRAP VECTOR
                        ;PROCESSER STATUS BITS
000000                          KM=0                    ;KERNEL MODE
040000                          SM=040000               ;SUPERVISORY MODE
140000                          UM=140000               ;USER MODE
000000                          PKM=0                   ;PREVIOUS KERNEL MODE
010000                          PSM=10000               ;PREVIOUS SUPERVISORY MODE
030000                          PUM=30000               ;PREVIOUS USER MODE
004000                          REG=4000                ;REGISTER SET BIT
000200                          PRTY4=200               ;PRIORITY LEVEL 4
```



```
000340                          PRTY7=340               ;PRIORITY LEVEL 7
000000                          PRTY0=0                 ;PRIORITY LEVEL 0

                        ;STACK POINTERS
000770                          KPTR=770                ;KERNELS INITIAL STACK POINTER
000750                          SPTR=750                ;SUPERVISORS INITIAL STACK POINTER
000730                          UPTR=730                ;USER INITIAL STACK SETTING
000000                          .=0
                                .NLIST  MC,MD,SEQ
000000  000002                  .+2
000002  000000                  HALT
000004  000006                  .+2
000006  000000                  HALT
000010  000012                  .+2
000012  000000                  HALT
000014  000016                  .+2
000016  000000                  HALT
000020  000022                  .+2
000022  000000                  HALT
000024  000026                  .+2
000026  000000                  HALT
000030  000032                  .+2
000032  000000                  HALT
000034  000036                  .+2
000036  000000                  HALT
000040  000042                  .+2
000042  000000                  HALT
000044  000046                  .+2
000046  000000                  HALT
000050  000052                  .+2
000052  000000                  HALT
000054  000056                  .+2
000056  000000                  HALT
000060  000062                  .+2
000062  000000                  HALT
000064  000066                  .+2
000066  000000                  HALT
000070  000072                  .+2
000072  000000                  HALT
000074  000076                  .+2
000076  000000                  HALT
000100  000102                  .+2
000102  000000                  HALT
000104  000106                  .+2
000106  000000                  HALT
000110  000112                  .+2
000112  000000                  HALT
000114  000116                  .+2
000116  000000                  HALT
000120  000122                  .+2
000122  000000                  HALT
000124  000126                  .+2
000126  000000                  HALT
000130  000132                  .+2
```

```
001132  000000                  HALT
001134  001136                  .+2
001136  000000                  HALT
001140  001142                  .+2
001142  000000                  HALT
001144  001146                  .+2
001146  000000                  HALT
001150  001152                  .+2
001152  000000                  HALT
001154  001156                  .+2
001156  000000                  HALT
001160  001162                  .+2
001162  000000                  HALT
001164  001166                  .+2
001166  000000                  HALT
001170  001172                  .+2
001172  000000                  HALT
001174  001176                  .+2
001176  000000                  HALT
001200  001202                  .+2
001202  000000                  HALT
001204  001206                  .+2
001206  000000                  HALT
001210  001212                  .+2
001212  000000                  HALT
001214  001216                  .+2
001216  000000                  HALT
001220  001222                  .+2
001222  000000                  HALT
001224  001226                  .+2
001226  000000                  HALT
001230  001232                  .+2
001232  000000                  HALT
001234  001236                  .+2
001236  000000                  HALT
001240  001242                  .+2
001242  000000                  HALT
001244  001246                  .+2
001246  000000                  HALT
001250  001252                  .+2
001252  000000                  HALT
001254  001256                  .+2
001256  000000                  HALT
001260  001262                  .+2
001262  000000                  HALT
001264  001266                  .+2
001266  000000                  HALT
001270  001272                  .+2
001272  000000                  HALT
001274  001276                  .+2
001276  000000                  HALT
001300  001302                  .+2
001302  000000                  HALT
001304  001306                  .+2
```



```
001306  000000                  HALT
001310  001312                  .+2
001312  000000                  HALT
001314  001316                  .+2
001316  000000                  HALT
001320  001322                  .+2
001322  000000                  HALT
001324  001326                  .+2
001326  000000                  HALT
001330  001332                  .+2
001332  000000                  HALT
001334  001336                  .+2
001336  000000                  HALT
001340  001342                  .+2
001342  000000                  HALT
001344  001346                  .+2
001346  000000                  HALT
001350  001352                  .+2
001352  000000                  HALT
001354  001356                  .+2
001356  000000                  HALT
001360  001362                  .+2
001362  000000                  HALT
001364  001366                  .+2
001366  000000                  HALT
001370  001372                  .+2
001372  000000                  HALT
001374  001376                  .+2
001376  000000                  HALT
                                .LIST   ME
        001200                  .=200
001200  001167  000600          JMP     @START          ;GO START TEST
        001000                  .=1000
001000  000000          ;CNT:   0                       ;CONTAINS PASS COUNT
001002  000000          TEMP:   0
001004  005067  177770  START:  CLR     CNT             ;CLEAR PASS COUNT
001010  012706  001500  REGIN:  MOV     #STKPTR,SP      ;INITIALIZE THE STACK POINTER
001014  016737  177760  177570  MOV     CNT,@#DISPLAY   ;DISPLAY PASS COUNT
001022  012737  000400  177570  MOV     #400,@#SWR      ;LOAD PDP11/45 MICRO BREAK REGISTER?
001030  001443                  BEQ     .+10
001032  113737  177570  177770  MOVB    @#SWR,@#UBREAK  ;LOAD MICRO BREAK REG WITH SR0-7

                        ;TEST THAT RTT POPS THE TOP WORD OFF THE STACK INTO THE PC.
001040  010701          T0:     SCOPE
001042  005046                  CLR     -(SP)           ;PUSH A PS6 ONTO THE STACK
001044  012746  001056          MOV     #T0A,-(SP)      ;PUSH A PC ONTO THE STACK
001050  000006                  RTT
001052  000000                  HLT                     ;RTT DID NOT POP TOP WORD OFF THE
                                                        ;STACK INTO THE PC
001054  000755                  BR      BEGIN           ;RESTART TEST IF ERROR
001056  000400          T0A:    BR      T1              ;GO TO THE NEXT TEST

                        ;TEST THAT RTT POPS TWO WORDS OFF THE STACK
001060  010701          T1:     SCOPE
```

```
001076  010600                  T1A:    MOV     SP,R0           ;GET THE STACK POINTER
001100  022700  000500                  CMP     #STKPTR,R0      ;DID RTT POP TWO WORDS
001104  001401                          BEQ     T2
001106  000000                          HLT                     ;ERROR: RTT DID NOT POP TWO WORDS

                                ;TEST THAT RTT POPS THE TWO WORDS OFF THE STACK AND INTO THE
                                ;PC AND PSW RESPECTIVELY.
001110  010701                  T2:     SCOPE
001112  012706  000500                  MOV     #STKPTR,SP      ;SET STACK PTR
001116  005046                          CLR     -(SP)           ;PUSH STATUS ON THE STACK
001120  012746  001134                  MOV     #T2A,-(SP)      ;PUSH PC ON THE STACK
001124  012767  000357  176644          MOV     #357,PSW        ;PRESET STATUS
001132  000006                          RTT
001134  016700  176636          T2A:    MOV     PSW,R0          ;GET STATUS AFTER RTT
001140  005700                          TST     R0              ;CHECK THAT RTT LOADED
001142  001401                          BEQ     T3              ;NEW STATUS FROM STACK
001144  000000                          HLT                     ;ERROR: RTT FAILED TO LOAD STATUS

001146  010701                  T3:     SCOPE
001150  012706  000500                  MOV     #STKPTR,SP      ;SET STACK PTR
001154  012746  000357                  MOV     #357,-(SP)      ;PUSH STATUS ON THE STACK
001160  012746  001172                  MOV     #T3A,-(SP)      ;PUSH PC ON THE STACK
001164  005067  176606                  CLR     PSW             ;PRE SET STATUS
001170  000006                          RTT
001172  016700  176600          T3A:    MOV     PSW,R0          ;GET STATUS AFTER RTT
001176  022700  000357                  CMP     #357,R0         ;CHECK STATUS AFTER RTT
001202  001411                          BEQ     T4
001204  000000                          HLT                     ;ERROR: INCORRECT STATUS AFTER RTT

                                ;CHECK THAT CP CAN TIME OUT TRAP
001206  012737  001224  000004          MOV     #TORET,@#ERRVEC ;LOAD TIMEOUT TRAP VECTOR
001214  005037  173000                  CLR     @#173000        ;ADDRESS 173000 ALWAYS TIMES OUT ON
                                                                ;DATIP/DATO BUS CYCLE
001220  000000                          HLT                     ;ERROR: FAILED TO TIME OUT TRAP
001222  000672                          BR      BEGIN           ;LOOP TEST
001224  022626          TORET:  CMP     (6)+,(6)+       ;RESTORE THE STACK

                                ;TEST THAT THE CP DOES THE INSTRUCTION FOLLOWING THE RTT IF THE TBIT
                                ;GETS SET (BY THE RTT)
001226  010701                  T4:     SCOPE
001230  012706  000500                  MOV     #STKPTR,SP      ;SET STACK PTR
001234  012767  001266  176552          MOV     #T4B,TBITVEC    ;LOAD 'T' BIT TRAP VECTOR AND
001242  005067  176550                  CLR     TBITVEC+2       ;ITS STATUS
001246  012746  000020                  MOV     #TBIT,-(SP)     ;SET UP STACK SUCH THAT RTT CAUSES
                                                                ;'T' BIT TO SET
001252  012746  001260                  MOV     #T4A,-(SP)
001256  000006                          RTT
001260  000240          T4A:    NOP                     ;CP SHOULD DO THIS INSTRUCTION AND
                                                        ;THEN TRAP
001262  000000          T4AA:   HLT                     ;PROCESSER DID NOT ACKNOWLEDGE 'T'
                                                        ;BIT TRAP
001264  000405                  BR      T5
```



```
001266  011600          T4B:    MOV     (SP),R0         ;GET PC WHEN 'T' TRAP OCCURRED
001270  022730  001262                  CMP     #T4AA,R0        ;DID CP DO THE NOP
001274  001401                          BEQ     T5
001276  000000                          HLT                     ;ERROR: 'T' BIT DID NOT TRAP AFTER
                                                                ;THE NOP
                                ;THIS TEST IS THE SAME AS T4 EXCEPT THAT THE INSTRUCTION FOLLOWING
                                ;THE RTT (WHICH SETS THE 'T' BIT) IS A THREE WORD (MOV PSW,2) INSTRUCTION.
001300  010701                  T5:     SCOPE
001302  012706  000500                  MOV     #STKPTR,SP      ;SET STACK PTR
001306  012767  001344  176500          MOV     #T5B,TBITVEC    ;LOAD 'T' BIT TRAP VECTOR
001314  012746  000020                  MOV     #TBIT,-(SP)     ;SETUP TO SET 'T' BIT
001320  012746  001332                  MOV     #T5A,-(SP)
001324  005067  177452                  CLR     TEMP            ;
001330  000006                          RTT                     ;RTT SETS THE 'T' BIT
001332  016767  176440  177442  T5A:    MOV     PSW,TEMP                ;MOVE STATUS TO MEM, LOCATION 2
001340  000000                          HLT                     ;'T' BIT DID NOT CAUSE A TRAP
001342  000407                          BR      T6              ;
001344  016700  177432          T5B:    MOV     TEMP,R0         ;GET STATUS THAT RTT LOADED
001350  022700  000020                  CMP     #TBIT,R0        ;DID RTT LOAD 'T' BIT INTO STATUS
001354  001402                          BEQ     T6
001356  003000                          HLT                     ;ERROR: EITHER RTT DID NOT LOAD
001360  000000                          HLT                     ;'T' BIT INTO STATUS OR 'T' BIT TRAPPED
                                                                ;BEFORE PSW WAS SAVED.
                                ;TEST THAT THE CP ACKNOWLEDGES THE 'T' BIT BEING SET BEFORE EXECUTING
                                ;THE INSTRUCTION FOLLOWING AN RTI
001362  010701                  T6:     SCOPE
001364  012706  000500                  MOV     #STKPTR,SP      ;SET STACK PTR
001370  012767  001412  176416          MOV     #T6B,TBITVEC    ;LOAD 'T' BIT TRAP VECTOR
001376  012746  000020                  MOV     #TBIT,-(SP)     ;SET UP TO SET 'T' BIT
001402  012746  001410                  MOV     #T6A,-(SP)
001406  000002                          RTI                     ;RTI SETS THE 'T' BIT
001410  000000          T6A:    HLT                     ;ERROR: CP SHOULD HAVE TRAPPED
                                                        ;BEFORE THIS INSTRUCTION WAS EXECUTED.
001412  000240          T6B:    NOP
001414  000498                  BR      T7              ;GO TO NEXT TEST

                                ;TEST THAT THE CP DOES NOT ACKNOWLEDGE THE 'T' BIT IF THE INSTRUCTION
                                ;FOLLOWING THE RTT CLEARS THE 'T' BIT.
```

```
001416  010701                  T7:     SCOPE
001420  012706  000500                  MOV     #STKPTR,SP      ;THIS IS WHAT THE STACK LOCKS
001424  012767  001462  176362          MOV     #T7C,TRAPVEC    ;LIKE BEFORE THE RTT
001432  005046                          CLR     -(SP)           ;670 = T7A = SP=678
001434  012746  001456                  MOV     #T7B,-(SP)      ;672 =  20 =
001440  012746  000020                  MOV     #TRAP,-(SP)     ;674 = T7B =
001444  012746  001454                  MOV     #T7A,-(SP)      ;676 =   0 =
001450  000240                          NOP                     ;700 =   ? = UNKNOWN
                                                                ;STACK AFTER THE RTT
001452  000006                          RTT                     ;674 = T7B = SP=674
                                                                ;676 =   0 = AND 'T' BIT
                                                                ;700 =   ? = IS SET
                                                                ;STACK AFTER THE RTI
001454  000002                  T7A:    RTI                     ;700 =   ? = SP=700
                                                                ;            AND 'T' BIT
                                                                ;            IS CLEAR
001456  000240                  T7B:    NOP
001460  000401                          BR      EL45
001462  000000                  T7C:    HLT                     ;CP ACKNOWLEDGED THE 'T' BIT

                                ;TEST IF THE PROCESSER IS AN 11/45
001464  012737  002164  000010  EL45:   MOV     #END,@#10       ;SET RESERVED INST TRAP VECTOR
001472  000230                          SPL     0               ;WILL TRAP TO END IF AN 11/40

                                ;TEST THAT AN RTI LOADS THE STATUS WORD PROPERLY IN KERNEL MODE.
001474  010701                  T10:    SCOPE
001476  005067  176274                  CLR     PSW
001502  012706  000770                  MOV     #KPTR,KSP       ;SET KERNEL STACK POINTER
001506  012716  054340                  MOV     #SM+PSM+REG+PRTY7,(KSP)
001512  012746  001526                  MOV     #T10A,-(KSP)
001516  005067  177260                  CLR     TEMP
001522  000002                          RTI                     ;POP 2 TOP WORDS OF KERNEL STACK INTO
                                                                ;STATUS AND PC
001524  000000                          HLT                     ;ERROR! RTI FAILED
001526  016767  176244  177246  T10A:   MOV     PSW,TEMP        ;GET STATUS WORD
001534  005067  176236                  CLR     PSW
001540  022767  054340  177234          CMP     #SM+PSM+REG+PRTY7,TEMP
001546  001401                          BEQ     .+4
001550  000000                          HLT                     ;ERROR! INCORRECT STATUS

                                ;THIS TEST SAME AS ABOVE EXEPT THAT RTT IS FROM KERNEL TO USER MODE.
001552  010701                  T11:    SCOPE
001554  012767  000340  176214          MOV     #PRTY7,PSW
001562  012706  000770                  MOV     #KPTR,KSP
001566  012716  174000                  MOV     #UM+PUM+REG,(KSP)
001572  012746  001606                  MOV     #T11A,-(KSP)
001576  005067  177200                  CLR     TEMP
001602  000006                          RTT
001604  000000                          HLT                     ;ERROR! RTT FAILED

001606  016767  176164  177166  T11A:   MOV     PSW,TEMP        ;GET STATUS WORD
001614  005067  176156                  CLR     PSW
001620  022767  174000  177154          CMP     #UM+PUM+REG,TEMP
```



```
001626  001401                          BEQ     .+4
001630  000000                          HLT                     ;ERROR! INCORRECT STATUS WORD

                                ;TEST RTT LOADS THE STATUS WORD CORRECTLY IN SUPERVISORY MODE
001632  010701                  T12:    SCOPE
001634  012767  340000  176134          MOV     #SM,PSW         ;SUPERVISORY MODE!!!
001642  012706  000750                  MOV     #SPTR,SSP       ;SET SUPERVISORY STACK POINTER
001646  012716  174340                  MOV     #UM+PUM+REG+PRTY7,(SSP)
001652  012746  001660                  MOV     #T12A,-(SSP)
001656  000006                          RTT
001660  016767  176112  177114  T12A:   MOV     PSW,TEMP        ;GET STATUS WORD
001666  005067  176104                  CLR     PSW             ;KERNEL MODE!!!
001672  022767  174000  177102          CMP     #UM+PUM+REG,TEMP
001700  001401                          BEQ     .+4
001702  000000                          HLT                     ;ERROR! INCORRECT STATUS

                                ;TEST THAT RTI DOES NOT CLEAR BITS15-11 IN SUPERVISORY MODE.
001704  010701                  T13:    SCOPE
001706  012767  074200  176062          MOV     #SM+PUM+REG+PRTY4,PSW
001714  012706  000750                  MOV     #SPTR,SSP
001720  012716  000340                  MOV     #PRTY7,(SSP)    ;LOAD RETURN STATUS ON SUPER STACK
001724  012746  001736                  MOV     #T13A,-(SSP)
001730  005067  177046                  CLR     TEMP
001734  000006                          RTT
001736  016767  176034  177036  T13A:   MOV     PSW,TEMP
001744  005067  176026                  CLR     PSW
001750  022767  074200  177024          CMP     #SM+PUM+REG+PRTY4,TEMP
001756  001401                          BEQ     .+4
001760  000000                          HLT                     ;ERROR! INCORRECT STATUS AFTER RTT

                                ;TEST THAT RTI LOADS STATUS PROPERLY IN USER MODE
001762  010701                  T14:    SCOPE
001764  012767  140000  176004          MOV     #UM,PSW         ;USER MODE!!!
001772  012706  000730                  MOV     #UPTR,USP       ;SET USER'S STACK POINTER
001776  012716  174340                  MOV     #UM+PUM+REG+PRTY7,(USP)
002002  012746  002014                  MOV     #T14A,-(USP)
002006  005067  176770                  CLR     TEMP
002012  000002                          RTI

002014  016767  175776  176760  T14A:   MOV     PSW,TEMP        ;GET STATUS WORD
002022  005067  175770                  CLR     PSW             ;KERNEL MODE!!!
002026  022767  174000  176746          CMP     #UM+PUM+REG,TEMP
002034  001401                          BEQ     .+4
002036  000000                          HLT                     ;ERROR! INCORRECT STATUS AFTER RTI

                                ;TEST THAT RTT DOES NOT CLEAR STATUS BITS 15-11 IN USER MODE.
002040  010701                  T15:    SCOPE
002042  012767  174340  175726          MOV     #UM+PUM+REG+PRTY7,PSW           ;USER MODE!!!
002050  012706  000730                  MOV     #UPTR,USP
002054  005046                          CLR     -(USP)
002056  012746  002070                  MOV     #T15A,-(USP)
002062  005067  176714                  CLR     TEMP
002066  000006                          RTT
```

```
002070  016747  175772  174774  T15A1:  MOV     PSW,TEMP
002076  005047  175674                  CLR     PSW                     ;KERNEL MODE!!!
002102  022747  174340  176672          CMP     #UM*PUM*REG*PRTY7,TEMP
002110  001401                          BEQ     .+4
002112  000000                          HLT

                                ;TEST THAT BITS 13-11 CAN BE CLEARED BY RTT IN KERNEL MODE.
002114  010701                  T16:    SCOPE
002116  012747  034000  175652          MOV     #PUM*REG,PSW            ;KERNEL MODE!!!
002124  012706  000770                  MOV     #KPTR,KSP               ;SET KERNEL STACK
002130  005016                          CLR     (KSP)
002132  012746  002146                  MOV     #T16A,-(KSP)
002136  012767  177777  176636          MOV     #-1,TEMP                ;PRESET TEMP
002144  000006                          RTT
002146  016767  175624  176626  T16A1:  MOV     PSW,TEMP                ;GET STATUS AFTER RTT
002154  005767  176622                  TST     TEMP                    ;CHECK STATUS AFTER RTT
002160  001401                          BEQ     .+4
002162  000000                          HLT                             ;ERROR! INCORRECT STATUS AFTER RTT

002164  012737  000012  000810  END:    MOV     #12,@#10                ;SET RESERVED INST TO HALT AT 12
002172  005267  176602                  INC     ICNT
002176  005767  176576                  TST     ICNT
002202  001402                          BEQ     DONE
002204  000167  176600                  JMP     BEGIN
002210  012767  000007  175350  DONE1:  MOV     #7,TPBUF
002216  105767  175342                  TSTB    TPCSR
002222  100375                          BPL     .-4
002224  013702  000042                  MOV     @#42,R2                 ;GET DECTAPE MONITOR RETURN ADDRESS
002230  001404                          BEQ     DONE1                   ;DO NOT RETURN IF (42)=0
002232  004712                          JSR     7,(2)                   ;RETURN TO DECTAPE MONITOR
002234  000240                          NOP                             ;ACT11
002236  000240                          NOP                             ;OVERLAY
002240  000240                          NOP                             ;AREA
002242  000167  176536          DONE11: JMP     START                   ;RESTART TEST

        000001                          .END
```



```
BEGIN    001010         DISPLA = 177570         DONE     002210         DONE1    002242
EL45     001464         END      000244         ERRVEC = 000004         HLT    = 000000
ICNT     021000         KH     = 004000         KPTR   = 000770         KSP    =%000006
PC     =%000007         PKH    = 000000         PRTYP  = 000200         PRTY4  = 000200
PRTY7  = 000340         PSW    = 177776         PUM    = 030000         R0     =%000000
REG    = 004000         RESVEC = 000010         R0     =%000000         R1     =%000001
R2     =%000002         R3     =%000003         R4     =%000004         R5     =%000005
SCOPE  = 010701         SM     = 040000         SP     =%000006         SPTR   = 000750
SSP    =%000006         START    001004         STKPTR = 000500         SWR    = 177570
TBIT   = 000020         TBITVE = 000014         TEMP     001002         TORET    001224
TPBUF  = 177566         TPCSR  = 177564         T0       001040         T0A      001056
T1       001060         T1A      001076         T10      001474         T10A     001526
T11      001552         T11A     001606         T12      001632         T12A     001660
T13      001704         T13A     001736         T14      001762         T14A     002014
T15      002040         T15A     002070         T16      002114         T16A     002146
T2       001110         T2A      001134         T3       001146         T3A      001172
T4       001226         T4A      001260         T4AA     001262         T4B      001266
T5       001300         T5A      001332         T5B      001344         T6       001362
T6A      001410         T6B      001412         T7       001416         T7A      001454
T7B      001456         T7C      001462         UBREAK = 177770         UM     = 140000
UPTR   = 000732         USP    =%000006         .      = 002246


 ERRORS DETECTED:  0
```