Type of page : 265

4 K.

1/2

13/01/1972

53ØPMO

P 860 CPU TEST ØPTIØNS

5

4 K. WØRDS MEMØRY

code 12 N.C. 5111 199 86431

[les pages 18; 28; 38; 48; 58. ne sont pas tirées para que blanches]

```
                IDENT   S3
*******  *******  ******************
**
*         S30PM0  P860  CPU OPTIONS TEST PROGRAM FOR 4K. WORDS MEMORY
**
*******  *******  ******************

*                 ASSEMBLY DATE:  13/JANUARY/1972

*                 PROGRAMMER:  BULOIS M
*                 FOR INFORMATIONS:  TEL. TO C.T.I.: 702 29 91  EXT. 278

*******  *******  ******************
**
*           OPTIONS:  MULTIPLY
*                     MULTIPLY WITH CONSTANT
*                     MULTIPLY REGISTERS/REGISTERS
*                     DIVIDE
*                     DIVIDE WITH CONSTANT
*                     DIVIDE REGISTERS/REGISTERS
*                     DOUBLE ADDITION
*                     DOUBLE ADDITION REGISTER
*                     DOUBLE ADDITION WITH CONSTANT
*                     DOUBLE SUBTRACT
*                     DOUBLE SUBTRACT REGISTER
*                     DOUBLE SUBTRACT WITH CONSTANT
**
*******  *******  ******************
```

```
00030                          EJECT
00031
00032  0000          BOOT      RES     32          BOOTSTRAP LOCATIONS
00033  0040  0000  F           DATA    ITROUT
00034  0042  0000  F           DATA    ITROUT
00035  0044  0000  F           DATA    ITROUT
00036  0046  0000  F           DATA    ITROUT.
00037  0048  0000  F           DATA    ITROUT
00038  004A  0000  F           DATA    ITROUT
00039  004C  0000  F           DATA    ITROUT
00040  004E  0000  F           DATA    ITROUT
00041  0050  0000  F           DATA    ITROUT
00042  0052  0000  F           DATA    ITROUT
00043  0054  0000  F           DATA    ITROUT
00044  0056  0000  F           DATA    ITROUT
00045  0058  0000  F           DATA    ITROUT
00046  005A  0000  F           DATA    ITROUT
00047  005C  0000  F           DATA    ITROUT
00048  005E  0000  F           DATA    ITROUT
00049  0060  4900      ITROUT  RIL     A1          READ INTERUPT LINE
00050  0062  207F              HLT                 UNEXPECTED INTERRUPT, PROGRAM DESTROYED
00051  0064                    RES     2
```

```
                        EJECT
                        AORG    /00C
00008 00CA 0F20         ABL     JAM
      00CC 0000 F
00056
00057                   ******  ******  ******************
00058                   **
00059                   ORGTB0  EQU     *
00060 00CE FFFF                 DATA    X'FFFF'
00061 00D0 8000                 DATA    X'8000'
00062 00D2 4000                 DATA    X'4000'
00063 00D4 2000                 DATA    X'2000'
00064 00D6 1000                 DATA    X'1000'
00065 00D8 0800                 DATA    X'0800'
00066 00DA 0400                 DATA    X'0400'
00067 00DC 0200                 DATA    X'0200'
00068 00DE 0100                 DATA    X'0100'
00069 00E0 0080                 DATA    X'0080'
00070 00E2 0040                 DATA    X'0040'
00071 00E4 0020                 DATA    X'0020'
00072 00E6 0010                 DATA    X'0010'
00073 00E8 0008                 DATA    X'0008'
00074 00EA 0004                 DATA    X'0004'
00075 00EC 0002                 DATA    X'0002'
00076 00EE 0001                 DATA    X'0001'
00077 00F0 0000         ORGTB1  DATA    X'0000'
00078 00F2 0001         ORGTB3  DATA    X'0001'
00079 00F4 0002                 DATA    X'0002'
00080 00F6 0004                 DATA    X'0004'
00081 00F8 0008         ORGTB4  DATA    X'0008'
00082 00FA 0010                 DATA    X'0010'
00083 00FC 0020                 DATA    X'0020'
00084 00FE 0040                 DATA    X'0040'
00085 0100 0080         WMEMOR  DATA    X'0080'
00086 0102 0100                 DATA    X'0100'
00087 0104 0200                 DATA    X'0200'
00088 0106 0400                 DATA    X'0400'
00089 0108 0800                 DATA    X'0800'
00090 010A 1000                 DATA    X'1000'
00091 010C 2000         CMEMOR  DATA    X'2000'
00092 010E 4000                 DATA    X'4000'
00093 0110 8000                 DATA    X'8000'
00094 0112 FFFF         ORGTB2  DATA    X'FFFF'
00095 0114 0000                 DATA    X'0000'
00096 0116 0001                 DATA    X'0001'
00097 0118 FFFF                 DATA    X'FFFF'
00098                   **
00099                   ******  ******  ******************
```

```
00100                          EJECT
00110
00120
00130  011A                MEMSTK   RES   20
00134                      STKP     EQU   *-2
00135  0142                NBOFML   RES   1          ML COUNTER FOR EXECUTION INDEX, ML
00136  0144  00F0           MLIAD    DATA  ORGTBI
00137  0146                INDEXN   RES   1          INDEX WORD
00138  0148                MLCNT    RES   1          USE FOR COMPARE 14 REGISTERS
00139  014A                SAVA4    RES   1          USED TO SAVE REGISTER A4
00140  014C                SAVA15   RES   1          SAVE ADDRESS FOR A15
00141  014E                NBRMS    RES   1          CONTAIN NUMBER OF REG, TO BE STOR,
00142  0150                STAKMS   RES   15
00143                      STAKND   EQU   *-2
00144  015E  0150          MSIAD    DATA  STAKMS
00145  0160  0000          CCIAD    DATA  0
00146  0162                NBNSH    RES   1          NUMBER OF POSITIONS FOR SHIFT INSTRUCTION,
00147  0164                NBNSH1   RES   1          NUMBER OF SHIFTS TO MULT, AND DIV, UTILISAT,
00148  0166  0000          PASCT1   DATA  0
00149  0168  0000          PASCT2   DATA  0
00150
00151                      JAM      EQU   *
```

```
00122                          EJECT
00123
00124                  ******  ******  ***   **********
00125                  **
00126                  *       MU      MULTIPLY
00127                  **
00128                  ******  ******  ****************
00129
00130  017A 8120       MU00    LDKL    A1,X'FFFF'
       017C FFFF
00131  017E 0220               LDK     A2,0
00132  0180 C340               MU      ORGTB1          MULTIPLY EXPECT, RESULT: (A1)=0 (A2)=0
       0182 00F0
00133  0184 57A0               RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00134  0186 5302               RF(0)   *+4             JP IF CR = 0
00135  0188 207F               HLT                     *** INSTRUCTION MU00 DID NOT MODIF, CR
00136  018A E920               CHK     A1,X'0000'
       018C 0020
00137  018E 5302               RF(0)   *+4             JP IF (REG, A1) = X'0000'
00138  0190 207F               HLT                     *** INSTRUCTION MU00 EXECUTION WAS WRONG
00139  0192 EA20               CHK     A2,X'0000'
       0194 0020
00140  0196 5302               RF(0)   *+4             JP IF (REG, A2) = X'0000'
00141  0198 207F               HLT                     *** INSTRUCTION MU00 EXECUTION WAS WRONG
00142  019A 8120               LDKL    A1,X'FFFF'
       019C FFFF
00143  019E 8220               LDKL    A2,X'FFFF'
       01A0 FFFF
00144  01A2 C340       MU01    MU      ORGTB1          MULTIPLY EXPECT, RESULT : (A1)=0 (A2)=0
       01A4 00F0
00145  01A6 57A0               RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00146  01A8 5302               RF(0)   *+4             JP IF CR = 0
00147  01AA 207F               HLT                     *** INSTRUCTION MU01 DID NOT MODIF, CR
00148  01AC E920               CHK     A1,X'0000'
       01AE 0020
00149  01B0 5302               RF(0)   *+4             JP IF (REG, A1) = X'0000'
00150  01B2 207F               HLT                     *** INSTRUCTION MU01 EXECUTION WAS WRONG
00151  01B4 EA20               CHK     A2,X'0000'
       01B6 0020
00152  01B8 5302               RF(0)   *+4             JP IF (REG, A2) = X'0000'
00153  01BA 207F               HLT                     *** INSTRUCTION MU01 EXECUTION WAS WRONG
00154  01BC 0100               LDK     A1,0
00155  01BE 87A0               LDKL    A15,STKP
       01C0 0140
00156  01C2 F7A1               CF      A15,MU2
       01C4 0002 F
00157  01C6 6F20               ABL     MUKTST
       01C8 0000 F
```

```
                        EJECT
                ******  ******  ****  **********
                **
                *       MUK     MULTIPLY WITH CONSTANT
                **
                ******  ******  ******************

MUKTST  EQU     *
01CA 8120       LDKL    A1,X'FFFF'
01CC FFFF
01CE 97A0       LDKL    A15,STKP
01D0 0140
01D2 F7A1       CF      A15,MU2
01D4 0000 F
01D6 5720       RF(7)   *+2             PASS IN ORDER TO DEBUGGING
01D8 9F20       ABL     MUKTST
01DA 0000 F

                ******  ******  ******************
                **
                *       MUR     MULTIPLY REGISTERS/REGISTERS TEST
                **
                ******  ******  ******************

MURTST  EQU     *
01DC 97A0       LDKL    A15,STKP
01DE 0140
MURG8   EQU     *
01E0 0101       LDK     A1,1
01E2 F7A1       CF      A15,MURPRO      TO MULTIPLY BY A8-REGISTER
01E4 0000 F
01E6 5720       RF(7)   *+2             PASS IN ORDER TO DEBUGGING
MURG1   EQU     *
01E8 0102       LDK     A1,2
01EA F7A1       CF      A15,MURPRO      TO MULTIPLY BY A1 REGISTER
01EC 0000 F
01EE 5720       RF(7)   *+2             PASS IN ORDER TO DEBUGGING
MURG9   EQU     *
01F0 0103       LDK     A1,3
01F2 F7A1       CF      A15,MURPRO      TO MULTIPLY BY A9 REGISTER
01F4 0000 F
01F6 5720       RF(7)   *+2             PASS IN ORDER TO DEBUGGING
MURG10  EQU     *
01F8 0105       LDK     A1,5
01FA F7A1       CF      A15,MURPRO      TO MULTIPLY BY A10 REGISTER
01FC 0000 F
01FE 5720       RF(7)   *+2             PASS IN ORDER TO DEBUGGING
MURG11  EQU     *
0200 0107       LDK     A1,7
```

```
00199  0202  F7A1            CF      A15,MURPRO      TO MULTIPLY BY A11 REGISTER
       0204  0000  F
0020-  0205  5700            RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00201                MURG4   EQU     *
00202  0208  0108            LDK     A1,8
00203  020A  F7A1            CF      A15,MURPRO      TO MULTIPLY BY A4 REGISTER
       020C  0000  F
00204  020E  5700            RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00205                MURG12  EQU     *
00206  0210  0109            LDK     A1,9
00207  0212  F7A1            CF      A15,MURPRO      TO MULTIPLY BY A12 REGISTER
       0214  0000  F
00208  0216  5700            RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00209                MURG5   EQU     *
00210  0218  010A            LDK     A1,10
00211  021A  F7A1            CF      A15,MURPRO      TO MULTIPLY BY A5 REGISTER
       021C  0000  F
00212  021E  5700            RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00213                MURG13  EQU     *
00214  0220  010B            LDK     A1,11
00215  0222  F7A1            CF      A15,MURPRO      TO MULTIPLY BY A13 REGISTER
       0224  0000  F
00216  0226  5700            RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00217                MURG6   EQU     *
00218  0228  010C            LDK     A1,12
00219  022A  F7A1            CF      A15,MURPRO      TO MULTIPLY BY A6 REGISTER
       022C  0000  F
00220  022E  5700            RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00221                MURG14  EQU     *
00222  0230  010D            LDK     A1,13
00223  0232  F7A1            CF      A15,MURPRO      TO MULTIPLY BY A14 REGISTER
       0234  0000  F
00224  0236  5700            RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00225                MURG7   EQU     *
00226  0238  010E            LDK     A1,14
00227  023A  F7A1            CF      A15,MURPRO      TO MULTIPLY BY A7 REGISTER
       023C  0000  F
00228  023E  5700            RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00229                MURG15  EQU     *
00230  0240  010F            LDK     A1,15
00231  0242  F7A1            CF      A15,MURPRO      TO MULTIPLY BY A15 REGISTER
       0244  0000  F
00232  0246  5700            RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00233  0248  0F20            ABL     OVTST
       024A  0000  F
```

```
00234                          EJECT
00235
00236                  ****** ****** **** **********
00237                  **
00238                  *       MURPRO  MUR SUBROUTINE
00239                  **
00240                  ****** ****** ******************
00241
00242                  MURPRO  EQU     *
00243  024C  8141              ST      A1,REGMUL
       024E  0060  F
00244  0250  5700              RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00245  0252  5700              RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00246  0254  5700  F           RF(7)   MU2
00247
00248                  ****** ****** ******************
00249                  **
00250                  *       MULTIPLY MULTIPLICATOR: 2 TO 4096
00251                  **
00252                  ****** ****** ******************
00253
00254                  MU2     EQU     *
00255  0256  5700              RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00256  0258  840E              LDK     A4,14
00257  025A  870E              LDK     A7,14
00258  025C  8202              LDK     A2,2
00259  025E  8501              LDK     A5,1
00260  0260  F7A1              CF      A15,MUS0        TO MULTIPLY 2   BY ORGTB TABLE
       0262  0000  F
00261                  MU4     EQU     *
00262  0264  5700              RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00263  0266  8480              LDK     A4,13
00264  0268  8780              LDK     A7,13
00265  026A  8602              LDK     A6,2
00266  026C  8204              LDK     A2,4
00267  026E  F7A1              CF      A15,MUS0        TO MULTIPLY 4   BY ORGTB TABLE
       0270  0000  F
00268                  MU8     EQU     *
00269  0272  5700              RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00270  0274  840C              LDK     A4,12
00271  0276  870C              LDK     A7,12
00272  0278  8603              LDK     A6,3
00273  027A  8208              LDK     A2,8
00274  027C  F7A1              CF      A15,MUS0        TO MULTIPLY 8   BY ORGTB TABLE
       027E  0000  F
00275                  MU16    EQU     *
00276  0280  5700              RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00277  0282  8408              LDK     A4,11
00278  0284  8708              LDK     A7,11
00279  0286  8604              LDK     A6,4
```

```
00280  0280  0210          LDK     A2
0028   028A  F7A1          CF      A15    30        TO MULTIPLY 16  BY ORGTB TABLE
       028C  0000  F
00282                MU32  EQU     *
00283  028E  5700          RF(7)   *+2               PASS IN ORDER TO DEBUGGING
00284  0290  040A          LDK     A4,10
00285  0292  070A          LDK     A7,10
00286  0294  0605          LDK     A6,5
00287  0296  0220          LDK     A2,32
00288  0298  F7A1          CF      A15,MUS0          TO MULTIPLY 32  BY ORGTB TABLE
       029A  0000  F
00289                MU64  EQU     *
00290  029C  5700          RF(7)   *+2               PASS IN ORDER TO DEBUGGING
00291  029E  0409          LDK     A4,9
00292  02A0  0709          LDK     A7,9
00293  02A2  0606          LDK     A6,6
00294  02A4  0240          LDK     A2,64
00295  02A6  F7A1          CF      A15,MUS0          TO MULTIPLY 64  BY ORGTB TABLE
       02A8  0000  F
00296                MU128 EQU     *
00297  02AA  5700          RF(7)   *+2               PASS IN ORDER TO DEBUGGING
00298  02AC  0408          LDK     A4,8
00299  02AE  0708          LDK     A7,8
00300  02B0  0607          LDK     A6,7
00301  02B2  0280          LDK     A2,128
00302  02B4  F7A1          CF      A15,MUS0          TO MULTIPLY 128 BY ORGTB TABLE
       02B6  0000  F
00303                MU256 EQU     *
00304  02B8  5700          RF(7)   *+2               PASS IN ORDER TO DEBUGGING
00305  02BA  0407          LDK     A4,7
00306  02BC  0707          LDK     A7,7
00307  02BE  0608          LDK     A6,8
00308  02C0  8200          LDKL    A2,256
       02C2  0100
00309  02C4  F7A1          CF      A15,MUS0          TO MULTIPLY 256 BY ORGTB TABLE
       02C6  0000  F
00310                MU512 EQU     *
00311  02C8  5700          RF(7)   *+2               PASS IN ORDER TO DEBUGGING
00312  02CA  0406          LDK     A4,6
00313  02CC  0706          LDK     A7,6
00314  02CE  0609          LDK     A6,9
00315  02D0  8200          LDKL    A2,512
       02D2  0200
00316  02D4  F7A1          CF      A15,MUS0          TO MULTIPLY 512 BY ORGTB TABLE
       02D6  0000  F
00317                MU1024 EQU    *
00318  02D8  5700          RF(7)   *+2               PASS IN ORDER TO DEBUGGING
00319  02DA  0405          LDK     A4,5
00320  02DC  0705          LDK     A7,5
00321  02DE  060A          LDK     A6,10
```

```
00322  02E8  0220          LDKL    A2,1024
       02EA  0400
0032-  02EC  F7A1          CF      A15,  S0        TO MULTIPLY 1024 BY ORGTH T  ,E
       02EE  0000  F
00324                MU2048 EQU     *
00325  02F0  5700          RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00326  02F2  0404          LDK     A4,4
00327  02F4  0704          LDK     A7,4
00328  02F6  060B          LDK     A6,11
00329  02F8  0220          LDKL    A2,2048
       02FA  0800
00330  02FC  F7A1          CF      A15,MUS0        TO MULTIPLY 2048 BY ORGTB TABLE
       02FE  0000  F
00331                MU4096 EQU     *
00332  0300  5700          RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00333  0302  0403          LDK     A4,3
00334  0304  0703          LDK     A7,3
00335  0306  060C          LDK     A6,12
00336  0308  0220          LDKL    A2,4096
       030A  1000
00337  030C  F7A1          CF      A15,MUS0        TO MULTIPLY 4096 BY ORGTB TABLE
       030E  0000  F
00338  0310  F03E          RTN     A15             RETURN
```

```
                        EJECT
                ******  ******  ******************
                **
                *       MUS0    MULTIPLY COMMON ROUTINE
                **
                ******  ******  ******************

030A            FMUR    RES     1               MULTIPLY REGISTERS/REGISTERS FLAG
                *                                       IF = 0 MU INSTRUCTION
                *                                       IF = 1 MUR INSTRUCTION
                *                                       IF = -1 MUK INSTRUCTION
030C            REGMUL  RES     1               REGISTER MULT. USED
030E            MULTW   RES     1               MULTIPLICANDE
                MUS0    EQU     *
0310 67C1               ST      A15,SAVA15      SAVE RETURN
0312 014C
0314 6141               ST      A1,FMUR
0316 030A
0318 6320               LDK     A3,32
031A 6241               ST      A2,MULTW
031C 030E
031E 6641               ST      A6,NBNSH1
0320 6174
                MUS00   EQU     *
0322 6120               LDKL    A1,X'FFFF'
0324 FFFF
0326 6249               LD      A2,MULTW
0328 030E
032A 6441               ST      A4,SAVA4        SAVE REGISTER A4
032C 014A
032E 60C0               LD      A6,FMUR
0330 030A
0332 5100 F             RF(1)   MURTRO          JP IF MUR INSTRUCTION
0334 5200 F             RF(2)   MUKTRO          MUK TREATMENT
0336 C04C       MUS1    MU      ORGTB0,A3
0338 00CE
033A 5700               RF(7)   *+2             PASS IN ORDER TO DEBUGGING
033C 5700               RF(7)   *+2             PASS IN ORDER TO DEBUGGING
033E 5422               RF(4)   *+4             JP IF CR RESULT # 0
0340 207F               HLT             ***     AFTER INSTRUCTION MUS1 CR RESULT WAS WRONG
0342 5100 F             RF(1)   MUS1A           JP IF CR RESULT = 1
0344 5220 F             RF(2)   MUS1B           JP IF CR RESULT = 2
0346 207F               HLT             ***     AFTER INSTRUCTION MUS1 CR RESULT WAS WRONG
0348 5700 F             RF(7)   MUS1C
                MUS1A   EQU     *
034A 8440               LD      A4,SAVA4
034C 014A
034E EC20               CWK     A4,0
0350 0000
```

```
00378  0352  5100  F        RF(1)    MUS1                  JP IF (REG. A4) > 0
0037'  0354  207F           HLT                      ***   AFTER INSTRUCTION MUS1 CR R   LT WAS WRONG
00380  0356  5700  F        RF(7)    MUS1
00381                MUS1B   EQU      *
00382  0358  8440           LD       A4,SAVA4
       035A  014A
00383  035C  EC20           CWK      A4,0
       035E  0000
00384  0360  5302           RF(0)    **4                   JP IF (REG. A4) = 0
00385  0362  207F           HLT                      ***   AFTER INSTRUCTION MUS1 CR RESULT WAS WRONG
00386                MUS1C   EQU      *
00387  0364  8440           LD       A4,SAVA4
       0366  014A
00388  0368  EF20           CWK      A7,0
       036A  0000
00389  036C  5503  F        RF(5)    MUS1E                 JP IF (REG. A7) IS NOT POSITIV
00390  036E  E920           CWK      A1,0
       0370  0000
00391  0372  5302           RF(0)    **4                   JP IF (REG. A1) = X'0000'
00392  0374  207F           HLT                      ***   INSTRUCTION MUS1 EXECUTION WAS WRONG
00393  0376  8540           LD       A6,N5NSH1
       0378  0174
00394  037A  854C           LD       A5,ORGTB0,A3
       037C  00CE
00395                MUS1D   EQU      *
00396  037E  3D01           SLA1     A5
00397  0380  9620           ADKL     A6,-1
       0382  FFFF
00398  0384  5908           RB(1)    MUS1D                 JP IF (REG. A6) > 0
00399  0386  E005           CWR      A5,A2
00400  0388  5302           RF(0)    **4                   JP IF (REG. A5) = (REG. A2)
00401  038A  207F           HLT                      ***   INSTRUCTION MUS1 EXECUTION WAS WRONG
00402  038C  9720           ADKL     A7,-1
       038E  FFFF
00403  0390  9320           ADKL     A3,-2
       0392  FFFE
00404  0394  9420           ADKL     A4,-1
       0396  FFFF
00405  0398  5978           RB(1)    MUS00                 JP IF (REG. A4) > 0
00406  039A  8700           LD       A15,SAVA15            RESTORE A15
       039C  014C
00407  039E  F03E           RTN      A15                   RETURN
00408                MUS1E   EQU      *
00409  03A0  EA20           CWK      A2,0
       03A2  0000
00410  03A4  5302           RF(0)    **4                   JP IF (REG. A2) = X'0000'
00411  03A6  207F           HLT                      ***   INSTRUCTION MUS1 EXECUTION WAS WRONG
00412  03A8  8501           LDK      A5,1
00413  03AA  5F2E           RB(7)    MUS1D
```

```
00414                           EJECT
00415
00416                   ******  ******  ****  **********
00417                   **
00418                   *       MURTR0  MUR TREATMENT
00419                   **
00420                   ******  ******  ******************
00421
00422           MURTR0  EQU     *
00423  03AC 0200        LDK     A2,0
00424  03AE 8140        LD      A1,REGMUL
       03B0 030C
00425           MURTR   EQU     *
00426  03B2 1201        ADK     A2,1
00427  03B4 9120        ADKL    A1,-1
       03B6 FFFF
00428  03B8 5C08        RB(4)   MURTR           JP TF (NEG, A1) = 1
00429  03BA 3A01        SLA1    A2
00430  03BC 8528        LDR     A5,A2
00431  03BE AA20        ORKL    A2,X'C000'
       03C0 C000
00432  03C2 8241        ST      A2,MURTR1
       03C4 0000  F
00433  03C5 3D01        SLA1    A5
00434  03C8 3D01        SLA1    A5
00435  03CA 3D01        SLA1    A5
00436  03CC 3D01        SLA1    A5
00437  03CE 3D01        SLA1    A5
00438  03D0 3D01        SLA1    A5
00439  03D2 AD20        ORKL    A5,X'804C'
       03D4 804C
00440  03D6 8541        ST      A5,LDREGM
       03D8 0000  F
00441  03DA 8240        LD      A2,MULTM
       03DC 03DE
00442  03DE 0000  LDREGM  DATA   0
00443  03E0 00CE        DATA    ORGT00
00444  03E2 0000  MURTR1  DATA   0             MULTIPLY REGISTERS/REGISTERS: A1 TO A15
00445  03E4 5700        RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00446  03E6 5700        RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00447  03E8 5FB0        RB(7)   MUS1+4
00448
00449                   ******  ******  ******************
00450                   **
00451                   *       MUKTR0  MUK TREATMENT
00452                   **
00453                   ******  ******  ******************
00454
00455           MUKTR0  EQU     *
00456  03EA 8540        LD      A5,ORGT00,A3
```

```
                03EC  00CE
00457           03EE  8541          ST      A5,   TR1+2
                03F0  0000  F
00458           03F2  5700          RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00459           03F4  C020  MUXTR1  DATA    X'C020'         MULTIPLY WITH CONSTANT
00460           03F6  0000          DATA    0               CONSTANT = 1 TO X'4000'
00461           03F8  5700          RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00462           03FA  5700          RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00463           03FC  5FC4          RB(7)   MUS1+4
```

```
                                EJECT
                        ******  ******  ***  ***********
                        **
                        *       DV      DIVIDE
                        **
                        ******  ******  ******  *********

                DVTST   EQU     *
03FE 0100               LDK     A1,0
0400 0200               LDK     A2,0
0402 C840       DV00    DV      ORGTB1          DIVIDE, EXPECT, RESULT: (A1) = 0 (A2) = 0
0404 02F0
0406 5703               RF(7)   *+2             PASS IN ORDER TO DEBUGGING
0408 5302               RF(3)   *+4             JP IF CR RESULT = 3
040A 207F               HLT                     *** CR RESULT OF INSTRUCTION DV00 WAS WRONG
040C E920               CWK     A1,0
040E 0000
0410 5002               RF(0)   *+4             JP IF (REG. A1) = X'0000'
0412 207F               HLT                     *** INSTRUCTION DV00 EXECUTION WAS WRONG
0414 EA20               CWK     A2,0
0416 0000
0418 5002               RF(0)   *+4             JP IF (REG. A2) = X'0000'
041A 207F               HLT                     *** INSTRUCTION DV00 EXECUTION WAS WRONG
041C 8120               LDKL    A1,X'FFFF'
041E FFFF
0420 8220               LDKL    A2,X'FFFF'
0422 FFFF
0424 C840       DV01    DV      ORGTB1          DIVIDE, EXPECT, RESULT: (A1) = 0 (A2) = 0
0426 02F0
0428 5703               RF(7)   *+2             PASS IN ORDER TO DEBUGGING
042A 5302               RF(3)   *+4             JP IF CR RESULT = 3
042C 207F               HLT                     *** CR RESULT OF INSTRUCTION DV01 WAS WRONG
042E E920               CWK     A1,X'FFFF'
0430 FFFF
0432 5002               RF(0)   *+4             JP IF (REG. A1) = X'FFFF'
0434 207F               HLT                     *** INSTRUCTION DV01 EXECUTION WAS WRONG
0436 EA20               CWK     A2,X'FFFF'
0438 FFFF
043A 5002               RF(0)   *+4             JP IF (REG. A2) = X'FFFF'
043C 207F               HLT                     *** INSTRUCTION DV01 EXECUTION WAS WRONG
043E 0100               LDK     A1,0
0440 0200               LDK     A2,0
0442 C840       DV10    DV      ORGTB2          DIVIDE EXPECT, RESULT: (A1) = 0 (A2) = 0
0444 0112
0446 5703               RF(7)   *+2             PASS IN ORDER TO DEBUGGING
0448 5302               RF(0)   *+4             JP IF CR RESULT = 0
044A 207F               HLT                     *** CR RESULT OF INSTRUCTION DV10 WAS WRONG
044C E920               CWK     A1,0
044E 0000
```

```
00504  0450  5002          RF(0)   **2          JP IF (REG, A1) = X'0000'
005    0452  207F          HLT                  *** INSTRUCTION DV10 EXECUTION WRONG
0050   0454  EA20          CWK     A2,-
       0456  0020
00507  0458  5002          RF(0)   **4          JP IF (REG, A2) = X'0000'
00508  045A  207F          HLT                  *** INSTRUCTION DV10 EXECUTION WAS WRONG
00509  045C  0100          LDK     1,0
00510  045E  87A3          LDKL    A15,STKP
       0460  0140
00511  0462  F7A1          DF      A15,DV2
       0464  0200  F
00512  0466  8F20          ABL     DVKTST
       0468  0000  F
```

```
                                EJECT
                ******  ******  ****  **********
                **
                *       DVK     DIVIDE WITH CONSTANT
                **
                ******  ******  ********  *********

                DVKTST  EQU     *
0045A 8120              LDKL    A1,X'FFFF'
0045C FFFF
0045E 87A0              LDKL    A15,STKP
0460 0140
0462 F7A1               CF      A15,DV2
0464 0008 F
0466 5700               RF(7)   *+2             PASS IN ORDER TO DEBUGGING
0468 8F20               ABL     DVRTST
046A 0000 F

                ******  ******  ******************
                **
                *       DVR     DIVIDE REGISTERS/REGISTERS TEST
                **
                ******  ******  ******************

                DVRTST  EQU     *
047C 87A0               LDKL    A15,STKP
047E 0140
                DVRG8   EQU     *
0480 8101               LDK     A1,1
0482 F7A1               CF      A15,DVRPRO      TO DIVIDE BY A8  REGISTER
0484 0000 F
0486 5700               RF(7)   *+2             PASS IN ORDER TO DEBUGGING
                DVRG9   EQU     *
0488 8103               LDK     A1,3
048A F7A1               CF      A15,DVRPRO      TO DIVIDE BY A9  REGISTER
048C 0000 F
048E 5700               RF(7)   *+2             PASS IN ORDER TO DEBUGGING
                DVRG10  EQU     *
0490 8105               LDK     A1,5
0492 F7A1               CF      A15,DVRPRO      TO DIVIDE BY A10 REGISTER
0494 0000 F
0496 5700               RF(7)   *+2             PASS IN ORDER TO DEBUGGING
                DVRG3   EQU     *
0498 8106               LDK     A1,6
049A F7A1               CF      A15,DVRPRO      TO DIVIDE BY A3  REGISTER
049C 0000 F
049E 5700               RF(7)   *+2             PASS IN ORDER TO DEBUGGING
                DVRG11  EQU     *
04A0 8107               LDK     A1,7
```

```
00554  04A2  F7A1          CF      A15,DVRPRO      TO DIVIDE BY A11 REGISTER
       04A4  0000  F
00555  04A6  5700          RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00556                DVRG4  EQU     *
00557  04A8  5700          RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00558  04AA  5700          RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00559  04AC  5700          RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00560  04AE  5700          RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00561                DVRG12 EQU     *
00562  04B0  0109          LDK     A1,9
00563  04B2  F7A1          CF      A15,DVRPRO      TO DIVIDE BY A12 REGISTER
       04B4  0000  F
00564  04B6  5700          RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00565                DVRG5  EQU     *
00566  04B8  010A          LDK     A1,10
00567  04BA  F7A1          CF      A15,DVRPRO      TO DIVIDE BY A5 REGISTER
       04BC  0000  F
00568  04BE  5700          RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00569                DVRG13 EQU     *
00570  04C0  010B          LDK     A1,11
00571  04C2  F7A1          CF      A15,DVRPRO      TO DIVIDE BY A13 REGISTER
       04C4  0000  F
00572  04C6  5700          RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00573                DVRG6  EQU     *
00574  04C8  010C          LDK     A1,12
00575  04CA  F7A1          CF      A15,DVRPRO      TO DIVIDE BY A6 REGISTER
       04CC  0000  F
00576  04CE  5700          RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00577                DVRG14 EQU     *
00578  04D0  010D          LDK     A1,13
00579  04D2  F7A1          CF      A15,DVRPRO      TO DIVIDE BY A14 REGISTER
       04D4  0000  F
00580  04D6  5700          RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00581                DVRG7  EQU     *
00582  04D8  010E          LDK     A1,14
00583  04DA  F7A1          CF      A15,DVRPRO      TO DIVIDE BY A7 REGISTER
       04DC  0000  F
00584  04DE  5700          RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00585                DVRG15 EQU     *
00586  04E0  010F          LDK     A1,15
00587  04E2  F7A1          CF      A15,DVRPRO      TO DIVIDE BY A15 REGISTER
       04E4  0000  F
00588  04E6  5700          RF(7)   *+2             PASS IN ORDER TO DEBUGGING
00589  04E8  8F20          ABL     DVDIF
       04EA  0000  F
```

```
                                EJECT
                ******  ******  ***  ***********
                **
                *       DIVIDE TEST WITH REMAINDER # 0
                **
                ******  ******  ********  *********

                DVDIF   EQU     *
00599 04EC 8120         LDKL    A1,-10
      04EE FFF6
00600 04F0 8141         ST      A1,DVWORD
      04F2 0000 F
00601 04F4 8120         LDKL    A1,-1
      04F6 FFFF
00602 04F8 8220         LDKL    A2,-52
      04FA FFCC
00603 04FC C840   DVRST1  DV      DVWORD
      04FE 0000 F
00604 0500 5102         RF(1)   *+4             JP IF CR RESULT = 1
00605 0502 207F         HLT                     CR RESULT OF INSTRUCTION DVRST1 WAS WRONG
00606 0504 EA20         CRK     A2,5
      0506 0005
00607 0508 5002         RF(0)   *+4             JP IF CR RESULT = 0
00608 050A 207F         HLT                 ***  INSTRUCTION DVRST1 EXECUTION WAS WRONG
00609 050C E920         CRK     A1,-2
      050E FFFE
00610 0510 5002         RF(0)   *+4             JP IF CR RESULT = 0
00611 0512 207F         HLT                 ***  INSTRUCTION DVRST1 EXECUTION WAS WRONG
00612 0514 810A         LDK     A1,10
00613 0516 8141         ST      A1,DVWORD
      0518 0000 F
00614 051A 8120         LDKL    A1,-1
      051C FFFF
00615 051E 8220         LDKL    A2,-52
      0520 FFCC
00616 0522 C840   DVRST2  DV      DVWORD
      0524 0000 F
00617 0526 5202         RF(2)   *+4             JP IF CR RESULT = 2
00618 0528 207F         HLT                 ***  CR RESULT OF INSTRUCTION DVRST2 WAS WRONG
00619 052A EA20         CRK     A2,-5
      052C FFFB
00620 052E 5002         RF(0)   *+4             JP IF CR RESULT = 0
00621 0530 207F         HLT                 ***  INSTRUCTION DVRST2 EXECUTION WAS WRONG
00622 0532 E920         CRK     A1,-2
      0534 FFFE
00623 0536 5002         RF(0)   *+4             JP IF CR RESULT = 0
00624 0538 207F         HLT                 ***  INSTRUCTION DVRST2 EXECUTION WAS WRONG
00625 053A 8120         LDKL    A1,-10
      053C FFF6
```

```
00626  0538  8141          ST      A1,DVWORD
       053A  0000  F
00627  053C  8100          LDK     A1,0
00628  053E  8234          LDK     A2,52
00629  0540  C840  DVRST3  DV      DVWORD
       0542  0000  F
00630  0544  EA20          CHK     A2,-5
       0546  FFFB
00631  0548  5002          RF(0)   **4                     JP IF CR RESULT = 0
00632  054A  207F          HLT                             *** INSTRUCTION DVRST3 EXECUTION WAS WRONG
00633  054C  E920          CHK     A1,2
       054E  0002
00634  0550  5002          RF(0)   **4                     JP IF CR RESULT = 0
00635  0552  207F          HLT                             *** INSTRUCTION DVRST3 EXECUTION WAS WRONG
00636  0554  8100          LDK     A1,0
00637  0556  8141          ST      A1,DVWORD
       0558  0000  F
00638  055A  8120          LDKL    A1,-1
       055C  FFFF
00639  055E  8220          LDKL    A2,-52
       0560  FFCC
00640  0562  C840  DVRST4  DV      DVWORD
       0564  0000  F
00641  0566  5302          RF(3)   **4                     JP IF CR RESULT = 3
00642  0568  207F          HLT                             *** CR RESULT OF INSTRUCTION DVRST4 WAS WRONG
00643  056A  8100          LDK     A1,0
00644  056C  8141          ST      A1,DVWORD
       056E  0000  F
00645  0570  8234          LDK     A2,52
00646  0572  C840  DVRST5  DV      DVWORD
       0574  0000  F
00647  0576  5302          RF(3)   **4                     JP IF CR RESULT = 3
00648  0578  207F          HLT                             *** CR RESULT OF INSTRUCTION DVRST5 WAS WRONG
00649  057A  8120          LDKL    A1,-1
       057C  FFFF
00650  057E  8141          ST      A1,DVWORD
       0580  0000  F
00651  0582  8220          LDKL    A2,-1
       0584  FFFF
00652  0586  C840  DVRST6  DV      DVWORD
       0588  0000  F
00653  058A  5102          RF(1)   **4
00654  058C  207F          HLT                             *** CR RESULT OF INSTRUCTION DVRST6 WAS WRONG
00655  058E  EA20          CHK     A2,1
       0590  0001
00656  0592  5002          RF(0)   **4                     JP IF CR RESULT = 0
00657  0594  207F          HLT                             *** INSTRUCTION DVRST6 EXECUTION WAS WRONG
00658  0596  E920          CHK     A1,0
       0598  0000
00659  059A  5002          RF(0)   **4                     JP IF CR RESULT = 0
```

```
0A669 05A2 207F            HLT                 *** INSTRUCTION DVRST6 EXECUTI   S WRONG
0A66A 05A4 6120            LDKL    A1,
      05A5 FFFF
0A66B 05A6 6141            ST      A1,DVHORD
      05A7 0000 F
0A66C 05A9 6100            LDK     A1,0
0A66D 05AB 6201            LDK     A2,1
0A66E 05AD C643    DVRST7  DV      DVHORD
      05AE 0002 F
0A66F 05B0 5222            RF(2)   *+4                 JP IF  CR RESULT = 2
0A670 05B2 207F            HLT                 *** CR RESULT OF INSTRUCTION DVRST7 WAS WRONG
0A671 05B3 EA20            CHK     A2,-1
      05B4 FFFF
0A672 05B6 5202            RF(0)   *+4                 JP IF CR RESULT = 0
0A673 05B8 207F            HLT                 *** INSTRUCTION DVRST7 EXECUTION WAS WRONG
0A674 05B9 E920            CHK     A1,0
      05BA 0000
0A675 05BC 5202            RF(0)   *+4                 JP IF CR RESULT = 0
0A676 05BE 207F            HLT                 *** INSTRUCTION DVRST7 EXECUTION WAS WRONG
0A677 05BF 8F20            ABL     DATST
      05C0 0000 F
```

```
                                EJECT
                ******  ******  ****  ************
                *       DVRPRO  DVR SUBROUTINE
                ******  ******  ****************
                DVRPRO  EQU     *
05CC  8141      ....    ST      A1,REGDVR
05CE  0700  F
                RF(7)   *+2             PASS IN ORDER TO DEBUGGING
                RF(7)   *+2             PASS IN ORDER TO DEBUGGING
05D4  5700  F   RF(7)   DV2
                ******  ******  ****************
                **
                *       DIVIDE  DIVISOR: 2 TO 16234
                **
                ******  ******  ****************

                DV2     EQU     *
05D6  5700      RF(7)   *+2             PASS IN ORDER TO DEBUGGING
05D8  0202      LDK     A2,2            (REG. A2) = DIVISOR
05DA  043A      LDK     A4,58           (REG. A4) = DIVIDEND
05DC  070D      LDK     A7,13           (REG. A7) = NUMBER IN ORDER TO CHECK
                *                       THE CONDITION REGISTER WHEN THIS ONE = 3
05DE  0601      LDK     A6,1            (REG. A6) = SHIFTS NUMBER TO EXECUTE
                *                       IN ORDER TO FIND THE CORRECT RESULT
05E0  F7A1      CF      A15,DV0         TO DIVIDE ORGTB TABLE BY 2
05E2  0000  F
                DV4     EQU     *
05E4  5700      RF(7)   *+2             PASS IN ORDER TO DEBUGGING
05E6  0204      LDK     A2,4            (REG. A2) = DIVISOR
05E8  0438      LDK     A4,56           (REG. A4) = DIVIDEND
05EA  070C      LDK     A7,12           (REG. A7) = NUMBER IN ORDER TO CHECK
                *                       THE CONDITION REGISTER WHEN THIS ONE = 3
05EC  0602      LDK     A6,2            (REG. A6) = SHIFTS NUMBER TO EXECUTE
                *                       IN ORDER TO FIND THE CORRECT RESULT
05EE  F7A1      CF      A15,DV0         TO DIVIDE ORGTB TABLE BY 4
05F0  0000  F
                DV8     EQU     *
05F2  5700      RF(7)   *+2             PASS IN ORDER TO DEBUGGING
05F4  0208      LDK     A2,8            (REG. A2) = DIVISOR
05F6  0436      LDK     A4,54           (REG. A4) = DIVIDEND
05F8  070B      LDK     A7,11           (REG. A7) = NUMBER IN ORDER TO CHECK
                *                       THE CONDITION REGISTER WHEN THIS ONE = 3
05FA  0603      LDK     A6,3            (REG. A6) = SHIFTS NUMBER TO EXECUTE
                *                       IN ORDER TO FIND THE CORRECT RESULT
05FC  F7A1      CF      A15,DV0         TO DIVIDE ORGTB TABLE BY 8
05FE  0000  F
                DV16    EQU     *
0600  5700      RF(7)   *+2             PASS IN ORDER TO DEBUGGING
```

```
00721  0602  0210          LDK   A2,
0072   0604  0434          LDK   A4,
00723  0606  070A          LDK   A7,1          (REG. A2) = DIVISOR
                                               (REG. A4) = DIVIDEND
                                               (REG. A7) = NUMBER IN ORDER    CHECK
00724                      *                   THE CONDITION REGISTER WHEN THIS ONE = 3
00725  0608  0604          LDK   A6,4          (REG. A6) = SHIFTS NUMBER TO EXECUTE
00726                      *                   IN ORDER TO FIND THE CORRECT RESULT
00727  060A  F7A1          CF    A15,DV0       TO DIVIDE ORGTB TABLE BY 16
       060C  0000  F
00728                DV32  EQU   *
00729  060E  5700          RF(7) *+2           PASS IN ORDER TO DEBUGGING
00730  0610  0220          LDK   A2,32         (REG. A2) = DIVISOR
00731  0612  0432          LDK   A4,50         (REG. A4) = DIVIDEND
00732  0614  0709          LDK   A7,9          (REG. A7) = NUMBER IN ORDER TO CHECK
00733                      *                   THE CONDITION REGISTER WHEN THIS ONE = 3
00734  0616  0605          LDK   A6,5          (REG. A6) = SHIFTS NUMBER TO EXECUTE
00735                      *                   IN ORDER TO FIND THE CORRECT RESULT
00736  0618  F7A1          CF    A15,DV0       TO DIVIDE ORGTB TABLE BY 32
       061A  0000  F
00737                DV64  EQU   *
00738  061C  5700          RF(7) *+2           PASS IN ORDER TO DEBUGGING
00739  061E  0240          LDK   A2,64         (REG. A2) = DIVISOR
00740  0620  0430          LDK   A4,48         (REG. A4) = DIVIDEND
00741  0622  0708          LDK   A7,8          (REG. A7) = NUMBER IN ORDER TO CHECK
00742                      *                   THE CONDITION REGISTER WHEN THIS ONE = 3
00743  0624  0606          LDK   A6,5          (REG. A6) = SHIFTS NUMBER TO EXECUTE
00744                      *                   IN ORDER TO FIND THE CORRECT RESULT
00745  0626  F7A1          CF    A15,DV0       TO DIVIDE ORGTB TABLE BY 64
       0628  0000  F
00746                DV128 EQU   *
00747  062A  5700          RF(7) *+2           PASS IN ORDER TO DEBUGGING
00748  062C  0280          LDK   A2,128        (REG. A2) = DIVISOR
00749  062E  042E          LDK   A4,46         (REG. A4) = DIVIDEND
00750  0630  0707          LDK   A7,7          (REG. A7) = NUMBER IN ORDER TO CHECK
00751                      *                   THE CONDITION REGISTER WHEN THIS ONE = 3
00752  0632  0607          LDK   A6,7          (REG. A6) = SHIFTS NUMBER TO EXECUTE
00753                      *                   IN ORDER TO FIND THE CORRECT RESULT
00754  0634  F7A1          CF    A15,DV0       TO DIVIDE ORGTB TABLE BY 128
       0636  0000  F
00755                DV256 EQU   *
00756  0638  5700          RF(7) *+2           PASS IN ORDER TO DEBUGGING
00757  063A  8220          LDKL  A2,256        (REG. A2) = DIVISOR
       063C  0100
00758  063E  042C          LDK   A4,44         (REG. A4) = DIVIDEND
00759  0640  0705          LDK   A7,5          (REG. A7) = NUMBER IN ORDER TO CHECK
00760                      *                   THE CONDITION REGISTER WHEN THIS ONE = 3
00761  0642  0608          LDK   A6,8          (REG. A6) = SHIFTS NUMBER TO EXECUTE
00762                      *                   IN ORDER TO FIND THE CORRECT RESULT
00763  0644  F7A1          CF    A15,DV0       TO DIVIDE ORGTB TABLE BY 256
       0646  0000  F
00764                DV512 EQU   *
```

```
00765  0648  5700            RF(7)   *+2          PASS IN ORDER TO DEBUGGING
00766  064A  8220            LDKL    A2,          (REG. A2) = DIVISOR
       064C  0200
00767  064E  042A            LDK     A4,42        (REG. A4) = DIVIDEND
00768  0650  0705            LDK     A7,5         (REG. A7) = NUMBER IN ORDER TO CHECK
00769                  *                           THE CONDITION REGISTER WHEN THIS ONE = 3
00770  0652  0609            LDK     A6,9         (REG. A6) = SHIFTS NUMBER TO EXECUTE
00771                  *                           IN ORDER TO FIND THE CORRECT RESULT
00772  0654  F7A1            CF      A15,DV0      TO DIVIDE ORGTB TABLE BY 512
       0656  0000  F
00773                  DV1024  EQU     *
00774  0658  5700            RF(7)   *+2          PASS IN ORDER TO DEBUGGING
00775  065A  8220            LDKL    A2,1024      (REG. A2) = DIVISOR
       065C  0400
00776  065E  0428            LDK     A4,40        (REG. A4) = DIVIDEND
00777  0660  0704            LDK     A7,4         (REG. A7) = NUMBER IN ORDER TO CHECK
00778                  *                           THE CONDITION REGISTER WHEN THIS ONE = 3
00779  0662  060A            LDK     A6,10        (REG. A6) = SHIFTS NUMBER TO EXECUTE
00780                  *                           IN ORDER TO FIND THE CORRECT RESULT
00781  0664  F7A1            CF      A15,DV0      TO DIVIDE ORGTB TABLE BY 1024
       0666  0000  F
00782                  DV2048  EQU     *
00783  0668  5700            RF(7)   *+2          PASS IN ORDER TO DEBUGGING
00784  066A  8220            LDKL    A2,2048      (REG. A2) = DIVISOR
       066C  0800
00785  066E  0426            LDK     A4,38        (REG. A4) = DIVIDEND
00786  0670  0703            LDK     A7,3         (REG. A7) = NUMBER IN ORDER TO CHECK
00787                  *                           THE CONDITION REGISTER WHEN THIS ONE = 3
00788  0672  060B            LDK     A6,11        (REG. A6) = SHIFTS NUMBER TO EXECUTE
00789                  *                           IN ORDER TO FIND THE CORRECT RESULT
00790  0674  F7A1            CF      A15,DV0      TO DIVIDE ORGTB TABLE BY 2048
       0676  0000  F
00791                  DV4096  EQU     *
00792  0678  5700            RF(7)   *+2          PASS IN ORDER TO DEBUGGING
00793  067A  8220            LDKL    A2,4096      (REG. A2) = DIVISOR
       067C  1000
00794  067E  0424            LDK     A4,36        (REG. A4) = DIVIDEND
00795  0680  0702            LDK     A7,2         (REG. A7) = NUMBER IN ORDER TO CHECK
00796                  *                           THE CONDITION REGISTER WHEN THIS ONE = 3
00797  0682  060C            LDK     A6,12        (REG. A6) = SHIFTS NUMBER TO EXECUTE
00798                  *                           IN ORDER TO FIND THE CORRECT RESULT
00799  0684  F7A1            CF      A15,DV0      TO DIVIDE ORGTB TABLE BY 4096
       0686  0000  F
00800                  DV8192  EQU     *
00801  0688  5700            RF(7)   *+2          PASS IN ORDER TO DEBUGGING
00802  068A  8220            LDKL    A2,8192      (REG. A2) = DIVISOR
       068C  2000
00803  068E  0422            LDK     A4,34        (REG. A4) = DIVIDEND
00804  0690  0701            LDK     A7,1         (REG. A7) = NUMBER IN ORDER TO CHECK
00805                  *                           THE CONDITION REGISTER WHEN THIS ONE = 3
```

```
00806  0892  0540          LDK     A5,1
00807                *
00808  0894  F7A1          CF      A15,
       0895  0000  F
00809                DV1638  EQU     *
00810  0898  5700          RF(7)   *+2
00811  089A  8220          LDKL    A2,16384
       089C  4000
00812  089E  0420          LDK     A4,32
00813  08A0  060E          LDK     A6,14
00814                *
00815  08A2  F7A1          CF      A15,DV0
       08A4  0000  F
00815  08A6  F03E          RTN     A15
```

(REG. A5) = SHIFTS NUMBER T XECUTE
IN ORDER TO FIND THE CORREC ESULT
TO DIVIDE ORGTB TABLE BY 81

PASS IN ORDER TO DEBUGGING
(REG. A2) = DIVISOR
(REG. A4) = DIVIDEND
(REG. A6) = SHIFTS NUMBER TO EXECUTE
IN ORDER TO FIND THE CORRECT RESULT
TO DIVIDE ORGTB TABLE BY 16384
RETURN

```
00817                          EJECT
00818
00819                  ****** ****** ****. **********
00820                  **
00821                  *       DVS    DIVIDE COMMON ROUTINE
00822                  **
00823                  ****** ****** ******************
00824  06A8  0000     FDVR    DATA   0                 DIVIDE FLAG. IF = 0 DV INSTRUCTION
00825                  *                                        IF = 1 DVR INSTRUCTION
00826                  *                                        IF = -1 DVK INSTRUCTION
00827  06AA            REGDVR  RES    1                 REGISTER DIV. USED
00828  06AC            DVRORD  RES    1                 CONTAIN THE ACTUAL DIVISOR
00829
00830                  DVS     EQU    *
00831  06AE  87C1              ST     A15,SAVA15        SAVE RETURN
       06B0  014C
00832  06B2  8141              ST     A1,FDVR
       06B4  06A8
00833  06B6  8241              ST     A2,DVRORD
       06B8  06AC
00834  06BA  033C              LDK    A3,60
00835  06BC  8641              ST     A6,NBNSH1
       06BE  0174
00836                  DVS00   EQU    *
00837  06C0  EC20              CRK    A4,28
       06C2  001C
00838  06C4  5100  F           RF(1)  DVS0              JP IF (REG, A4) > 14
00839  06C6  0100              LDK    A1,0
00840  06C8  6250              LD     A2,ORGT83,A4
       06CA  83F2
00841  06CC  5700  F           RF(7)  DVS000
00842                  DVS0    EQU    *
00843  06CE  814C              LD     A1,ORGT83-32,A3
       06D0  0002
00844  06D2  0200              LDK    A2,0
00845                  DVS000  EQU    *
00846  06D4  8441              ST     A4,SAVA4          SAVE REGISTER A4
       06D6  014A
00847  06D8  6BC0              LD     A6,FDVR
       06DA  06A8
00848  06DC  5100  F           RF(1)  DVRTR0            JP IF DVR INSTRUCTION
00849  06DE  5200  F           RF(2)  DVKTR0            JP IF DVR INSTRUCTION
00850  06E0  C840      DVS1    DV     DVRORD            DIVIDE BY (DVRORD)
       06E2  06AC
00851  06E4  5700              RF(7)  *+2               PASS IN ORDER TO DEBUGGING
00852  06E6  5700              RF(7)  *+2               PASS IN ORDER TO DEBUGGING
00853  06E8  5320  F           RF(5)  DVS1A             JP IF CR RESULT = 3
00854  06EA  5102              RF(1)  *+4               JP IF CR RESULT = 1
00855  06EC  227F              HLT                     *** CR RESULT OF DIVISION DVS1 WAS WRONG
00856  06EE  E920              CRK    A1,X'0000'
```

```
                 06F0  9000
00857  06F2  6002          RF(0)   *+4          JP IF (REG, A1) = X'C000'
00858  06F4  207F          HLT                  *** INSTRUCTION DVS1 EXECUTION W.  WRONG
00859  06F6  6550          LD      A6,ORGT33,A4
                 06F8  06F2
00860  06FA  6540          LD      A5,N8NSM1
                 06FC  0174
00861                DVS1B   EQU     *
00862  06FE  3EE1          SRC1    A6
00863  0700  EE20          CWK     A6,X'4000'
                 0702  4000
00864  0704  5000  F       RF(0)   DVS1D        JP IF (REG, A6) = X'4000'
00865  0706  9520          ADKL    A5,-1
                 0708  FFFF
00866  070A  590E          RB(1)   DVS1B        JP IF (REG, A5) > 0
00867                DVS1D   EQU     *
00868  070C  6440          LD      A4,SAVA4     RESTORE REGISTER A4
                 070E  014A
00869  0710  EE08          CWR     A6,A2
00870  0712  5002          RF(0)   *+4          JP IF (REG, A2) = (REG, A6)
00871  0714  207F          HLT                  *** INSTRUCTION DVS1 EXECUTION WAS WRONG
00872  0716  EE20          CWK     A6,X'0001'
                 0718  0001
00873  071A  5000  F       RF(0)   EXITDV       JP IF (REG,A6) = X'0001'
00874                DVS1C   EQU     *
00875  071C  9720          ADKL    A7,-1
                 071E  FFFF
00876  0720  9320          ADKL    A3,-2
                 0722  FFFE
00877  0724  9420          ADKL    A4,-2
                 0726  FFFE
00878  0728  590A          RB(1)   DVS00        JP IF (REG, A4) > 0
00879                EXITDV  EQU     *
00880  072A  67C0          LD      A15,SAVA15   RESTORE A15
                 072C  014C
00881  072E  F03E          RTN     A15          RETURN
00882                DVS1A   EQU     *
00883  0730  6440          LD      A4,SAVA4
                 0732  014A
00884  0734  EC20          CWK     A4,20
                 0736  001C
00885  0738  5102          RF(1)   *+4          JP IF (REG, A4) > 14
00886  073A  207F          HLT                  *** CR RESULT OF INSTRUCTION DVS1 WAS WRONG
00887  073C  5F22          RB(7)   DVS1C
```

```
                        EJECT
                ******  ******  ****  ***********
                **
                *       DVRTR0  DVR TREATMENT
                **
                ******  ******  ******************

                DVRTR0  EQU     *
0873E 3600              LDK     A6,0
0740 8540               LD      A5,REGDVR
0742 06AA
                DVRTR   EQU     *
0744 1641               ADK     A6,1
0746 9320               ADKL    A5,-1
0748 FFFF
074A 5C28               RB(4)   DVRTR           JP TF (REG, A5) # 0
074C 3E41               SLA1    A6
074E 8518               LDR     A5,A6
0750 AE20               ORKL    A6,X'C800'
0752 C800
0754 8641               ST      A5,DVRTR1
0756 0200  F
0758 3D01               SLA1    A5
075A 3D21               SLA1    A5
075C 3D21               SLA1    A5
075E 3D21               SLA1    A5
0760 3D21               SLA1    A5
0762 3D21               SLA1    A5
0764 AD20               ORKL    A5,X'8040'
0766 8040
0768 8541               ST      A5,LDREG
076A 0020  F
076C 0000       LDREG   DATA    0
076E 064C               DATA    DVWORD
0770 0000       DVRTR1  DATA    0               DIVIDE REGISTERS/REGISTERS: A1 TO A15
0772 5F30               RB(7)   DVS1+4

                ******  ******  ******************
                **
                *       DVKTR0  DVK TREATMENT
                **
                ******  ******  ******************

                DVKTR0  EQU     *
0774 8540               LD      A5,DVWORD
0776 06AC
0778 8541               ST      A5,DVKTR1+2
077A 0020  F
077C C820       DVKTR1  DATA    X'C820'         DIVIDE WITH CONSTANT
```

```
00930  077E  0002  DATA  0          CONSTANT = (DYWORD)
00931  1782  6F9E  RB(7)  DVS.
```

```
                                    EJECT
0782  0000  ZERO0   DATA    X'0000'
0784  0000          DATA    X'0000'
0786  FFFF  ONES0   DATA    X'FFFF'
0788  7FFF          DATA    X'7FFF'
078A  1111  FOUR11  DATA    X'1111'
078C  1111          DATA    X'1111'
078E  2222  FOUR22  DATA    X'2222'
0790  2222          DATA    X'2222'
0792  4444  FOUR44  DATA    X'4444'
0794  4444          DATA    X'4444'
0796  8888  FOUR88  DATA    X'8888'
0798  8888          DATA    X'8888'
079A  7FFF  MAX1    DATA    X'7FFF'
079C  7FFF          DATA    X'7FFF'
*******  *******  ****************
**
*       DOUBLE ADDITION TEST
**
*******  *******  ****************

DATST   EQU     *
079E  0120          LDK     A1,0
07A0  87A0          LDKL    A15,STKP
07A2  0140
07A4  F7A1          CF      A15,DACOM       TO DOUBLE ADD COMON ROUTINE
07A6  0000  F
07A8  8F20          ABL     DARTST
07AA  0280  F

*******  *******  ****************
**
*       DOUBLE ADDITION REGISTER TEST
**
*******  *******  ****************

DARTST  EQU     *
07AC  0101          LDK     A1,1
07AE  87A0          LDKL    A15,STKP
0780  0140
0782  F7A1          CF      A15,DACOM       TO DOUBLE ADD COMON ROUTINE
0784  0000  F
0786  8F20          ABL     DAKTST
0788  0000  F

*******  *******  ****************
**
*       DAK     DOUBLE ADD WITH CONSTANT
```

```
00976                    **
00977                    ******  ******  ****  **********
00978
00979                    DAKTST  EQU     *
00980  078A  5120               LDKL    A1,X'FFFF'
       078C  FFFF
00981  078E  87A0               LDKL    A15,STKP
       0790  0140
00982  0792  F7A1               CF      A15,DACOM       TO DOUBLE ADD COMMON ROUTINE
       0794  0004  F
00983  0796  8F20               ABL     DSTST
       0798  0000  F
```

```
                        EJECT
                *******  *******  ***  ************
                **
                *       DACOM   DOUBLE ADD COMMON ROUTINE
                **
                *******  *******  ******************
07CA            FDAR    RES     1               DOUBLE ADD FLAG, IF = 0, DA INSTRUCTION
                *                                       IF = 1, DAR INSTRUCTION
                *                                       IF =-1, DAK INSTRUCTION
                DACOM   EQU     *
07CC 87C1               ST      A15,SAVA15      SAVE REGISTER A15
07CE 814C
07D0 8141               ST      A1,FDAR
07D2 07CA
07D4 6120               LDKL    A1,X'FFFF'
07D6 FFFF
07D8 6220               LDKL    A2,X'7FFF'
07DA 7FFF
07DC 6340               LD      A3,FDAR
07DE 07CA
07E0 5100 F             RF(1)   TRDAR0          JP IF DAR INSTRUCTION
07E2 5200 F             RF(2)   TRDAK0          JP IF DAK INSTRUCTION
07E4 0040       DA0     DA      ZERO0
07E6 0762
07E8 5202               RF(2)   *+4             JP IF CR RESULT = 2
07EA 207F               HLT                     *** CR RESULT OF INSTRUCTION DA0  WAS WRONG
07EC E920               CRK     A1,X'FFFF'
07EE FFFF
07F0 5402               RF(0)   *+4             JP IF (REG. A1) = X'FFFF'
07F2 207F               HLT                     *** INSTRUCTION DA0  EXECUTION WAS WRONG
07F4 EA20               CRK     A2,X'7FFF'
07F6 7FFF
07F8 5002               RF(0)   *+4             JP IF (REG. A2) = X'7FFF'
07FA 207F               HLT                     *** INSTRUCTION DA0  EXECUTION WAS WRONG
07FC 6100               LDK     A1,0
07FE 6200               LDK     A2,0
0800 6540               LD      A5,FDAR
0802 07CA
0804 5100 F             RF(1)   TRDAR1          JP IF DAR INSTRUCTION
0806 5200 F             RF(2)   TRDAK1          JP IF DAK INSTRUCTION
0808 0240       DA1     DA      ONES0
080A 0766
080C 5202               RF(2)   *+4             JP IF CR RESULT = 2
080E 207F               HLT                     *** CR RESULT OF INSTRUCTION DA1  WAS WRONG
0810 E920               CRK     A1,X'FFFF'
0812 FFFF
0814 5002               RF(0)   *+4             JP IF (REG. A1) = X'FFFF'
0816 207F               HLT                     *** INSTRUCTION DA1  EXECUTION WAS WRONG
```

```
01023  0815  EA20          CHK   A2,X'7FFF'
       081A  7FFF
01024  081C  5002          RF(0)  *+4                  JP IF (REG. A2) = X'7FFF'
01025  081E  207F          HLT                  ***  INSTRUCTION DA1   EXECUTION WAS WRONG
01026  0820  8140          LD    A1,FOUR11
       0822  078A
01027  0824  8240          LD    A2,FOUR11
       0826  078A
01028  0828  8340          LD    A3,FDAR
       082A  07CA
01029  082C  5100  F       RF(1)  TRDAR2               JP IF DAR INSTRUCTION
01030  082E  5200  F       RF(2)  TRDAK2               JP IF DAK INSTRUCTION
01031  0830  0040    DA2   DA     FOUR11
       0832  078A
01032  0834  5102          RF(1)  *+4                  JP IF CR RESULT = 1
01033  0836  207F          HLT                  ***  CR RESULT OF INSTRUCTION DA2   WAS WRONG
01034  0838  E920          CHK   A1,X'2222'
       083A  2222
01035  083C  5002          RF(0)  *+4                  JP IF (REG. A1) = X'2222'
01036  083E  207F          HLT                  ***  INSTRUCTION DA2   EXECUTION WAS WRONG
01037  0840  EA20          CHK   A2,X'2222'
       0842  2222
01038  0844  5002          RF(0)  *+4                  JP IF (REG. A2) = X'2222'
01039  0846  207F          HLT                  ***  INSTRUCTION DA2   EXECUTION WAS WRONG
01040  0848  8140          LD    A1,FOUR22
       084A  078E
01041  084C  8240          LD    A2,FOUR22
       084E  078E
01042  0850  8340          LD    A3,FDAR
       0852  07CA
01043  0854  5100  F       RF(1)  TRDAR4               JP IF DAR INSTRUCTION
01044  0856  5200  F       RF(2)  TRDAK4               JP IF DAK INSTRUCTION
01045  0858  0040    DA4   DA     FOUR22
       085A  078E
01046  085C  5102          RF(1)  *+4                  JP IF CR RESULT = 1
01047  085E  207F          HLT                  ***  CR RESULT OF INSTRUCTION DA4   WAS WRONG
01048  0860  E920          CHK   A1,X'4444'
       0862  4444
01049  0864  5002          RF(0)  *+4                  JP IF (REG. A1) = X'4444'
01050  0866  207F          HLT                  ***  INSTRUCTION DA4   EXECUTION WAS WRONG
01051  0868  EA20          CHK   A2,X'4444'
       086A  4444
01052  086C  5002          RF(0)  *+4                  JP IF (REG. A2) = X'4444'
01053  086E  207F          HLT                  ***  INSTRUCTION DA4   EXECUTION WAS WRONG
01054  0870  8140          LD    A1,FOUR44
       0872  0792
01055  0874  8240          LD    A2,FOUR44
       0876  0792
01056  0878  8340          LD    A3,FDAR
       087A  07CA
```

```
01057  087C  5100  F          RF(1)    TRDA              JP IF DAR INSTRUCTION
01058  087E  5200  F          RF(2)    TRDA              JP IF DAK INSTRUCTION
01059  0880  0240      DA8    DA       FOUR
       0882  0792
01060  0884  5302             RF(3)    *+4               JP IF CR RESULT = 3
01061  0886  207F             HLT                 ***    CR RESULT OF INSTRUCTION DA8    WAS WRONG
01062  0888  E920             CMK      A1,X'8889'
       088A  8889
01063  088C  5202             RF(2)    *+4               JP IF (REG. A1) = X'8889'
01064  088E  207F             HLT                 ***    INSTRUCTION DA8    EXECUTION WAS WRONG
01065  0890  EA20             CMK      A2,X'8888'
       0892  8888
01066  0894  5202             RF(2)    *+4               JP IF (REG. A2) = X'8888'
01067  0896  207F             HLT                 ***    INSTRUCTION DA8    EXECUTION WAS WRONG
01068  0898  5140             LD       A1,FOUR88
       089A  0798
01069  089C  5240             LD       A2,FOUR88+2
       089E  0798
01070  08A0  5340             LD       A3,FDAR
       08A2  07DA
01071  08A4  5100  F          RF(1)    TDAR16            JP IF DAR INSTRUCTION
01072  08A6  5200  F          RF(2)    TDAK16            JP IF DAK INSTRUCTION
01073  08A8  0240      DA16   DA       FOUR88
       08AA  0798
01074  08AC  5302             RF(3)    *+4               JP IF CR RESULT = 3
01075  08AE  207F             HLT                 ***    CR RESULT OF INSTRUCTION DA16   WAS WRONG
01076  08B0  E920             CMK      A1,X'1110'
       08B2  1110
01077  08B4  5202             RF(2)    *+4               JP IF (REG. A1) = X'1110'
01078  08B6  207F             HLT                 ***    INSTRUCTION DA16   EXECUTION WAS WRONG
01079  08B8  EA20             CMK      A2,X'1110'
       08BA  1110
01080  08BC  5302             RF(2)    *+4               JP IF REG. A2) = X'1110'
01081  08BE  207F             HLT                 ***    INSTRUCTION DA16   EXECUTION WAS WRONG
01082  08C0  8F20             ABL      CODARK
       0802  0020  F
```

```
                                EJECT
                        **
                        *       TRDAR0  FROM DA0
                        **
                TRDAR0  EQU     *
0BC4  0400              LDK     A4,0
0BC6  0520              LDK     A5,0
0BC8  0210      DAR0    DAR     A4
0BCA  5FE4              RB(7)   DA0+4
                        **
                        *       TRDAK0  FROM DA0
                        **
                TRDAK0  EQU     *
0BCC  0020      DAK0    DATA    X'0020'
0BCE  0020              DATA    X'0020'
0BD0  0000              DATA    X'0000'
0BD2  5FEC              RB(7)   DA0+4
                        **
                        *       TRDAR1  FROM DA1
                        **
                TRDAR1  EQU     *
0BD4  8320              LDKL    A3,X'FFFF'
0BD6  FFFF
0BD8  8420              LDKL    A4,X'7FFF'
0BDA  7FFF
0BDC  0230      DAR1    DAR     A3
0BDE  5FD4              RB(7)   DA1+4
                        **
                        *       TRDAK1  FROM DA1
                        **
                TRDAK1  EQU     *
0BE0  0020      DAK1    DATA    X'0020'
0BE2  FFFF              DATA    X'FFFF'
0BE4  7FFF              DATA    X'7FFF'
0BE6  5FDC              RB(7)   DA1+4
                        **
                        *       TRDAR2  FROM DA2
                        **
                TRDAR2  EQU     *
0BE8  8520              LDKL    A5,X'1111'
0BEA  1111
0BEC  8620              LDKL    A6,X'1111'
0BEE  1111
0BF0  0014      DAR2    DAR     A5
0BF2  5FC8              RB(7)   DA2+4
                        **
                        *       TRDAK2  FROM DA2
                        **
                TRDAK2  EQU     *
```

```
01129  03F4  0020    DAK2    DATA    X'02
01130  03F6  1111            DATA    X'11
01131  03F8  1111            DATA    X'11
01132  03FA  5FC8            RB(7)   DA2+4
01133                *
01134                *       TRDAR4  FROM DA4
01135                *
```

```
                                EJECT
                        TRDAR4  EQU     *
0113A 08FC 8720                 LDKL    A7,X'2222'
      08FE 2222
01140 0900 8BA0                 LDKL    A8,X'2222'
      0902 2222
01141 0904 D01C         DAR4    DAR     A7
01142 0906 5FAC                 RB(7)   DA4+4
01143                   **
01144                   *       TRDAK4  FROM DA4
01145                   **
01146                   TRDAK4  EQU     *
01147 0908 D020         DAK4    DATA    X'D020'
01148 090A 2222                 DATA    X'2222'
01149 090C 2222                 DATA    X'2222'
0114A 090E 5F54                 RB(7)   DA4+4
0114B                   **
0114C                   *       TRDAR8  FROM DA8
0114D                   **
0114E                   TRDAR8  EQU     *
0114F 0910 81A0                 LDKL    A9,X'4444'
      0912 4444
01150 0914 82A0                 LDKL    A10,X'4444'
      0916 4444
01151 0918 D036         DAR8    DAR     A9
01152 091A 5F98                 RB(7)   DA8+4
01153                   **
01154                   *       TRDAK8  FROM DA8
01155                   **
01156                   TRDAK8  EQU     *
01157 091C D020         DAK8    DATA    X'D020'
01158 091E 4444                 DATA    X'4444'
01159 0920 4444                 DATA    X'4444'
0115A 0922 5FA6                 RB(7)   DA8+4
0115B                   **
0115C                   *       TDAR16  FROM DA16
0115D                   **
0115E                   TDAR16  EQU     *
0115F 0924 83A0                 LDKL    A11,X'8888'
      0926 8888
01160 0928 84A0                 LDKL    A12,X'8888'
      092A 8888
01161 092C D00E         DAR16   DAR     A11
01162 092E 5F84                 RB(7)   DA16+4
01163                   **
01164                   *       TDAK16  FROM DA16
01165                   **
01166                   TDAK16  EQU     *
01167 0930 D020         DAK16   DATA    X'D020'
```

```
21180  0032  0000          DATA    X'0
21181  0034  0000          DATA    X'0
21182  0036  5F8C          RB(7)   DA16+
```

```
                                EJECT
                        ******  ******  ****.  /**********.
                        **
                        *       CODARK  DOUSLE AUD COMMON ROUTINE (CONTINUATION)
                        **
                        ******  ******  ******************.
                        CODARK  EQU     *
                                LDKL    A1,X'7FFF'
                                LDKL    A2,X'7FFF'
                                LD      A3,FDAR
                                RF(1)   TDAOV1          JP IF DAR INSTRUCTION
                                RF(2)   TDAKV1          JP IF DAK INSTRUCTION
                        DAOVE1  DA      ORGTB1
                                RF(3)   *+4             JP IF CR RESULT = 3
                                HLT                     *** CR RESULT OF INSTRUCTION DAOVE1 WAS WRONG
                                LDK     A1,X'0000'
                                LDK     A2,X'0001'
                                LD      A3,FDAR
                                RF(1)   TDAOV2          JP IF DAR INSTRUCTION
                                RF(2)   TDAKV2          JP IF DAK INSTRUCTION
                        DAOVE2  DA      MAX1
                                RF(3)   *+4             JP IF CR RESULT = 2
                                HLT                     *** CR RESULT OF INSTRUCTION DAOVE2 WAS WRONG
                                LDKL    A1,X'FFFF'
                                LDKL    A2,X'7FFF'
                                LD      A3,FDAR
                                RF(1)   TDAOV3          JP IF DAR INSTRUCTION
                                RF(2)   TDAKV3          JP IF DAK INSTRUCTION
                        DAOVE3  DA      ORGTB1
                                RF(0)   *+4             JP IF CR RESULT = 0
                                HLT                     *** CR RESULT OF INSTRUCTION DAOVE3 WAS WRONG
                                CMK     A1,X'0000'
                                RF(0)   *+4             JP IF (REG. A1) = X'0000'
                                HLT                     *** INSTRUCTION DAOVE3 EXECUTION WAS WRONG
                                CMK     A2,X'0000'
                                RF(0)   *+4             JP IF (REG. A2) = X'0000'
                                HLT                     *** INSTRUCTION DAOVE3 EXECUTION WAS WRONG
```

```
 01221  0990  0100          LDK     A
 01222  0992  0201          LDK     A2,
 01223  0994  0340          LD      A3,
        0996  07CA
 01224  0998  5108  F       RF(1)   TDAOV4          JP IF DAR INSTRUCTION
 01225  099A  5208  F       RF(2)   TDAKV4          JP IF DAK INSTRUCTION
 01226  099C  0740    DAOVE4 DA      ONE50
        099E  0756
 01227  09A0  5002          RF(0)   **+4            JP IF CR RESULT = 0
 01228  09A2  207F          HLT                ***  CR RESULT OF INSTRUCTION DAOVE4 WAS WRONG
 01229  09A4  6920          CMK     A1,X'0000'
        09A6  0000
 01230  09A8  5002          RF(0)   **+4            JP IF (REG. A1) = X'0000'
 01231  09AA  207F          HLT                ***  INSTRUCTION DAOVE4 EXECUTION WAS WRONG
 01232  09AC  6A20          CMK     A2,X'0000'
        09AE  0000
 01233  09B0  5002          RF(0)   **+4            JP IF (REG. A2) = X'0000'
 01234  09B2  207F          HLT                ***  INSTRUCTION DAOVE4 EXECUTION WAS WRONG
 01235  09B4  67C0          LD      A15,SAVA15      RESTORE REGISTER A15
        09B6  014C
 01236  09B8  F83E          RTN     A15             RETURN
```

```
                                EJECT
                *
                *       TDA0V1 - FROM DA0VE1
                *
                TDA0V1  EQU     *
0983  85A0              LDKL    A13,X'0020'
0985  0020
0988  85A0              LDKL    A14,X'0001'
098A  0001
098C  D31E      DAR0V1  DAR     A13
098E  5F76              RB(7)   DA0VE1+4
                *
                *       TDAKV1 - FROM DA0VE1
                *
                TDAKV1  EQU     *
0990  0020      DAKV1   DATA    X'0020'
0992  0000              DATA    X'0000'
0994  0001              DATA    X'0001'
0996  5F7E              RB(7)   DA0VE1+4
                *
                *       TDA0V2 - FROM DA0VE2
                *
                TDA0V2  EQU     *
0998  85A0              LDKL    A14,X'7FFF'
099A  7FFF
099C  87A0              LDKL    A15,X'7FFF'
099E  7FFF
09A0  D31A      DAR0V2  DAR     A14
09A2  5F76              RB(7)   DA0VE2+4
                *
                *       TDAKV2 - FROM DA0VE2
                *
                TDAKV2  EQU     *
09A4  0022      DAKV2   DATA    X'0022'
09A6  7FFF              DATA    X'7FFF'
09A8  7FFF              DATA    X'7FFF'
09AA  5F7E              RB(7)   DA0VE2+4
                *
                *       TDA0V3 - FROM DA0VE3
                *
                TDA0V3  EQU     *
09AC  0400              LDK     A4,X'0002'
09AE  0501              LDK     A5,X'0001'
09B0  D010      DAR0V3  DAR     A4
09B2  5F6E              RB(7)   DA0VE3+4
                *
                *       TDAKV3 - FROM DA0VE3
                *
                TDAKV3  EQU     *
```

```
01283  09E6  D020        DAKV3   DATA    X'D020'
01284  09E8  0300                DATA    X'0300'
01285  09EA  0001                DATA    X'0001'
01286  09EC  5F76                RB(7)   DAOVE3+4
01287                     **
01288                     *       TDAKV4  FROM DAOVE4
01289                     **
01290                     TDAOV4  EQU     *
01291  09EE  83A0                LDKL    A11,X'FFFF'
       09F0  FFFF
01292  09F2  84A0                LDKL    A12,X'7FFF'
       09F4  7FFF
01293  09F6  D20E        DAROV4  DAR     A11
01294  09F8  5F5E                RB(7)   DAOVE4+4
01295                     **
01296                     *       TDAKV4  FROM DAOVE4
01297                     **
01298                     TDAKV4  EQU     *
01299  09FA  D020        DAKV4   DATA    X'D020'
01300  09FC  FFFF                DATA    X'FFFF'
01301  09FE  7FFF                DATA    X'7FFF'
01302  0A00  5F66                RB(7)   DAOVE4+4
```

```
                                EJECT
                ******  ******  ******  **********
                **
                *       DS      DOUBLE SUBTRACT TEST
                **
                ******  ******  ****************
DSTST           EQU     *
0A02  6100              LDK     A1,0
0A04  87A0              LDKL    A15,STKP
0A06  0140
                        CF      A15,DSCOM       TO DOUBLE SUBTRACT COMMON ROUTINE
0A08  F7A1
0A0A  0000  F
                        ABL     DSRTST
0A0C  6F20
0A0E  0000  F

                ******  ******  ****************
                **
                *       DSR     DOUBLE SUBTRACT REGISTER/REGISTERS TEST
                **
                ******  ******  ****************
DSRTST          EQU     *
0A10  0101              LDK     A1,1
0A12  87A0              LDKL    A15,STKP
0A14  0140
                        CF      A15,DSCOM       TO DOUBLE SUBTRACT COMMON ROUTINE
0A16  F7A1
0A18  0000  F
                        ABL     DSRTST
0A1A  6F20
0A1C  0000  F

                ******  ******  ****************
                **
                *       DSK     DOUBLE SUBTRACT WITH CONSTANT
                **
                ******  ******  ****************
DSKTST          EQU     *
0A1E  8120              LDKL    A1,X'FFFF'
0A20  FFFF
0A22  87A0              LDKL    A15,STKP
0A24  0140
                        CF      A15,DSCOM       TO DOUBLE SUBTRACT COMMON ROUTINE
0A26  F7A1
0A28  0000  F
0A2A  9041              IM      PASCT1
0A2C  0178
0A2E  5200  F           RF(2)   INCR32          JP IF (COUNTER 1: PASCT1 = X'8000'
0A32  0FCA              AB      X'CA'
INCR32          EQU     *
```

```
                        EJECT
                ******  ******  ******  **********
                **
                *       DS      DOUBLE SUBTRACT TEST
                **
                ******  ******  ****************

                DSTST   EQU     *
0A82 8102               LDK     A1,2
0A84 87A0               LDKL    A15,STKP
0A86 0140
                        CF      A15,DSCOM       TO DOUBLE SUBTRACT COMMON ROUTINE
0A88 F7A1
0A8A 0020 F
                        ABL     DSRTST
0A8C 8F20
0A8E 0000 F

                ******  ******  ******************
                **
                *       DSR     DOUBLE SUBTRACT REGISTER/REGISTERS TEST
                **
                ******  ******  ******************

                DSRTST  EQU     *
0A10 8101               LDK     A1,1
0A12 87A0               LDKL    A15,STKP
0A14 0140
                        CF      A15,DSCOM       TO DOUBLE SUBTRACT COMMON ROUTINE
0A16 F7A1
0A18 0000 F
                        ABL     DSRTST
0A1A 8F20
0A1C 0200 F

                ******  ******  ******************
                **
                *       DSK     DOUBLE SUBTRACT WITH CONSTANT
                **
                ******  ******  ******************

                DSKTST  EQU     *
0A1E 8120               LDKL    A1,X'FFFF'
0A20 FFFF
0A22 87A0               LDKL    A15,STKP
0A24 0140
0A26 F7A1               CF      A15,DSCOM       TO DOUBLE SUBTRACT COMMON ROUTINE
0A28 0000 F
0A2A 9041               IM      PASCT1
0A2C 0178
0A2E 5200 F             RF(2)   INCR32          JP IF (COUNTER 1: PASCT1 = X'8000'
0A30 8FCA               AB      X'CA'
                INCR32  EQU     *
```

```
01343  0A32  0100          LDK     A1,
0113   0A34  0141          ST      A1,  ,T1
       0A36  0170
01345  0A38  0041          IM      PASCT2
       0A3A  0178
01346  0A3C  0FCA          AB      X'CA'       LOOP ON TEST
01347  0A3E  287F          HLT                 HALT  HALT  HALT  HALT  HALT  HALT
01348
```

```
                        EJECT
                *******  ******  ****  **********
                **
                *       DSCOM   DOUBLE SUBTRACT COMMON ROUTINE
                **
                *******  ******  ******************

0A42            FDSR    RES     1               DOUBLE SUBTRACT FLAG. IF=0 DS INSTRUCTION
                *                                                     IF=1 DSR INSTRUCTION
                *                                                     IF=-1 DSK INSTRUCTION
                DSCOM   EQU     *
0A42 87C1               ST      A15,SAVA15
0A44 014C
0A46 6141               ST      A1,FDSR
0A48 0A40
0A4A 6123               LDKL    A1,X'8888'
0A4C 8888
0A4E 6223               LDKL    A2,X'0888'
0A50 0888
0A52 8340               LD      A3,FDSR
0A54 0A40
0A56 6100 F             RF(1)   TRDSR8          JP IF DSR INSTRUCTION
0A58 6200 F             RF(2)   TRDSK8          JP IF DSK INSTRUCTION
0A5A 0840       DS8     DS      FOUR8
0A5C 0795
0A5E 5002               RF(0)   **4             JP IF CR RESULT = 0
0A60 207F               HLT                     *** CR RESULT OF INSTRUCTION DS8  WAS WRONG
0A62 6920               CRK     A1,X'0000'
0A64 0000
0A66 5002               RF(0)   **4             JP IF (REG. A1) = X'0000'
0A68 207F               HLT                     *** INSTRUCTION DS8  EXECUTION WAS WRONG
0A6A EA20               CRK     A2,X'0000'
0A6C 0000
0A6E 5002               RF(0)   **4             JP IF (REG. A2) = X'0000'
0A70 207F               HLT                     *** INSTRUCTION DS8  EXECUTION WAS WRONG
0A72 6120               LDKL    A1,X'4444'
0A74 4444
0A76 6223               LDKL    A2,X'4444'
0A78 4444
0A7A 8343               LD      A3,FDSR
0A7C 0A40
0A7E 6140 F             RF(1)   TRDSR4          JP IF DSR INSTRUCTION
0A80 6240 F             RF(2)   TRDSK4          JP IF DSK INSTRUCTION
0A82 0844       DS4     DS      FOUR4
0A84 0792
0A86 5002               RF(0)   **4             JP IF CR RESULT = 0
0A88 207F               HLT                     *** CR RESULT OF INSTRUCTION DS4  WAS WRONG
0A8A 6920               CRK     A1,X'0000'
0A8C 0000
```

```
01386  0A9E  5002          RF(0)    **4          JP IF (REG, A1) = X'0000'
01387  0AA0  207F          HLT                   *** INSTRUCTION DS4  EXECUTION WRONG
01388  0AA2  EA20          CWK      A2,X'0000'
       0AA4  0000
01389  0AA6  5002          RF(0)    **4          JP IF (REG, A2) = X'0000'
01390  0AA8  207F          HLT                   *** INSTRUCTION DS4  EXECUTION WAS WRONG
01391  0AAA  6120          LDKL     1,X'2222'
       0AAC  2222
01392  0AAE  6220          LDKL     A2,X'2222'
       0AB0  2222
01393  0AB2  6340          LD       A3,FDSR
       0AB4  0A40
01394  0AB5  5140  F       RF(1)    TRDSR2       JP IF DSR INSTRUCTION
01395  0AB8  5280  F       RF(2)    TRDSK2       JP IF DSK INSTRUCTION
01396  0ABA  0540     DS2  DS       FOUR22
       0ABC  076E
01397  0ABE  5002          RF(0)    **4          JP IF CR RESULT = 0
01398  0AC0  207F          HLT                   *** CR RESULT OF INSTRUCTION DS2  WAS WRONG
01399  0AC2  E920          CWK      A1,X'0000'
       0AC4  0000
01400  0AC6  5002          RF(0)    **4          JP IF (REG, A1) = X'0000'
01401  0AC8  207F          HLT                   *** INSTRUCTION DS2  EXECUTION WAS WRONG
01402  0ACA  EA20          CWK      A2,X'0000'
       0ACC  0000
01403  0ACE  5002          RF(0)    **4          JP IF (REG, A2) = X'0000'
01404  0AD0  207F          HLT                   *** INSTRUCTION DS2  EXECUTION WAS WRONG
01405  0AD2  6120          LDKL     A1,X'1111'
       0AD4  1111
01406  0AD6  6220          LDKL     A2,X'1111'
       0AD8  1111
01407  0ADA  6340          LD       A3,FDSR
       0ADC  0A40
01408  0ADE  5100  F       RF(1)    TRDSR1       JP IF DSR INSTRUCTION
01409  0AE0  5200  F       RF(2)    TRDSK1       JP IF DSK INSTRUCTION
01410  0AE2  0540     DS1  DS       FOUR11
       0AE4  076A
01411  0AE6  5002          RF(0)    **4          JP IF CR RESULT = 0
01412  0AE8  207F          HLT                   *** CR RESULT OF INSTRUCTION DS1  WAS WRONG
01413  0AEA  E920          CWK      A1,X'0000'
       0AEC  0000
01414  0AEE  5002          RF(0)    **4          JP IF (REG, A1) = X'0000'
01415  0AF0  207F          HLT                   *** INSTRUCTION DS1  EXECUTION WAS WRONG
01416  0AF2  EA20          CWK      A2,X'0000'
       0AF4  0000
01417  0AF6  5002          RF(0)    **4          JP IF (REG, A2) = X'0000'
01418  0AF8  207F          HLT                   *** INSTRUCTION DS1  EXECUTION WAS WRONG
01419  0AFA  6F20          ABL      CODSRK
       0AFC  0000  F
```

```
                        EJECT
                *
                *       TRDSR8  FROM DS8
                *
                TRDSR8  EQU     *
0AEE 6320               LDKL    A3,X'8888'
0AF2 8883
0AF2 6420               LDKL    A4,X'8888'
0AF4 8888
0AF6 D80C       DSR8    DSR     A3
0AF8 5F9C               RB(7)   DS8+4
                *
                *       TRDSK8  FROM DS8
                *
                TRDSK8  EQU     *
0AFA D820       DSK8    DSK     X'8888',X'8888'
0AFC 8888
0AFE 8888
0B00 5FA4               RB(7)   DS8+4
                *
                *       TRDSR4  FROM DS4
                *
                TRDSR4  EQU     *
0B02 6523               LDKL    A5,X'4444'
0B04 4444
0B06 6620               LDKL    A6,X'4444'
0B08 4444
0B0A D814       DSR4    DSR     A5
0B0C 5F68               RB(7)   DS4+4
                *
                *       TRDSK4  FROM DS4
                *
                TRDSK4  EQU     *
0B0E D820       DSK4    DSK     X'4444',X'4444'
0B10 4444
0B12 4444
0B14 5F90               RB(7)   DS4+4
                *
                *       TRDSR2  FROM DS2
                *
                TRDSR2  EQU     *
0B16 8720               LDKL    A7,X'2222'
0B18 2222
0B1A 6BA0               LDKL    A8,X'2222'
0B1C 2222
0B1E D81C       DSR2    DSR     A7
0B20 5F74               RB(7)   DS2+4
                **
```

```
                        *       TRDSK2  FROM DS2
                        **
                TRDSK2  EQU     *
0822  0824       DSK2    DSK     X'2222',X'2222'
0824  2222
0826  2222
                        RB(7)   DS2+4
                        **
                        *       TRDSR1  FROM DS1
                        **
                TRDSR1  EQU     *
0824  81A9              LDKL    A9,X'1111'
0826  1111
0828  82A8              LDKL    A12,X'1111'
082A  1111
082C  DB06      DSR1    DSR     A9
082E  DF50              RB(7)   DS1+4
                        **
                        *       TRDSK1  FROM DS1
                        **
                TRDSK1  EQU     *
0830  0620      DSK1    DSK     X'1111',X'1111'
0832  1111
0834  1111
0836  DF08              RB(7)   DS1+4
```

```
                                EJECT
                        ******  ******  ****  **********
                        **
                        *       CODSRK  DOUBLE SUBTRACT COMMON ROUTINE (CONTINUATION)
                        **
                        ******  ******  ****************
                CODSRK  EQU     *
0836 8120               LDKL    A1,X'1111'
0838 1111
083A 8220               LDKL    A2,X'1111'
083C 1111
083E 8340               LD      A3,FDSR
0840 0A40
0842 510C F             RF(1)   TRDSS1          JP IF DSR INSTRUCTION
0844 520A F             RF(2)   TDSSK1          JP IF DSK INSTRUCTION
0846 0840       DSS1    DS      FOUR22
0848 670E
084A 5202               RF(2)   *+4             JP IF CR RESULT = 2
084C 207F               HLT                     *** CR RESULT OF INSTRUCTION DSS1 WAS WRONG
084E 8120               LDKL    A1,X'2222'
0850 2222
0852 8220               LDKL    A2,X'2222'
0854 2222
0856 8340               LD      A3,FDSR
0858 0A40
085A 5100 F             RF(1)   TRDSS2          JP IF DSR INSTRUCTION
085C 5208 F             RF(2)   TDSSK2          JP IF DSK INSTRUCTION
085E 0840       DSS2    DS      FOUR44
0860 0792
0862 5202               RF(2)   *+4             JP IF CR RESULT = 2
0864 207F               HLT                     *** CR RESULT OF INSTRUCTION DSS2 WAS WRONG
0866 8120               LDKL    A1,X'4444'
0868 4444
086A 8220               LDKL    A2,X'4444'
086C 4444
086E 8340               LD      A3,FDSR
0870 0A40
0872 5106 F             RF(1)   TRDSS3          JP IF DSR INSTRUCTION
0874 520C F             RF(2)   TDSSK3          JP IF DSK INSTRUCTION
0876 0840       DSS3    DS      FOUR88
0878 0798
087A 5302               RF(3)   *+4             JP IF CR RESULT = 3
087C 207F               HLT                     *** CR RESULT OF INSTRUCTION DSS3 WAS WRONG
087E 8120               LDK     A1,X'8888'
0880 8888               LDK     A2,X'4444'
0882 8220               LD      A3,FDSR
0884 0A40
0886 5100 F             RF(1)   TRDSS4          JP IF DSR INSTRUCTION
0888 5208 F             RF(2)   TDSSK4          JP IF DSK INSTRUCTION
```

```
01518  0B92  D940      DS54    DS      DRGT#1
       0B94  00F0
7151   0B96  32x2              RF(2)   *+4             JP IF CK RESULT = 2
01515  0B9A  207F              PLT                     *** CK RESULT OF INSTRUCTION DS54 WAS WRONG
01519  0B9A  87C0              LD      A15,SAVA15
       0B9C  014C
01520  0B9E  F03E              RTN     15              RETURN
```

```
                                EJECT
                        **
                        *       TRDSS1  FROM DSS1
                        **
                        TRDSS1  EQU     *
                                LDKL    11,X'2222'
                                LDKL    A12,X'2222'
                        DSSR1   DSR     A11
                                RB(7)   DSS1+4
                        **
                        *       TRDSS2  FROM DSS2
                        **
                        TRDSS2  EQU     *
                                LDKL    A13,X'4444'
                                LDKL    A14,X'4444'
                        DSSR2   DSR     A13
                                RB(7)   DSS2+4
                        **
                        *       TRDSS3  FROM DSS3
                        **
                        TRDSS3  EQU     *
                                LDKL    A14,X'6868'
                                LDKL    A15,X'6868'
                        DSSR3   DSR     A14
                                RB(7)   DSS3+4
                        **
                        *       TRDSS4  FROM DSS4
                        **
                        TRDSS4  EQU     *
                                LDK     A0,X'0001'
                        DSSR4   DSR     A2
                                RB(7)   DSS4+4
                        **
                        *       TDSSK1  FROM DSS1
                        **
                        TDSSK1  EQU     *
                        DSSK1   DSK     X'2222',X'2222'
                                RB(7)   DSS1+4
                        **
                        *       TDSSK2  FROM DSS2
                        **
```

```
01563                          TDSSK2   EQU   *
01564  0BB2 D820               DSSK2    DSK   X'4(  ,X'4444'
       0BB4 4444
       0BB6 4444
01565  0BB8 5F70                        RB(7)  DSS2+4
01566                          **
01567                          *        TDSSK3 FROM DSS3.
01568                          **
01569                          TDSSK3   EQU   *
01570  0BD4 D820               DSSK3    DSK   X'8888',X'0888'
       0BD6 8888
       0BD8 8888
01571  0BE0 5F80                        RB(7)  DSS3+4
01572                          **
01573                          *        TDSSK4 FROM DSS4.
01574                          **
01575                          TDSSK4   EQU   *
01576  0BE2 D820               DSSK4    DSK   X'4444',X'4444'
       0BE4 4444
       0BE6 4444
01577  0BE8 5F54                        RB(7)  DSS4+4
```

```
C1578
0157
0158.
```

EJECT

END

SYMBOL TABLE

[illegible]	[illegible]	R	ITROUT	0000	R	JAM	0000	A	ORGTB0	00CE	A			
ORGTB1	00F0	A	ORGTB3	00F2	A	ORGTB4	00F8	A	WHEMOR	0100	A			
CHEMOR	010C	A	ORGTB2	0112	A	MEMSTK	011A	A	STKP	0140	A			
MODFHL	0142	A	MLIAD	0144	A	INDEXN	0146	A	RLCNT	0148	A			
SAVA4	014A	A	SAVA16	014C	A	KBRMS	014E	A	STAKMS	0150	A			
STAKND	016C	A	MSIAD	016E	A	CCIAD	0170	A	KBRSM	0172	A			
KBRSM1	0174	A	PASCT1	0176	A	PASCT2	0178	A	MUL0	0180	A			
MU01	01A2	A	MU2	0256	A	MURTST	01CA	A	MURTST	010C	A			
MURG8	01E0	A	MURPRO	024C	A	MURG1	01E8	A	MURG9	01F0	A			
MURG10	01F8	A	MURG11	0200	A	MURG4	0208	A	MURG12	0210	A			
MURG5	0218	A	MURG13	0220	A	MURG6	0228	A	MURG14	0230	A			
MURG7	0238	A	MURG15	0240	A	DVTST	03FE	A	REGMUL	030C	A			
MU00	0310	A	MU4	0264	A	MU8	0272	A	MU16	0280	A			
MU32	028E	A	MU64	029C	A	MU128	02AA	A	MU256	02B8	A			
MU512	02C6	A	MU1024	02D5	A	MU2048	02E8	A	MU4096	02F8	A			
MULTR	030A	A	MULTM	030E	A	MUS00	0322	A	MURTR0	03AC	A			
MURTR0	035A	A	MUS1	0336	A	MUS1A	034A	A	MUS1B	0358	A			
MUS1C	0364	A	MUS1E	03A2	A	MUS1D	037E	A	MURTR	03B2	A			
MURTR1	03B2	A	LDREGM	03DE	A	MURTR1	03F4	A	DV00	0402	A			
DV01	0424	A	DV16	0442	A	DV2	0505	A	DVRTST	040A	A			
DVRTST	047C	A	DVRG8	04C0	A	DVRPRO	05CC	A	DVRG9	0488	A			
DVRG10	0492	A	DVRG3	0498	A	DVRG11	04A0	A	DVRG4	04A8	A			
DVRG12	04B0	A	DVRG5	0488	A	DVRG13	04C0	A	DVRG6	04C8	A			
DVRG14	04D0	A	DVRG7	04D8	A	DVRG15	04E0	A	DVDIF	04EC	A			
DVRG0	04AC	A	DVRST1	04FC	A	DVRST2	0522	A	DVRST3	0548	A			
DVRST4	0553	A	DVRST5	0578	A	DVRST6	059C	A	DVRST7	05B0	A			
DATST	07BE	A	REGDVR	05AA	A	DV8	05AE	A	DV4	05E4	A			
DV8	05F2	A	DV16	0600	A	DV32	060E	A	DV64	061C	A			
DV128	062A	A	DV256	0638	A	DV512	0648	A	DV1024	0656	A			
DV2048	0668	A	DV4096	0678	A	DVB192	0688	A	DV1638	0696	A			
FDVR	06A8	A	DV000	06C0	A	DV00	06CE	A	DVS000	06D4	A			
DVRTR0	076E	A	DVRTR0	0774	A	DVS1	06E0	A	DVS1A	0730	A			
DVS1B	06FE	A	DVS1D	073C	A	EXITDV	072A	A	DVS1C	071C	A			
DVRTR	0744	A	DVRTR1	0778	A	LDREG	075C	A	DVRTR1	077C	A			
ZERO0	0782	A	ONE50	0735	A	FOUR11	079A	A	FOUR22	078E	A			
FOUR44	0782	A	FOUR88	0795	A	MAX1	079A	A	DAOGM	07CC	A			
DARTST	07AC	A	DARTST	07BA	A	DSTST	0A02	A	FDAR	07DA	A			
TRDAR0	08C4	A	TRDAR8	08CC	A	DAW	07E4	A	TRDAR1	0804	A			
TRDAR1	08D0	A	DA1	0838	A	TRDAR2	08E8	A	TRDAR2	08F4	A			
DA2	0836	A	TRDAR4	08FC	A	TRDAR4	0960	A	DA4	0858	A			
TRDAR8	0910	A	TRDAR5	091C	A	DA8	0860	A	TDAR16	0924	A			
TDAR16	0938	A	DA16	08A8	A	CDDARK	0938	A	DAR0	08C8	A			
DAR2	08DC	A	DAR1	08EC	A	DAR1	08E0	A	DAR2	08F0	A			
DAR3	08F4	A	DAR4	0904	A	DAR4	0968	A	DAR8	0918	A			
DAR8	0910	A	DAR16	092C	A	DAR16	0930	A	TDADV1	09E6	A			
TDADV1	0902	A	DADVE1	0948	A	TDADV2	09CA	A	TDADV2	0908	A			
DADVE2	090C	A	TDADV3	090E	A	TDADV3	09E6	A	DADVE3	0974	A			
TDADV4	09CE	A	TDADV4	09EA	A	DADVE4	0998	A	DADV1	09BE	A			

```
DARV1   09C2  A   DAROV2  0902  A   DARV2   09?   A   DAROV3  09L2  A
DARV3   09E0  A   DAROV4  09F8  A   DARV4   09    A   DSCOM   0A42  A
DSKTST  0A1D  A   DSKTST  0A1E  A   INCR32  0A3C  A   FDSR    0A4M  A
TRDSR3  0A6E  A   TRDSK3  0AFA  A   DS3     0A5A  A   TRDSR4  0B02  A
TRDSK4  0B0E  A   DS4     0A62  A   TRDSR2  0B16  A   TRDSK2  0B22  A
DS2     0AAA  A   TRDSR1  0B2A  A   TRDSK1  0B36  A   DS1     0AD2  A
DRDSRK  0B3E  A   DSR3    0AF8  A   DSK3    0AFA  A   DSR4    0B4A  A
DSK4    0B4E  A   DSR2    0B1E  A   DSK2    0B22  A   DSR1    0B32  A
DSK1    0B36  A   TRDSS1  0B4M  A   TDSSK1  0BCA  A   DSS1    0B4E  A
TRDSS2  0BAC  A   TDSSK2  0BD2  A   DSS2    0B66  A   TRDSS3  0BDB  A
TDSSK3  0BDA  A   DSS3    0B7E  A   TRDSS4  0BC4  A   TDSSK4  0BE2  A
DSS4    0B92  A   DSSR1   0BA6  A   DSSR2   0BB4  A   DSSR3   0BC0  A
DSSR4   0BC8  A   DSSK1   0BCA  A   DSSK2   0BD2  A   DSSK3   0BDA  A
DSSK4   0BE2  A

     ASS.ERR.    00000
```