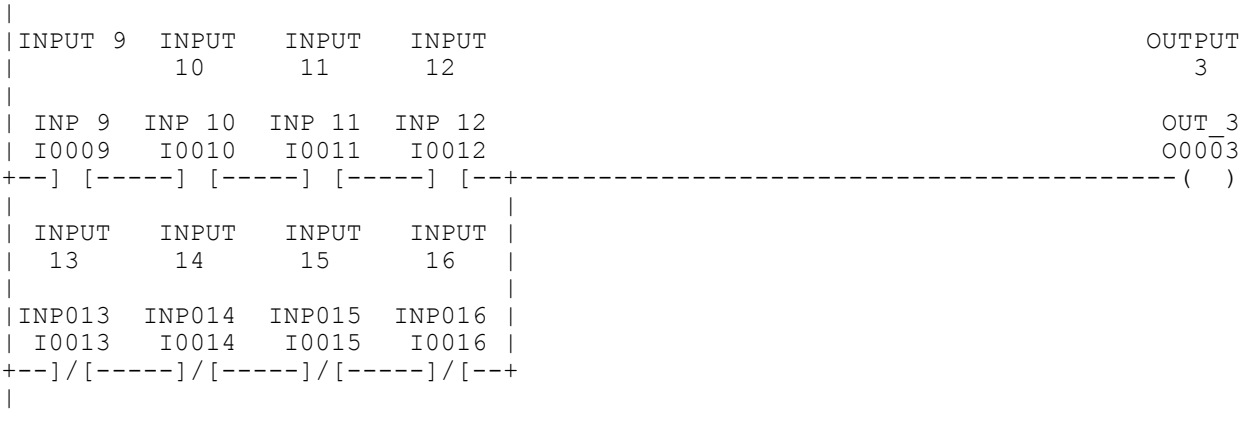
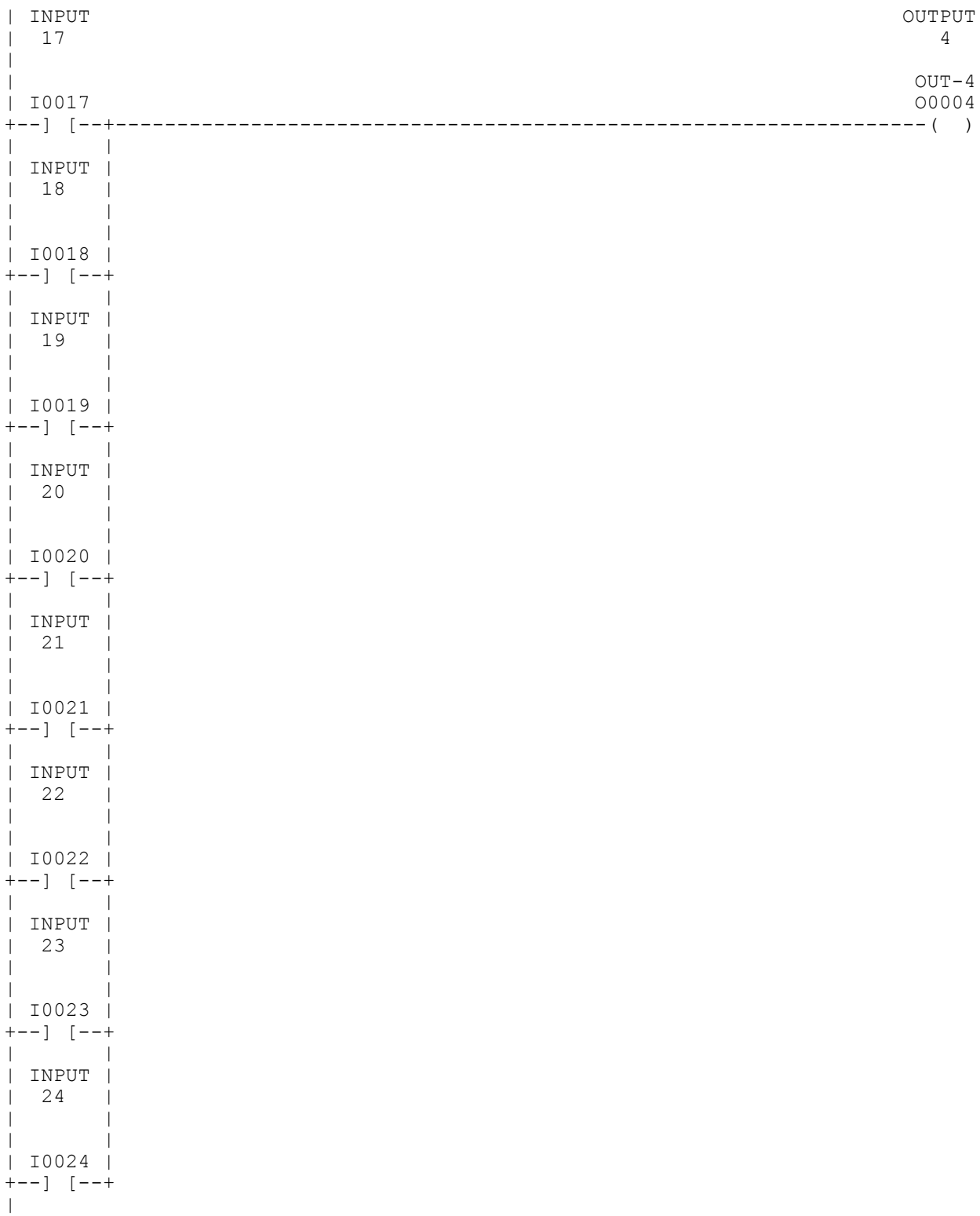
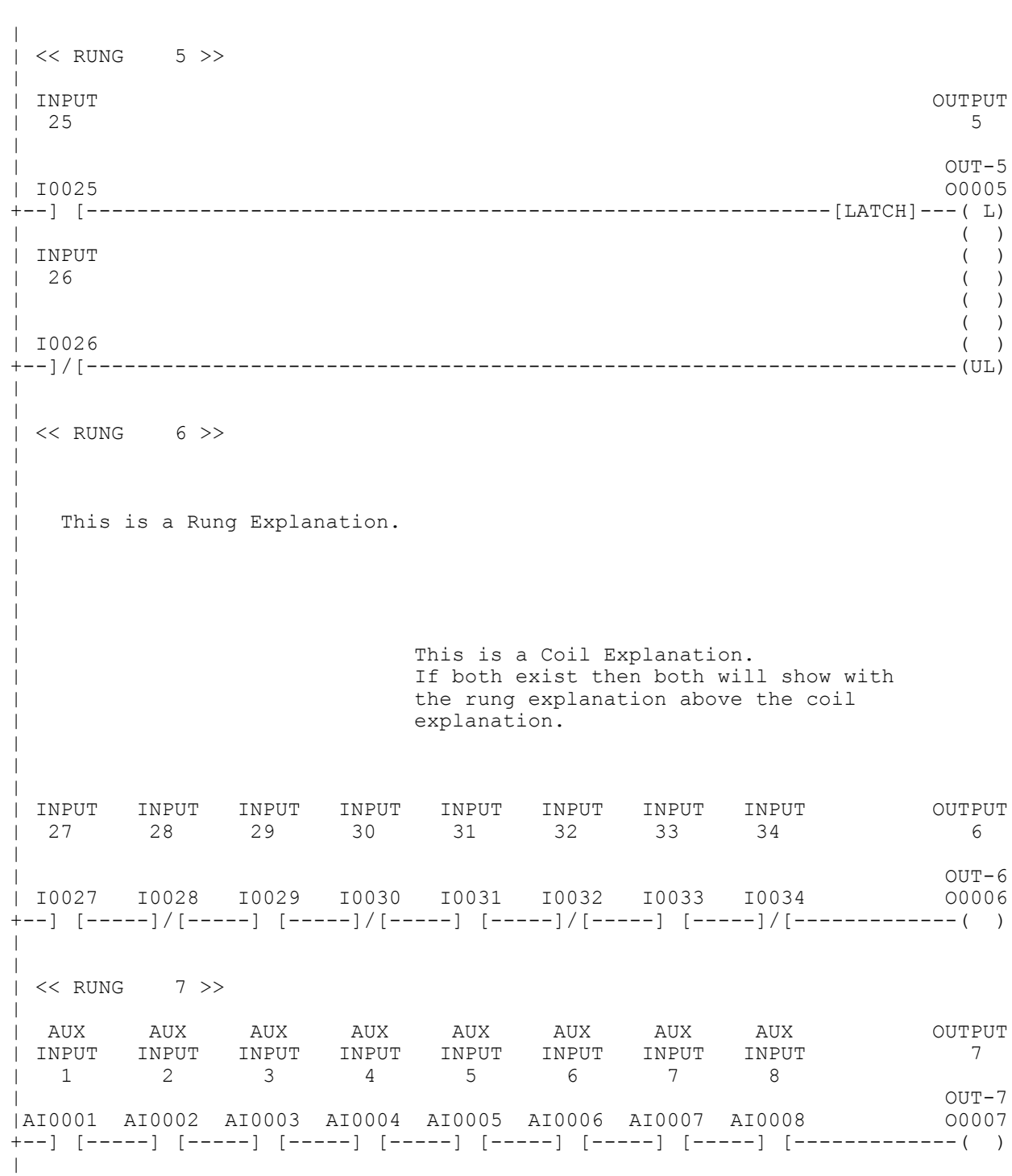
```
|
| << RUNG    0 >>
|
|
|
|
|
|
+[   Start of Program  ]-
|
|
| << RUNG    1 >>
|
|
|
|   This program is totally useless...
|
|   as far as controlling any process.  Now on the other hand if you
|   would like to see how different instructions are converted read on.
|
|   This program was converted using the NONE option.  No I/O addresses
|   were converted.  Each I/O addresses will point to the INT array
|   (I[xxx].yy or O[xxx].yy).
|
|   The TRASH REGISTER option was set to R0299.
|
|   This section of rungs show how RELAYS are converted.
|
|
|
|INPUT 1 INPUT 2 INPUT 3 INPUT 4                                        OUTPUT
|                                                                           1
|
| INP_1   INP_2   INP_3   INP_4                                          OUT_1
| I0001   I0002   I0003   I0004                                          O0001
+--] [-----] [-----] [-----] [-----------------------------------------------(  )
|
|
| << RUNG    2 >>
|
|INPUT 5 INPUT 6 INPUT 7 INPUT 8                                        OUTPUT
|                                                                           2
|
| INP-5   INP-6   INP-7   INP 8                                          OUT_2
| I0005   I0006   I0007   I0008                                          O0002
+--]/[-----]/[-----]/[-----]/[-----------------------------------------------(  )
|
```


RUNG 3
INPUT 9
INPUT 10
INPUT 11
INPUT 12
OUTPUT 3
INP 9
INP 10
INP 11
INP 12
OUT_3
I0009
I0010
I0011
I0012
O0003
INPUT 13
INPUT 14
INPUT 15
INPUT 16
INP013
INP014
INP015
INP016
I0013
I0014
I0015
I0016

```
|
| << RUNG    4 >>
|
```


INPUT
17
OUTPUT
4
OUT-4
O0004
I0017
INPUT
18
I0018
INPUT
19
I0019
INPUT
20
I0020
INPUT
21
I0021
INPUT
22
I0022
INPUT
23
I0023
INPUT
24
I0024

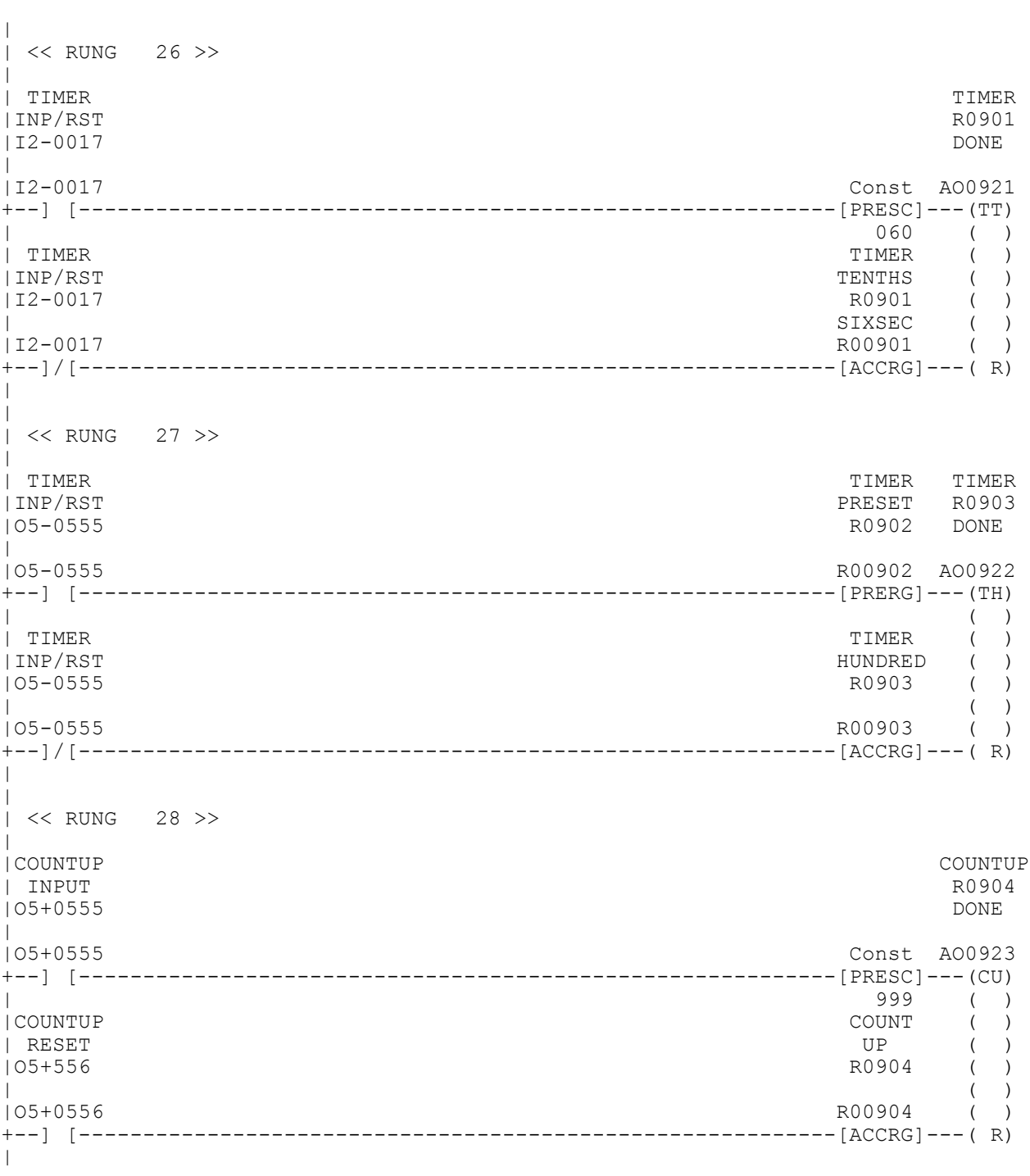

INPUT 25
OUTPUT 5
OUT-5
I0025
O0005
[LATCH]
(L)
INPUT 26
I0026
(UL)
<< RUNG 6 >>
This is a Rung Explanation.
This is a Coil Explanation.
If both exist then both will show with
the rung explanation above the coil
explanation.
INPUT 27
INPUT 28
INPUT 29
INPUT 30
INPUT 31
INPUT 32
INPUT 33
INPUT 34
OUTPUT 6
OUT-6
I0027
I0028
I0029
I0030
I0031
I0032
I0033
I0034
O0006
<< RUNG 7 >>
AUX INPUT 1
AUX INPUT 2
AUX INPUT 3
AUX INPUT 4
AUX INPUT 5
AUX INPUT 6
AUX INPUT 7
AUX INPUT 8
OUTPUT 7
OUT-7
AI0001
AI0002
AI0003
AI0004
AI0005
AI0006
AI0007
AI0008
O0007

```
|
| << RUNG    8 >>
|
|  AUX     AUX     AUX     AUX     AUX     AUX     AUX     AUX           OUTPUT
| INPUT   INPUT   INPUT   INPUT   INPUT   INPUT   INPUT   INPUT            8
|   9      10      11      12      13      14      15      16
|                                                                        OUT-8
|AI0009  AI0010  AI0011  AI0012  AI0013  AI0014  AI0015  AI0016          O0008
+--]/[-----]/[-----]/[-----]/[-----]/[-----]/[-----]/[-----]/[-------------(  )
|
|
| << RUNG    9 >>
|
|  AUX                                                                   OUTPUT
| INPUT                                                                    9
|  65
|
|AI0065                                                                   O0009
+--] [-----------------------------------------------------------------------(  )
|
|
| << RUNG   10 >>
|
|  AUX                                                                   OUTPUT
| INPUT                                                                   10
|  66
|
|AI0066                                                                   O0010
+--] [-----------------------------------------------------------------------(  )
|
|
| << RUNG   11 >>
|
|  AUX                                                                   OUTPUT
| INPUT                                                                   11
|  67
|
|AI0067                                                                   O0011
+--] [-----------------------------------------------------------------------(  )
|
|
| << RUNG   12 >>
|
|  AUX                                                                   OUTPUT
| INPUT                                                                   12
|  68
|
|AI0068                                                                   O0012
+--] [-----------------------------------------------------------------------(  )
|
```

```
|
| << RUNG   13 >>
|
|  AUX                                                                OUTPUT
| INPUT                                                                 13
|  129
|
|AI0129                                                               O0013
+--] [-------------------------------------------------------------------(  )
|
|
| << RUNG   14 >>
|
|  AUX                                                                OUTPUT
| INPUT                                                                 14
|  130
|
|AI0130                                                               O0014
+--] [-------------------------------------------------------------------(  )
|
|
| << RUNG   15 >>
|
|  AUX                                                                OUTPUT
| INPUT                                                                 15
|  512
|
|AI0512                                                               O0015
+--] [-------------------------------------------------------------------(  )
|
|
| << RUNG   16 >>
|
|  AUX                                                                OUTPUT
| INPUT                                                                 16
| 1024
|AUX END
|AI1024                                                               O0016
+--] [-------------------------------------------------------------------(  )
|
|
| << RUNG   17 >>
|
|OUTPUT                                                               OUTPUT
|  17                                                                   17
|
|
| O0017                                                               O0017
+--]/[-------------------------------------------------------------------(  )
|
```

```
|
| << RUNG   18 >>
|
|  AUX                                                                  OUTPUT
|OUTPUT                                                                   33
|   1
|
|AO0001                                                                  O0033
+--]/[-------------------------------------------------------------------(  )
|
|
| << RUNG   19 >>
|
|  AUX                                                                  OUTPUT
|OUTPUT                                                                   37
|   2
|
|AO0002                                                                  O0037
+--]/[-------------------------------------------------------------------(  )
|
|
| << RUNG   20 >>
|
|  AUX                                                                  OUTPUT
|OUTPUT                                                                   38
|   7
|
|AO0007                                                                  O0038
+--]/[-------------------------------------------------------------------(  )
|
|
| << RUNG   21 >>
|
|  AUX                                                                  OUTPUT
|OUTPUT                                                                   40
|  12
|
|AO0012                                                                  O0040
+--]/[-------------------------------------------------------------------(  )
|
|
| << RUNG   22 >>
|
|  AUX                                                                  OUTPUT
|OUTPUT                                                                   48
|  32
|
|AO0032                                                                  O0048
+--]/[-------------------------------------------------------------------(  )
|
```

```
|
| << RUNG   23 >>
|
|  AUX                                                                  ONE
|OUTPUT                                                                 SHOT
|  44                                                                  AO0914
|
|AO0044                                                                AO0914
+--] [-------------------------------------------------------------------(OS)
|
|
| << RUNG   24 >>
|
|  AUX                                                                  LATCH
|OUTPUT                                                                AO0918
|  86
|
|AO0086                                                                AO0918
+--] [-----------------------------------------------------------[LATCH]---( L)
|                                                                        (  )
|  AUX                                                                   (  )
|OUTPUT                                                                  (  )
|  919                                                                   (  )
|                                                                        (  )
|AO0919                                                                  (  )
+--] [-------------------------------------------------------------------(UL)
|
|
| << RUNG   25 >>
|
|
|
|   TIMERS AND COUNTERS
|
|
|
| TIMER                                                                 TIMER
| INPUT                                                                 R0900
|O0-0017                                                                DONE
|
|O0-0017                                                         Const  AO0920
+--] [-----------------------------------------------------------[PRESC]---(TS)
|                                                                  060    (  )
| TIMER                                                           TIMER   (  )
| RESET                                                          SECONDS  (  )
|I0-0017                                                          R0900   (  )
|                                                                         (  )
|I0-0017                                                         R00900   (  )
+--] [-----------------------------------------------------------[ACCRG]---( R)
|
```


RUNG 26
TIMER INP/RST I2-0017
TIMER R0901 DONE
I2-0017
Const PRESC 060
TIMER TENTHS R0901
TIMER SIXSEC R00901
I2-0017
ACCRG
AO0921 TT
RUNG 27
TIMER INP/RST O5-0555
TIMER PRESET R0902
TIMER R0903 DONE
O5-0555
R00902
PRERG
AO0922 TH
TIMER INP/RST O5-0555
TIMER HUNDRED R0903
O5-0555
R00903
ACCRG
RUNG 28
COUNTUP INPUT O5+0555
COUNTUP R0904 DONE
O5+0555
Const PRESC 999
COUNTUP RESET O5+556
COUNT UP R0904
O5+0556
R00904
ACCRG
AO0923 CU

```
|
| << RUNG   29 >>
|
|COUNTDN                                                          COUNT  COUNTDN
| INPUT                                                           DOWN    R0906
|I9+0003                                                         PRESET   DONE
|
|I9+0003                                                         R00905  AO0924
+--] [-------------------------------------------------------------[PRERG]---(CD)
|                                                                          (  )
|COUNTDN                                                          COUNT    (  )
| RESET                                                           DOWN     (  )
|I9+0004                                                          R0906    (  )
|                                                                          (  )
|I9+0004                                                         R00906    (  )
+--] [-------------------------------------------------------------[ACCRG]---( R)
|
|
| << RUNG   30 >>
|
|
|   SHIFT/MOVE FUNCTIONS
|
|
|   THIS SHIFT USES AN ADDRESS THAT IS ON THE WORD BOUNDRY.
|
|
|
| SHIFT
| I0065
|
|
| I0065
+[SHIFT]-                                                                 (  )
|
|
| << RUNG   31 >>
|
|
|
|   THIS SHIFT USES AN ADDRESS THAT IS ON A BYTE BOUNDRY.
|   THE ADDRESS WILL CHANGE TO THE WORD BOUNDRY AND A NOTE
|   WILL BE MADE IN THE LOG FILE.
|
|
|
|
| SHIFT
| I0073
|
|
| I0073
+[SHIFT]-                                                                 (  )
|
```

```
|
| << RUNG   32 >>
|
|I/O REG I/O REG
|SOURCE  RESULT
| I0081   R0907
|
| I0081   R0907
+[ I/O  TO REG ]-                                                ( )
|
|
| << RUNG   33 >>
|
|REG I/O REG I/O
|SOURCE  RESULT
| R0908   O0081
|
| R0908   O0081
+[ REG  TO I/O ]-                                                ( )
|
|
| << RUNG   34 >>
|
|BIN BCD BIN BCD
|SOURCE  RESULT
| R0909   O0097
|
| R0909   O0097
+[ BIN  TO BCD ]-                                                ( )
|
|
| << RUNG   35 >>
|
|BCD BIN BCD BIN
|SOURCE  RESULT
| I0097   R0910
|
| I0097   R0910
+[ BCD  TO BIN ]-                                                ( )
|
*
* << RUNG   36 >>
*
*
*
*   MCR ZONE
*
*
*
*  MCR
* INPUT
* I0035
*
* I0035   Const
+--] [---[ MCR ]-                                                ( )
*           002
```

```
*
* << RUNG   37 >>
*
*
*
*   BASIC ARITHMETIC FUNCTIONS
*
*
*
*  ADD     ADD     ADD
*SOURCE  SOURCE  RESULT
*   A       B       C
*OPER_A  OPER B  OPER-C
* R0911   R0912   R0913
+[  A   +   B   =   C  ]-                                              (  )
*
*
* << RUNG   38 >>
*
* TIMER    ADD     ADD                                                 ADD
*TENTHS  SOURCE  RESULT                                               FAULT
* R0901     B       C                                                AO0925
*SIXSEC
* R0901   R0915   R0916                                              AO0925
+[  A   +   B   =   C  ]-------------------------------------------------(  )
*
*
* << RUNG   39 >>
*
*COMPARE COMPARE   SUB     SUB    TRASH
*SOURCE  SOURCE  SOURCE  SOURCE  REGISTR
*   A       B       A       B
*
* R0917   R0918   R0919   R0920   R0299
+[   A  :    B ]-[  A   -   B   =   C  ]-                               (  )
*
*
* << RUNG   40 >>
*
*
*
*   SUBTRACT WITH A RESULT OF THE "TRASH REGISTER" WILL REMOVE
*   THE SUBTRACT INSTRUCTION AND LEAVE THE COMPARE.
*
*
*
*  SUB     SUB    TRASH                                                A < B
*SOURCE  SOURCE  REGISTR
*   A       B
*
* R0921   R0922   R0299                                              AO0926
+[  A   -   B   =   C  ]-------------------------------------------------(  )
*
```

```
*
* << RUNG   41 >>
*
*
*
*   SKIP ZONE
*
*
*
*
*
*
*
*         Const
+[NO OP]-[SKIP ]-                                                    (  )
*           002
*
* << RUNG   42 >>
*
*
*
*   SPECIAL FUNCTIONS
*
*
*
*SERIAL                                                               SCREQ
* COMM                                                                COMM
*REQUEST                                                            FAILURE
*
* R0890                                                            AO0927
+[SCREQ]-------------------------------------------------------------(  )
*
*
* << RUNG   43 >>
*
* DATA                                                                DPREQ
*PROCESS                                                              COMM
*REQUEST                                                            FAILURE
*
* R0880                                                            AO0928
+[DPREQ]-------------------------------------------------------------(  )
*
```

```
|
| << RUNG   44 >>
|
|
|
|   DATA MOVE FUNCTIONS
|
|
|
|                 MOVE
|                RESULT
|                   B
|
|  Const          R00923
+[   A     MOVE      B ]-                                              (  )
| +00123
|
| << RUNG   45 >>
|
| MOVE            MOVE
|  L/R            LEFT
|SOURCE          RESULT
|
|  I0129          R00924
+[  MOVE LEFT  8 BITS  ]-                                              (  )
|
|
| << RUNG   46 >>
|
| MOVE            MOVE
|  L/R            RIGHT
|SOURCE          RESULT
|
|  I0129          R00925
+[  MOVE RIGHT 8 BITS  ]-                                              (  )
|
|
| << RUNG   47 >>
|
| DATA
|PROCESS
|REQUEST
|
| R00880
+[                        BLOCK MOVE                              ]-    (  )
|         +01257  +00002  +00000  +00321  +00211  +00000  +00000
```

```
|
| << RUNG   48 >>
|
|
|
|   SIGNED ARITHMETIC FUNCTIONS
|
|
|
| ADDX    ADDX    ADDX
|SOURCE  SOURCE  RESULT
|   A       B       C
|
| R00926  R00927  R00928
+[   A ADDX  B  =    C ]-                                              (  )
|
|
| << RUNG   49 >>
|
| SUBX            TRASH
|SOURCE          REGISTR
|   A
|
| R00929   Const  R00299
+[   A SUBX  B  =    C ]-                                              (  )
|         +00002
|
| << RUNG   50 >>
|
|         SUBX    TRASH                                               A < B
|        SOURCE  REGISTR
|           B
|
|  Const  R00930  R00299                                            AO0929
+[   A SUBX  B  =    C ]-------------------------------------------------(  )
| +00050
|
| << RUNG   51 >>
|
|  MPY     MPY     MPY
|SOURCE  SOURCE  RESULT
|   A       B       C
|
| R00931  R00932  R00933
+[   A MPY   B  =    C         ]-                                        (  )
|
|
| << RUNG   52 >>
|
|  DVD                    DVD     DVD
|SOURCE                 RESULT  RESULT
|   A                     QUO     REM
|
| R00934           Const  R00935  R00936
+[   A    DVD        B     QUO     REM ]-                               (  )
|                 +00002
```

```
|
| << RUNG   53 >>
|
|  DVD                    DVD    TRASH
|SOURCE                  SOURCE  REGISTR
|   A                      QUO
|
| R00937          Const  R00938  R00299
+[   A    DVD       B     QUO     REM ]-                                  (  )
|                +00003
|
| << RUNG   54 >>
|
|
|
|   DOUBLE PRECISION ARITHMETIC
|
|
|
| DPADD                           DPADD
|SOURCE                          RESULT
|   A                               C
|
| R00939           Const          R00941
+[   A    DPADD      B       =       C         ]-                          (  )
|                +0000075000
|
| << RUNG   55 >>
|
| DPSUB           DPSUB           DPSUB
|SOURCE          SOURCE          RESULT
|   A               B               C
|
| R00943          R00945          R00947
+[   A    DPSUB      B       =       C         ]-                          (  )
|
|
| << RUNG   56 >>
|
| DPSUB                                                                  DP
|RESULT                                                                GREATER
|   C                                                                  RESULT
|
| R00947           Const                                               AO0930
+[   A GREATER THAN   B        ]-----------------------------------------(  )
|                +0000032768
```

```
|
| << RUNG   57 >>
|
|
|
|   FLOATING POINT ARITHMETIC
|
|
|
| FADD                          FADD
|SOURCE                        RESULT
|   A                             C
|
| R00949          Const         R00951
+[  A    FADD       B       =       C          ]-                          (  )
|                +5.000000-01
|
| << RUNG   58 >>
|
| FSUB           FSUB           FSUB
|SOURCE         SOURCE         RESULT
|   A              B              C
|
| R00953         R00955         R00957
+[  A    FSUB       B       =       C          ]-                          (  )
|
|
| << RUNG   59 >>
|
| FMULT                         FMULT
|SOURCE                        RESULT
|   A                             C
|
| R00960          Const         R00962
+[  A    FMULT      B       =       C          ]-                          (  )
|                +1.000000+01
|
| << RUNG   60 >>
|
| FDIV           FDIV           FDIV
|SOURCE         SOURCE         RESULT
|   A              B              C
|
| R00964         R00966         R00968
+[  A    FDIV       B       =       C          ]-                          (  )
|
|
| << RUNG   61 >>
|
| FDIV                                                                  FP
|RESULT                                                              GREATER
|   C                                                                RESULT
|
| R00968                  Const                                      AO0931
+[  A    FP GREATER THAN   B          ]-----------------------------------(  )
|                        +0.000000+00
```

```
|
| << RUNG   62 >>
|
|INT FP          INT FP
|SOURCE          RESULT
|I4+0001          R0970
|
|I4+0001          R00970
+[ INTEGER  TO  FLOATING POINT ]-                                    (  )
|
|
| << RUNG   63 >>
|
|INT FP          FP INT
|RESULT          RESULT
| R0970          I4+0033
|
| R00970         I4+0033
+[ FLOATING POINT  TO  INTEGER ]-                                    (  )
|
|
| << RUNG   64 >>
|
| EQUAL   EQUAL                                                      A = B
|SOURCE  SOURCE                                                     RESULT
|   A       B
|
| R06230  R06231                                                    AO0932
+[  A EQUAL B  ]-----------------------------------------------------(  )
|
|
| << RUNG   65 >>
|
|
|
|   TABLE MOVE FUNCTIONS
|
|
|
|SRC TAB SRC TAB
|SOURCE  POINTER
| R1000
|
| R01000  R01001  Const
+[ SRC-TO-TABLE     LEN]-                                            (  )
|                   001
|
| << RUNG   66 >>
|
|SRC TAB SRC TAB
|SOURCE  POINTER
|
|
| R01003  R01004  Const
+[ SRC-TO-TABLE     LEN]-                                            (  )
|                   005
```

```
|
| << RUNG   67 >>
|
|TAB DST TAB DST
| TABLE  POINTER
|
|
| R01010  R01012  Const
+[TABLE-TO-DEST    LEN]-                                          (  )
|                  001
|
| << RUNG   68 >>
|
|TAB DST TAB DST
| TABLE  POINTER
|
|
| R01013  R01019  Const
+[TABLE-TO-DEST    LEN]-                                          (  )
|                  005
|
| << RUNG   69 >>
|
|TAB TAB         TAB TAB
| TABLE           TABLE
|   A               B
|
| R01020          R01025  Const
+[TBL A    MOVE   TBL B     LEN]-                                 (  )
|                           001
|
| << RUNG   70 >>
|
|TAB TAB         TAB TAB
| TABLE           TABLE
|   A               B
|
| R01020          R01025  Const
+[TBL A    MOVE   TBL B     LEN]-                                 (  )
|                           005
|
| << RUNG   71 >>
|
|TAB EXT         TAB EXT
| TABLE           TABLE
|   A               B
|
| R06355          R07634   Const
+[   A  MOVE TBL EXT  B    LEN ]-                                 (  )
|                             064
```

```
|
| << RUNG   72 >>
|
|
|
|   THIS RUNG IS SHOWING THE INDIRECT ADDRESS FUNCTION.
|
|
|
|
|
|
|IR00301         IR00302   Const
+[   A  MOVE TBL EXT  B    LEN ]-                                    (  )
|                            001
|
| << RUNG   73 >>
|
|
|   LIST FUNCTIONS
|
|
|
|ADD TOP         ADD TOP
|SOURCE          POINTER
|
|
|  I0113          R01030  Const
+[ SRC ADD-TO-TOP  LIST     LEN]-                                    (  )
|                           010
|
| << RUNG   74 >>
|
|ADD TOP         REM BOT
|POINTER          DEST
|
|
| R01030           O0113  Const
+[ LIST REM-FM-BOT DEST     LEN]-                                    (  )
|                           010
|
| << RUNG   75 >>
|
|LIST TO          LIST
| SORT            ORDER
|
|
| R01040          R01050  Const
+[LIST A   SORT  LIST B     LEN]-                                    (  )
|                           010
```

```
|
| << RUNG   76 >>
|
|
|
|   MATRIX FUNCTIONS
|
|
|
|  AND     AND     AND
|SOURCE  SOURCE  RESULT
|   A       B       C
|
| R01060  R01061  R01062  Const
+[  A  AND  B   =   C      LEN ]-                                          (  )
|                           001
|
| << RUNG   77 >>
|
|  AND     AND     AND
|SOURCE  SOURCE  RESULT
|   A       B       C
|
| R01063  R01066  R01069  Const
+[  A  AND  B   =   C      LEN ]-                                          (  )
|                           003
|
| << RUNG   78 >>
|
|  IOR     IOR     IOR
|SOURCE  SOURCE  RESULT
|   A       B       C
|
| R01072  R01073  R01074  Const
+[  A  IOR  B   =   C      LEN ]-                                          (  )
|                           001
|
| << RUNG   79 >>
|
|  IOR     IOR     IOR
|SOURCE  SOURCE  RESULT
|   A       B       C
|
| R01075  R01078  R01081  Const
+[  A  IOR  B   =   C      LEN ]-                                          (  )
|                           003
|
| << RUNG   80 >>
|
|  EOR     EOR     EOR
|SOURCE  SOURCE  RESULT
|   A       B       C
|
| R01084  R01085  R01086  Const
+[  A  EOR  B   =   C      LEN ]-                                          (  )
|                           001
```

```
|
| << RUNG   81 >>
|
|  EOR     EOR     EOR
|SOURCE  SOURCE  RESULT
|   A       B       C
|
| R01087  R01090  R01093  Const
+[  A  EOR  B   =   C      LEN ]-                                          (  )
|                           003
|
| << RUNG   82 >>
|
|  INV     INV
|SOURCE  RESULT
|   A       B
|
| R01096  R01097  Const
+[   A INV   B     LEN ]-                                                 (  )
|                   001
|
| << RUNG   83 >>
|
|  INV     INV
|SOURCE  RESULT
|   A       B
|
| R01098  R01101  Const
+[   A INV   B     LEN ]-                                                 (  )
|                   003
|
| << RUNG   84 >>
|
|        COMPARE COMPARE COMPARE COMPARE
|         INPUT   REFER   MASK    FAULT
|                                BIT NUM
|
|          I0137  R01110  R01111  R01112  Const
+[COMPARE INPUT     REF    MASK   FAULT     LEN]-                          (  )
|                                           001
|
| << RUNG   85 >>
|
| INPUT                   BIT                                              AUX
|  133                  SET/RST                                          OUTPUT
|                       MATRIX                                             933
|
| I0133    Const          R01130  Const                                  AO0933
+--] [---[ BIT   SET    MATRIX      LEN]-----------------------------------(  )
|          00001                     001
```

```
|
| << RUNG   86 >>
|
| INPUT                   BIT                                          AUX
|  134                   SET/RST                                      OUTPUT
|                        MATRIX                                         934
|
| I0134    Const          R01130  Const                              AO0934
+--] [---[ BIT  CLEAR   MATRIX      LEN]-----------------------------------(  )
|          00001                    001
|
| << RUNG   87 >>
|
|                SHIFT L
|                MATRIX
|
|
|          Const  R01131  Const
+[ SHIFT LEFT N   MATRIX   LEN ]-                                          (  )
|          00001            002
|
| << RUNG   88 >>
|
|                SHIFT R
|                MATRIX
|
|
|          Const  R01133  Const
+[ SHIFT RT   N   MATRIX   LEN ]-                                          (  )
|          00001            002
|
| << RUNG   89 >>
|
|
|
|   CONTROL FUNCTIONS
|
|
|
|                DO SUB
|                 COUNT
|
|
|         Const   R01150
+[ DO SUB    N    REPS ]-                                                 (  )
|           001
|
| << RUNG   90 >>
|
|
|
|
|
|
+[ENDSW]-
|
```

```
|
| << RUNG   91 >>
|
|
|
|   SUBROUTINE 1
|   CONTROL FUNCTIONS
|
|
|                                                                    AUX
|                                                                  OUTPUT
|                                                                    935
|
|                                                                  AO0935
+[NO OP]-------------------------------------------------------------(  )
|
|
| << RUNG   92 >>
|
|
|
|
|
|
+[ SUSPEND I/O ]-                                                    (  )
|
|
| << RUNG   93 >>
|
|
|
|
|          Const   Const
+[ DO I/O START    END ]-                                            (  )
|          00001   00997
|
| << RUNG   94 >>
|
|STATUS
|
|
|
| R01151
+[    STATUS   ]-                                                    (  )
|
```

```
|
| << RUNG   95 >>
|
|
|
|
|
|               Const
+[ 90-70  I/O   RACK   SERVICE ]-                                        (  )
|               001
|
| << RUNG   96 >>
|
|                          COMM
|                          BLOCK
|
|
|          Const         R01160
+[WINDOW ADDRESS     COMM BLOCK]-                                       (  )
|            0259
|
| << RUNG   97 >>
|
|
|
|
|
|
+[   Return    ]-
|
|
| << RUNG   98 >>
|
|
|
|
|
|
+[ENDSW]-
|
|
| << RUNG   99 >>
|
|
|
|
|
|
+[ENDSW]-
|
```

***** UNREFERENCED RUNG EXPLANATION LIST *****

NO UNREFERENCED RUNG NUMBERS FOUND