STICHTING
MATHEMATISCH CENTRUM
2e BOERHAAVESTRAAT 49
AMSTERDAM
REKENAFDELING

MR 61

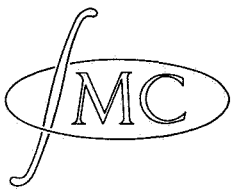
Text of the second ALGOL 60 translator for the X1

by

P.J.J. van de Laarschot

and

J. Nederkoorn



May 1963

Printed at the Mathematical Centre at Amsterdam,49,2nd Boerhaavestraat.
The Netherlands.

The Mathematical Centre, founded the 11th of February 1946, is a non - profit institution aiming at the promotion of pure mathematics and its applications, and is sponsored by the Netherlands Government through the Netherlands Organization for Pure Scientific Research (Z.W.O.) and the Central National Council for Applied Scientific Research in the Netherlands (T.N.O.), by the Municipality of Amsterdam and by several industries.

Table of contents

1. Preface

This report contains the text of the second ALGOL 60 translator program, written at the Mathematical Centre.
In [1] we gave a provisional user's description of the system. Complete descriptions, explanations etc. are being prepared. The text of the loader and of the so-called ALP-complex (a set of arithmetical and administrative routines, an interpreter using variable word length and variable precision) is not yet included here.

[1] MR 54, Voorlopige gebruiksaanwijzing van de tweede MC ALGOL-60 vertaler, by P.J.J. van de Laarschot and J. Nederkoorn, a report of the Mathematical Centre, Amsterdam, Jan. 1963.

[2] Revised report on the Algorithmic language ALGOL 60, edited by Peter Naur, Regnecentralen, Copenhagen, 1962.

Paragraph definitions for the X1 of the Mathematical Centre

```
DP EZ  0   X 6          begin translator
DP EE 4464 E Z 0        OBC6(output buffer class 6)
DP EF 80 E E 0          begin prestack list and TLI
DP EH 1024 E F 0        begin address pile
DP EK 320 E H 0         variable part of name list
DP EL 42   X 0          length fixed address pile
DP ER 151  X 0          length fixed part name list
DP ES  9   X 3          (27D16)
DP ET 8192 X 0          size of living store
```

```
DP ZE  0 E Z 0      working space translator
DP ZF 16 Z E 2      RFS(read flexowr.symbol)
DP ZH 31 Z F 3      NAS(next ALGOL symbol)
DP ZK 29 Z H 5      NRS(next relevant symbol)
DP ZL  9 Z K 2      RUND(rd until next delim.)
DP ZR  0 Z L 6      SPS(start prescan)
DP ZS 17 Z R 1      FPL(fill prescan list)
DP ZT  4 Z S 1      APL(augment prescan list)
DP ZW 10 Z T 0      FTL(fill translator list)
DP ZU  7 Z W 0      PSP(prescan program)
DP ZY 14 Z U16      Basic cycle
DP ZN 23 Z Y 1      begin
DP FZ 14 Z N 0      constants
DP FE 17 F Z 1      LFN(look for name)
DP FF 28 F E 1      FNL(fill namelist)
DP FH 10 F F 0      + -
DP FK 27 F H 0      relational operators
DP FL 12 F K 0      multipl.op. and to the power
DP FR 15 F L 0      logical "not"
DP FS 12 F R 0      other log.operators
DP FT 17 F S 0      ? or stopcode
DP FW  3 F T 1      go to
DP FU  7 F W 0      if
DP FY 21 F U 0      then
DP FN 29 F Y 0      else
DP HZ  8 F N 1      for
DP HE  6 H Z 0      do
DP HF 22 H E 1      distribution of ","
DP HH 29 H F 0      :
DP HK 24 H H 0      distribution of ";"
DP HL 31 H K 0      distribution of ":="
DP HR 31 H L 1      step
DP HS 15 H R 0      until
DP HT 10 H S 0      while
DP HW 18 H T 0      distribution of "("
DP HU 29 H W 0      distribution of ")"
DP HY 30 H U 0      opening square bracket
DP HN 25 H Y 0      closing square bracket
DP KZ 29 H N 1      opening string quote
DP KE 23 K Z 0      end
DP KF 15 K E 0      own
DP KH  6 K F 0      Boolean
DP KK 23 K H 0      integer
DP KL  3 K K 0      real
DP KR  3 K L 0      array
DP KS 15 K R 0      switch
```

```
DP KT 11 K S 0      procedure
DP KW  6 K T 0      string
DP KU  3 K W 0      label
DP KY  9 K U 0      value
DP KN  3 K Y 0      require
DP LZ  4 K N 0      put
DP LE  5 L Z 0      PCP(prepare compl.parameter)
DP LF 23 L E 0      NNP(name and number process.)
DP LH 26 L F 0      name processor
DP LK 11 L H 1      number processor
DP LL 19 L K 0      backward scanner I
DP LR 22 L L 0      "  "         "      II
DP LS  0 L R 2         "         "      III
DP LT  7 L S 0         "         "      IV
DP LW  4 L T 1      implicit jumps
DP LU  6 L W 0      procedure identifiers
DP LY 23 L U 1      array identifiers
DP RE  0 L Y 1      labels and switches
DP RF 24 R E 0      conditional expressions
DP RH 28 R F 0      for statements
DP RK 16 R H 0      conditional statements
DP RL  8 R K 0      end of statement
DP RR 17 R L 0      IPW(insert parameter word)
DP RS 16 R R 0      "(" of formal parameter part
DP RT 12 R S 0      "(" in expressions
DP RW 11 R T 0      backward scanner V
DP RU 17 R W 0      punch tape feed
DP RY  6 R U 0      autostarts
DP RN 23 R Y 0      type stop and line number
DP SZ 19 R N 0      start translation
DP SE 20 S Z 1      INB(introduction new block) I
DP SF 26 S E 1      simple block introduction
DP SH 30 S F 0      INB II, transfer from prescanlist
DP SK 29 S H 1      INB III, produce SCC
DP SL 27 S K 1      INB IV
DP SR  2 S L 1      TR(transport)
DP SS 12 S R 0      switches and labels in INB
DP ST 21 S S 0      procedures in INB
DP SW 12 S T 0      extension input program (DU)
DP SU 13 S W 0      ";" after formal par. part
DP SY 10 S U 0      ";" after specification
DP SN  3 S Y 1      ";" after value part
DP TZ 10 S N 0      ";" after switch declaration
DP TE 21 T Z 0      ";" after array declaration
```

```
DP TF  6 T E 0      ";"after variable decl.
DP TH  7 T F 0      "," in value list
DP TK  8 T H 0      "," formal par. separator
DP TL 17 T K 0      "," specification separator
DP TR 27 T L 4      "," array segment separ.
DP TS 18 T R 0      "," switch list separator
DP TT  0 T S 1      "," type list separator
DP TW  4 T T 1      FNC(fill name code) I
DP TU 19 T W 0      "," actual par. separator
DP TY  9 T U 0      "," for list separator
DP TN 22 T Y 0      "," index separator
DP WZ  9 T N 0      "," bound pair separator
DP WE  7 W Z 0      ")" after formal parameter
DP WF 16 W E 0      ")" after actual parameter
DP WH 24 W F 0      := in switch declaration
DP WK 30 W H 0      := in for statement
DP WL 27 W K 0      PSE(prepare switch element)
DP WR 17 W L 0      SAP(simple actual parameter)
DP WS 18 W R 2      TAP(translate act.parameter)
DP WT 11 W S 1      TPW(tail of parameter word)
DP WW  8 W T 0      FNC II(fill name code arrays)
DP UZ 22 W W 0      working space output
DP UE 16 U Z 0      FOT(fill object tape)
DP UF  0 U E 1      constants of output program
DP UH 22 U F 0      AIB(add imparity bit)
DP UK 25 U H 1      AOB(administration output )
DP UL  0 E E 0      OBC6(output buffer class 6)
DP UR 23 U K 1      punch end of object tape
DP YZ 31 U R 0      constants
DP YE 13 Y Z 0      constants
DP RZ  0 Y E 6      fixed contents address pile
DP LN  0 R Z 2      fixed contents name list
```

extension input program; read directive DU

```
   DA  24   X 2
   DI
24   2T   0 S W 0 A
   DA   0   W 0
 0   2S   2   W 0 A
 1   2T   8   D 9 A
 2   6T  21   D 514
 3   2S   0       A
 4   6S  31   X 0
 5   2Y   1   X16
 6 U 1A   2       A P
 7   6A  24   X 0
 8 Y 2T  27   D12 A
 9   2S  31   X 0
10   0P   1   SS
11   0S  24   X 0
12   2T   4   W 0 A
   DC
   DO
```

RFS (read flexowriter symbol)

```
   DA    0 Z F 0
   DI
 0    2B    0    E 0    Z        initial state ?
 1    2B    1         A
 2 Y  6B    0    E 0
 3    2Z    1    X16
 4    6S   24    X 0    Z
 5 Y  6S    1    E 0
 6 Y  2T    3    F 0 A
 7 U  OLS 124         A Z
 8 Y  2T    5    F 0 A
 9 U  OLS 122         A Z
10 Y  2T    5    F 0 A
11 U  OLS 127         A Z
12 Y  2T    3    F 0 A
13 U  OLS  62         A Z
14 N  1S   16         A Z
15 Y  2T   21    F 0 A
16    1S   10         A Z
17 Y  4B    2    E 0              LC
18 Y  2T   21    F 0 A
19    2T   26    F 3 A
20 Y  7Y 1003    C 0
21    2A    1    E 0              case definition
22    2S   24    X 0
23    OLA 124         A Z
24 Y  OS  129         A
25    2A    0         A
26    OD    3         A
27    4P         SB
28    2S    4    F 1 B
29    2P    3    AA
30    4P         AB
31    3P    0    SS  B
   DC
   DO
```

```
   DA    0 Z F 1
   DI
 0    2LS 255        A
 1 U  0LS 63         A Z
 2 Y  7Y  1002  C 0
 3    2T    8    X 0    E
   DN
 4 +131391
 5 +4129855
 6 +526143
 7 +8011583
 8 +10698559
 9 +4152639
10 +4129599
11 +4130309
12 +7801151
13 +4144959
14 +16191
15 +1916735
16 +2105151
17 +2309951
18 +4144959
19 +4144959
20 +1854015
21 +4136511
22 +2236735
23 +5848895
24 +7749439
25 +4145471
26 +4134207
27 +4134935
28 +4135743
29 +4144959
30 +4144959
31 +4133907
   DC
   DO
```

```
   DA    0 Z F 2
   DN
 0 +4144918
 1 +4135449
 2 +4151103
 3 +4144959
 4 +723519
 5 +4132159
 6 +1118271
 7 +5783359
 8 +4144959
 9 +4145215
10 +4131903
11 +4132622
12 +4133439
13 +4144959
14 +4144959
15 +4345407
16 +4147263
17 +6448447
18 +8011583
19 +10633023
20 +4152639
21 +4145983
22 +4154459
23 +7824191
24 +4144959
25 +5062463
26 +3686207
27 +3881535
28 +4079423
29 +4144959
30 +4144959
31 +3622463
   DC
   DO
```

```
   DA    0 Z F 3
   DN
 0 +4143423
 1 +4013119
 2 +7880511
 3 +7749439
 4 +4148287
 5 +4141119
 6 +4141874
 7 +4142655
 8 +4144959
 9 +4144959
10 +4140846
11 +4144945
12 +4142388
13 +4160063
14 +4144959
15 +2499903
16 +4139071
17 +2894655
18 +5914431
19 +4144959
20 +4159807
21 +4138815
22 +4139561
23 +4140351
24 +4144959
25 +4144959
   DI
26    2S  24    X 0
27 U 1S  11        A Z
28 Y 2T  21 Z F 0 A
29    2A   1 Z E 0   Z
30    2T  20 Z F 0 A
   DC
   DO
```

NAS(next ALGOL symbol into S)

```
   DA    0 Z H 0
   DI
 0    2B    0    E 0    Z     initial state ?
 1 N  2T    5    H 0 A
 2    2A    0      A
 3    6A    3    E 0         QC
 4    6A    4    E 0         stock
 5    3S    4    E 0    P
 6 Y  6S    4    E 0
 7 Y  2T    9    H 0 A
 8    6T    0    F 0 0       RFS
 9 U  0LS  90      A Z
10 Y  2T   23    H 0 A
11 U  0LS 162      A Z
12 Y  2T   29    H 0 A
13 U  0LS 163      A Z
14 Y  2T   20    H 1 A
15    2A    3    E 0    Z
16 N  2T   10    X 0    E
17    4P         SA
18 U  0LA  93      A Z
19 N  1A  118      A Z
20 N  1A    1      A Z
21 Y  2T    5    H 0 A
22    2T   10    X 0    E
23    6T    0    F 0 0       RFS
24 U  0LS  72      A Z
25 N  7S    4    E 0
26 N  2S   90      A
27 Y  2S   92      A
28    2T   10    X 0    E
29    6T    0    F 0 0       RFS
30 U  0LS  77      A Z
31 Y  2S   69      A
   DC
   DO
```

```
   DA    0 Z H 1
   DI
 0 Y 2T   10    X 0    E
 1 U OLS  72        A  Z
 2 Y 2S   75        A
 3 Y 2T   10    X 0    E
 4 U OLS  74        A  Z
 5 Y 2S   102       A
 6   2A    1        A
 7 Y 4A    3    E 0
 8 Y 2T   10    X 0    E
 9 U OLS  70        A  Z
10 Y 2S   103       A
11 Y 5A    3    E 0
12 Y 2T   10    X 0    E
13 U OLS  162       A  Z
14 Y 2T   29    H 0 A
15   7Y   1004  C 0
16   6T    0    F 0 0        RFS
17 U OLS  163       A  Z
18 Y 2T   16    H 1 A
19   2T    9    X 0    E
20   6T   16    H 1 1
21 U 1S   70        A  Z
22 Y 2S   71        A
23 Y 2T   10    X 0    E
24 U 1S   76        A  E
25 Y 2T   30    H 1 A
26 U OLS  72        A  Z
27 Y 2S   80        A
28 N OLS   3        A
29   2T   10    X 0    E
30 U OLS  90        A  Z
31 Y 2S   68        A
   DC
   DO
```

```
   DA    0 Z H 2
   DI
 0 Y 2T  10    X 0    E
 1 U OLS 10       A Z
 2 Y 2S  110      A
 3 Y 2B   4       A
 4 Y 2T  13    H 5 A
 5 U OLS 12       A Z
 6 Y 2S  97       A
 7 Y 2B   6       A
 8 Y 2T  13    H 5 A
 9 U OLS 13       A Z
10 Y 2S  86       A
11 Y 2B   1       A
12 Y 2T  13    H 5 A
13 U OLS 16       A Z
14 Y 2S  81       A
15 Y 2B   3       A
16 Y 2T  13    H 5 A
17 U OLS 21       A Z
18 Y 2S  114      A
19 Y 2B   4       A
20 Y 2T  13    H 5 A
21 U OLS 24       A Z
22 Y 2S  106      A
23 Y 2B   2       A
24 Y 2T  13    H 5 A
25 U OLS 25       A Z
26 Y 2S  112      A
27 Y 2B   8       A
28 Y 2T  13    H 5 A
29 U OLS 27       A Z
30 Y 2S  109      A
31 Y 2B   3       A
   DC
   DO
```

```
   DA    0 Z H 3
   DI
 0 Y 2T   13    H 5 A
 1 U OLS  30        A Z
 2 Y 2S   95        A
 3 Y 2B    4        A
 4 Y 2T   13    H 5 A
 5 U OLS  31        A Z
 6 Y 2S   115       A
 7 Y 2B    4        A
 8 Y 2T   13    H 5 A
 9 U OLS  32        A Z
10 Y 2S   96        A
11 Y 2B    4        A
12 Y 2T   13    H 5 A
13 U OLS  38        A Z
14 Y 2S   107       A
15 Y 2B    6        A
16 Y 2T   13    H 5 A
17    6S    4    E 0
18    6T    0    F 0 0          RFS
19    OLS 163        A Z
20 N 7Y   1005  C 0
21    6T   16    H 1 1
22 U OLS  10        A Z
23 Y 2S   117       A
24 Y 2B    3        A
25 Y 2T   13    H 5 A
26 U OLS  14        A Z
27 Y 2S   104       A
28 Y 2B    3        A
29 Y 2T   13    H 5 A
30 U OLS  15        A Z
31 Y 2S   82        A
   DC
   DO
```

```
   DA    0 Z H 4
   DI
 0 Y 2B    0          A
 1 Y 2T   13    H 5   A
 2 U OLS  21          A Z
 3 Y 2S   84          A
 4 Y 2B    2          A
 5 Y 2T   13    H 5   A
 6 U OLS  32          A Z
 7 Y 2S   111         A
 8 Y 2B    4          A
 9 Y 2T   13    H 5   A
10 U OLS  17          A Z
11 Y 2S   83          A
12 Y 2B    2          A
13 Y 2T   13    H 5   A
14 U OLS  27          A Z
15 Y 2S   116         A
16 Y 2B    2          A
17 Y 2T   13    H 5   A
18 U OLS  29          A Z
19 Y 2T    4    H 5   A
20   2S    4    E 0
21 U OLS  11          A Z
22 Y 2S   107         A
23 Y 2B    5          A
24 Y 2T   13    H 5   A
25 U OLS  15          A Z
26 Y 2S   85          A
27 Y 2B    1          A
28 Y 2T   13    H 5   A
29 U OLS  14          A Z
30 Y 2S   105         A
31 Y 2B    1          A
   DC
   DO
```

```
   DA    0 Z H 5
   DI
 0 Y 2T   13    H 5 A
 1   2S   108       A
 2   2B    5        A
 3   2T   13    H 5 A
 4   6T    0    F 0 0          RFS
 5 U OLS  163       A Z
 6 N 7Y   1005  C 0
 7   6T   16    H 1 1
 8   OLS  14        A Z
 9 Y 2S   94        A
10 Y 2B    1        A
11 N 2S   113       A
12 N 2B    3        A
13   6B    5    E 0   Z
14   6S    4    E 0
15 Y 2T   23    H 5 A
16   6T    0    F 0 0          RFS
17 U OLS  163       A Z
18 N 7Y   1006  C 0
19   6T   16    H 1 1
20   2B    1        A
21   5B    5    E 0   Z
22 N 2T   16    H 5 A
23   2A    4    E 0
24 U 1A   81        A Z
25 Y 1S   29        A Z
26   4P         AS
27 Y 2T   13    H 5 A
28   2T   10    X 0   E
   DC
   DO
```

```
   DA    0 Z K 0
   DI
 0    2B    0    E 0    Z   initial state ?
 1 N 2T    6    K 0 A
 2    6B    6    E 0        CFLA
 3    6B    7    E 0        stock
 4    6T    0    H 0 2      NAS
 5    2T    9    K 0 A
 6    3A    7    E 0    P
 7 Y 3S    7    E 0
 8 N 6T    0    H 0 2      NAS
 9    6S    8    E 0        last relevant symbol read
10    2A    0         A
11    6A    7    E 0
12    2A    3    E 0    Z
13 N 2T    6    K 2 A
14 U 1S  120         A Z
15 N 1S  121         A Z
16 Y 7Y 1007 C 0
17    2S    8    E 0
18 U 1S   97         A Z
19 N 2T   29    K 0 A
20    2A    6    E 0    Z
21 N 7Y 1008 C 0
22    6T    0    H 0 2      NAS
23 U 1S   91         A Z
24 N 2T   22    K 0 A
25    2A    3    E 0    Z
26 N 2T   22    K 0 A
27    6T    0    H 0 2      NAS
28    2T    9    K 0 A
29 U 1S  104         A Z
30 N 1S   91         A Z
31 Y 6A    6    E 0
   DC
   DO
```

```
   DA    0 Z K 1
   DI
 0 Y 2T   6    K 2. A
 1   2S   8    E 0
 2 U 1S  105        A Z
 3 N 2T  15    K 1 A
 4   6T   0    H 0 2      NAS
 5 U 1S  91         A Z
 6 Y 2T  13    K 1 A
 7 U 1S  105        A Z
 8 Y 2T  13    K 1 A
 9 U 1S  84         A Z
10 Y 2T  13    K 1 A
11 U 1S  122        A Z
12 N 2T   4    K 1 A
13   7S   7    E 0
14   2T   4    K 2 A
15 U 1S  99         A Z
16 N 2T   4    K 2 A
17   6T   0    H 0 2      NAS
18 U 1S  63         A P
19 Y 7S   7    E 0
20 Y 2T   4    K 2 A
21 U 1S   9         A P
22 N 7Y  1009  C 0
23   6T   0    H 0 2      NAS
24 U 1S  63         A P
25 Y 2T  29    K 1 A
26 U 1S   9         A P
27 N 7Y  1009  C 0
28   2T  23    K 1 A
29 U 1S  90         A Z
30 N 7Y  1010  C 0
31   6T   0    H 0 2      NAS
   DC
   DO
```

```
   DA    0 Z K 2
   DI
0  U 1S   98          A Z
1  N 7Y   1011  C 0
2    2S   87          A
3    6S    8    E 0
4    2A    1          A
5    6A    6    E 0
6    2A    8    E 0
7    4A    5    E 1
8    2T   11    X 0     E
   DC
   DO
```

RUND(read until next delimiter)

```
   DA    0 Z L 0
   DI
 0    6T    0    K 0 3       NRS
 1    2S   12    E 0         Flag word
 2    2LS  31       A
 3    5S   12    E 0
 4 U  1A   63       A P
 5 U  1A    9       A E
 6 Y  2T   25    L 1 A       non-identifier
 7    2S    2       A
 8    4S   12    E 0
 9    2S    0       A
10    6S   13    E 0
11    2B   19    E 0         NLSC
12    6B   20    E 0         TNLSC
13    3S    0       A
14    1P    6    SA
15    2B   20    E 0
16    6S    0    X 0 B
17    6T    0    K 0 3       NRS
18 U  1A   63       A P
19 Y  2T   31    L 0 A
20    2S    1       A
21    4S   13    E 0
22    2S   13    E 0
23    2LS   3       A Z
24    2B   20    E 0
25 N  2S    0    X 0 B
26 N  2T   14    L 0 A
27    0B    1       A
28 U  1B   23    E 0   P
29 N  2T   12    L 0 A
30    7P
31 U  1A   88       A Z
   DC
   DO
```

```
   DA    0 Z L 1
   DI
 0 N 1A  89        A Z
 1 Y 7Y  1012 C 0
 2   2A   8   E 0
 3   2B   1       A
 4 U 1A  104       A Z
 5 Y 4B   9   E 0
 6 U 1A  105       A Z
 7 Y 5B   9   E 0
 8   2T  13   X 0   E     exit RUND
 9 U 1LA 89        A Z
10 Y 6S  16   E 0
11   2T  13   L 1 A
12   6T   0   K 0 3        NRS
13 U 1LA 88        A Z
14 N 2T  21   L 1 A
15   2S  20   E 1   Z
16 Y 7Y  1012 C 0
17   6S  18   E 0
18   2S   0       A
19   6S  20   E 1
20   2T  13   L 2 A
21 U 1LA 89        A Z
22 Y 2T   5   L 3 A
23 U 1A   9       A P
24   2T  12   X 0   Z
25   2B   0       A
26   6B  16   E 0
27   6B  15   E 0
28   6B  18   E 0
29   6B  17   E 0
30   2S   1       A
31   6S  20   E 1
   DC
   DO
```

```
   DA    0 Z L 2
   DI
 0    6T   9    L 1 4
 1 Y  2T  25    L 2 A        non-number
 2    2S  31    L 5
 3    1S  15    E 0   P
 4 Y  2S  16    E 0
 5 Y  0X  10        A
 6 Y  6S  16    E 0
 7 Y  2S  15    E 0
 8 Y  0X  10        A
 9 Y  6S  15    E 0
10    3S  18    E 0
11 N  0S   1        A
12    4S  17    E 0
13    6T  12    L 1 4
14 N  2T   2    L 2 A
15 U  1A  63        A P
16 N  7Y  1014 C 0
17    2S  18    E 0   Z
18 Y  2S  15    E 0   Z
19 N  2S   9        A
20 N  4S  12    E 0
21 N  2T   0    L 4 A
22    2S  13        A
23    4S  12    E 0
24    2T   3    L 1 A
25 U  1A  117       A Z      logical value
26 Y  2S   1        A
27 N  2S   0        A
28 U  1A  115       A E
29 Y  2T   3    L 1 A        to exit
30    6S  16    E 0
31    2S   5        A
   DC
   DO
```

```
   DA   0 Z L 3
   DI
 0    4S  12    E 0
 1    6T   0    K 0 3          NRS
 2 U  1A  76        A P
 3 N  7Y  1013 C 0
 4    2T  31    L 0 A
 5    2S   9        A
 6    4S  12    E 0
 7    6T   0    K 0 3          NRS
 8 U  1A  64        A Z
 9 Y  6T   0    K 0 3          NRS
10 Y  2T  18    L 3 A
11 U  1A  65        A Z
12 N  2T  18    L 3 A
13    6T   0    K 0 3          NRS
14 U  1A   9        A P
15 Y  7Y  1015 C 0
16    3A   8    E 0
17    2T  20    L 3 A
18 U  1A   9        A P
19 Y  7Y  1015 C 0
20    6A  18    E 0
21    2T  27    L 3 A
22    2S  18    E 0   P
23 N  5P        AA
24    0X  10        A Z
25 N  2T  17    L 5 A
26    6S  18    E 0
27    6T   0    K 0 3          NRS
28 U  1A   9        A P
29 N  2T  22    L 3 A
30    2S  18    E 0
31    4S  17    E 0
   DC
   DO
```

```
   DA    0 Z L 4
   DI
 0    3A  15   E 0
 1    3S  16   E 0
 2    6P       AS     Z
 3 Y  7S  17   E 0
 4 Y  2T  23   L 5 A
 5    1B  2100     A
 6    7B  18   E 0
 7    2B   8       A
 8 U  2A  17   E 0    P
 9 N  2T  21   L 4 A
10    7A  15   E 0
11    2T   3   L 5 A
12 U  0A  15   D14 B P
13 N  3P   1   AS
14    0D  15   D14 B
15    7S  15   E 0
16    0D  15   D14 B
17    3A  23   D14 B
18 N  0A   1       A
19    4A  18   E 0
20    3A  15   E 0
21 U  0B  17   E 0    P
22 Y  3B  17   E 0    Z
23 N  4B  17   E 0
24 N  2T  12   L 4 A
25    2T   7   L 5 A
26    2X  15   D14 B
27    3S  15   E 0
28    0X  15   D14 B
29    0P   1   AS     P
30 Y  1P   1   AS
31    7A  15   E 0
   DC
   DO
```

```
   DA    0 Z L 5
   DI
 0    2A  23    D14 B
 1 N  1A   1        A
 2    4A  18    E 0
 3 U  1B  17    E 0    P
 4 Y  2B  17    E 0    Z
 5 N  5B  17    E 0
 6 N  2T  26    L 4 A
 7    2S  18    E 0
 8 U  3LS 4095      A Z
 9 N  2T  17    L 5 A
10    6S  17    E 0
11    2S   1        A
12    6S  18    E 0
13    2A   8    E 0
14 U  1A  63        A P
15 Y  2T  31    L 0 A
16    7Y  1014  C 0
17    2S  29    L 5
18    6S  15    E 0
19    6S  16    E 0
20    2S  4095      A
21    6S  17    E 0
22    2T  11    L 5 A
23    2S  30    L 5
24    6S  15    E 0
25    2S   0        A
26    6S  16    E 0
27    6S  17    E 0
28    2T  11    L 5 A
   DN
29 +67108863
30 +33554432
31 +6710885
   DC
   DO
```

SPS(start prescan)

```
   DA    0 Z R 0
   DI
 0   2A  19       A
 1   6Y   2   X16
 2   0T  21   D21    P
 3   2A   0       A
 4   2B   0 E E 0 A
 5   7A   0   X 0 B
 6   0B   1       A
 7 U 1B   0 E T 0 A Z
 8 N 2T   5 Z R 0 A
 9   6A   0   E 0          initial state
10   6A   2   E 0          LC
11   6A   5   E 1          sum check
12   6A   9   E 1          own-flag
13   6A   9   E 0          BEC
14   6A  11   E 0          OBC
15   6A  30   E 0          BC
16   6A  14   E 0          DECFLA
17   6A   3   E 1          round bracket c.
18   6A   4   E 1          square br. counter
19   6A   2   E 1
20   6A   7   E 1
21   6A  11   E 1
22   2A   3   D 0
23   2LA  1       A
24   6A   8   E 1
25   2A   1       A
26   6A   6   E 1          LPSL
27   2A   0 E F 0 A
28   6A  29 Z E 0
29   6A  26   E 0          TLSC
30   2B   1-E T 0 A
31   6B  25 Z E 0          PLIE
   DC
   DO
```

```
   DA    0 Z R 1
   DI
 0   6B  23   E 0       PLIB
 1   1B   1       A
 2   6B   1   X 0 B
 3   2S   0 E K 0 A
 4   6S  22 Z E 0       INLSC
 5   6S  19   E 0       NLSC
 6   6S  27   E 0       BAP
 7   6T   0   T 0 0     APL
 8   2S   1       A
 9   2B  26   E 0
10   6S  32767 X 0 B
11   6T  18   U 3 1     6 * FTL(+o)
12   2S  11   E 0
13   6T   0   W 0 0     FTL(OBC)
14   3S   0       A
15   6T   0   W 0 0     FTL(-0)
16   2T   0   U 0 A     PSP
   DC
   DO
```

FPL(fill prescan list)

```
   DA    0 Z S 0
   DI
 0   2S    0        A
 1   2T    3    S 0 A
 2   2S    1        A
 3   0S   11    E 0
 4   0S   11    E 0
 5   6S    0    X 0
 6   2A   24    E 0
 7   2S   23    E 0 A
 8   4P         SB
 9   2S    0    X 0 B      search proper block cell
10   5A    0    X 0 B
11   4T    8    S 0 0 E
12   1S   23    E 0
13   1S   24    E 0
14   0S    1        A
15   6S    0    X 0
16   2B   23    E 0
17   2T   23    S 0 A
18   0B   24    E 0
19   2S    0    X 0 B
20   1B   24    E 0        shifting names in
21   6S    0    X 0 B      other block cells
22   0B    1        A
23   4T   18    S 0 0 E
24   6B   22    E 0        "INLSC"
25   2B   20    E 0
26 U 1B   19    E 0   Z
27   2S    0    X 0 B
28   2B   22    E 0        transfer name
29   0B   24    E 0
30   6S   32767 X 0 B
31 Y 2T    8    X 0   E
   DC
   DO
```

```
   DA    0 Z S 1
   DI
0    2B   1        A
1    5B  20    E 0          TNLSC
2    5B  22    E 0          "INLSC"
3    2T  25    S 0 A
   DC
   DO
```

APL(augment prescan list)

```
  DA    0 Z T 0
  DI
0   2B   2        A
1   6B  24   E 0
2   2B   1        A
3   0B  19   E 0          NLSC
4   6B  20   E 0          TNLSC
5   2A  25   E 0          PLIE
6   6A  32767 X 0 B
7   1A   1        A
8   6A   0   X 0 B
9   2T   0   S 0 A
  DC
  DO
```

FTL(fill translator list)

```
   DA    0 Z W 0
   DI
0    2B  26    E 0          TLSC
1  U 1B  27    E 0    Z     upper bound
2  Y 7P
3    6S   0    X 0 B
4    0B   1        A
5    6B  26    E 0
6    2T   8    X 0    E
   DC
   DO
```

PSP(prescan program)

```
   DA    0 Z U 0
   DI
 0   2B   0        A
 1   6B  28    E 0
 2   6T   0    L 0 5          RUND
 3   2S   8    E 0
 4 U 1S  84        A P
 5 N 2T   0    U 0 A
 6 U 1S  85        A Z
 7 Y 2T   1    U 2 A
 8 U 1S  89        A P
 9 N 2T   0    U 0 A
10 U 1S  90        A Z
11 Y 2T   0    U13 A
12 U 1S  91        A Z
13 Y 2T   0    U 4 A
14 U 1S  97        A P
15 N 2T   0    U 0 A
16 U 1S 101        A P
17 N 2T  23    U 1 A
18 U 1S 102        A Z
19 Y 2T  17    U 1 A
20 U 1S 103        A Z
21 Y 7Y 1018 C 0
22 U 1S 104        A Z
23 Y 2T   4    U 2 A
24 U 1S 105        A Z
25 Y 2T  10    U 4 A
26 U 1S 106        A Z
27 Y 2T   0    U 6 A
28 U 1S 108        A P
29 N 2T   0    U 7 A
30 U 1S 109        A Z
31 Y 2T   0    U10 A
   DC
   DO
```

```
   DA    0 Z U 1
   DI
 0 U 1S   110        A Z
 1 Y 2T    0   U11   A
 2 U 1S   111        A Z
 3 Y 2T    0   U14   A
 4 U 1S   112        A Z
 5 Y 2T    7   U15   A
 6 U 1S   115        A P
 7 N 2T   10   U 9   A
 8 U 1S   122        A Z
 9 N 7Y   1022 C 0
10   2T   28   U 4   A
11   1B   29   E 0     Z
12 N 7Y   1016 C 0
13   6T    0   U 3 0
14   7Y   1000 C 0                  end prescan
15   6T    0   K 0 3                NRS
16 U 1A   102        A Z
17   2S    1        A
18 Y 4S    3   E 0                  QC
19 U 1A   103        A Z
20 Y 5S    3   E 0     Z
21 N 2T   15   U 1   A
22   2T    0   U 0   A
23   2A    1        A
24 U 1S   98         A Z
25 Y 4A    3   E 1
26 U 1S   99         A Z
27 Y 5A    3   E 1
28 U 1S   100        A Z
29 Y 4A    4   E 1
30 U 1S   101        A Z
31 Y 5A    4   E 1
   DC
   DO
```

```
   DA    0 Z U 2
   DI
 0   2T   0   U 0 A
 1   6T  18   U 3 1
 2   6T   8   U 3 1
 3   2T   0   U 0 A
 4   2S 5000     A
 5   6T   0   W 0 0          FTL(begin)
 6   2A  28   E 0   Z
 7 N 2T   0   U 0 A
 8   6T   0   L 0 5          RUND
 9 U 1A 105       A P
10 U 1A 112       A E
11 Y 2A   0       A
12 N 2A   1       A
13   6A  14   E 0            DECFLA
14 Y 2T   3   U 0 A
15   3B   1       A
16   4B  26   E 0            TLSC
17   6T   8   U 3 1
18   6T  18   U 3 1
19   3S   1       A
20   6T   0   W 0 0          FTL(-1)
21   2S 5000     A
22   6T   0   W 0 0          FTL(begin)
23   2T   3   U 0 A
   DC
   DO
```

```
   DA    0 Z U 3
   DI
 0    2A    3    E 1    Z        bracket
 1 Y  2A    4    E 1    Z          counters
 2 Y  2T    8    X 0    E
 3    7Y 1027   C 0
 4    2S   12    E 0              Flag word
 5    2LS   2         A Z
 6 N  2T    8    X 0    E
 7    7Y 1026   C 0
 8    2S   11    E 0
 9    6T    0    W 0 0            FTL(OBC)
10    3S    0         A
11    6T    0    W 0 0            FTL(-0)
12    2S   30    E 0
13    0S    1         A
14    6S   30    E 0              BC
15    6S   11    E 0              OBC
16    6T    0    T 0 0            APL
17    2T    9    X 0    E
18    2S    6         A
19    6S    0    X 0
20    2S    0         A
21    6T    0    W 0 0            FTL(+0)
22    4T   20    U 3 0  P
23    2T    9    X 0    E
24    6T   11    U 5 1
25    2A    0    X 0 B
26    6A   11    E 0              restore OBC
27    2T   10    X 0    E
   DC
   DO
```

```
   DA    0 Z U 4
   DI
 0    6T    0    U 5 2
 1    0A  5000       A Z
 2 Y  6A   14    E 0              DECFLA
 3 N  1B    8         A
 4 N  6B   26    E 0
 5 N  6T   24    U 3 2
 6 N  2A    1         A
 7 N  6A   14    E 0
 8    6T    0    U 3 0
 9    2T    0    U 0 A
10    2B   26    E 0              TLSC
11    2A  32767 X 0 B
12 U  1A  5000       A Z
13 N  6T    0    U 5 2
14 N  2T   10    U 4 A
15    2A  32766 X 0 B
16    0A    1         A Z
17 N  2A    1         A
18 N  5A   26    E 0
19 Y  1B   10         A
20 Y  6B   26    E 0
21 Y  6T   24    U 3 2
22    2A    9    E 0    Z         BEC
23 N  2A    0         A
24 N  2T    7    U 4 A
25    6T    0    K 0 3            NRS
26    1A  122         A Z
27 N  7Y  1020 C 0
28    6T    0    U 5 2
29    2B   26    E 0
30    2T   11    U 1 A
   DC
   DO
```

```
   DA   0 Z U 5
   DI
 0   2B  26   E 0                    emptying translator list
 1   3A  32767 X 0 B P
 2 N 2T  10   X 0   E
 3   1B  10       A
 4   6B  26   E 0
 5   6T  11   U 5 1
 6   2A   8   X 0 B
 7   6A  11   E 0                    restore OBT
 8   0B   2       A
 9   6B  26   E 0
10   2T   0   U 5 A
11   2S  19   E 0                    NLSC
12   6S  20   E 0                    TNLSC
13   2A   6       A
14   6A   0   X 0
15   2B  26   E 0
16   2A   2   X 0 B
17   2B  20   E 0
18   6A   0   X 0 B
19   2B   1       A
20   4B  20   E 0
21   4B  26   E 0
22   4T  15   U 5 0 P
23   5B  20   E 0
24   0B  20   E 0
25   1B  19   E 0
26   6B  24   E 0
27   6T   0   S 0 0                  FPL
28   2B   6       A
29   5B  26   E 0
30   2B  26   E 0
31   2T   9   X 0   E
   DC
   DO
```

```
   DA   0 Z U 6
   DI
 0   2A   1        A
 1   6A   9   E 1              own-flag
 2   2B  26   E 0
 3   2A  32767 X 0 B
 4   1A  5000     A Z          begin ?
 5 N 7Y  1028 C 0
 6   2A  32766 X 0 B
 7   0A   1      A Z
 8 Y 1B   1      A
 9   2A  32765 X 0 B Z
10 Y 2A  32764 X 0 B Z
11 Y 2A   2      A
12 Y 4A   6   E 1
13   6T  19   U 6 6
14   1A  107      A Z
15 N 1A   1      A Z
16 N 1A   1      A Z
17 Y 2T   3   U 0 A
18   7Y  1029 C 0
19   6T   0   L 0 5            RUND
20   2S  14   E 0   Z          DECFLA
21 N 2T  14   X 0   E
22   7Y  1031 C 0
   DC
   DO
```

```
   DA   0 Z U 7
   DI
 0 U 1S  107       A Z
 1 Y 2A   1        A
 2 N 2A   2        A
 3   6A  11    E 1
 4   2A   9    E 1    Z      own-flag
 5   6T  19    U 6 6
 6 N 2T  19    U 7 A
 7 U 1A  110       A Z
 8 Y 2T   0    U11 A
 9 U 1A  112       A Z
10 Y 2T   7    U15 A
11   6T   0    U 8 6
12   6T  10    U 8 0
13   2A  32766 X 0 B
14   0A   1        A Z
15 Y 1B   1        A
16   2A   2    E 1
17   4A  32766 X 0 B
18   2T  20    U 9 A          initialization
19 U 1A  110       A Z
20 Y 2T   4    U11 A
21   6T   0    U 8 6
22   2B  26    E 0
23   2A  32766 X 0 B
24   0A   1        A Z
25 Y 1B   2        A
26 N 1B   1        A
27   2A   0        A
28   6A   9    E 1            own-flag
29   2T  16    U 7 A
   DC
   DO
```

```
   DA    0 Z U 8
   DI
 0    6T    4    U 3 0
 1    2A    8    E 0
 2 U  1A   87         A Z
 3    2S    1         A
 4    4S    2    E 1
 5 Y  6T    0    L 0 5          RUND
 6 Y  2T    0    U 8 A
 7 U  1A   91         A Z
 8 Y  2T   14    X 0    E
 9    7Y 1030 C 0
10    2B   26    E 0
11    2A  32767 X 0 B
12 U  1A  5000       A Z
13 N  2T   20    U 9 A          initialization
14    2T    8    X 0    E
15    2S   64         A
16    2T   26    U 8 A
17    2S   11    E 1    Z
18 Y  2S   32         A
19 Y  2T   26    U 8 A
20 U  1S    1         A Z
21 Y  2S  288         A
22 Y  2T   26    U 8 A
23 U  1S    2         A Z
24 Y  2S  292         A
25 N  2S  293         A
26    2P   16    SS
27    2T    8    X 0    E
28    2S   20    E 0            TNLSC
29    0S    2         A
30    6S   20    E 0
31    1S   19    E 0
   DC
   DO
```

```
   DA    0 Z U 9
   DI
 0    0S   1         A
 1    6S  24    E 0
 2    2T   8    X 0    E
 3    2B  20    E 0
 4    2S   0    E 1
 5    2A   1    E 1
 6    6S   0    X 0 B
 7    6A  32767 X 0 B
 8    6T   0    S 0 0          FPL
 9    2T   9    X 0    E
10    2A  28    E 0    Z
11  N 2B  26    E 0
12  N 2A  32767 X 0 B
13  N 1A  5000       A Z
14  Y 7Y  1032  C 0
15    6T   0    L 0 5          RUND
16  U 1A  87         A Z
17  Y 2T  15    U 9 A
18  U 1A  91         A Z
19  N 7Y  1033  C 0
20    2S   0         A
21    6S   2    E 1
22    6S   7    E 1
23    6S  10    E 1
24    6S  11    E 1
25    2T   2    U 0 A
   DC
   DO
```

```
   DA    0 Z U10
   DI
 0    2A    3          A
 1    6A   11    E 1                    own-flag
 2    2A    9    E 1     Z
 3    6T   19    U 6 6
 4 N  2T   16    U10 A
 5 U  1A  110          A Z
 6 Y  2T    0    U11 A
 7 U  1A  112          A Z
 8 Y  2T    7    U15 A
 9    6T    0    U 8 6
10    6T   10    U 8 0
11    2A    8    E 1     Z              ALP ?
12    2A    2    E 1
13 N  0A    2    E 1
14    6A    2    E 1
15    2T   13    U 7 A
16 U  1A  110          A Z
17 Y  2T    4    U11 A
18    6T    0    U 8 6
19    2A    8    E 1     Z
20    2A    2    E 1
21 N  0A    2    E 1
22    6A    2    E 1
23    2T   22    U 7 A
24 U  1A  105          A Z
25 Y  7Y  1024 C 0
26    1A   99          A Z
27    2T    8    X 0
   DC
   DO
```

```
   DA    0 Z U11
   DI
 0    2B  26    E 0
 1    2A  32767 X 0 B
 2    1A  5000      A Z
 3 N  2T  10    U 9 A
 4    2A   0        A
 5    6A   2    E 1
 6    6A   7    E 1
 7    6A  10    E 1
 8    6T  19    U 6 6
 9    6T   4    U 3 0
10 U  1A  87        A Z
11    2S   1        A
12    4S   7    E 1
13 Y  2T   8    U11 A
14 U  1A  100       A Z
15 N  7Y  1034 C 0
16    2S   1        A
17    7S   3 Z E 1                colon counter
18    4S   4 Z E 1                squ.br.counter
19    6T   0 Z L 0 5              RUND
20    2S   1        A
21 U  1A  100       A Z
22 Y  2T  18 Z U11 A
23 U  1A  90        A Z
24 N  2T  16 Y E 5 A              insertion
25 U  1S   4 Z E 1   Z
26 N  7Y  1034 C 0
27    4S  10 Z E 1                dimension counter
28    4S   3 Z E 1
29    2T  19 Z U11 A
   DC
   DO
```

```
   DA    0 Z U12
   DI
 0    2A    9    E 1    Z          own-flag
 1 Y  2T    9    U12 A
 2    2B   26    E 0
 3    2S   32766 X 0 B
 4    0S    1         A Z
 5 Y  1B    1         A
 6    2S    7    E 1
 7    4S   32761 X 0 B
 8    4S    6    E 1               LPSL
 9    2A    9    E 1    Z
10    2S   10    E 1
11 N  0S   10    E 1
12 N  0S    6         A
13 Y  0S    4         A
14    2X    7    E 1
15    4S    2    E 1
16    6T    0    L 0 5             RUND
17 U  1A   87         A Z
18 Y  2A    0         A
19 Y  2T    6    U11 A
20 U  1A   91         A Z
21 N  7Y  1035 C 0
22    2B   26    E 0
23    2A    9    E 1    Z
24 Y  2T   13    U 7 A
25    2A   32766 X 0 B
26    0A    1         A Z
27 Y  1B    3         A
28 N  1B    2         A
29    2T   27    U 7 A
   DC
   DO
```

```
   DA    0 Z U13
   DI
 0    6T   4    U 3 0
 1    2B  26    E 0
 2    3A  32767 X 0 B P
 3    2S   1         A
 4 Y  2T  18    U13 A
 5    2A  32767 X 0 B
 6    1A  5000      A Z
 7 Y  2A  32766 X 0 B
 8 Y  0A   1         A Z
 9 Y  2T  18    U13 A
10    2A  5000      A
11    1A  32767 X 0 B P
12 Y  0B   2         A
13 Y  2T  18    U13 A
14    2A  5000      A
15    1A  32767 X 0 B Z
16 Y  1B   1         A
17 Y  2T   2    U13 A
18    4S  32762 X 0 B
19    6T  28    U 8 0
20    2S   0         A
21    6S   1    E 1                  name code II
22    6T  15    U 8 0
23    0S  24    E 0
24    6S   0    E 1                  name code I
25    6T   3    U 9 1
26    2S   0         A
27    6S  14 Z  E 0                  DECFLA
28    2T   0 Z  U 0 A
   DC
   DO
```

```
   DA    0 Z U14
   DI
 0    2A  28    E 0    Z
 1 N  2T  10    U 9 A
 2    6T  19    U 6 6
 3    1A  92        A Z
 4 N  7Y  1021  C 0
 5    6T   4    U 3 0
 6    6T  28    U 8 0
 7    6T   0    K 0 3          NRS
 8    2S   1        A
 9 U  1A  100       A Z
10 Y  4S   3    E 1
11 U  1A  101       A Z
12 Y  5S   3    E 1
13    2B   3    E 1    Z
14 N  2T   7    U14 A
15 U  1A  87        A Z
16 Y  4S   2    E 1
17 U  1A  91        A Z
18 N  2T   7    U14 A
19    4S   2    E 1
20    2P  20    SS
21    0S  24    E 0
22    2B  20    E 0
23    6S   0    X 0 B
24    2A   2    E 1
25 U  1A  31        A P
26    0T   0        A
27    2P  16    AA
28    6A  32767 X 0 B
29    6T   0    S 0 0          FPL
30    2A   2    E 1
31    0A   2    E 1
   DC
   DO
```

```
   DA    0 Z U15
   DI
 0    0A   1         A
 1    2B  26    E 0
 2    2S  32766 X 0 B
 3    0S   1         A Z
 4 Y  1B   1         A
 5    4A  32762 X 0 B
 6    2T  20    U 9 A          initializations
 7    2A  28    E 0   Z
 8 N  2T  10    U 9 A
 9    2A   1         A
10    6A  28    E 0
11    6T   0    L 0 5          RUND
12    6T   4    U 3 0
13    6T  28    U 8 0
14    0T   0         A
15    2S   8    E 0
16 U  1S  91         A Z
17 Y  2T   1    U16 A
18 U  1S  98         A Z
19 N  7Y  1023 C 0
20    2A   1         A
21    4A   2    E 1
22    6T   0    K 0 3          NRS
23 U  1A  87         A Z
24 Y  2T  20    U15 A
25    6T  24 Z U10 0
26 N  2T  22    U15 A
27    6T   0    L 0 5          RUND
28    2S  12    E 0
29    2LS  0         A Z
30 N  7Y  1024 C 0
31    1A  91         A Z
   DC
   DO
```

```
   DA    0 Z U16
   DI
 0 N 7Y  1025 C 0
 1   2A   2   E 1
 2 U 1A  31       A P
 3 Y 7Y  1017 C 0
 4   2B  20   E 0          TNLSC
 5   2P  16   AA
 6   6A  32767 X 0 B
 7   6T  17   U 8 0
 8   0S  24   E 0
 9   6S   0   X 0 B
10   6T   2   S 0 0        FPL
11   6T   8   U 3 1
12   6T  18   U 3 1
13   2T  20   U 9 A
   DC
   DO
```

basic cycle

```
   DA    0 Z Y 0
   DI
 0   6T    0    L 0 5          RUND
 1   2A   12    E 0            Flag word
 2 U 2LA   2          A Z      NFLA
 3 Y 2T    6    Y 0 A
 4   2LA   0 F Z 0      Z      NIFLA
 5 Y 6T    0 F E 0 1          LFN
 6   2B    8 Z E 0            delimiter
 7 U 1B   65          A P
 8 N 2T    0 F H 0 A          + -
 9 U 1B   69          A P
10 N 2T    0 F L 0 A          * / ÷ ↑
11 U 1B   75          A P
12 N 2T    0 F K 0 A          relational operators
13 U 1B   76          A Z
14 Y 2T    0 F R 0 A          logical "not"
15 U 1B   80          A P
16 N 2T    0 F S 0 A          other log. operators
17 U 1B   122         A Z
18 Y 2T    0 F T 0 A          "?" or stopcode
19   2T   61-Z Y 0 B
20   2A    0 F W 0            go to
21   0A    0 F U 0            if
22   0A    0 F Y 0            then
23   0A    0 F N 0            else
24   0A    0 H Z 0            for
25   0A    0 H E 0            do
26   0A    0 H F 0            ,
27   0B  1036 X 0             .
28   0B  1036 X 0             "small 10"
29   0A    0 H H 0            :
30   0A    0 H K 0            ;
31   0A    0 H L 0            :=
   DC
   DO
```

```
   DA    0 Z Y 1
   DI
 0    OB   1036 X 0          "space"
 1    OA    0 H R 0          step
 2    OA    0 H S 0          until
 3    OA    0 H T 0          while
 4    OB   1036 X 0          comment
 5    OA    0 H W 0          (
 6    OA    0 H U 0          )
 7    OA    0 H Y 0          opening square bracket
 8    OA    0 H N 0          closing square bracket
 9    OA    0 K Z 0          opening string quote
10    OB   1036 X 0          closing string quote
11    OA    0 Z N 0          begin
12    OA    0 K E 0          end
13    OA    0 K F 0          own
14    OA    0 K H 0          Boolean
15    OA    0 K K 0          integer
16    OA    0 K L 0          real
17    OA    0 K R 0          array
18    OA    0 K S 0          switch
19    OA    0 K T 0          procedure
20    OA    0 K W 0          string
21    OA    0 K U 0          label
22    OA    0 K Y 0          value
   DC
   DO
```

begin

```
   DA    0 Z N 0
   DI
 0   2A   12    E 0
 1 U 2LA 16384     A Z        PROFLA
 2 Y 4P          AS
 3 Y 6T    0     W 0 0        FTL(Flag word)
 4   2S    0 F Z 0            BbeginP
 5   2B   26 Z E 0
 6 N 1B    1          A
 7 Y 0S    2          A        BbeginC
 8   6T    1     W 0 0        FTL(bracket)
 9   2B    1 F Z 0 A
10   6T    1 K N 0 0          require
11 N 7Y  1037 C 0
12   6T    0 L Z 0 0          put
13   2T    0 Z Y 0 A          to basic cycle
   DC
   DO
```

constants

```
   DA    0 F Z 0
   DN
 0 +0.25
   DU
 1 +11011111111101111111111111
   DN
 2 +0.5
   DU
 3 +10000000010000000000000
 4 +0
 5 +10000000000000000000
 6 +11000000001110111111110000
   DN
 7 +0
   DU
 8 +10011000000000000000000
   DN
 9 +0.125
   DU
10 +10001000000001001000
11 +11000000001110011111111000
12 +1000
13 +10011000000010000000000
14 +10000000000000000000000
   DN
15 +0.03125
16 +0.375
   DU
17 +100000000000000000
18 +1000000000000000000
19 +100000000000000000
20 +11101000000000000000
21 +11110000000000000000
22 +11000000000000000000000
23 +10000000000000000000000011
24 +10000000000000000000001001
25 +10000000000000000000000111
26 +10110000000000000000000000
27 +10000000000000000000001010
28 +10000000000000000000001100
29 +10111011110110111111100000
30 +0
31 +10000000000000000000000001
   DC
   DO
```

```
   DA    0 F Z 1
   DU
 0 +1 0000000000000000000000010
 1 +1 1110000000001101 11
 2 +1 00000000000000000000000
 3 +1 000000000000000000
 4 +1 0000000000000000000001 000
 5 +1 001 00000000000000000000000
 6 +1 01 00000000000000000000000
 7 +1 001 00000000000000000000000
 8 +1 0011 0000000000000000000000
 9 +1 01 0000000000000000000000000
10 +1 0001 000000000000000000000000
11 +1 01 01 000000000000000000000000
12 +1 01 0000000000000000000000
13 +1 000000000000000000
14 +1 10000000000000000000000
15 +1111111111111111111111111
16 +1 0000000000000000
   DC
   DO
```

LFN(look for name)

```
   DA    0 F E 0
   DI
 0    3B  19 Z E 0              NLSC
 1    7B  16   E 1
 2    7B   3   E 2
 3    0B   3       A
 4    0B  20   E 0              TNLSC
 5    6B  15   E 1
 6    2B  15   E 1
 7    1B   2       A
 8    6B   0   X 0
 9    2B  16   E 1
10    6T  16 F E 1 0
11    2S  32767 X 0 B           name code I
12    4P        SB
13 U 1B  15 Z E 1   Z
14 N 5B  16   E 1
15 N 2T   9 F E 0 A
16    2B  16 Z E 1
17    6B  17   E 1
18    2B  20   E 0
19    6B  18   E 1
20    2B  17   E 1
21    2S  32765 X 0 B
22    2B  18   E 1
23    1S   0   X 0 B Z          compare name word
24 N 2B  15   E 1
25 N 5B  16   E 1
26 N 2T   6 F E 0 A
27    2B   1       A
28    5B  17 Z E 1
29    5B  18   E 1
30    4T  20 F E 0 0 P
31    2B  16 Z E 1
   DC
   DO
```

```
   DA    0 F E 1
   DI
 0    2S   32767 X 0 B
 1    2A   32766 X 0 B
 2    6S    0 Z E 1          name code I
 3    6A    1   E 1          name code II
 4    2A    0       A
 5    3P   16   SS
 6 U  2LS  80       A Z      non-designational ?
 7 N  2A   16       A        DEFLA
 8 N  2T   14 F E 1 A
 9 U  2LS   4       A Z      non-arithmetical ?
10 Y  2T   14 F E 1 A
11    2A    8       A
12 U  2LS   1       A Z
13 Y  OA    4       A
14    4A   12 Z E 0          Flag word
15    2T    9   X 0   E
16 U  1B   22   E 0   P
17 Y  2T    8   X 0   E
18    2A    3   E 2   Z
19 Y  2T   25 F E 1 A
20    2B    0 L N 0 A        switch over to fixed
21    0B    0 E R 0 A        part of name list
22    6B   16 Z E 1
23    2A    0       A
24    6A    3   E 2
25 U  1B    0 L N 0 A P
26 Y  2T    8   X 0   E
27    7Y  1048  C 0
   DC
   DO
```

FNL(fill name list)

```
  DA   0 F F 0
  DI
0   2B  20 Z E 0          TNLSC
1   0B   3        A
2   6B  19   E 0          NLSC
3 U 1B  23   E 0   P      upper bound
4 Y 7P
5   2A   0   E 1          name code I
6   2S   1   E 1          name code II
7   6A  32767 X 0 B
8   6S  32766 X 0 B
9   2T   8   X 0   E
  DC
  DO
```

+ -

```
   DA    0 F H 0
   DI
 0    2A   12 Z E 0          Flag word
 1 U  2LA   3       A Z      NFLA & KFLA
 2 N  2T   15 F H 0 A
 3 U  2LA   5 F Z 0   Z      OFLA
 4 Y  2T   15 F H 0 A
 5    2B    6 F Z 0 A        non-operational +
 6    6T    1 K N 0 0        require
 7 N  7Y   1038 C 0
 8    6T    0 L Z 0 0        put
 9    2A    8 Z E 0
10    1A   64       A Z
11 Y  2T    0   Y 0 A
12    6T    0 L E 0 4        PCP
13    2S   10 F Z 0           NEG
14    2T   25   H 0 A
15    2A   15       A
16    6T    0 L F 0 5        NNP
17    2B   11 F Z 0 A
18    6T    0 K N 0 0        require
19 N  7Y   1038 C 0
20    6T    0 L Z 0 0        put
21    2A    8 Z E 0          delimiter
22    OLA  64       A Z
23    2S    1 F Z 1          ADD
24 N  OS    1       A        SUB
25    6T    0 Z W 0 0        FTL(macro-order)
26    2T    0   Y 0 A
   DC
   DO
```

relational operators

```
   DA    0 F K 0
   DI
 0   2A  13         A
 1   6T   0 L F 0 5          NNP
 2   6T   0 R F 0 6          conditional expressions
 3   2B  11 F Z 0 A
 4   6T   0 K N 0 0          require
 5 N 7Y  1067 C 0
 6   2B  29 Y E 1 A
 7   6T   0 L Z 0 0          put
 8   2S   1 Y E 2
 9   0S   8 Z E 0            delimiter
10   1S   9         A
11   2T  25 F H 0 A          FTL etc.
   DC
   DO
```

multiplying operators and to the power

```
   DA   0 F L 0
   DI
 0   2A  17       A
 1   2S   8 Z E 0          delimiter
 2   0LS 69       A Z       to the power ?
 3 Y 2A  19       A
 4   6T   0 L F 0 5         NNP
 5   2B  11 F Z 0 A
 6   6T   0 K N 0 0         require
 7 N 7Y  1039 C 0
 8   6T   0 L Z 0 0         put
 9   2S   8 Z E 0
10 U 0LS 69       A Z
11   0S  10 F Z 0
12   1S  81       A
13 Y 0S  19 F Z 0
14   2T  25 F H 0 A         FTL etc.
   DC
   DO
```

logical "not"

```
   DA   0 F R 0
   DI
 0   2B   2 Y E 2 A
 1   6T   0 K N 0 0      require
 2 N 7Y  1068 C 0
 3   2B  30 Y E 4 A
 4   6T   0 L Z 0 0      put
 5   2B  26 Z E 0
 6   2S  32767 X 0 B
 7   1S   4 Y E 2   Z    forbid repeated"not"
 8 Y 7Y  1068 C 0
 9   6T   0 L E 0 4      PCP
10   2S   4 Y E 2
11   2T  25 F H 0 A      FTL etc.
   DC
   DO
```

logical operators

```
   DA    0 F S 0
   DI
 0   2A   88        A
 1   1A    8 Z E 0
 2   6T    0 L F 0 5        NNP
 3   6T    0 R F 0 6        conditional expressions
 4   2B    5 Y E 2 A
 5   6T    0 K N 0 0        require
 6 N 7Y  1069 C 0
 7   6T    0 L Z 0 0        put
 8   2A   12 Z E 0
 9   2LA   3        A Z
10 Y 7Y  1069 C 0
11   2S   88        A
12   1S    8 Z E 0
13   2P   15   SS
14   0S    8 Z E 0          delimiter
15   1S    9        A
16   2T   25 F H 0 A        FTL etc.
   DC
   DO
```

? or stopcode

```
   DA   0 F T 0
   DI
 0   6T   0 R W 0 5       backward scanner V
 1   2B   0       A
 2   2S  73       A
 3   6T   0 U E 0 2       FOT(stop)
 4   2B   1       A
 5   7B   0   Z 0
 6   2S  26 Z E 1
 7 U 1S   6   E 1   Z     PPP = LPSL ?
 8 Y 2S  12   E 1   Z     APC
 9 Y 2S  11   E 0   Z     OBC
10 Y 2S  19   E 0
11 Y 1S  22   E 0   Z     NLSC = INLSC
12 Y 2S  26   E 0
13 Y 1S  29   E 0   Z     TLSC = ITLSC
14 Y 3B  25   E 0
15 Y 0B  23   E 0   Z     PLIE = PLIB
16 Y 2S   5   E 1   Z     sum check
17 N 7Y 1036  C 0         something wrong
18   2B   6   E 1   Z
19 N 1B   1       A
20 N 6B   6   E 1
21 N 0B  25   E 1         PSLB
22 N 2A   0   X 0 B
23 N 4P       AS
24 N 3P  21   AA
25 N 4P       AB
26 N 2LS 32767    A
27 N 6T   0 U E 0 2       FOT(prestack list)
28 N 2T  18 F T 0 A
29   2A   3   D 0   P
30 Y 6T   0 U R 0 3       punch end of object tape
31   2A   8       A
   DC
   DO
```

```
  DA    0 F T 1
  DI
0 U 2LA 26    X 0    Z
1 Y 2T  31    T 0 A
2   7Y  1001  C 0        end of translation
  DC
  DO
```

go to

```
   DA    0 F W 0
   DI
0    2B    0 Y Z 0 A
1    6T    0 K N 0 0        require
2 N 7Y   1056 C 0
3    6T    0 L Z 0 0        put
4    2S   12 Y Z 0
5    6T    0 Z W 0 0        FTL(JU)
6    2T    0   Y 0 A
   DC
   DO
```

if

```
   DA    O F U O
   DI
 0   2S  12 Z E O
 1 U 2LS  2 F Z O    Z      SFLA
 2 Y 2T  11   U O A         if E
 3   2B   9 Y E 2 A
 4   6T   0 K N O O         require
 5 N 7Y  1070 C O
 6   6T   0 L Z O O         put
 7   2S  12 Z E O
 8   6T   0   W O O         FTL(Flag word)
 9   2S  15 Y E 2           BifS
10   2T  19 F U O A
11   2B  13 Y E 2 A
12   6T   0 K N O O         require
13 N 7Y  1070 C O
14   6T   0 L E O 4         PCP
15   6T   0 W L O 4         PSE
16   2S  12 Z E O
17   6T   0   W O O         FTL(Flag word)
18   2S  16 Y E 2           BifE
19   6T   0 Z W O O         FTL(bracket)
20   2T   0   Y O A
   DC
   DO
```

then

```
   DA    0 F Y 0
   DI
 0   2A    0        A
 1   6T    0 L F 0 5          NNP
 2   6T    0 R F 0 6          conditional expressions
 3   2S   12 Z E 0
 4 U 2LS   3        A Z
 5 Y 7Y  1071 C 0
 6   2B   17 Y E 2 A
 7   6T    0 K N 0 0          require
 8 N 7Y  1071 C 0
 9   6T    0 L Z 0 0          put
10   2B   26 Z E 0
11   2S   32767 X 0 B
12 U 1S   15 Y E 2   Z        BifS on transl. list
13 Y 2B   11 Y E 3 A
14 Y 6T    0 L Z 0 0          SFLA & OFLA
15   2B   26 Z E 0
16   2S   32767 X 0 B
17 Y 2T   21 F Y 0 A
18 U 1S   16 Y E 2   Z
19 N 7Y  1071 C 0
20   2S   16 Y E 2
21   1S    2        A
22   6S   32767 X 0 B        replace bracket
23   2B    0        A
24   2S    2        A
25   6T    0 U E 0 2          FOT(ICOJU)
26   2B   12        A
27   6T    0 U E 0 2          FOT(D12)
28   2T    0 Z Y 0 A
   DC
   DO
```

else

```
   DA    0 F N 0
   DI
 0    2A    0         A
 1    6T    0 L F 0 5          NNP
 2    6T    0 R F 0 6          conditional expressions
 3    6T    0 R H 0 6          for statements
 4    2B   21 Y E 2 A
 5    6T    0 K N 0 0          require
 6 N  7Y  1072 C 0
 7    6T    0 L Z 0 0          put
 8    2B   26 Z E 0
 9    2S   32767 X 0 B
10    0S    2         A
11 U 1S   15 Y E 2    Z        BthenS on translator list
12 Y 2S    2 F Z 0
13 Y 4S   12 Z E 0             SFLA
14 Y 2T   22 F N 0 A
15 U 1S   16 Y E 2    Z
16 N 7Y  1072 C 0
17    2S    5 F Z 0
18    4S   12 Z E 0             OFLA
19    2S   12    E 0
20    2LS  31         A Z
21 Y 7Y  1072 C 0
22    6T    0 L W 0 3          implicit jump
23    2B    1         A
24    5B   26 Z E 0
25    6T    0 S F 0 1          FTL(this moment's PRA)
26    2S   15 Y E 2
27 Y 1S    4         A        BelseS
28 N 1S    5         A        BelseE
29    6T    0 Z W 0 0          FTL(bracket)
30 N 2B   26    E 0
31 N 2S   32765 X 0 B
   DC
   DO
```

```
  DA    0 F N 1
  DI
0 N 3LS 31          A          store T in Flag word
1 N 6S  32765 X 0 B            in translator list
2 N 2S  12 Z E 0
3 N 2LS 31          A
4 N 4S  32765 X 0 B
5 Y 2B   4 Y E 5 A
6 Y 6T   0 L Z 0 0             FOFLA := 0
7   2T   0 Z Y 0 A
  DC
  DO
```

for

```
   DA    0 H Z 0
   DI
0    2B    3 Y E 3 A
1    6T    0 K N 0 0          require
2  N 7Y  1074 C 0
3    6T    0 L Z 0 0          put
4    6T    1 S E 0 4          INB
5    2T    0 Z Y 0 A
   DC
   DO
```

do

```
   DA    0 H E 0
   DI
 0   2A    0         A
 1   6T    0 L F 0 5          NNP
 2   6T    0 R F 0 6          conditional expressions
 3   2B   11 Y E 3 A
 4   6T    0 K N 0 0          require
 5 N 7Y  1076 C 0
 6   6T    0 L Z 0 0          put
 7   2S   12 Z E 0
 8 U 2LS  15         A Z
 9 Y 7Y  1076 C 0
10 U 2LS  15 Y E 3    Z       WIFLA
11 Y 2T   18 H E 0 A
12 U 2LS   8         A Z
13 N 7Y  1076 C 0
14   2S   15 Y E 3
15   5S   12 Z E 0
16   2S   21         A
17   2T   25 H E 0 A
18 U 2LS   8         A Z
19 Y 7Y  1076 C 0
20 U 2LS  16 Y E 3    Z       SUFLA
21 N 2S   16 Y E 3
22 N 5S   12 Z E 0
23 N 2S   24         A
24 Y 2S   20         A
25   2B    0         A
26   6T    0 U E 0 2          FOT(FOR2 or 3 or 6)
27   2B    0         A
28   2S    3         A
29   6T    0 U E 0 2          FOT(RET)
30   2S    4 F Z 1
31   2B   26 Z E 0
   DC
   DO
```

```
   DA    0 H E 1
   DI
 0 U 1S  32767 X 0 B Z        Bfor on translator list
 1 N 7Y  1076 C 0
 2   1S   1       A
 3   6S  32767 X 0 B          replace by Bdo
 4   2S. 13 U Z 0
 5   2P  15   SS
 6   0LS 12 U Z 0
 7   0LS 26 F Z 0
 8   6S  32763 X 0 B          this moment"s PRA
 9   2B   0       A
10   2S  18       A
11   6T   0 U E 0 2           FOT(FORO)
12   2T  15 H E 1 A
13   6T   0 U E 0 2           FOT(D12)
14   2T   0 Z Y 0 A
15   2B   5       A
16   2S   8       A
17   0S  30 Z E 1
18   0S  30   E 1             2 * number of labels
19   6T   0 U E 0 2           FOT(LN
20   2B  12       A
21   2T  13 H E 1 A
   DC
   DO
```

distribution of ","

```
   DA    0 H F 0
   DI
 0   2A   12 Z E 0
 1 U 2LA   4 Y E 0    Z        VAFLA
 2 N 2T    0 T H 0 A           in value list
 3 U 2LA 16384      A Z        PROFLA
 4 Y 2T    8 H F 0 A
 5 U 2LA   0 F Z 0    Z        NIFLA
 6 N 2T    0 T K 0 A           formal parameter part
 7   2T    0 T L 0 A           specification part
 8 U 2LA 1024       A Z        AFLA
 9 N 2T    0 T R 0 A           array declaration
10 U 2LA 4096       A Z        SIFLA
11 N 2T    0 T S 0 A           switch
12 U 2LA 224        A Z
13 N 2T    0 T T 0 A           type list
14 U 2LA   2 F Z 1    Z        PAFLA
15 N 2T    0 T U 0 A           actual parameter list
16   2A    0          A
17   6T    0 L F 0 5           NNP
18   6T    0 R F 0 6           conditional expressions
19   2A   12 Z E 0             FOFLA
20 U 2LA  13 F Z 1    Z        for list
21 N 2T    0 T Y 0 A
22   2B   26 Z E 0
23   2S  32767 X 0 B           Bindex on translator list ?
24 U OLS   0 Y E 1    Z
25 Y 2T    0 T N 0 A           index separator
26 U OLS   1 Y E 1    Z        Bboundpair on transl.list?
27 Y 2T    0 W Z 0 A           bound pair separator
28   7Y  1063 C 0
   DC
   DO
```

":"

```
   DA    0 H H 0
   DI
 0   2A   12 Z E 0
 1 U 2LA   2 F Z 0   Z     SFLA
 2 Y 2T   16 H H 0 A       bound pair
 3   2B    8 Y Z 0 A
 4   6T    0 K N 0 0       require
 5 N 7Y  1055 C 0
 6   6T    0 L Z 0 0       put
 7   2A    1 Z E 1         name code II
 8   3P   16    AA         isolate  number in
 9   4P         AB         address pile
10   2A   13 U Z 0
11   2P   15    AA
12   0A   12    Z 0        this moment's PRA
13   0B   27 Z E 0         BAP
14   6A    0    X 0 B
15   2T    0 Z Y 0 A
16   2A    0      A
17   6T    0 L F 0 5       NNP
18   6T    0 R F 0 6       conditional expressions
19   2B    4 Y Z 0 A
20   6T    0 K N 0 0       require
21 N 7Y  1055 C 0
22   6T    0 L Z 0 0       put
23   2T    0 Z Y 0 A
   DC
   DO
```

distribution of ";"

```
   DA   0 H K 0
   DI
 0   2A  12 Z E 0
 1 U 2LA  2 F Z 0   Z     SFLA
 2 N 2T  16 H K 0 A       end of statement
 3 U 2LA 16384     A Z    PROFLA
 4 Y 2T  10 H K 0 A       not in proc.heading
 5 U 2LA 16 F Z 1   Z     VAFLA
 6 N 2T   0 S N 0 A       value part
 7 U 2LA 24 Y E 1   Z
 8 N 2T  24 H K 0 A
 9   2T   0 S U 0 A       formal parameter part
10 U 2LA 4096      A Z    SIFLA
11 N 2T   0 T Z 0 A       switch declaration
12 U 2LA 1024      A Z    AFLA
13 N 2T   0 T E 0 A       array declaration
14 U 2LA 224       A Z
15 N 2T   0 T F 0 A       variable declaration
16   6T   0 R L 0 7       end of statement
17 N 7Y  1054 C 0
18   2B  26 Z E 0
19   2A  32767 X 0 B
20   0LA 23 F Z 0   Z     Bproc on translator list?
21 Y 6T   0 R W 0 5       backward scanner V
22 Y 6T  13 R F 0 3       FOT(this moment'sPRA)
23   2T   0 Z Y 0 A
24   2A   0 Z E 1         name code I
25 U 2LA 17 F Z 0   Z     non-formal ?
26 N 2T   0 S Y 0 A       specification part
27 U 2LA 15 F Z 0   Z     require function
28 N 2LA  0 F Z 0   Z     procedure
29 Y 7Y  1036 C 0
30   2T   0 S U 0 A       formal parameter part
   DC
   DO
```

:=

```
   DA    0 H L 0
   DI
 0    2A    0        A
 1    6A    5 Z E 2
 2    2S   12    E 0
 3 U 2LS 4096      A Z     SIFLA
 4 N 2T    0 W H 0 A       := in switch declar.
 5 U 2LS 2048      A Z     LEFLA
 6 Y 2T   13 H L 0 A       no subscript expr.to the left
 7    2B   26 Z E 0
 8    2A 32767 X 0 B       ITAR on translator list ?
 9    0LA 14 Y E 1    Z
10 N 7Y   1065 C 0
11    2B    1        A
12    5B   26 Z E 0        postpone indexing
13 U 2LS   3 F Z 1    Z    FOFLA
14 N 2T    0 W K 0 A       := in for statement
15    2A   49        A      ST
16    0A   16 F Z 1
17    6A    6 Z E 2         macro-order
18    2S    1        A
19    6S    5    E 2
20    2A    0    E 1        name code I
21 U 2LA 15 F Z 0    Z
22 Y 2T   11 H L 1 A       non-procedure
23    4S    5 Z E 2
24 U 2LA 19 F Z 0    Z
25 Y 2A   51        A      STIP
26 N 2A   52        A      STRP
27    0A   16 F Z 1
28    6A    6 Z E 2
29    2B   26    E 0
30    2S 32767 X 0 B
31 U 1S   23 F Z 0    Z    Bproc on translator list ?
   DC
   DO
```

```
   DA   0 H L 1
   DI
 0 N 1S   0 F Z 0   Z      BbeginP
 1 N 1B   1         A
 2 N 2T  30 H L 0 A
 3   2S  32764 X 0 B
 4   1S  16 Z E 1   Z
 5 N 2T   1 H L 1 A
 6   6B  13 Z E 2
 7   6T  10 H R 0 0        require function
 8   2S   2 F Z 1
 9 U 2LS 32766 X 0 B Z     mark value assignment in
10 Y 4S  32766 X 0 B Z     name code
11   2A   5         A
12   6T   0 L F 0 5        NNP
13 N 2B  13 Z E 2
14 N 2S  32761 X 0 B
15 N 0LS 30 Y E 4
16 N 6T   0 Z W 0 0        FTL(N)
17   2S  12 Z E 0
18 U 2LS  2 F Z 0   Z      SFLA
19 Y 2A  16 F Z 1
20 Y 0A   5 Z E 2
21 Y 4A   6 Z E 2
22   2B  15 Y E 1 A
23   6T   0 K N 0 0        require
24 N 7Y  1066 C 0
25   6T   0 L Z 0 0        put
26   2LA  8         A Z
27   2S   6 Z E 2
28 Y 0S  1024      A
29   6T   0 Z W 0 0        FTL(macro-order)
30   2T   0 Z Y 0 A
   DC
   DO
```

step

```
   DA   0 H R 0
   DI
 0   2A    0        A
 1   6T    0 L F 0 5          NNP
 2   6T    0 R F 0 6          conditional expressions
 3   2B   19 Y E 3 A
 4   6T    0 K N 0 0          require
 5 N 7Y  1077 C 0
 6   6T    0 L Z 0 0          put
 7   2B    0        A
 8   2S   22        A
 9   2T   13 H E 1 A          FOT(D12) etc.
10   2B   16 Z E 1            insertion in :=
11   2S  32767 X 0 B
12   2LS   0 F Z 0   Z
13 Y 7Y  1066 C 0
14   2T    8   X 0   E
   DC
   DO
```

/ erratum: FOT(FOR4) etc.

until

```
   DA    0 H S 0
   DI
0    2A    0        A
1    6T    0 L F 0 5          NNP
2    6T    0 R F 0 6          conditional expressions
3    2B   23 Y E 3 A
4    6T    0 K N 0 0          require
5 N  7Y   1079 C 0
6    6T    0 L Z 0 0          put
7    2B    0        A
8    2S   23        A
9    2T   13 H E 1 A          FOT(FOR5) etc.
   DC
   DO
```

while

```
   DA    0 H T 0
   DI
 0   2A    0        A
 1   6T    0 L F 0 5
 2   6T    0 R F 0 6
 3   2B   27 Y E 3 A
 4   6T    0 K N 0 0
 5 N 7Y  1080 C 0
 6   6T    0 L Z 0 0
 7   2B    0        A
 8   2S   49        A
 9   2T   13 H E 1 A
10   2B   25 Z E 1
11   0B   24 Z E 1
12   2S   10 Z E 2
13   0LS  14 F Z 1
14   6S  32767 X 0 B
15   1B    1        A
16   4T   14 H T 0 0 P
17   2T    8   X 0   E
   DC
   DO
```

NNP
conditional expressions

require

put

FOT(ST) etc.
insertion in closing
square bracket

"("

```
   DA    0 H W 0
   DI
 0    2S   12 Z E 0
 1 U 2LS 16384     A Z       PROFLA
 2 N 2T   26 H W 0 A         in formal par. part  ?
 3 U 2LS  31       A Z       T
 4 Y 2T    0 R T 0 A         in expression
 5   2A   31       A
 6   6T    0 L F 0 5         NNP
 7   2A    0 Z E 1           name code I
 8 U 2LA  15 F Z 0    Z      require procedure
 9 Y 7Y   1062 C 0
10   2S   12 Z E 0
11   6T    0    W 0 0        FTL(Flag word)
12 U 2LA   2 F Z 0    Z      non-macro order ?
13   2S    1 Z E 1           name code II
14   3P   16    SS
15 Y 0LS  128       A
16 Y 0LS   2 F Z 0
17 Y 6T    0 Z W 0 0         FTL(check number)
18 Y 2S   21 Y E 0
19 Y 6T    0 Z W 0 0         FTL(B(actpar)
20 Y 6T    0 R R 0 1         IPW(insert parameter word)
21 Y 2B   22 Y E 0 A
22 Y 2T    7 R T 0 A         require and put etc.
23   0LS  65       A Z
24 N 7Y   1040 C 0           require one parameter
25   2T    4 R T 0 A
26 U 2LS   2 F Z 0    Z      SFLA
27 Y 2T    0 R S 0 A         indeed in form.p.part
28   2T    3 H W 0 A
   DC
   DO
```

")"

```
   DA    0 H U 0
   DI
 0   2A  12 Z E 0
 1 U 2LA 16384    A Z       PROFLA
 2 N 2T   0 W E 0 A         in formal par. part
 3 U 2LA  2 F Z 1   Z       PAFLA
 4 N 2T   0 W F 0 A         in actual par.part
 5   2A   0       A
 6   6T   0 L F 0 5         NNP
 7   6T   0 R F 0 6         conditional expressions
 8   2A  12 Z E 0
 9   2LA 31       A
10   6A  31   E 0           T'
11   2B  10 Y E 1 A
12   6T   0 K N 0 0         require
13   2A  12 Z E 0
14   2LA 31       A
15 Y 1A   0       A P
16 Y 2B  26 Z E 0
17 Y 2S  32767 X 0 B
18 Y OLS 19 Y E 1   Z
19 Y 2S  32765 X 0 B
20 N 7Y  1064 C 0
21 U 2LS 25 Y E 2   Z
22 Y 2LS 26 Y E 2
23 Y OLS 27 Y E 2   Z
24 Y 2LA  8       A Z       ARFLA
25 Y 7Y  1078 C 0
26   6T   0 R W 0 5         backward scanner V
27   2B  10 Y E 1 A
28   6T   0 L Z 0 0         put
29   2T   0 Z Y 0 A
   DC
   DO
```

opening square bracket

```
   DA   0 H Y 0
   DI
 0   2S  12 Z E 0
 1   2LS 1024     A Z      AFLA
 2 Y 2T  13 H Y 0 A        non-declaration
 3   2B  20 Y E 4 A
 4   6T   0 K N 0 0        require
 5 N 7Y  1086 C 0
 6   2S  12 Z E 0
 7   6T   0   W 0 0        put
 8   2S   1 Y E 1
 9   6T   0 Z W 0 0        FTL(Bboundpair)
10   2B  24 Y E 4 A
11   0B   0     A Z        condition:= -
12   2T   6 T R 0 A        "," in array segment
13   2A  30     A
14   6T   0 L F 0 5        NNP
15   2S   0 Z E 1          name code I
16   2LS  5 F Z 0   Z      non array name ?
17 Y 2S  12 Z E 0
18 Y 6T   0   W 0 0        FTL(Flag word)
19   2B  24 Y E 4 A
20   6T   0 L Z 0 0        put
21   2S   0 Y E 1
22   6T   0 Z W 0 0        FTL(Bind)
23 Y 2T   0   Y 0 A
24   2T   5 T N 0 A        decrease check number
   DC
   DO
```

closing square bracket

```
   DA    0 H N O
   DI
 0   2A    0          A
 1   6T    0 L F 0 5          NNP
 2   6T    0 R F 0 6          conditional expressions
 3   2B    4 Y Z 0 A
 4   6T    0 K N 0 0          require
 5 N 7Y  1088 C 0
 6   6T    0 R W 0 5          backward scanner V
 7   6T   18 H N 1 0          T' := T
 8 U 2LA 1024       A Z       AFLA
 9 N 2T   13 H N 0 A          declaration
10   2B   30 Y E 2 A
11   6T    0 L Z 0 0          put
12   2T    0 Z Y 0 A
13   2S    1          A
14   4S   10    E 2           dimension counter
15 U 2LA 256       A Z       WOFLA
16   6T   23 H N 1 1          own business
17 U 2LA  96       A Z
18   2S   16          A       ROAD
19 N 0S    1          A       IOAD
20   2T   27 H N 0 A
21 U 2LA  32       A Z
22 N 2S   15          A       BAD
23 N 2T   27 H N 0 A
24 U 2LA  64       A Z
25   2S   14          A       IAD
26 Y 1S    1          A       RAD
27   2B    0          A
28   6T    0 U E 0 2          FOT(array declarator)
29   2S    9 Z E 2
30   2B    6          A
31   6T    0 U E 0 2          FOT(N)
   DC
   DO
```

```
   DA    0 H N 1
   DI
 0 U 2LA 256        A Z        WOFLA
 1 N 6T   17 W W 0 0           own location
 2 Y 2S   21 Z E 1
 3   2B    5        A
 4   6T    0 U E 0 2           FOT(LN)
 5   2S   10 Z E 2             dimension counter
 6 N 0S   10   E 2
 7 Y 0S    4        A
 8 N 0S    6        A
 9   6S   11   E 2             2n + 6 or n + 4
10   2X    9   E 2
11 N 4S   23   E 1             ss
12 Y 4S   21   E 1             qq
13   6T    0 W W 0 1           FNC II
14   2A    0        A          restore
15   6A    9 Z E 2             dimension counter and
16   6A   10   E 2             array counter
17   2T   10 H N 0 A
18   2A   12 Z E 0
19   2LA  31        A
20   6A   31   E 0             T' := T
21   2A   12   E 0
22   2T    8   X 0   E
23 Y 2T   21 H N 0 A
24   2S    9 Z E 2             array counter
25   4S   24 Z E 1             ppp
26   6S    0   X 0
27   6T   10 H T 0 0
28   2T    9   X 0   E
   DC
   DO
```

opening string quote

```
   DA   0 K Z 0
   DI
 0   2B  30 Y E 2 A
 1   6T   0 K N 0 0        require
 2 N 7Y  1073 C 0
 3   6T   0 L Z 0 0        put
 4   6T   0   E 0 4        PCP
 5   2S   2 Y E 3
 6   6T   0 Z W 0 0        FTL(Bstrq)
 7   2S   1       A
 8   6S   7   E 2
 9   6T   0   K 0 3        NRS
10   2S   1       A
11 U 1A  102       A Z
12 Y 4S   7   E 2
13 U 1A  103       A Z
14 Y 5S   7   E 2   Z
15   2B   9       A
16 Y 2S  127       A        closing symbol
17 N 4P        AS
18   6T   0 U E 0 2        FOT(string symbol)
19 N 2T   9 K Z 0 A
20   2A   1       A
21   6A  31 Z E 0          T' := 1
22   2T   0 Z Y 0 A
   DC
   DO
```

end

```
   DA    0 K E 0
   DI
 0   6T    0 R L 0 7          end of statement
 1 N 7Y   1051 C 0
 2   2B   26 Z E 0
 3   2A   32767 X 0 B
 4 U OLA   0 F Z 1   Z        BbeginC on translator list ?
 5 Y 2T   13 K E 0 A
 6 U OLA  31 F Z 0   Z        BbeginB ?
 7 Y 2T   11 K E 0 A
 8   OLA   0 F Z 0   Z        BbeginP
 9 N 7Y   1051 C 0
10   2A    1        A Z
11   6T    0 R W 0 5          backward scanner V
12   6T   13   F 0 3          FOT(PRA)
13 Y 6T    0   W 0 5          backward scanner V
14   2T    0 Z Y 0 A
   DC
   DO
```

own

```
   DA    0 K F 0
   DI
0    6T    8 K H 0 0        PROFLA
1  N 7Y  1059 C 0
2    6T   16   H 0 5        try block introduction
3    6T    6   S 0 1        require and put
4    2A  256       A        WOFLA
5    2T    2   U 0 A
   DC
   DO
```

Boolean

try block introduction
require and put

```
    DA   0 K H 0
    DI
 0   6T   3 K H 0 6      combined call
 1   2A  32       A
 2   2T   2   U 0 A
 3   6T   8   H 0 0      PROFLA = 0 ?
 4 N 6T   4   U 0 1      require and put for decl.
 5 Y 6T  16   H 0 5      try block introduction
 6 Y 6T  11   H 0 1      require and put for spec.
 7   2T  14   X 0   E
 8   2A  12 Z E 0
 9   2LA 16384    A Z    PROFLA
10   2T   8   X 0
11   2B   5 Y E 0 A
12   6T   0 K N 0 0      require
13 N 7Y  1058 C 0
14   6T   0 L Z 0 0      put
15   2T   9   X 0   E
16   2B  26 Z E 0
17   2A   0 F Z 1
18 U 1A  32767 X 0 B Z   BbeginC on translator list
19 Y 2S   2       A
20 Y 5S  26 Z E 0
21 Y 6T   0 S E 0 4      INB I
22   2T  13   X 0   E
   DC
   DO
```

erratum: interchange comment of lines 4 & 6.

integer

```
   DA   0 K K 0
   DI
0    6T   3 K H 0 6      see Boolean
1    2A  64       A      INFLA
2    2T   2   U 0 A
   DC
   DO
```

real

```
   DA    0 K L 0
   DI
0    6T    3 K H 0 6        see Boolean
1    2A  128        A       REFLA
2    2T    2    U 0 A
   DC
   DO
```

array

```
   DA    0 K R 0
   DI
 0   6T    8 K H 0 0          PROFLA
 1 N 6T   13   R 0 1          require and put
 2 Y 6T   16   H 0 5          try block introduction
 3 Y 6T    8   R 0 1          require and put
 4   2A    0       A
 5   6A    9 Z E 2            array counter
 6   2A 1024       A          AFLA
 7   2T    2 K U 0 A
 8   2B    9 Y E 0 A
 9   6T    0 K N 0 0          require
10 N 7Y 1061 C 0
11   6T    0 L Z 0 0          put
12   2T    9   X 0   E
13   2B   17 Y E 0 A
14   2T    9 K R 0 A
   DC
   DO
```

switch

```
   DA    0 K S 0
   DI
 0   6T    8 K H 0 0      PROFLA
 1 N 6T    4   U 0 1      require and put for spec.
 2 Y 6T   16   H 0 5      try block introduction
 3 Y 6T    6   S 0 1      require and put for decl.
 4   2A  4096      A      SIFLA
 5   2T    2   U 0 A
 6   2B   13 Y E 0 A
 7   6T    0 K N 0 0      require
 8 N 7Y  1060 C 0
 9   6T    0 L Z 0 0      put
10   2T    9   X 0   E
   DC
   DO
```

procedure

```
   DA    0 K T 0
   DI
0    6T    8 K H 0 0     PROFLA
1 Y 6T   16   H 0 5     try block introduction
2   6T   13   R 0 1     require and put
3 N 2A  512       A     POFLA
4 Y 2A  16384     A     PROFLA
5   2T    2   U 0 A
   DC
   DO
```

string

```
  DA    0 K W 0
  DI
0   6T    4 K U 0 1     require and put
1   2A  8192      A     STRIFLA
2   2T    2 K U 0 A
  DC
  DO
```

label

```
   DA    0 K U 0
   DI
0    6T    4 K U 0 1      require and put
1    2A  2048      A      LEFLA
2    4A  12 Z E 0
3    2T    0   Y 0 A
4    2B    0 Y E 0 A
5    6T    0 K N 0 0      require
6 N  7Y  1057 C 0
7    6T    0 L Z 0 0      put
8    2T    9   X 0   E
   DC
   DO
```

value

```
  DA    0 K Y 0
  DI
0   6T    4 K U 0 1      require and put
1   2A    4 Y E 0        VAFLA
2   2T    2 K U 0 A
  DC
  DO
```

require

```
   DA    0 K N 0
   DI
0    2A   12 Z E 0
1    2LA   0   X 0 B
2    0LA   1   X 0 B Z
3    2T    8   X 0
   DC
   DO
```

put

```
   DA    0 L Z 0
   DI
0    2A   12 Z E 0
1    3LA   2   X 0 B
2    0LA   3   X 0 B
3    6A   12   E 0
4    2T    8   X 0   E
   DC
   DO
```

PCP(prepare complicated parameter)

```
   DA   0 L E 0
   DI
 0   2A  12 Z E 0
 1 U 2LA  2 F Z 1   Z    PAFLA
 2 Y 2T  21 L E 0 A      no parameter
 3   2B  26 Z E 0
 4   2S  32767 X 0 B
 5   0LS 21 Y E 0   Z
 6 N 2T  21 L E 0 A      no begin of parameter
 7   2LA 15 F Z 0   Z
 8 N 5A  12 Z E 0        implicit jump over
 9 N 6T   0 L W 0 3      implicit subroutine
10   2A   1       A
11   0A  11 Z E 0
12 U 3LA 31        A Z
13 Y 6A  11   E 0        BN
14 N 7Y  1053 C 0
15   2B  26 Z E 0
16   2S  13 U Z 0
17   2P  15   SS
18   0LS 12   Z 0
19   0LS 16 F Z 0   P    store PORD-tail in
20   6S  32761 X 0 B     translator list
21   2A   0 Z E 1
22   2T  12   X 0
   DC
   DO
```

NNP(name and number processor)

```
   DA    0 L F 0
   DI
 0   6A   10 Z E 0          priority number
 1   2A   12   E 0          NFLA
 2 U 2LA   2        A Z     Name processor
 3 N 2T    0 L H 0 A
 4 U 2LA   1        A Z     KFLA
 5 N 2T    0 L K 0 A        number processor
 6   2A   12 Z E 0
 7   2LA 2051      A
 8   0LA 2048      A Z      LEFLA
 9 Y 2A   31   E 0          T'
10 Y 4A   12   E 0          T
11   2B    1        A       from name processor or
12   6B   13   E 1          number processor
13   2B    1        A       from backward scanner
14   5B   26   E 0
15   2B   26   E 0
16   2S    0   X 0 B P
17   2A    0        A
18 Y 2P    5   AS
19 N 6S   12   E 0
20 N 0LS   0   D15
21 N 2T   13   X 0   E      exit backward scanner V
22   4P        AB
23   4P        SA
24   3P    5   SS    Z
25   2T    0 L L 0 B        distributive jump
   DC
   DO
```

name processor

```
   DA    0 L H 0
   DI
 0   2A    0 Z E 1          name code I
 1 U 2LA  17 F Z 0   Z
 2   2S    1 Z E 1          name code II
 3 Y 2T    6 L H 0 A
 4 U 2LS   0 F Z 0   Z
 5 Y 7Y  1045 C 0
 6 U 2LA   9 F Z 0   Z
 7 N 7Y  1046 C 0
 8 U 2LA  18 F Z 0   Z
 9 N 2T    9 R E 0 A        switch identifier
10 U 2LA   2 F Z 1   Z
11 N 2T    0 R E 0 A        label identifier
12 U 2LA   5 F Z 0   Z
13 N 2T    0 L Y 0 A        array identifier
14 U 2LA  15 F Z 0   Z
15 N 2T    0 L U 0 A        procedure identifier
16   6T    0   E 0 4        PCP
17   2LA  17 F Z 0   Z
18   2A    8 Z E 0          delimiter
19 Y 2T   24 L H 0 A        non-formal
20   0LA  92       A Z      := ?
21   2S   42       A        TFA
22 N 1S    1       A        TFR
23   2T    1 L H 1 A
24   2S   27       A        TRA
25   0LA  92       A Z
26 N 0S    4       A        TRR
27   2A    0 Z E 1
28   2LA  19 F Z 0   Z
29 Y 0S    1       A        TIA or TIR
30   2A    1 Z E 1
31   2LA   9 F Z 0   Z
   DC
   DO
```

```
   DA    0 L H 1
   DI
 0 N OS    2         A P      own version
 1   2B    0         A
 2   6T    0 U E 0 2          FOT( take order)
 3   2B    1 Z E 1
 4   4P        BS
 5   2B    1         A
 6   6T    0 U E 0 2          FOT(DA)
 7 N 2B    0         A
 8 N 2S    6         A
 9 N 6T    0 U E 0 2          FOT(TNR)
10   2T   11 L F 0 A
   DC
   DO
```

number processor

```
   DA    0 L K 0
   DI
 0   6T    0 L E 0 4        PCP
 1   2A   12 Z E 0
 2 U 2LA   4        A Z
 3   2S   25        A        TCR
 4 N 2S   26        A        TCI
 5   2B    0        A
 6   6T    0 U E 0 2        FOT(take constant)
 7 Y 2S   15 Z E 0
 8 N 2S   16 Z E 0
 9   2B    8        A
10   6T    0 U E 0 2        FOT(integer or head of
11 N 2T   11 L F 0 A                mantissa)
12   2A    8 Z E 1   Z
13 N 2S   16 Z E 0
14 N 6T    0 U E 0 2        FOT(tail of mantissa)
15   2S   17 Z E 0
16   2B    7        A
17   6T    0 U E 0 2        FOT(exponent)
18   2T   11 L F 0 A        NNP
   DC
   DO
```

backward scanner I
distribution list

```
   DA    0 L L 0
   DI
 0   OA    0   T 0        macro-order
 1   OA    4   R 0        DA
 2   OA    4   R 0        NAP(number in address pile)
 3   OA    4   R 0        PDA
 4   OA    4   R 0        PPA
 5   OA    4   R 0        LN
 6   OA    4   R 0        N
 7   OA    4   R 0        head of PORD
 8   OA    5   R 1        bracket
 9   OA    0   R 0        NLSC(name list counter)
10   OA   21   R 0        APC(address pile counter)
11   OA    4   R 0        PRA( for address pile)
12   OA    2   R 0        PORD-tail = PRA
13   OA    8   R 0        PORD-tail = DA
14   OA    6   R 0        PORD-tail = DA + N
15   OA    0   S 0        PORD-tail = NAP+ N
16   OA    0   R 1        check number
17   OA    8   R 1        BN
18   OA   10   R 1        qq= LN non-own var.& arr.
19   OA   12   R 1        rr= LN own variables
20   OA   14   R 1        ss= LN own arrays
21   OA   16   R 1        ppp=LN in prestack list
   DC
   DO
```

backward scanner II

```
   DA    0 L R 0
   DI
 0   6S   19 Z E 0          NLSC
 1   2T   13 L F 0 A
 2   6T    0 U E 0 2        FOT(D12) S unchanged
 3   2B   11       A
 4   6T    0 U E 0 2        FOT(PRA)
 5   2T   13 L F 0 A
 6   3P   15   SS     P     DA + N
 7   6S   14 Z E 1
 8   3P   15   AS           DA
 9   3P   16   SS
10   2LA  31       A
11   6S    2   E 1
12   4P        AS
13   2B    6       A
14   6T    0 U E 0 2        FOT(BN)
15   2B    1       A
16   2S    2 Z E 1
17 Y 6T    0 U E 0 2        FOT(LN)
18 Y 2S   14 Z E 1
19 Y 2B    6       A
20   2T    4 L R 0 A
21   6S   14 Z E 1          punching address pile
22   2B   12   E 1
23 U 1B   14   E 1     P
24 Y 1B    1       A
25 Y 0T    0       A
26 Y 0B   27   E 0           BAP
27 Y 2S    0   X 0 B
28 Y 2B   11       A
29 Y 6T    0 U E 0 2        FOT(PRA)
30 Y 2T   22 L R 0 A
31   2T   13   F 0 A
   DC
   DO
```

```
   DA    0 L R 1
   DI
 0 U 2LS 15        A Z
 1 Y 2T  18 L R 1  A
 2 U 2LS 272       A Z
 3 Y 7Y  1040 C 0
 4   2T  18 L R 1  A
 5   2B   1        A
 6   4B  26 Z E 0
 7   2T  13   X 0   E        exit backward scanner
 8   6S  11   E 0
 9   2T  13 L F 0 A
10   6S  21 Z E 1
11   2T  13 L F 0 A
12   6S  22 Z E 1
13   2T  13 L F 0 A
14   6S  23 Z E 1
15   2T  13 L F 0 A
16   6S  24 Z E 1
17   2T  13 L F 0 A
18   2B  26 Z E 0
19   2A  32767 X 0 B          check number transformed
20 U 0LA 31 L R 1   Z         into metaparameter N
21 Y 0B   1        A          of FTAA
22 Y 6B  26 Z E 0
23 Y 6A  32767 X 0 B
24 Y 2A  32765 X 0 B
25 Y 6A  32766 X 0 B
26 Y 5P       SS
27 Y 2LS 127      A
28 Y 0S   2 Y E 5
29 Y 6S  32765 X 0 B
30   2T  13 L F 0 A
   DU
31 +11101000000000101011      FTAA
   DC
   DO
```

backward scanner III

```
  DA   0 L S 0
  DI
0   6A  14 Z E 1
1   2LS 32767    A
2   2B   2       A
3   6T   0 U E 0 2        FOT(NAP)
4   2S  14 Z E 1   P
5   3P  20   SS
6   2T  19 L R 0 A
  DC
  DO
```

backward scanner IV
produce and check macro-order

```
   DA    0 L T 0
   DI
 0 U 2A   13 Z E 1    Z     no priority test ?
 1 N 3P   20   AA
 2 N 1A   10 Z E 0    P     rank - priority at least
 3 N 4P       AA      Z     zero?
 4 N 2T    5 L R 1 A
 5   6S   14 Z E 1
 6   2LS 127       A
 7   6T    0 U E 0 2        FOT(macro-order)
 8   2S   14 Z E 1
 9 U 2LS 1024      A Z     Boolean bit absent ?
10   2LS 127       A
11   2A   12   E 0
12 N 2T   16 L T 0 A
13 U 1S   71      A P
14 U 1S   66      A E
15 Y 2T   19   T 0 A
16 U 2LA 24       A Z     Boolean test
17 N 7Y  1043 C 0
18   2T   13   F 0 A
19 U OLS  1       A Z     JU ?
20 N 2T   24   T 0 A
21 U 2LA 16       A Z     designational test
22 Y 7Y  1042 C 0
23   2T   13   F 0 A
24 U OLS 72       A Z     NEG ?
25 Y 2T   29   T 0 A
26 U 1S   66      A P
27 U 1S   54      A E
28 Y 2T   13   F 0 A
29 U 2LA  8       A Z     arithmetical test
30 Y 7Y  1044 C 0
31 U 1S   66      A P
   DC
   DO
```

```
  DA   0 L T 1
  DI
0 U 1S  60       A E
1 N OLA  8       A         ARFLA
2 N 6A  12 Z E 0
3   2T  13 L F 0 A
  DC
  DO
```

implicit jump

```
   DA    0 L W 0
   DI
0    2B    0        A
1    2S    0        A
2    6T    0 U E 0 2          FOT(IJU)
3    2B   12        A
4    6T    0   E 0 2          FOT(D12)
5    2T   11   X 0   E
   DC
   DO
```

procedure identifiers

```
   DA    0 L U 0
   DI
 0   6T    0 L E 0 4          PCP
 1   2S    8 Z E 0            delimiter
 2 U OLS  92        A Z       := ?
 3 Y 2T   11 L F 0 A
 4 U OLS  98        A Z       "(" ?
 5 Y 2T   10 L U 0 A
 6   2S    1 Z E 1            name code II
 7   3P   16   SS
 8   2LS  15        A Z
 9 N 7Y   1047 C 0
10   2A   12 Z E 0
11 U 2LA  31 Y E 4   Z        SFLA & PAFLA
12 Y 2S    0 Z E 1
13 Y 2LS   0 F Z 0   Z
14 Y 7Y   1089 C 0
15   2S    0 Z E 1
16 U 2LS   2 F Z 0   Z        no macro-order ?
17 Y 2T   30 L U 0 A
18 U 2LA   2 F Z 0   Z        SFLA
19 N 2LS   0 F Z 0   Z
20 N 2S    7        A
21 N OLS  20 F Z 0
22 N 6T    0 Z W 0 0          FTL(RR)
23   2S    1 Z E 1
24   3P   10   SS
25   2LS  31        A
26   OS   75        A
27   OLS  21 F Z 0
28   6T    0 Z W 0 0          FTL(macro-order)
29   2T   11 L F 0 A
30   2LA   2 F Z 0   Z        SFLA
31 N 2A    1        A
   DC
   DO
```

```
   DA   0 L U 1
   DI
 0   6A  14 Z E 1
 1 N 2S   7       A
 2 N 0LS 20 F Z 0
 3 N 6T   0 Z W 0 0      FTL(RR)
 4   2A   0 Z E 1
 5 U 2LA 17 F Z 0   Z
 6 N 2S   6       A
 7 N 0LS 20 F Z 0
 8 N 6T   0 Z W 0 0      FTL(TNR)
 9   2B   1 Z E 1
10   4P       BS
11 Y 0LS  2 F Z 1
12 N 0LS 15 F Z 0
13 N 2B  14 Z E 1
14 N 5B  26   E 0
15   6T   0   W 0 0      FTL(NAP or DA)
16   2S  22 F Z 0
17   6T   0 Z W 0 0      FTL( N = 0 )
18   2S  48       A      PC
19 N 1S   2       A      FPCR
20 N 0S  14   E 1        FPCP
21   0LS 20 F Z 0
22   2T  28 L U 0 A
   DC
   DO
```

array identifiers

```
   DA   0 L Y 0
   DI
 0   6T   0   E 0 4      PCP
 1   2B  26 Z E 0
 2   2S  32767 X 0 B
 3   1S  21 Y E 0   Z    B(actpar on translator list ?
 4 Y 2S  18 F Z 0
 5 Y 4S  12 Z E 0        PIFLA
 6   2S  14 Y E 1        ITAR
 7   6T   0 Z W 0 0      FTL(ITAR)
 8   2S  12   E 0
 9   6T   0   W 0 0      FTL(Flag word)
10   2A   1   E 1
11   4P       AB
12   4P       BS
13   0LS 15 F Z 0
14   6T   0 Z W 0 0      FTL(DA)
15 U 2LA  9 F Z 0   Z
16 Y 2S  35         A    TAA
17 N 2S  36         A    OTAA
18   2LA  2   Z 0   Z
19 N 2S  37         A    VTAA
20 Y 3A   0 Z E 1
21 Y 2LA 17 F Z 0   Z
22 Y 2S  43         A    FTAA
23   0LS 20   Z 0
24   6T   0 Z W 0 0      FTL(macro-order)
25   2S   1   E 1
26   3P  16   SS
27   2LS 287      A      isolate dimensions
28   0LS 128      A      marker array check
29   0LS  2 F Z 0
30   6T   0 Z W 0 0      FTL(check number)
31   2T  13   X 0   E
   DC
   DO
```

fixed contents of name list

	DA 0 L N O DN		NAP
0	+59086335	abs	0
1	+4265984	name code	II
2	+52756483	name code	I
3	+48768231	sign	1
4	+4267009		
5	+52690947		
6	+61715687	sqrt	2
7	+4274178		
8	+52756483		
9	+48839167	sin	3
10	+4271107		
11	+52756483		
12	+59513343	cos	4
13	+4272132		
14	+52756483		
15	+48955391	ln	5
16	+4273157		
17	+52756483		
18	+53517823	exp	6
19	+4270086		
20	+52756483		
21	+38710903	entier	7
22	+57114623		
23	+4268039		
24	+52690948		
25	+27598047	read	8
26	+4203528		
27	+52690947		
28	+48838351		
29	+62914559	print	9
30	+65545		
31	+2097156		
	DC		
	DO		

```
   DA   0 L N 1
   DN                                    NAP
 0 -53284352          TAB              10
 1 +1034
 2 +35651587
 3 -19668584          NLCR             11
 4 +2059
 5 +35651587
 6 -27995160          XEEN             12
 7 +4264972
 8 +52690947
 9 -51189312          SPACE            13
10 -46137344
11 +65549
12 +2097156
13 +53230311          stop             14
14 +5010
15 +35651587
16 -29560832          SUM              15
17 +4456463
18 +19136515
19 -27857496          PRINTTEXT        16
20 -7015992
21 -14680064
22 +65552
23 +2097157
24 -27986608          EVEN             17
25 +4259857
26 +19136515
27 +61224535          arctan           18-
28 +48594943
29 +4259858
30 +19202052
31 -15081128          FLOT             19
   DC
   DO
```

```
   DA   0 L N 2
   DN                                    NAP
 0 +131091
 1 +2097155
 2 -14787752          FIXT             20
 3 +196628
 4 +2097155
 5 +28021903          hand             21
 6 +4259840
 7 +19202051
 8 -46369880          PUTEXT           22
 9 -14778368
10 +65558
11 +2097156
12 -23333976          PUSPACE          23
13 -46937088
14 +65559
15 +2097156
16 -31886424          PUNLCR           24
17 -19660800
18 +24
19 +2097156
20 -25594952          RUNOUT             25
21 -14876672
22 +25
23 +2097156
24 -46511160          TAPEND           26
25 -48660480
26 +26
27 +2097156
28 -23465536          STOPCODE         27
29 -46897344
30 +27
31 +2097156
   DC
   DO
```

```
   DA    0 L N 3
   DU                                                    NAP
 0 +1101001100111100001010101 11      28      FLOP
 1 +1100000000000111 00
 2 +10000000000000000000011
 3 +1101001111001011011010101 11
 4 +1100000000000111 01                                29      FIXP
 5 +100000000000000000001 1
 6 +1101101101001100101011011 11      30      INPROD
 7 +1010001100111111111111111 1111
 8 +100010100000000000111 10
 9 +1001001000000000000000100
10 +1010001100101001011101101 11      31      RANDOM
11 +1100011100111111111111111 1111
12 +100000000000000000111 11
13 +1001001010000000000000100
14 +1101101110001010011101111 11      32      SETRANDOM
15 +1100111010001100101001011 11
16 +1100011111111111111111111 1111
17 +100000000001 00000
18 +10000000000000000000101
19 +1010101101111001101001011 11      33      ABSFIXT
20 +1110001110010110111111111
21 +1100000000001 00001
22 +10000000000000000001 00
23 +1010101101111001101001011 11      34      ABSFIXP
24 +1101001110010110111111111
25 +1100000000001 00010
26 +10000000000000000001 00
27 +1110001001110010110101011 11      35      FACTOR
28 +1101101100111111111111111
29 +1000001000000000010001 1
30 +1001001000000000000000100
31 +1001011100011010011101101 11      36      REMAINDER
   DC
   DO
```

```
   DA   0 L N 4                                  NAP
   DU
 0 +1 01 001 1 01 0001 1 001 01 01 101 111
 1 +1 101 1 01 1111111111111111111111
 2 +1 00001 000000000001 001 00
 3 +1 001 001 0000000000000000101
 4 +1 01 0001 001 111 01 01 1111111111      37    GCD
 5 +1 00001 000000000001 001 01
 6 +1 001 001 0000000000000000011
 7 +1 101 001 01 0001 1 1 001 1 011111      38    readn
 8 +1 011111111111111111111111111
 9 +1 0000000000000001 001 1 0
10 +1 00000000000000000001 00
11 +1 001 1 00001 1111 1 001 1 101 00111    39    PU7BIT
12 +111 0001 011 01 1111111111111111
13 +1 000000000001 001 11
14 +1 00000000000000000001 00
15 +1 001 1 00001 1111 01 001 1 101 10111   40    RE7BIT
16 +111 0001 011 01 1111111111111111
17 +1 0000000000000001 01 000
18 +1 001 001 0000000000000000100
19 +1 011 101 001 001 1 01 1 01 1 001 111   41    printn
20 +1 011 101 1 101 1111111111111111
21 +1 0000000000000001 01 001              name code II
22 +1 000000000000000000100                name code I
   DC
   DO
```

fixed contents of address pile

	DA	0	R	Z	0			paragraph letters
	DI							
0		OA	1	E	S	2	abs	LZ
1		OA	2		S	2	sign	LE
2		OA	3		S	2	sqrt	LF
3		OA	4		S	2	sin	LH
4		OA	5		S	2	cos	LK
5		OA	6		S	2	ln	LL
6		OA	7		S	2	exp	LR
7		OA	8		S	2	entier	LS
8		OA	9		S	2	read	LT
9		OA	10		S	2	print	LW
10		OA	11		S	2	TAB	LU
11		OA	12		S	2	NLCR	LY
12		OA	13		S	2	XEEN	LN
13		OA	14		S	2	SPACE	RZ
14		OA	18		S	2	stop	RK
15		OA	21		S	2	SUM	RS
16		OA	20		S	2	PRINTTEXT	RR
17		OA	22		S	2	EVEN	RT
18		OA	19		S	2	arctan	RL
19		OA	16		S	2	FLOT	RF
20		OA	15		S	2	FIXT	RE
21		OA	17		S	2	hand	RH
22		OA	23		S	2	PUTEXT	RW
23		OA	24		S	2	PUSPACE	RU
24		OA	25		S	2	PUNLCR	RY
25		OA	26		S	2	RUNOUT	RN
26		OA	27		S	2	TAPEND	SZ
27		OA	28		S	2	STOPCODE	SE
28		OA	29		S	2	FLOP	SF
29		OA	30		S	2	FIXP	SH
30		OA	31		S	2	INPROD	SK
31		OA	0		S	3	RANDOM	SL
	DC							
	DO							

```
   DA    0 R Z 1                          paragraph
   DI                                     letters
0    OA    1 E S 3        SETRANDOM       SR
1    OA    2   S 3        ABSFIXT         SS
2    OA    3   S 3        ABSFIXP         ST
3    OA    4   S 3        FACTOR          SW
4    OA    5   S 3        REMAINDER       SU
5    OA    6   S 3        GCD             SY
6    OA    7   S 3        readn           SN
7    OA    8   S 3        PU7BIT          TZ
8    OA    9   S 3        RE7BIT          TE
9    OA   10   S 3        printn          TF
   DC
   DO
```

labels and switches

```
   DA    0 R E 0
   DI
 0   2S  39        A          TAL
 1   6T  20 R E 0 0
 2   2B   0        A
 3   6T   0 U E 0 2           FOT(TAL or TFL)
 4   2B   1 Z E 1
 5   4P      BS
 6   2B   1        A
 7   6T   0 U E 0 2           FOT(DA)
 8   2T  11 L F 0 A
 9   6T   0 L E 0 4           PCP
10   6T   0 W L 0 4           PSE
11   2B   1 Z E 1
12   4P      BS
13   0LS 15 F Z 0
14   6T   0 Z W 0 0           FTL(DA)
15   2S  40        A
16   6T  20 R E 0 0
17   0S  19 F Z 0
18   6T   0 Z W 0 0           FTL(SWIND or FSWIND)
19   2T  11 L F 0 A
20   2A   0 Z E 1
21   2LA 17 F Z 0   Z
22 N 0S   5        A
23   2T   8   X 0   E
   DC
   DO
```

conditional expressions

```
   DA   0 R F 0
   DI
 0   2B  26 Z E 0
 1   2S  32767 X 0 B
 2   1S  24 F Z 0   Z      BelseE on translator list ?
 3 N 2T  14   X 0   E
 4   6T  19 R F 0 4
 5   2S  12 Z E 0          T to the right of else
 6   6S  31   E 0
 7   6T   0 R W 0 5        backward scanner V
 8   2S  12 Z E 0          T to the left of  else
 9   0LS 31   E 0
10   2LS 24       A Z      T's compatible ?
11 N 7Y  1050 C 0
12   2T   0 R F 0 A
13   2S  13 U Z 0
14   2P  15   SS
15   0LS 12   Z 0
16   2B  11       A
17   6T   0 U E 0 2        FOT(PRA)
18   2T  11   X 0   E
19   2S   8 Z E 0
20 U 1S  75       A P      relational ?
21 U 1S  70       A E
22 Y 2T  26 R F 0 A
23   2S  32765 X 0 B
24   2LS 24       A Z      T to the left Boolean ?
25 Y 2T  14   X 0   E      conditional expr.unfinished
26   6T  13 R F 0 3
27   2T  12   X 0   E
   DC
   DO
```

for statements

```
   DA   0 R H 0
   DI
 0   2A   0        A
 1   6A  10 Z E 0          priority number
 2   6T  11 L F 0 5        backward scanner
 3   2B  26 Z E 0
 4   2S  32767 X 0 B
 5   1S  25 F Z 0   Z      Bdo on translator list
 6 N 2T  14   X 0   E
 7   6T   0 L W 0 3        implicit jump
 8   2B  26 Z E 0
 9   2S  13 U Z 0
10   2P  15   SS
11   OLS 12   Z 0
12   OLS 26 F Z 0          this moment's PRA
13   6S  32764 X 0 B
14   6T   0 R W 0 5        backward scanner V
15   2T   3   H 0 A
   DC
   DO
```

conditional statements

```
   DA    0 R K 0
   DI
0    2B   26 Z E 0
1    2S   32767 X 0 B
2 U  1S   27 F Z 0   Z      BelseS  or
3 N  1S   28 F Z 0   Z      BthenS on translator list ?
4 N  2T   14   X 0   E
5    6T   13 R F 0 3        produce PRA
6    6T    0 R W 0 5        backward scanner V
7    2T    0 R K 0 A
   DC
   DO
```

end of statement

```
   DA    0 R L 0
   DI
 0   2A    0        A
 1   6T    0 L F 0 5        NNP
 2   6T    0 R F 0 6        conditional expressions
 3   2B   29 F Z 0 A
 4   6T    0 K N 0 0        require
 5 Y 6T   11 R L 0 0        require statement processed
 6   6T    0 R H 0 6        for statements
 7   6T    0   K 0 6        conditional statements
 8   3S   15 F Z 1
 9   6S   12 Z E 0          clear Flag word
10   2T   15   X 0
11   2A   12 Z E 0
12 U 2LA   2 F Z 0   Z
13 N 2LA   3       A Z
14 N 3A    0 Z E 1
15 N 2LA  15 F Z 0   Z
16   2T    8   X 0
   DC
   DO
```

IPW(insert parameter word (= PORD))

```
   DA   0 R R 0
   DI
 0   2B  26 Z E 0
 1   2S   5       A
 2   1B   1       A
 3   2A   2       A
 4   6T   0 S R 0 0        TR
 5   0B   8       A
 6   6B  26 Z E 0
 7   2S  32766 X 0 B
 8   3A   1       A
 9   5A  32763 X 0 B
10   0A  32766 X 0 B
11   3LS 15       A        decrease check number
12   2LA 15       A        modulo 16
13   6S  32766 X 0 B
14   4A  32766 X 0 B
15   2T   9   X 0   E
   DC
   DO
```

"(" of formal parameter part

```
      DA    0 R S 0
      DI
   0    2B   27 Y E 1 A
   1    6T    0 K N 0 0      require
   2 N  7Y  1062 C 0
   3    2S    1 Y E 0        SFLA & PROFLA
   4    6S   12 Z E 0
   5    6T    0 S E 0 4      INB I
   6    3S   29 Y E 1        PROFLA,COFLA & NIFLA
   7    6S   12 Z E 0
   8    6T    0   W 0 0      FTL(Flag word)
   9    2S   30 Y E 1
  10    6T    0 Z W 0 0      FTL(Bforpar)
  11    2T    0   Y 0 A
      DC
      DO
```

"(" in expressions

```
   DA    0 R T O
   DI
 0   6T    0 W L 0 4      PSE
 1   6T    0 L E 0 4      PCP
 2   2S   12 Z E 0
 3   6T    0 Z W 0 0      FTL(Flag word)
 4   2S   19 Y E 1
 5   6T    0 Z W 0 0      FTL(B(E)
 6   2B   20 Y E 1 A
 7   6T    0 K N 0 0      require
 8 N 7Y  1062 C 0
 9   6T    0 L Z 0 0      put
10   2T    0 Z Y 0 A
   DC
   DO
```

backward scanner V

```
   DA   0 R W 0
   DI
 0   2A   0        A
 1   6A  13 Z E 1
 2   2B  26   E 0
 3   2S  32767 X 0 B
 4   1B   1        A
 5 U 1S  23 F Z 0   Z   Bproc or
 6 N 1S   0   Z 0   Z   BbeginP on translator list ?
 7 Y 1B   3        A
 8   6B  26 Z E 0       TLSC
 9 Y 3S   2   X 0 B     name code I
10 Y 2LS  0 F Z 0   Z   function ?
11 N 2T  13 L F 0 A
12   2B   0   X 0 B     require assignment to
13   2S  32766 X 0 B    procedure identifier
14   2LS  2 F Z 1   Z
15 Y 7Y  1049 C 0
16   2T  13 L F 0 A
   DC
   DO
```

punch tape feed

```
   DA   0 R U 0
   DI
0    2B  100       A
1    6B   0   X 0
2    2S   0       A
3    6T   0 U K 0 0       AOB
4    4T   2 R U 0 0 P
5    2T   9   X 0   E
   DC
   DO
```

autostarts

```
   DA  23    X 2
   DI
23   2T    0 R Y 0 A
   DA    0   Y 0
   DI
 0    4P         AA     P
 1 Y  5P         SS     P
 2 N  7Y  1052   C 0
 3 U  1A  14          A Z
 4 Y  3S  12          A
 5 Y  7S  26     X 0
 6 Y  2T   1     D 3 A                  G
 7 U  1A   9          A P
 8 Y  7Y  1052   C 0
 9    4P         AB
10    3S   0          A
11    6S  27     X 0
12    2T  13     Y 0 B P                distributive
13    3LS  0 Z R 0 C                    jump
14    3LS  0 S Z 0 C
15    0B  1052   C 0
16    0B  1052   C 0
17    0B  1052   C 0
18    0B  1052   C 0
19    0B  1052   C 0
20    3LS  0 R N 0 C
21    0B  1052   C 0
22    0B  1052   C 0
   DC
   DO
```

type stop number and line number

```
   DA   0 R N 0
   DI
 0   6T  10   D28 0     NLCR
 1   2B  23   X 0
 2   2S  32767 X 0 B
 3   3LS 13   N 0
 4   1S  14   N 0   Z
 5 N 4P       BS
 6 Y 2S  32767 X 0 B
 7   2LS 16383    A
 8 U 1S  199      A P
 9   6T   0   D22 0     stop number
   DT
10 A G4 NL4 XN
   DI
11 Y 6T  15 R N 0 1     line number
12   2T  19   D 7 A
13 N 0A  16383 X 0
14   7Y   0   C 0
15   2S   2 Z E 0
16   6T   0   D22 0
   DT
17 A G8 -L8 XN
   DI
18   2T   9   X 0   E
   DC
   DO
```

start translation

```
   DA    0 S Z 0
   DI
 0   2B    1         A
 1   7B   11 Z E 0          BN
 2   2A    0         A
 3   6A    0 U Z 0          state of output routines
 4   6A   13   Z 0                              (0=initial)
 5   7A   14   Z 0          controlling directive
 6   2S   27   D16
 7   0S  169         A
 8   6S   12   Z 0          PRA
 9   6A    9   Z 0          bit number partial pentad
10   6A    8   Z 0          partial pentad
11   6A   10   Z 0          parity pentad
12   6A   11   Z 0          pentad counter
13   6A    0 Z E 0          initial state of RUND
14   6A    2   E 0          LC
15   6A   12   E 1          APC
16   6A   31   E 0          T'
17   6B   24   E 1          ppp
18   6B   26   E 1          PPP
19   6A   21   E 1          qq
20   6A   22   E 1          rr
21   6A   23   E 1          ss
22   2B    0 E F 0 A
23   6B   25 Z E 1          PSLB
24   2S   14 F Z 1
25   6S    0   X 0 B
26   3S   15 F Z 1
27   6S   12 Z E 0          Flag word
28   2S    0 E K 0 A
29   6S   19 Z E 0          NLSC
30   6S   22   E 0          INLSC
31   0B    6   E 1          LPSL
   DC
   DO
```

```
   DA    0 S Z 1
   DI
 0   6B   26 Z E 0        TLSC
 1   6B   29   E 0        ITLSC
 2   2A    0 E H 0 A
 3   6A   27 Z E 0        BAP
 4   2S    0 E L 0 A
 5   2B    0 R Z 0 A
 6   1A    0   Z 0 A
 7   0B    1-E L 0 A
 8   6T    0 S R 0 0      TR(fixed contents of address
 9   2S    5 Z E 1                              pile)
10   7S    5   E 1        sum check
11   2A    3   D 0   P
12 Y 6T    0 R U 0 1      punch tape feed
13   6T    1 S E 0 4      INB I
14   2A    0       A
15   6A    9 Z E 2        initial states
16   6A   10   E 2        of prescan program
17   6A   11   E 2
18   6A   12   E 2
19   2T    0   Y 0 A
   DC
   DO
```

INB(introduction new block) I

```
   DA    0 S E 0
   DI
 0   6T    0 L W 0 3     implicit jump
 1   2S   12 Z E 0
 2 U 2LS 16384     A Z   PROFLA
 3 N 3S   15 F Z 1
 4   6T    0 Z W 0 0     FTL(Flag word)
 5   2S   19 Z E 0
 6   0LS   5 F Z 1
 7   6T    0 Z W 0 0     FTL(NLSC)
 8   2S   12 Z E 1
 9   0LS   6 F Z 1
10   6T    0 Z W 0 0     FTL(APC)
11   2S   21 Z E 1
12   0LS   7 F Z 1
13   6T    0 Z W 0 0     FTL(qq)
14   2S   22 Z E 1
15   0LS   8 F Z 1
16   6T    0 Z W 0 0     FTL(rr)
17   2S   23 Z E 1
18   0LS   9 F Z 1
19   6T    0 Z W 0 0     FTL(ss)
20   2S   11 Z E 0   P
21   0LS  10 F Z 1
22 Y 6T    0 Z W 0 0     FTL(BN)
23   2S    1         A
24   4S   11 Z E 0   Z
25   2B   10         A
26 Y 2S    0 E L 0 A
27 N 2S   12 Z E 1
28 Y 6S   12   E 1
29   2A   11   E 0
30 Y 7A   11   E 0
31   6T    0 U E 0 2     FOT(APC)
   DC
   DO
```

```
   DA    0 S E 1
   DI
 0 Y 2S   31 F Z 0
 1 Y 6T    0 Z W 0 0    FTL(BbeginB)
 2 Y 2T    0 S H 0 A
 3   2A   12 Z E 0
 4 U 2LA   3 F Z 1    Z FOFLA
 5 N 2T   26 S F 0 A
 6 U 2LA 16384      A Z PROFLA
 7 Y 2T    8 S F 0 A
 8   2S   24 Z E 1
 9   OLS  11 F Z 1
10   6T    0 Z W 0 0    FTL(ppp)
11   2S    3          A
12   6T    0    W 0 0   FTL(RET)
13   2S   16    E 1
14   6T    0    W 0 0   FTL(NID)
15   2S    1    E 1
16   6T    0    W 0 0   name code II
17   4P         SB
18   6T    4 S F 0 0
19   0B   27 Z E 0
20   6S    0    X 0 B   fill address pile
21   2S    0 Z E 1
22   6T    0    W 0 0   FTL(name code I)
23   2S   23 F Z 0
24   6T    0 Z W 0 0    FTL(Bproc)
25   2T   26 S F 0 A
   DC
   DO
```

simple block introduction

```
   DA    0 S F 0
   DI
 0   6T    4 S F 0 0   FTL(this moment's PRA)
 1   0LS  26 F Z 0
 2   6T    0 Z W 0 0
 3   2T    9   X 0   E
 4   2S   13 U Z 0
 5   2P   15   SS
 6   0LS  12   Z 0
 7   2T    8   X 0   E
 8   2S    7       A       simple block introd.
 9   6T    0 Z W 0 0       FTL(RR)
10   6T    4 S F 0 0
11   0LS   8 Y E 0
12   6T    0 Z W 0 0       FTL(this moment's PRA)
13   2S   22 F Z 0
14   6T    0 Z W 0 0       FTL( N = 0 )
15   2S   48       A
16   2T   26 S H 1 A       FTL(PC) two times
17   2S   12 Z E 0
18   6T    0   W 0 0       FTL(Flag word)
19   2S    3       A
20   6T    0   W 0 0       FTL(RET)
21   2S   12   E 1
22   0LS   6 F Z 1
23   6T    0 Z W 0 0       FTL(APC)
24   2S   31 F Z 0
25   6T    0 Z W 0 0       FTL(BbeginB)
26   2S   11   E 0
27   3LS  31       A Z
28 N 7Y 1053 C 0
29   2T    0 S H 0 A
   DC
   DO
```

INB II
transfer of names from prescan list
to name list

```
   DA    0 S H O
   DI
 0   2S   19 Z E 0          NLSC
 1   6S    2   E 2
 2   2B   23   E 0          PLIB
 3   2S    0   X 0 B        new PLIB
 4   6S    3   E 2
 5   2S    1   X 0 B
 6   6S    0   E 2          FF
 7   2S    2   X 0 B
 8   6S   31   E 1          EE
 9   2S    3   X 0 B
10   6S   30   E 1          DD
11   2S    4   X 0 B
12   6S   29   E 1          CC
13   2S    5   X 0 B
14   6S   28   E 1          BB
15   2S    6   X 0 B
16   6S   27   E 1          AA
17   2A    2       A
18   6A    1   X 0
19   2S    3   E 2   Z
20   1S   23   E 0
21 N 1S    7       A
22 Y 1S    1       A
23   0S   19   E 0
24 N 6S    1   E 2
25   1S   19   E 0
26 N 0B    7       A
27 Y 0B    1       A
28   3A   23   E 0
29 N 1A    7       A
30 Y 1A    1       A
31   0A   19   E 0
   DC
   DO
```

```
   DA    0 S H 1
   DI
 0    4S   19 Z E 0
 1    6T    0 S R 0 0       TR
 2    2B    3 Z E 2
 3    6B   23   E 0
 4    2S    0   X 0 B P
 5    4T   20 S H 0 1 P
 6    2B   23 Z E 0
 7    2S    0   X 0 B
 8    6S   23   E 0
 9    2S   12   E 0         Flag word
10    2LS  13 F Z 1   Z     FOFLA
11    2A   12 Z E 1
12    6A   13   E 2         temporary APC
13    0A   30   E 1
14    6A   12   E 1         APC
15 N 6T    0 L W 0 3       implicit jump
16 N 6T    0 S F 0 1       4 * FTL(PRA)
17 N 6T    0 Z W 0 0
18 N 6T    0   W 0 0
19 N 6T    0   W 0 0
20 N 2S    4 F Z 1
21 N 6T    0 Z W 0 0       FTL(Bfor)
22 N 2T   17 S K 1 A
23    2B   11 Z E 0   Z
24 Y 2T    3 S K 0 A
25    2T    0   K 0 A
26    6T    0 Z W 0 0       insertion in SFO
27    6T    0   W 0 0
28    2T   17 S F 0 A
   DC
   DO
```

INB III
production of SCC
filling up prestack list

```
          DA   O S K O
          DI
       0   2B   0       A
       1   2S   5       A
       2   6T   0 U E 0 2      FOT(SCC)
       3   2S  11 Z E 0
       4   2B   6       A
       5   6T   0 U E 0 2      FOT(BN)
       6   2S   8       A
       7   2A  12 Z E 0
       8 U 2LA 16384     A Z    PROFLA
       9 N 2A   1   E 1         name code II
      10 N 3P  15   AA
      11 N 2LA 63       A
      12 N 6A   0   X 0
      13   6S  19   E 1         pp
      14   0S   0   X 0
      15   6S  18   E 1         LN for LAD
      16   0S  30   E 1
      17   0S  30   E 1
      18   0S  31   E 1
      19   6S  21   E 1         qq
      20   0S  27   E 1
      21   6S  22   E 1         rr
      22   0S  28   E 1
      23   6S  23   E 1         ss
      24   0S  29   E 1
      25   2B   5       A
      26   6T   0 U E 0 2      FOT(LN)
      27   2A  28 Z E 1   Z
      28 Y 2A  29   E 1   Z
      29 Y 2S   0       A
      30 N 2S  22   E 1
      31   6T  22 S K 1 2      FOT(LN)
          DC
          DO
```

```
   DA   0 S K 1
   DI
 0 Y 2T  19 S K 1 A      INB IV
 1   2A  26 Z E 1        PPP
 2   0A   2       A
 3   6A  24   E 1        ppp
 4   0A   0   E 2        FF
 5   6A  26   E 1
 6   2A   1       A
 7   2B  25   E 1        PSLB
 8   4A   0   X 0 B
 9   2A  12 U Z 0
10   0B  24 Z E 1
11   0LA 15 F Z 0
12   6A  32766 X 0 B     BI(block identifier)
13   2A  28 Z E 1
14   0LA 12 F Z 1
15   6A  32767 X 0 B
16   2T  19 S K 1 A
17   2S   8       A
18   6S  18 Z E 1
19   2S  30 Z E 1   Z
20 N 2T   0 S L 0 A
21   2T   0   S 0 A      skip LAD-production
22   2A   0       A
23   6A  22 Z E 1
24   2A  28   E 1
25   6A  23   E 1
26   2T   0 U E 0 A
   DC
   DO
```

INB IV
production of LAD

```
   DA   0 S L 0
   DI
 0   2B   0       A
 1   2S   8       A
 2   6T   0 U E 0 2     FOT(LAD)
 3   2B   5       A
 4   2S  18 Z E 1
 5   6T   0 U E 0 2     FOT(LN)
 6   2B   5       A
 7   2S  30 Z E 1
 8   6T   0 U E 0 2     FOT(N)
 9   6S   0   X 0
10   2B   1 Z E 2
11   6B  14   E 1
12   2B  14   E 1
13   4P       BA
14   2B  32767 X 0 B
15   5B  14   E 1
16   4P       AB
17   2A  32766 X 0 B Z  label ?
18 N 2T  12 S L 0 A
19   2A  13 Z E 2       temporary APC
20   2P   6   AA
21   0A  11   E 0
22   2P  10   AA
23   0A  18   E 1
24   2S   2       A
25   4S  18   E 1
26   6A  32766 X 0 B    name code II touched up
27   2S  13   E 2
28   2B   2       A
29   6T   0 U E 0 2     FOT(NAP)
30   2S   1       A
31   4S  13 Z E 2
   DA   0 S L 1
   DI
 0   4T  12   L 0 0 P
 1   2T   0 S S 0 A
   DC
   DO
```

TR(transport)
A = distance
B = first address to be transported
S = number

```
          DA    0 S R 0
          DI
       0    6A    0    X 1     P
       1 N  2A    1         A
       2 Y  3A    1         A
       3    6A    1    X 1
       4    6S    0    X 0     Z
       5 N  2S    0    X 0 B
       6 N  0B    0    X 1
       7 N  6S    0    X 0 B
       8 N  1B    0    X 1
       9 N  0B    1    X 1
      10 N  4T    5    R 0 0 P
      11    2T    8    X 0     E
          DC
          DO
```

switches and labels in INB;
construction of name code II

```
   DA   0 S S 0
   DI
 0   2B   1 Z E 2
 1   6B  14   E 1
 2   2B  14   E 1
 3   5P       BA
 4   0A   2   E 2   Z
 5 Y 2T   0 S T 0 A
 6   2A  32767 X 0 B
 7 U 2LA 18 F Z 0   Z   no switch ?
 8   2LA 32767    A
 9   5A  14 Z E 1
10 Y 2T   2 S S 0 A
11   2A  32766 X 0 B
12   4P       AS
13   3P  15   SS
14   0S   1       A
15   2A  11 Z E 0
16   0P  10   AA
17   0A  18   E 1
18   4S  18   E 1
19   6A  32766 X 0 B
20   2T   2 S S 0 A
   DC
   DO
```

procedures in INB
adds number in address pile to name code
of procedures, declared in the block

```
   DA   0 S T 0
   DI
 0   2B  19 Z E 0        NLSC
 1   6B  14   E 1
 2   2B  14   E 1
 3 U 1B   1   E 2   Z
 4 N 2A  12   E 1        APC
 5 N 4A  32766 X 0 B
 6 N 2B  32767 X 0 B
 7 N 5B  14   E 1
 8 Y 2T  12   X 0   E
 9   2S   1        A
10   4S  12   E 1
11   2T   2 S T 0 A
   DC
   DO
```

";" after formal parameter part

```
  DA    0 S U 0
  DI
0   2B    4 Y E 1 A
1   6T    0 K N 0 0     require
2 N 7Y  1054 C 0
3   2A   12 Z E 0
4 U 2LA  2        A Z
5 N 2B    6 Y E 1 A     clear Flag word
6 Y 2B   23 Y E 1 A     COFLA & SFLA
7   6T    0 L Z 0 0     put
8 N 6T    0 S E 0 4     INB I
9   2T    0 Z Y 0 A
  DC
  DO
```

";" after specification
See also page 154, construction of value assignments

```
   DA   0 S Y 0
   DI
 0   2B   2 Y E 1 A
 1   6T   0 K N 0 0
 2 N 7Y  1054 C 0
 3   2A   0 Z E 1
 4 U 2LA 30 Y E 0   Z
 5 Y 7Y  1054 C 0
 6   2S  12 Z E 0
 7 U 2LS 1024     A Z  AFLA
 8 Y 2T  12 S Y 0 A
 9 U 2LS 14848    A Z
10 N 7Y  1054 C 0
11   2T  11 T L 0 A
12 U 2LS 512      A Z  POFLA
13 Y 2T  20 S Y 0 A
14 U 2LS 15360    A Z
15 N 7Y  1054 C 0
16   6T  28 T L 2 0
17 U 2LS 224      A Z
18 Y 7Y  1054 C 0
19   2T  30 T L 0 A
20 U 2LS 2048     A Z  LEFLA
21 Y 2T  25 S Y 0 A
22 U 2LS 13824    A Z
23 N 7Y  1054 C 0
24   2T  23 T L 1 A
25 U 2LS 8192     A Z  STRIFLA
26 Y 2T  30 S Y 0 A
27   2LS 8160     A Z
28 N 7Y  1054 C 0
29   2T  31 T L 2 A
30 U 2LS 4096     A Z  SIFLA
31 Y 2T  16 T L 2 A
   DC
   DO
```

```
  DA   0 S Y 1
  DI
0 U 2LS 12256     A Z
1 N 7Y  1054 C 0
2   2T  31 T L 2 A      calculate Q from Flag word
  DC
  DO
```

";" after value part

```
   DA   0 S N 0
   DI
0    2B  26 Y E 0 A
1    6T   0 K N 0 0   require
2 N  7Y  1054 C 0
3    2S   1 Z E 1     name code II
4    0S   2 F Z 0     value bit
5    2B  16 Z E 1     NID
6    6S  32766 X 0 B
7    2B  23 Y E 1 A
8    6T   0 L Z 0 0   put
9    2T   0 Z Y 0 A
   DC
   DO
```

";" after switch declaration

```
   DA    0 T Z 0
   DI
 0   6T   16 T S 0 0   simple case ?
 1 N 6T    0 L F 0 5   NNP
 2 N 6T    0 R F 0 6   conditional expressions
 3   2A    1       A
 4   4A   26 Z E 0
 5   2B   26   E 0
 6   6T   22 T S 0 0   shifting in translator list
 7 Y 6T   27   S 0 0   insert PDA
 8 N 2S    8 Z E 2
 9 N 0S    9 F Z 0
10 N 6S  32767 X 0 B
11 N 2B    0       A
12 N 2S    4       A
13 N 6T   21 W F 0 2   FOT(EIS)
14   2A   12 Z E 0
15 U 2LA  16       A Z  DEFLA
16 Y 7Y   1090 C 0
17   2LA  15 F Z 0   Z  PEFLA
18 Y 6T   13 R F 0 3
19   6T    0 R W 0 5     backward scanner V
20   2T    0 Z Y 0 A
   DC
   DO
```

";" after array declaration

```
  DA   0 T E 0
  DI
0   2B  22 Y E 4 A
1   6T   0 K N 0 0  require
2 N 7Y  1085 C 0
3   2B  11 Y E 0 A
4   6T   0 L Z 0 0  put
5   2T   0 Z Y 0 A
  DC
  DO
```

";" after variable declaration

```
   DA    0 T F 0
   DI
0    2B    6 Y E 1 A
1    6T    0 K N 0 0   require
2 N  7Y   1054 C 0
3    6T    5 T T 0 1   like "," type list
4    2B    6 Y E 1 A                 separator
5    6T    0 L Z 0 0   put
6    2T    0 Z Y 0 A
   DC
   DO
```

"," in value list

```
  DA   0 T H 0
  DI
0   2B  26 Y E 0 A
1   6T   0 K N 0 0  require
2 N 7Y  1063 C 0
3   2S   1 Z E 1    name code II
4   0S   2 F Z 0    value bit
5   2B  16 Z E 1    NID
6   6S  32766 X 0 B
7   2T   0 Z Y 0 A
  DC
  DO
```

"," formal variable separator

```
   DA   0 T K 0
   DI
 0   2B  28 Y E 0 A
 1   6T   0 K N 0 0  require
 2 N 7Y  1063 C 0
 3   2A  20 Z E 0
 4   1A  19   E 0
 5   0A  30 Y E 0
 6   6A   0 Z E 1    name code I
 7   6T  10 T K 0 0
 8   6T   0 F F 0 0  FNL
 9   2T   0 Z Y 0 A
10   2A  11   E 0    BN
11   2P  10   AA
12   0A  19   E 1    pp
13   6A   1   E 1    name code II
14   2S   2       A
15   4S  19   E 1    pp:= pp+2
16   2T   8   X 0   E
   DC
   DO
```

"," specification separator
construction of value assignments

```
   DA   O T L O
   DI
 0   2B   2 Y E 1 A
 1   6T   0 K N 0 0   require
 2 N 7Y  1063 C 0
 3   2A   0 Z E 1
 4 U 2LA 30 Y E 0   Z   AFLA
 5 Y 7Y  1063 C 0
 6   2S  12 Z E 0
 7 U 2LS 1024     A Z
 8 Y 2T  23 T L 0 A
 9 U 2LS 14848    A Z
10 N 7Y  1063 C 0
11   6T  28 T L 2 0
12   2B   0       A
13 U 2LS 32       A Z  BOFLA
14 N 2S  11       A    VBAD
15 N 2T  19 T L 0 A
16 U 2LS 64       A Z  INFLA
17 N 2S  10       A    VIAD
18 Y 2S   9       A    VRAD
19   6T   0 U E 0 2    FOT(macro-order)
20   2B   5       A
21   6T  18 T L 4 3    FOT(LN)
22   2T  31 T L 2 A    calculate Q from Flag word
23 U 2LS 512      A Z  POFLA
24 Y 2T  19 T L 1 A
25 U 2LS 15360    A Z
26 N 7Y  1063 C 0
27   6T  28 T L 2 0
28 U 2LS 224      A Z
29 Y 7Y  1063 C 0
30   2B   0       A
31 U 2LS 96       A Z
   DC
   DO
```

```
   DA    0 T L 1
   DI
 0 Y 2S   27        A      TRA
 1 N 2S   28        A      TIA
 2   6T    0 U E 0 2       FOT(macro-order)
 3   2B    1        A
 4   6T   14 T L 4 3       FOT(DA)
 5   2B    0        A
 6   2S   46        A
 7   6T    0 U E 0 2       FOT(FPCR)
 8   2B    6        A
 9   2S    0        A
10   6T    0 U E 0 2       FOT(N)
11   2B    1        A
12   6T   14 T L 4 3       FOT(DA)
13   2B    0        A
14   2S    6        A
15   6T    0 U E 0 2       FOT(TNR)
16   2S   49        A
17   6T    0 U E 0 2       FOT(ST)
18   2T   31 T L 2 A       calculate Q
19 U 2LS 2048      A Z     LEFLA
20 Y 2T   6 T L 2 A
21 U 2LS 13824     A Z
22 N 7Y  1063 C 0
23   6T   28 T L 2 0
24   2B    0        A
25   2S   28        A
26   6T    0 U E 0 2        FOT(TIA)
27   2B    1        A
28   6T   14 T L 4 3       FOT(DA)
29   2B    0        A
30   2S   44        A
31   6T    0 U E 0 2       FOT(TFL)
   DC
   DO
```

```
   DA   0 T L 2
   DI
 0   2B   1        A
 1   6T  14 T L 4 3      FOT(DA)
 2   2B   0        A
 3   2S  75        A
 4   6T   0 U E 0 2      FOT(STL)
 5   2T  31 T L 2 A      calculate Q
 6 U 2LS 8192      A Z   STRIFLA
 7 Y 2T  11 T L 2 A
 8   2LS 8160      A Z
 9 N 7Y  1063 C 0
10   2T  31 T L 2 A      calculate Q
11 U 2LS 4096      A Z   SIFLA
12 Y 2T  16 T L 2 A
13 U 2LS 12256     A Z
14 N 7Y  1063 C 0
15   2T  31 T L 2 A      Calculate Q
16   6T  28 T L 2 0
17   2B   0        A
18 U 2LS 96        A Z
19 Y 2S  27        A
20 N 2S  28        A
21   6T   0 U E 0 2      FOT(TRA or TIA)
22   2B   1        A
23   6T  14 T L 4 3      FOT(DA)
24   2B   0        A
25   2S  41        A
26   6T   0 U E 0 2      FOT(TFR)
27   2T  11 T L 1 A
28   2A   1 Z E 1        if value return
29 U 2LA  2 F Z 0   Z    else go on
30 N 2T   8   X 0   E
31   2S  12 Z E 0
   DC
   DO
```

```
   DA   0 T L 3
   DI
 0 U 2LS 1536      A Z    AFLA & POFLA
 1 N 2T  21 T L 4 A
 2   2A   1 Z E 1        name code II
 3 U 2LA  2 F Z 0   Z    erase value bit
 4 N 1A   2   Z 0
 5 N 2B  16 Z E 1
 6 N 6A  32766 X 0 B
 7 N 2A   0        A
 8 Y 2A   2        A
 9 U 2LS 64        A Z    INFLA
10 N 0A   4        A
11 U 2LS 128       A Z    REFLA
12 N 0A   5        A
13 U 2LS 512       A Z    POFLA
14 Y 2T  26 T L 3 A
15   0A  32        A
16 U 2LS 224       A Z    BOFLA,INFLA,REFLA
17 N 0A  256       A
18   2T   0 T L 4 A
19   2A  10        A
20 U 2LS 32        A Z    BOFLA
21 N 2T   0 T L 4 A
22 U 2LS 64        A Z    INFLA
23   0A   4        A
24 Y 0A   1        A
25   2T   0 T L 4 A
26 U 2LS 2048      A Z    LEFLA
27 N 0A  64        A
28 U 2LS 4096      A Z    SIFLA
29 N 0A  16        A
30 U 2LS 8192      A Z    STRIFLA
31 N 0A  128       A
   DC
   DO
```

```
   DA   0 T L 4
   DI
 0   2P  16   AA
 1   2B  16 Z E 1          NID
 2   2B  32767 X 0 B
 3   4P       BS
 4   2B  16 Z E 1
 5   6S  32767 X 0 B
 6   4A  32767 X 0 B
 7   2S   0 F Z 0
 8   4S  32766 X 0 B
 9   2S   8 Z E 0
10 U 1S  87        A Z
11 N 2B   6 Y E 1 A
12 N 6T   0 L Z 0 0          put
13   2T   0 Z Y 0 A
14   2S   1   E 1           name code II
15   2LS 32767    A
16   6T   0 U E 0 2          FOT(DA)
17   2T  11   X 0   E
18   2S   1 Z E 1
19   2LS 1023     A
20   2T  16 T L 4 A          FOT(LN)
21 U 2LS 1024     A Z        AFLA
22 N 2T  19 T L 3 A
23   2A   1 Z E 1
24   2LA  2 F Z 0   Z        non-value ?
25 N 1S  512      A
26   2T   2 T L 3 A
   DC
   DO
```

"," array segment separator

```
   DA    0 T R 0
   DI
 0   2A   12 Z E 0
 1   2LA  2048      A Z     LEFLA
 2 N 2T   10 T R 0 A       after bound pair list
 3   2B   20 Y E 4 A
 4   6T    0 K N 0 0       require
 5 N 7Y   1084 C 0
 6   2A    1         A
 7   4A    9 Z E 2         array counter
 8   6T    0 T W 0 1       FNC I
 9   2T   15 T R 0 A
10   2B   22 Y E 4 A
11   6T    0 K N 0 0       require
12 N 7Y   1084 C 0
13   2A   2048      A
14   5A   12 Z E 0         LEFLA
15 N 2B   24 Y E 4 A
16 N 6T    0 L Z 0 0       put
17   2T    0 Z Y 0 A
   DC
   DO
```

"," switch list separator

```
   DA    0 T S 0
   DI
 0   6T   16 T S 0 0        simple case
 1 N 6T    0 L F 0 5        NNP
 2 N 6T    0 R F 0 6        conditional expressions
 3   2A    1        A
 4   4A   26 Z E 0
 5   2B   26   E 0
 6   4A   32763 X 0 B       counting elements
 7   6T   22 T S 0 0        shift in translator list
 8 Y 6T   27   S 0 0        insert PDA
 9 N 2S    8 Z E 2
10 N 0S    9 F Z 0
11 N 6S   32767 X 0 B
12 N 2B    0        A
13 N 2S    4        A
14 N 6T   21 W F 0 2        FOT(EIS)
15   2T   26 W H 0 A
16   2A   12 Z E 0
17 U 2LA 2048      A Z      LEFLA
18 Y 2B   26   E 0
19 Y 2S   32767 X 0 B       Bswitch on translator list?
20 Y 1S    9 Y E 4   Z
21   2T    8   X 0
22   2S   32766 X 0 B P
23 N 2T    8   X 0   E
24   6S   32767 X 0 B
25   1B    1        A
26   2T   22 T S 0 A
27   2S    1 Z E 1
28   2LS 32767     A
29   0S   13 Y E 4
30   6S   32767 X 0 B
31   2T    8   X 0   E
   DC
   DO
```

"," type list separator

```
   DA   0 T T 0
   DI
 0   2B   6 Y E 1 A
 1   6T   0 K N 0 0        require
 2 N 7Y  1063 C 0
 3   6T   5 T T 0 1        see also the corresponding
 4   2T   0 Z Y 0 A        ";"
 5   2S  12 Z E 0
 6 U 2LS 32       A Z      BOFLA
 7 Y 2T  10 T T 0 A
 8   2A   0       A
 9   2T  13 T T 0 A
10 U 2LS 64       A Z      INFLA
11 Y 2A   5       A
12 N 2A   4       A
13   2P  16   AA
14   0A  20 Z E 0           TNLSC
15   1A  19   E 0
16   0A   3       A
17   6A   0   E 1           name code I
18   2A  11   E 0           BN
19   2P  10   AA
20 U 2LS 256      A Z      WOFLA
21 N 0A   9 F Z 0
22 N 0A  22 Z E 1           rr
23 Y 0A  21   E 1           qq
24   6A   1   E 1           name code II
25   2A   1       A
26 N 2T  31 T T 0 A
27 U 2LS 128      A Z      REFLA
28 N 0A   8 Z E 1
29   4A  21   E 1           qq
30   2T   2 T T 1 A
31 U 2LS 128      A Z      REFLA
   DC
   DB
```

```
  DA    0 T  T 1
  DI
0 N 0A    8 Z  E 1
1   4A   22    E 1        rr
2   6T    0 F  F 0 0      FNL
3   2T    9    X 0    E
  DC
  DO
```

FNC(fill name code) I

```
   DA    0 T W 0
   DI
 0    2S   12 Z E 0
 1    2A    8        A
 2 U  2LS  32        A Z          BOFLA
 3 N  2T    7 T W 0 A
 4 U  2LS  64        A Z          INFLA
 5    0A    4        A
 6 Y  0A    1        A
 7    2P   16    AA
 8    0A   20 Z E 0               TNLSC
 9    1A   19    E 0              -NLSC
10    0A    3        A
11    6A    0    E 1              name code I
12    2A   11    E 0              BN
13    2P   10    AA
14 U  2LS 256         A Z          WOFLA
15 N  0A    9 F Z 0
16    6A    1 Z E 1               provisional name code II
17    6T    0 F F 0 0             FNL
18    2T    9    X 0    E
   DC
   DO
```

"," actual parameter separator

```
  DA    0 T U 0
  DI
0   6T   11 W S 0 7          TAP
1   2B   16 Y E 4 A
2   6T    0 K N 0 0          require
3   6T    0 L Z 0 0          put
4 N 7Y  1063 C 0
5   2LA 31       A Z
6 Y 7Y  1063 C 0
7   6T    0 R R 0 1          IPW
8   2T    0 Z Y 0 A
  DC
  DO
```

"," for list separator

```
   DA   0 T Y 0
   DI
 0   2B  31 Y E 3 A
 1   6T   0 K N 0 0   require
 2 N 7Y  1081 C 0
 3   2S  12 Z E 0
 4 U 2LS 15       A Z
 5 Y 7Y  1081 C 0
 6 U 2LS 15 Y E 3   Z        WIFLA
 7 Y 2T  12 T Y 0 A
 8 U 2LS  8       A Z
 9 N 7Y  1081 C 0
10   2S  21       A          FOR3
11   2T  17 T Y 0 A
12 U 2LS  8       A Z
13 Y 7Y  1081 C 0
14 U 2LS 16 Y E 3   Z        SUFLA
15 N 2S  24       A          FOR6
16 Y 2S  20       A          FOR2
17   2B   0       A
18   6T   0 U E 0 2          FOT(macro-order)
19   2B  31 Y E 3 A
20   6T   0 L Z 0 0          put
21   2T   0 Z Y 0 A
   DC
   DO
```

"," index separator

```
   DA    0 T N 0
   DI
0    2B  28 Y E 4 A
1    6T   0 K N 0 0          require
2 N 7Y  1087 C 0
3    2B  24 Y E 4 A
4    6T   0 L Z 0 0          put
5    3A   1       A
6    2B  26 Z E 0
7    6T  13 W E 0 2          counting indices
8    2T   0 Z Y 0 A
   DC
   DO
```

"," index separator

//bound pair

```
      DA    0 W Z 0
      DI
 0     2B     4 Y Z 0 A
 1     6T     0 K N 0 0          require
 2 N   7Y   1083 C 0
 3     6T     0 L Z 0 0          put
 4     2A     1       A
 5     4A    10 Z E 2            dimension counter
 6     2T     0     Y 0 A
      DC
      DO
```

")" after formal parameter

```
   DA    0 W E 0
   DI
 0    2B   28 Y E 0 A
 1    6T    0 K N 0 0          require
 2 N  7Y  1064 C 0
 3    2A   20 Z E 0
 4    1A   19   E 0
 5    0A   30 Y E 0
 6    6A    0 Z E 1           name code I
 7    6T   10 T K 0 0         name code II
 8    6T    0 F F 0 0         FNL
 9    6T    0 R W 0 5         backward scanner V
10    2S    0 F Z 0
11    5S   12 Z E 0           erase NIFLA
12    2T    0   Y 0 A
13    2S  32766 X 0 B         index counter
14    6T   10 R R 0 1         part of IPW
15    2T   10   X 0   E
   DC
   DO
```

")" after actual parameter

```
   DA    0 W F 0
   DI
 0   6T   11 W S 0 7          TAP
 1   2S   12 Z E 0
 2   2LS  15 F Z 0   Z        PEFLA
 3 Y 6T   13 R F 0 3          FOT(PRA)
 4   2B   28 Y E 2 A
 5   6T    0 K N 0 0          require
 6 N 7Y   1064 C 0
 7   2A   12 Z E 0
 8   2LA  31       A Z
 9 Y 7Y   1064 C 0
10   2B   26 Z E 0
11   2S   32767 X 0 B
12   0LS  21 Y E 0   Z        B(actpar on translator list?
13 N 7Y   1064 C 0
14   6T    0 R W 0 5          backward scanner V
15   2S   12 Z E 0
16   2LS  31       A
17   6S   31   E 0            T' := T
18   2B   15 Y E 3 A
19   6T    0 L Z 0 0          put
20   2T    0 Z Y 0 A
21   2A    1       A
22   5A   11 Z E 0
23   2T    0 U E 0 A          FOT
   DC
   DO
```

:= in switch declaration

```
   DA    0 W H 0
   DI
 0    2B    4 Y E 4 A
 1    6T    0 K N 0 0          require
 2 N  7Y  1082 C 0
 3    6T    0 L Z 0 0          put
 4    3S   15 F Z 1            cleared Flag word
 5    6T    0 Z W 0 0          FTL
 6    2S    3 Y E 5
 7    6T    0 Z W 0 0          FTL(N)
 8    2S    1   E 1
 9    2LS 1023     A
10    0S   12 F Z 1
11    6T    0 Z W 0 0          FTL(LN)
12    2S    8 Y E 4
13    6T    0 Z W 0 0          FTL(SID)
14    2S    9 Y E 4
15    6T    0 Z W 0 0          FTL(Bswitch)
16    6T   18 W H 0 0
17    2T    0 Z Y 0 A
18    2S   13 U Z 0
19    2P   15   SS
20    0LS  12   Z 0
21    6S    8 Z E 2
22    2T    8   X 0   E
23    2B    2 Y E 5 A
24    6T    0 L Z 0 0          put
25    2T   16 W H 0 A
26    2A   12 Z E 0
27    2LA  16       A Z        DEFLA
28 N  2T   23 W H 0 A
29    7Y  1090 C 0
   DC
   DO
```

:= in for statement

```
   DA   0 W K 0
   DI
 0   2A   5       A
 1   6T   0 L F 0 5          NNP
 2   2B   7 Y E 3 A
 3   6T   0 K N 0 0          require
 4 N 7Y  1075 C 0
 5   6T   0 L Z 0 0          put
 6   2B  26 Z E 0
 7   2S  32767 X 0 B
 8   0LS  4 F Z 1   Z        Bfor on translator list?
 9 N 7Y  1075 C 0
10   2B   0       A
11   2S  19       A
12   6T   0 U E 0 2          FOT(FOR1)
13   6T  13 R F 0 3          FOT(PRA)
14   2B   0       A
15   2S  48       A
16   6T   0 U E 0 2          FOT(PC)
17   2B   6       A
18   2S   0       A
19   6T   0 U E 0 2          FOT(N)
20   2B  12       A
21   6T   0 U E 0 2          FOT(D12)
22   2B   0       A
23   2S   7       A
24   6T   0 U E 0 2          FOT(RR)
25   6T   0 L W 0 3          implicit jump
26   2T   0 Z Y 0 A
   DC
   DO
```

PSE(prepare complicated switch element)

```
   DA   O W L O
   DI
 0   2A  12 Z E O
 1 U 2LA 4096     A Z        SIFLA
 2 Y 2T  12   X O   E
 3   2B  26 Z E O
 4   2S  32767 X O B
 5   1S   9 Y E 4   Z        Bswitch on translator list?
 6 N 2T  12   X O   E
 7   2LA 15 F Z O   Z        PEFLA
 8 N 5A  12 Z E O
 9 N 6T   0 L W O 3          implicit jump
10   2A   1       A
11 N 4A   8 Z E 2            element counter
12   0A  11 Z E O            BN
13 U 3LA 31       A Z
14 Y 6A  11 Z E O
15 N 7Y  1053 C O
16   2T  12   X O   E
   DC
   DO
```

SAP(simple actual parameter)

```
   DA    0 W R O
   DI
 0   2S  12 Z E 0
 1 U 2LS  1       A Z
 2 Y 2T  21 W R 0 A          no constant
 3   6T   0 L E 0 4          PCP for PORD-tail
 4   2A  12 Z E 0
 5 U 2LA  4       A Z        IFLA
 6 Y 2S  15 Z E 0
 7 N 2S  16   E 0
 8   2B   8       A
 9   6T  21 W F 0 2          FOT(integer or first part
10 N 2T  17 W R 0 A                 mantissa)
11   2A   8 Z E 1   Z
12 N 2S  16   E 0
13 N 6T   0 U E 0 2          FOT(tail of mantissa)
14   2S  17 Z E 0
15   2B   7       A P
16   6T   0 U E 0 2          FOT(exponent)
17   2A   7       A
18 N 1A   1       A P
19   6T   3 W S 0 3          PORD-head
20   2T  15   X 0   E
21   2S   0 Z E 1
22 U 2LS  9 F Z 0   Z
23 Y 2T   0 W R 1 A
24 U 2LS 17 F Z 0   Z        non-formal
25 Y 7Y 1036 C 0
26   2A 3072       A P
27   6T   3 W S 0 3          PORD-head
28   2A   1 Z E 1
29   2LA 32767     A
30   6T   0 W T 0 0          PORD-tail
31   2T  15   X 0   E        exit formal
   DC
   DO
```

```
   DA    0 W R 1
   DI
 0 U 2LS 17 F Z 0    Z          non-formal ?
 1 N 2T  26 W R 0 A
 2 U 2LS 15 F Z 0    Z          non-procedure ?
 3 Y 2T  17 W R 1 A
 4   2A   1 Z E 1
 5   1P  16   SA
 6   0P  15   AS
 7   2LA 14 Y E 4
 8 U 2LS  0 F Z 0    Z
 9 N 3LA 31744       A
10   6T   4 W T 0 0             PORD-tail
11 U 2LS 14 F Z 0    Z
12   2A  800        A
13 N 2A  548        A
14 N 2LS 19 F Z 0    Z
15 N 0A   1        A P
16   2T  19 W R 0 A
17 U 2LS  5 F Z 0    Z          non-array ?
18 Y 2T  30 W R 1 A
19   2A   1 Z E 1
20   1P  16   SA
21   0P  15   AS
22 U 2LA  2 F Z 1    Z
23   2LA 14 Y E 4
24   6T   2 W T 0 0             PORD-tail
25   2A  1032       A
26 N 0A  1024       A
27 U 2LS 19 F Z 0    Z
28 N 0A   1        A P
29   2T  19 W R 0 A
30 U 2LS 18 F Z 0    Z          non-switch ?
31 Y 2T   5 W R 2 A
   DC
   DO
```

```
   DA    0 W R 2
   DI
 0   2A    1 Z E 1
 1   2LA 32767     A
 2   6T    0 W T 0 0                    PORD-tail
 3   2A  1040        A P
 4   2T  19 W R 0 A
 5 U 2LS   2 F Z 1     Z                non-label?
 6 Y 2T  12 W R 2 A
 7   2A    1 Z E 1
 8   2LA 32767     A
 9   6T    0 W T 0 0                    PORD-tail
10   2A  1088        A P
11   2T  19 W R 0 A
12   2A    1 Z E 1
13 U 2LA 14 F Z 0     Z
14   2LA 32767     A
15   6T    0 W T 0 0
16   2A  1024        A
17   2T  26 W R 1 A
   DC
   DO
```

TAP(translate actual parameter)

```
   DA    0 W S 0
   DI
 0    2B   1        A
 1    5B  26 Z E 0
 2 U OLA 128       A Z
 3    2B  26 Z E 0
 4    6A  32762 X 0 B          store PORD-head in
 5    2S  31 Y E 0             translator list
 6    4S  32762 X 0 B
 7 N 2B    0        A
 8 N 2S    4        A
 9 N 6T   21 W F 0 2           FOT(EIS)
10    2T  11    X 0   E
11    2B  26 Z E 0
12    2S  32767 X 0 B
13 U OLS 21 Y E 0   Z         B(actpar on translator
14 Y 2A   12 Z E 0                    list ?
15 Y 2LA 2048      A Z        LEFLA
16 Y 2T    0 W R 0 A          the simple case
17 U OLS 14 Y E 1   Z         ITAR on translator list?
18 Y 3S   12 Z E 0
19 Y 2LS 18 F Z 0   Z         PIFLA
20 Y 2A  512        A
21 Y 6T    0 W S 0 3          PORD-head
22    2A    0        A
23    6T    0 L F 0 5          NNP
24    6T    0 R F 0 6          conditional expressions
25 Y 2T   15    X 0   E       exit subscript expression
26    2B  26 Z E 0
27    2S  32767 X 0 B
28    OLS  2 Y E 3   Z        Bstrq on translator list ?
29 Y 2A  128        A
30 Y 6T    8 W S 1 3
31 Y 2T   15    X 0   E       exit string
   DC
   DO
```

```
   DA    0 W S 1
   DI
 0   2S   12 Z E 0
 1 U 2LS  16       A Z      DEFLA
 2 N 2A   576        A
 3 N 6T    4 W S 0 3
 4 N 2T   15    X 0   E     exit designational
 5   2A   516        A Z                expression
 6 N 6T    3 W S 0 3
 7   2T   15    X 0   E
 8   2B    1        A
 9   5B   11 Z E 0
10   2T    0 W S 0 A
   DC
   DO
```

TPW(tail of parameter word)
stores tail of parameter word (PORD) in translator list

```
      DA    0 W T 0
      DI
0       0A  10 Y E 4            DA
1       2T   5 W T 0 A
2       0A  11 Y E 4            DA + N
3       2T   5 W T 0 A
4       0A  12 Y E 4            NAP + N
5       2B  26 Z E 0
6       6A  32761 X 0 B
7       2T   8   X 0   E
      DC
      DO
```

FNC II (fill name code arrays)

```
   DA   0 W W 0
   DI
 0   2S   9 Z E 2          array counter
 1   6S   0   X 0
 2   2S  10   E 2          dimension counter
 3   2P  16   SS
 4   2A  12   E 0
 5   2LA 256      A Z      WOFLA
 6 N 6T  18 W W 0 0
 7 Y OS  21 Z E 1          99
 8   2B  19   E 0
 9   2A  32767 X 0 B
10   2LA 32767    A
11   1S  11   E 2
12   4S  32766 X 0 B
13   6A  12   E 2
14   1B  12   E 2
15   4T   9 W W 0 0 P
16   2T   9   X 0   E
17   2S   0       A
18   OS  23 Z E 1          ss
19   OS  10   E 2          dimensions
20   OS   2       A
21   2T   8   X 0   E
   DC
   DO
```

FOT(fill object tape)
B= type specificator
S= element offered for output

```
      DA    O U E O
      DI
   0  U 2A    3    D 0    P
   1    6A    5    Z 0
   2    6S    6    Z 0
   3    6T   13    F 0 0
   4  N 2T   28    E 0 A          skip punching
   5    2S    0    Z 0    Z
   6  N 2T   16    E 0 A
   7    0A   13    Z 0
   8  U 1A   26      A P
   9  N 6A   13    Z 0
  10  N 2T   16    E 0 A
  11    1A   27      A
  12    6A   13    Z 0    P
  13  N 7A   13    Z 0    P
  14  Y 2A    1      A
  15  Y 4A   12    Z 0
  16  U 1B   14    Z 0    Z       same directive ?
  17  N 3A   14    Z 0    P
  18    6B   14    Z 0
  19  N 5P        AA
  20  N 2B    5      A
  21  N 6T    0    H 0 1          insert directive
  22    2B    7    Z 0
  23    1B   11      A P
  24  Y 4B   14    Z 0            change D12
  25  N 2B   11    F 0 B
  26  N 2A    6    Z 0
  27  N 6T   21    H 1 1          AIB
  28    2A    5    Z 0
  29    2S    6    Z 0
  30    2B    7    Z 0
  31    2T   10    X 0    E
      DC
      DO
```

constants of output program

	DA 0 U F 0 DN	bits in obj.prog.	bits on tape	
0	+7340040	7	8	macro-order
1	+15728656	15	16	DA or BI
2	+20971531	20	11	NAP
3	+22020112	21	16	PDA
4	+22020117	21	21	PRA
5	+10485771	10	11	LN
6	+5242886	5	6	N
7	+12582925	12	13	PORD-head or char.
8	+28311580	27	27	integer or mant.
9	+7340040	7	8	string symbol
10	+11	0	11	APC
11	+21	0	21	element of address
12	+20971520 DC DO	20	0	D12 pile

insertion in FOT

```
    DA  13 U F 0
    DI
13    2A   1       A
14  U 1B  12       A Z
15  Y 4A  12 Z E 1
16  U 1B  11       A Z
17  Y 5A  12 Z E 1
18    6B   7 U Z 0
19    2A   0   F 0 B
20    3P  20   AA
21    2T   8   X 0   E
    DC
    DO
```

AIB(add imparity bit)
A= output unit offered
B= number of bits of output unit

```
   DA    O U H O
   DI
 0   2S   8   Z 0
 1 U 1B  27       A P
 2 N 6Z  31   X 0 B
 3 N 6S   0   X 1
 4 N 0A   0   X 1
 5 N 2S   0       A
 6 U 1B   5       A Z
 7 Y 0S   0   D15
 8   0B   9   Z 0
 9   6B   9   Z 0
10   1B  28       A
11   6S   4   X 1
12   7B   3   X 1
13   2S  11   Z 0
14   1S   5       A Z
15 Y 7S  11   Z 0
16 Y 2B   4       A
17 N 2B   5       A
18 N 2S   1       A
19 N 4S  11   Z 0
20   5B   9   Z 0
21   0B   3   X 1
22   2S   4   X 1
23   0P   0   SA  B
24   6S   3   X 1
25   0LS 10   Z 0
26   6S  10   Z 0
27 N 2S   3   X 1
28 N 2T   7   H 1 A
29 U 2LS 16       A Z
30 N 0LS 17       A
31   4P       SB
   DC
   DO
```

```
   DA    0 U H 1
   DI
 0   2S  27030     A
 1   1P   1    SS  B P
 2   2S   3    X 1
 3 Y 0S   0    D15        make impair
 4   0P   1    SS
 5   2B   0         A
 6   6B  10    Z 0        erase parity pentad
 7   6A   3    X 1
 8   6T   0    K 0 0      AOB
 9   2A   3    X 1
10   2B   0         A
11   6B   4    X 1
12   2S   9    Z 0
13   1S   4         A P
14 N 0S  11    Z 0
15 N 1S   5         A Z
16 Y 2T  12    H 0 A
17   2B   9    Z 0
18   0P   0    AA  B
19   6A   8    Z 0
20   2T   9    X 0    E
21   4P        AS
22   7Z  31    X 0 B Z
23 Y 2T   0 U H 0 A
24   7Y 1036   C 0
   DC
   DO
```

AOB(administration output buffer)

```
   DA    O U K O
   DI
 0    2A    0    Z O    P    initial state ?
 1 Y  2B    5       A
 2 Y  7A    0    Z O
 3 Y  6A    1    Z O
 4 Y  7B    2    Z O
 5 Y  6A    3    Z O
 6 Y  6B    4    Z O
 7    2B    4    Z O
 8 U  1B    2    Z O    Z
 9 Y  2A    3    Z O
10 Y  1A    1    Z O    Z
11 Y  2T    7    K O A       wait, no more space left in
12    2A   31       A       output buffer
13    6Z   27    X 2 B
14    2B    3    Z O
15    2LA   0    L O B       filling buffer
16    5A    0    L O B
17    4S    0    L O B
18    2B    3    Z O A
19    0Y   64    X24
20    6T   10    K 114       cyclical counter
21 U  2A    2    Z O    P
22 Y  2T    8    X O    E
23    6A    1    Z O
24    6S    2    Z O
25    2A    8    X O
26    6T    8    D 114       standard entrance type-punch
27    2B    1    Z O         program
28    2S    0    L O B
29    2B    2    Z O
30    7Z   27    X O B
31    6T   15    D 114       TPWW
   DC
   DO
```

```
   DA    0 U K 1
   DI
 0    6Z    1    X16
 1    2B    1    Z 0 A
 2    6T   10    K 114          cyclical counter
 3    2A    1    Z 0
 4    2S    2    Z 0
 5 U  1S    4    Z 0    Z
 6 Y  1A    3    Z 0    Z
 7 N  2T   27    K 0 A
 8    7S    2    Z 0
 9    2T   13    D 1 A          standard exit
10    2A    1    X 0 B
11    6A   15    Z 0
12 U  1A   20         A P
13 N  0A    5         A
14 Y  2A    5         A
15    6A    1    X 0 B
16    2S    0    X 0 B
17    4P         SA
18 Y  0S    1         A
19 Y  2LS  31         A
20 Y  6S    0    X 0 B
21    2S   15    Z 0
22    2T   22    X 0    E
   DC
   DO
```

punch end of object tape

```
   DA    O U R O
   DI
 0   2S  13    Z O
 1   OP  15    SS
 2   OS  12    Z O
 3   2B  11        A
 4   6T   0    E O 2          FOT(address of object program)
 5   2B  12        A
 6   6T   0    E O 2          FOT(concluding directive)
 7   2S  11    Z O   Z        pentad counter
 8 N 1S   6        A
 9   3X   5        A
10   1S   9    Z O   Z
11 Y 2S  31        A
12 Y 2T  28    R O A
13   4P        SB    P
14 N OB  30        A
15   1B   1        A Z
16 N 2S   0    D15
17 Y 2S   0        A
18   2A   0        A
19 N 7Z   1-   X 3 B
20   OA   8    Z O
21 N OP   0    SA  B
22   6B   6    X 1
23   OB   9    Z O
24   6B   9    Z O
25   1B  27        A
26   6T  11    H O 1          AIB( concluding ones)
27   2S   6    X 1
28   6T   0    K O O          AOB( number of concluding ones )
29   6T   0 R U O 1           punch tape feed
30   2T  11    X O   E
   DC
   DO
```

```
   DA    0 Y Z 0
   DU
 0 +1101011111101111111111111
 1 +1000000000000000000000000
 2 +1010000000010000000000000
 3 +0
 4 +1011101111111011111111000
 5 +1000
 6 +1000000010000000000
 7 +1000000000000000000
 8 +1011111111101111111111111
 9 +10010
10 +1000000000000000
11 +0
12 +10000000000000001
   DC
   DO
```

```
   DA    0 Y E 0
   DU
 0 +1101111111111111111111111111
 1 +1000000000010000000000000000
 2 +1010000000000000000000000000
 3 +1000000000000000000000000000
 4 +1000000000000000
 5 +1011111111111111011111111
 6 +0
 7 +1110000000000000000000000000
 8 +1100000000000000000000000000
 9 +1111111101111100001111
10 +0
11 +1110000000000000000000000000
12 +1100000000000000000000000000
13 +1111111111111111111111111111
14 +1000000000000000000000000000
15 +1010000000000000000000000000
16 +1000000000000000000000000000
17 +1111111101111110001111
18 +0
19 +1110000000000000000000000000
20 +1000000000000000000000000000
21 +1000000000000000000000000100
22 +1000000001011111111110010
23 +10
24 +1011111000010000000000000000
25 +1101000000000000000000000000
26 +1111111111011111111111111111
27 +1000000001000000000000010
28 +1011111111101111111111111111
29 +1000000000000000000000010
30 +1000000000000011
31 +1110000000000000000000000000
   DC
   DO
```

```
   DA    0 Y E 1
   DU
 0 +1 00000000000000000000001111
 1 +1 00000000000000000000010000
 2 +1 01111111111100000010000001 0
 3 +1 00000000000000000000000010
 4 +1 11111111111111111100000000
 5 +1 00000000100000000000000
 6 +1 11111111111111111000011111
 7 +1 10000000000000000000000010
 8 +1 11111111111101111111111111
 9 +1 00000000000000000000000000
10 +1 101 11000011100111111 00000
11 +0
12 +1 0001 00000001 00000000000
13 +1 00000000000100000000000
14 +1 1101 0000000001 00110
15 +1 01 1101 111 011 01 1111111 0011
16 +1 0
17 +1 01 0001 00001 001 00000000000
18 +1 00000000000000000000
19 +1 00000000000000000000000110
20 +1 10000000011100111111100000
21 +0
22 +1111 10000100000000000000
23 +1 00000000000000000000
24 +111111111 00000
25 +11111111111101111111111111111
26 +1 01 00000000000000000000000000
27 +11111111111101111111000100 10
28 +1 00000000000000000000000010
29 +1 001 111111111 01 111111111111111
30 +1 00000000000000000000000101
31 +1 001 1 00000001 00000000000
   DC
   DO
```

```
   DA    0 Y E 2
   DU
 0 +1 0000000000000000000
 1 +1 1 01 0000000000000000
 2 +1 1 0001 1 0001 1 1 01 1 1 1 1 1 1 1 1 1 1
 3 +1 0000000000000000000
 4 +1 1 000000000001 00001 1
 5 +1 1 0000000001 1 1 001 1 1 1 1 1 1 000
 6 +0
 7 +1 001 1 00000001 00000000000
 8 +1 0000000000000000000
 9 +1 1 1 1 1 1 01 1 1 01 1 1 1 1 1 1 1 1 1 1 1
10 +0
11 +1 0000011 0001 00000000000000
12 +1 0000000000000000000
13 +1 001 00001 1 1 01 1 1 1 1 1 1 1 1 1
14 +0
15 +1 000000000000000000000001 1 1 0
16 +1 000000000000000000000001 1 01
17 +1 1 001 0000011 1 001 1 1 1 1 1 1 000
18 +0
19 +1 001 1 00000001 00000000000
20 +1 0001 0000000000000000000
21 +1 001 000001 1 1 001 1 1 1 1 1 00000
22 +0
23 +1 01 0001 00000001 00000000000
24 +0
25 +1 1 1 1 1 0000000000000000000000
26 +1 1 1 1 1 000000000000000
27 +1 1 1 1 0000000000000000
28 +1 1 01 0000001 1 1 1 01 1 1 1 1 1 00000
29 +1 0000000000000000000000
30 +1 1 01 01 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
31 +1 001 0000000000000000000
   DC
   DO
```

```
   DA    0 Y E 3
   DU
 0 +1 0001 00000001 00000000000
 1 +1 000000000001 00000000000
 2 +1 00000000000000000001 0001
 3 +1 1 01 1111111101111111111111
 4 +1 00000000000000000000000000
 5 +1 00001 0001 000000000000000
 6 +1 00001 0000000000000000000
 7 +1 1111111 01111111 01111111011
 8 +1 01 000000000000000000001 01 0
 9 +1 01 0001 00000001 00000000000
10 +1 00000000000000000000
11 +1 1 01 1 1 01 001 1 1 1 01 1 1 1 1 1 10000
12 +1 0000000000000000000
13 +1 01 00011 0000001 00000000000
14 +1 01 000000000000000000000000
15 +1 000000000000000000
16 +1 00000000000000000
17 +1 01 0001 00001 001 00000000000
18 +1 000000000001 00000000000
19 +1 1 01 1 1 01 1 1 1 1 1 1 01 1 1 1 1 1 1 000
20 +1 0000000000000001 000
21 +1 0001 001 00001 00000000000
22 +1 001 00000000000000000
23 +1 1 01 1 1 01 1 1 1 1 1 1 101 1 1 1 1 1 1 000
24 +1 01 00000000000001 000
25 +1 0001 00000001 00000000000
26 +1 0000000000000000000
27 +1 1 01 1 1 01 1 1 1 1 1 1 01 1 1 1 1 1 1 1 000
28 +1 0000000000000001 000
29 +1 0001 01 000001 00000000000
30 +1 01 000000000000000000
31 +1 1 01 1 1 01 001 1 1 1 01 1 1 1 1 1 10000
   DC
   DO
```

```
   DA    0 Y E 4
   DU
 0 +1 000000000000000000
 1 +1 0001 011 00001 00000000000
 2 +1 000000000000000000
 3 +1 11 01 0000000000000000000000
 4 +1 11111111111111111111111111
 5 +1 00000000001 00000001 001 0
 6 +1 01 01 00000001 00000000000
 7 +1 01 000000000000000000
 8 +1 11 01 0000000000011 00
 9 +1 0000000000000000001 001 0
10 +1 101 000000000000000000000
11 +1 11 00000000000000000000000
12 +1 111 00000000000000000000
13 +1 1 000000000000000000000
14 +1 111111111111111111
15 +1 1 000000000000000000001
16 +1 1 000000001 111 01 1111 100000
17 +0
18 +1 001 1 00000001 00000000000
19 +1 0000000000000000000
20 +1 1111 01 1111111 01 000000011
21 +1 1 000000000000000000001 0
22 +1 1111 01 1111111 01 000011111
23 +1 1 000000000001 00000000000
24 +1 11111111111111111111111111
25 +1 000001 000000000000000000
26 +1 11111111111111111111111111
27 +1 0000000000000000000
28 +1 011 101 11111 0001 1111 11 000
29 +1 000
30 +1 011 1 000000000000000000000
31 +1 001 000000000000000000000
   DC
   DO
```

```
   DA    0 Y E 5
   DU
 0 +1 00000000000000000000000
 1 +0
 2 +11 0000000000000000000001
 3 +101 000000000000000000001
 4 +1 0001 00000001 00000000000
 5 +1 0000000000000000000
 6 +1 0000000000000000000
 7 +0
   DC
   DO


   DA  16 Y E 5
   DI
16 U  1A   87          A Z        insertion in prescan
17 N  2T   21 Y E 5 A               program
18 U  1S    4 Z E 1     Z
19 Y  5S    3 Z E 1
20    2T   19 Z U11 A
21 U  1A   101         A Z
22 Y  5S    4 Z E 1     Z
23 N  2T   19 Z U11 A
24    2S    3 Z E 1     Z
25 N  7Y   1034 C 0
26    2T    0 Z U12 A
   DC
   DO


   DS
```

3. Contents of working space

0	ZEO	state of reading routines; 0 ≡ initial
1		case definition of Flexowriter
2		LC (line counter)
3		QC (string quote counter)
4		stock of NAS
5		CFLA (comment flag); 0 ≡ comment allowed
6		
7		stock of NRS
8		last relevant symbol read
9		BEC (begin - end counter)
10		priority number
11		OBC or BN; counts block level
12		Flag word
13		SC (symbol counter)
14		DECFLA (declaration flag during prescan)
15		FNW (facultative number word; head of mantissa)
16		INW (imperative number word; integer or tail mantissa)
17		exponent
18		
19		NLSC (name list counter)
20		TNLSC (temporary NLSC)
21		
22		INLSC (initial NLSC)
23		PLIB (begin of prescan list)
24		
25		PLIE (end of prescan list)
26		TLSC (translator list counter)
27		BAP (begin address pile)
28		BFLA (procedure heading flag during prescan)
29		ITLSC (initial TLSC)
30		BC (block counter; counts total number of blocks)
31	ZEO	T' (last 5 bits of Flagword, in stock)

```
 0 ZE1  name code I
 1      name code II
 2
 3      round bracket counter
 4      square bracket counter
 5      sum check of relevant symbols
 6      LPSL (length of prestack list)
 7
 8      if 0 then produce ALP-program
 9      own-flag
10      dimension counter
11      real, integer or Boolean function
12      APC (address pile counter)
13      if 0 then skip priority test in backward scanning
14
15
16      NID (address of Name code I of IDentifier in name list)
17
18      LN for LAD or for SID
19      pp (LN of formals)
20
21      qq (LN of non-own local variables or storage functions)
22      rr (LN of own variables)
23      ss (LN of own arrays)
24      ppp (address in prestack list, relative to PSLB)
25      PSLB (begin of prestack list)
26      PPP (address of new block in prestack list, relative to PSLB)
27      AA (length of storage for non-own local variables and storage
                                                              functions)
28      BB (length of storage for own variables)
29      CC (length of storage for own storage functions)
30      DD (number of labels)
31 ZE1  EE (number of switches + 2 * total number of switch elements)
```

0	ZE2	FF (number of own arrays in block)
1		
2		
3		
4		
5		
6		
7		string quote counter
8		switch element counter
9		array counter
10	ZE2	dimension counter

0	UZO	state of output routines (0≡initial)
1		⎫ cyclical counter for
2		⎭ emptying buffer
3		⎫ cyclical counter for
4		⎭ filling buffer
5		contents of A-register before FOT
6		" " S- " " "
7		" " B- " " "
8		partial pentad
9		number of bits of partial pentad
10		parity pentad
11		pentad counter
12		physical address of PRA
13		bit number of PRA
14	UZO	controlling directive

4.

4.1. Adaptation to other installations

The following paragraph definitions at the beginning of the tape are installation-dependent (see page 1):

```
DPEZ    m1 XO     m1 = (27D16) + 87
DPES    m2 XO     m2 = (27D16)
DPET    m3 XO     m3 = number of X1-words in living store
```

4.2. Punching a binary tape

The standard tape consists of the translator program text published in this report, followed by a copy of P74 (read binary tape fast) located in OX1 - 3OX1.
After reading in the standard tape the start address, the stop address and the console word switches are given the following sets of values:

	(1DO) :=	(2DO) :=	(3DO) :=
1.	OX1	31	15
2.	25XO	1	12
3.	(27D16)+13	13	12
4.	23X2	2	12
5.	(27D16)+167	4341	62

After every setting of the switches, give autostart 9.

5. Adding names of standard functions and machine code procedures

The translating system may be extended by adding new machine code procedures to the complex tape. Because these procedures need not be explicitly declared, the fixed contents of name list and address pile have to be duly supplemented as follows:

1. Choose a free paragraph name for the new procedure. As is well known, by means of the thirteen letters Z,E,F,H,K,L,R,S,T,W,U,Y and N 169 ordered letter pairs can be formed, from ZZ to NN, and one of the numbers 1(1)169 is associated to each of them in an obvious way.
At the time of writing the paragraph names ZZ,....,TF have been used for other purposes, so TH is the first free name. Its associated number n0 is 107.
We now define n1(TH) = n0-65 = 42.
The fixed contents of the address pile are now supplemented by adding a correction tape to the standard translator tape:

```
DPRZ  4398 EZO
 DA   n0 RZO DI
 OA   n1 ESO
      DCDO
```

2. Calculate the quantities R and Q for the name code.
These are binary numbers, the bits of which are numbered from right to left $R_0,\ldots,R_3$ and $Q_0,\ldots,Q_9$.

R_0:= if function procedure then 1 else 0

$R_1:=R_2:=R_3:=0$ for all procedures

Q_0:= if real procedure then 1 else 0

$Q_1:=Q_5:=Q_4:=Q_6:=Q_7:=Q_9:=0$ for all procedures considered here.

Q_2:= if real procedure V integer procedure then 1 else 0.

$Q_5 = 1$

$Q_8 = R_0$

Another quantity n2 is defined by

n2:= if number of formal parameters is undefined

then 16
else number of formal parameters modulo 16

The internal representation of basic ALGOL symbols is chosen in such a way, that symbols occurring in identifiers are represented by 6-bit numbers. Of these letters and digits at most four are shifted from left to right into an X1-word. The remaining 3 or 9 or 15 or 21 bits are filled up with ones. E.g.: the letters of ABSFIXT are stored as follows

first name word:	F	S	B	A	111
second name word:	T	X	I	1 11111	111

name code I = number of name words +2 + 2 ↑16 * Q
name code II = n1 + 2 ↑16 * n2 + 2 ↑22 * R
n3 = previous paragraph definition of ER
n4 = previous paragraph definition of EL

The correction tape for the name list has the following form:

```
DPLN  4462  EZO
 DA   n3   LNO DN
"first name word"
"second name word"
      .....
"last name word"
 name code II
 nade code  I
    DCDO
```

3. Among the first paragraph definitions of the name list the definitions of EL and ER have to be changed.

n3:= n3+ number of name words +2

n4:= n4+1

DPER n3 XO

DPEL n4 XO

4. In general,it will not be necessary to change other paragraph definitions. In paragraph RZ 22 X1-words have been left open for adding new addresses and in paragraph LN 43 X1-words have been left open for name words and name codes.

5. After these corrections of the standard translator tape a new binary tape will have to be made. In the fifth set of switch values at the time of writing (2DO) was 4341. But now
(2DO):= previous value of (2DO) + number of name words + 2.

6. Internal representation of ALGOL 60 basic symbols

symbol	code	symbol	code	symbol	code	symbol	code
0	0	w	32	+	64	while	96
1	1	x	33	-	65	comment	97
2	2	y	34	×	66	(	98
3	3	z	35	/	67	)	99
4	4			÷	68	[	100
5	5	A	37	↑	69	]	101
6	6	B	38	>	70	'	102
7	7	C	39	≥	71	'	103
8	8	D	40	=	72	begin	104
9	9	E	41	≤	73	end	105
a	10	F	42	<	74	own	106
b	11	G	43	≠	75	Boolean	107
c	12	H	44	¬	76	integer	108
d	13	I	45	∧	77	real	109
e	14	J	46	∨	78	array	110
f	15	K	47	⊃	79	switch	111
g	16	L	48	≡	80	procedure	112
h	17	M	49	go to	81	string	113
i	18	N	50	if	82	label	114
j	19	O	51	then	83	value	115
k	20	P	52	else	84	true	116
l	21	Q	53	for	85	false	117
m	22	R	54	do	86	tab	118
n	23	S	55	,	87	nlcr	119
o	24	T	56	.	88	'	120
p	25	U	57	10	89	"	121
q	26	V	58	:	90	stopcode or?	122
r	27	W	59	;	91		
s	28	X	60	:=	92		
t	29	Y	61	␣	93		
u	30	Z	62	step	94		
v	31			until	95		

N.B. The Flexowriter symbols tab nlcr ' and " are not accepted as ALGOL 60 symbols except in strings.

The Flexowriter symbols ? or stopcode denote the end of the ALGOL program.

The ALGOL symbol ␣ corresponds with the Flexowriter symbol space. It has no significance outside strings.

7. Flexowriter code

Symbol Upper Case	lower case	code	Symbol Upper Case	lower case	code
A	a	97	∧	0	32
B	b	98	∨	1	1
C	c	115	x	2	2
D	d	100	/	3	19
E	e	117	=	4	4
F	f	118	;	5	21
G	g	103	[	6	22
H	h	104	]	7	7
I	i	121	(	8	8
J	j	81	)	9	25
K	k	82	|	—	14
L	l	67	>	<	49
M	m	84	"	+	112
N	n	69	?	,	91
O	o	70	'	10	59
P	p	87	:	.	107
Q	q	88	¬	-	64
R	r	73	tab		62
S	s	50	nlcr		26
T	t	35	lower case		122
U	u	52	Upper Case		124
V	v	37	space		16
W	w	38	erase		127
X	x	55	stopcode		11
Y	y	56			
Z	z	41			

Addenda and corrigenda in MR61, Text of the second ALGOL 60 translator for the X1.

1. Corrections.

The correction tapes described can be simply added to (or read in after) the binary tape of the translator, provided this tape has been made following the instructions given in 4.1. and 4.2. and using the standard tape.
Most of the corrections refer exclusively to the use of the translator as a text-tester and are of no consequence to the object tapes produced optionally.

1.1

```
        DP YE 4014  EZ0
        DP ZU  763  EZ0
        DP ZE    0  EZ0
        DP HL 1836  EZ0
        DP FK 1504  EZ0
        DP ZL  469  EZ0
        DP LR 2349  EZ0
        DP HY 2001  EZ0


        DA   16 YE 5  DI
16   U  1A   87      A Z
17   N  2T   21 YE 5 A
18   U  1S    4 ZE 1   Z
19   Y  5S    3 ZE 1
20      2T   19 ZU11 A
21   U  1A   98      A Z
22   Y  4S    4 ZE 1
23   U  1A   99      A Z
24   Y  5S    4 ZE 1
25   U  1A 101       A Z
26   Y  5S    4 ZE 1   Z
27   N  2T   19 ZU11 A
28      2S    3 ZE 1   Z
29   Y  2T    0 ZU12 A
30      7Y 1034 C  0
        DCD0
```

This correction improves the checking of array declarations.

```
1.2.          DA   22 YE 0  DN
              + 16822258
              DA   24 YE 0
              + 49827840
              DCD0
```

This correction refers to function designators occurring in switch declarations.

```
1.3.          DA  20 HL 0  DI
     20       2T  26 ZU 9 A
              DA  26 ZU 9
     26       2A  12 ZE 0
     27      2LA2048      A Z
     28    Y 2A   0 ZE 1
     29    Y 2T  21 HL 0 A
     30       2A   1      A P
     31       2T  11 HL 1 A
              DCD0
```

Before this correction, procedure identifiers occurring in subscript expressions would sometimes give difficulties.

```
1.4.          DA  2 FK 0  DI
      2       0T  0     A
              DCD0
```

This is an adaptation to the definition of the relation, given in the Revised Report, section 3.4.1., differing from that in the original Report.

```
1.5.          DA  0 ZU 2 DI
      0       2T 24 ZU 2 A
              DA 24 ZU 2
```

```
24    U  1S  98       A Z
25    Y  2T   0 ZU 0 A
26    U  1S  99       A Z
27    N  2T  23 ZU 6 A
28       6T   0 ZL 0 5        RUND
29    U  1A  98       A Z
30    N  2T   3 ZU 0 A
31       7Y1062  C 0
         DA  23 ZU 6
23    U  1S 101       A Z
24    N  2T  12 YE 5 A
25       6T   0 ZL 0 5
26       2S  12 ZE 0
27      2LS   3       A Z
28    N  7Y1088  C 0
29       2T   3 ZU 0 A
         DA  15 HY 0
15       2T   8 YE 5 A
         DA   8 YE 5
 8       2S   0 ZE 1
 9    U 2LS  31 YE 5    Z
10    N  2T  16 HY 0 A
11       7Y1086  C 0
12       2S  12 ZE 0
13      2LS   2       A Z
14    N  2T   0 ZU 0 A
15       7Y1086  C 0
         DA  31 YE 5
31       0A   0  X 0    C
         DCD0
```

This correction causes syntactical errors due to the omission of operators or identifiers, or to discrepancies between declarations of identifiers and their use in expressions, to be detected sooner and more completely.

```
1.6.          DA  25 ZU 0  DI
     25    Y  2T  28 ZU 3 A
              DA  28 ZU 3
     28       3A   7 ZE 0
     29    U  1A 122      A Z
     30    N  2T  10 ZU 4 A
     31       2T  28 ZU10 A
              DA  28 ZU10
     28       2A   9 ZE 0   Z
     29    Y  2T  10 ZU 4 A
              7Y1020  C 0
              DCD0
```

This correction prevents the translator program from running wild in some cases of missing end's.

```
1.7.          DA   27 LR 1  DI
           Y 2LS   15     A
              DCD0
```

Without this correction, a formal array identifier would cause difficulties.

```
1.8.          DA   20 YE 1 DN
               +   50393056
              DCD0
```

This correction improves the checking of the syntactical situation preceding any delimiter "(".

1.9. Now follow some corrections not affecting the translator program, but only section 5., describing the "procedure" of adding names to the system.

Page 199 : for DPRZ ... DCD0

read

DPRZ 4206 EZ 0

```
                        DA    n1 RZ 0
                        DI  0A n0 ES 0
                        DCD0

          for           Q1 := Q5 := Q4 ...
          read
                        Q1 := Q3 := Q4 ...

Page 200 : for          DPLN ... EZ 0
          read
                        DPLN 4270 EZ 0

          for           nade code I
          read
                        name code I
```

2. Addenda

The following will probably be of only secondary importance to other X1-installations than that of the Mathematical Centre. It describes the provisions taken for adding 21 names to the fixed part of the name-list. The corresponding machine code procedures have not been - and will not be - added to the complex tape. So the only effect of adding these names will be, that ALGOL 60 procedures calling for the corresponding procedures may pass the translator when used as a text-tester.

Page 1 : installation-dependent paragraph definitions.

```
DP EZ    0  X 6
DP EE 4576 EZ 0
DP EF   80 EE 0
DP EH 1024 EF 0
DP EK  320 EH 0
DP EL   67  X 0
DP ER  250  X 0
DP ES    9  X 3
DP ET12288  X 0
```

Page 4 : paragraph definitions for RZ and LN

```
DP RZ 4206 EZ 0
DP LN 4302 EZ 0
```

Page 117 : additional names and name codes in fixed part of name list

	DA 23 LN 4 DN	
23	- 46 369 880	PUTEXT1
24	+ 3 963 391	
25	+ 65 558	
26	+ 2 097 156	
27	+ 38 092 375	available
28	+ 44 405 935	
29	+ 31 457 279	
30	+ 4 194 347	
31	+ 19 136 517	

	DA 0 LN 5	
0	+ 59 486 311	cons
1	+ 4 325 420	
2	+ 19 136 515	
3	+ 56 957 439	car
4	+ 4 259 885	
5	+ 19 136 515	
6	+ 57 055 743	cdr
7	+ 4 259 886	
8	+ 19 136 515	
9	+ 46 938 711	atom
10	+ 4 259 887	
11	+ 18 874 371	
12	+ 55 017 471	eq
13	+ 4 325 424	
14	+ 18 874 371	
15	+ 56 965 863	startlisp
16	+ 59 321 071	

```
17          + 54 525 951
18          +     65 585
19          +  2 097 157
20          + 44 733 543      call
21          +    131 122
22          +  2 097 155
23          + 27 598 047      readstring
24          + 38 648 551
25          + 34 340 863
26          +     65 587
27          +  2 097 157
28          + 48 838 351      printstring
29          + 57 587 951
30          + 34 317 823
31          +  4 259 892
            DCD0

        DA    0 LN 6  DI
 0          + 19 136 517
 1          + 61 750 479      pustring
 2          + 34 317 535
 3          +  4 259 893
 4          + 19 136 516
 5          + 23 805 119      number
 6          + 57 114 623
 7          +  4 259 894
 8          + 18 874 372
 9          + 63 608 063      value
10          + 31 457 279
11          +  4 259 895
12          + 19 136 516
13          + 38 648 551      string
14          + 34 340 863
15          +  4 259 896
16          + 19 136 516
17          + 30 055 535      delete
18          + 30 343 167
```

```
19          +     65 593
20          +  2 097 156
21          - 15 081 048      PLOT
22          +  4 390 970
23          + 19 136 515
24          - 15 081 048      PLOTFRAME
25          - 30 216 872
26          - 46 137 344
27          +    393 275
28          +  2 097 157
29          - 15 081 048      PLOTTEXT
30          - 14 789 688
31          +    524 348
        DA    0 LN 7
 0          +  2 097 156
 1          - 15 081 048      PLOTAXIS
 2          - 17 368 784
 3          +    327 741
 4          +  2 097 156
 5          - 15 081 048      PLOTCURVE
 6          - 10 783 936
 7          - 46 137 344
 8          +  4 390 974
 9          + 19 136 517
10          - 23 176 360
11          - 15 080 960
12          +    589 887      FIXPLOT
13          +  2 097 156
14          - 44 315 344
15          - 31 819 408
16          - 15 073 280      ABSFIXPLOT
17          +    589 888
18          +  2 097 157
19          - 23 469 736
20          - 15 080 960      FLOPLOT
21          +    589 889
```

```
22          + 2 097 156
23          - 46 897 280
24          +     65 558      KODE
25          + 2 097 155
            DCD0
```

Page 119 : additions to fixed contents of address pile

```
            DA   10 RZ 1  DI
10          0A   11 ES 3          PUTEXT1        TH
11          0A   12 ES 3          available      TK
12          0A   13 ES 3          cons           TL
13          0A   14 ES 3          car            TR
14          0A   15 ES 3          cdr            TS
15          0A   16 ES 3          atom           TT
16          0A   17 ES 3          eq             TW
17          0A   18 ES 3          startlisp      TU
18          0A   19 ES 3          call           TY
19          0A   20 ES 3          readstring     TN
20          0A   21 ES 3          printstring    WZ
21          0A   22 ES 3          pustring       WE
22          0A   23 ES 3          number         WF
23          0A   24 ES 3          value          WH
24          0A   25 ES 3          string         WK
25          0A   26 ES 3          delete         WL
26          0A   27 ES 3          PLOT           WR
27          0A   28 ES 3          PLOTFRAME      WS
28          0A   29 ES 3          PLOTTEXT       WT
29          0A   30 ES 3          PLOTAXIS       WU
30          0A   31 ES 3          PLOTCURVE      WY
31          0A    0 ES 4          FIXPLOT        WN
32          0A    1 ES 4          ABSFIXPLOT     UZ
32          0A    2 ES 4          FLOPLOT        UE
33          0A    3 ES 4          KODE           UF
            DCD0
```

Unlike the corrections, described in 1., the addenda cannot be effectuated by adding a correction tape to the binary tape. A new standard tape has to be made. Then, according to the instructions given in 4.2. of MR61, a new binary tape can be made.
For 4341 in line 5 now 4472 should be read.

Amsterdam, 1-7-64

P.J.J. van de Laarschot
J. Nederkoorn

Addenda and corrigenda (II) in MR 61, Text of the second ALGOL 60 translator for the X1.

The correction tapes described can be simply added to (or read in after) the binary tape of the translator, provided this tape has been made following the instructions given in 4.1. and 4.2. and using the standard tape.
The following additions have been written with the intention to find more than one failure in a program during one checkrun. When a failure has been detected, the user may try to scan further (after G7) with "Start Next Address". This facility, that already existed for some stops, has now been extended to a larger number, including stop 1048 (name undeclared).
All extra information delivered by restarting the program after detection of the first mistake, should be interpreted with great caution. No guarantee can be given, that stop numbers printed out in these circumstances, will make sense. If restarting results in the printing (after autostart G7) of non-existing stop numbers, or if the program stops reading the tape, no further information can be expected.

```
          DP ZE    0 EZ0
          DP FE 1407 EZ0
          DP RZ 4206 EZ0
          DP TT 3417 EZ0
          DP ZY 1289 EZ0
          DP RN 2727 EZ0
          DP RY 2704 EZ0
          DP ZR  661 EZ0
          DP SZ 2746 EZ0

          DA    27 FE 1  DI
          2T    16 RZ 2 A    ⇒
          DA    16 RZ 2  DI
    16    7Y  1048 ¢C 0          name undeclared
    17    2S     8      A
    18    4S    12 ZE 0          type arithmetical
    19    6T     5 TT 0 1    ⇒ fill name code I, name code II and FNL
    20    2T     6 ZY 0 A    ⇒ Basic Cycle
    21  N 0A 16383  X 0
    22    7Y     0 ¢C 0
    23    7P
    24    7P
15RN0⇒ 25   2S     2 ZE 0          line-number
    26    6T     0  D22 0 DT
    27  A G8-L8    XN     DI
    28    2A    23 RZ 2
    29    6A     8  X 0          restore λ0
    30    2T     9  X 0    E ⇒
    31    DCD0
```

```
          DA      0 RN 0  DI
G7⟹ 0     2A      8  X 0
     1    6A     23 RZ 2          save λ0
     2    2A      9  X 0
     3    6A     24 RZ 2          save λ1
     4    6T     10  D28 0    ⟹ NLCR
     5    2B     23  X 0
     6    2S 32767  X 0   B
     7   3LS     21 RZ 2
     8    1S     22 RZ 2   Z
     9  N 4P        BS
    10  Y 2S 32767  X 0   B
    11   2LS 16383       A
    12  U 1S   199       A P
    13    6T      0  D22 0 DT
    14  A G4 NL4    XN     DI
    15  Y 6T     25 RZ 2 1     ⟹ type line number
    16    2A     24 RZ 2
    17    6A      9  X 0          restore λ1
    18    2T     19  D 7 A     ⟹
          DCD0

          DA     13 RY 0  DI
         3LS      0 ZR 0 A        SPS
         3LS      0 SZ 0 A        START
          DA     20 RY 0  DI
         3LS      0 RN 0 A        type stop number
          DCD0

          DA      2 ZR 0  DI
          1Y      4 XP
          DCD0
```

Amsterdam, 1-8-'64
P.J.J. v.d. Laarschot
J. Nederkoorn.