EUR 5024 e

COMMISSION OF THE EUROPEAN COMMUNITIES

PLOTTY
AN INTERACTIVE PLOTTING PROGRAM FOR GRAPHS ON A TELETYPE BY MEANS OF THE SPC-16 MINI-COMPUTER

by

J. EDER and C. PAGNY

1973

Joint Nuclear Research Centre
Ispra Establishment - Italy

LEGAL NOTICE

This report is on sale at the addresses listed on cover page 4

at the price of B.Fr. 70.—

Commission of the
European Communities
D.G. XIII - C.I.D.
29, rue Aldringen
L u x e m b o u r g

September 1973

EUR 5024 e

PLOTTY - AN INTERACTIVE PLOTTING PROGRAM FOR GRAPHS ON A TELETYPE BY MEANS OF THE SPC-16 MINI-COMPUTER
by J. EDER and C. PAGNY

Commission of the European Communities
Joint Nuclear Research Centre - Ispra Establishment (Italy)
Luxembourg, September 1973 - 50 Pages - B.Fr. 70.—

A number of processors and utility subroutines have been developed which permit the plotting of curves and graphs on a teletype. These programs run on a system with a minimum configuration, only a SPC-16 mini-computer and a standard teletype is required. Actually the routines are executed in an 8K-DBOS environment, however the limited versions can be used even on 4K — stand alone systems. The PLOTTY processor is interactive in the sense that parameters required for the execution are directly asked by the program and input after the response of the operator. Thus, this processor can be run by operators with little or no programming experience.

EUR 5024 e

PLOTTY - AN INTERACTIVE PLOTTING PROGRAM FOR GRAPHS ON A TELETYPE BY MEANS OF THE SPC-16 MINI-COMPUTER
by J. EDER and C. PAGNY

Commission of the European Communities
Joint Nuclear Research Centre - Ispra Establishment (Italy)
Luxembourg, September 1973 - 50 Pages - B.Fr. 70.—

A number of processors and utility subroutines have been developed which permit the plotting of curves and graphs on a teletype. These programs run on a system with a minimum configuration, only a SPC-16 mini-computer and a standard teletype is required. Actually the routines are executed in an 8K-DBOS environment, however the limited versions can be used even on 4K — stand alone systems. The PLOTTY processor is interactive in the sense that parameters required for the execution are directly asked by the program and input after the response of the operator. Thus, this processor can be run by operators with little or no programming experience.

EUR 5024 e

PLOTTY - AN INTERACTIVE PLOTTING PROGRAM FOR GRAPHS ON A TELETYPE BY MEANS OF THE SPC-16 MINI-COMPUTER
by J. EDER and C. PAGNY

Commission of the European Communities
Joint Nuclear Research Centre - Ispra Establishment (Italy)
Luxembourg, September 1973 - 50 Pages - B.Fr. 70.—

A number of processors and utility subroutines have been developed which permit the plotting of curves and graphs on a teletype. These programs run on a system with a minimum configuration, only a SPC-16 mini-computer and a standard teletype is required. Actually the routines are executed in an 8K-DBOS environment, however the limited versions can be used even on 4K — stand alone systems. The PLOTTY processor is interactive in the sense that parameters required for the execution are directly asked by the program and input after the response of the operator. Thus, this processor can be run by operators with little or no programming experience.

EUR 5024 e

COMMISSION OF THE EUROPEAN COMMUNITIES

PLOTTY

AN INTERACTIVE PLOTTING PROGRAM FOR GRAPHS ON A TELETYPE BY MEANS OF THE SPC-16 MINI-COMPUTER

by

J. EDER and C. PAGNY

1973

Joint Nuclear Research Centre
Ispra Establishment - Italy

ABSTRACT

A number of processors and utility subroutines have been developed which permit the plotting of curves and graphs on a teletype. These programs run on a system with a minimum configuration, only a SPC-16 mini-computer and a standard teletype is required. Actually the routines are executed in an 8K-DBOS environment, however the limited versions can be used even on 4K — stand alone systems. The PLOTTY processor is interactive in the sense that parameters required for the execution are directly asked by the program and input after the response of the operator. Thus, this processor can be run by operators with little or no programming experience.

KEYWORDS

P-CODES
PROGRAMMING
DATA PROCESSING
PLOTTERS

1. INTRODUCTION

A general problem with minicomputers which are used for data-acquisition is to have a quick orientation on the data accumulated in the memory. The simplest way is to print the interesting values in form of a table and to translate the binary stored data into decimal format.
Although this procedure needs only little programming effort (*), a standard and ready-to-use processor has been written. The processor TABLE may be loaded by the DBOS-Executive and will then produce a preformatted hard copy on the teletype for all memory locations within prescribed limits.
More significant than a table is a graphical display of the results of a measurement.
At small installations a graphical display (video, oszillo, plotter) is often not available. Here the standard teletype must serve as a simple display peripheral (of course a lineprinter with a 132 characters carriage would do the job quicker and with higher resolution).
The processor PLOTTY is designed to display a maximum of 7 curves simultaneously. Once PLOTTY is loaded and executed, it will ask the

---

(*) (where no formatting and only hexadecimal output is sufficient the DBOS control command $PDUMP may be used).

operator for various parameters. Most of them are selfexplanatory (see sample program; the german test can easily be translated to any other language version). An error recovery is provided after each response of the operator, so that answers can be cancelled or modified. By typing ← (back arrow) the operator may step back one question or by typing ↑ (up arrow) exit from the program.

A number of subroutines is included in this report, since they are of general interest and simplify the construction of new programs. They are stored on the user-library file UL on the disc in PGS-format and can be called together with programs from the SPC-16 subroutine library LB directly by the core-load-builder and linkage editor. The UL-library will be extended in the future and is available to all users of the CAP-16 assembler. Since the SPC-16 FORTRAN IV-compiler has the capability to support calls to subroutines in assembler language, these library programs may be used even in FORTRAN programs.

2. GENERAL CONVENTIONS FOR U$ SUBROUTINE MODULES

We propose to identify all user-written subroutines of general interest by a 6 character name beginning with U$.

In order to arrive at a versatile, flexible and machine independent library system which can be produced and applied by various programmers resp. operators independently, some standard rules have to be set up. As far as the SPC-16 input/output system (IOS) is used, we refer to the SPC-16 operation manuals (CAP-16, FSOS, DBOS, etc.). The herein described U$ subroutines dont make use of the IOS, all I/O operations are direct and refer only to the teletype, thus they may be run even on teletype-only configurations. Entering a subroutine, it is supposed the teletype is in ''Receive & Echo'' mode. Exit resp. return from a subroutine should be made not before the teletype is switched back to the same mode, if a mode change within the subroutine ocurred. I/O operations should proceed with all interupts disabled in order to avoid interferences with an eventually simultaneously used IOS.

U$ subroutines should not contain a permanent DSECT area. The structure proposed in the operation manuals should be followed strictly.

3. DESCRIPTION OF ROUTINES

The following programs are described in this report

PROCESSORS	TABLE	3.1
	PLOTTY	3.2
SUBROUTINES	U$BYDC	3.3
	U$DEHE	3.4
	U$EDIT	3.5
	U$HEDE	3.6
	U$MAXN	3.7
	U$OPDO	3.8
	U$PLOT	3.9
	U$RESP	3.10
	U$VAL	3.11

All programs are available in source format on papertape. For use under the DBOS executive the processors are stored on the DC disc file and the subroutines are stored on the disc's UL user library file.
The listings given below are in the CAP-16 assembler language of the SPC-16 computer family (see CAP 16 manual for language specifications). As far as the flow diagrams are concerned the operating registers contents are indicated by a lower case r before the registers symbol.

3.1 TABLE

FEATURES:

- Printout of a delimited core memory area with sequentially stored bytes
- Translation of the binary values into decimal (by use of U$BYDC)
- The results are edited by the teletype in two half lines of 8 bytes each separated by spaces.

CALLING SEQUENCE:

```
$JOB
$TABLE @aaaa #R1: bbbb # R2: eeee
```

where

aaaa	address in hexadecimal where the processor is to be loaded
bbbb	address* of the first byte in core memory to be printed
eeee	address* of the last byte in core memory to be printed

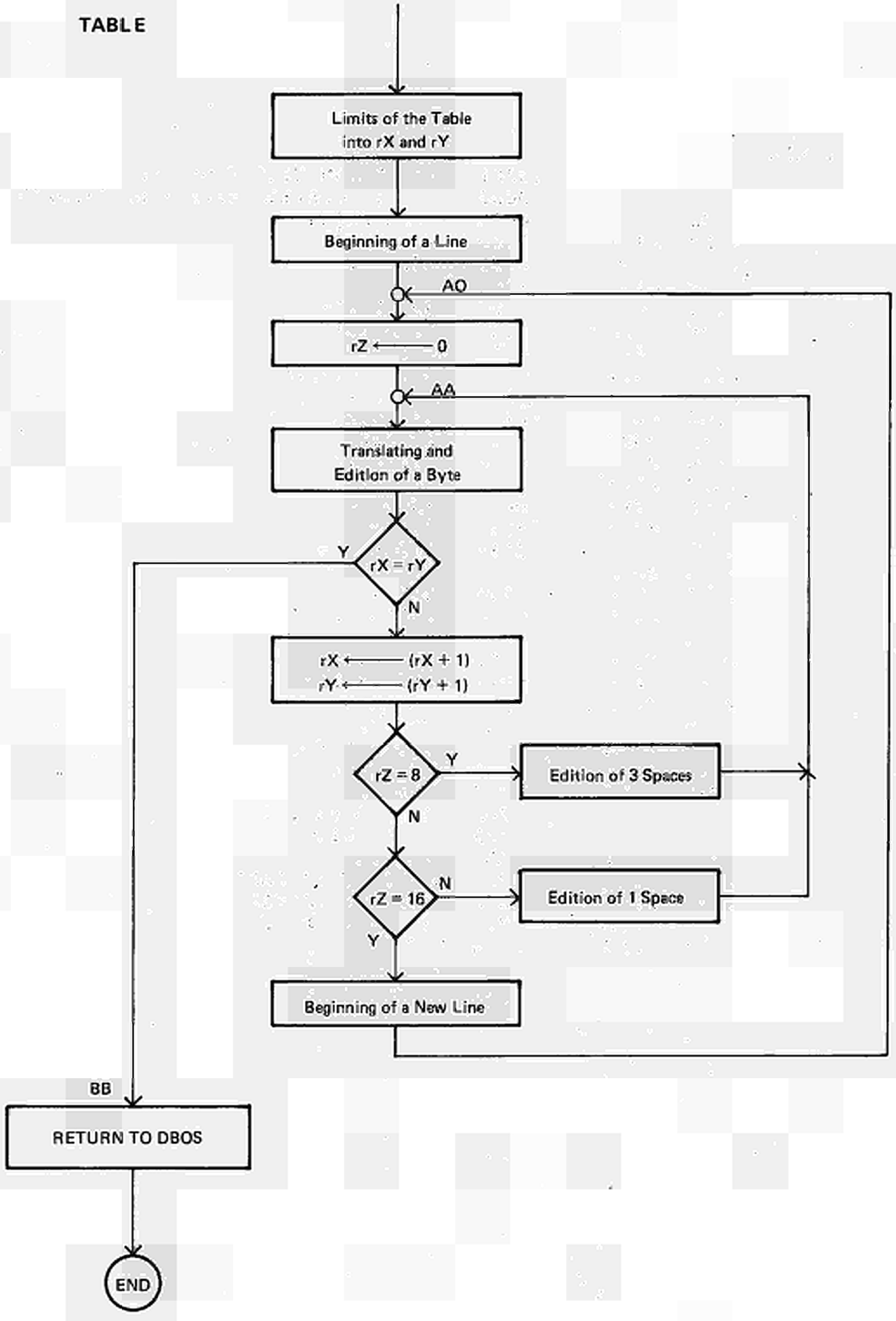
* : in hexadecimal

Program returns automatically to DBOS

MEMORY REQUIREMENTS

$7F_{16}$ words

TABLE


Limits of the Table into rX and rY
Beginning of a Line
AO
rZ ← 0
AA
Translating and Edition of a Byte
rX = rY
Y
N
rX ← (rX + 1)
rY ← (rY + 1)
rZ = 8
Y
Edition of 3 Spaces
N
rZ = 16
N
Edition of 1 Space
Y
Beginning of a New Line
BB
RETURN TO DBOS
END

```
?$SI=PR
?$A16
                                  0001    REF E$MON,U$EDIT,U$BYDC
                                  0002    REF E$CORE,E$MONE,E$SDC,F$IOR,IOS
        0000                      0003   TABLE PSECT
        0000  0939                0004    ADD X,X
        0001  0A59                0005    ADD Y,Y
        0002  074E                0006    INCR Y
        0003  641F                0007    JSR U$EDIT
        0004  0020 P              0008    DC NEWLIN
        0005  0660                0009   A0 ZERO Z
        0006  641D                0010   AA JSR U$BYDC
        0007                      0011   ASC DS 2
        0009  6419                0012    JSR U$EDIT
        000A  0007 P              0013    DC ASC
        000B  0A26                0014    SUBC X,Y
        000C  2C15                0015    SKZ BB
        000D  072E                0016    INCR X
        000E  076E                0017    INCR Z
        000F  0166                0018    SUBVC Z,8
        0010  0008
        0011  2C06                0019    SKZ A8
        0012  0166                0020    SUBVC Z,16
        0013  0010
        0014  2C06                0021    SKZ A16
        0015  640D                0022    JSR U$EDIT
        0016  001F P              0023    DC BLANC
        0017  73EE                0024    JMP AA
        0018  640A                0025   A8 JSR U$EDIT
        0019  001E P              0026    DC SPACE
        001A  73EB                0027    JMP AA
        001B  6407                0028   A16 JSR U$EDIT
        001C  0020 P              0029    DC NEWLIN
        001D  73E7                0030    JMP A0
        001E  A0A0                0031   SPACE DC X'A0A0'
        001F  A000                0032   BLANC DC X'A000'
        0020  8D8A                0033   NEWLIN DC X'8D8A'
        0021  0000                0034    DC 0
        0022  6402                0035   BB JSR E$MON
              0000 P              0036    END TABLE

        0023  0000 X
        0024  0000 X
        0025  0000 X
 NO  ERRORS
 DBOS CC
?$OD
?
```

```
    DEOS CC
?$EUS@1E00@1E00
```

```
1A00/1A47M
1A00=0001 0203 0405 0607 0809 0A0B 0C0D 0E0F
1A08=1011 1213 1415 1617 1819 1A1B 1C1D 1E1F
1A10=2021 2223 2425 2627 2829 2A2B 2C2D 2E2F
1A18=3031 3233 3435 3637 3839 3A3B 3C3D 3E3F
1A20=4041 4243 4445 4647 4849 4A4B 4C4D 4E4F
1A28=5051 5253 5455 5657 5859 5A5B 5C5D 5E5F
1A30=6061 6263 6465 6667 6869 6A6B 6C6D 6E6F
1A38=7071 7273 7475 7677 7879 7A7B 7C7D 7E7F
1A40=8081 8283 8485 8687 8889 8A8B 8C8D 8E8F
```

```
 DEOS CC
?$JOB
?$TABLE@1000#R1:1A00#R2:1A47
  0   1   2   3   4   5   6   7    8   9  10  11  12  13  14  15
 16  17  18  19  20  21  22  23   24  25  26  27  28  29  30  31
 32  33  34  35  36  37  38  39   40  41  42  43  44  45  46  47
 48  49  50  51  52  53  54  55   56  57  58  59  60  61  62  63
 64  65  66  67  68  69  70  71   72  73  74  75  76  77  78  79
 80  81  82  83  84  85  86  87   88  89  90  91  92  93  94  95
 96  97  98  99 100 101 102 103  104 105 106 107 108 109 110 111
112 113 114 115 116 117 118 119  120 121 122 123 124 125 126 127
128 129 130 131 132 133 134 135  136 137 138 139 140 141 142 143
                                                          DEOS C
?
```

3.2 PLOTTY

FEATURES:

- Plotting of up to seven curves simultaneously on the teletype
- Interactive
- Selection of number of curves (max 7)
- Selection of number of points/curve (length)
- Selection of channels (in multiplexer-systems)
- Selection of path
- Free choice of any printable ASCII character
- Automatic selection of scale factors
- Optionnal overriding of automatic scalefactors
- Automatic drawing of axis
- Indication of overflow
- Error recovery

CALLING SEQUENCE:

```
$JOB
$PLOTTY   @ aaaa
```

where aaaa hexadecimal loading address of the processor

PROGRAM DESCRIPTION

The program responds with

```
PLOT ROUTINE    VERSION 72
ZAHL DER KURVEN (MAX 7) =
```

An input is expected now. The operator should type the number of curves desired (in decimal). A carriage return (CR) terminates the input, however the following characters are recognised as control signals

↑	repeat question
←	repeat question
CR	repeat question
blank	take the previous value and print it

If the input value is > 7, modulo 8 will be taken as result. After a correct input the program continues to put the next question (see sample program).

ZAHL DER MESSPUNKTE/KURVE	=	number of points in decimal
STARTADRESSE	=	hexadecimal start address
KANAL	=	multiplexor channel (0 ÷15)
SCHRITT	=	path
ZEICHEN	=	any printable ASCII character

Automatic scaling feature:

The maximum value of any curve is calculated and printed as a hexadecimal byte. A scalefactor is then calculated by which all points of the curve are divided, in order to remain below the value 64 (limitation given by the teletype carriage). This feature may be overridden by typing the desired scalefactor behind the question <OK? >. The program will then respond with the new scalefactor.

MEMORY REQUIREMENTS

195_{16} words

```
                    0001    REF ESMON,U$EDIT,U$RESP,U$VAL,U$MAXN,
                    0002    REF U$ORDO,U$PLOT,U$DEHE,U$HEDE
                    0003    DEF KAN,START,SCHR,TAE,ZDP,MAXX
                    0004    DEF ZDK,H1,SHFT,ZCHN,ERROR
0000                0005    DSECT
      003F          0006  TTY EQU X'3F'
      0000          0007  RY EQU 0
0000  0001          0008  ZDK DC 1
0001  000A          0009  ZDP DC 10
0002  0000          0010  TAE DC 0
0003  0000          0011  H1 DC 0
0004                0012  MAXX DS 7
000E                0013  KAN DS 7
0012                0014  START DS 7
0019                0015  SCHR DS 7
0020                0016  SHFT DS 7
0027                0017  ZCHN DS 7
002E  8D8A          0018  TEX1 TEXT '$8D$8APLOT=ROUTINE
002F  D0CC
0030  CFD4
0031  BDD2
0032  CFD5
0033  D4C9
0034  CEC5
0035  A0A0
0036  A0A0
0037  D6C5          0019    TEXT 'VERSION 72'
0038  D2D3
0039  C9CF
003A  CEA0
003E  E7E2
003C  0000          0020    DC 0
003D  8D8A          0021  TEX2 TEXT '$8D$8A$8A$8AZAHL DER
003E  8A8A
003F  DAC1
0040  C8CC
0041  A0C4
0042  C5D2
0043  A0A0
0044  CBD5          0022    TEXT 'KURVEN (MAX.7)   =  '
0045  D2D6
0046  C5CE
0047  A0A8
0048  CDC1
0049  D8AE
004A  B7A9
004B  A0A0
004C  A0BD
004D  A0A0
004E  0000          0023    DC 0
004F  A0B0          0024  TEX22 TEXT ' 0'
0050  8D8A          0025  TEX3 TEXT '$8D$8AZAHL DER MESS '
0051  DAC1
0052  C8CC
0053  A0C4
0054  C5D2
0055  A0CD
0056  C5D3
0057  D3A0
0058  D0D5          0026    TEXT 'PUNKTE/KURVE =  '
0059  CECB
005A  D4C5
005E  AFCB
```

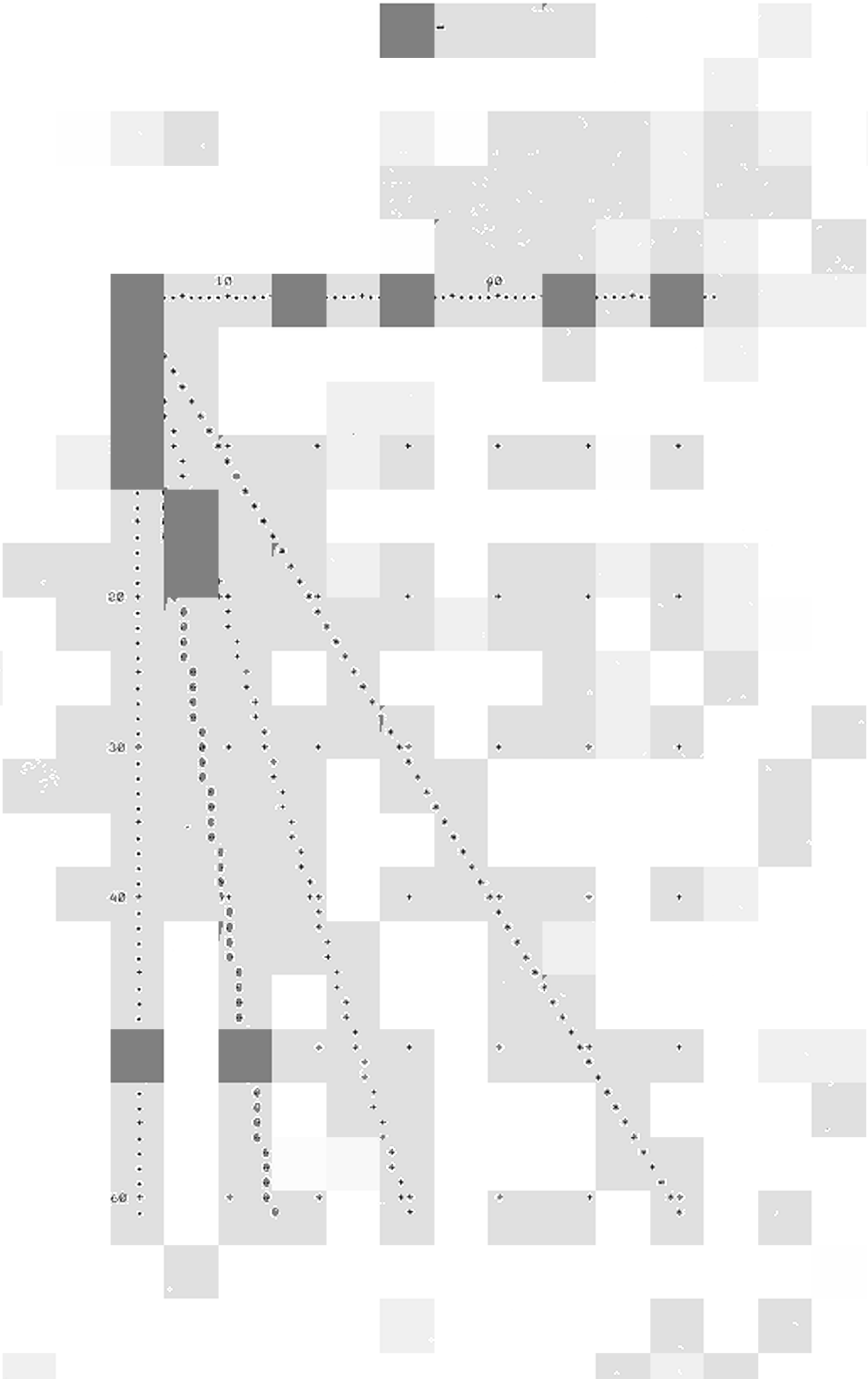
```
005C  D5D2
005D  D6C5
005E  A0ED
005F  A0A0
0060  0000      0027   DC 0
0061  8A8A      0028  TEX4 TEXT '$8A$8A'
0062  8D8A      0029  TEX41 TEXT '$8D$8A**** KURVE     '
0063  AAAA
0064  AAAA
0065  A0CE
0066  D5D2
0067  D6C5
0068  A0A0
0069  A0A0
006A  A0B0      0030  TEX42 TEXT ' 0 **** '
006E  A0AA
006C  AAAA
006D  AAA0
006E  0000      0031   DC 0
006F  8D8A      0032  TEX5 TEXT '$8D$8A$8ASTARTADRESSE  =   '
0070  8AD3
0071  D4C1
0072  D2D4
0073  C1C4
0074  D2C5
0075  D3D3
0076  C5A0
0077  A0BD
0078  A0A0
0079  A000      0033   DC X'A000'
007A  8D8A      0034  TEX6 TEXT '$8D$8AKANAL          =    '
007E  CEC1
007C  CEC1
007D  CCA0
007E  A0A0
007F  A0A0
0080  A0A0
0081  A0A0
0082  EDA0
0083  A0A0
0084  0000      0035   DC 0
0085  8D8A      0036  TEX7 TEXT '$8D$8ASCHRITT.       =    '
0086  D3C3
0087  C8D2
0088  C9D4
0089  D4A0
008A  A0A0
008B  A0A0
008C  A0A0
008D  EDA0
008E  A0A0
008F  0000      0037   DC 0
0090  8D8A      0038  TEX8 TEXT '$8D$8AZEICHEN        =    '
0091  DAC5
0092  C9C3
0093  C8C5
0094  CEA0
0095  A0A0
0096  A0A0
0097  A0A0
0098  EDA0
0099  A0A0
009A  A000      0039   DC X'A000'
009E  8D8A      0040  TEX9 TEXT '$8D$8AMAXIMALWERT    =    '
009C  CDC1
009D  D8C9
009E  CDC1
009F  CCD7
```

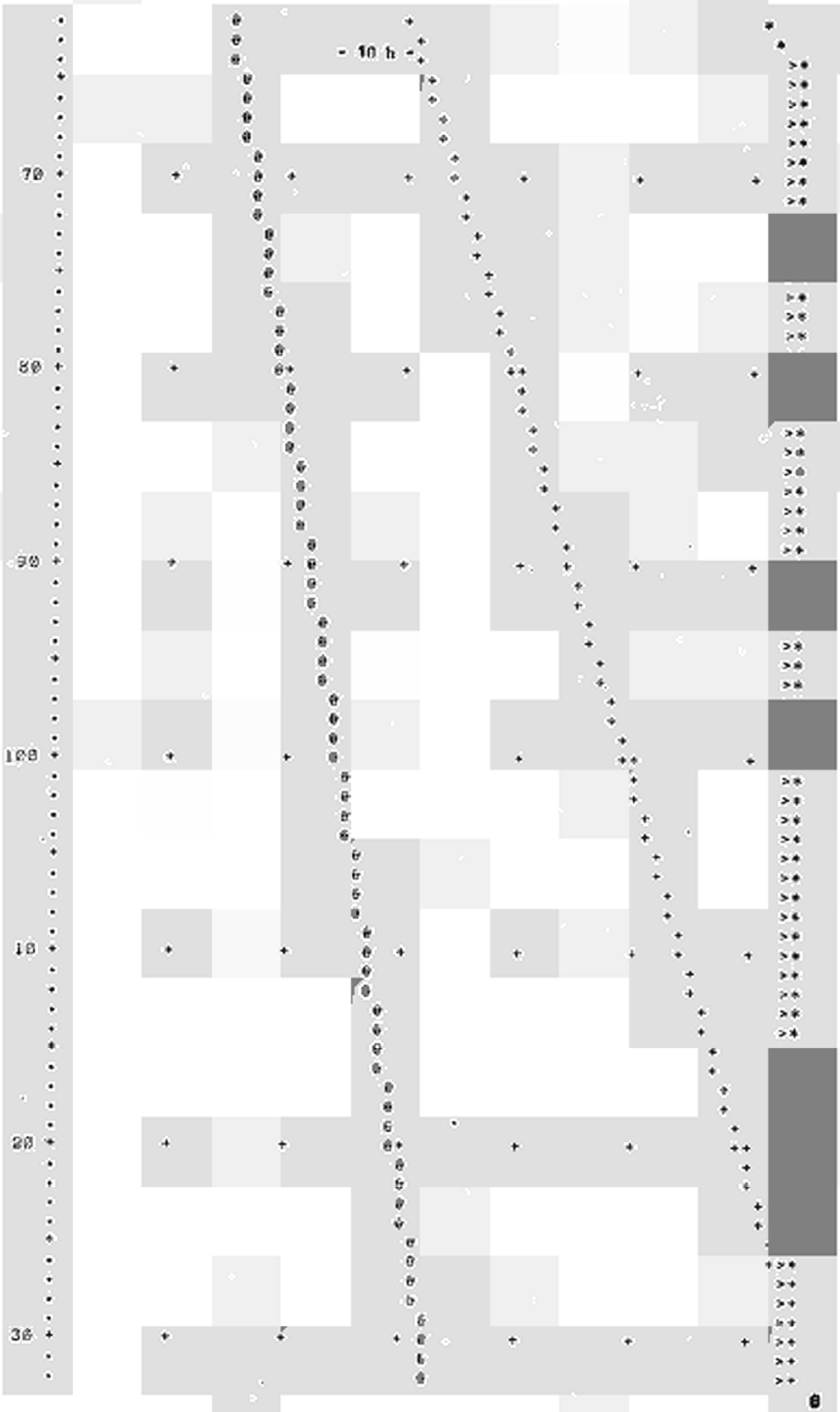
```
00A0  C5D2
00A1  D4A0
00A2  A0A0
00A3  EDA0
00A4  A0A0
00A5  A000      0041   DC X'A000'
00A6  8D8A      0042  TEX10 TEXT '$8D$8A'
00A7  D9AD      0043  TEX43 TEXT 'Y-SKALA  *'
00A8  D3CB
00A9  C1CC
00AA  C1A0
00AB  A0AA
00AC  0000      0044   DC 0
00AD  A0A0      0045  TEX11 TEXT '  OK? '
00AE  CFCB
00AF  BFA0
00B0  0000      0046   DC 0
00B1  2080      0047  FEHL WRITE CC$,ERT
00B2  0000 X
00B3  80E4
00B4  0017      0048  ERT DC 23,0
00B5  0000
00B6  8AC9      0049   TEXT '$8AINVALID DECIMAL-NUMBER$8D'
00B7  CED6
00B8  C1CC
00B9  C9C4
00BA  A0C4
00BB  C5C3
00BC  C9CD
00BD  C1CC
00BE  ADCE
00BF  D5CD
00C0  C2C5
00C1  D28D
      00C2      0050  DLENG EQU $-$$
                0051  *
0000            0052   PSECT
0000            0053   DS DLENG
00C2  0402      0054  VORSP INH
00C3  01D5      0055   LDV D,$$
00C4  0000 D
00C5  6494      0056  ANF1 JSR U$EDIT
00C6  002E D    0057   DC TEX1
00C7  6492      0058  ANF2 JSR U$EDIT
00C8  003E D    0059   DC TEX2
00C9  0155      0060   LDV Y,' 0'
00CA  A0B0
00CB  D85F      0061   STR Y,TEX42
00CC  006A
00CD  648D      0062   JSR U$RESP
00CE  73F8      0063   JMP ANF2
00CF  73F7      0064   JMP ANF2
00D0  7005      0065   JMP SDW2
00D1  73F5      0066   JMP ANF2
00D2  0197      0067   ANDV E,7
00D3  0007
00D4  D880 X    0068   STR E,ZDK
00D5  7002      0069   JMP L3
00D6  C8A0 X    0070  SDW2 LDR C,ZDK
00D7  6484      0071   JSR U$VAL
                0072  *
00D8  6481      0073  L3 JSR U$EDIT
00D9  0050 D    0074   DC TEX3
00DA  6480      0075   JSR U$RESP
00DB  73EE      0076   JMP ANF2
00DC  73EA      0077   JMP ANF2
00DD  7004      0078   JMP SDW3
00DE  73F9      0079   JMP L3
```

```
00DF  647D       0080   JSR U$LEHE
00E0  D881 X     0081   STR E,ZLP
00E1  7003       0082   JMP L4
00E2  C8A1 X     0083  SDW3 LDR C,ZLP
00E3  647A       0084   JSR U$HEDE
00E4  6477       0085   JSR U$VAL
                 0086  *
00E5  0620       0087  L4 ZERO X
00E6  F81F       0088  L41 INCM TEX42
00E7  006A
00E8  6471       0089  L42 JSR U$EDIT
00E9  0061 D     0090   DC TEX4
                 0091  *
00EA  646F       0092  L5 JSR U$EDIT
00EB  006F D     0093   DC TEX5
00EC  646E       0094   JSR U$RESP
00ED  73D9       0095   JMP ANF2
00EE  73F9       0096   JMP L42
00EF  73FA       0097   JMP L5
00F0  73F9       0098   JMP L5
00F1  D992 X     0099   STR E,START,X
00F2  7002       0100   JMP L6
00F3  C9E2 X     0101   LDR C,START,X
00F4  6467       0102   JSR U$VAL
                 0103  *
00F5  6464       0104  L6 JSR U$EDIT
00F6  007A D     0105   DC TEX6
00F7  6463       0106   JSR U$RESP
00F8  73CE       0107   JMP ANF2
00F9  73F0       0108   JMP L5
00FA  73FA       0109   JMP L6
00FB  73F9       0110   JMP L6
00FC  6460       0111   JSR U$LEHE
00FD  D98E X     0112   STR E,KAN,X
00FE  7003       0113   JMP L7
00FF  C9AE X     0114   LDR C,KAN,X
0100  645D       0115   JSR U$HEDE
0101  645A       0116   JSR U$VAL
                 0117  *
0102  6457       0118  L7 JSR U$EDIT
0103  0085 D     0119   DC TEX7
0104  6456       0120   JSR U$RESP
0105  73C1       0121   JMP ANF2
0106  73EE       0122   JMP L6
0107  73FA       0123   JMP L7
0108  73F9       0124   JMP L7
0109  6453       0125   JSR U$LEHE
010A  D999 X     0126   STR B,SCHR,X
010B  7003       0127   JMP L8
010C  C9E9 X     0128   LDR C,SCHR,X
010D  6450       0129   JSR U$HEDE
010E  644D       0130   JSR U$VAL
                 0131  *
010F  644A       0132  L8 JSR U$EDIT
0110  0090 D     0133   DC TEX8
0111  10FF       0134   TEST KY,TTY
0112  73FE       0135   JMP $-1
0113  18EF       0136   DTIR A,TTY
0114  0106       0137   SUBVC A,' '
0115  00A0
0116  2C03       0138   SKZ SDW8
0117  D91F       0139   STR A,ZCHN,X
0118  0027 X
0119  7005       0140   JMP L9
011A  C91F       0141  SDW8 LDR A,ZCHN,X
011B  0027 X
011C  9803 X     0142   STBY A,H1
011D  643C       0143   JSR U$EDIT
011E  0003 D     0144   DC H1
                 0145  *
```

```
011F  643A      0146  L9 JSR U$EDIT
0120  009E D    0147   DC TEX9
0121  643D      0148   JSR U$MAXN
0122  C9A4 X    0149   LDR C,MAXX,X
0123  6438      0150   JSR U$VAL
                0151  *
0124  6435      0152  L10 JSR U$EDIT
0125  00A6 D    0153   DC TEX10
0126  C984 X    0154   LDR B,MAXX,X
0127  0155      0155   LDV Y,-1
0128  FFFF
0129  01B5      0156   LDV C,1
012A  0001
012B  0187      0157   ANDVC B,X'80'
012C  0080
012D  2404      0158   SKN LL1
012E  0187      0159   ANDVC B,X'40'
012F  0040
0130  2403      0160   SKN LL2
0131  7004      0161   JMP LL3
0132  074E      0162  LL1 INCR Y
0133  0DB9      0163   ADD C,C
0134  074E      0164  LL2 INCR Y
0135  0DB9      0165   ADD C,C
0136  D95F      0166  LL3 STR Y,SHFT,X
0137  0020 X
0138  6423      0167   JSR U$VAL
0139  6420      0168  L11 JSR U$EDIT
013A  00AD D    0169   DC TEX11
013B  641F      0170   JSR U$RESP
013C  738A      0171   JMP ANF2
013D  73E1      0172   JMP L9
013E  7015      0173   JMP ORD
013F  73E4      0174   JMP L10
0140  0660      0175   ZERO Z
0141  01B5      0176   LDV C,1
0142  0001
0143  0290      0177  LL4 SRA E,1
0144  2408      0178   SKN LL5
0145  D97F      0179   STR Z,SHFT,X
0146  0020 X
0147  F95F      0180   DECM SHFT,X
0148  0020 X
0149  7006      0181   JMP LL6
014A  6415      0182  ERROR IOS FEHL
014E  80E1
014C  737A      0183   JMP ANF2
014D  076E      0184  LL5 INCR Z
014E  03AE      0185   SRC C,15
014F  73F3      0186   JMP LL4
0150  6409      0187  LL6 JSR U$EDIT
0151  00A6 D    0188   DC TEX10
0152  6409      0189   JSR U$VAL
0153  73E5      0190   JMP L11
0154  072E      0191  ORD INCR X
0155  E820 X    0192   CMR X,ZDK
0156  278F      0193   SKM L41
0157  6409      0194   JSR U$ORDO
0158  6409      0195   JSR U$PLOT
0159  6409      0196   JSR E$MON
      00C2 P    0197   END VORSP

015A  0000 X
015B  0000 X
015C  0000 X
015D  0000 X
015E  0000 X
015F  0000 X
0160  0000 X
0161  0000 X
0162  0000 X
0163  0000 X
NO  ERRORS
 DBOS CC
?$SI=TY
```

```
  DBOS CC
?$GRAPHD,BUS@1E00@1B00
1A00/1A48M
1A00=0001 0203 0405 0607 0809 0A0B 0C0D 0E0F
1A08=1011 1213 1415 1617 1819 1A1B 1C1D 1E1F
1A10=2021 2223 2425 2627 2829 2A2B 2C2D 2E2F
1A18=3031 3233 3435 3637 3839 3A3B 3C3D 3E3F
1A20=4041 4243 4445 4647 4849 4A4B 4C4D 4E4F
1A28=5051 5253 5455 5657 5859 5A5B 5C5D 5E5F
1A30=6061 6263 6465 6667 6869 6A6B 6C6D 6E6F
1A38=7071 7273 7475 7677 7879 7A7B 7C7D 7E7F
1A40=8081 8283 8485 8687 8889 8A8B 8C8D 8E8F
1A48=2603
B47/C2K,A
0C09
C09
0C09=0402
C09G

PLOT=ROUTINE    VERSION 72


ZAHL DER  KURVEN (MAX.7)   =  3
ZAHL DER MESS PUNKTE/KURVE =  132


**** KURVE     1 ****

STARTADRESSE  =    1A00
KANAL         =    0
SCHRITT       =    1
ZEICHEN       =     *
MAXIMALWERT   =     0083
Y-SKALA  *0004  OK? 1
Y-SKALA  *0001  OK?


**** KURVE     2 ****

STARTADRESSE  =    1A00
KANAL         =    0
SCHRITT       =    1
ZEICHEN       =     +
MAXIMALWERT   =     0083
Y-SKALA  *0004  OK? 2
Y-SKALA  *0002  OK?


**** KURVE     3 ****

STARTADRESSE  =    1A00
KANAL         =    0
SCHRITT       =    1
ZEICHEN       =     @
MAXIMALWERT   =     0083
Y-SKALA  *0004  OK?


**** KURVE   1   Y-SKALA *0001
++++ KURVE   2   Y-SKALA *0002
@@@@ KURVE   3   Y-SKALA *0004
```


10
20
30
40
60


- 10 h -
70
50
30
100
10
20
30
0

```
 $GRAPHD
PLOT=ROUTINE     VERSION 72


ZAHL DER  KURVEN (MAX.7)    =   A
ZAHL DER MESS PUNKTE/KURVE =   A

 INVALID DECIMAL-NUMBER


ZAHL DER  KURVEN (MAX.7)    =   2
ZAHL DER MESS PUNKTE/KURVE =   10


**** KURVE      1 ****

STARTADRESSE  =    1A00
KANAL         =
KANAL         =    A

 INVALID DECIMAL-NUMBER


ZAHL DER  KURVEN (MAX.7)    =   2
ZAHL DER MESS PUNKTE/KURVE =   10


**** KURVE      1 ****

STARTADRESSE  =    1A00
KANAL         =    2
SCHRITT       =    B

 INVALID DECIMAL-NUMBER


ZAHL DER  KURVEN (MAX.7)    =   2
ZAHL DER MESS PUNKTE/KURVE =   10


**** KURVE      1 ****

STARTADRESSE  =    1A00
KANAL         =    2
SCHRITT       =    8
ZEICHEN       =     .
MAXIMALWERT   =     004A
Y-SKALA   *0002   OK?


**** KURVE      2 ****

STARTADRESSE  =
```

DBOS CC

3.3 USBYDC

FEATURES:

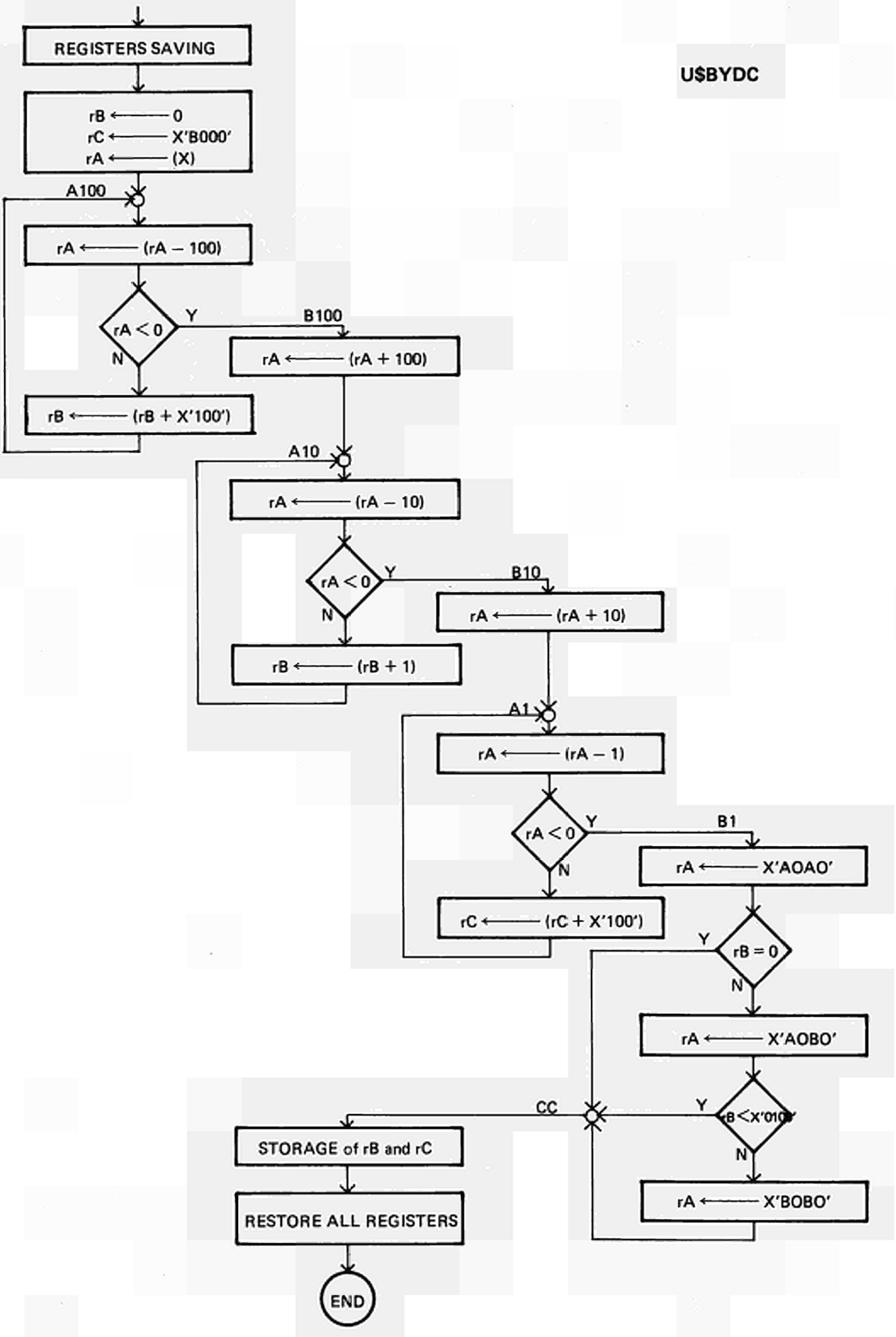
- This routine converts one byte from hexadecimal into decimal.
- The byte address must be contained in the X - Register (byte-mode addressing)
- A three digit decimal number in ASCII code is transmitted to the calling routine occupying 2 words, the first byte of which contains an ASCII blank, this simplifying printout.

CALLING SEQUENCE:

```
REF     USBYDC
:
LDV     X, ADDR          ADDR = byte address
JSR     USBYDC
DS      2                reserve two locations for ASCII
Return from subroutine   result
```

MEMORY REQUIREMENTS

37_{16} words


U$BYDC
REGISTERS SAVING
rB ← 0
rC ← X'B000'
rA ← (X)
A100
rA ← (rA – 100)
rA < 0
Y
N
B100
rA ← (rA + 100)
rB ← (rB + X'100')
A10
rA ← (rA – 10)
rA < 0
Y
N
B10
rA ← (rA + 10)
rB ← (rB + 1)
A1
rA ← (rA – 1)
rA < 0
Y
N
B1
rA ← X'AOAO'
rC ← (rC + X'100')
rB = 0
Y
N
rA ← X'AOBO'
CC
rB < X'0100'
Y
N
rA ← X'BOBO'
STORAGE of rB and rC
RESTORE ALL REGISTERS
END

```
 $SI=PR
?$A16
                        0001   DEF U$BYDC
       0000             0002  U$BYDC PSECT
       0000  F4C0       0003   ENT 9
       0001  C020
       0002  09D5
       0003  0139
       0004  0009
       0005  D020
       0006  05E1
       0007  C821       0004   LDR X,1,,1
       0008  0F75       0005   RTR Z,E
       0009  0177       0006   ANDV Z,X'7FFF'
       000A  7FFF
       000B  0680       0007   ZERO E
       000C  01E5       0008   LDV C,X'E0'
       000D  00E0
       000E  8100       0009   LLEY A,0,X
       000F  0116       0010  A100 SUBV A,100
       0010  0064
       0011  2603       0011   SKM B100
       0012  0199       0012   ADDV B,X'100'
       0013  0100
       0014  73FA       0013   JMP A100
       0015  0119       0014  B100 ADDV A,100
       0016  0064
       0017  0116       0015  A10 SUBV A,10
       0018  000A
       0019  2602       0016   SKM B10
       001A  078E       0017   INCR B
       001B  73FB       0018   JMP A10
       001C  0119       0019  B10 ADDV A,10
       001D  000A
       001E  0702       0020  A1 DECR A
       001F  2602       0021   SKM B1
       0020  C7AE       0022   INCR C
       0021  73FC       0023   JMP A1
       0022  0115       0024  B1 LDV A,X'A0A0'
       0023  A0A0
       0024  0C95       0025   RTR B,B
       0025  2C07       0026   SKZ CC
       0026  0115       0027   LDV A,X'A0E0'
       0027  A0E0
       0028  0186       0028   SUBVC B,X'0100'
       0029  0100
       002A  2602       0029   SKM CC
       002B  0115       0030  B0 LDV A,X'B0B0'
       002C  B0B0
       002D  089D       0031  CC OR B,A
       002E  06A4       0032   EXEY C
       002F  D380       0033   STR B,0,Z
       0030  D3A1       0034   STR C,1,Z
       0031  0402       0035   INH
       0032  D0C0       0036   STR D,0
       0033  F480       0037   LARS *0
       0034  01F9       0038   ADDV B,2
       0035  0002
       0036  05E3       0039   RTRN B
                        0040   END
 NO  ERRORS
 DEOS CC
?$EOD
?
```

3.4 U$DEHE

FEATURES:

- Converts a 4 digit decimal number into hexadecimal equivalent
- The number to be converted must be contained in BCD-code in the B-register
- After conversion, the hexadecimal equivalent is contained in the B-register
- An error return address ERROR is provided for the case where the number to be converted is not in BCD-code

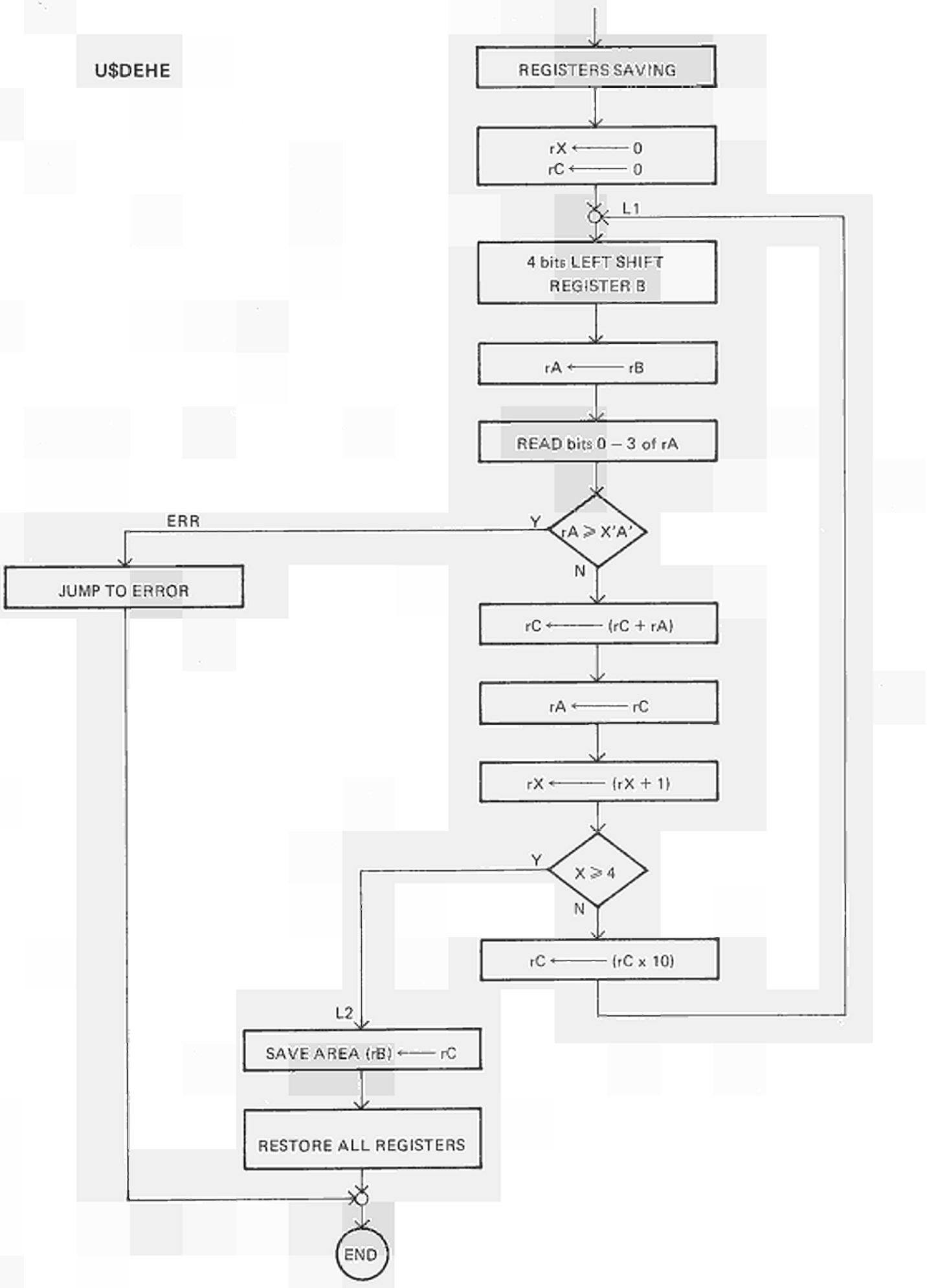
CALLING SEQUENCE:

```
REF     U$DEHE
DEF     ERROR       define linkage for error routine
  :
  :
LDR     B, ADDR     ADDR = address of BCD number
JSR     U$DEHE
return from subroutine
```

MEMORY REQUIREMENTS

24_{16} words

U$DEHE


REGISTERS SAVING
rX ⟵ 0
rC ⟵ 0
L1
4 bits LEFT SHIFT REGISTER B
rA ⟵ rB
READ bits 0 – 3 of rA
rA ⩾ X'A'
Y
ERR
JUMP TO ERROR
N
rC ⟵ (rC + rA)
rA ⟵ rC
rX ⟵ (rX + 1)
X ⩾ 4
Y
N
rC ⟵ (rC x 10)
L2
SAVE AREA (rB) ⟵ rC
RESTORE ALL REGISTERS
END

```
  DBOS CC
?$COPY,PR,PP
?
 $JOB
?$SI=PR
?$A16
```

```
                                0001    DEF U$DEHE
                                0002    REF ERROR
          0000                  0003  U$DEHE PSECT
          0000  F4C0            0004    ENT 9
          0001  C020
          0002  09D5
          0003  0139
          0004  0009
          0005  D020
          0006  05E1
          0007  0620            0005    ZERO X
          0008  06A0            0006    ZERO C
          0009  038B            0007  L1 SRC B,12
          000A  0C15            0008    RTR A,B
          000B  0117            0009    ANDV A,X'000F'
          000C  000F
          000D  0106            0010    SUBVC A,X'000A'
          000E  000A
          000F  2E10            0011    SKP ERR
          0010  08B9            0012    ADD C,A
          0011  0D15            0013    RTR A,C
          0012  072E            0014    INCR X
          0013  0126            0015    SUBVC X,4
          0014  0004
          0015  2E05            0016    SKP L2
          0016  0DB9            0017    ADD C,C
          0017  0DB9            0018    ADD C,C
          0018  08B9            0019    ADD C,A
          0019  0DB9            0020    ADD C,C
          001A  73EE            0021    JMP L1
          001B  0402            0022  L2 INH
          001C  D8A4            0023    STR C,B,,1
          001D  D0C0            0024    STR D,0
          001E  F480            0025    LARS *0
          001F  05E3            0026    RTRN E
          0020  D0C0            0027  ERR STR D,0
          0021  F480            0028    LARS *0
          0022  7400            0029    JMP ERROR
                                0030    END
          0023  0000 X
 NO   ERRORS
  DBOS CC
?$EOD
?
```

3.5 U$EDIT

FEATURES:

- Types a string of ASCII characters on system teletype without making a call to IOS
- The characterstring to be output may be of any length, it is terminated by a byte containing binary zero.

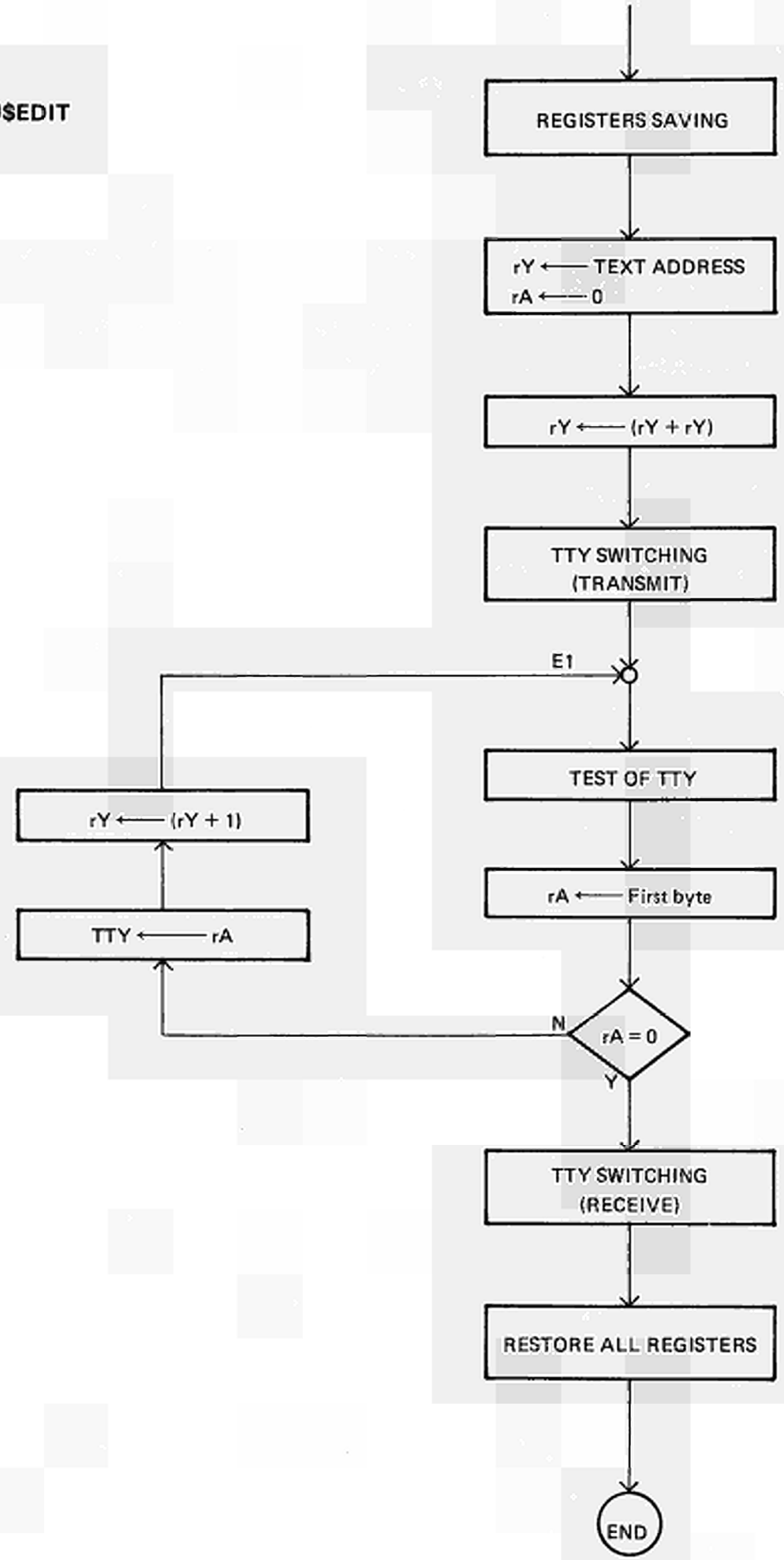
CALLING SEQUENCE

```
        REF     U$EDIT
         :
LABEL   TEXT    'text in ASCII'
        DC      0
         :
        JSR     U$EDIT
        DC      LABEL
        Return from subroutine
```

MEMORY REQUIREMENTS

22_{16} words

U$EDIT


REGISTERS SAVING
rY ⟵ TEXT ADDRESS
rA ⟵ 0
rY ⟵ (rY + rY)
TTY SWITCHING
(TRANSMIT)
E1
TEST OF TTY
rA ⟵ First byte
rA = 0
N
Y
rY ⟵ (rY + 1)
TTY ⟵ rA
TTY SWITCHING
(RECEIVE)
RESTORE ALL REGISTERS
END

```
 $SI=PR
?$A16
                                0001    DEF U$EDIT
        0000                    0002  U$EDIT PSECT
        0000    F4C0            0003    ENT 9
        0001    C020
        0002    09D5
        0003    0139
        0004    0009
        0005    D020
        0006    05E1
        0007    0F75            0004    RTR Z,E
        0008    0177            0005    ANDV Z,X'7FFF'
        0009    7FFF
        000A    C340            0006    LDR Y,0,Z
        000E    0115            0007    LDV A,X'20'
        000C    0020
        000D    187E            0008    DTOR A,X'3E'
        000E    0544            0009    DSPL Y
        000F    0A59            0010    ADD Y,Y
        0010    0600            0011    ZERO A
        0011    0402            0012    INH
        0012    103F            0013    CTRL 0,X'3F'
        0013    0402            0014  E1 INH
        0014    10FF            0015    TEST 0,X'3F'
        0015    73FE            0016    JMP $-1
        0016    8200            0017    LDBY A,0,Y
        0017    0815            0018    RTR A,A
        0018    2C04            0019    SKZ E2
        0019    187F            0020    DTOR A,X'3F'
        001A    05E1            0021    RISE E
        001B    074E            0022    INCR Y
        001C    73F6            0023    JMP E1
        001D    143F            0024  E2 CTRL 4,X'3F'
        001E    D0C0            0025    STR D,0
        001F    F480            0026    LARS *0
        0020    07EE            0027    INCR E
        0021    05E3            0028    RTRN E
                                0029    END
 NO   ERRORS
 DBOS CC
?$EOS-D
?
```

3.6 U$HEDE

FEATURES:

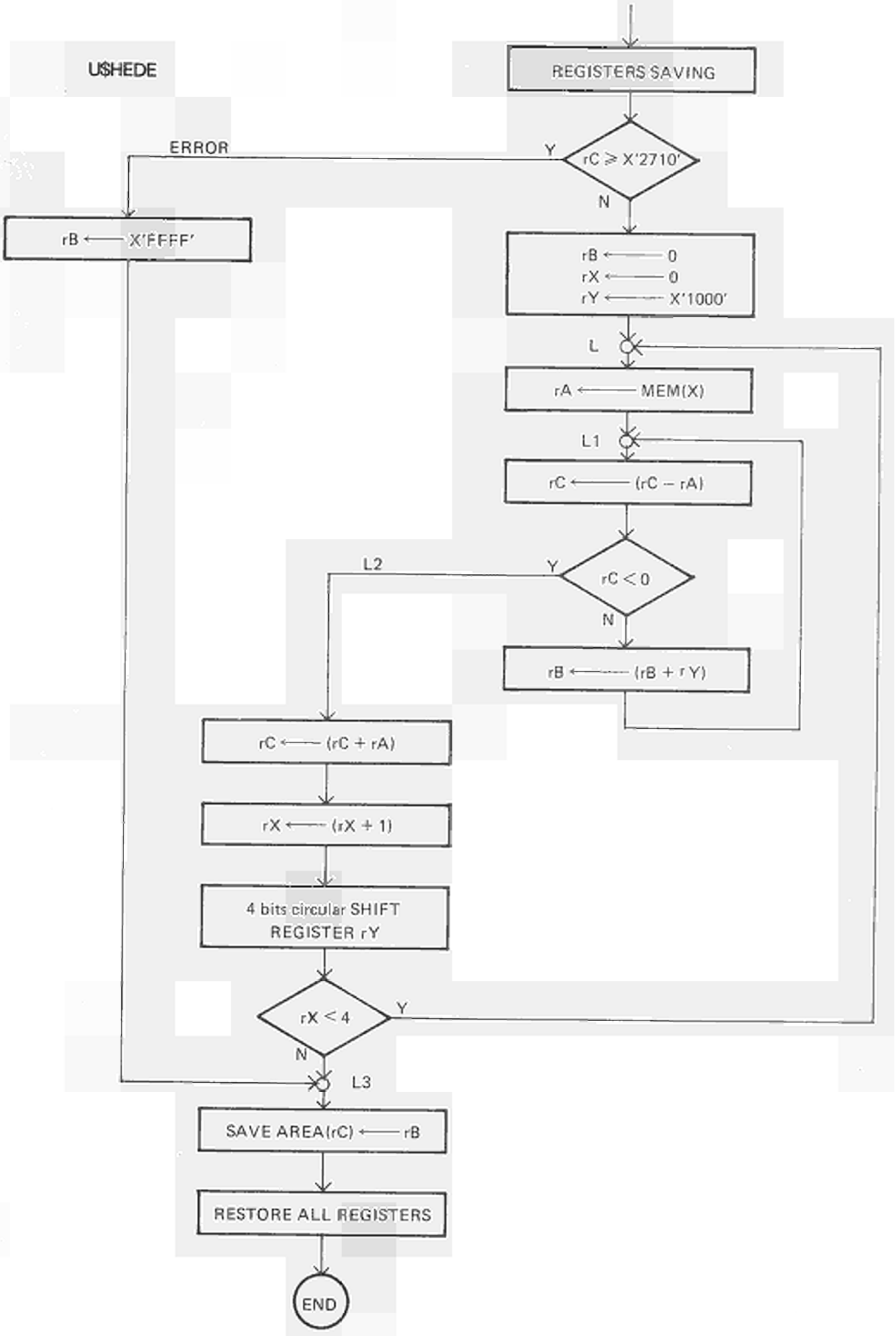
- Converts a 4 digit hexadecimal number (or a 16 bit binary word) into it's decimal equivalent in BCD code
- Number to be converted and BCD-results are stored in the C-register
- The maximum binary value to be converted is $270F_{16}$ (= 9999_{10}), higher values are returned with $FFFF_{16}$ in the C-register.

CALLING SEQUENCE

```
REF     U$HEDE
 :
 :
LDR     C, ADDR          ADDR = address of word to be converted
JSR     U$HEDE
Return from subroutine
```

MEMORY REQUIREMENTS

28_{16} words


U$HEDE
REGISTERS SAVING
rC ≥ X'2710'
Y
ERROR
N
rB ← X'FFFF'
rB ← 0
rX ← 0
rY ← X'1000'
L
rA ← MEM(X)
L1
rC ← (rC − rA)
rC < 0
Y
L2
N
rB ← (rB + r Y)
rC ← (rC + rA)
rX ← (rX + 1)
4 bits circular SHIFT
REGISTER rY
rX < 4
Y
N
L3
SAVE AREA(rC) ← rB
RESTORE ALL REGISTERS
END

```
                        0001    DEF U$HEDE
                        0002   *
0000                    0003   U$HEDE PSECT
0000  F4C0               0004    ENT 9
0001  C020
0002  09D5
0003  0139
0004  0009
0005  D020
0006  05E1
0007  01A6               0005    SUBVC C,X'2710'
0008  2710
0009  2E11               0006    SKP ERROR
000A  0680               0007    ZERO B
000B  0620               0008    ZERO X
000C  0155               0009    LDV Y,X 1000'
000D  1000
000E  C11F               0010   L LDR A,MEM,X
000F  0022 P
0010  08B6               0011   L1 SUB C,A
0011  2602               0012    SKM L2
0012  0A99               0013    ADD B,Y
0013  73FC               0014    JMP L1
0014  08B9               0015   L2 ADD C,A
0015  072E               0016    INCR X
0016  0343               0017    SRC Y,4
0017  0126               0018    SUBVC X,4
0018  0004
0019  27F4               0019    SKM L
001A  7002               0020    JMP L3
001B  0195               0021   ERROR LDV B,X'FFFF'
001C  FFFF
001D  0402               0022   L3 INH
001E  D885               0023    STR B,C,,1
001F  D0C0               0024    STR D,0
0020  F480               0025    LARS *0
0021  05E3               0026    RTRN E
                        0027   *
0022  03E8               0028   MEM DC X'3E8'
0023  0064               0029    DC X'64'
0024  000A               0030    DC X'A'
0025  0001               0031    DC 1
                        0032   *
                        0033    END

 NO   ERRORS
 DBOS CC
?$EOD
?
```

3.7 U$MAXN

FEATURES:

- Finds the largest value within a series of X vectors of bytes defined by

Register X		contains the vector number
START(X)	=	start address of vector X
KAN(X)	=	number of multiplexor channel corresponding to vector X
SCHR(X)	=	step of values to be taken in the vector X
ZDP	=	number of values to be taken from each vector
TAB	=	intermediary pointer
MAXX(X)	=	contains the result after return

- All parameters are transmitted to the subroutine by a DEF statement in the calling routine and via the X-register.

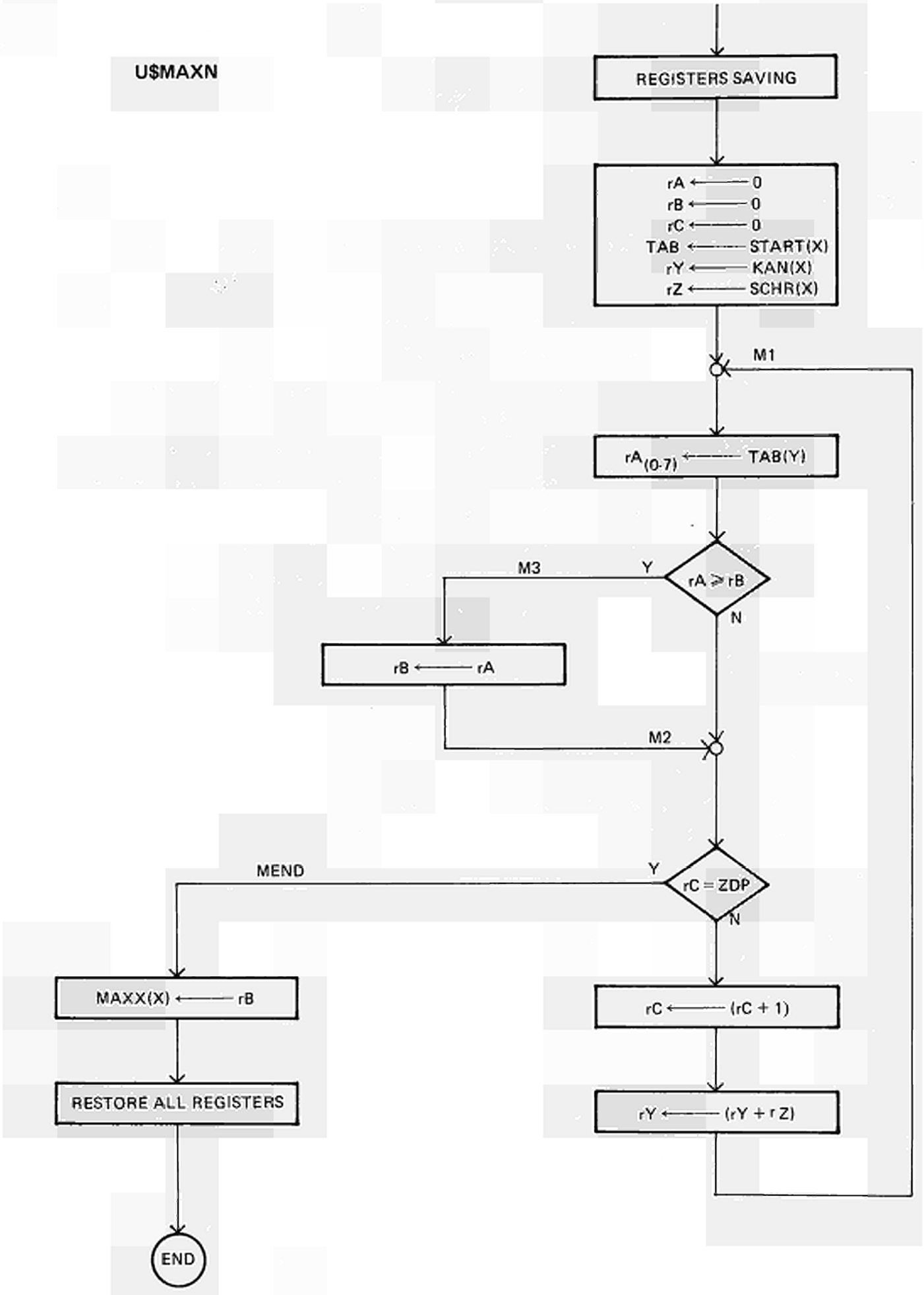
CALLING SEQUENCE

```
REF     U$MAXN
DEF     START, KAN, SCHR, ZDP, TAB, MAXX
 :
 :
LDV     X, number of vector to be analyzed
JSR     U$MAXN

Return from subroutine
```

MEMORY REQUIREMENTS

26_{16} words

U$MAXN


REGISTERS SAVING
rA ← 0
rB ← 0
rC ← 0
TAB ← START(X)
rY ← KAN(X)
rZ ← SCHR(X)
M1
rA(0-7) ← TAB(Y)
rA ≥ rB
Y
M3
N
rB ← rA
M2
rC = ZDP
Y
MEND
N
MAXX(X) ← rB
rC ← (rC + 1)
RESTORE ALL REGISTERS
rY ← (rY + rZ)
END

```
 $SI=PR
?$A16
                                0001    DEF U$MAXN
                                0002    REF KAN,START,SCHR,TAB,ZDP,MAXX
        0000                    0003  U$MAXN PSECT
        0000  F4C0              0004    ENT 9
        0001  C020
        0002  09D5
        0003  0139
        0004  0009
        0005  D020
        0006  05E1
        0007  C821              0005    LDR X,1,,1
        0008  0600              0006    ZERO A
        0009  0680              0007    ZERO B
        000A  06A0              0008    ZERO C
        000B  C15F              0009    LDR Y,KAN,X
        000C  0000 X
        000D  C17F              0010    LDR Z,START,X
        000E  0000 X
        000F  D07F              0011    STR Z,TAB
        0010  0000 X
        0011  C17F              0012    LDR Z,SCHR,X
        0012  0000 X
        0013  07AE              0013    INCR C
        0014  861F              0014  M1 LDBY A,*TAB,Y
        0015  0000 X
        0016  0C06              0015    SUBC A,B
        0017  2E06              0016    SKP M3
        0018  E0BF              0017  M2 CMR C,ZDP
        0019  0000 X
        001A  2C05              0018    SKZ MEND
        001B  07AE              0019    INCR C
        001C  0B59              0020    ADD Y,Z
        001D  73F6              0021    JMP M1
        001E  0895              0022  M3 RTR B,A
        001F  73F8              0023    JMP M2
        0020  D19F              0024  MEND STR B,MAXX,X
        0021  0000 X
        0022  0402              0025    INH
        0023  D0C0              0026    STR D,0
        0024  F480              0027    LARS *0
        0025  05E3              0028    RTRN E
                                0029    END
 NO  ERRORS
 DBOS CC
?$EOD
?
```

3.8 U$ORDO

FEATURES:

- This subroutine makes part of the PLOTTY processor and prepares before plotting the Y-axis a short summary of the curves to be plotted, indicating the choosen Y-scalefactors. For each curve to be plotted a line is typed with the format:

  sss KURVE n Y-SKALA * 000f

  where sss = selected plotting character
  n = number of curve
  f = Y-scalefactor

  (see also sample program)

  A short commentary may be added by the operator (e.g. typing < TEMPERATURE >, < PRESSURE >, ... etc.), in order to ease the identification of various curves. Each line has to be transmitted typing a carriage return.
- U$ORDO calls two subroutines, U$EDIT and U$VAL.

CALLING SEQUENCE

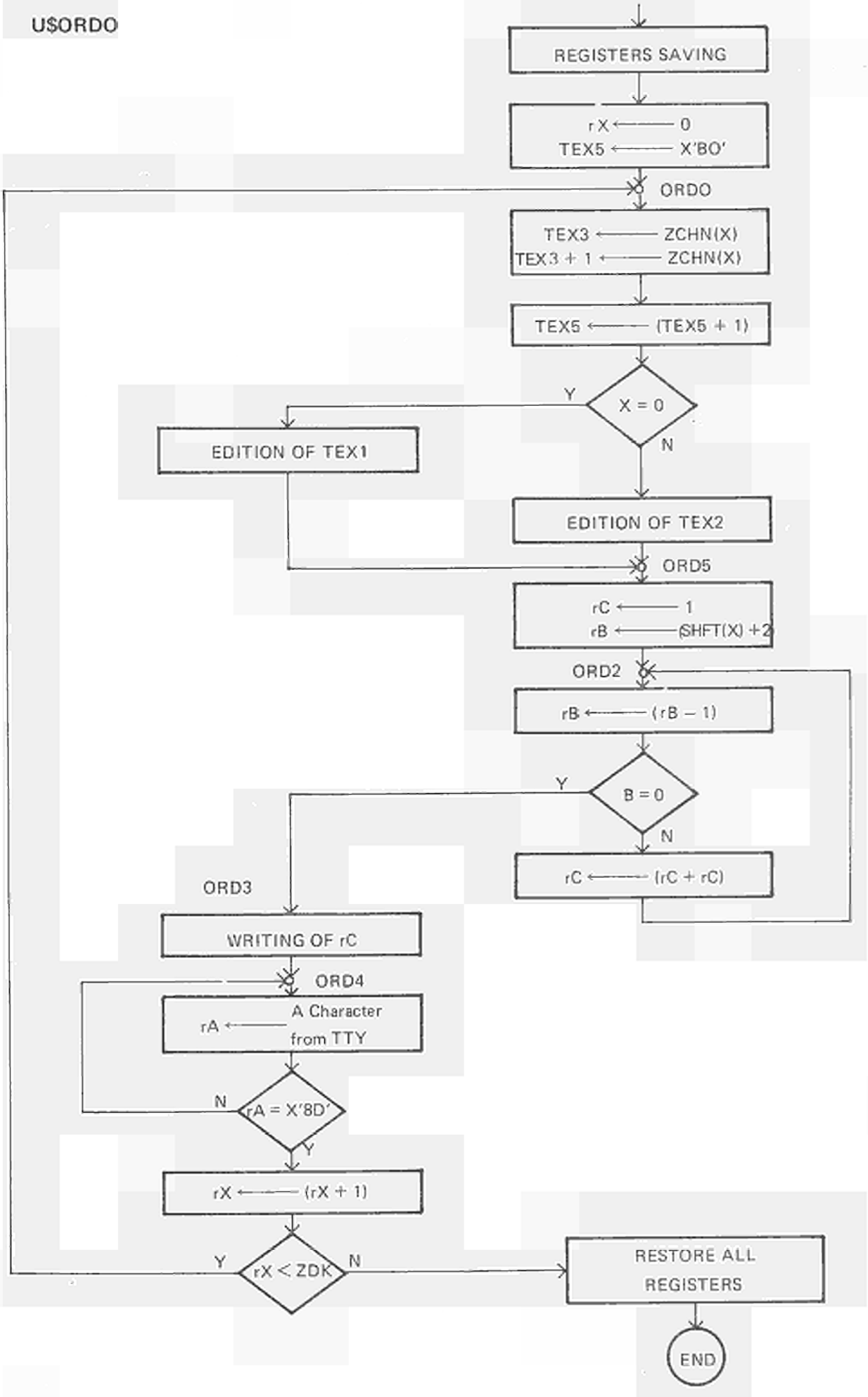
```
REF     U$ORDO
DEF     ZCHN, SHFT, ZDK
 :
ZDV     X, curve number
JSR     U$ORDO
```

Return from subroutine

MEMORY REQUIREMENTS

$4C_{16}$ words

U$ORDO


REGISTERS SAVING
rX ← 0
TEX5 ← X'BO'
ORDO
TEX3 ← ZCHN(X)
TEX3 + 1 ← ZCHN(X)
TEX5 ← (TEX5 + 1)
X = 0
Y
N
EDITION OF TEX1
EDITION OF TEX2
ORD5
rC ← 1
rB ← (SHFT(X) +2
ORD2
rB ← (rB – 1)
B = 0
Y
N
rC ← (rC + rC)
ORD3
WRITING OF rC
ORD4
rA ← A Character from TTY
rA = X'8D'
N
Y
rX ← (rX + 1)
rX < ZDK
Y
N
RESTORE ALL REGISTERS
END

```
?$SI=PR
?$A16

                        0001    DEF U$ORDO
                        0002    REF U$EDIT,U$VAL
                        0003    REF ZCHN,SHFT,ZDK
                        0004  *
      0000              0005  U$ORDO PSECT
      0000  F4C0        0006    ENT 9
      0001  C020
      0002  09D5
      0003  0139
      0004  0009
      0005  D020
      0006  05E1
      0007  0155        0007  ORDIN LDV Y,' 0'
      0008  A0F0
      0009  D05F        0008    STR Y,TEX5
      000A  0042 P
      000B  0620        0009    ZERO X
      000C  C15F        0010  ORD0 LDR Y,ZCHN,X
      000D  0000 X
      000E  D05F        0011    STR Y,TEX3
      000F  003C P
      0010  D05F        0012    STR Y,TEX3+1
      0011  003D P
      0012  905F        0013    STBY Y,TEX3
      0013  003C P
      0014  905F        0014    STBY Y,TEX3+1
      0015  003D P
      0016  F01F        0015    INCM TEX5
      0017  0042 P
      0018  0935        0016    RTR X,X
      0019  2403        0017    SKN ORD1
      001A  642F        0018    JSR U$EDIT
      001B  003A P      0019    DC TEX1
      001C  7002        0020    JMP ORD5
      001D  642C        0021  ORD1 JSR U$EDIT
      001E  003B P      0022    DC TEX2
      001F  01E5        0023  ORD5 LDV C,1
      0020  0001
      0021  C19F        0024    LDR B,SHFT,X
      0022  0000 X
      0023  0199        0025    ADDV B,2
      0024  0002
      0025  0782        0026  ORD2 DECR B
      0026  2C02        0027    SKZ ORD3
      0027  0DB9        0028    ADD C,C
      0028  73FC        0029    JMP ORD2
      0029  6421        0030  ORD3 JSR U$VAL
      002A  0402        0031    INH
      002B  10FF        0032  ORD4 TEST 0,X'3F'
      002C  73FE        0033    JMP $-1
      002D  18EF        0034    DTIR A,X'3F'
      002E  0106        0035    SUBVC A,X'8D'
      002F  008D
      0030  25FA        0036    SKN ORD4
      0031  05E1        0037    RISE E
      0032  072E        0038    INCR X
      0033  E03F        0039    CMR X,ZDK
      0034  0000 X
      0035  27D6        0040    SKM ORD0
      0036  0402        0041    INH
      0037  D0C0        0042    STR D,0
      0038  F480        0043    LARS *0
      0039  05E3        0044    RTRN E
                        0045  *
      003A  8A8A        0046  TEX1 TEXT '$8A$8A'
      003B  8D8A        0047  TEX2 TEXT '$8D$8A'
      003C              0048  TEX3 DS 2
      003D  A8CE        0049  TEX4 TEXT ' KURVE
      003E  D5D2
      003F  D6C5
      0040  A0A0
      0041  A0A0        0050  TEX5 TEXT ' 0   Y-SKALA *'
      0042  A0E0
      0043  A0A0
      0044  A0D9
      0045  ADD3
      0046  CEC1
      0047  CCC1
      0048  A0AA
      0049  0000        0051    DC 0
                        0052  *
                        0053    END
      004A  0000 X
      004B  0000 X
 NO  ERRORS
 DBOS CC
?$EOD
?
```

3.9 U$PLOT

FEATURES:

- This subroutines is part of the PLOTTV processor and draws the axis as well as the curves by constructing vectors named DRZL which form one print-line.
- The required arguments are transferred by means of DEF/REF-statements.

CALLING SEQUENCE

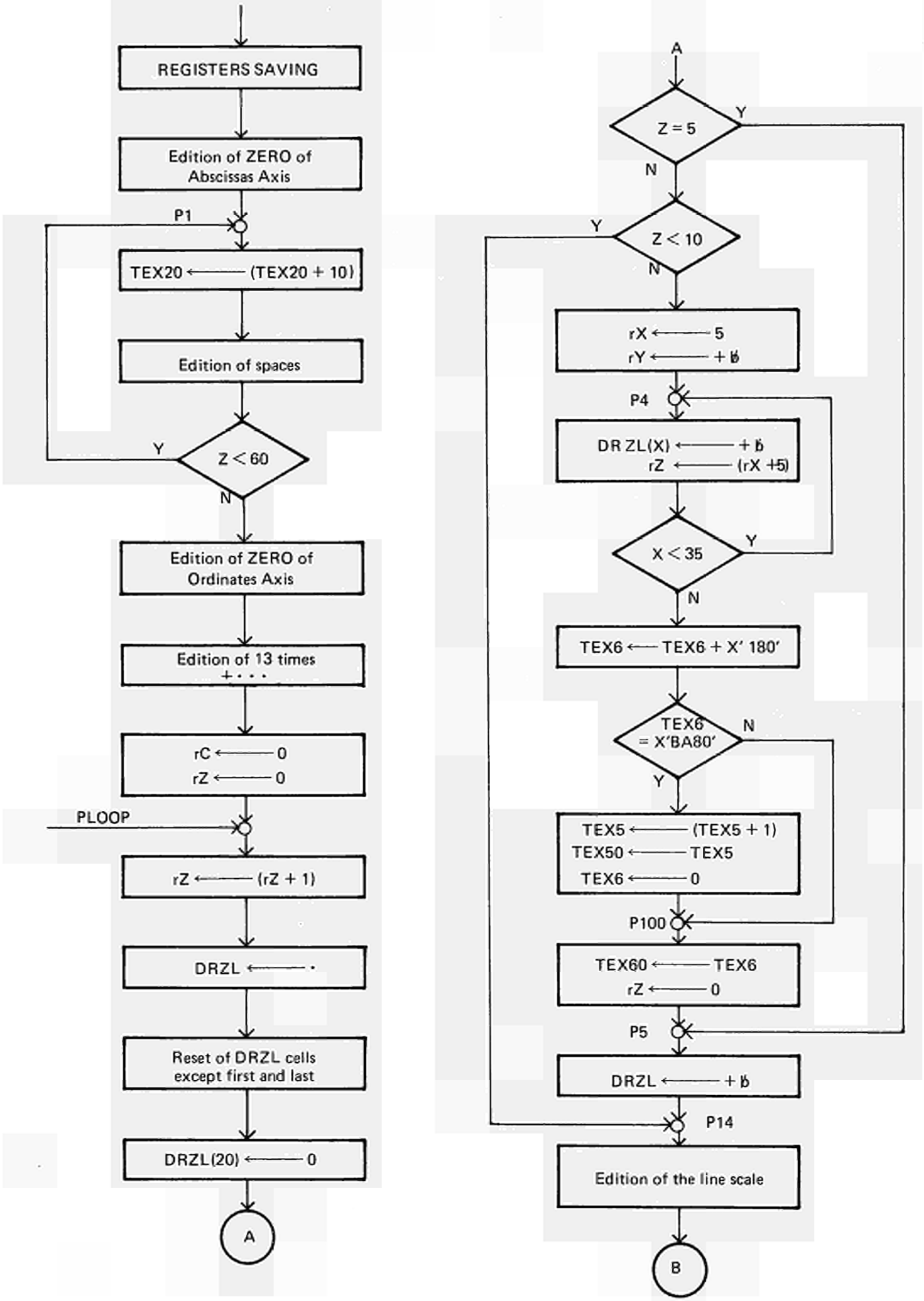
```
REF     U$PLOT
DEF     TAB, SCHR, SHFT, ZCHN
DEF     KAN, ZDK, ZDP, START
 :
JSR     U$PLOT
```

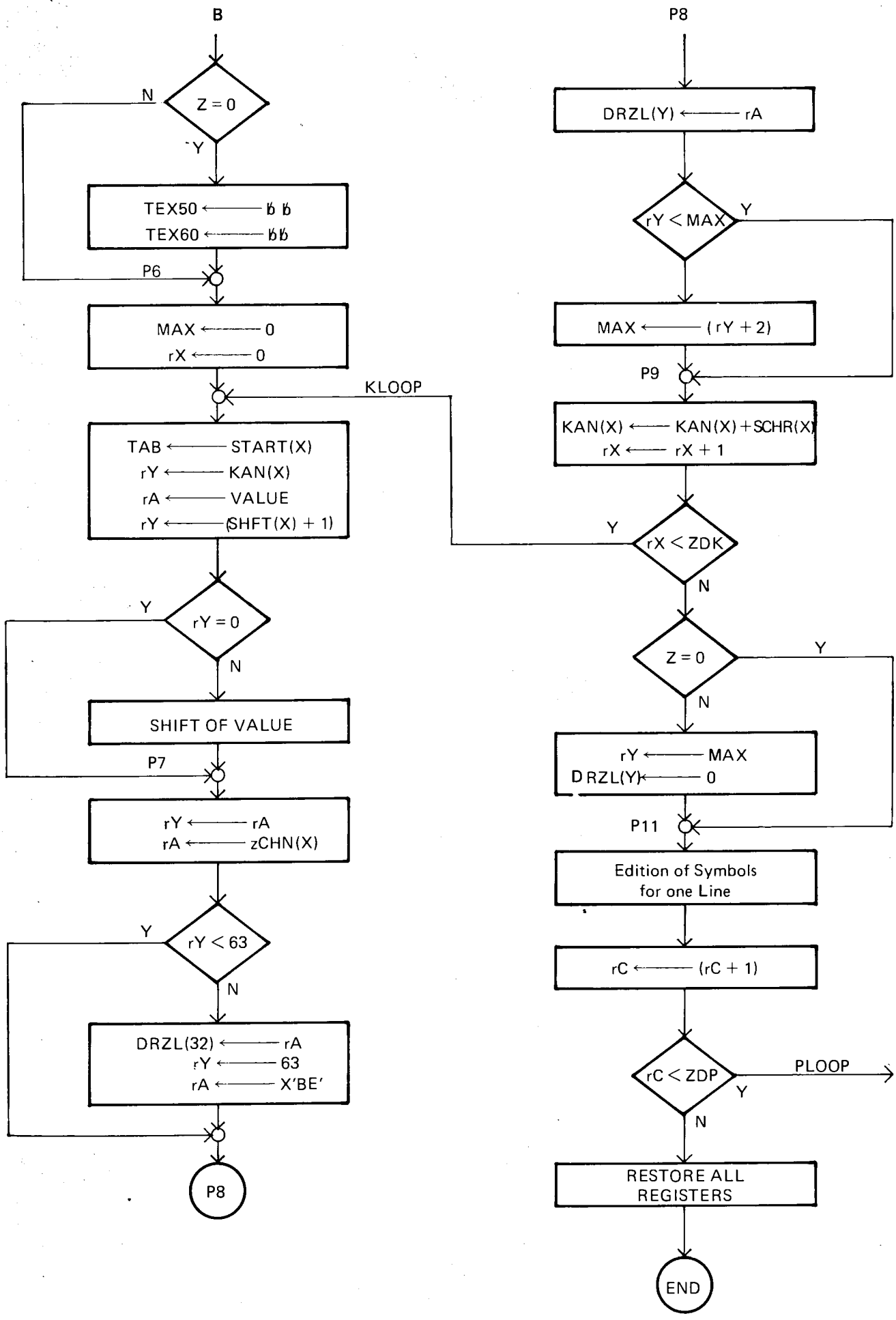
Return from subroutine

MEMORY REQUIREMENTS

$E8_{16}$ words

U$PLOT


REGISTERS SAVING
Edition of ZERO of Abscissas Axis
P1
TEX20 ← (TEX20 + 10)
Edition of spaces
Z < 60
Y
N
Edition of ZERO of Ordinates Axis
Edition of 13 times +. . .
rC ← 0
rZ ← 0
PLOOP
rZ ← (rZ + 1)
DRZL ← .
Reset of DRZL cells except first and last
DRZL(20) ← 0
A
A
Z = 5
Y
N
Z < 10
Y
N
rX ← 5
rY ← + b
P4
DR ZL(X) ← + b
rZ ← (rX +5)
X < 35
Y
N
TEX6 ← TEX6 + X' 180'
TEX6 = X'BA80'
N
Y
TEX5 ← (TEX5 + 1)
TEX50 ← TEX5
TEX6 ← 0
P100
TEX60 ← TEX6
rZ ← 0
P5
DRZL ← + b
P14
Edition of the line scale
B


B
Z = 0
N
Y
TEX50 ← ♭ ♭
TEX60 ← ♭♭
P6
MAX ← 0
rX ← 0
KLOOP
TAB ← START(X)
rY ← KAN(X)
rA ← VALUE
rY ← (SHFT(X) + 1)
Y
rY = 0
N
SHIFT OF VALUE
P7
rY ← rA
rA ← zCHN(X)
Y
rY < 63
N
DRZL(32) ← rA
rY ← 63
rA ← X'BE'
P8
P8
DRZL(Y) ← rA
rY < MAX
Y
MAX ← (rY + 2)
P9
KAN(X) ← KAN(X) + SCHR(X)
rX ← rX + 1
Y
rX < ZDK
N
Z = 0
Y
N
rY ← MAX
DRZL(Y) ← 0
P11
Edition of Symbols
for one Line
rC ← (rC + 1)
rC < ZDP
PLOOP
Y
N
RESTORE ALL
REGISTERS
END

```
 LEOS CC
?$JOB
?$SI=PR
?$A16
                          0001    LEF U$PLOT
                          0002    REF U$EDIT
                          0003    REF TAB,SCHR,SHFT,ZCHN,KAN
                          0004    REF ZDK,ZDP,STAR1
                          0005    *
       0000               0006    U$PLOT PSECT
       0000  F4C0         0007    ENT 9
       0001  C020
       0002  09D5
       0003  0139
       0004  C009
       0005  D020
       0006  05E1
                          0008    *GRADUATION DES ABSCISSES
       0007  64DF         0009    JSR U$EDIT
       0008  00CD P       0010    DC TEX1
       0009  C07F         0011    LER Z,TEX20
       000A  00D7 P
       000B  0179         0012    P1 ADDV Z,X'100'
       000C  0100
       000D  D07F         0013    STR Z,TEX20
       000E  00D7 P
       000F  64D6         0014    JSR U$EDIT
       0010  00D3 P       0015    DC TEX2
       0011  0166         0016    SUBVC Z,'60'
       0012  E6B0
       0013  27F7         0017    SKM P1
       0014  64D1         0018    JSR U$EDIT
       0015  00D9 P       0019    DC TEX3
       0016  0660         0020    ZERO Z
       0017  076E         0021    P2 INCR Z
       0018  64CD         0022    JSR U$EDIT
       0019  00DD P       0023    DC TEX30
       001A  0166         0024    SUBVC Z,X'L'
       001B  000D
       001C  27FA         0025    SKM P2
                          0026    *MISE A ZERO DU TEXTE
       001D  06A0         0027    ZERO C
       001E  0660         0028    ZERO Z
       001F  076E         0029    PLOOP INCR Z
       0020  0155         0030    LDV Y,'. '
       0021  AEA0
       0022  D05F         0031    STR Y,DRZL
       0023  00A9 P
       0024  0155         0032    LDV Y,1
       0025  0001
       0026  0115         0033    LDV A,'  '
       0027  A0A0
       0028  D21F         0034    P3 STR A,DRZL,Y
       0029  00A9 P
       002A  074E         0035    INCR Y
       002B  0146         0036    SUBVC Y,X'2C'
       002C  0020
       002D  27FA         0037    SKM P3
       002E  0600         0038    ZERO A
       002F  D21F         0039    STR A,DRZL,Y
       0030  00A9 P
                          0040    *GRADUATION LES ORDONNEES
       0031  0166         0041    SUBVC Z,5
       0032  0005
       0033  2C22         0042    SKZ P5
       0034  0166         0043    SUBVC Z,10
       0035  000A
       0036  2623         0044    SKM P10
       0037  0135         0045    LDV X,5
```

```
0038  0005
0039  0155       0046   LDV Y,12 h,-
003A  AEA0
003B  D15F       0047  P4 STR Y,DRZL,X
003C  00A9 P
003D  0139       0048   ADDV X,5
003E  0005
003F  0126       0049   SUBVC X,35
0040  0023
0041  27F9       0050   SKM P4
0042  C05F       0051   LDR Y,TEX6
0043  00E4 P
0044  0159       0052   ADDV Y,X'100'
0045  0100
0046  0146       0053   SUBVC Y,X'EAE0'
0047  EAE0
0048  2408       0054   SKN P100
0049  F01F       0055   INCM TEX5
004A  00E5 P
004B  C05F       0056   LDR Y,TEX5
004C  00E5 P
004D  D05F       0057   STR Y,TEX50
004E  00E1 P
004F  0155       0058   LDV Y,'00'
0050  E0E0
0051  D05F       0059  P100 STR Y,TEX6
0052  00E4 P
0053  D05F       0060   STR Y,TEX60
0054  00E2 P
0055  0660       0061   ZERO Z
0056  0155       0062  P5 LDV Y,'+ '
0057  AEA0
0058  D05F       0063   STR Y,DRZL
0059  00A9 P
005A  648E       0064  P10 JSR USEDIT
005B  00E0 P     0065   DC TEX4
005C  0E75       0066   RTR Z,Z
005D  2406       0067   SKN P6
005E  0155       0068   LDV Y,'
005F  A0A0
0060  D05F       0069   STR Y,TEX50
0061  00E1 P
0062  D05F       0070   STR Y,TEX60
0063  00E2 P
0064  0620       0071  P6 ZERO X
0065  D03F       0072   STR X,MAX
0066  00CB P
0067  C11F       0073  KLOOP LDR A,START,X
0068  0000 X
0069  547D       0074   STA TAB
006A  C19F       0075   LDR B,KAN,X
006B  0000 X
006C  0C55       0076   RTR Y,B
006D  0600       0077   ZERO A
006E  861F       0078   LDBY A,*TAB,Y
006F  0000 X
0070  C15F       0079   LDR Y,SHFT,X
0071  0000 X
0072  074E       0080   INCR Y
0073  2C06       0081   SKZ P7
0074  0742       0082   DECR Y
0075  0157       0083   ANDV Y,X'F'
0076  000F
0077  0159       0084   ADDV Y,X'210'
0078  0210
0079  0550       0085   XEC Y
007A  0855       0086  P7 RTR Y,A
007B  C11F       0087   LDR A,ZCHN,X
007C  0000 X
007D  0146       0088   SUBVC Y,63
007E  003F
007F  2606       0089   SKM P8
```

```
0080  901F      0090   STBY A,DRZL+32
0081  00C9 P
0082  0155      0091   LDV Y,63
0083  003F
0084  0115      0092   LDV A,X'BE'
0085  00BE
0086  921F      0093  P8 STBY A,DRZL,Y
0087  00A9 P
0088  E05F      0094   CMR Y,MAX
0089  00CE P
008A  2604      0095   SKM P9
008B  0159      0096   ADDV Y,2
008C  0002
008D  D05F      0097   STR Y,MAX
008E  00CE P
008F  C15F      0098  P9 LDR Y,SCHR,X
0090  0000 X
0091  0C59      0099   ADD Y,B
0092  D15F      0100   STR Y,KAN,X
0093  0000 X
0094  072E      0101   INCR X
0095  E03F      0102   CMR X,ZDK
0096  0000 X
0097  27CF      0103   SKM KLOOP
0098  0E75      0104   RTR Z,Z
0099  2C05      0105   SKZ P11
009A  C05F      0106   LDR Y,MAX
009B  00CE P
009C  0600      0107   ZERO A
009D  921F      0108   STBY A,DRZL,Y
009E  00A9 P
009F  6446      0109  P11 JSR U$EDIT
00A0  00A9 P    0110    DC DRZL
00A1  07AE      0111    INCR C
00A2  E0BF      0112    CMR C,ZDP
00A3  0000 X
00A4  277A      0113    SKM PLOOP
00A5  0402      0114    INH
00A6  D0C0      0115    STR D,0
00A7  F480      0116    LARS *0
00A8  05E3      0117    RTRN E
                0118  *
00A9            0119  DRZL DS 34
00CB  0000      0120  MAX DC 0
00CC            0121  MEM DS 1
00CD  8D8A      0122  TEX1 TEXT '$8D$8A$8A$8A     0'
00CE  8A8A
00CF  A0A0
00D0  A0A0
00D1  A0E0
00D2  0000      0123    DC 0
00D3  A0A0      0124  TEX2 TEXT '
00D4  A0A0
00D5  A0A0
00D6  A0A0
00D7  B0E0      0125  TEX20 TEXT '00'
00D8  0000      0126    DC 0
00D9  8D8A      0127  TEX3 TEXT '$8D$8A   0'
00DA  A0A0
00DB  A0E0
00DC  A000      0128    DC X'A000'
00DD  ABAE      0129  TEX30 TEXT '+...'
00DE  AEAE
00DF  AE00      0130    DC X'AE00'
00E0  8D8A      0131  TEX4 TEXT '$8D$8A'
00E1  A0A0      0132  TEX50 TEXT '  '
00E2  A0A0      0133  TEX60 TEXT '  '
00E3  A000      0134    DC X'A000'
00E4  B0B0      0135  TEX6 TEXT '00'
00E5  A0E0      0136  TEX5 TEXT ' 0'
                0137    END
00E6  0000 X
00E7  0000 X
NO  ERRORS
```

3.10 U$RESP

FEATURES:

- This routine inputs from system-teletype without making use of IOS up to 4 hexadecimal characters (0,1 ... 9, A, ... F)
- Input is terminated by a carriage return (CR)
- Input of more than 4 characters cancels the most left-hand character
- The input is decoded from ASCII to binary and stored in the B-register
- Return is the calling program is made by the help of some control characters, which permits a conditional branching
- The Y-register contains information whether there was an input (y = 0) or not (y = 1). This indication is then stored in location YY which is available to the calling program via a REF YY-statement.

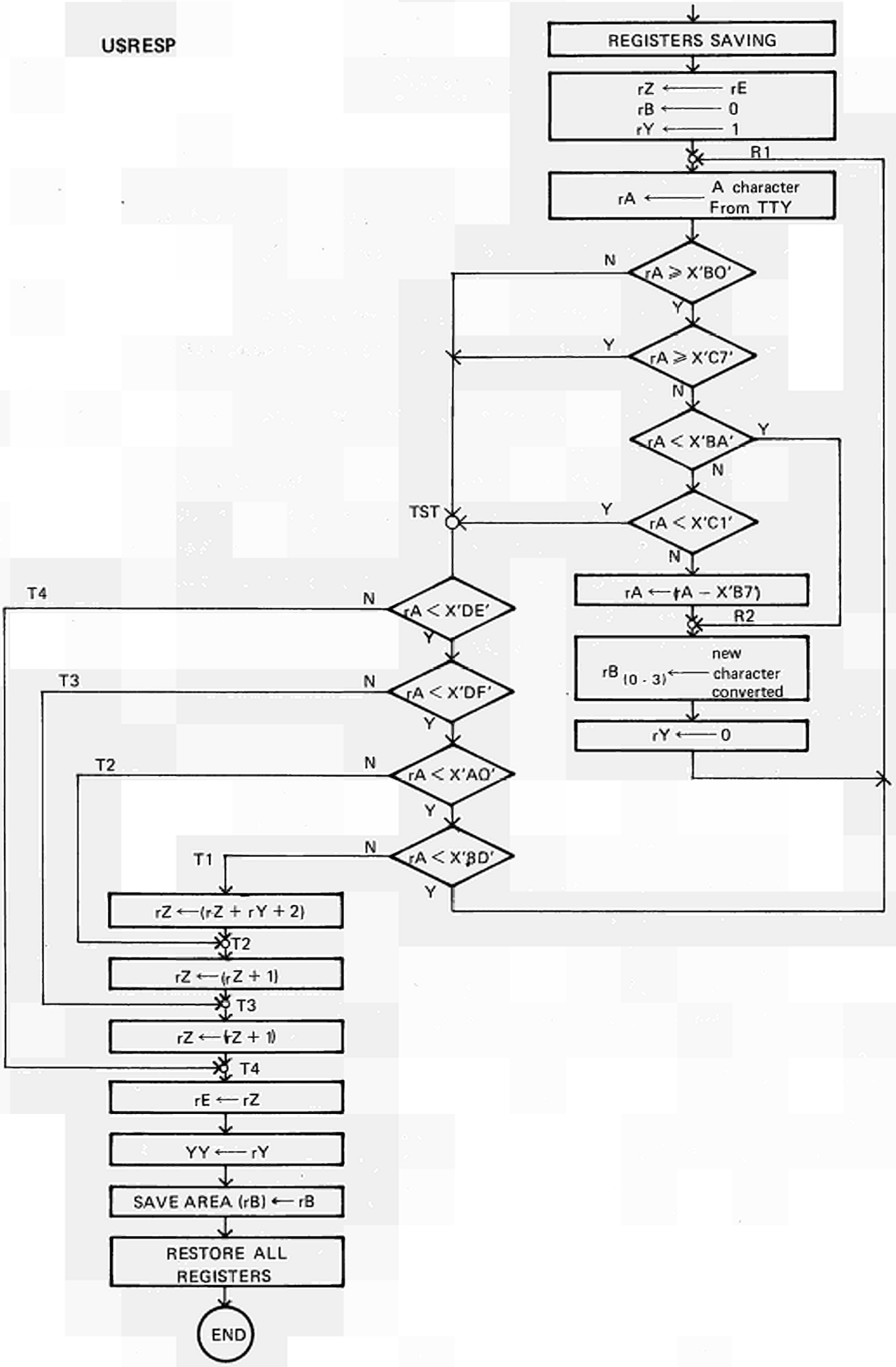
CALLING SEQUENCE

```
REF     U$RESP
REF     YY          optional
 :
JSR     U$RESP
1st  return after typing ↑
2nd  return after typing ←
3rd  return after typing blank
4th  return after typing CR without preceeding input
5th  return after input followed by CR
```

MEMORY REQUIREMENTS

41_{16} words

USRESP


REGISTERS SAVING
rZ ← rE
rB ← 0
rY ← 1
R1
rA ← A character From TTY
N
rA ≥ X'BO'
Y
Y
rA ≥ X'C7'
N
rA < X'BA'
Y
N
TST
Y
rA < X'C1'
N
rA ← (rA − X'B7)
R2
rB (0 - 3) ← new character converted
rY ← 0
T4
N
rA < X'DE'
Y
T3
N
rA < X'DF'
Y
T2
N
rA < X'AO'
Y
T1
N
rA < X'BD'
Y
rZ ← (rZ + rY + 2)
T2
rZ ← (rZ + 1)
T3
rZ ← (rZ + 1)
T4
rE ← rZ
YY ← rY
SAVE AREA (rB) ← rB
RESTORE ALL REGISTERS
END

```
$SI=PR
?$A16
                                  - 35 -
                                   DEF U$RESP,YY
0000                              U$RESP PSECT
0000   F4C0                        ENT 9
0001   C020
0002   09D5
0003   0139
0004   0009
0005   D020
0006   05E1
0007   0F75                        RTR Z,E
0008   0680                        ZERO B
0009   0155                        LDV Y,-1
000A   FFFF
000B   0402                       R1 INH
000C   10FF                        TEST 0,X'3F'
000D   73FE                        JMP $-1
000E   18FF                        DTIR A,X'3F'
000F   05E1                        RISE E
0010   0106                        SUBVC A,X'B0'
0011   00E0
0012   2613                        SKM TST
0013   0106                        SUBVC A,X'C7'
0014   00C7
0015   2E10                        SKP TST
0016   0106                        SUBVC A,X'BA'
0017   00BA
0018   2605                        SKM R2
0019   0106                        SUBVC A,X'C1'
001A   00C1
001B   260A                        SKM TST
001C   0116                        SUBV A,X'B7'
001D   00B7
001E   0117                       R2 ANDV A,X'F'
001F   000F
0020   038B                        SRC B,12
0021   0197                        ANDV B,X'FFF0'
0022   FFF0
0023   089D                        OR B,A
0024   0640                        ZERO Y
0025   73E5                        JMP R1
0026   0106                       TST SUBVC A,X'DE'
0027   00DE
0028   2C0F                        SKZ T4
0029   0106                        SUBVC A,X'DF'
002A   00DF
002B   2C0B                        SKZ T3
002C   0106                        SUBVC A,X'A0'
002D   00A0
002E   2C07                        SKZ T2
002F   0106                        SUBVC A,X'8D'
0030   008D
0031   2C01                        SKZ T1
0032   73D8                        JMP R1
0033   0A79                       T1 ADD Z,Y
0034   0179                        ADDV Z,2
0035   0002
0036   076E                       T2 INCR Z
0037   076E                       T3 INCR Z
0038   D884                       T4 STR B,B,,1
0039   D05F                        STR Y,YY
003A   0040 P
003B   D867                        STR Z,B,,1
003C   D0C0                        STR D,0
003D   0402                        INH
003E   F480                        LARS *0
003F   05E3                        RTRN E
0040                              YY DS 1
                                   END
NO  ERRORS
 DBOS CC
?$EQD
```

3.11 U$VAL

FEATURES:

- Converts the hexadecimal value contained in the C-register into a string of 4 ASCII - characters and types it on the system teletype.

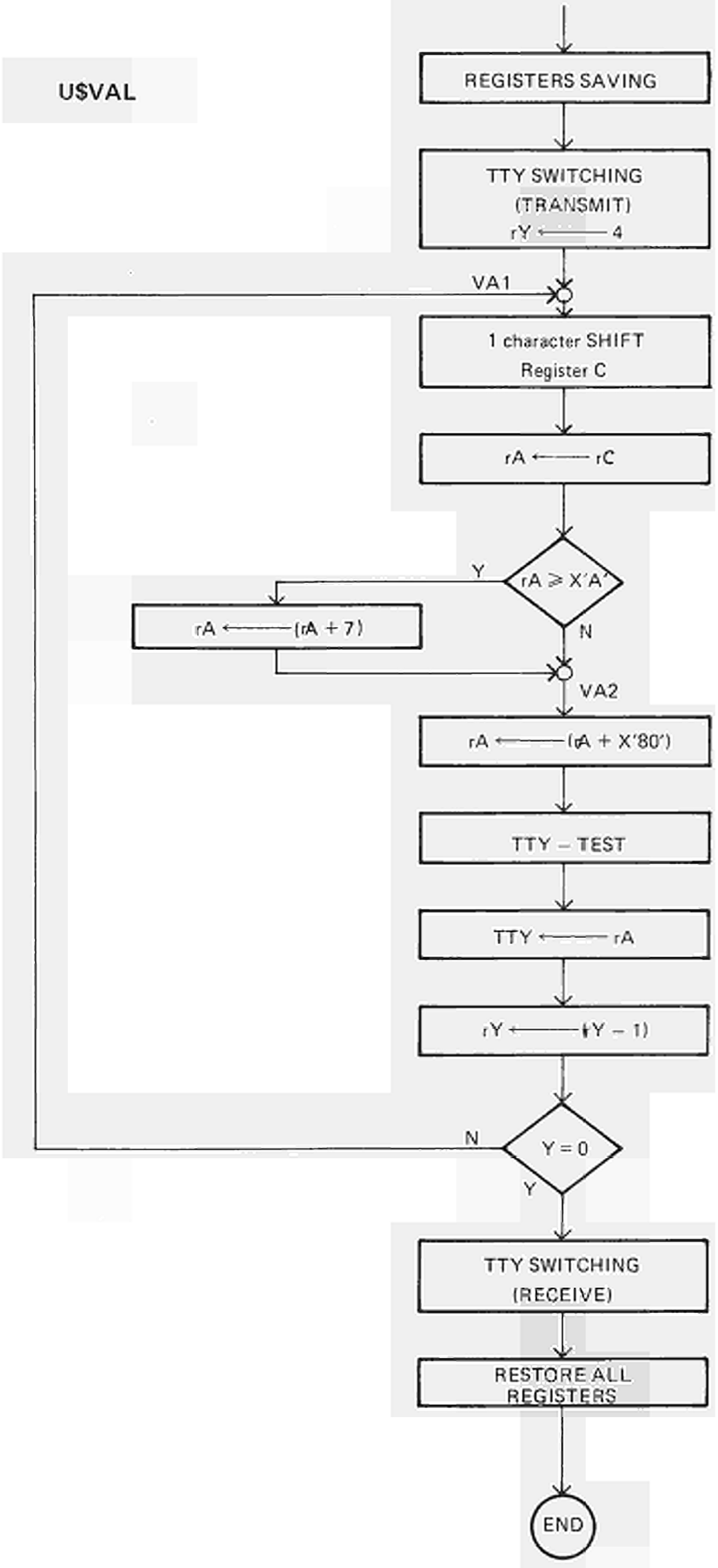
CALLING SEQUENCE

```
REF     U$VAL
 :
LDV     C, nnnn 4 hexadecimal values to be printed
JSR     U$VAL
Return from subroutine
```

MEMORY REQUIREMENTS

28_{16} words

U$VAL

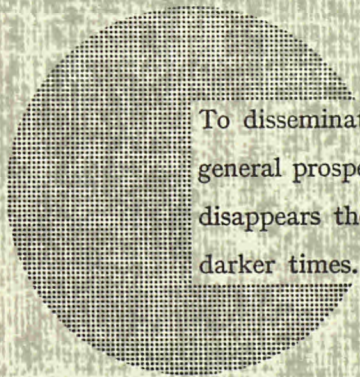

REGISTERS SAVING
TTY SWITCHING
(TRANSMIT)
rY ← 4
VA1
1 character SHIFT
Register C
rA ← rC
rA ≥ X'A'
Y
rA ← (rA + 7)
N
VA2
rA ← (rA + X'80')
TTY – TEST
TTY ← rA
rY ← (rY – 1)
Y = 0
N
Y
TTY SWITCHING
(RECEIVE)
RESTORE ALL
REGISTERS
END

```
 $SI=PR
?$A16
                                0001    DEF U$VAL
        0000                    0002  U$VAL PSECT
        0000  F4C0              0003    ENT 9
        0001  C020
        0002  09D5
        0003  0139
        0004  0009
        0005  D020
        0006  05E1
        0007  0F75              0004    RTR Z,E
        0008  0177              0005    ANDV Z,X'7FFF'
        0009  7FFF
        000A  0115              0006    LDV A,X'20'
        000B  0080
        000C  187E              0007    DTOR A,X'3E'
        000D  103F              0008    CTRL 0,X'3F'
        000E  0155              0009    LDV Y,4
        000F  0004
        0010  05E1              0010  VA1 RISE E
        0011  03AE              0011    SRC C,12
        0012  0D15              0012    RTR A,C
        0013  0117              0013    ANDV A,X'F'
        0014  000F
        0015  0106              0014    SUBVC A,X'A'
        0016  000A
        0017  2602              0015    SKM VA2
        0018  0119              0016    ADDV A,7
        0019  0007
        001A  0119              0017  VA2 ADDV A,X'B0'
        001B  00B0
        001C  0402              0018    INH
        001D  10FF              0019    TEST 0,X'3F'
        001E  73FE              0020    JMP $-1
        001F  187F              0021    DTOR A,X'3F'
        0020  0742              0022    DECR Y
        0021  25EE              0023    SKN VA1
        0022  10FF              0024    TEST 0,X'3F'
        0023  73FE              0025    JMP $-1
        0024  143F              0026    CTRL 4,X'3F'
        0025  D0C0              0027    STR D,0
        0026  F480              0028    LARS *0
        0027  05E3              0029    RTRN E
                                0030    END
 NO  ERRORS
 DBOS CC
?$EOD
?
```

NOTICE TO THE READER

All scientific and technical reports published by the Commission of the European Communities are announced in the monthly periodical **"euro-abstracts".** For subscription (1 year: BF.1025) or free specimen copies please write to :

Office for Official Publications
of the European Communities
Case postale 1003
Luxembourg 1
(Grand-Duchy of Luxembourg)



To disseminate knowledge is to disseminate prosperity — I mean general prosperity and not individual riches — and with prosperity disappears the greater part of the evil which is our heritage from darker times.

Alfred Nobel

SALES OFFICES

The Office for Official Publications sells all documents published by the Commission of the European Communities at the addresses listed below, at the price given on cover. When ordering, specify clearly the exact reference and the title of the document.

UNITED KINGDOM

H.M. Stationery Office
P.O. Box 569
London S.E. 1 — Tel. 01-928 69 77, ext. 365

BELGIUM

Moniteur belge — Belgisch Staatsblad
Rue de Louvain 40-42 — Leuvenseweg 40-42
1000 Bruxelles — 1000 Brussel — Tel. 12 00 26
CCP 50-80 — Postgiro 50-80

Agency :
Librairie européenne — Europese Boekhandel
Rue de la Loi 244 — Wetstraat 244
1040 Bruxelles — 1040 Brussel

DENMARK

J.H. Schultz — Boghandel
Møntergade 19
DK 1116 København K — Tel. 14 11 95

FRANCE

Service de vente en France des publications des Communautés européennes — Journal officiel
26, rue Desaix — 75 732 Paris - Cédex 15e
Tel. (1) 306 51 00 — CCP Paris 23-96

GERMANY (FR)

Verlag Bundesanzeiger
5 Köln 1 — Postfach 108 006
Tel. (0221) 21 03 48
Telex: Anzeiger Bonn 08 882 595
Postscheckkonto 834 00 Köln

GRAND DUCHY OF LUXEMBOURG

Office for Official Publications of the European Communities
Case postale 1003 — Luxembourg
Tel. 4 79 41 — CCP 191-90
Compte courant bancaire: BIL 8-109/6003/200

IRELAND

Stationery Office — The Controlle
Beggar's Bush
Dublin 4 — Tel. 6 54 01

ITALY

Libreria dello Stato
Piazza G. Verdi 10
00198 Roma — Tel. (6) 85 08
CCP 1/2640

NETHERLANDS

Staatsdrukkerij- en uitgeverijbedrijf
Christoffel Plantijnstraat
's-Gravenhage — Tel. (070) 81 45 11
Postgiro 42 53 00

UNITED STATES OF AMERICA

European Community Information Service
2100 M Street, N.W.
Suite 707
Washington, D.C., 20 037 — Tel. 296 51 31

SWITZERLAND

Librairie Payot
6, rue Grenus
1211 Genève — Tel. 31 89 50
CCP 12-236 Genève

SWEDEN

Librairie C.E. Fritze
2, Fredsgatan
Stockholm 16
Post Giro 193, Bank Giro 73/4015

SPAIN

Libreria Mundi-Prensa
Castello 37
Madrid 1 — Tel. 275 51 31

OTHER COUNTRIES

Office for Official Publications of the European Communities
Case postale 1003 — Luxembourg
Tel. 4 79 41 — CCP 191-90
Compte courant bancaire: BIL 8-109/6003/200

OFFICE FOR OFFICIAL PUBLICATIONS OF THE EUROPEAN COMMUNITIES
Case postale 1003 — Luxembourg