EUR 4967 e

COMMISSION OF THE EUROPEAN COMMUNITIES

# A REAL TIME OPERATING SYSTEM FOR SMART DATA TERMINALS

by

J. EDER and W. HAMMANS

1974


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. 60.—

**Commission of the**
**European Communities**
**D.G. XIII - C.I.D.**
29, rue Aldringen
Luxembourg
May 1974

EUR 4967 e

COMMISSION OF THE EUROPEAN COMMUNITIES

# A REAL TIME OPERATING SYSTEM FOR SMART DATA TERMINALS

by

J. EDER and W. HAMMANS

1974


Joint Nuclear Research Centre

Ispra Establishment — Italy

## ABSTRACT

The need for an effective communication link between various digital data sources scattered around a big laboratory environment and a commonly shared data logging system (SPC-16 minicomputer with disc and magtape) led to the development of SMART datastations (1).

SMART datastations are active, they can take over partially control of the computer. The terminals' user has real time access and may influence the data flow in the computer.

A software package is described in this report which allows to control several simultaneously working terminals in a disc based operating system (DBOS-16). A control routine recognizes a set of control commands from the terminals and stores incoming data due to the respective station address on disc. Some off-line routines are provided for file assignment, code conversion and editing of IBM compatible magnetic tapes.

CONTENTS

1. INTRODUCTION 5

2. CONTROL ROUTINE CSSM 8

3. FILE ASSIGNMENT SMART 12

4. CODE CONVERSION SMCON 13

5. MAGTAPE EDITING SMTAP 16

Figures

1. SMART Software Structure

2. FLOW CHART CSSM

3. FLOW CHART XFER

4. INTERLACED CALLS FROM DIFFERENT SMART STATIONS

5. STRUCTURE OF CORE RESIDENT BUFFERS

6. FLOW CHART SMCON

7. FLOW CHART SMTAP

8. IBM COMPATIBEL TAPE FORMAT


MANUAL
SOURCE ON
TRANSMISSION

## 1. INTRODUCTION

SMART datastations are terminals which accept data from a digital logging system (e.g. scanner, digital voltmeter) and from the terminal keyboard. Thus commands or auxiliary data may be inserted in the data stream coming from the instrumentation.
Three different data types are recognized by the control routine

- control commands
- auxiliary data
- measurement data

All types of data are transmitted in the same format as words of 4 bytes length or 32 bits. The leftmost byte, the first to be transmitted, contains the information about data type, function and eventually exponent of the following 6 BCD-coded digits. Since no decimal point is transmitted, the exponent is implicitly negativ. E.g. the value

65.4321 = 654321 E-4

would be transmitted in the format

| T4 | 65 | 43 | 21 |
|---|---|---|---|

where T stands for a 4 bit data type and function characteristic.

| FUN | EXP | D6 | D5 | D4 | D3 | D2 | D1 |
|---|---|---|---|---|---|---|---|

The three most significant bits of FUN are reserved for identification of measurement data respectively the physical unity of the measurement (e.g. Volt, Ohm, Ampère, etc.). The least significant bit of FUN determines if the value was a measurement data (bit set or logically one) or terminal data (bit reset).

FUN = XXX0 control command or auxiliary data from terminal keyboard

FUN = XXX1 measurement data

By means of the keyboard the terminal user cannot only send data but also induce the computer to execute a series of commands. The number nine in the mostleft digit (D6) is reserved for this scope. In the present state of the design 6 commands are understood by the computer but an extension is easily possible ; f.i. scanner control commands could be incorporated into the repertoire.

| Keyboard command | Action |
|---|---|
| 900000<br>910000 | Zero System buffer<br>Initialize file address |
| 920000<br>930000 | Transfer system buffer to disc file<br>Return to DBOS-monitor |
| 940000<br>990000 | Error return to DBOS<br>Attention computer operator |

A number of checks and controls are built into the program in order to detect erroneous conditions and to warn the operator. The following messages might appear on the system teletype :

- REQUEST
- ADDRESS ERROR
- DISK ERROR
- LAST SECTOR
- COEFF. ERROR
- TRANSMISSION ERROR
- OPERATOR ATTENTION

Each message is preceded by the SMART-station number in order to identify the terminal which caused the error.

The structure of the software package is schematically represented in fig. 1. The individual routines are described in more detail in the next chapters.

## 2. CONTROL ROUTINE CSSM

Since all SMART stations work in real time with immediate access to the CPU, the control routine is core resident. Only one control routine handles all stations however each station has its own core buffer which contains also the station-specific addresses etc.

Separate files on the disc are provided for each station with individual capacities. The assignment of the files (addresses and number of sectors) may be changed at any time by the computer operator. For this scope the routine SMART is foreseen, which is available for overlay and can be reloaded from disc when required.

The SMART driver, CSSM, is constructed in a way that it is supported by DBOS-16. It is fully reentrant, that means, it may be interrupted by itself.

A station requesting service sends an interrupt signal to the CPU. Supposed the interrupt system is enabled, the actual executing program is stopped and program control is switched to the SMART interrupt service routine entry (= CSSM). Now the address of the calling station is determined and the according buffer is addressed. The arriving data block of 4 bytes is checked if it was sent from a digital source or from the datastations keyboard (FUN = XXX1 or XXX0). In the latter

case further examination shows whether special action is required or not.

Fig. 2 shows the flow chart of the driver. Two save areas are provided, SAVE and SAVEB. The first contains the register contents and return address of the interrupted program while SAVEB is foreseen to manage reentry in the case where CSSM is interrupted by itself.

Each SMART station is equiped with its own address line DTFA (Device Test Function A) and has access to the common interrupt bus. If an user presses the REQUEST-key at the SMART keyboard, both signals DTFA and interrupt are activated. The interrupt starts CSSM which initiates a scanning of all DTFA signals. The first DTFA line found to be active is accepted. The station number N is determined and the interrupt signal is reset and disabled. If after any interrupt an active DTFA line is not found, a message

SMART ≠ n ADDRESS ERROR

is sent to the computer teletype.

The program checks now if the driver is still busy from a former request. If this is the case, a reminder flag is set in the buffer of the station which made the request, indicating this request must be handled later on. There-

upon the driver returns to the last interrupted program (which is of course the CSSM routine itself). A further call from the same station is then blocked and data is not put in until this last request could be terminated. Thus, no data will be lost when the driver is busy.

In fig. 4 an example is given for interleaved calls from different stations. After the request from the first station has been processed, a check on the reminder flags is made. If there is any flag set, the data from the corresponding station is put in and the interrupt is again enabled for request from other stations. When all reminder flags are reset, the driver returns to the interrupted program resp. to the DBOS monitor.

Data transfer to the disc is automatic. As soon as the core resident station buffer is full, a disc transfer is initiated (fig. 3) to the appropriate file sector. Checks are built in, to detect "DISC ERROR" and "LAST SECTOR". With the last message the operator is warned that end-of-file condition has occurred. If the overflow was less than one sector (160 values of 4 bytes each), no data is lost. It can be recuperated from the station buffer. However, if the overflow is more than one sector, the operator will be warned again and the last sector will be lost.

Data transfer can be commanded too, from the SMART terminal by typing 92 00 00 (see control command table).

Before starting a new measurement the station buffer should be cleared by the command 90 00 00. By typing 91 00 00 the first sector of the stations file can be addressed if desired.

For the case of error recovery two commands are available which allows the terminal user to return to the DBOS-system : 93 00 00 returns to the DBOS monitor, whereas 94 00 00 returns to the error entry, which saves the whole core memory on disc for a later dump. This command should be used only in case of emergency since the real time system is consequently disabled and the transmission of other stations may be disturbed.

## 3. FILE ASSIGNMENT SMART

Disc resident files are assigned to the individual stations. With the program SMART the parameters (FIRST SECTOR and LAST SECTOR) may be changed due to the requirements of the stations' user.

The program SMART is disc resident and can be called into core by typing SSMART. This program is interactive, it asks the operator via teletype the required information (sector numbers). Thus, knowledge of the software or of the operation system is not necessary.

If the number of SMART terminals is to be changed, only one statement in the SMART routine has to be modified. The number of stations n under CSSM control is contained in

```
TSSM EQU n
```

After this modification, recompilation and execution of the SMART routine the necessary core resident buffer areas are automatically reserved. Each station requires a buffer area of 327 words. The individual buffer areas are stacked beginning from the bottom of the core (see fig. 5), so that additional areas can be inserted without distroying the already existing ones.

In order to protect the whole buffer zone from being overlaid by other programs running under DBOS, the last available core location is automatically communicated to DBOS.

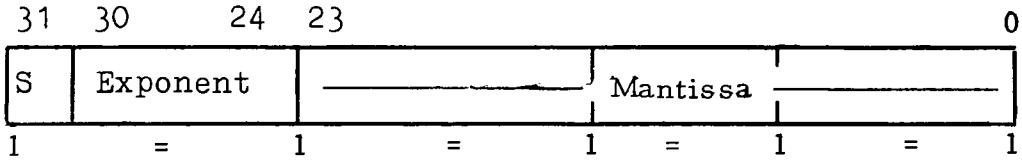
The SMART routine itself does not require any permanent core capacity. It may be overlaid by other programs after the desired file assignment has been made. After execution SMART returns to the DBOS monitor.

## 4. CODE CONVERSION SMCON

The data format delivered by SMART terminals is generally not suited for further calculations. In order to minimize the busy-time of the driver CSSM no on line code conversion has been foreseen.
As already mentioned in chapter 1, the SMART terminals transmit 6 digits of data and an one digit (negative) exponent. The digits may be coded in

1248 code for keyboard data
1224 code for measurement data.

In order to get compatibility with the SPC - 16 FORTRAN routines, the SMART data must be converted into the following floating point format :




The mantissa is normalized, the high order bit (23) being 1. The exponent is biased by hexadecimal $40_{16}$. Negative real values are characterized by the two's complement of the entire argument (sign bit (31) = 1), however SMART data are only positive.

SMCON is a disc resident program which can be called into core by typing SSMCON. This routine asks the user for the desired SMART-station number and responds typing out the numbers of the occupied sectors of the corresponding disc-file.
Starting with the first sector, the subroutine USDVDE converts the measurement data into 8 digit BCD words. It follows the conversion into doubleword hexadecimal integers and thereupon into reals. When all sectors are converted and reinserted into the disc file, the program returns to the DBOS monitor.

Subroutines called : USEDIT, USVAL, USDEHE ) described in EUR 1583

USDVDE

USD8HE

USMPY

DSA )
DSR )
FSE10 ) system program library
FSD11 )
FSM11 )

Program length $232_{16} = 562_{10}$ words (16 bit)
(without subroutines).

## 5. EDITING OF MAGNETIC TAPES SMTAP

It is often desired to process data gathered by the SPC - 16 system at the computing center on IBM machines. For the data exchange a standard 9 track magnetic tape is available as an SPC - 16 peripheral.

In order to get compatibility with the IBM standards, further code conversions are necessary The ASCII-characters, generally used on most minicomputers, have to be transcoded into EBCDIC-characters and the SPC - 16 floating point format must be converted into IBM floating point format.

For ASCII - EBCDIC conversion the subroutine USAEBC is used, which converts 2 ASCII characters (one 16 bit word) into EBCDIC characters by table look up. Some characters which have no EBCDIC-equivalent are set to codes normally not used by EBCDIC, e.g.

| Hexadecimal characters | ASCII | EBCDIC |
|---|---|---|
| CARRIAGE RETURN | 8A | FE |
| LINE FEED | 8D | FF |

These codes could be utilized as control characters at the IBM-side to branch to appropriate format statements.

A comparison between IBM short floating point format and the SPC - 16 representation shows the following differences :

The IBM exponent works with base 16, whereas the SPC - 16 uses base 2. Both exponents are biased by hexadecimal 40. Negative values of the mantissa are represented in the IBM floating format by the sign bit (31) and by the 2's complement of the mantissa only.
Wordlength is identical, however the precision in the IBM case is somewhat reduced, since 4 bits of the mantissa are required for the base 16. On the other hand there is an extended range of the exponent

| | |
|---|---|
| SPC - 16 floating | $5.4 * 10^{-20} \div 9.2 * 10^{+18}$ |
| | $5.4 * 10^{-79} \div 7.2 * 10^{+75}$ |

The subroutine USRIBM takes care for correct conversion.
Additional to the numberic information the user may wish to have a comment on the tape which simplifies the identification of big quantities of measurement data. For this purpose the subroutine USVSM has been prepared. This routine edits a standard comment of 130 characters maximum, corresponding to one print line. The information, the user wants to put into the comment (Measurement label, date, etc.) is put in in form of a dialogue with the teletype. That means the teletype writes prepared questions and the user inserts the answers.
In case of any error, say the answer does not fit into the standard comment format, the question is repeated until the correct answer has been given. The comment line may be formated with leading blanks. Similar as with FORTRAN-format statements the first character of the comment line can be interpreted as carriage control character and may assume one of the following values : blanc, 0, 1 or +

SMTAP executes furthermore all commands for positioning of the magtape (USTPOS), opening the file and writing a header on tape (USHEAD) and closing the file (USTMRK).

Figure 8 demonstrates the file structure of a magnetic tape. The standard data record length is 640 bytes or 160 real x 4 values. The number of records in one file is arbitrary. (These standard specifications can be changed easily). On input to IBM machines the first 3 records will normally be located as dummy records.

After termination control is returned to DBOS and SMTAP may be overlaid.

Subroutines called : USTPOS
USHEAD
USTMRK

USEDIT )
USRESP ) described in EUR 1583
USDEHE )

USVSM
USTVSP
USDKTO
USAEBC
USRIBM

Program length = $D0_{16}$ = $208_{10}$ words (16 bit).

REFERENCES

1) J. EDER, F. SOREL, E. BETTENDROFFER :
Terminal SMART - 77 for improved dialogue between experiments and minicomputer
EUR 5060, 1973. pg. 286 - 288.

2) E. BETTENDROFFER, F. SOREL :
SMART - terminal for automation of experiments by minicomputers.
(To be published).

3) J. EDER, C. PAGNY :
PLOTTY, an interactive plotting program for graphs on a teletype by means of the SPC - 16 minicomputer.
EUR 5024, 1973

```
                    0001  ********************************************
                    0002  *                                          *
                    0003  *                  C$SM                    *
                    0004  *                                          *
                    0005  *        SMART-REAL-TIME-DRIVER.           *
                    0006  *                                          *
                    0007  *             (EDER-20/05/73)              *
                    0008  *                                          *
                    0009  ********************************************
                    0010  *
                    0011          DEF     C$SM
                    0012          REF     IOS,F$IOR,U$EDIT
                    0013  *
    1000            0014  CTRL    EQU     X'1000'
    10C0            0015  TESTA   EQU     X'10C0'
    11C0            0016  TESTB   EQU     X'11C0'
    1C80            0017  DTIRB   EQU     X'1C80'
    1000            0018  SOURCE  EQU     X'1000'
                    0019  *
0000                0020          PSECT
                    0021  *
0000  7072          0022  C$SM    JMP     I$SM
0001  8000          0023          DC      X'8000'
0002                0024  SML     DS      1
0003                0025  CADR    DS      1
0004  0000          0026  BUSY    DC      0
0005  7FFF          0027  MTIM    DC      X'7FFF'
0006  00EC P        0028  PROG    DC      C90
0007  00EE P        0029          DC      C91
0008  00F1 P        0030          DC      C92
0009  00F3 P        0031          DC      C93
000A  00F6 P        0032          DC      C94
000B  0126 P        0033          DC      RESTOR
000C  0126 P        0034          DC      RESTOR
000D  0126 P        0035          DC      RESTOR
000E  0126 P        0036          DC      RESTOR
000F  00F8 P        0037          DC      C99
0010  308D          0038  DKFILE  DC      X'308D',F$IOR,0
0011  0000 X
0012  0000
0013  A3B1          0039  DSTN    DC      '#1'
0014  8D8A          0040  DST     TEXT    '$8D$8ASMART '
0015  D3CD
0016  C1D2
0017  D4A0
0018  0000          0041  DN      DC      0,0
0019  0000
001A  0021 P        0042  ETEXT   DC      TXT0
001B  0026 P        0043          DC      TXT1
001C  002E P        0044          DC      TXT2
001D  0035 P        0045          DC      TXT3
001E  003C P        0046          DC      TXT4
001F  0043 P        0047          DC      TXT5
0020  004E P        0048          DC      TXT6
0021  A0D2          0049  TXT0    TEXT    ' REQUEST'
0022  C5D1
0023  D5C5
0024  D3D4
```

```
0025  0000      0050          DC      0
0026  A0C1      0051  TXT1    TEXT    ' ADDRESS ERROR'
0027  C4C4
0028  D2C5
0029  D3D3
002A  A0C5
002B  D2D2
002C  CFD2
002D  0000      0052          DC      0
002E  A0C4      0053  TXT2    TEXT    ' DISK ERROR '
002F  C9D3
0030  CBA0
0031  C5D2
0032  D2CF
0033  D2A0
0034  0000      0054          DC      0
0035  A0CC      0055  TXT3    TEXT    ' LAST SECTOR'
0036  C1D3
0037  D4A0
0038  D3C5
0039  C3D4
003A  CFD2
003B  0000      0056          DC      0
003C  A0C3      0057  TXT4    TEXT    ' COEFF.ERROR'
003D  CFC5
003E  C6C6
003F  AEC5
0040  D2D2
0041  CFD2
0042  0000      0058          DC      0
0043  A0CF      0059  TXT5    TEXT    ' OPERATOR ATTENTION '
0044  D0C5
0045  D2C1
0046  D4CF
0047  D2A0
0048  C1D4
0049  D4C5
004A  CED4
004B  C9CF
004C  CEA0
004D  0000      0060          DC      0
004E  A0D4      0061  TXT6    TEXT    ' TRANSMISSION ERROR '
004F  D2C1
0050  CED3
0051  CDC9
0052  D3D3
0053  C9CF
0054  CEA0
0055  C5D2
0056  D2CF
0057  D2A0
0058  0000      0062          DC      0
0059  8D8A      0063  CRLF    DC      X'8D8A',0
005A  0000
005B              0064  SAVE    DS      12
0067              0065  SAVEB   DS      12
                  0066  *
```

```
0073  F0DF      0067  I$SM   SARS    SAVEB
0074  0067 P
0075  A01F      0068        TBIT    0,BUSY
0076  0004 P
0077  2407      0069        SKN     BSY
0078  F0DF      0070        SARS    SAVE
0079  005B P
007A  43F4      0071        LDA     SAVEB+8
007B  53E7      0072        STA     SAVE+8
007C  01D5      0073        LDV     D,SAVE
007D  005B P
007E  7002      0074        JMP     ANF
007F  01D5      0075  BSY   LDV     D,SAVEB
0080  0067 P
                0076  *
0081  0660      0077  ANF   ZERO    Z
0082  C03F      0078  IDENT LDR     X,CADR  X=CORE
0083  0003 P
0084  0B39      0079        ADD     X,Z
0085  C120      0080        LDR     X,0,X   X=BUF(Z)
0086  C142      0081        LDR     Y,2,X   Y=DSF(X)
0087  40AF      0082        LDA     =TESTA
0088  0A1D      0083        OR      A,Y
0089  0510      0084        XEC     A       TESTA(X)
008A  7002      0085        JMP     ID1
008B  D86B      0086        STR     Z,11,,1
008C  7006      0087        JMP     ENABL
008D  076E      0088  ID1   INCR    Z
008E  E07F      0089        CMR     Z,SML
008F  0002 P
0090  27F1      0090        SKM     IDENT
0091  40A6      0091        LDA     =1
0092  7047      0092        JMP     ERR
                0093  *
0093  C142      0094  ENABL LDR     Y,2,X
0094  40A4      0095        LDA     =CTRL
0095  0A1D      0096        OR      A,Y
0096  0510      0097        XEC     A       SEND DCP(X)
0097  B01F      0098        SBIT    0,BUSY
0098  0004 P
0099  2C06      0099        SKZ     INPUT
009A  409D      0100        LDA     =1
009B  D101      0101        STR     A,1,X
009C  7097      0102        JMP     RETRN
                0103  *
009D  0640      0104  QUEUE ZERO    Y
009E  D141      0105        STR     Y,1,X
009F  0402      0106        INH
                0107  *
```

```
00A0  4099       0108  INPUT  LDA    =TESTB
00A1  0A1D       0109         OR     A,Y
00A2  C0BF       0110         LDR    C,=DTIRB
00A3  013B P
00A4  0ABD       0111         OR     C,Y
00A5  0660       0112         ZERO   Z
00A6  C05F       0113  INP1   LDR    Y,MTIM
00A7  0005 P
00A8  0742       0114  INP2   DECR   Y
00A9  2620       0115         SKM    ETIME
00AA  0510       0116         XEC    A        TESTB(X,Z)
00AB  73FC       0117         JMP    INP2
00AC  05B0       0118         XEC    C        DTIRB(X,Z)
00AD  9B89       0119         STBY   B,9,Z,1
00AE  076E       0120         INCR   Z
00AF  0166       0121         SUBVC  Z,4
00B0  0004
00B1  27F4       0122         SKM    INP1
00B2  05E1       0123         RISE   E
                 0124  *
00B3  4809       0125  CHCOE  LDA    9,1
00B4  0107       0126         ANDVC  A,SOURCE
00B5  1000
00B6  2C17       0127         SKZ    COEF
                 0128  *
00B7  C140       0129  RANGE  LDR    Y,0,X
00B8  0A55       0130         RTR    Y,Y
00B9  260E       0131         SKM    JXF1
00BA  0146       0132         SUBVC  Y,319
00BB  013F
00BC  2E09       0133         SKP    JXF
                 0134  *
00BD  0959       0135  STORE  ADD    Y,X
00BE  D207       0136         STR    A,7,Y
00BF  480A       0137         LDA    10,1
00C0  D208       0138         STR    A,8,Y
00C1  0159       0139         ADDV   Y,2
00C2  0002
00C3  0956       0140         SUB    Y,X
00C4  D140       0141         STR    Y,0,X
00C5  7060       0142         JMP    RESTOR
                 0143  *
00C6  6033       0144  JXF    JSR    XFER
00C7  4809       0145         LDA    9,1
00C8  0640       0146  JXF1   ZERO   Y
00C9  73F3       0147         JMP    STORE
                 0148  *
00CA  0B15       0149  ETIME  RTR    A,Z
00CB  2C01       0150         SKZ    ET1      IF FIRST TIME
00CC  406F       0151         LDA    =6
00CD  700C       0152  ET1    JMP    ERR
                 0153  *
```

```
00CE  0855      0154  COEF  RTR     Y,A
00CF  0642      0155        ZLBY    Y
00D0  0156      0156        SUBV    Y,X'90'
00D1  0090
00D2  27E4      0157        SKM     RANGE   IF NOT CTRL-COMMAND
00D3  0146      0158        SUBVC   Y,10
00D4  000A
00D5  2E03      0159        SKP     ECOE    IF>99
00D6  C25F      0160        LDR     Y,PROG,Y
00D7  0006 P
00D8  0543      0161        RTRN    Y       BRANCH TO CC-TAB.
                0162  *
00D9  4063      0163  ECOE  LDA     =4
00DA  0855      0164  ERR   RTR     Y,A
00DB  C25F      0165        LDR     Y,ETEXT,Y       LOAD TEXT
00DC  001A P
00DD  C09F      0166        LDR     B,DSTN
00DE  0013 P
00DF  C86B      0167        LDR     Z,11,,1 LOAD STATION#
00E0  0B99      0168        ADD     B,Z
00E1  D09F      0169        STR     B,DN
00E2  0018 P
00E3  D05F      0170        STR     Y,MESS
00E4  00E8 P
00E5  6458      0171        JSR     U$EDIT
00E6  0014 P    0172        DC      DST
00E7  6456      0173        JSR     U$EDIT
00E8            0174  MESS  DS      1
00E9  6454      0175        JSR     U$EDIT
00EA  0059 P    0176        DC      CRLF
00EB  703A      0177        JMP     RESTOR
                0178  *
00EC  602C      0179  C90   JSR     CLBUF
00ED  7038      0180        JMP     RESTOR
                0181  *
00EE  C183      0182  C91   LDR     B,3,X
00EF  D186      0183        STR     B,6,X
00F0  7035      0184        JMP     RESTOR
                0185  *
00F1  6008      0186  C92   JSR     XFER
00F2  7033      0187        JMP     RESTOR
                0188  *
00F3  404B      0189  C93   LDA     =X'80'
00F4  D807      0190  C931  STR     A,7,,I
00F5  7030      0191        JMP     RESTOR
                0192  *
00F6  4049      0193  C94   LDA     =X'81'
00F7  73FC      0194        JMP     C931
                0195  *
00F8  4048      0196  C99   LDA     =5
00F9  73E0      0197        JMP     ERR
                0198  *
```

```
00FA  0955       0199  XFER   RTR     Y,X
00FB  0F95       0200         RTR     B,E
00FC  05E1       0201         RISE    E
00FD  0159       0202         ADDV    Y,5
00FE  0005
00FF  D05F       0203         STR     Y,DKFILE+2
0100  0012 P
0101  C146       0204         LDR     Y,6,X
0102  E143       0205         CMR     Y,3,X    CS>=FS?
0103  2E02       0206         SKP     XF0      YES
0104  C143       0207         LDR     Y,3,X    NO,CS<FS
0105  D146       0208         STR     Y,6,X    CS=FS
0106  C144       0209  XF0    LDR     Y,4,X
0107  E146       0210         CMR     Y,6,X    LS<CS?
0108  2607       0211         SKM     XF1      YES
0109  C146       0212         LDR     Y,6,X    NO,LS>=CS
010A  6437       0213         IOS     DKFILE
010B  0010 P
010C  0702       0214         DECR    A
010D  2C02       0215         SKZ     XF1
010E  4034       0216         LDA     =2
010F  73CA       0217         JMP     ERR
0110  074E       0218  XF1    INCR    Y
0111  D146       0219         STR     Y,6,X    CS=CS+1
0112  6006       0220         JSR     CLBUF
0113  C144       0221         LDR     Y,4,X
0114  E146       0222         CMR     Y,6,X    LS>=CS?
0115  2E02       0223         SKP     XF2      YES
0116  402D       0224         LDA     =3       NO,END OF FILE
0117  73C2       0225         JMP     ERR
0118  0583       0226  XF2    RTRN    B
                 0227  *
0119  0FB5       0228  CLBUF  RTR     C,E
011A  05E1       0229         RISE    E
011B  0600       0230         ZERO    A
011C  0640       0231         ZERO    Y
011D  D140       0232         STR     Y,0,X
011E  0975       0233         RTR     Z,X
011F  D307       0234  CL     STR     A,7,Z
0120  074E       0235         INCR    Y
0121  076E       0236         INCR    Z
0122  0146       0237         SUBVC   Y,320
0123  0140
0124  27FA       0238         SKM     CL
0125  05A3       0239         RTRN    C
                 0240  *
```

```
0126  0660        0241  RESTOR ZERO    Z
0127  C03F        0242  NXTZ   LDR     X,CADR
0128  0003 P
0129  0B39        0243         ADD     X,Z
012A  C120        0244         LDR     X,0,X
012B  C141        0245         LDR     Y,1,X
012C  0A55        0246         RTR     Y,Y
012D  256F        0247         SKN     QUEUE
012E  076E        0248         INCR    Z
012F  E07F        0249         CMR     Z,SML
0130  0002 P
0131  27F5        0250         SKM     NXTZ
0132  301F        0251         RBIT    0,BUSY
0133  0004 P
                  0252  *
0134  0402        0253  RETRN  INH
0135  F880        0254         LARS    0,,1
0136  05E3        0255         RTRN    E
                  0256  *
                  0257         END
0137  10C0
0138  0001
0139  1000
013A  11C0
013B  1C80
013C  0006
013D  0004
013E  0000 X
013F  0080
0140  0081
0141  0005
0142  0000 X
0143  0002
0144  0003
     00  ERRS
 DBOS CC
?$EOD
?
```

```
                    0001  ****************************************
                    0002  *                                      *
                    0003  *                SMART                 *
                    0004  *                                      *
                    0005  *          DISK-FILE-ASSIGMENT         *
                    0006  *                                      *
                    0007  *            (EDER-10/05/73)           *
                    0008  *                                      *
                    0009  ****************************************
                    0010  *
                    0011          DEF     INIT,T$SML,CORE
                    0012          REF     IOS,F$IOR,E$MON
                    0013          REF     E$PUT,E$CORE,E$SDC
                    0014          REF     U$RESP,U$VAL
                    0015  *
                    0016  #BUFF   DS      1;
                    0017          DC      0,#1;
                    0018          DS      2;
                    0019          DC      640;
                    0020          DS      321
                    0021  *
      0001          0022  T$SML   EQU     1
      0011          0023  SM      EQU     X'11'
      001F          0024  DSF     EQU     X'1F'
      004F          0025  IVECT   EQU     X'4F'
      2FDF          0026  BOOT    EQU     X'2FDF'
      2E97          0027  CORE    EQU     BOOT-T$SML*328
                    0028  *
2E97                0029          DSECT   CORE
                    0030  *
                    0031          DO      T$SML,1
2E97  2E98          0032          DC      BOOT-327*?
                    0033          DO      T$SML,1
2E99  0000          0034          BUFF    DSF-T$SML+?
2E9A  001F
2E9D  0280
2E9E                0035  *
0000                0036          PSECT
                    0037  *
0000                0038  SAVZ    DS      1
0001  0087          0039  MESS    DC      X'87',F$IOR,0
0002  0000 X
0003  0000
0004  000A          0040  DST     DC      10,0
0005  0000
0006  8D8A          0041          TEXT    '$8D$8ASMART '
0007  D3CD
0008  C1D2
0009  D4A0
000A  A3B1          0042          DC      '#1'
000B  A3B1          0043  DSTN    DC      '#1'
000C  000E P        0044  ETEXT   DC      TXT0
000D  0018 P        0045          DC      TXT1
000E  000F          0046  TXT0    DC      15,0
000F  0000
0010  A0C6          0047          TEXT    ' FIRST SECTOR ='
0011  C9D2
0012  D3D4
0013  A0D3
0014  C5C3
0015  D4CF
0016  D2A0
0017  BDA0
0018  000F          0048  TXT1    DC      15,0
0019  0000
001A  A0CC          0049          TEXT    ' LAST  SECTOR ='
001B  C1D3
001C  D4A0
001D  A0D3
001E  C5C3
001F  D4CF
0020  D2A0
0021  BDA0
                    0050  *
```

```
0022  C03F       0051  INIT   LDR     X,E$PUT
0023  0000 X
0024  444F       0052         LDA     E$SDC
0025  544F       0053         STA     0
0026  0402       0054         INH
0027  C131       0055         LDR     X,SM,X  X=ADDR. C$SM
0028  D03F       0056         STR     X,IVECT
0029  004F
002A  0155       0057         LDV     Y,T$SML Y=NO.OF STAT.
002B  0001
002C  D142       0058         STR     Y,2,X   C$SM+2
002D  0155       0059         LDV     Y,CORE
002E  2E97
002F  D143       0060         STR     Y,3,X BEGIN BUF.ZONE(C$SM+3)
0030  D05F       0061         STR     Y,E$CORE
0031  0000 X
0032  0544       0062         DSPL    Y
                 0063  *
0033  0660       0064         ZERO    Z
0034  D164       0065         STR     Z,4,X
0035  C33F       0066  NEXTZ  LDR     X,CORE,Z
0036  2E97
0037  D07F       0067         STR     Z,SAVZ
0038  0000 P
0039  403B       0068         LDA     =0
003A  6016       0069  RES1   JSR     ERRM
003B  143F       0070         CTRL    4,X'3F'
003C  6439       0071         JSR     U$RESP
003D  73F6       0072         JMP     NEXTZ-1
003E  73FD       0073         JMP     $-2
003F  7028       0074         JMP     SDW1
0040  7002       0075         JMP     $+3
0041  D183       0076         STR     B,3,X
0042  D186       0077         STR     B,6,X
0043  4033       0078         LDA     =1
0044  600C       0079  RES2   JSR     ERRM
0045  143F       0080         CTRL    4,X'3F'
0046  642F       0081         JSR     U$RESP
0047  73EC       0082         JMP     NEXTZ-1
0048  73FD       0083         JMP     $-2
0049  7024       0084         JMP     SDW2
004A  7001       0085         JMP     $+2
004B  D184       0086         STR     B,4,X
004C  076E       0087         INCR    Z
004D  0166       0088         SUBVC   Z,T$SML
004E  0001
004F  27E5       0089         SKM     NEXTZ
0050  7427       0090         JMP     E$MON
                 0091  *
```

```
0051  0FB5       0092  ERRM   RTR     C,E
0052  05E1       0093         RISE    E
0053  0855       0094         RTR     Y,A
0054  C25F       0095         LDR     Y,ETEXT,Y
0055  000C P
0056  C09F       0096         LDR     B,DSTN
0057  000B P
0058  C07F       0097         LDR     Z,SAVZ
0059  0000 P
005A  0B99       0098         ADD     B,Z
005B  D09F       0099         STR     B,DSTN-1
005C  000A P
005D  0195       0100         LDV     B,DST
005E  0004 P
005F  D09F       0101         STR     B,MESS+2
0060  0003 P
0061  6417       0102         IOS     MESS
0062  0001 P
0063  D05F       0103         STR     Y,MESS+2
0064  0003 P
0065  6413       0104         IOS     MESS
0066  0001 P
0067  05A3       0105         RTRN    C
                 0106  *
0068  C1A6       0107  SDW1   LDR     C,6,X
0069  6410       0108         JSR     U$VAL
006A  0115       0109         LDV     A,X'F0'
006B  00F0
006C  187E       0110         DTOR    A,X'3E'
006D  73D5       0111         JMP     RES2-1
006E  C1A4       0112  SDW2   LDR     C,4,X
006F  640A       0113         JSR     U$VAL
0070  0115       0114         LDV     A,X'F0'
0071  00F0
0072  187E       0115         DTOR    A,X'3E'
0073  73D8       0116         JMP     RES2+8
0074             0117         LPOOL   6
      0022 P     0118         END     INIT

0074  0000 X
0075  0000
0076  0000 X
0077  0001
0078  0000 X
0079  0000 X
007A  0000 X
    00  ERRS
 DBOS CC
?$EOD
?
```

```
                    0001  ********************************************
                    0002  *                                          *
                    0003  *                  SMCON                   *
                    0004  *                                          *
                    0005  * CONVERTS AND RESTORES DISK-STORED SMART-*
                    0006  * DATA IN REAL-NUMBERS(GA-FORMAT)          *
                    0007  *                                          *
                    0008  *           (HAMMANS-16/06/73)             *
                    0009  *                                          *
                    0010  ********************************************
                    0011  *
                    0012          DEF     CODERR,ERROR
                    0013          REF     IOS,F$IOR,E$MON,E$CORE,E$PUT
                    0014          REF     U$EDIT,U$RESP,U$VAL,U$DVDE
                    0015          REF     U$D8HE,D$A,D$R,F$E10,F$D11
                    0016          REF     F$M11
                    0017  *
0000                0018          DSECT
                    0019  *
0000  2080          0020  TITLE   WRITE   CC$,TXT
0001  0000 X
0002  8003
0003  0023          0021  TXT     DC      35,0
0004  0000
0005  8AC3          0022          TEXT    '$8ACODE-CONVERSION'
0006  CFC4
0007  C5AD
0008  C3CF
0009  CED6
000A  C5D2
000B  D3C9
000C  CFCE
000D  A0D0          0023          TEXT    ' PROGRAM FOR SMART$8D'
000E  D2CF
000F  C7D2
0010  C1CD
0011  A0C6
0012  CFD2
0013  A0D3
0014  CDC1
0015  D2D4
0016  8DA0
0017  8D8A          0024  STATNO  TEXT    '$8D$8A$8ASMART NO.= '
0018  8AD3
0019  CDC1
001A  D2D4
001B  A0CE
001C  CFAE
001D  BDA0
001E  0000          0025          DC      0
001F  8AC1          0026  SECTNO  TEXT    '$8AACTUAL SECTORS:  '
0020  C3D4
0021  D5C1
0022  CCA0
0023  D3C5
0024  C3D4
0025  CFD2
0026  D3BA
0027  A0A0
```

```
0028  0000      0027          DC      0
0029  A0AD      0028  SPACE   TEXT    ' -'
002A  A000      0029          DC      X'A000'
002B  8D8A      0030  LINE    TEXT    '$8D$8A'
002C  0000      0031          DC      0
002D  2080      0032  EFLAG1  WRITE   CC$,ETXT1
002E  0000 X
002F  8030
0030  0010      0033  ETXT1   DC      16,0
0031  0000
0032  8DC9      0034          TEXT    '$8DILLEGAL NUMBER$8A$8D'
0033  CCCC
0034  C5C7
0035  C1CC
0036  A0CE
0037  D5CD
0038  C2C5
0039  D28A
003A  8DA0
003B  2080      0035  EFLAG2  WRITE   CC$,ETXT2
003C  0000 X
003D  803E
003E  0017      0036  ETXT2   DC      23,0
003F  0000
0040  8AC9      0037          TEXT    '$8AINVALID DEC'
0041  CED6
0042  C1CC
0043  C9C4
0044  A0C4
0045  C5C3
0046  C9CD      0038          TEXT    'IMAL NUMBER$8D'
0047  C1CC
0048  A0CE
0049  D5CD
004A  C2C5
004B  D28D
004C  2080      0039  EFLAG3  WRITE   CC$,ETXT3
004D  0000 X
004E  804F
004F  0018      0040  ETXT3   DC      24,0
0050  0000
0051  8A8D      0041          TEXT    '$8A$8DILLEGAL STATION-'
0052  C9CC
0053  CCC5
0054  C7C1
0055  CCA0
0056  D3D4
0057  C1D4
0058  C9CF
0059  CEAD
005A  CED5      0042          TEXT    'NUMBER$8D'
005B  CDC2
005C  C5D2
005D  8DA0
005E  0061      0043  READ    DC      X'61',F$IOR,SECBUF
005F  0000 X
0060  0061 D
0061  0280      0044  SECBUF  AREA    640
0062  0000
01A3  00A1      0045  WRITE   DC      X'A1',F$IOR,SECBUF
01A4  0000 X
01A5  0061 D
01A6            0046  SRANGE  DS      1
01A7            0047  RESULT  DS      2
01A9            0048  PART2   DS      2
01AB  0011      0049  SM      DC      X'11'
01AC            0050  SNUM    DS      1
01AD            0051  FSECT   DS      1
01AE  44A0      0052  BASE    DC      10.
01AF  0000
01B0            0053  EXP     DS      1
      01B1      0054  DL      EQU     $-$$
                0055  *
```

```
0000                0056          PSECT
                    0057  *
0000                0058          DS      DL
                    0059  *
01B1  0402          0060  SM1CON  INH
01B2  01D5          0061          LDV     D,$$     SET D-BASE
01B3  0000 D
01B4  6471          0062          IOS     TITLE
01B5  8000
01B6  C03F          0063          LDR     X,E$PUT
01B7  0000 X
01B8  CD3F          0064          LDR     X,*SM,X
01B9  01AB
01BA  C142          0065          LDR     Y,2,X    Y=T$SML
01BB  646B          0066  L1      JSR     U$EDIT
01BC  0017 D        0067          DC      STATNO
01BD  646A          0068          JSR     U$RESP
01BE  73FC          0069          JMP     $-3
01BF  73FD          0070          JMP     $-2
01C0  73FC          0071          JMP     $-3
01C1  73F9          0072          JMP     $-6
01C2  D89F          0073          STR     B,SNUM
01C3  01AC
01C4  E85F          0074          CMR     Y,SNUM   STAT.NO.OK?
01C5  01AC
01C6  265C          0075          SKM     STATER   NO
01C7  C143          0076          LDR     Y,3,X    YES,Y=CORE
01C8  C83F          0077          LDR     X,SNUM
01C9  01AC
01CA  0959          0078          ADD     Y,X
01CB  645B          0079          JSR     U$EDIT
01CC  001F D        0080          DC      SECTNO
01CD  C2A3          0081          LDR     C,3,Y    C=FIRST SECT.
01CE  645A          0082          JSR     U$VAL
01CF  D8BF          0083          STR     C,FSECT
01D0  01AD
01D1  6455          0084          JSR     U$EDIT
01D2  0029 D        0085          DC      SPACE
01D3  C2A6          0086          LDR     C,6,Y    C=CURR.SECT.
01D4  D8BF          0087          STR     C,SRANGE
01D5  01A6
01D6  07A2          0088          DECR    C
01D7  6451          0089          JSR     U$VAL
01D8  644E          0090          JSR     U$EDIT
01D9  002B D        0091          DC      LINE
01DA  C85F          0092          LDR     Y,FSECT
01DB  01AD
01DC  D85F          0093  LOOP    STR     Y,SECBUF+1
01DD  0062
01DE  6447          0094          IOS     READ     READ 1 SECT.IN
01DF  805E
01E0  0135          0095          LDV     X,2
01E1  0002
01E2  C09F          0096  CONT    LDR     B,=SECBUF
01E3  022A P
01E4  0999          0097          ADD     B,X
01E5  6445          0098          JSR     U$DVDE   CONVERT
                    0099  *
```

```
01E6  0C15      0100          RTR     A,B
01E7  0601      0101          ZRBY    A         EXPONENT
01E8  0604      0102          EXBY    A
01E9  D81F      0103          STR     A,EXP
01EA  01B0
01EB  0682      0104          ZLBY    B         NR.-PART 1
01EC  0186      0105          SUBVC   B,X'9A'   NUMBER>99?
01ED  009A
01EE  2E31      0106          SKP     ERROR     YES,ILLEGAL
01EF  D8BF      0107          STR     C,PART2   NO
01F0  01A9
01F1  643A      0108          JSR     U$D8HE    CONVERT
01F2  81A9      0109          ARG     PART2
01F3  81A7      0110          ARG     RESULT
01F4  0000      0111          DC      0
01F5  6437      0112          JSR     D$R       CONVERT 16BIT-
01F6  81A7      0113          ARG     RESULT    INTEG. TO REAL
01F7  D81F      0114          STR     A,RESULT
01F8  01A7
01F9  D89F      0115          STR     B,RESULT+1
01FA  01A8
01FB  C81F      0116          LDR     A,BASE
01FC  01AE
01FD  C89F      0117          LDR     B,BASE+1
01FE  01AF
01FF  C8BF      0118          LDR     C,EXP
0200  01B0
0201  642C      0119          JSR     F$E10     10.0**EXP.
0202  D81F      0120          STR     A,PART2
0203  01A9
0204  D89F      0121          STR     B,PART2+1
0205  01AA
0206  0115      0122          LDV     A,X'4180'          1.0(REAL)
0207  4180
0208  0680      0123          ZERO    B
0209  6425      0124          JSR     F$D11     1./10.0**EXP.)
020A  81A9      0125          ARG     PART2
020B  6424      0126          JSR     F$M11     EXP. CORRECTUR
020C  81A7      0127          ARG     RESULT
020D  D91F      0128          STR     A,SECBUF,X
020E  0061
020F  072E      0129          INCR    X
0210  D99F      0130          STR     B,SECBUF,X
0211  0061
              0131  *
```

```
0212  072E      0132          INCR    X
0213  0126      0133          SUBVC   X,320   SECT.BUF.COMPL.?
0214  0140
0215  25CC      0134          SKN     CONT    NO
0216  640F      0135          IOS     WRITE   YES  WRITE BUF.OUT
0217  81A3
0218  074E      0136          INCR    Y
0219  E85F      0137          CMR     Y,SRANGE  S.-RANGE COMPL.?
021A  01A6
021B  27C0      0138          SKM     LOOP    NO
021C  7414      0139          JMP     E$MON   YES
                0140  *
021D  6408      0141  CODERR  IOS     EFLAG1
021E  802D
021F  73FC      0142          JMP     $-3
0220  6405      0143  ERROR   IOS     EFLAG2
0221  803B
0222  73F9      0144          JMP     $-6
0223  6402      0145  STATER  IOS     EFLAG3
0224  804C
0225  7395      0146          JMP     L1
      01B1 P    0147          END     SM1CON

0226  0000 X
0227  0000 X
0228  0000 X
0229  0000 X
022A  0061 D
022B  0000 X
022C  0000 X
022D  0000 X
022E  0000 X
022F  0000 X
0230  0000 X
0231  0000 X
    00  ERRS
 DBOS CC
?$EOD
?
```

```
                    0001  ****************************************
                    0002  *                                      *
                    0003  *                SMTAP                 *
                    0004  *                                      *
                    0005  *       WRITES SMART-FILE ON TAPE      *
                    0006  *                                      *
                    0007  *          (HAMMANS-18/06/73)          *
                    0008  *                                      *
                    0009  ****************************************
                    0010  *
                    0011          DEF     ASCERR,ERROR,HTERR,HDERR
                    0012          DEF     TRERR
                    0013          REF     IOS,F$IOR,E$MON
                    0014          REF     U$TPOS,U$HEAD,U$TMRK,U$VSM
                    0015          REF     TEXTAB,FLAG,U$TVSP,L2,U$DKT0
                    0016  *
0000                0017          DSECT
                    0018  *
0000  2080          0019  COM     WRITE   CC$,TX0
0001  0000 X
0002  8003
0003  000D          0020  TX0     DC      13,0
0004  0000
0005  C3CF          0021          TEXT    'COMMENT?-Y/N!$8D'
0006  CDCD
0007  C5CE
0008  D4BF
0009  ADD9
000A  AFCE
000B  A18D
000C  2080          0022  ASCR    WRITE   CC$,TX1
000D  0000 X
000E  800F
000F  0010          0023  TX1     DC      16,0
0010  0000
0011  C1D3          0024          TEXT    'ASCII-REC.?-Y/N!$8D'
0012  C3C9
0013  C9AD
0014  D2C5
0015  C3AE
0016  BFAD
0017  D9AF
0018  CEA1
0019  8DA0
001A  2040          0025  ANSW    READ    CC$,BUFFER
001B  0000 X
001C  801D
001D  0002          0026  BUFFER DC      2,0
001E  0000
001F                0027          DS      1
0020  2080          0028  EBCR    WRITE   CC$,TX2
0021  0000 X
0022  8023
0023  0011          0029  TX2     DC      17,0
0024  0000
0025  C5C2          0030          TEXT    'EBCDIC-REC.?-Y/N!$8D'
0026  C3C4
0027  C9C3
0028  ADD2
0029  C5C3
002A  AEBF
002B  ADD9
002C  AFCE
002D  A18D
002E  2080          0031  ETX1    WRITE   CC$,TX3
002F  0000 X
0030  8031
0031  000D          0032  TX3     DC      13,0
0032  0000
0033  8AC9          0033          TEXT    '$8AINPUT-ERROR$8A$8D'
0034  CED0
0035  D5D4
0036  ADC5
0037  D2D2
0038  CFD2
0039  8A8D
```

```
003A  2080        0034  ETX2    WRITE   CC$,TX4
003B  0000 X
003C  803D
003D  0016        0035  TX4     DC      22,0
003E  0000
003F  8AC9        0036          TEXT    '$8AILLEGAL'
0040  CCCC
0041  C5C7
0042  C1CC
0043  A0C1        0037          TEXT    ' ASC-CHARACTER$8D'
0044  D3C3
0045  ADC3
0046  C8C1
0047  D2C1
0048  C3D4
0049  C5D2
004A  8DA0
004B  2080        0038  ETX3    WRITE   CC$,TX5
004C  0000 X
004D  804E
004E  0017        0039  TX5     DC      23,0
004F  0000
0050  8AC9        0040          TEXT    '$8AINVALID DEC'
0051  CED6
0052  C1CC
0053  C9C4
0054  A0C4
0055  C5C3
0056  C9CD        0041          TEXT    'IMAL NUMBER$8D'
0057  C1CC
0058  A0CE
0059  D5CD
005A  C2C5
005B  D28D
005C  2080        0042  ETX4    WRITE   CC$,TX6
005D  0000 X
005E  805F
005F  0015        0043  TX6     DC      21,0
0060  0000
0061  B8C1        0044          TEXT    $8AHEADER/TRAILER-ERROR$8D'
0062  C8C5
0063  C1C4
0064  C5D2
0065  AFD4
0066  D2C1
0067  C9CC
0068  C5D2
0069  ADC5
006A  D2D2
006B  CFD2
006C  8DA0
006D  2080        0045  ETX5    WRITE   CC$,TX7
006E  0000 X
006F  8070
0070  000D        0046  TX7     DC      13,0
0071  0000
0072  8AC8        0047          TEXT    '$8AHEADER-ERROR$8D'
0073  C5C1
0074  C4C5
0075  D2AD
0076  C5D2
0077  D2CF
0078  D28D
0079  2080        0048  ETX6    WRITE   CC$,TX8
007A  0000 X
007B  807C
007C  000E        0049  TX8     DC      14,0
007D  0000
007E  8AD4        0050          TEXT    '$8ATRAILER-ERROR$8D'
007F  D2C1
0080  C9CC
0081  C5D2
0082  ADC5
0083  D2D2
0084  CFD2
0085  8DA0
      0086        0051  DPART   EQU     $-$$
                  0052  *
```

```
                    0053  *
                    0054  *
0000                0055        PSECT
                    0056  *
0000                0057        DS      DPART
0086  0402          0058  SMTAP INH
0087  01D5          0059        LDV     D,$$
0088  0000 D
0089  643E          0060        JSR     U$IPOS
008A  643E          0061        JSR     U$HEAD
008B  0135          0062        LDV     X,-1      SET CODE-INDEX
008C  FFFF
008D  643C          0063  TXT1  IOS     COM
008E  8000
008F  643A          0064        IOS     ANSW      INP.Y OR N
0090  801A
0091  0195          0065        LDV     B,'N '
0092  CEA0
0093  E89F          0066        CMR     B,BUFFER+2        COMMENT?
0094  001F
0095  2C0C          0067        SKZ     NO        NO
0096  0195          0068        LDV     B,'Y '
0097  D9A0
0098  E89F          0069        CMR     B,BUFFER+2        COMMENT?
0099  001F
009A  2C01          0070        SKZ     YES       YES
009B  7017          0071        JMP     EFLAG
009C  072E          0072  YES   INCR    X
009D  0935          0073        RTR     X,X
009E  240D          0074        SKN     EBC
009F  642A          0075  TXT2  IOS     ASCR
00A0  800C
00A1  73ED          0076        JMP     TXT1+2
00A2  0935          0077  NO    RTR     X,X
00A3  260C          0078        SKM     DKT0
00A4  240B          0079        SKN     DKT0
00A5  D03F          0080        STR     X,FLAG    RESET ASC-FLAG
00A6  0000 X
00A7  6423          0081  ASC   JSR     U$VSM
00A8  072E          0082        INCR    X
00A9  6420          0083  TXT3  IOS     EBCR
00AA  8020
00AB  73E3          0084        JMP     TXT1+2
                    0085  *
00AC  0126          0086  EBC   SUBVC   X,1
00AD  0001
00AE  2DF8          0087        SKZ     ASC
00AF  641C          0088        JSR     U$TVSP
00B0  641C          0089  DKT0  JSR     U$DKT0
00B1  641C          0090        JSR     U$TMRK
00B2  741C          0091  RET   JMP     E$MON
```

```
                        0092  *
00B3  6416              0093  EFLAG  IOS      ETX1
00B4  802E
00B5  0935              0094         RTR      X,X
00B6  27D6              0095         SKM      TXT1
00B7  2DE7              0096         SKZ      TXT2
00B8  73F0              0097         JMP      TXT3
                        0098  *
00B9  6410              0099  ASCERR IOS      ETX2
00BA  803A
00BB  73F6              0100         JMP      RET
00BC  640D              0101  ERROR  IOS      ETX3
00BD  804B
00BE  7411              0102         JMP      L2
00BF  640A              0103  HTERR  IOS      ETX4
00C0  805C
00C1  740D              0104         JMP      E$MON
00C2  6407              0105  HDERR  IOS      ETX5
00C3  806D
00C4  740A              0106         JMP      E$MON
00C5  6404              0107  TRERR  IOS      ETX6
00C6  8079
00C7  7407              0108         JMP      E$MON
      0086 P            0109         END      SMTAP

00C8  0000 X
00C9  0000 X
00CA  0000 X
00CB  0000 X
00CC  0000 X
00CD  0000 X
00CE  0000 X
00CF  0000 X
00D0  0000 X
    00  ERRS
 DBOS CC
?$EOD
?
```

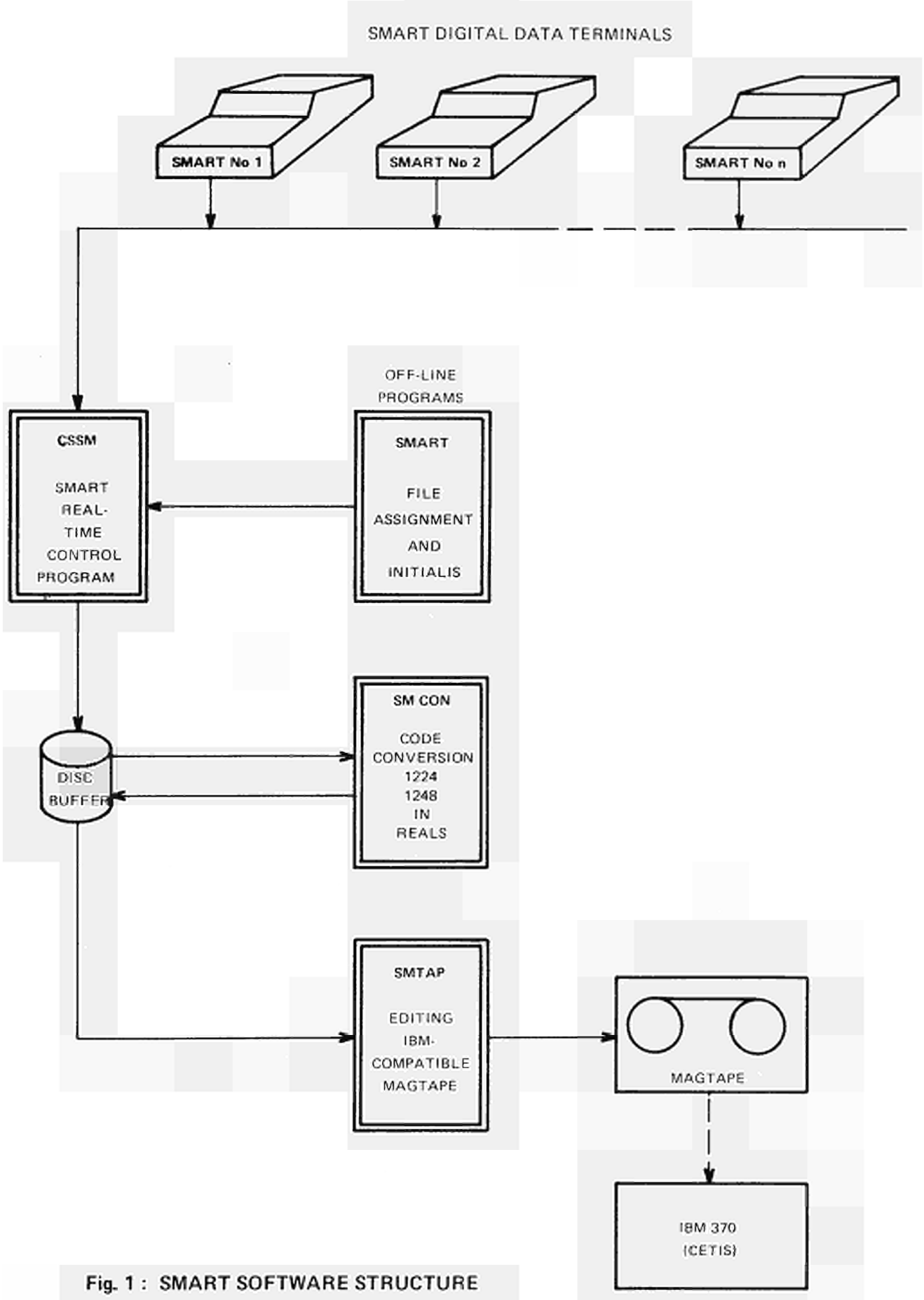



Fig. 1 : SMART SOFTWARE STRUCTURE

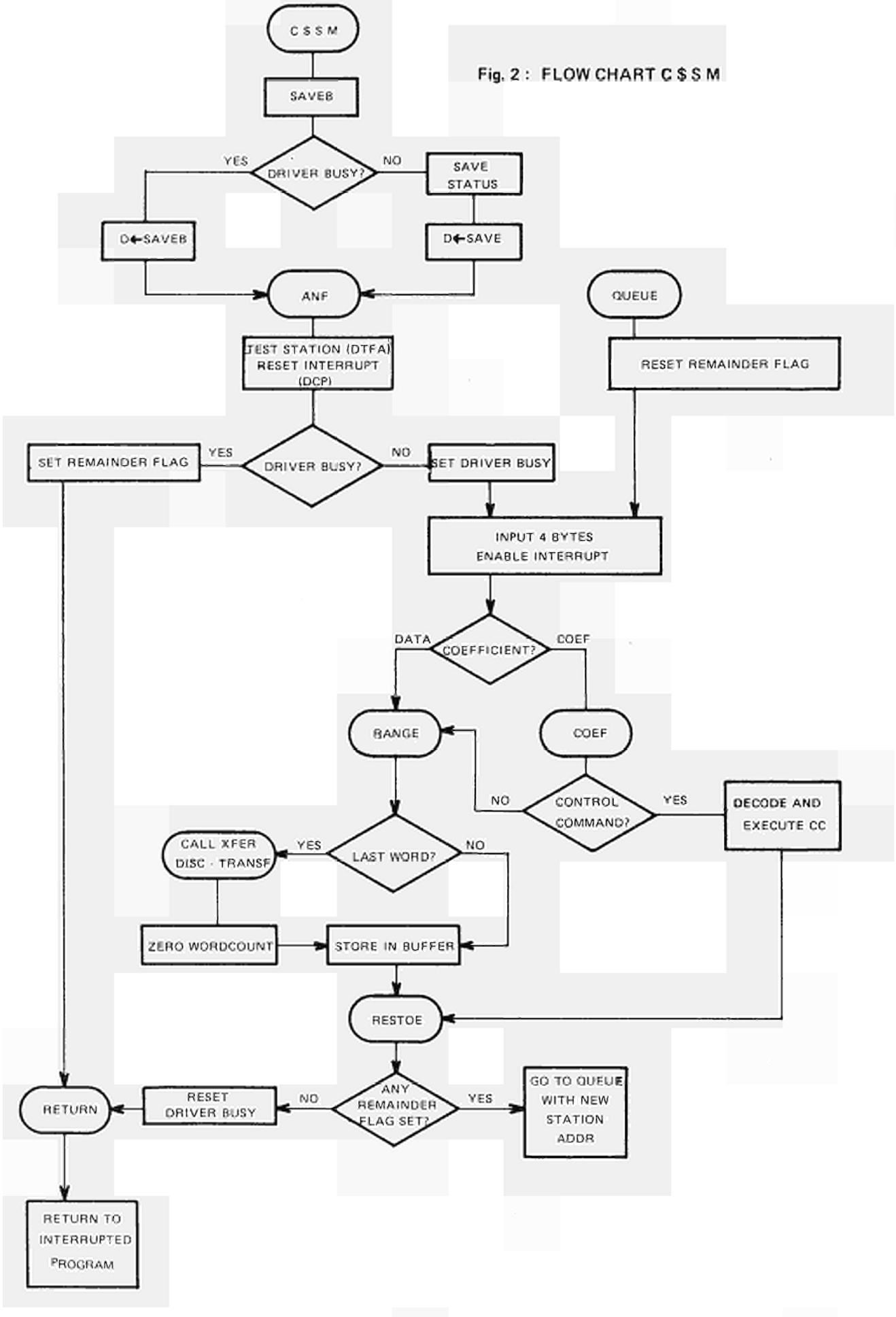



Fig. 2 : FLOW CHART C $ S M

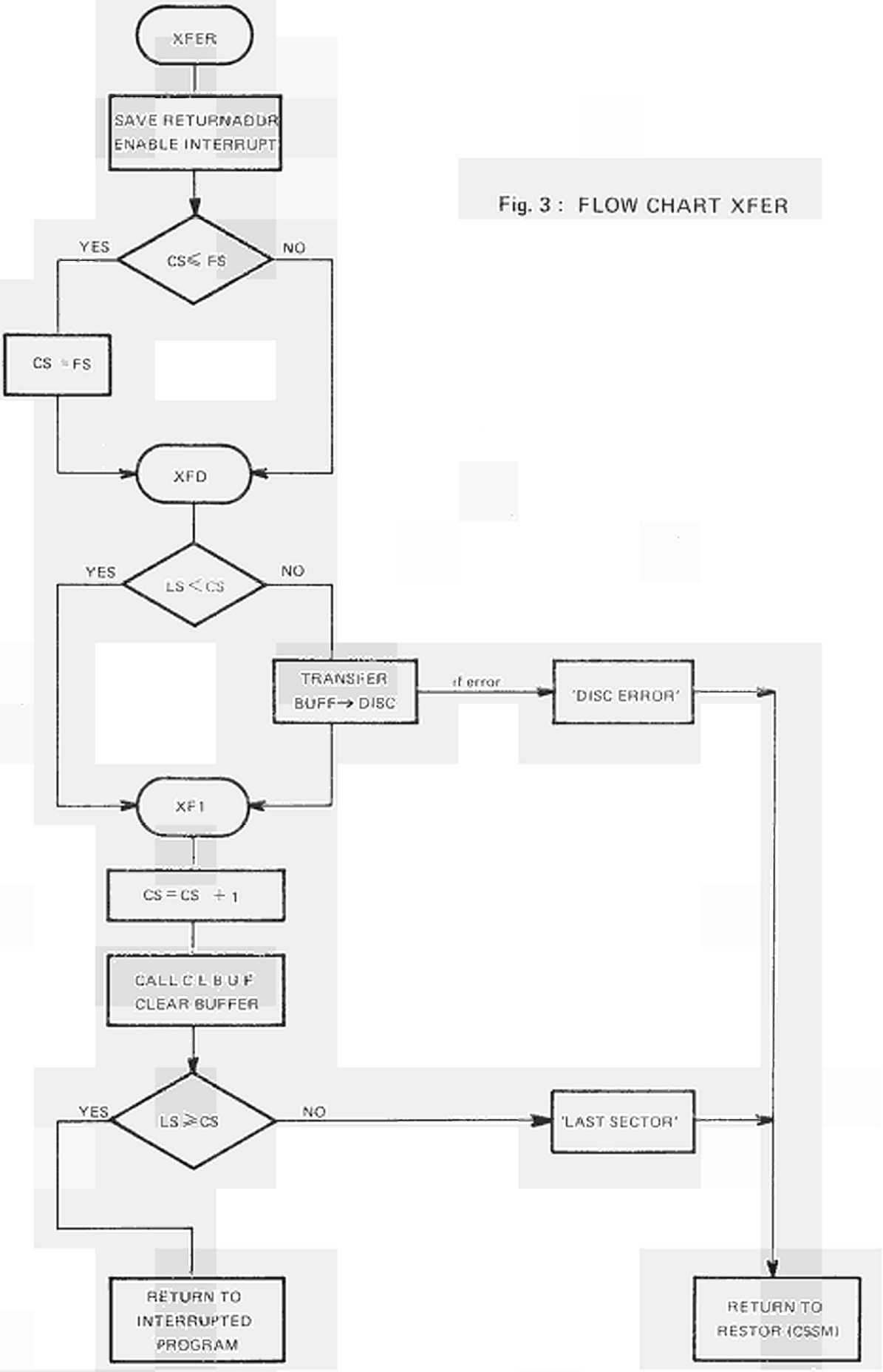



Fig. 3 : FLOW CHART XFER

FS = FIRST SECTOR OF DISC FILE
CS = CURRENT SECTOR
LS = LAST SECTOR

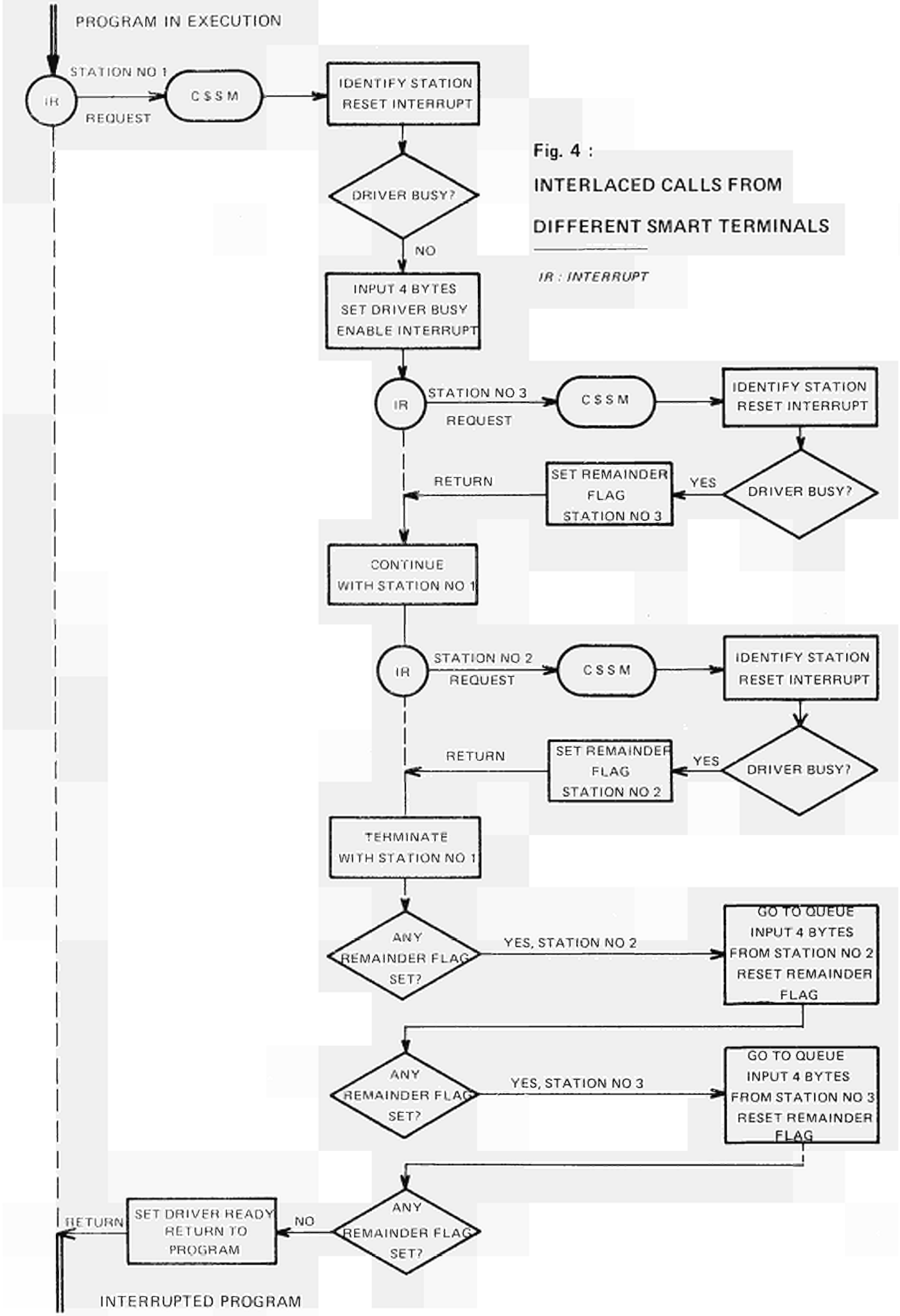



**Fig. 4 :**

**INTERLACED CALLS FROM DIFFERENT SMART TERMINALS**

*IR : INTERRUPT*

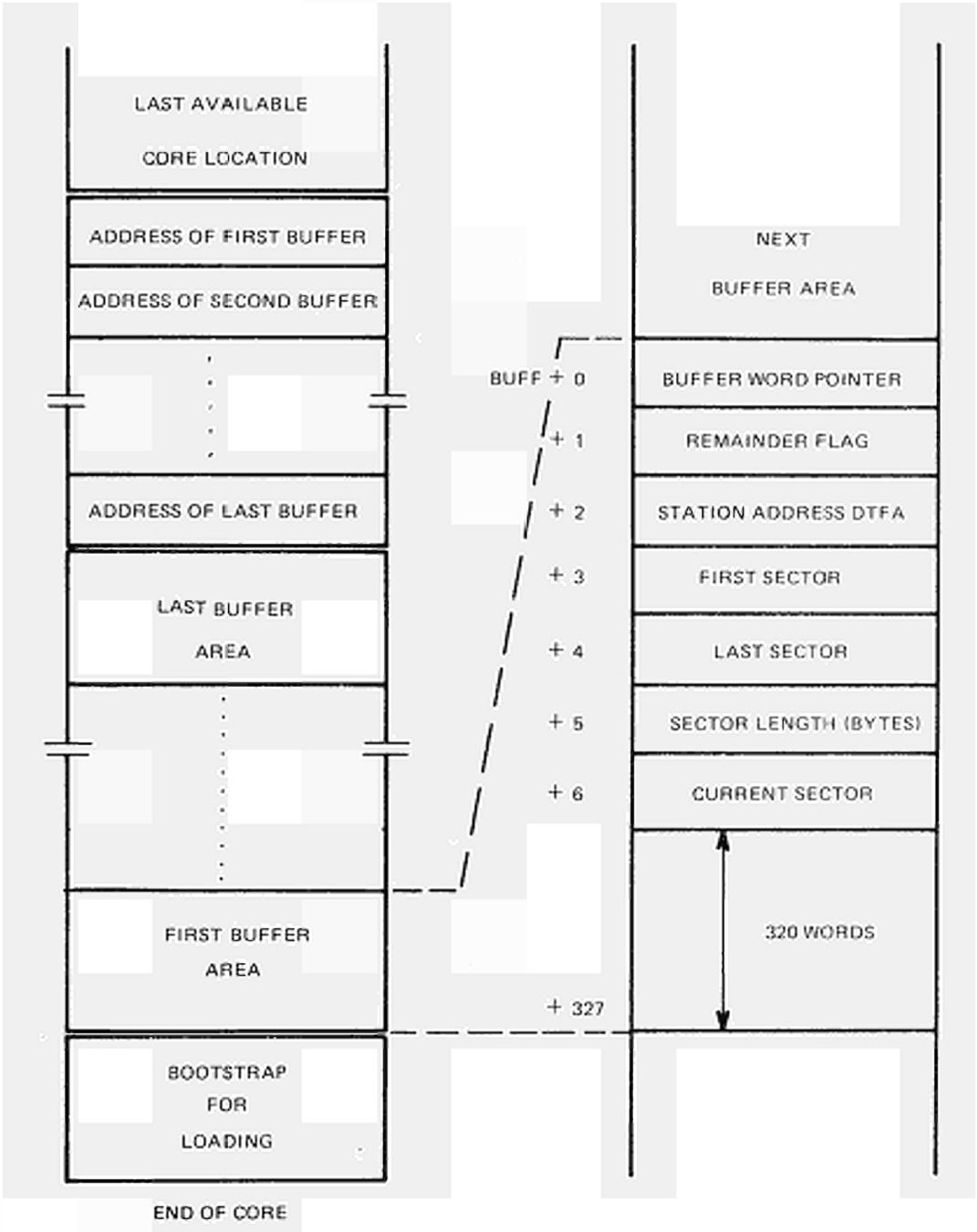



Fig. 5 : STRUCTURE OF CORE-RESIDENT BUFFERS

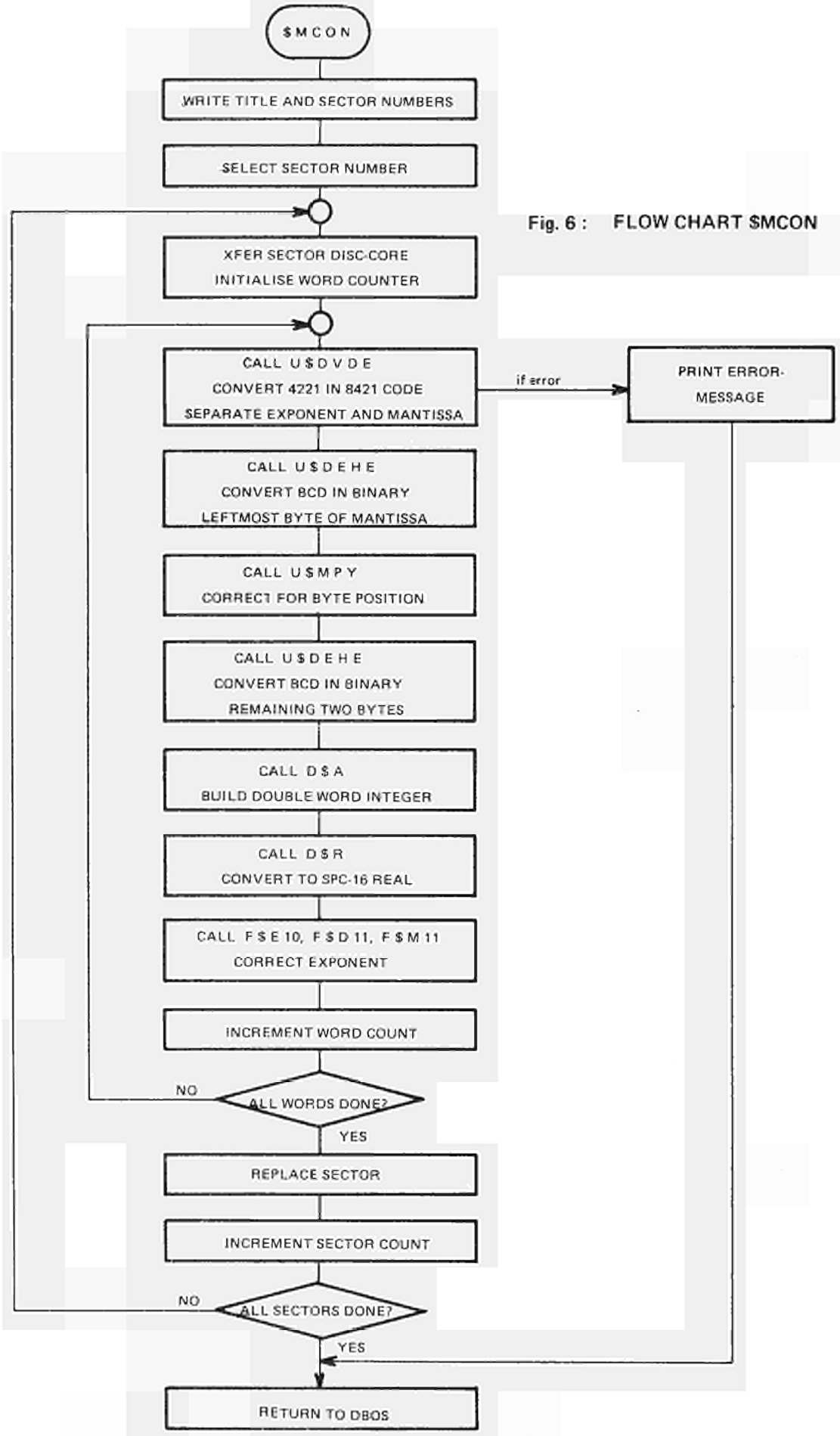



Fig. 6 : FLOW CHART $MCON

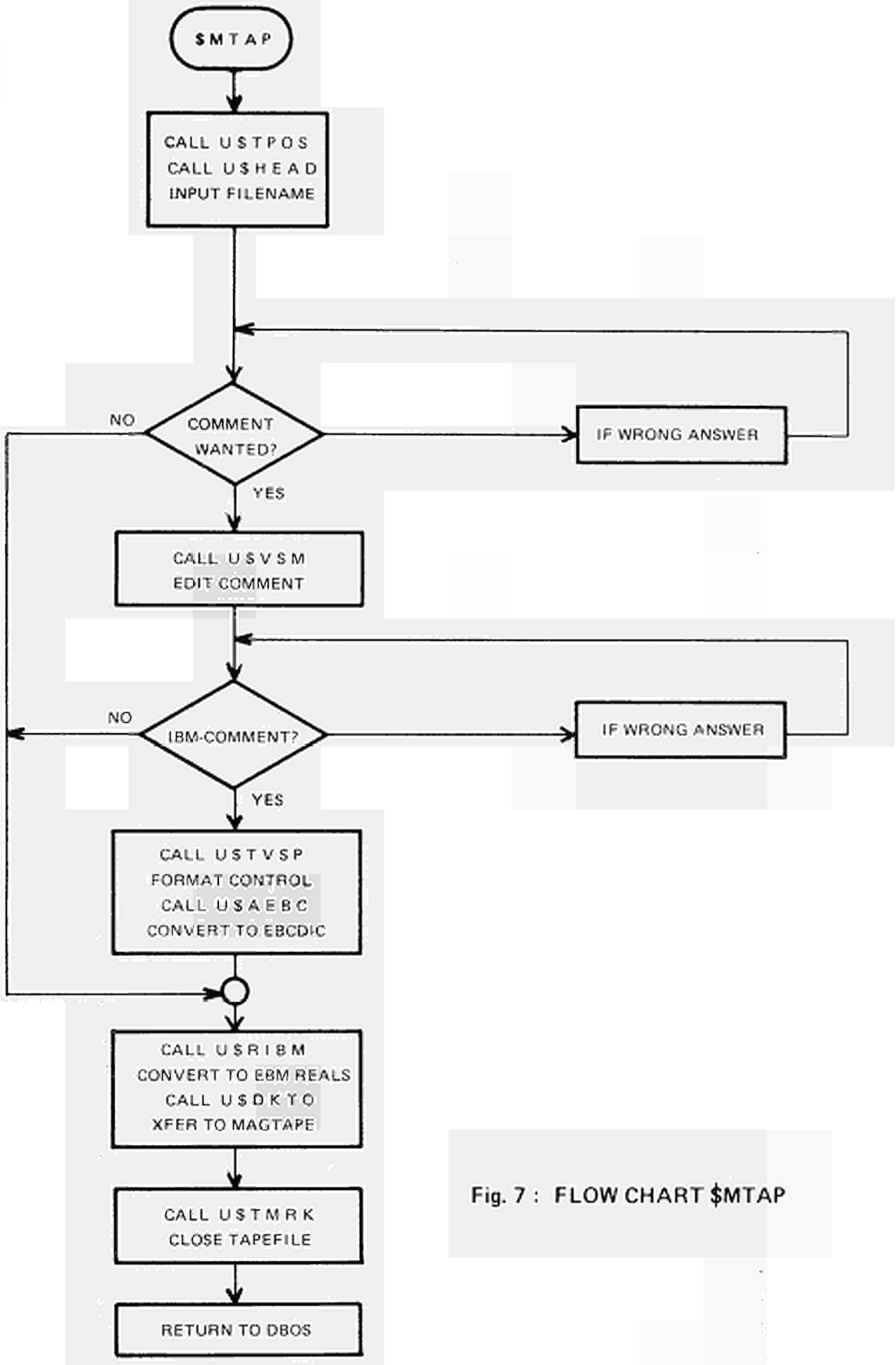
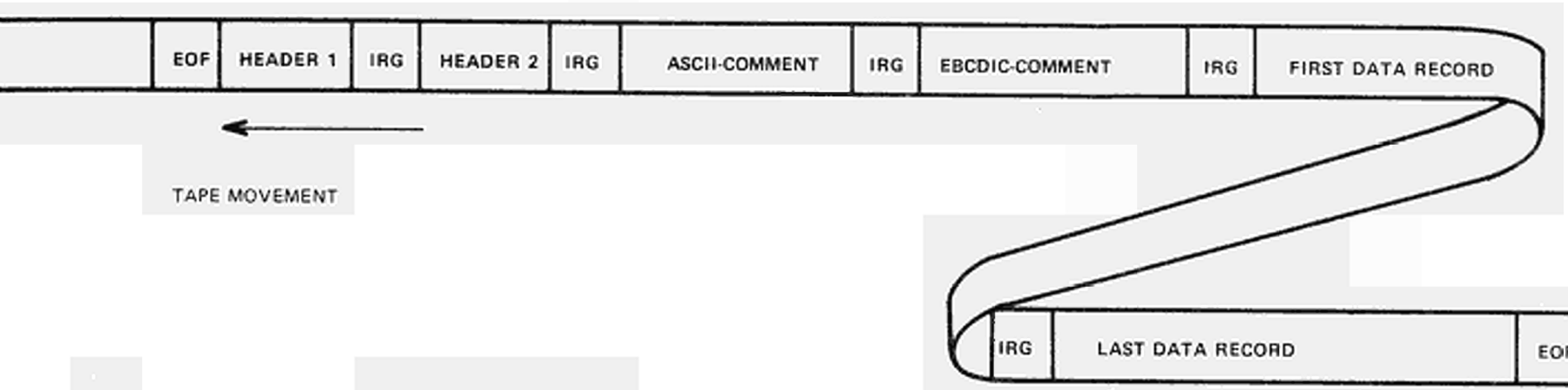



Fig. 7 : FLOW CHART $MTAP



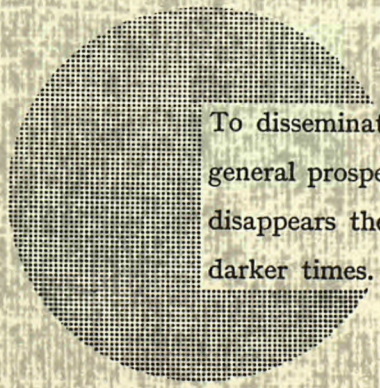

| | |
|---|---|
| EOF | End of file tape mark |
| HEADER 1 | 3 words (6 bytes) preamble (binary) $3_{16}$ , $0_{16}$ , $0_{16}$ |
| HEADER 2 | 3 words (6 bytes) filename (in ASCII) |
| ASCII COMMENT | up to 132 bytes comment (in ASCII) |
| EBCDIC COMMENT | EBCDIC equivalent comment |
| DATA RECORD | 640 bytes records with IBM short floating format values ( 160 real* 4 ) |
| TRAILER | 3 words (6 bytes) postamble (binary) $0_{16}$ , $0_{16}$ , $0_{16}$ if last file, else equal HEADER 1 |
| IRG | inter record gaps |

**Fig. 8 : SPC - 16 – IBM COMPATIBLE TAPE FORMAT (9 track tape record)**

## 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: B.Fr. 1 025,—) or free specimen copies please write to:

**Office for Official Publications**
**of the European Communities**
**Boîte 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*
Boite postale 1003 — Luxembourg 6171
Tel. 4 79 41 — CCP 191-90
Compte courant bancaire: BIL 8-109/6003/200

## IRELAND

*Stationery Office — The Controller*
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*
Boite 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
Boite postale 1003 — Luxembourg 6506