

PRL

TECHNICAL NOTE

TN- 89 - 66

A Z 80 BASED GENERAL PURPOSE  
LABORATORY MICROCOMPUTER

R. N. MISRA

AND

I. T. KRIPALANI

SEPTEMBER, 1989

PHYSICAL RESEARCH LABORATORY  
NAVRAANGPURA  
AHMEDABAD - 380 009

DOCUMENT CONTROL AND DATA SHEET

- 
1. Report No. : PRL-TN-89-86
- 
2. Title and Subtitle : 3. Report Date : September 1989  
A Z 80 BASED GENERAL 4. Type of Report : Technical Report  
PURPOSE LABORATORY 5. Pages : 126  
MICROCOMPUTER 6. Price : Unpriced  
7. No. of References : 11
- 
8. Author(s) : R. N. MISRA & I. T. KRIPALANI
- 
9. A Purpose : Design details of Z 80 Microcomputer,  
Software listings,  
B Useful For : A users manual for Z 80 Microcomputer  
and text for Z-80 based design
- 
10. Originating Unit : Planetary Atmospheres & Aeronomy Divin.  
Division & Address : Physical Research Laboratory,  
Navrangpura, Ahemdabad - 380009
- 
11. Sponsoring Agency : None
- 
12. Abstract : Design and fabrication details of Z-80  
based Microcomputer has been provided  
in the document. The Microcomputer  
is self contained and can be used  
for general laboratory application.  
Software has been explained in details
- 
13. Key-words : 1. Micro-processor. 2. Micro-computer  
3. Z - 80
- 
14. Distribution Statement : Distribution Unlimited
- 
15. Security classification : Unclassified

## I N D E X

1.	Abstract ..	1
2.	Introduction ..	2
3.	Design ..	5
3.1	Clock Generator Logic ..	6
3.2	Z-80 Buffer & Memory I/O Select ..	8
3.3	CRT Controller & Associated Circuitry ..	10
3.4	Memory and I/O ..	12
3.5	Power Supply, RS232C Interface and Cassette Tape I/O ..	14
3.6	Keyboard ..	15
4.	Software ..	16
5.	Acknowledgements ..	27
6.	References ..	28

### TABLES :

- I. Display Parameters
- II. Contents of Registers of Video Display  
Controller 6845
- III. Bus Details
- IV. I/O Expansion Bus
- V. Serial Printer Baud Rate Control.

## APPENDICES

- A. PARALLEL PORT ASSIGNMENT
- B. 8251 BAUD RATE CLOCK PAD LOCATION
- C. I/O CONNECTIONS ON IC -16 SOCKET
- D. I/O CONNECTIONS ON IC -20 SOCKET
- E. CONNECTIONS OF POWER SUPPLY CARD
- F. LIST OF SUBROUTINES AND ENTRY POINT FOR SERIAL MONITOR PROGRAM
- G. DISSEMBLED SERIAL MONITOR PROGRAM
- H. Hex DUMP OF SERIAL MONITOR PROGRAM
- I. LIST OF SUBROUTINES AND ENTRY POINTS FOR KEYBOARD MONITOR PROGRAM
- J. DISSEMBLED KEYBOARD MONITOR PROGRAM
- K. Hex DUMP OF KEYBOARD MONITOR PROGRAM
- L. FLOW CHARTS OF PROGRAM MOVE AND DISSEMBLED PROGRAM.
- M. Z - 80 DISASSEMBLER
- N. TINY BASIC COMMANDS
- O. PIGGYBACK RAM
- P. LIST OF COMPONENTS
- Q. CONNECTOR OF KEY BOARD CARD

## LIST OF ILLUSTRATIONS

1. BLOCK DIAGRAM
- 2 . Z-80 BUFFERS, MEMORY & I/O SELECT LOGIC.
3. MEMORY AND I/O MAP.
4. MEMORY AND I/O CIRCUIT DIAGRAM.
5. POWER SUPPLY, BATTERY BACKUP OF RAM & RS232C  
INTERFACE
6. KEYBOARD ORGANIZATION.
7. CLOCK AND VIDEO DISPLAY GENERATOR.
8. CASSETTE TAPE RECORDER I/O.
9. PCB LAYOUT AND PLACEMENT OF PARTS.
10. PCB LAYOUT OF POWER SUPPLY CARD.
11. PHOTOGRAPH OF THE INSTRUMENT.

ABSTRACT

General purpose 8 bit micro-computer is quite often needed for laboratory automation and control purposes. A micro-computer based on Z-80 microprocessor has been designed for such applications. It has utility software like disassembler, programme relocate, EPROM programmer etc. resident in the monitor and a tiny basic has also been provided on separate EPROM. It can accommodate full 64k byte RAM/ROM onboard and thus obviates need for any memory expansion card. 16k byte RAM is in battery backup mode to serve as short term storage and audio cassette I/O has been provided for mass storage. A buffered I/O interface port is used to connect the external control hardware to the microcomputer. CRT controller and associated video circuitry are built-in in the system and a simple 47 key keyboard has been provided. Only an ordinary TV receiver is required for the display. Design, construction and software details have been given in this document.

## 2.0 INTRODUCTION

The first 8 bit microprocessor 8080, introduced by INTEL in 1973, paved the way for the rapid development of microprocessor based instrumentation. After some time, some engineers of INTEL, who were involved in the development of 8080, left to form their own company ZILOG, enhanced the 8080 substantially and introduced it as Z-80. The new microprocessor was fast, could run all the programs written for 8080 and had many more extra instructions. It became so successful that it commands 1/3rd of the whole 8 bit microprocessor market as of today<sup>1</sup> (EDN, Nov. 27, 1986). INTEL's answer to Z-80, i.e. 8085 is nowhere near Z-80 in performance. Although the latter also commands substantial share of 8 bit microprocessors, and new comers are usually introduced to the world of microprocessors through 8085, it is advisable for the users to go for Z-80 microprocessor even if it takes a little more time to understand and is slightly more complex both hardware and software wise. 8080 type control signals can easily be derived through extra hardware and use of the additional instructions of Z-80 may be deferred till the user gets familiar to the other instructions. Hardware design is usually implemented

through 74LS series of IC's. The CMOS version of Z-80 and its peripherals are available which can be used along with 74 HCT series of CMOS IC's to reduce the power consumption of the system. One usually goes for 8080 peripherals even if those can not use the special facilities provided by the Z-80.

IBM PC's and its clones have proliferated and pushed the other small computers based on standard 8 bit microprocessors towards obsolescence. INTEL also suggests 8088 to be used as a replacement for 8085. However, the instruction set of 8086/8088 is quite complex and therefore it is not simple to use for hand assembly. In this comparison, the Z-80 instructions set is simple and yet effective for most of the applications.

A large number of utility programs have been written or adapted from 8085 based microcomputer. These were used for program development on systems similar to SDK-85. These include disassembler, Tape I/O, Tiny basic, Relocator, EPROM programmer etc. A touch keyboard type alpha-numeric terminal was also developed by the authors.<sup>2</sup> These various programs



have been consolidated together to form a powerful Z-80 monitor useful for laboratory applications. The necessary hardware including the CRT controller and video logic has been accommodated on a single board. A simple keyboard based on SPST keys, has been designed which interfaces to the microcomputer through parallel ports. These two boards of 12" x4.5" size and a small board of 6"x3" for the power supply, are all that is needed to assemble one's own microcomputer which is versatile enough to be used for various laboratory applications.

The detailed design has been described in the following sections. Circuit diagrams and software have been included in the Appendix. Guide-lines for operation and Printed Circuit board layout have also been included to facilitate fabrication of the unit by the reader. This document is intended to be used as operation and Technical Manual of the equipment.

### 3.0 DESIGN

Circuit simplicity and versatility had been the basic goals while designing the system. The whole microcomputer along with the CRT controller as well as associated video circuitry has been accommodated on a single printed circuit board to keep the mounting hardware to a minimum. The keyboard PCB is also of the same size and fits over it with the help of a set of six 4 mm bolts, nuts and spacers. Power supply board is separate and contains RS-232C and Tape Recorder Interface as well. All these PCBs along with the power transformer have been accommodated in a sheet metal box of double U construction. The total circuit can be divided in the following parts :

1. Crystal oscillator, timing chain, video shift register and associated logic,
2. Z-80, signal buffers and control bus signal generators,
3. CRT controller and associated circuits,
4. Memory and I/O,
5. Power supply, RS232C and cassette tape I/O interface.
6. The keyboard.

These have been described in separate sections in the following pages.

### 3.1 CLOCK GENERATOR LOGIC

The microcomputer requires a number of clock signals for its proper operation. these have been listed below :

1. Microprocessor clock,
2. System clock for serial port 8251,
3. System clock for CRT controller 6845,
4. A set of baud rate clocks for 8251 covering different baud rates.
5. Dot clock and
6. Load pulse for video shift register.

Block diagram of clock generator has been given in Fig.7A. A single crystal oscillator based on 7404 inverter and 8 MHZ crystal has been used to generate all the above signals. It is passed through two divider chains. One chain comprising of synchronous counter 74191 is used to generate the load pulse for the video pulse generator. As the character format for the system is 5x7 pixels, 74191 has been configured for divide by 6 operation, in down count mode. The ripple clock output is connected to load input and Max/Min output is used as load pulse. The output at Qc is a pulse train of 1.33 MHZ and is used as character clock for 6845 and system clock for 8251. 4 MHZ would have exceeded 8251's rating. All the parallel inputs of the 74191 have been connected to ground on the printed circuit board in such a way that those can be selectively cut to provide

logical '1' to the parallel inputs. As the dot counter is a divide by 6 circuit, inputs to B&C are disconnected from ground to provide '6' to the parallel input.

The second timing chain consists of two 4 bit binary counter type 7493 and one 4 bit synchronous counter also of type 74191, configured for divide by 13 operation by removing track on A,C & D inputs in a manner similar to divide by 6 circuit. 8 MHZ crystal oscillator output is divided by 2 to get 4 MHZ system clock for Z-80 and subsequently by 13 to get the basic baud rate clock of 307.2 KHZ. This output is further subdivided to get baud rate clock for baud rate of 150, 300, 600, 1200, 2400, 4800 & 9600 in x16 mode of 8251. If the dot clock is to be increased to change the display format, it would become necessary to separate the two chains. In that case a separate 4 MHZ clock oscillator for the microprocessor would be needed. Printed circuit track can be easily cut at 'N' to achieve this configuration.

The Z-80 requires a system clock which goes close to 5v. It has been realised by connecting a 330 Ohm resistor on the output terminal of the buffer gate. All the baud rate clock outputs have been brought on the pads on the PCB to facilitate jumper or switch selection of baud rate. The circuit diagram of the clock generator has been given in Fig. 7B.

### 3.2 Z-80 BUFFER AND MEMORY I/O SELECT LOGIC

The circuit diagram of this section has been given in Fig.2. The Z-80 outputs for control signals  $\overline{RD}$ ,  $\overline{WR}$ ,  $\overline{IORQ}$ ,  $\overline{MREQ}$  and  $M1$  to specify Memory, I/O or interrupt operations. These have been converted into 8080 like control signals  $\overline{MR}$ ,  $\overline{MW}$ ,  $\overline{IOR}$ ,  $\overline{IOW}$ , and  $INTA$  with the help of TTL gates. Interrupt and NMI signals have been inverted to make them also similar to interrupt signals of 8080. Due to some difference between Z-80 and 8080 in so far as I/O operations are concerned, upper and lower bytes of Address lines do not carry the same information and I/O address has to be exclusively derived from the lower Address lines A0-A7 and  $\overline{IORQ}$  signal has to be used for enabling I/O devices. A separate I/O decoder has been realised by 3 to 8 decoder type 74LS138. There is a separate Memory decoder chip 74LS138 which generates eight Memory select signals covering 8K byte each. The uppermost 8K byte is further sub-divided into E0, E8, F0 & F8 address space with the help of another 74LS138. The latter is enabled with  $\overline{MREQ}$  signal to avoid interaction between memory and I/O operation. CRT controller is located at F8 in memory space and F000-F7FF have been occupied by the display RAM.

Data and address buses have been buffered with the help of 74LS245 and two 74LS244's respectively. Bus request and Acknowledge signals have also been made compatible to HOLD and HLDA of 8080/8085 but as all the control signals have not been gated through BUSACK, DMA operation is not possible on the present system without substantial hardware changes.

### 3.3 CRT CONTROLLER AND ASSOCIATED CIRCUITRY

The circuit diagram of CRT controller has been given in Fig.7B. It is based upon popular 6845 of Motorola. The  $R/\bar{W}$  line is joined to  $\bar{M}\bar{W}$  line of system bus and  $\bar{E}$  signal has been generated by NAND'ing  $\bar{M}\bar{W}$  &  $\bar{M}\bar{R}$  lines. 6845 is located at F800 in the memory space and its individual register may be accessed at F800 and the contents to be loaded there in at F801. The address wraps up to FFFF and this address space is not available for any other use. A separate data buffer isolates 6845 from data bus. This is enabled by F800 decode line and direction is controlled by  $\bar{M}\bar{W}$  line.

6845 generates 14 address lines MA0-MA13 for the video RAM. Only 11 lines MA0-MA10 have been used and those are multiplexed with the system data bus A0-A10 with the help of 3 units of quad 2 input multiplexer type 74157. The Z-80 communicates to the display RAM which is 2Kx8 RAM type 6116 located at address F000, with the help of F000 decode signal from 74LS138 applied to pin 1 of all the 74157. The data bus buffer 74LS245 used for 6116 is also enabled by the same line. Data direction is controlled by  $\bar{M}\bar{W}$  line as in case of 6845. The 6116 is always enabled and read and write is performed by  $\bar{W}\bar{E}$  pin (21) which is

driven by  $\overline{FO} + \overline{MW}$  signal. The data output of 6116 is latched in 74LS374 with the LD pulse generated in 74191 dot clock counter, which also loads video data from the character generator 2716, into shift register type 74165.

The Row select lines R0, R1 & R2 are connected to A0, A1 & of 2716 and data at the output of the latch 74LS374 to A3-A10 of 2716. All the 8 outputs of 2716 have been converted to 74165 but as the dot clock counter counts to only 6, only 6 bit data is displayed on screen.

The 74165 is driven by dot clock (8 MHZ) and its serial output is passed into XOR gate to be gated through the DISPTMG (Display Timing) signal from 6845 which has been delayed by two character locations by 74175. The output of XOR gate is passed through a NAND gate along with the CUDISP (Display Timing) signal also delayed by two character location as in case of DISPTMG signal. The output of the NAND gate is video signal which is mixed with horizontal and vertical sync signals available at 6845 pins to generate the composite signal. The video signal can be inverted for reverse video via inversion of DISPTMG signal in a XOR gate.



### 3.4. MEMORY AND I/O

The static RAMs have increased in density to substantial extent and 8Kx8 RAMs of type 6264 are easily available. These have been used and no attempt has been made to use dynamic RAM which would not have served much purpose. The memory and I/O map of the system has been given in Fig.3. and the circuit Diagram of the same in Fig.4. Only five sockets for 8Kx8 RAM/ROM have been provided and two RAM's can be stacked one over the other to utilize full 64K memory space (Appendix A). The extra static RAM's mounted on the top have been used for battery backup to save programs for short term. Left most socket on the board has been used for monitor. Two 1.5V penlight zinc carbon cells have been used for backup application. These are on trickle charge during power on and can be used for two to three years without much problems. Ni-Cd cells provide a little less voltage (1.25V) and loose charge if left for more than one or two weeks. Ordinary cells have proved to be quite cost effective in this application.

The circuit diagram of battery backup section has been included in Fig. 5. The power off condition is sensed from

the raw DC supply. The transistor T1 gets switched off at a threshold of 7V, which is well below the minimum 8 volts required by the regulator. The logic signal available at collector of T1 is inverted in T2 and used as CS2 signal for the static RAM. The signal goes logical one after a time delay but goes to zero abruptly to avoid data contamination during power ON-OFF.

Two I/O devices used on board are the programmable peripheral Interface PPI 8255 and universal synchronous asynchronous Receive Transmit (USART) 8251. These are located at 08 and 00 in the I/O space. The 8251 uses the 1.33 MHz word clock as its system clock signal. The RxD, TxD signals of 8251 and 24 I/O lines of 8255 have been brought on two 16 pin IC sockets. While Port A and Port B of 8255 have been used for keyboard, Port C lines have been utilized for Tape I/O and additional printer port. Details of these signals have been given in Appendix A.

### 3.5 POWER SUPPLY, RS232C INTERFACE & CASSETTE TAPE I/O

The circuit diagram of the power supply, Cassette I/O and RS232C interface has been given in Fig.5. The power supply is of straight forward design and + & - 12V supplies are of unregulated type. The latter have been used for RS232C converter type 1488. The cassette tape recording format and circuitry has been covered in detail elsewhere<sup>8</sup>. The 'ONE's are recorded as double pulse of 100 Micro seconds width and 400 micro seconds spacing while ZERO's are recorded as a single pulse of 100 micro seconds width. Bit to bit time is 1000 micro seconds and the last bit is followed by a 1000 micro seconds interval to indicate the start of new byte. The waveform is generated on PC1 (Appendix A) and attenuated before feeding to the mike input of the tape recorder. The signal recovery is performed by a quad operational amplifier LM324 and associated circuitry. The operational amplifiers are used as phase equalizer, zero crossing detector schmitt-trigger and LED driver respectively. The steady glow of LED indicates proper recording. The recording is fairly insensitive to the setting of volume level of the cassette tape recorder.

### 3.6 KEYBOARD

The keyboard is based upon 57 mechanical keys of SP ST type arranged in a 8x8 matrix. One of the columns is assigned to CAPS LOCK KEY which is of locking type. The rest of 56 keys are assigned to cross points of 8 rows and 7 columns. The row containing CAPS Lock Key has also been assigned to SHIFT Control and cursor control keys.

The keyboard design has been simplified by using software for keyboard scanning and checking for each key. Very good software had been developed by Padmanabhan et al<sup>9</sup>, which has been adapted for the present keyboard. Keyscan signals are of a polarity inverse to that given in the article and cursor control, CTRL and shift keys are handled in a slightly different way. Use of software for keyboard is justified in a small microcomputer as the software is usually short and the microcomputer wastes most of its time waiting for the user to enter data or command. The total keyboard software requires a little more than 256 bytes of memory. The organization of the keyboard has been given in Fig.6. while the rows are driven from port A of 8255 through a set of 8 diodes, the column lines have been pulled up by eight resistors of 4.7 K each, and are read at the port B. The software returns the ASCII code corresponding to the key pressed in register A.

#### 4.0 SOFTWARE

The Monitor has all the routines necessary to run the system and use the tiny basic interpreter. The latter is the modified version of the one described in "The Best of Interface Age" by Roger Rauskolb<sup>10</sup>. The original one was meant for the CP/M based systems while the interpreter used in the present system is based on Cassette tape recorder. As it has been stated earlier, the monitor, though based on SDK85's serial monitor, has been expanded to include many useful commands and utility programs. The list of commands in Alphabetical order have been given below

- B Byte Search
- C Compare Memory
- D Display Memory (as in SDK 85)
- F Fill Memory
- G (GO) Run Program (as in SDK 85)
- I Insert Data (as in SDK 85)
- L List (Disassemble)
- M Move Memory (as in SDK 85)
- P Program Move (Relocate)
- R Read Tape
- S Substitute Memory (as in SDK 85)
- W Write Tape
- X Exam Register (as in SDK 85)

Apart from the above commands, there are the following Utility programs :

1. EPROM Programmer I for 2716
2. EPROM Programmer II for 2716/32/32A/64/64A/128
3. Subtract 16 bit
4. Tiny Basic

The various commands and utility programs have been described below :

1. B; Byte Search : This command searches for a specified byte in the memory space indicated. Each address where this byte has been found, is printed at every SPACE Key. Command is terminated by CR.

B XXXX      YYYY      < CR >

(Start Addr.) (End Addr.)

NN      SP

(Specified byte)

Result is printed on console in the following way

PPPP              NN

(LOCATION) (DATA BYTE)

2. C; Compare : This command compares two areas of RAM space and location, the contents of which are different from destination area are printed in a way similar to B Command.

```
C      XXXX      YYYY      ZZZZ      <CR>
```

(Start Addr.)(End Addr.)(Target Start Addr.)

```
PPPP      QQ
```

(Location)(Data Byte, which is different)

The command prints each location by SP till all the memory area is covered. Or else it may be terminated by <CR>

3. D; Display : This command displays the specified memory area on the console :

```
D      XXXX      YYYY      < CR >
```

(Start Addr.) (End Addr.)

```
DOFES OFFF
```

```
OFES 00 00 A9 0C CA 0D BA 0F
```

```
OFF0 A0 0D EE 0E 97 0C C2 0C
```

is a typical print out.

4. F; Memory Fill : This command fills the specified memory area with the specified byte

```
F      XXXX      YYYY      < CR >
```

(Start Address) (End Address)

```
PP      < CR >
```

(Specified byte)

5. G; GO : This command transfers control to the present contents of program counter or else the specified location (optional)

```
G      XXXX      < CR >
```

(Specified Address)

6. I; Insert : This command is used to enter hexadecimal data in the memory location specified in the command

```
I      XXXX      < CR >
```

(Start Address)

```
PP    QQWW      ZZ
```

would enter PP QQ WW ZZ at locations starting at XXXX

The command is terminated by any illegal character (Non hex - Alphabet)

7. L; List : This command prints the disassembled program within the specified addresses :

```
L      XXXX      YYYY      < CR >
```

(Start Address) (End Address)

8. M; Move Memory : This command moves the data block contained within the first two addresses to the location starting at the third address

```
M      XXXX      YYYY      WWWW      < CR >
```

(Source Start) (Source End) (Destination)

9. P; Program Move : This program is used to relocate a section of program. All the call addresses referred to within the memory space being relocated are adjusted according to the specified run address. The addresses outside the range are not changed



P        XXXX        YYYY        ZZZZ        < CR >

(Source Start) (Source End) (Destination)

NNNN                MMMM                PPPP        < CR >

(Current Run-Start)(Current Run-End)(Destination Run)

It would relocate program within (XXXX) to (YYYY) memory space, which was originally meant to be run from NNNN to MMMM, to memory space starting at ZZZZ such that it would be capable of running at location starting at PPPP .

10. R; Read Tape : This command reads a block of data from magnetic/paper tape and loads it at the memory address, specified in the command.

R                XXXX                < CR >

(Loading Address)

No. of bytes should be indicated in the block of data.

11. S; Substitute Memory : This command displays one byte at a time which may be just examined or else modified by typing a new data.

S                XXXX                < SP >

(Address)

Command is terminated by < CR >

12. Write Tape : This command writes a block of data specified by the command on to a magnetic cassette tape/paper tape.

W                XXXX                YYYY                < CR >

(Start of Block) (End of Block)

13. X; Exam Register : This program is used to examine and modify contents of microprocessor register X < CR > would display all the registers. X A/B/C/D/E .. SP/PC <SP> would display the particular register which may be modified. Command is then terminated by <CR>

### UTILITIES

#### 1. EPROM Programmer I

This program is located from 0F00 to 0F9F. The program uses RAM locations at 20BC - 20C1 as address pointers as given below :

20 BC - 20 BD	EPROM start address
20 BE - 20 BF	Data source start address
20 C0 - 20 C1	Data source End address+1

Most significant byte is always entered first. The calls for programmer are given below.

### PROGRAM

G 0F80 , TUN ON + 25V < CR >

+25 V supply for the EPROM programme is turned on before GO command is executed. The program LED glows till all the locations are programmed. The program then returns to monitor. For example if contents of the pointers are

20 BC - 20 BD	= 0780
20 BE - 20 BF	= 3080

20 C0 - 20 C1 = 3100

The data in location 3080 to 30FF (1 less than 3100) is programmed on 0780 - 07FF locations of 2716.

### READ/COPY

G OF94 <CR>

The program would copy the content of PROM starting at address given in location 20BC - 20BD to the RAM are starting at address given in 20BE - 20BF and up to one address less than the content of 20C0 - 20C1 For example, for the contents of pointers :

20 BC - 20 BD	..	0100
20 BE - 20 BF	..	3800
20 C0 - 20 C1	..	3880

The GO OF94 command would copy the PROM data located at 0100 to 017F at RAM address 3800 to 387F which can be read by D command. The programme then returns to the monitor.

## 2 EPROM PROGRAMMER II

This program uses a different EPROM programmer hardware and can programme 2716/2732/2732A/2764 & 2764A with switch selection. The Address pointers are the same as in case of EPROM Programmer I.

### PROGRAM COMMANDS

GO OE60	CR for 2732/2732A
GO OE80	CR for 2716/2764/2764A

COPY/READ

G OE70 CR for 2732/2732A

G OE88 CR for 2716/2764/2764A

The VPP voltages are selected along with the EPROM type. The ON-OFF of VPP is done by software. A switch for EPROM as been provided which should be turned on only when EPROM is seated in the ZIF socket. The 24 pin EPROM are seated in lower 24 pins of the 28 pin socket. LEDs indicate the power on, VPP ON and programming ON. Rest of the operation is same as in EPROM Programmer I.

3. SUBTRACT 16 BIT

This program is located at OF51. The data in DE register pair is subtracted from the contents of HL register pair and the result is returned in BC register pair.

4. Tiny Basic

A tiny basic has been provided. It has to be copied at RAM location 8100 (HD) to 8BFF, Either from cassette tape orseperate EPROM through EPROM Programmer interface. Then the

G 8AA0 CR

Command transfers the control to TINY BASIC.

## CONFIGURATION AND OPERATION

### CONFIGURATION

The location of various printed circuit boards, connectors and their interconnections have been shown in Figures 9 and 10 respectively. The Keyboard mounting has also been shown in Figure 9. The microcomputer can be configured to work in stand alone mode along with a monochrome monitor or black and White TV. The scanning rates as given in the Table I closely conform to CCIR B. Non-interlaced scanning is used but it has no effect on the synchronization. The video signal available at the connector is of positive going type and care is to be taken to properly interface it to TV or monitor as the case may be. The keyboard monitor program given in Appendix J has to be used in this configuration. On reset it behaves like a usual microcomputer with its star (\*) prompt. All the monitor commands described in section 4.0 are available in this mode. The microcomputer can be used as a CRT terminal by running the programs located at 0050 for line mode and that at 00B0 for local mode. Keyboard monitor EPROM type 2764 has to be inserted in socket IC23 for these stand alone modes. The main RS232C serial port available in the microcomputer is to be used for communication to the host computer, and the

auxiliary serial port for serial printer. Print screen command invoked by CTRL-O is used to dump the screen data on Printer. More details on this operation can be seen from Reference 2. The Reference 9 can be seen for further command usage.

If stand alone mode is not needed the microcomputer can be configured for operation with separate serial terminal (CRT or TTY). The main RS232C port, in this case, is used for communication to the terminal. Keyboard need not be wired and ports PA & PB remain free for the user. Even 6845 and related circuitry may be dispensed with if display is not required. The microcomputer needs the serial monitor EPROM 2764 at location 0000(H) shown by socket IC23 in the artwork. The contents of serial monitor have been given in Appendix G.

#### OPERATION :

At reset the microcomputer works in monitor mode and displays \* prompt. Cassette tape recorder can be used for storage of data and mike, speaker and Remo jacks corresponding to those available at the domestic mono-cassette tape recorders have been provided. Cassette tape routines are located at 0200 (H) in the monitor. It

contains facility for motor ON-OFF as well. More details of cassette tape operation can be had from Reference 8. EPROM programmer software is built in. A small hardware interface would be needed to use it. The author proposes to describe this interface in details in a technical note to be published shortly.

The tiny basic requires less than 3K bytes of memory space. The serial monitor is shorter and can accommodate the tiny basic. However, the keyboard subroutines of keyboard monitor occupy sub-stancial memory space and it is necessary to look for other mode of storage. It is not possible to include details of tiny basic in this document which however will be presented in a technical note being prepared at present.

5.0

ACKNOWLEDGEMENTS

The hardware and software described in this document has been developed in bits and pieces over a period of 3 years. Authors are thankful to Miss C. Latha who contributed towards the Z-80 disassembler during her term as a Programmer Trainee in the year 1985-86. Mr. J.K. Dave contributed towards the fabrication of electronic hardware and Mr. M.N. Trivedi helped us in providing a suitable enclosure for the instrument. Appreciation for the work presented here, by Professor Satya Prakash, B.H. Subbaraya and Vijaykumar, helped the authors a great deal to bring the instrument in this form.



## 6.0 REFERENCES

1. EDN Nov.27,1986 Special Issue on Microprocessors and Microcomputers.
2. A low cost Touch Keyboard CRT Terminal, R.N. Misra and I.T.Kripalani, PRL Technical Note TN-86-53, March 1986.
3. ASCII Keyboard and Video Monitor for the Microprocessor Kit, K. Padmanabhan, Electronics for you, May 1984, pp. 51-64.
4. Osborne 4 and 8 Bit Microprocessor Hand-Book, Osborne/Mcgraw Hill 1981, pp . 7.9 and 7.10.
5. VDU card, Elektor, September 1983, pp. 9.38 to 9.45
6. 6845 Based Video Monitor, K. Padmanabhan and B. Murugan, Electronics for you, April 1987, p.77.
7. HD6845 Users Manual, Hitachi.
8. A microcomputer based ionospheric scintillation data recorder, R.N. Misra and M.B. Dadhania, PRL TechnicalNote TN-86-56, oct. 1986.
9. SDK-85 System Design Kit Users Manual, Intel corporation 3065 Bowers Avenue, Santa Clara, California 95051, USA.
10. The ARRL Microcontroller, Jon Bloom QST, Vol.70, july 1986.
11. Dr. wang's Palo Alto Tiny Basic, Roger Rauskolb, Best of Interface Age, Chapter 2.

TABLE IDISPLAY PARAMETERS

<u>Sr.No</u>	<u>Item</u>	<u>Specifications</u>
1.	Scan Mode	Non Interlace
2.	Horizontal deflection frequency	$f_H$ 15.503 KHZ H = 64.5 Micro sec.
3.	Vertical deflection frequency	$f_V$ = 49.7 HZ V = 20.1 msec.
4.	Dot frequency	8 MHZ
5.	Character Cell H x V	6 x 8
6.	Displayed characters (Row x Line)	24 x 64
7.	H Sync width	6 Micro sec
8.	Cursor display	Block non-blinking

TABLE IIInitialization of 6845

<u>Register No.</u>	<u>Content(CHD)</u>	<u>End Result</u>
R0	56	86 characters in one row, H = 64 Micro sec.
R1	40	64 Displayable characters
R2	43	Location of SYNC
R3	08	H sync width 6 Micro sec.
R4	26	39 Rows in one frame
R5	00	Vertical Adj. for V Sync
R6	18	24 Displayable rows
R7	20	V sync location 32 row
R8	00	Non interlace mode
R9	07	Max. Raster Address 7+1=8 for 6x8 character cell.
R10	00	Cursor start & Mode, non-blink
R11	07	Cursor End,Block cursor.
R12	00	Memory start Address
R13	00	
R14	30	Cursor Position xx11,0000,0000,0000
R15	00	F000(CHD)
R16	xx	Light Pen Register content
R17	xx	00xx,xxxx,xxxx,xxxx

TABLE III

BUS CONNECTIONS

1	GND(Bottom most)	A	GND
2	D0	B	A0
3	D1	C	A1
4	D2	D	A2
5	D3	E	A3
6	D4	F	A4
7	D5	H	A5
8	D6	J	A6
9	D7	K	A7
10	$\overline{MR}$	L	A8
11	$\overline{MW}$	M	A9
12	$\overline{IOR}$	N	A10
13	$\overline{IOW}$	P	A11
14	$\overline{MREQ}$	R	A12
15	$\overline{RFSH}$	S	A13
16	SELECT	T	A14
17	V BATT	U	A15
18	$\overline{INTA}$	V	CLOCK OUT
19	NMI	W	HLDA
20	INTR	X	HOLD
21	RESET OUT	Y	RESET(BUTTON)
22	+5V(Topmost)	Z	+5 Volts

Connector type 44 Pin 0.156" OEN - 8607 -044

TABLE IVI/O Expansion Port25 Pin Male Connector ( D Type )

<u>Pin No.</u>	<u>Signal Type</u>	<u>Pin No</u>	<u>Signal Type</u>
1	D7	13	+5 Volts
2	D6	14	A7
3	D5	15	A6
4	D4	16	A5
5	D3	17	A4
6	D2	18	A3
7	D1	19	A2
8	D0	20	A1
9	$\overline{\text{IOW}}$	21	A0
10	$\overline{\text{IOR}}$	22	CLK OUT
11	RESET OUT	23	GND
12	GND	24	GND

TABLE VContents of 20B1 & 20B2 for desired printer port Baud Rate

<u>Sr. No.</u>	<u>Contents of Memory</u>		<u>Baud Rate</u>
	<u>20B1(1H)</u>	<u>20B2(1H)</u>	
1.	80CHD	01CHD	110
2.	84CHD	00CHD	300
3.	40CHD	00CHD	600
4.	21CHD	00CHD	1200
5.	10CHD	00CHD	2400
6.	08CHD	00CHD	4800

APPENDIX - A

I/O Port Assignments, 8255 at 08, Control Code 8A(8D)

PORT A - Key scan

PORT B - Key INPUT

PORT C

PC0	-	REMO T/R ON (OUT)
PC1	-	MIKE DATA (OUT)
PC2	-	PRINTER OUT (OUT)
PC3	-	FREE (OUT)
PC4	-	SPEAKER (INPUT)
PC5	-	FREE (DO)
PC6	-	FREE (DO)
PC7	-	FREE (DO)

APPENDIX - BPADS ON LEFT SIDE AT THE MICRO C PCB

<u>PIN No.</u>	<u>SIGNAL</u>	<u>REMARKS</u>
1. (TOPMOST)	BAUD CLK 4800 )	
2.	-do- 9600 )	TO ROTARY SWITCH
3.	-DO- 2400 )	
4. )		
5. )	CPU CLOCK SELECT	
6. )		
7.	0 4 MHZ	CPU CLOCK
8.	BAUD CLOCK 300 )	
9.	-do- 600 )	TO ROTARY SWITCH
10.	-do- 150 )	
11.	-do- 1200 )	
12.	BAUD CLOCK	8251 RxC/TxC TERMINAL (POLE OF ROTARY SWITCH)
13.	BAUD CLOCK 2400	TO ROTARY SWITCH

NOTE : ROTARY SWITCH IS OF 1 POLE 8 WAY TYPE USED FOR  
BAUD RATE SELECTION.



APPENDIX - CDETAILS OF 16 PIN SOCKETS USED FOR I/O CONNECTIONS

IC-18

<u>PIN No.</u>	<u>SIGNAL TYPE</u>	<u>REMARK</u>
1.	TxD	TO POWER SUPPLY CARD
2.	RxD	-do-
3.	NC	
4.	NC	
5.	NC	
6.	NC	
7.	NC	
8.	GND	
9.	PA4    )	
10.	PA3    )	
11.	PA5    )	
12.	PA2    )	PORT A TO KEY BOARD
13.	PA6    )	
14.	PA0    )	
15.	PA7    )	
16.	PA1    )	

APPENDIX - DIC 20PIN No.SIGNAL TYPEREMARKS

1.	PC3	
2.	PB0	
3.	PB7	
4.	PC1	
5.	PC4	
6.	PC6	
7.		
8.	PC7	
9.	PC5	
10.	PC0	
11.	PC2	
12.	PB5	
13.	PB1	
14.	PB4	
15.	PB3	
16.	PB2	

APPENDIX - ECONNECTIONS OF POWER SUPPLY CARD

<u>Pin No.</u>	<u>(WIDE EDGE)</u>	<u>SIGNAL TYPE</u>	<u>REMARKS</u>
1.	CEXTREME LEFT	DIGITAL OUTPUT	TO PC4 RECOVERED DATA
2.		TAPE IN(AUDIO)	SPK JACK FROM TAPE RECORDER
3.		GND	
4.		DATA INPUT	FROM PC1 DATA TO BE RECORDED
5.		AUDIO OUT	MIK JACK TO TAPE RECORDER
6.		NC	
7.		NC	
8.		NC	
9.		NC	
10.		DIGITAL IN	TAPE RECORDER MOTOR CONTROL FROM PC0.
11.		REMOTE CONTROL	REMO JACK OF TAPE RECORDER
12.		TTL DATA OUT	TO C SID
13.		RS232C DATA IN	FROM CONSOLE UNIT PIN 3 OF 25 PIN MALE I
14.		TTL DATA OUT	RxD OF 8251
15.		RS232C DATA IN	FROM PIN 3 OF 25 PIN MALE II

16.	TTL DATA OUT	NC
17.	RS232C DATA IN	NC
18.	+5V	REGULATOR OUTPUT
19	RS232C OUT	PIN 2 OF 25 PIN MALE I
20.	TTL DATA IN	FROM SOD/PRINTER PC2
21	RS232C OUT	PIN 2 OF 25 PIN MALE II
22	TTL DATA IN	TxD OF 8251.
23.	RS232C OUT	NC
24.	TTL DATA IN	NC
25.	NC	
26.	GND	GND
27.	NC	

## ( NARROW SIDE )

28.	+12V	
29.	+12V	TO 7805 REGULATOR
30.	-12V	
31.	9V AC )	
32.	9V AC )	FROM 9-0-9 TRANSFORMER
33.	GND	

APPENDIX - FLIST OF IMPORTANT SUB-ROUTINE AND TABLES - SERIAL  
MONITOR PROGRAMME.

<u>Address Range (Hex)</u>	<u>Description</u>
1. 0000-003E	Reset and Interrupt vectors
2. 0040-01FF	Not used
3. 0200-029E	Byte write and Byte Read used for Cassette I/O
4. 02A0-0300	Cassette T/R write & Read in Block mode
5. 030E-0312	Port Initialization for Cassette I/O
6. 0313-031A	Motor 'ON' ) Cassette T/R
7. 02F8-02FF	Motor 'OFF' )
8. 0318-0343	'Restore' Modified for Z-80
9. 03E0-03F8	'Save status'
10. 03FA-07FF	'SDK-85 Serial Monitor with facility for extra commands.'
11. 0590-0599	'C1' Character In
12. 05C4-05CE	'C0' Character Out
13. 0500-05D8	'8251' Initialize & jump

14.	078C-079F	Signon Message
15.	0C42-0C85	'Utilities required for additional commands.
16.	0C86-0C96	'Get Command' for additional Commands.
17.	0C97-0CA8	Memory Fill 'F' Command
18.	0CA9-0CC1	Search Byte 'B' Command
19.	0CC2-0CE4	Compare Memory 'C' Command
20.	0CF0-0CFF	'Print and Wait' used by Commands
21.	0D30-0D9F	Programme Move 'P' Command
22.	0DA0-0DC6	'P' Command Entry Point
23.	0DCA-0DCF	Used by Paper Punch sub-routine
24.	0DD0-0DDF	Subroutine used by 'P' Command
25.	0E00-0E5A	EPROM Programmer II
26.	0E00-0E8F	Program & Read Calls
27.	0E90-0EBB	Paper Tape Punch
28.	0EC0-0ECF	Used by Paper Tape Punch
29.	0ED0-0EED	Paper Tape Read
30.	0EEE-0EFF	Disassemble or 'L' (List) Command Entry Point
31.	0F00-0F50	2716 Programmer (I)
32.	0F51-0F5E	Subtract DE from HL result in BC
33.	0F00-0F79	2716 Copy/Read

34.	0F7A-0F7F	Used by 2716 Programmer
35.	0F80	2716 Program Entry
36.	0F8A	2708 Programm Call (deleted)
37.	0F94	2716 Read Call
38.	0F9E	Paper Tape Punch Entry Point
39.	0FAE	Paper Tape Read Entry Point
40.	0FBA	Paper Tape Read by 'R' Command
41.	0FC0	7 Extra Command
42.	0FD2	Execution Command
43.	0800-0C30	Z-80 Disassembler
44.	0C31	Dissembler Entry Point

#### LIST OF TABLES

1.	03C0 - 03DF	6845 Registers & their Content
2.	078C - 079F	Signon Message
3.	0FE1 - 0FE6	Instruction type used by P command (8085)
4.	0FE8 - 0FF8	Address Table for 7 Extra commd.
5.	0FF8 - 0FFF	ASCII Table
6.	0898 - 089F	Register Table Called by P Command
7.	0A4C - 0A58	HL, IX, IY

8.	0A5B - 0A8F	Rotation and Bit Instruction of Z-80 ASCII Table
9.	0B70 - 0B9F	Arithmetic Instructions
10.	0BA0 - 0BCB	Valid FD/DD Instructions
11.	0BD0 - 0BFF	Length of FD/DD Instructions
12.	1000 - 1393	Instruction Mnemonics 00-3F&Co-FF
13.	1395 - 1523	ED Instruction Mnemonics
14.	1525 - 155A	Valid CB Instructions
15.	1560 - 15DF	Instruction length & type
16.	15E0 - 15E6	ED Instructions of 4 Byte length



APPENDIX - G

```

* .L000 003F      INITIALIZATION AND RESTART ADDRESSES
0000 31A020      !INIT STACK POINTER
0003 C3D005      ! JUMP TO INIT ROUTINE
0006 FF          !RST 0
0007 FF          !RST 1
0008 C30820      !ADDRESSES IN PAGE 0
000B FF          !POINT TO LOCATIONS IN
000C FF          !RAM AT 2008 CH) AND UP
000D FF          !TO BE USED BY TINY BASIC
000E 3E0D        !USED BY TINY BASIC
0010 C31020      !RST 2
0013 FF          !RST 3
0014 FF          !RST 4
0015 FF          !RST 5
0016 FF          !RST 6
0017 FF          !RST 7
0018 C31820      !RST 7.5 (8085)
001B FF          !RST 8
001C FF          !RST 9
001D FF          !RST 10
001E FF          !RST 11
001F FF          !RST 12
0020 C32020      !RST 13
0023 FF          !RST 14
0024 C32420      !RST 15
0027 FF          !RST 16
002B C32820      !RST 17
002B FF          !RST 18
002C C32C20      !RST 19
002F FF          !RST 20
0030 C33020      !RST 21
0033 FF          !RST 22
0034 C33420      !RST 23
0037 FF          !RST 24
003B C33820      !RST 25
003B FF          !RST 26
003C C33C20      !RST 27
LP SP,20A0      !INIT STACK POINTER
JP 05D0         ! JUMP TO INIT ROUTINE
RST 38H        !RST 0
RST 38H        !RST 1
JP 2008        !ADDRESSES IN PAGE 0
RST 38H        !POINT TO LOCATIONS IN
RST 38H        !RAM AT 2008 CH) AND UP
RST 38H        !TO BE USED BY TINY BASIC
LD A,0D        !USED BY TINY BASIC
JP 2010        !RST 2
RST 38H        !RST 3
RST 38H        !RST 4
RST 38H        !RST 5
RST 38H        !RST 6
RST 38H        !RST 7
JP 2018        !RST 7.5 (8085)
RST 38H        !RST 8
RST 38H        !RST 9
RST 38H        !RST 10
RST 38H        !RST 11
RST 38H        !RST 12
JP 2020        !RST 13
RST 38H        !RST 14
JP 2024        !RST 15
RST 38H        !RST 16
JP 2028        !RST 17
RST 38H        !RST 18
JP 202C        !RST 19
RST 38H        !RST 20
JP 2030        !RST 21
RST 38H        !RST 22
JP 2034        !RST 23
RST 38H        !RST 24
JP 2038        !RST 25
RST 38H        !RST 26
JP 203C        !RST 27

```

CASSETTE TAPE I/O

\* .L0200 0215

```

0200 DBC2      ;100 USEC PULSE*
0202 F602      IN A,(C2)      ;PC2 IS MADE '1'
0204 D3C2      OR 02
0206 D5        OUT (C2),A
0207 1619      PUSH DE
0209 15        LD D,19
020A C20902    DEC D
020B D1        JP NZ,0209
020C DBC2      POP DE
020E DBC2      IN A,(C2)
0210 E63D      AND 3D
0212 D3C2      OUT (C2),A
0214 C9        RET

```

\* .L0215 0221

```

;100 USEC PULSE, 400 USEC DELAY,
CALL 0200      ;100 USEC PULSE
PUSH DE
LD D,6B        C:400 USEC DELAY
DEC D
JP NZ,021B
POP DE
RET

```

\* .L02200221 022F

```

;ZERO', '100 USEC PULSE 900USEC
DELAY
CALL 0200      ;100 USEC PULSE
PUSH DE
LD D,FB
DEC D
JP NZ,0227
POP DE
RET
RST 38H
RST 38H

```

\* .L0230 0250

```

ONE BIT, BIT WRITE
LD A,C
RLA
LD C,A
JP C,0239
JP C,0221
CALL 0215
CALL 0215
RET
LD B,0B
CALL 0230
DEC B
JP NZ,0242
LD DE,00A4
CALL 05F1
RET
;EQUIL TO ONE BIT TIME

```

\* .L0250 0279

```

;BYTE READ,
LD B,0B
LD C,00
LD D,00
INC D
IN A,(C2)
AND 10
JP NZ,0256
INC D
IN A,(C2)
AND 10
JP Z,025E
DEC B
JP Z,0279
LD A,D
CP 53
CALL 0288
JP NC,0254
CALL 0290
JP 0254
;REPEAT

```

\* .L0279 0288

```

;MEASURE TIME DELAY,
LD A,D
CP A6
CALL 0288
JP NC,0285
CALL 0290
LD A,C
RET
RST 38H
;SAVE AND SHIFT, COMPARE AND INSERT CARRY,
PUSH AF
LD A,C
RLA
LD C,A
POP AF
RET
RST 38H

```

\* .L0288 02BF

```

;SAVE A REG
;DATA IN C REG TO A REG
;CARRY TO LSB
;ONE MORE BIT COMPILED
;RESTORE A REG
PUSH AF
LD A,C
RLA
LD C,A
POP AF
RET
RST 38H

```

\* .L0290 029F

```

;WAIT,
IN A,(C2)
AND 10
JP NZ,0290
IN A,(C2)
AND 10
JP Z,0297
RET
;STATUS OF PC4
;WAIT AT 'ONE'
;STATUS OF PC4
;WAIT AT 'ZERO'
RET

```

```

* .L02DC 02FB
02DC D5
02DD 1620
02DF D5
02E0 CD5002
02E3 D1
02E4 FEFF
02E6 C2DD02
02E9 15
02EA C2DF02
02ED CD5002
02F0 FEFF
02F2 CAED02
02F5 D1
02F6 C9
02F7 FF
* *
.L02FB 02FF
02FB DRC2
02FA E6FE
02FC D3C2
02FE C9
* *
.L0300 030E
0300 C5
0301 CD5002
0304 C1
0305 77
0306 23
0307 0B
0308 78
0309 B1
030A C20003
030D C9
* *
.L030E 0313
030E 3E88
0310 D3C3
0312 C9
.L0313 0320
0313 DBC2
0315 F601
0317 D3C2
0319 C9
031A FF
031B 00
031C 00
031D 00
031E 00
031F 00

* .L02AD 02BB
02AD C5
02AE 4E
02AF CD4002
02B2 23
02B3 C1
02B4 0R
02B5 78
02B6 B1
02B7 C2AD02
02BA C9
* *
.L02BB 02CB
02BB C5
02BC 1E00
02BE CD4002
02C1 1EFF
02C3 CD4002
02C6 C1
02C7 C9
* *
.L02CB 02DC
02CB D5
02C9 111000
02CC D5
02CD 11FFFF
02D0 CBF105
02D3 D1
02D4 1R
02D5 7A
02D6 B3
02D7 C2CC02
02DA D1
02DB C9

```

```

* .L02A0 02AD
02A0 1680
02A2 4R
02A3 D5
02A4 CD4002
02A7 D1
02AB 15
02A9 C2A202
02AC C9
* *
.L02A0 02BB
02BB C5
02BC 1E00
02BE CD4002
02C1 1EFF
02C3 CD4002
02C6 C1
02C7 C9
* *
.L02CB 02DC
02CB D5
02C9 111000
02CC D5
02CD 11FFFF
02D0 CBF105
02D3 D1
02D4 1R
02D5 7A
02D6 B3
02D7 C2CC02
02DA D1
02DB C9

```

```

* .L02A0 02AD
02A0 1680
02A2 4R
02A3 D5
02A4 CD4002
02A7 D1
02AB 15
02A9 C2A202
02AC C9
* *
.L02A0 02BB
02BB C5
02BC 1E00
02BE CD4002
02C1 1EFF
02C3 CD4002
02C6 C1
02C7 C9
* *
.L02CB 02DC
02CB D5
02C9 111000
02CC D5
02CD 11FFFF
02D0 CBF105
02D3 D1
02D4 1R
02D5 7A
02D6 B3
02D7 C2CC02
02DA D1
02DB C9

```

```

L05031B 0344 'MODIFIED RESTOR'
NOP
#CODES IRRELEVANT TO Z-80
#DELETED, 00 INSERTED
031B 00
031C 00
031D 00
031E 00
031F 00
0320 00
0321 3AF120
0324 E608
0326 CA2D03
0329 FB
032A C33103
032D 37
032E D23103
0331 21E920
0334 F9
0335 D1
0336 C1
0337 F1
0338 2AF420
033B F9
033C 2AF220
033F E5
0340 2AEF20
0343 C9
. *
.L0390 039A 'DELAY 0.5 MILLISECONDS'
PUSH DE
LD DE,003F
CALL 05F1
POP DE
RET
RST 38H

L03903A0 0389 6845 INITIALISAYION
LP SP,20A0 #INIT STACK
LD HL,03C0 #TABLE FOR 6845 REGS
LD B,0D #NO OF REGS
LD A,(HL)
LD (F800),A #6845 REG ADDR
INC HL
LD A,(HL)
LD (F801),A #6845 REG CONTENTS
INC HL
INC HL
DEC B
JP NZ,03A8
JP 03E0

. *
.D03C0 03FF #845REGISTERS AND CONTENTS
03C0 00 56 01 3F 02 47 03 AB 04 26 05 00 06 18 07 20
03D0 08 00 09 07 0A 00 0B 07 0C 00 0D 00 FF FF FF FF

```

```

L03E0 0436 'MODIFIED INITIALISATION'
LD (20EF),HL #CODE SAME AS SDK-85
POP HL #SHIFTED HERE #WARM START
LD (20F2),HL #WARM START RST 1
PUSH AF
POP HL
LD (20ED),HL
LD HL,0000
ADD HL,SP
LD (20F4),HL
LD HL,20ED
LD SP,HL
PUSH BC
PUSH DE
NOP
LD HL,078C #PRINT SIGNON MESSAGE
LD B,14
LD C,(HL)
CALL 05C4
INC HL
DEC B
JP NZ,03FF
LD HL,20A0
LD SP,HL
LD C,2E
CALL 05FB
JP 0414
CALL 061F
CALL 05FB
LD A,C
LD BC,0006
LD HL,07AE
CP (HL)
JP Z,042D
INC HL
DEC C
JP NZ,0421
JP 0FC0
LD HL,07A0
ADD HL,BC
ADD HL,BC
LD A,(HL)
INC HL
LD H,(HL)
LD L,A
03E0 22EF20
03E3 E1
03E4 22F220
03E7 F5
03E8 E1
03E9 22ED20
03EC 210000
03EF 39
03F0 22F420
03F3 21ED20
03F6 F9
03F7 C5
03F8 D5
03F9 00
03FA 218C07
03FD 0614
03FF 4E
0400 CDC405
0403 23
0404 05
0405 C2FF03
0408 21A020
040B F9
040C 0E2E
040E CDF805
0411 C31404
0414 CD1F06
0417 CDF805
041A 79
041B 010600
041E 21AE07
0421 BE
0422 CA2D04
0425 23
0426 0D
0427 C22104
042A C3C00F
042D 21A007
0430 09
0431 09
0432 7E
0433 23
0434 66
0435 6F

```

```

040E CDF805
0411 C31404
0414 CD1F06
0417 CDF805
041A 79
041B 010600
041E 21AE07
0421 BE
0422 CA2D04
0425 23
0426 0D
0427 C22104
042A C3C00F
042D 21A007
0430 09
0431 09
0432 7E
0433 23
0434 66
0435 6F
#JUMP TO SEARCH FOR
#ADDITIONAL COMMANDS

```

```

* *
* *
* *
* *
* *
.L0590 059A 'CI' CHAR IN
0590 DB01 IN A,(01) #SERIAL PORT
0592 E602 AND 02 #STATUS
0594 CA9005 JP Z,0590 #IF NO CHAR LOOP
0597 DB00 IN A,(00) #BACK,ELSE TAKE
0599 C9 RET #ONE CHAR
* *
.L05C0 05CF*
.L05C4 05CF 'CO ' CHAR OUT
05C4 DB01 IN A,(01) #SERIAL PORT
05C6 E601 AND 01 #STATUS
05C8 CA6405 JP Z,05C4 #IF NO CHAR PENDING
05CB 79 LD A,C #OUTPUT ONE CHAR
05CC D300 OUT (00),A #FROM C REG
05CE C9 RET
* *
.L05D0 05E5 'INITIALIZATION OF 8251'
05D0 0E06 LD C,06 #6 BYTES
05D2 21E505 LD HL,05E5 #AT 05E5
05D5 CD9003 CALL 0390 #DELAY TO STABILIZE 8251
05D8 7E LD A,(HL) #GET BYTE 1
05D9 D301 OUT (01),A #OUT TO CONTROL OF 8251
05DB 23 INC HL #NEXT BYTE
05DC 0D DEC C #BYTE COUNT
05DD C2D505 JP NZ,05D5 #REPEAT TILL ALL BYTES
05E0 C3E003 JP 03E0 #SENT TO 8251, THEN TO
05E3 FF RST 38H
05E4 FF RST 38H

.D05E5 05EA
05E5 00 00 00 40 CE 05
*
.D078C 079F 'SIGNON MESSAGE:Z-80 RNH 86 '
078C 0D 0A 5A 2D
0790 38 30 20 52 4E 4D 20 41 50 52 20 38 36 20 0D 0A

```

```

* *
* *
* *
* *
* *
.L03A0 03BB '6845 INITIALIZATION '
03A0 31A020 LP SP,20A0 #INIT STACK
03A3 21C003 LD HL,03C0 #TABLE FOR 6845 REGS
03A6 060D LD B,0D #NO DE DATA
03AB 7E LD A,(HL) #
03A9 3200F8 LD (F800),A #6845 REG ADDR
03AC 23 INC HL #NEXT
03AD 7E LD A,(HL)
03AE 3201F8 LD (F801),A #6845 REG CONTENTS
03B1 23 INC HL
03B2 05 DEC B
03B3 C2AB03 JP NZ,03AB
03B6 C3E003 JP 03E0
* *
.D03C0 03DF '6845 REG NOS AND THEIR CONTENTS '
03C0 00 56 01 3F 02 47 03 AB 04 26 05 00 06 18 07 20
03D0 08 00 09 07 0A 00 0B 07 0C 00 0D 00 FF FF FF FF

```

```

LOC42 OC4F          'HALF BYTE'
OC42 CD1F06        CALL 061F      ;GET CHAR
OC45 CDF805        CALL 05FB      ;ECHO
OC48 CDE805        CALL 05BB      ;CNVBN, CONVERT TO BINARY
OC4B E80F          AND OF      ; RETAIN
OC4D C9           RET          ; FLOWER NIBBLE
OC4E FF           RST 38H
*
.LOC50 OC5F
OC50 CD420C        CALL OC42      ;CALL HALF BYTE, HIGH NIBBLE
OC53 17           RLA          ;SHIFT 4 POS LEFT
OC54 17           RLA          ;TO BRING IT TOLEFT
OC55 17           RLA
OC56 17           RLA
OC57 47           LD B, A      ;SAVE IN D REG
OC58 C5           PUSH BC      ;
OC59 CD420C        CALL OC42      ;HALF BYTE
OC5C C1           POP BC      ; BRING HIGH NIBBLE
OC5D B0           OR B        ; JOIN BOTH
OC5E C9           RET
*
.LOC60 OC77
OC60 C5           PUSH BC      ;SAVE
OC61 CDE805        CALL 05EB      ;CROUT
OC64 7A           LD A, D      ;GET D
OC65 CDC706        CALL 06C7      ;PRINT IN HEX
OC68 7B           LD A, E      ;GET E
OC69 CDE706        CALL 06C7      ;PRINT IN HEX
OC6C 0E20         LD C, 20
OC6E CDE405        CALL 05C4      ;SPACE
OC71 10           LD A, (DE)    ;GET DATA POINTED BY DE
OC72 CDC706        CALL 06C7      ; PRINT IN HEX
OC75 C1           POP BC
OC76 C9           RET
*
.LOC77 OC86
OC77 0E02         AND GENERATE MEM ADDR.
OC79 CDE806        LD C, 02      ; TWO ADDR
OC7C D1           CALL 065B      ;GETNM 2 ADDR IN 4 ASCII
OC7D E1           POP DE
OC7E E5           POP HL
OC7F EB          EX DE, HL
OC80 CD810F        CALL 0F51      ;SUBTRACT DE FROM HL
OC83 E1           POP HL
OC84 C9           RET
OC85 FF          RST 38H

LOC86 OC97          'GET COMMAND'
OC86 CD770C        CALL OC77      ;GET TWO ADDR & COUNT
OC89 C5           PUSH BC      ; SAVE COUNT
OC8A CD500C        CALL 0C50      ;FULL BYTE
OC8D 47           LD B, A      ;SAVE IB IN B REG
OC8E CD1F06        CALL 061F      ;GETCH
OC91 79           LD A, C      ;COMPARE TO CR
OC92 FE0D         CP OD
OC94 78           LD A, B      ;RESTORE BYTE
OC95 C1           POP BC      ; AND RESULT
OC96 C9           RET
*
.LOC97 OCA9        'F' CM COMMAND FILL MEMORY
OC97 CDB40C        CALL 0CB4      ;GET COMMAND
OC9A C21106        JP NZ, 0611     ;IF TERMINATOR NOT CR
OC9D 57           LD D, A      ;THEN ERR, ELSE
OC9E 72           LD (HL), D    ;LOAD DATA IN MEM
OC9F 23           INC HL       ;TILL ALL LOCATIONS
OCA0 0B           DEC BC
OCA1 78           LD A, B      ;FILLED
OCA2 B1           OR C
OCA3 C29E0C        JP NZ, 0C9E
OCA6 C30804        JP 0408
*
*
.LOCA9 OCC2        'B' COMMAND SEARCH BYTE
OCA9 CDB40C        CALL 0CB4      ;GET COMMAND
OCAC C21106        JP NZ, 0611     ; TEST TERMINATOR
OCAF 57           LD D, A      ;KEEP BYTE IN D
OCB0 BE          CP (HL)     ;COMPARE TO MEM
OCB1 CDF00C        CALL Z, 0CF0    ;IF EQUAL THEN PRINT ANDWAIT
OCB4 CA0804        JP Z, 0408    ;IF ZERO THEN WAIT NEW COMMAND
OCB7 23           INC HL
OCB8 0B           DEC BC
OCB9 78           LD A, B      ;COMPARE NEXT LOCATION
OCBA B1           OR C
OCBB CA0804        JP Z, 0408    ; ALL LOC OVER THEN QUIT
OCBE 7A           LD A, D
OCBF C3B00C        JP 0CB0      ;ELSE CONTINUE

OC97 CDB40C        CALL 0CB4      ;GET COMMAND
OC9A C21106        JP NZ, 0611     ;IF TERMINATOR NOT CR
OC9D 57           LD D, A      ;THEN ERR, ELSE
OC9E 72           LD (HL), D    ;LOAD DATA IN MEM
OC9F 23           INC HL       ;TILL ALL LOCATIONS
OCA0 0B           DEC BC
OCA1 78           LD A, B      ;FILLED
OCA2 B1           OR C
OCA3 C29E0C        JP NZ, 0C9E
OCA6 C30804        JP 0408
*
*
.LOCA9 OCC2        'B' COMMAND SEARCH BYTE
OCA9 CDB40C        CALL 0CB4      ;GET COMMAND
OCAC C21106        JP NZ, 0611     ; TEST TERMINATOR
OCAF 57           LD D, A      ;KEEP BYTE IN D
OCB0 BE          CP (HL)     ;COMPARE TO MEM
OCB1 CDF00C        CALL Z, 0CF0    ;IF EQUAL THEN PRINT ANDWAIT
OCB4 CA0804        JP Z, 0408    ;IF ZERO THEN WAIT NEW COMMAND
OCB7 23           INC HL
OCB8 0B           DEC BC
OCB9 78           LD A, B      ;COMPARE NEXT LOCATION
OCBA B1           OR C
OCBB CA0804        JP Z, 0408    ; ALL LOC OVER THEN QUIT
OCBE 7A           LD A, D
OCBF C3B00C        JP 0CB0      ;ELSE CONTINUE

```

```

LOCC2 0CE6      'C' COMMAND COMPARE TWO MEMORY ZONES
OCC2 0E03      LD C,03
OCC4 C15R06    CALL 065R
OCC7 C1        POP BC
OCC8 D1       POP DE
OCC9 E1       POP HL
OCCA C5       PUSH BC
OCCB E5       PUSH HL
OCCD EB       EX DE,HL
OCCD C0510F   CALL 0F51
OCE0 E1       POP HL
OCE1 D1       POP DE
OCE2 1A      LD A,(DE)
OCE3 BE      CP (HL)
OCE4 C4F00C   CALL NZ,0CF0
OCE7 C20804   JP NZ,0408
OCEA 23      INC HL
OCEB 13      INC DE
OCEC 0B      DEC BC
OCED 78      LD A,B
OCEE B1      OR C
OCEF C2D20C   JP NZ,0CD2
OCE3 C30804   JP 0408
OCE5 FF      RST 38H
. *
. *
.LOCF0 0100   'PRINT AND WAIT'
OCF0 E5      PUSH HL
OCF1 EB      EX DE,HL
OCF2 C5      PUSH BC
OCF3 C1A00C   CALL 0C60
OCF6 C1F0F6   CALL 061F
OCF9 79      LD A,C
OCFA FE20     CP 20
OCFC C1       POP BC
OCFD EB      EX DE,HL
OCFE E1       POP HL
OCFF C9      RET
;SAVE HL
;EXCHANGE DE & HL
;SAVE BC
;PRINT DE AND DATA POINTED BY IT
;GETCHAR
;IS IT SPACE
; RESTORE REGS

```

```

PROGRAMME MOV:FMV FOR Z-80'
LD00 0D3A
0D00 CD700D CALL 0D70 ;CALL COPY ADVANCE
0D03 FE40 CP 40 ;INSTRUCTION CODE <40(H)
0D05 DA0F0D JP C,0D0F ;WHEN PROCEED FOR LENGTH
0D08 FE00 CP C0 ;INSTRUCTION CODE <C0(H)
0D0A DA2B0D JP C,0D2B ;40-BF OF SINGLE BYTE
0D0D D6B0 SUB B0 ;FOR .CO SUB 80(H)
0D0F E5 PUSH HL
0D10 216015 LD HL,1560 ;INSTRUCTION LENGTH TABLE
0D13 85 DD A,L ;GET NTH ENTRY,N AS IN A
0D14 6F LD L,A ;INTO ACC
0D15 7E LD A,(HL)
0D16 E1 POP HL
0D17 F5 PUSH AF
0D18 E60F AND OF ;SAVE IT
0D1A C23A0D JP NZ,0D3A ;GET LO NIBBLE
0D1D F1 POP AF ;NON ZERO MEANS CB,DD,ED,FD
0D20 CA2B0D AND F0 ;FOR FD,NOW
0D23 FE20 JP Z,0D2B ;GET HI NIBBLE
0D25 CA570D CP 20 ;IF ZERO SINGLE BYTE INST
0D28 CD700D JP Z,0D57 ;IF 20 THEN 3 BYTE
0D2B D5 CALL 0D70 ;ELSE COPY ADVANCE *2 BYTE
0D2C E5 PUSH DE ;'TEST END'
0D2D 2AC020 LD HL,(20C0) ;GET END ADDR
0D30 E8 EX DE,HL
0D31 E1 POP HL
0D32 CDA006 CALL 06A0 ;CALL 'HILO'
0D35 D1 POP DE
0D36 D2000D JP NC,0D00 ;REPEAT IF SOME DATA LEFT
0D39 C9 RET ;ELSE ALL OVER
*
LD03A 0D43
0D3A FE04 CP 04 ;IF ACC HAS 04 IT IS CB
0D3C C2430D JP NZ,0D43 ;ELAS ELSE JUMP OVER
0D3F F1 POP AF
0D40 C32B0D JP 0D2B ;CB IS 2 BYTE INSTRUCTION
*
LD043 0D5D
0D43 FE03 CP 03 ;IF ACC HAS 03 IT IS ED
0D45 C25D0D JP NZ,0D5D ;ELSE JUMPOVER
0D48 F1 POP AF
0D49 CD700D CALL 0D70 ;COPY ADVANCE TO GET NEXT
0D4C C5 PUSH BC ;BYTE
0D4D E5 PUSH HL
0D4E CD6508 CALL 0865 ;TEST FOR 3 BYTE INST
0D51 E1 POP HL ;SUBROUTINE IN DISSEMBLER
0D52 05 DEC B ;BYTE COUNT IB IN B REG
0D53 C1 POP BC
0D54 CA2B0D JP Z,0D2B ;BRANCH FOR SINGLE BYTE
0D57 CD7D0D CALL 0D7D ;ELSE TEST AND ADD FOR
0D5A C32B0D JP 0D2B ;3 BYTE INST ,JUMP TESTEND
PROGRAMME MOV:FMV FOR Z-80'
LD05D 0D70
0D5D F1 POP AF
0D5E CD700D CALL 0D70 ;COPY COPY ADVANCE FOR NEXT
0D61 FE21 CP 21 ;BYTE:21,22&2A NEED TEST
0D63 CA570D JP Z,0D57 ;AND ADD OPERATION
0D66 FE22 CP 22
0D68 CA570D JP Z,0D57
0D6B FE2A CP 2A
0D6D C3D00D JP 0D00 ;REST OF IUT IT AT 0DD0(H)
*
LD070 0D7D
0D70 7E LD A,(HL) ;GET DATA POINTED BYHL HL
0D71 12 LD (DE),A ;AND LOAD IT AT ADDR POINTED
0D72 13 INC DE ;BY DE
0D73 23 INC HL
0D74 C9 RET
0D75 E1 POP HL ;GARBAGE
0D76 C1 POP BC
0D77 CD7D0D CALL 0D7D
0D7A C3370D JP 0D37 ;UFTO HERE
*
LD096 0D96 'TEST AND ADD'
0D96 7E LD A,(HL) ;GET INST BYTE
0D97 23 INC HL ;ADVANCE POINTER
0D98 E5 PUSH HL ;SAVE HL
0D99 66 LD H,(HL) ;NEXT BYTE TO H ;HL HAS 16 BIT
0D9A 6F LD L,A ;FIRST BYTE TO L:OPERAND OF IN
0D9B D5 PUSH DE ;SAVE DE
0D9C EB EX DE,HL ;DE HAS OPERAND NOW
0D9D CDF00D CALL 0DF0 ;TEST IF OPERAND IS LARGER
0D9E DA9A0D JP C,0D9A ;IF YES THEN JUMP OVER
0D9F D1 EX DE,HL ;
0D98 09 POP DE ;DE HAS OLD CONTENT
0D9C 09 ADD HL,BC ;ADD OFFSET TO OPERAND
0D9D 7D LD A,L ;GET LO BYTE
0D9E 12 LD (DE),A ;STORE AT DESTINATION ADDR
0D9F 13 INC DE ;NEXT LOCATION IS LOADED WITH
0D99 7C LD A,H ;HI BYTE
0D91 12 LD (DE),A
0D92 13 INC DE
0D93 E1 POP HL ;ADVANCE
0D94 23 INC HL ;ADVANCE
0D95 C9 RET

```



```

* .L0D96 0DA0 RESTORE AND RETURN'
0D96 F1 POP AF ;GARBAGE
0D97 C3370D JP 0D37 ;GARBAGE
0D9A E8 EX DE,HL ;
0D9B D1 POP DE ;RESTORE DE
0D9C C3800D JP 0D8D ;STORE OPERAND AS IT IS
0D9F FF RST 38H ;IT IS OUT SIDE ADDR RANGE
* .L0D9A 0DD0 'P' COMMAND:PROGRAMME MOVE,
0DA0 0E03 LD C,03 ;GET 3 ADDR,START,END&DEST
0DA2 C05806 CALL 065D ;USING GETNH
0DA5 0E03 LD C,03 ;GET 3 MORE,RUNST,RUNEND&RUNLOC
0DA7 C05806 CALL 065B ;AGAIN USING GETNH
0DA8 D1 POP DE ;NEW RUNADDR IN DE
0DAB E1 POP HL ;OLD RUNEND IN HL
0DAC 228C20 LD (208C),HL;STORE IT AT 208C(H)
0DAF E1 POP HL ;OLD RUNSTART IN HL
0DF0 228A20 LD (208A),HL;STORE IT AT 208A/BB(H)
0DE3 EB EX DE,HL ;NEW RUN ADDR IN HL OLD IN DE
0DE4 C0510F CALL 0F51 ;SUBTRACT HL-DE,RESULT IN BC
0DE7 D1 POP DE ;DEST ADDR IN DE
0DEB E1 POP HL ;END ADDR IN HL
0DB9 22C020 LD (20C0),HL;STORE IT AT 20C0/D1(H)
0DE0 E1 POP HL ;START ADDR AT IN HL
0DBD 228E20 LD (208E),HL;STORE IT AT 208E/BF(H)
0DC0 C0000D CALL 0D00 ;CALL FMOVE
0DC3 C30804 JP 040B ;RETURN TO MONITOR
0DC6 FF RST 38H
0DC7 FF RST 38H
0DC8 FF RST 38H
0DC9 FF RST 38H
0DCA CD770C CALL 0C77 ;USED BY PAPER PUNCH PROG
0DCD C3A40F JP 0FA4
* .L0DD0 0DEA 'PART OF FD/DD'
0DD0 CA570D JP Z,0D57 ;2A IS 3 BYTE INSTRUCTION
0DD3 C5 PUSH BC ;FOR OTHER IS INSTRUCTIONS
0DD4 E5 PUSH HL ;SAVE AND CALL MATCH TABLE
0DD5 C08208 CALL 0802 ;SUBROUTINE OF DISSEMBLER
0DD8 E1 POP HL
0DD9 05 DEC B ;B REG HAS INST LENGTH
0DDA CAE50D JP Z,0DE5 ;FOR SIGLE BYTE JUMPOVER
0DDD 05 DEC B ;
0DDE C1 POP BC ;2 AND 3 BYTE INSTRUCTIONS
0DDF C4700D CALL NZ,0D70;COPY ADVANCE
0DE2 C3280D JP 0D28 ;JUMP TESTEND
0DE5 C1 POP BC ;GARBAGE
0DE6 C32B0D JP 0D2B ;GARBAGE
0DE9 FF RST 38H
* .L0DFO 0DFF 'TEST'
0DFO 2ABA20 LD HL,(20BA) ;OLD RUN ADDR IN HL
0DF3 CDA006 CALL 06A0 ;CALL HILD
0DF6 DB RET C ;CARRY MEANS OLD ADDR IS GREATER
0DF7 2ABC20 LD HL,(208C) ;OLD END ADDR IN HL
0DFA CDA006 CALL 06A0 ;CALL HILD
0DFD 3F CCF ;NO CARRY MEANS IT IS LESS
0DFF C9 RET ;(HENCE OFFSET IS TO BE ADDED)

```

\* .LOE00 0E27

'EPROM PROGRAMMER,NEW CIRCUIT(VER II)

```
0E00 3E80      LD A,#80
0E02 D387      OUT (87),A
0E04 3E80      LD A,#80
0E06 D386      OUT (86),A
0E08 7E        LD A,(HL)
0E09 D384      OUT (84),A
0E0B 7B        LD A,#E
0E0C D385      OUT (85),A
0E0E 7A        LD A,#D
0E0F E61F      AND IF
0E11 F680      OR 80
0E13 D386      OUT (86),A
0E15 C0270E    CALL 0E27
0E18 23        INC HL
0E19 13        INC DE
0E1A 0B        DEC BC
0E1B 7B        LD A,#B
0E1C B1        OR C
0E1D C2080E    JP NZ,0E0B
0E20 D886      IN A,(86)
0E22 E61F      AND IF
0E24 D386      OUT (86),A
0E26 C9        RET
```

\* .LOE27 0E38

'50 MSEC PULSE' (4 MHZ CLOCK)

```
0E27 F5        PUSH AF
0E28 F620      OR 20
0E2A D386      OUT (86),A
0E2C D5        PUSH DE
0E2D 11FF23    LD DE,23FF
0E30 C0F105    CALL 05F1
0E33 D1        POP DE
0E34 F1        POP AF
0E35 D386      OUT (86),A
0E37 C9        RET
```

\* .LOE38 0E40

'OE/CE SELECT'

```
0E38 D5        PUSH DE
0E39 57        LD D,A
0E3A 3AB820    LD A,(208B)
0E3D R2        OR D
0E3E D1        POP DE
0E3F C9        RET
```

\*

.LOE40 0E58

'EPROM READ/COPY TO RAM'

```
0E40 3E90      LD A,#90
0E42 D387      OUT (87),A
0E44 7B        LD A,#E
0E45 D385      OUT (85),A
0E47 7A        LD A,#D
0E48 E61F      AND IF
0E4A 0E38      CALL OE/CE SELECT
0E4D D386      OUT (86),A
0E4F DB84      IN A,(84)
0E51 77        LD (HL),A
0E52 23        INC HL
0E53 13        INC DE
0E54 0B        DEC BC
0E55 7B        LD A,#B
0E56 B1        OR C
0E57 C2440E    JP NZ,0E44
RET             ;IF YES THEN RET,ELSE DO IT
```

\* .LOE60 0E6F

'2732 PROG CALL'

```
0E60 31A020    LP SP,20A0
0E63 3E60      LD A,#60
0E65 32B820    LD (208B),A ;STORED AT 208B(H)
0E68 C0400F    CALL 0F40
0E6B C0000E    CALL 0E00
0E6E CF        RST B
```

'2732 READ CALL'

```
0E70 31A020    LP SP,20A0
0E73 3E60      LD A,#60
0E75 32B820    LD (208B),A ;STORE AT 208B(H)
0E78 C0400F    CALL 0F40
0E7B C0400E    CALL 0E40
0E7E CF        RST B
```

\* .LOE80 0E88

'2716/2764 PROG A CALL'

```
0E80 31A020    LP SP,20A0
0E83 3E40      LD A,#40
0E85 C3650E    JP OE65 ;CODE FOR 2716/2764
;CODE FOR 2716/2764
;REST AS AT OE65(H)
;REST AS AT OE75
```

\* .LOE88 0E8F

'2716/2764 READ CALL'

```
0E88 31A020    LP SP,20A0
0E8B 3E40      LD A,#40
0E8D C3750E    JP OE75 ;CODE FOR 2716/2764
;CODE FOR 2716/2764
;REST AS AT OE75
```

```

* .LOE90 OE85 'PAPER TAPE PUNCH'
OE90 C5 ;SAVE BC
OE91 1E10 ;16 BYTES PER LINE
OE93 CDEB05 ;THEN CR-LF
OE96 OE20 ;SPACE
OE98 CDFA07 ;
OE9B 7E ;GET ONE BYTE
OE9C C1C706 ;PRINT TWO HEX AS ASCII
OE9F C1 ;RESTORE BC
OEAO CDBB0E ;COUNT BYTE
OEAB C8 ;RETURN IF ALL OVER
OEAC C5 ;ELSE SAVE BC
OEAD 23 ;NEXT LOCATION
OEAE 7B LD A ,E
OEAF FE08 CP 0B
OEA0 CA960E JP Z,OE96
OEA1 7B LD A ,E
OEA2 A7 AND A
OEA3 CA910E JP Z,OE91
OEA4 C39B0E JP OE9B
* .LOE80 OE8C 'COUNT BYTES IN BC'
OE8B 0B DEC BC
OE89 7B LD A ,B
OE8A B1 OR C
OE8B C9 RET
* .LOECC OECC '16 DEL CHARACTERS'
OECC C5 ;SAVE BC
OECC 1E10 ;16 CHAR
OECC OE7F ;DEL
OECC CDFA07 ;CALL 07FA
OECC 1D ;DEC E
OECC C2C30E JP NZ,OECC ;ALL OVER ,IF NO THEN
OECC C1 ;POP BC
OECC C9 RET
OECC FF RST 3BH
* .LOE90 OEE0 'PAPER TAPE READ'
OEE0 CDE10E CALL OEE1 ;GET HEX CHAR FROM STRING
OEE3 07 RLCA ;CONVERT TO 0-F
OEE4 07 RLCA ;TRANSFER TO HI NIBBLE
OEE5 07 RLCA
OEE6 07 RLCA
OEE7 47 LD B ,A ;SAVE IN B REG
OEE8 CDE10E CALL OEE1 ;GET MORE ,NEXT NIBBLE
OEEB B0 OR B ;ASSEMBLE TO GET A BYTE
OEEC 77 LD (HL) ,A ;STORE IN RAM
OEED 23 INC HL ;NEXT LOC
OEEE C3D00E JP OEE0 ;DO IT AGAIN
OEE1 CD1F06 CALL 061F ;READ ONE CHAR
OEE4 CD5E07 CALL 075E ;CALL VLDG,IS IT HEX
OEE7 D2E10E JP NC,OEE1 ;IF NOT TAKE NEXT CHAR
OEEA CDBB05 CALL 05BB ;CONVERT TO 0-F
OEEB C9 RET
* .LOEEE OEFF 'L' LIST OR DISSEMBLE COMMAND
OEFF OE02 LD C ,02 ;GET START AND END ADDRESSES
OEFF CDBB06 CALL 065B ;USING GETNM
OEFF D1 POP DE ;START ADDRESS
OEFF E1 POP HL
OEFF 22BE20 LD (20BE),HL ;STORED AT 20BE/BF(H)
OEFF 5B EX DE,HL
OEFF 22C020 LD (20C0),HL ;END ADDR AT 20C0/20C1(H)
OEFF C3340C JP 0C34 ;REST AT 20C34(H)

```

```

.LOF00 OF21
OF00 3E80 LD A,B0 ;ALL PORTS OUTPUT
OF02 D383 OUT (83),A ;CONTROL PORT
OF04 CD7A0F CALL OF7A ;CALL HI ADDRESS
OF07 F602 OR 02 ;OE IS HI
OF09 E672 AND 72
OF0B F5 PUSH AF ;SAVE
OF0C D382 OUT (82),A ;OUT TO PORT C
OF0E 7B LD A,E ;LD ADDRESS
OF0F D381 OUT (81),A ;TO B PORT
OF11 7E LD A,(HL) ;GET DATA BYTE
OF12 D380 OUT (80),A ;TO A PORT
OF14 F1 POP AF ;RECALL DATA FOR C PORT
OF15 CD210F CALL OF21 ;CALL PULSE
OF18 23 INC HL ;NEXT
OF19 0B DEC BC ;BYTE COUNT
OF1A 7B LD A,B ;
OF1B B1 OR C ;
OF1C C8 RET Z ;ALL BYTES OVER THEN END
OF1D 13 INC DE ;ELSE NEXT FROM LOC
OF1E C3040F JF OF04 ;PROGRAM NEXT LOC
*
.LOF21 OF32
OF21 F5 PUSH AF ;SAVE
OF22 F601 OR 01 ;CS IS HI
OF24 D382 OUT (82),A ;OUT
OF26 D5 PUSH DE ;SAVE DE
OF27 11FF23 LD DE,23FF ;DELAY COUNT 4.0 MHZ
OF2A CDF105 CALL 05F1 ;CALL DELAY
OF2D D1 POP DE ;RESTORE AND RET
OF2E F1 POP AF
OF2F D382 OUT (82),A ;CS IS LO
OF31 C9 RET
*
.LOF40 OF51
OF40 2ABE20 LD HL,(20BE) ;DATA AT 20BE/BF(H) TO
OF43 E5 PUSH HL
OF44 ER EX DE,HL ;DE,IT IS EPROM START
OF45 2AC020 LD HL,(20C0) ;DATA AT 20C0/C1(H) TO
OF4B CD510F CALL OF51 ;HL, NOW CALL 'SUBTRACT'
OF4E 2ARC20 LD HL,(208C) ;DATA AT 208C/BD(H) TO
OF4F E1 EX DE,HL ;DE,IT IS EPROM START
OF50 C9 POP HL
RET
*
.LOF51 OF5E
OF51 7B LD A,E ;SUBTRACT'
OF52 2F CPL ;2'S COMPLEMENT OF DE
OF53 5F LD E,A ;GENERATED
OF54 7A LD A,D
OF55 2F CPL
OF56 57 LD D,A
OF57 13 INC DE
OF58 19 ADD HL,DE
OF59 E5 PUSH HL ;HL-DE IN HL
OF5A C5 PUSH BC ;BC HAS HL-DE
OF5B E1 POP HL
OF5C C1 POP BC
OF5D C9 RET
OF5E FF RST 38H
*
.LOF60 OF7A
OF60 3E90 LD A,90 ;A PORT INPUT,B AND C
OF62 D383 OUT (83),A ;OUTPUT
OF64 CD7A0F CALL OF7A ;GET ADDRESSES
OF67 E670 AND 70 ;CE AND OE ARE '0'
OF69 D382 OUT (82),A
OF6B 7B LD A,E ;LD ADDR
OF6C D381 OUT (81),A ;TO PORT B
OF6E DB80 IN A,(80) ;READ ONE BYTE FROM
OF70 77 LD (HL),A ;EPROM AND STORE IT
OF71 23 INC HL ;IN RAM,NEXT LOC
OF72 13 INC DE ;DF EPROM ALSO
OF73 0B DEC BC ;BYTE COUNT
OF74 7B LD A,B
OF75 B1 OR C
OF76 C2640F JP NZ,OF64 ;READ MORE IF ALL BYTES
OF79 C9 RET ;NOT OVER ELSE QUIT
*
.LOF7A OF80
OF7A 7A LD A,D ;GET ADDRESSES'
OF7B 17 RLA ;HI ADDR
OF7C 17 RLA ;SHIFTED
OF7D 17 RLA ;LEFT 4 TIMES
OF7E 17 RLA ;TO TAKE CARE OF
OF7F C9 RET ;PADMANABHAN'S CKT
*
.LOF80 OF8A
OF80 31A020 LP SP,20A0 ;INIT STACK
OF83 CD400F CALL OF40 ;GET ADDR FROM 20BE/BF
OF86 CD000F CALL OF00 ;ETC AND CALL PROG
OF89 CF RST 8 ;TO MONITOR

```

```

*
.LOFBA OF94
'2708 PROGRAM CALL ;OLD 8085 ONLY'
LP SP,20A0 ;INIT STACK
CALL OF40 ;GET ADDR
CALL OE00 ;CALL PROG 2708
RST B ;TO MONITOR
*
.LOF94 OF9E
'2716 READ/COPY'
LP SP,20A0 ;INIT STACK
CALL OF40 ;GET ADDR
CALL OF60 ;CALL 2716 READ
RST B ;TO MONITOR
*
.LOF9E OFAE
'PAPER TAPE PUNCH'
LP SP,20A0 ;INIT STACK
CALL OF40 ;GET ADDR ;HL IS SOURCE
CALL OE00 ;16 DEL CHAR
CALL OE90 ;PUNCH DATA
CALL OE00 ;16 DEL CHAR
RST B ;TO MONITOR
*
.LOFAE OFB8
'PAPER TAPE READ'
LP SP,20A0 ;INIT STACK
CALL OF40 ;GET ADDR
CALL OE00 ;READ TAPE
RST B ;TO MONITOR
*
.LOFBA OFC0
'CALL FOR PAPER TAPE READ'
CALL OC77 ;VIA 'R' COMMAND
JP OFB4
*
.LOF00 OFD2
'JUMP HERE FOR 7 EXTRA COMMANDS'
LD RC,0007 ;7 COMMANDS
LD HL,OFFB ;CODES AT OFFB
CP (HL) ;COMPARE
JP Z,OFD2 ;IF FOUND EXECUTE IT
INC HL ;ELSE SEARCH MORE
DEC C
JP NZ,OFFC6
JP CF C31106
*
.LOFD2 OFDC
'EXECUTE COMMAND'
LD HL,OFEB ;COMMAND ADDRESS TABLE
ADD HL,BC ;ADDRESS LOADED
ADD HL,BC ;IN HL REG PAIR
LD A,(HL) ;LO IN A REG
INC HL ;HI
LD H,(HL) ;IN H REG
LD L,A ;SEE! LO IS IN L REG
JP (HL) ;GO AND EXECUTE

```

```

'SYMBOL TABLE CALLED BY *P*COMMAND'
FOR LO NIBBLES
DOFDC OFDE
OFDC 01 02 0A 06
;DOFDC OFE0
OFDC 01 02 0A 06
OFEO 0E

```

```

'SYMBOL TABLE FOR CX,DX,EX,FX INSTRUCTION'
USED BY *P*COMMAND
DOFE1 OFE7
OFE1 02 04 0A 0C 06 0E FF

```

```

'ADDRESS TABLE FOR 7 COMMANDS'
DOFEB OFF7
OFE8 00 00 A9 0C CA 0D BA 0F
OFF0 A0 0D EE 0E 97 0C C2 0C 43 46 4C 50 52 57 42 FF
;JP*
;DOFEB OFF7
OFE8 00 00 A9 0C CA 0D BA 0F
OFF0 A0 0D EE 0E 97 0C C2 0C

```

```

'ASCII TABLE FOR NEW COMMANDS'
DOFFB OFFF
OFFB 43 46 4C 50 52 57 42 FF
; C F L P R W B

```

APPENDIX - H

Z-80 RNM APR 86

\*

\*

00000 02FF

0000 31 A0 20 C3 D0 05 FF FF FF FF FF FF C3 08 20 FF FF FF FF FF C3 3E 0D  
0010 C3 10 20 FF FF FF FF FF FF C3 18 20 FF FF FF FF FF C3 2C 20 FF  
0020 C3 20 20 FF FF FF FF FF FF C3 28 20 FF C3 3C 20 FF  
0030 C3 30 20 FF FF FF FF FF FF C3 38 20 FF FF C3 3C 20 FF  
0040 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0050 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0060 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0070 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0080 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0090 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
00A0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
00B0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
00C0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
00D0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
00E0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
00F0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0100 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0110 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0120 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0130 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0140 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0150 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0160 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0170 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0180 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0190 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
01A0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
01C0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
01D0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
01E0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
01F0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0200 DB C2 F6 02 D3 C2 D5 16 19 15 C2 09 02 D1 DB C2  
0210 E6 3D 03 C2 C9 CD 00 02 D5 16 6B 15 C2 18 02 D1  
0220 C9 CD 00 02 D5 16 FB 15 C2 27 02 D1 C9 FF FF FF  
0230 79 17 4F DA 39 02 C3 21 02 CD 15 02 CD 15 02 C9  
0240 06 08 CD 30 02 05 C2 42 02 11 A4 00 CD F1 05 C9  
0250 06 08 CD 00 16 00 14 DB C2 E6 10 C2 56 02 14 DB  
0260 C2 E6 10 CA 5E 02 05 CA 79 02 7A FE 53 CD 8B 02  
0270 D2 54 02 CD 09 C2 C3 54 02 7A FE A6 CD 88 02 D2  
0280 B5 02 CD 90 02 79 C9 FF F5 79 17 4F F1 C9 FF FF  
0290 DB C2 E6 10 C2 90 02 DB C2 E6 10 CA 97 02 C9 FF  
02A0 16 80 4B D5 CD 40 02 D1 15 C2 A2 02 C9 C5 4E CD  
02B0 02 23 C1 0B 78 B1 C2 AD 02 C9 C5 1E 00 CD A0  
02C0 02 1E FF CD A0 02 C1 C9 D5 11 10 00 D5 11 FF FF  
02D0 CD F1 05 D1 1B 7A B3 C2 CC 02 D1 C9 D5 16 20 D5  
02E0 CD 50 02 D1 FE FF C2 DD 02 15 C2 DF 02 CD 50 02  
02F0 FE FF CA ED 02 D1 C9 FF DB C2 E6 FE D3 C2 C9

D0300 05FF

0300 C5 CD 50 02 C1 77 23 0R 78 B1 C2 00 03 C9 3E 8B  
0310 D3 C3 C9 DB C2 F6 01 D3 C2 C9 FF 00 00 00 00 00  
0320 00 3A F1 20 E6 08 CA 2D 03 FB C3 31 03 37 D2 31  
0330 03 21 E9 20 F9 D1 C1 F1 2A F4 20 F9 2A F2 20 E5  
0340 2A EF 20 C9 FF FF FF FF FF FF FF FF FF FF FF  
0350 FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0360 FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0370 FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0380 FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
0390 D5 11 3F 00 CD F1 05 D1 C9 FF FF FF FF FF FF  
03A0 31 A0 20 C0 03 06 7E 32 00 F8 23 7E 32 01  
03B0 F8 23 05 C2 AB 03 C3 E0 03 FF FF FF FF FF  
03C0 00 56 01 3F 02 47 03 AB 04 26 05 00 06 18 07 20  
03D0 08 00 09 07 0A 00 0B 07 0C 00 0D 00 FF FF FF  
03E0 22 EF 20 E1 22 F2 20 F5 E1 22 ED 20 21 00 00 39  
03F0 22 F4 20 21 ED 20 F9 C5 D5 00 21 8C 07 06 14 4E  
0400 CD C4 05 23 05 C2 F7 03 21 A0 20 F9 0E 2E CD F8  
0410 05 C3 14 04 CD 1F 06 CD F8 05 79 01 06 00 21 AE  
0420 07 BE CA 2D 04 23 0D C2 21 04 C3 C0 0F 21 A0 07  
0430 09 07 E9 23 66 6F E9 0E 02 CD 5B 06 D1 E1 CD EB  
0440 05 7C CD C7 06 7D CD C7 06 0E 20 CD F8 05 7E CD  
0450 C7 06 CD A0 06 D2 5E 04 CD EB 05 C3 08 04 23 7D  
0460 E6 0F C2 49 04 C3 3E 04 CD 26 06 D2 7D 04 7A FE  
0470 0D C2 11 06 21 F2 20 C3 B3 04 7A FE 0D  
0480 CD 11 06 C3 1B 03 0E 01 CD 5B 06 3E FF 32 FD 20  
0490 D1 CD 1F 06 4F CD F8 05 79 FE 1B CA C7 04 CD 79  
04A0 07 DA 91 04 CD 5E 07 D2 C1 04 CD BB 05 4F CD 3F  
04B0 07 3A FD 20 B7 C2 B9 04 13 EE FF 32 FD 20 C3 91  
04C0 04 CD 34 07 C3 11 06 CD 34 07 CD EB 05 C3 08 04  
04D0 0E 03 CD 5B 06 C1 E1 D1 E5 62 6B 7E 60 69 77 03  
04E0 78 B1 CA 08 04 13 E1 CD A0 06 D2 08 04 C3 DB 04  
04F0 CD 26 06 C5 E1 7A FE 20 CA 00 05 FE 2C C2 0B 04  
0500 7E CD C7 06 0E 2D CD F8 05 CD 26 06 D2 10 05 71  
0510 23 C3 F5 04 CD 1F 06 4F CD F8 05 79 FE 0D C2 27  
0520 05 CD EA 06 C3 08 04 4F CD 18 07 C5 E1 0E 20 CD  
0530 F8 05 79 32 FD 20 3A FD 20 CA 43 05 FE 2C  
0540 C2 08 04 7E B7 C2 4E 05 CD EB 05 C3 08 04 E5 5E  
0550 16 20 23 46 D5 D5 E1 C5 7E CD C7 06 F1 F5 B7 CA  
0560 67 05 2B 7E CD C7 06 0E 2D CD F8 05 CD 26 06 D2  
0570 B7 05 7A 32 FD 20 F1 E1 B7 CA 7E 05 70 2B 71 11  
0580 03 00 E1 19 C3 36 05 DA 32 FD 20 D1 D1 C3 7F 05  
0590 DB 01 E6 02 CA 90 05 78 00 C9 FB C9 07 C5 01 08  
05A0 00 11 02 02 CD EA 07 20 17 78 1F 47 0D C2 A1 05  
05B0 11 02 02 CD EA 07 20 17 78 1F 47 0D C2 A1 05  
05C0 F8 D6 07 C9 EB 01 E6 01 CA C4 05 79 D3 00 C9 FF  
05D0 0E 06 21 E5 05 CD 90 03 7E D3 01 23 0D C2 D5 05  
05E0 C3 E0 03 FF FF 00 00 40 CE 05 0E 0D CD F8 05  
05F0 C9 1B 7A B3 C2 F1 05 C9 41 3E 1B B8 C2 01 06 0E

D0600 08FF

0600 24 CD C4 05 3E 0D RB C2 0F 06 0E 0A CD C4 05 48  
0610 C9 0E 2A CD FB 05 CD EB 05 C3 08 0A 04 37 3F C9 CD  
0620 90 05 E6 7F 4F C9 E5 21 00 00 1E 00 CD 1F 06 4F  
0630 CD FB 05 CD 79 07 D2 45 06 51 E5 C1 E1 7B B7 C2  
0640 32 07 CA 1C 06 CD 5E 07 D2 11 06 CD BB 05 1E FF  
0650 29 29 29 06 00 4F 09 C3 2C 06 4E 03 79 E6 03  
0660 C8 67 CD 26 06 D2 11 06 C5 2D 25 CA 77 06 7A FE  
0670 0D CA 11 06 C3 62 06 7A FE 0D C2 11 06 01 7F FF  
0680 7D B7 CA 8A 06 C5 2D C2 85 06 C1 D1 E1 CD A0 06  
0690 D2 95 06 54 5D E3 D5 C5 E5 3D FB E1 E3 C3 99 06  
06A0 C5 47 E5 7A B3 CA C1 06 23 7C B5 CA C1 06 E1 D5  
06B0 3E FF AA 57 3E FF AB 5F 13 7D B3 7C 8A D1 7B C1  
06C0 C9 E1 78 C1 C3 32 07 E5 0F 4F CD E2 06 CD FB 05  
06D0 CD E2 06 CD FB 05 F1 E6 0F 4F CD E2 06 CD FB 05  
06E0 E1 C9 21 B4 07 06 00 09 4E C9 21 C4 07 4E 79 B7  
06F0 C2 F7 06 CD EB 05 C9 CD FB 05 0E 3D CD FB 05 23  
0700 5E 16 20 23 1A CD C7 06 7E B7 CA 12 07 1B 1A CD  
0710 C7 06 0E 20 CD FB 05 23 C3 ED 06 21 C4 07 11 03  
0720 00 7E B7 CA 11 06 B9 CA 2E 07 19 C3 21 07 23 44  
0730 4D C9 37 C9 3A FD 20 B7 C0 0E 00 CD 3F 07 C9 D5  
0740 E1 79 E6 0F 4F 3A FD 20 B7 C2 52 07 7E E6 F0 B1  
0750 77 C9 7E E6 0F 4F 79 0F 0F 0F 80 77 C9 79 FE  
0760 30 FA 1C 06 FE 39 FA 32 07 CA 32 07 FE 41 FA 1C  
0770 06 FE 47 F2 1C 06 C3 32 07 79 FE 2C CA 32 07 FE  
0780 0D CA 32 07 FE 20 CA 32 07 C3 1C 06 0D 0A 5A 2D  
0790 38 30 20 52 4E 4D 20 41 50 52 20 38 36 20 0D 0A  
07A0 00 00 14 05 F0 04 D0 04 86 04 68 04 37 04 44 47  
07B0 49 4D 53 58 30 31 32 33 34 35 36 37 38 39 41 42  
07C0 43 44 45 46 41 EE 00 42 EC 00 43 ER 00 44 EA 00  
07D0 45 E9 00 46 ED 00 49 F1 00 48 F0 00 4C EF 00 4D  
07E0 F0 01 53 F5 01 50 F3 01 00 00 E5 2A B1 20 29 1D  
07F0 C2 EE 07 EB E1 CD F1 05 C9 FF C3 CA 05 C3 90 05  
0800 7E F5 CD C7 06 F1 E5 FE 40 DA 1A 08 FE C0 D2 18  
0810 0E 07 CD A0 08 E1 C9 D6 80 21 60 15 00 85 6F  
0820 7E F5 E6 04 1F 1F 1F 1F 1F 1F 1F 1F 1F 1F 1F 1F  
0830 7B A7 CA 42 08 E1 E5 23 7E C5 CD C7 06 C1 05 C2  
0840 37 08 3E 09 B7 3C 47 CD A0 08 E1 C9 06 00 C9  
0850 32 B3 20 FE 04 C2 5B 08 06 01 C9 2A BE 20 FE 03  
0860 C2 79 08 23 7E 21 E0 15 06 06 2B 04 23 05 CA 77  
0870 08 BE C2 6C 08 06 02 04 C9 23 23 7E 32 B4 20 2B  
0880 7E 00 21 A0 08 06 29 CD 5C 09 D2 4D 08 79 21 D0  
0890 08 85 6F 46 04 C9 FF CD 42 43 44 45 48 4C 24 41  
08A0 0E 20 C5 CD FB 05 C1 05 C2 A0 08 C9 FF F1 E1 FE  
08B0 E1 D2 01 09 FE 39 CA 01 09 FE 34 DA 01 09 F5 23  
08C0 7E 32 B4 20 22 BE 20 3E FF 32 R5 20 F1 C3 01 09  
08D0 2B 23 7E FE 0D C2 D1 08 0D C2 D1 08 C9 E5 F5 21  
08E0 8B 0A CD 90 0A F1 E1 C9 E5 F5 21 5A 0A CD 90 0A  
08F0 F1 E1 C9 F1 E1 CD DD 08 2A BE 20 23 22 BE 20 C9

D0900 08FF

0900 7E FE 40 DA 26 09 FE DD CA 80 09 FE ED CA 3A 09 FE  
0910 FE CR CA 32 09 FE DD CA 80 21 FF 07 4F A7 C4 D0 08 CD A0  
0920 FD CA 80 09 D6 B0 21 C3 B7 09 23 7E 22 BE 20 21  
0930 0A C9 3E 00 32 B3 20 C3 09 D2 DD 08 79 21 94 13 A7 4F  
0940 25 15 06 36 CD 5C 09 0A 22 BE 20 C9 0E 00 2B 0D  
0950 00 C4 D0 08 CD A0 0A 22 BE 20 C9 0E 00 2B 0D  
0960 04 23 0C 05 CA 6D 09 BE C2 61 09 37 C9 37 3F C9  
0970 21 98 08 85 6F 7E FE 24 CC 23 0A 4F CD FB 05 C9  
0980 23 7E E5 21 A0 0B 06 29 CD 5C 09 D2 F4 08 E1 22  
0990 0E 20 7E FE CB C2 AF 08 E5 F5 23 00 7E FE 32 B4  
09A0 CA F3 08 E6 07 FE 06 C2 F3 08 F1 E1 23 7E 32 B4  
09B0 20 3E FF 32 R5 20 00 23 7E FE 40 DA F6 09 E5 F5  
09C0 E6 C0 1F 1F 1F 1F 1F 1F 1F 1F 1F 1F 1F 1F 1F 1F  
09D0 CD 90 0A 0E 20 CD FB 05 F1 E6 07 CD 70 09  
09E0 30 4F CD FB 05 2C CD FB 05 F1 E6 07 CD 70 09  
09F0 E1 23 22 BE 20 C9 E5 FE 38 D2 01 0A FE 30 D2 F4  
0A00 08 F5 E6 38 1F 1F 21 5F 0A 4F A7 C4 D1 08 CD  
0A10 90 0A 0E 20 CD FB 05 F1 E6 07 CD 70 09 E1 23 22  
0A20 BE 20 C9 0E 28 CD FB 05 3A B3 20 21 50 0A A7 4F  
0A30 C4 D0 08 CD 90 0A 3A B3 20 87 CA 48 0A 0E 2B CD  
0A40 F8 05 3A B4 20 CD C7 06 3E 29 C9 FF 44 20 0D  
0A50 48 48 4C 0D 49 58 0D 49 59 0D 48 41 4C 54 0D  
0A60 52 4C 43 0D 52 52 43 0D 52 43 0D 52 52 20 0D  
0A70 53 4C 41 0D 53 52 41 0D 53 52 41 0D 53 52 4C 0D  
0A80 42 49 54 0D 53 45 53 0D 53 45 54 0D 53 52 4C 0D  
0A90 23 7E FE 0D CB 4F CD FB 05 C3 90 0A FF FF FF  
0AA0 23 7E FE 0D CB 4F CD FB 05 C3 90 0A FF FF FF  
0AB0 24 C2 DE 0A 3A B3 20 00 E5 21 50 0A A7 4F CA D0  
0AC0 08 CD 90 0A E1 00 00 00 00 00 00 00 00 00 00 00  
0AD0 0E 2B CD FB 05 3A B4 20 CD C7 06 C3 A0 0A FE 3F  
0AE0 C2 0C 08 23 7E E5 2A BE 20 23 FE C2 01 08 23  
0AF0 7E CD C7 06 2B 7E CD C7 06 23 22 BE 20 E1 C3 A0  
0B00 0A 7E CD C7 06 22 BE 20 E1 C3 A1 0A 4F CD FB 05  
0B10 C3 A0 0A F5 FE 76 CA 41 0B 21 4B 0A CD 90 0A F1  
0B20 F5 E6 38 1F 1F 1F CD 70 09 0E 20 CD FB 05 0E 2C  
0B30 CD FB 05 F1 E6 07 CD 70 09 2A BE 20 23 22 BE 20  
0B40 C9 CD EB 08 F1 C3 FB 08 F5 E6 38 1F 1F 4F 21  
0B50 70 0B A7 C4 D0 08 05 C3 33 0B FF FF FF FF FF  
0B60 0E 2C CD FB 05 C3 33 0B FF FF FF FF FF FF FF  
0B70 41 44 44 20 41 2C 0D 41 44 43 20 41 2C 0D 53 55  
0B80 42 20 0D 53 42 43 20 41 2C 0D 41 4E 44 20 0D 58  
0B90 4F 52 20 0D 4F 52 20 0D 43 50 20 0D FF FF FF  
0BA0 09 19 21 22 23 29 2A 2B 34 35 36 39 46 4E 56 5E  
0BB0 66 6E 70 71 72 73 74 75 76 77 7E 86 8E 96 9E A6  
0BC0 AE B6 BE CR E1 E3 E5 E9 F9 FF FF FF FF FF FF  
0BD0 00 00 02 02 00 00 02 00 01 01 02 00 01 01 01  
0BE0 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01  
0BF0 01 01 01 01 02 00 00 00 00 00 00 00 00 00 2A BE 20

DOE00 OE00

OC00 3E 00 32 B3 20 32 B5 20 CD EB 05 7C CD C7 06 7D  
 OC10 CD C7 06 OE 20 CD F8 05 CD 00 CD 0E 20 CD F8 0B  
 OC20 CD 00 09 2A C0 20 EB 2A BE 20 CD A0 06 D2 FD 0B  
 OC30 C9 2A BE 20 E5 CD FD 0B E1 22 BE 20 CD EB 05 C3  
 OC40 08 04 CD 1F 06 CD F8 05 CD BE 05 E6 OF C9 FF  
 OC50 CD 42 0C 17 17 17 47 C5 CD 42 C1 80 C9 FF  
 OE60 C5 CD EB 05 7A CD C7 06 7B CD C7 06 20 CD C4  
 OE70 05 1A CD C7 06 C1 C9 OE 02 CD 5B 06 D1 E1 E5 EB  
 OE80 CD 51 0F E1 C9 FF CD 77 0C C5 CD 50 0C 47 CD 1F  
 OE90 04 79 FE OD 7B C1 C9 CD 86 0C C2 11 06 57 72 23  
 OEAO 08 7B B1 C2 9E 0C C3 08 04 CD 86 0C C2 11 06 57  
 OEBO BE CC FO OC CA 08 04 23 0B 7B B1 CA 08 04 7A C3  
 OECO 80 0E 03 CD 5B 06 C1 D1 E1 C5 E5 EB CD 51 0F  
 OED0 E1 D1 1A BE C4 FO OC C2 0B 04 23 13 0B 7B B1 C2  
 OEE0 D2 0C C3 08 04 FF FF FF FF FF FF FF FF FF FF  
 OEF0 E5 ER C5 CD 60 0C CD 1F 06 79 FE 20 C1 EB E1 C9  
 OEG0 CD 70 0D FE 40 DA OF 0D FE C0 DA 2B 0D D6 80 E5  
 OEH0 21 60 15 85 6F 7E E1 F5 E6 OF C2 3A OD F1 E6 F0  
 OEI0 CA 2B OD FE 20 CA 57 0D CD 70 0D D5 E5 2A C0 20  
 O EJ0 EB E1 CD A0 06 D1 D2 00 0D C9 FE 04 C2 43 OD F1  
 OE40 C3 2B OD FE 03 C2 5D OD F1 CD 70 0D C5 E5 CD 65  
 OE50 08 E1 05 C1 CA 2B OD CD 7D 0D C3 2B OD F1 CD 70  
 OE60 OD FE 21 CA 57 OD FE 22 CA 57 OD C3 2B OD  
 OE70 7E 12 13 23 C9 E1 C1 CD 7D 0D C3 37 OD 7E 23 13  
 OE80 66 6F D5 ER CD FO OD DA 9A OD EB D1 09 7D 12 13  
 OE90 7C 12 13 E1 23 C9 F1 C3 37 OD EB D1 C3 8D OD FF  
 OEAO 0E 03 CD 5B 06 0E 03 CD 5B 06 D1 E1 22 BC 20 E1  
 OEBO 22 BA 20 ER CD 51 0F D1 E1 22 C0 20 E1 22 BE 20  
 OECO CD 00 OD C3 08 04 FF FF FF CD 77 0C C3 A4 0F  
 OEEO CA 57 OD C5 E5 CD 82 08 E1 05 CA E5 OD 05 C1 C4  
 OEF0 70 OD C3 2B OD C1 C3 2B OD FF FF FF FF FF FF  
 OEG0 2A BA 20 CD A0 06 DB 2A BC 20 CD A0 06 3F C9 FF  
 OEEO 3E

DOE00 OFFF

OE00 3E 80 D3 B7 3E 80 D3 B8 00 D3 B8 00 D3 B8 00  
 OE10 1F F6 80 D3 B6 CD 27 0E 23 13 0B 78 B1 C2 0B OE  
 OE20 DB B6 E6 1F D3 B6 C9 F5 F6 20 D3 B6 D5 11 FF 23  
 OE30 CD F1 05 D1 F1 D3 B6 C9 D5 57 3A BB 20 B2 D1 C3  
 OE40 3E 90 D3 B7 7B D3 B5 7A E6 1F CD 38 OE D3 B6 DB  
 OE50 8A 77 23 13 0B 78 B1 C2 44 OE C9 FF FF FF  
 OE60 31 A0 20 3E 60 32 BB 20 CD 40 OF CD 00 OE CF FF  
 OE70 31 A0 20 3E 60 32 BB 20 CD 40 OF CD 00 OE CF FF  
 OE80 31 A0 20 3E 60 32 BB 20 CD 40 OF CD 00 OE CF FF  
 OE90 C5 1E 10 CD EB 05 OE 20 CD FA 07 7E CD C7 06 C1  
 OEAO CD BB OE C8 C5 23 1D 7B FE 0B CA 96 OE 7B A7 CA  
 OEBO 91 OE C3 9B OE FF FF FF 0B 78 B1 C9 FF FF FF FF  
 OECO C5 1E 10 OE 7F CD FA 07 1D C2 C3 OE C1 C9 FF FF  
 OED0 CD E1 OE 07 07 07 07 07 07 07 07 07 07 07 07 07  
 OEE0 OE CD 1F 06 CD 5E 07 D2 E1 OE CD BB 05 C9 OE 02  
 OEF0 CD SB 06 D1 E1 22 BE 20 ER 22 C0 20 C3 34 OC FF  
 OF00 3E 80 D3 B3 CD 7A OF F6 02 E6 72 F5 D3 82 7B D3  
 OF10 81 7E D3 80 F1 CD 21 OF 23 0B 78 B1 C8 13 C3 04  
 OF20 OF F5 F6 01 D3 82 D5 11 FF 23 CD F1 05 D1 F1 D3  
 OF30 82 C9 FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
 OF40 2A BE 20 E5 EB 2A C0 20 CD 51 OF 2A BC 20 EB E1  
 OF50 C9 7B 2F 5F 7A 2F 57 13 19 E5 C5 E1 C1 C9 FF FF  
 OF60 3E 90 D3 B3 CD 7A OF E6 70 D3 82 7B D3 81 DB 80  
 OF70 77 23 13 0B 78 B1 C2 64 OF C9 7A 17 17 17 C9  
 OF80 31 A0 20 OF CD 00 OF CD 31 A0 20 CD 40 OF C9  
 OF90 CD 00 OE CF 31 A0 20 CD 40 OF CD 60 OF CF 31 A0  
 OFAO 20 CD 40 OF CD C0 OE CD C0 OE CD C0 OE CD C0 OE  
 OFBO 20 CD 40 OF CD D0 OE CF FF FF CD 77 0C C3 B4 0F  
 OFCO 01 07 00 21 F8 OF BE CA D2 OF 23 0D C2 C6 OF C3  
 OFDO 11 06 21 EB OF 09 06 23 66 6F E9 01 0D BA 06  
 OFEO OE 02 04 OA OC 06 0E 9E 00 00 A9 OC CA OD BA 0F  
 OFF0 A0 OD EE OE 97 0C C2 OC 43 46 4C 50 52 57 42



D1300 15FF

1300 50 20 50 45 20 3F 0D 45 5B 20 44 45 2C 24 0D  
1310 43 41 4C 4C 20 50 45 2C 3F 0D 40 0D 58 4F 52  
1320 20 3F 0D 52 53 54 20 32 38 48 0D 52 45 3F 20 50  
1330 0D 50 4F 50 20 41 46 0D 4A 50 20 50 2C 3F 0D  
1340 44 49 0D 43 41 4C 40 50 2C 3F 0D 50 55 53  
1350 48 20 41 46 0D 4F 52 20 3F 0D 52 53 54 20 33 30  
1360 48 0D 52 45 5A 20 4D 0D 4C 44 20 53 50 2C 24 0D  
1370 41 0D 4A 4E 5A 20 3F 0D 45 4D 43 41 4C 4C 20  
1380 4D 2C 3F 0D 40 0D 43 50 20 3F 0D 52 53 54 20  
1390 33 38 48 0D 4F 49 4E 20 42 2C 28 43 29 0D 4F 55  
13A0 44 20 28 43 29 2C 42 0D 53 42 43 20 48 4C 2C 42  
13B0 43 0D 4C 44 20 28 3F 29 2C 42 43 0D 4E 45 47  
13C0 0D 52 45 54 4E 0D 49 4D 20 4F 0D 4C 44 20 49 2C  
13D0 41 0D 49 4E 20 43 2C 28 43 29 0D 4F 55 54 20 28  
13E0 43 29 2C 43 0D 41 44 43 20 48 4C 2C 42 43 0D 4C  
13F0 44 20 42 43 2C 28 3F 29 0D 52 45 54 49 0D 49  
1400 4E 20 44 2C 28 43 29 0D 4F 55 54 20 28 43 29 2C  
1410 44 0D 53 42 43 20 48 4C 2C 44 45 0D 4C 44 20 28  
1420 3F 3F 29 2C 44 45 0D 49 4D 20 31 0D 4C 44 20 41  
1430 2C 49 0D 49 4E 20 45 2C 28 43 29 0D 4F 55 54 20  
1440 28 43 29 2C 45 0D 41 44 43 20 48 4C 2C 44 45 0D  
1450 4C 44 20 44 45 2C 28 3F 3F 29 0D 49 4D 20 32 0D  
1460 49 4E 20 48 2C 28 43 29 0D 4F 55 54 20 28 43 29  
1470 2C 48 0D 53 42 43 20 48 4C 2C 48 4C 0D 52 52 44  
1480 0D 49 4E 20 4C 2C 28 43 29 0D 4F 55 54 20 28 43  
1490 29 2C 4C 0D 41 44 43 20 48 4C 2C 48 4C 0D 52 4C  
14A0 44 0D 53 42 43 20 48 4C 2C 53 50 0D 4C 44 20 28  
14B0 3F 3F 29 2C 53 50 0D 49 4E 20 41 2C 28 43 29 0D  
14C0 4F 55 54 20 28 43 29 2C 41 0D 41 44 43 20 48 4C  
14D0 2C 53 50 0D 4C 44 20 53 50 2C 28 3F 3F 29 0D 4C  
14E0 44 49 0D 43 50 49 0D 49 4E 49 0D 4F 55 54 49 0D  
14F0 4C 44 44 0D 43 50 44 0D 49 4E 44 0D 4F 55 54 49 0D  
1500 0D 4C 44 49 52 0D 43 50 49 52 0D 49 4E 49 52 0D  
1510 4F 54 49 52 0D 43 50 44 52 0D 49 4E 44 52 0D 4F  
1520 54 44 52 0D 51 52 53 56 57 58 59 5A 5B 5E 61 62  
1530 48 4D 50 51 52 53 56 57 58 59 5A 5B 5E 61 62  
1540 67 68 69 6A 6F 72 73 78 79 7A 7B 7C 7D 7E 7F 7G 7H  
1550 A9 AA AB B0 B1 B2 B3 B8 B9 BA BB FF FF FF FF FF FF  
1560 00 20 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
1570 10 20 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
1580 10 20 20 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
1590 10 20 20 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
15A0 00 00 20 20 00 00 00 00 00 00 00 00 00 00 00 00 00  
15B0 00 00 20 10 20 00 00 00 00 00 00 00 00 00 00 00 00  
15C0 00 00 20 00 20 00 00 00 00 00 00 00 00 00 00 00 00  
15D0 00 00 20 00 20 00 00 00 00 00 00 00 00 00 00 00 00  
15E0 43 4B 53 5B 73 7B FF FF FF FF FF FF FF FF FF FF FF FF  
15F0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

D1000 12FF

1000 4E 4F 50 0D 4C 44 20 42 43 2C 3F 0D 4C 44 20  
1010 2B 42 43 29 2C 41 0D 49 4E 43 20 42 43 0D 49 4E  
1020 43 20 4D 44 45 43 20 42 0D 4C 44 20 42 2C 3F  
1030 0D 52 4C 43 41 0D 45 58 20 41 46 2C 41 46 27 0D  
1040 41 44 44 20 24 2C 42 43 0D 44 41 20 2C 28 42  
1050 43 29 0D 44 45 43 20 42 43 0D 49 4E 43 20 43 0D  
1060 44 45 43 20 43 0D 4C 44 20 43 2C 3F 0D 52 52 43  
1070 41 0D 4A 4E 5A 20 3F 0D 4C 44 20 44 45 2C 3F  
1080 3F 0D 4C 44 20 28 44 45 29 2C 41 0D 49 4E 43 20  
1090 44 45 0D 49 4E 43 20 44 0D 44 45 43 20 44 0D 4C  
10A0 44 20 24 2C 3F 0D 52 4C 41 0D 4A 52 20 3F 0D 41  
10B0 44 44 20 24 20 2C 44 45 0D 4C 44 20 41 2C 28 44  
10C0 45 29 0D 44 45 43 20 44 45 0D 49 4E 43 20 43 0D  
10D0 44 45 43 20 45 0D 4C 44 20 45 2C 3F 0D 52 52 41  
10E0 0D 4A 52 20 4E 5A 2C 3F 0D 4C 44 20 24 2C 3F 3F  
10F0 0D 4C 44 20 28 3F 3F 29 2C 24 0D 49 4E 43 20 24  
1100 0D 49 4E 43 20 48 0D 44 45 43 20 48 0D 4C 44 20  
1110 48 2C 3F 0D 44 41 41 0D 4A 52 20 5A 2C 3F 0D 41  
1120 44 44 20 24 2C 24 0D 4C 44 20 24 2C 28 3F 3F 29  
1130 0D 44 45 43 20 24 20 0D 49 4E 43 20 4C 0D 44 45  
1140 43 20 4C 0D 4C 44 20 4C 2C 3F 0D 43 50 4C 0D 4A  
1150 52 20 4E 43 2C 3F 0D 4C 50 20 53 50 2C 3F 0D 4A  
1160 4C 44 20 28 3F 3F 29 2C 41 0D 49 4E 43 20 53 50  
1170 0D 49 4E 43 20 28 24 29 0D 44 45 43 20 28 24 29  
1180 0D 4C 44 20 28 24 29 2C 3F 0D 53 43 46 0D 4A 52  
1190 20 43 2C 3F 0D 41 44 44 20 24 2C 53 50 0D 4C 44  
11A0 20 41 2C 28 3F 3F 29 0D 44 45 43 20 53 50 0D 49  
11B0 4E 43 20 41 0D 44 45 43 20 41 0D 4C 44 20 41 2C  
11C0 3F 0D 43 43 46 0D 52 45 54 20 4E 5A 0D 50 4F 50  
11D0 20 42 43 0D 4A 50 20 4E 5A 2C 3F 3F 0D 4A 50 20  
11E0 3F 0D 43 41 4C 4C 20 4E 5A 2C 3F 0D 50 55  
11F0 53 48 20 42 43 0D 41 44 44 20 41 2C 3F 0D 52 53  
1200 54 20 4F 0D 52 45 54 20 5A 0D 52 45 54 0D 4A 50  
1210 20 5A 2C 3F 0D 40 0D 43 41 4C 4C 20 5A 2C 3F  
1220 3F 0D 43 41 4C 4C 20 3F 3F 0D 41 44 43 20 41 2C  
1230 3F 0D 52 53 54 20 38 0D 52 43 54 20 4E 43 0D 50  
1240 4F 50 20 44 45 0D 4A 50 20 4E 43 2C 3F 0D 4F  
1250 55 54 20 28 3F 29 2C 41 0D 43 41 4C 4C 20 4E 43  
1260 2C 3F 3F 0D 50 53 48 20 44 45 0D 53 55 42 20  
1270 3F 0D 52 53 54 20 31 30 48 0D 52 45 54 20 43 0D  
1280 45 58 58 0D 4A 50 20 43 2C 3F 3F 0D 49 4E 20 41  
1290 2C 28 3F 29 0D 43 41 4C 4C 20 43 2C 3F 3F 0D 40  
12A0 0D 53 42 43 20 41 2C 3F 0D 52 53 54 20 31 38 48  
12B0 0D 52 45 54 20 50 4F 50 20 24 0D 4A 50  
12C0 20 50 4F 2C 3F 3F 0D 45 58 20 28 53 50 29 2C 24  
12D0 0D 43 41 4C 4C 20 50 4F 2C 3F 0D 55 53 48  
12E0 20 24 0D 41 4E 44 20 3F 0D 52 53 54 20 32 30 48  
12F0 0D 52 45 54 20 50 45 0D 4A 50 20 28 24 29 0D 4A

D1600 18FF

1600 C3 A0 8A C2 1F 81 13 CD BC 86 29 DA 66 81 D5 EB  
1610 CD FE 86 E7 DA 0E 88 21 00 8F CD 9C 87 D1 C9 FE  
1620 18 3F D8 13 21 00 8F 07 85 6F 3E 00 8C 67 C9 23  
1630 CA 3A 81 C5 4E 06 00 09 C1 18 13 23 E3 C9 21 00  
1640 00 4A EF FE 30 D8 FE 3A D0 3E F0 A4 C2 66 81 04  
1650 C5 4A 4D 29 29 09 29 1A 13 E6 0F 85 6F 3E 00 8C  
1660 67 C1 1A F2 43 81 D5 11 6D 81 C3 E4 87 48 4F 57  
1670 3F 0D 4F 4B 0D 57 48 41 54 3F 0D 53 4F 52 59  
1680 0D 31 00 A0 CD 0E 00 11 72 81 97 CD 7F 88 21 95  
1690 81 22 C1 89 21 00 00 22 C9 89 22 C3 89 3E 3E CD  
16A0 14 88 D5 11 37 0F CD 3E 81 EF 7C B5 C1 CA D5 82  
16B0 1B 7C 12 1B 7D 12 C5 D5 79 93 F5 CD 57 88 D5 C2  
16C0 D2 81 D5 CD 73 88 C1 2A D5 89 CD 00 89 60 69 22  
16D0 D5 89 C1 2A D5 89 F1 E5 FE 03 CA 81 81 B5 6F 3E  
16E0 00 8C 67 11 00 8F E7 D2 0D 88 22 D5 89 D1 CD 09  
16F0 89 D1 E1 CD 00 89 C3 9D 81 4C 49 53 54 84 20 52  
1700 55 4E 83 15 4E 45 57 83 06 4C 4F 41 44 83 43 53  
1710 41 56 45 83 8A 42 59 45 83 ED 4E 45 58 54 85 08  
1720 4C 45 54 85 C8 4F 55 54 87 0B 50 4F 4B 45 87 47  
1730 57 41 49 54 87 1F 49 46 85 59 47 4F 54 4F 83 34  
1740 47 4F 53 55 42 84 70 52 45 54 55 52 4E 84 90 52  
1750 45 4D 85 55 46 4F 52 84 A9 49 4E 50 55 54 85 72  
1760 50 52 49 4E 54 84 38 53 54 4F 50 83 0F 85 C2 59  
1770 4E 55 20 43 DD A6 B5 AB 50 57 AD 27 D6 F0 EE 52  
1780 4A 86 67 49 4E 50 87 38 50 45 45 4B 87 57 55  
1790 53 52 87 61 41 42 53 86 F2 53 49 5A 45 86 FE 86  
17A0 AD 59 4F 55 20 43 FF 89 CC 33 CB 8F 64 C5 1D BF  
17B0 FF 54 4F 84 B9 87 E0 53 54 45 50 84 C3 84 C7 3E  
17C0 3D 85 D8 23 85 DE 3E 85 E4 3D 85 F3 3C 3D 85 EB  
17D0 3C 85 F9 85 FF 21 FB 81 EF D5 1A 13 FE 2E CA F7  
17E0 ED 82 23 D1 C3 D8 82 3E 7F 23 BE D2 F9 82 7E 23  
17F0 6E E6 FF 67 F1 E9 CD DC 87 21 D7 89 22 D5 89 CD  
1800 DC 87 C3 81 81 CD DC 87 11 D7 89 21 00 00 CD 5F  
1810 88 DA 81 81 EB 22 C1 89 EB 13 13 CD 85 89 21 19  
1820 82 C3 D8 82 DF D5 CD DC 87 C5 21 D5 89 CD 67 83 11 A0 20  
1830 C3 24 83 EF E5 D5 C5 06 D1 D5 EF 05 CA 8D 83 BE  
1840 02 7F 88 C3 BD 83 06 06 D1 D5 EF 05 CA 8D 83 BE  
1850 CD 5A 83 C3 47 83 E5 CD 0E 03 CD 13 03 CD CB  
1860 02 CD DC 02 21 A0 20 01 10 00 CD 05 03 2A A7 20  
1870 05 C1 E1 CD 00 03 CD F8 02 C9 EF E5 D5 C5 21 A0  
1880 20 06 06 E1 EF 77 23 13 05 C2 93 83 36 0D 23 E5 2A  
1890 D5 89 11 D5 89 CD 51 0F E1 71 23 70 C5 06 07 23  
18A0 36 00 05 C2 AF 83 21 D5 89 C1 CD C1 83 C1 D1 E1  
18B0 F7 C5 E5 21 A0 20 01 10 00 CD 0E 03 CD 13 03 CD  
18C0 CD 02 CD BB 02 CD AD 02 E1 C1 CD AD 02 CD BB 02  
18D0 CD F8 02 C9 31 A0 20 CD 0E 03 CD 13 03 CD FF FF  
18E0 31 A0 20 CD 0E 03 CD FB 02 C3 E0 03 FF FF FF  
18F0 31 A0 20 CD 0E 03 CD 03 E0 03 FF FF FF

D1900 1BFF

1900 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
1910 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
1920 CD 3E 81 CD DC 87 CD 57 88 DA 81 81 CD ED 88 CD  
1930 B5 89 CD 5F 88 C3 29 84 0E 06 CF 3B 06 CD 0E 00  
1940 C3 2B 83 CF 0D 06 CD 0E 00 C3 1B 83 CF 23 05 DF  
1950 4D C3 SA 84 CD 8B 88 C3 67 84 CF 2C 06 CD CD 87  
1960 C3 4C 84 CD 0E 00 F7 DF C5 CD B1 88 C1 C3 5A 84  
1970 CD 34 89 DF D5 CD 57 88 C2 67 81 2A C1 89 E5 2A  
1980 CD 89 E5 21 00 00 22 C9 89 39 22 C3 89 C3 24 83  
1990 CD DC 87 2A C3 89 7C B5 CA E0 87 F9 E1 22 C3 89  
19A0 E1 22 C1 89 D1 CD 18 89 F7 CD 34 89 CD BA 87 2R  
19B0 22 C9 89 21 B0 82 C3 D8 82 DF 22 CD 89 2A C1 89  
19C0 C3 D8 82 DF C3 CA 84 21 01 00 22 CD 89 2A C1 89  
19D0 22 CF 89 EB 22 D1 89 01 0A 00 2A C9 89 EB 60 68  
19E0 39 3E 09 7E 23 B6 CA 03 85 7E 2B 8A C2 E2 84 7E  
19F0 BB C2 E2 84 EB 21 00 00 39 44 AD 21 0A 00 19 CD  
1A00 09 89 F9 2A D1 89 EB F7 FF DA E0 87 22 C5 89 D5  
1A10 EB 2A C9 89 7C B5 CA E1 87 E7 CA 27 85 D1 CD 18  
1A20 89 2A C5 89 C3 0F 85 E2 35 2A CR 89 E5 19 EB  
1A30 2A C9 89 73 23 72 2A CD 89 F1 B7 F2 3F 85 EB CD  
1A40 B2 87 D1 DA 51 85 2A CF 89 22 C1 89 2A D1 89 EB  
1A50 F7 CD 18 89 F7 21 00 00 3E DF 7C B5 C2 2B 83 CD  
1A60 75 88 D2 24 83 C3 81 81 2A C7 89 F9 E1 22 C1 89  
1A70 D1 D1 D5 CD BB 88 C3 80 85 FF DA BA 85 C3 90 85  
1A80 D5 FF DA E0 87 1A 4F 97 12 D1 CD 7F 88 79 1B 12  
1A90 D5 EB 2A C1 89 E5 21 72 85 22 C1 89 21 00 00 39  
1AA0 22 C7 89 D5 3E 3A CD 14 88 11 37 8F DF 00 00 00  
1AB0 D1 EB 73 23 72 E1 22 C1 89 D1 F1 CF 2C 03 C3 72  
1AC0 85 F7 1A FE 0D CA D1 85 CD BA 87 CF 2C 03 C3 8B  
1AD0 85 F7 21 BE 82 C3 D8 82 CD 01 86 D8 6F C9 CD 01  
1AE0 86 C8 6F C9 CD 01 86 C8 D8 6F C9 CD 01 86 6F C8  
1AF0 D8 6C C9 CD 01 86 C0 6F C9 CD 01 86 D0 6F C9 E1  
1A00 C9 79 E1 C1 E5 C5 4F CD 16 86 EB E3 CD E2 87 D1  
1A10 21 00 3E 01 C9 CF 2D 06 21 00 00 C3 40 86 CF  
1A20 28 00 CD 4A 86 CF 2B 15 E5 CD 4A 86 EB E3 7C AA  
1A30 7A 19 D1 FA 25 86 AC F2 25 86 C3 66 81 CF 2D 83  
1A40 E5 CD 4A 86 CD A6 87 C3 2C 86 CD A7 86 CF 2A 2C  
1A50 E5 CD 4A 86 CD 06 00 CD A3 87 EB E3 CD A3 87 C7 B7  
1A60 CA 69 86 7A B2 EB C2 67 81 7D 21 00 00 87 CA 99  
1A70 86 19 DA 67 81 3D C2 71 86 C3 99 86 CF 2F 44 E5  
1A80 CD A7 86 06 00 CD A3 87 EB E3 CD A3 87 7A B3 CA  
1A90 67 81 C5 CD 86 87 60 69 C1 D1 7C B7 FA 66 81 78  
1AA0 B7 FC A6 87 C3 4D 86 21 7E 82 C3 D8 82 FF DA B6  
1AB0 86 7E 23 66 6F C9 CD 3E 81 78 B7 C0 CF 28 05 DF  
1AC0 CF 29 01 C9 C3 86 7C 86 7C B7 FA 66 81 B5  
1AD0 CA 66 81 D5 E5 2A D3 89 11 B0 89 E7 DA E2 86 21  
1AE0 00 81 5E 23 56 22 D3 89 E1 EB C5 CD 86 87 C1 D1  
1AF0 23 C9 CD BC 86 86 CD A3 87 7C BA FA 66 81 C9 2A D5

D1C00 1DFF

1C00 89 D5 EB 21 00 8F CD 9C 87 D1 C9 DF 7D 32 R1 89  
1C10 7D 32 2F DF 7D CD B0 89 CF 2C 03 C3 0B 87 F7 DF  
1C20 7D 32 BA 89 CF 2C 1B DF E5 CF 2C 07 DF 7D E1 67  
1C30 C3 32 87 26 00 C3 R3 89 CD BC 86 7D 32 BC 89 26  
1C40 00 C3 BB 89 C3 E0 87 DF E5 CF 2C 12 DF 7D E1 77  
1C50 CF 2C 03 C3 47 87 F7 CD BC 86 6E 26 00 C9 C3 E0  
1C60 87 C5 CF 28 1C DF CF 29 07 D5 11 80 87 D5 E5 C9  
1C70 CF 2C 0E E5 DF CF 29 09 C1 D5 11 80 87 D5 C5 C9  
1C80 D1 C1 C9 C3 E0 87 E5 6C 26 00 CD 91 87 41 7D E1  
1C90 67 0E FF OC CD 9C 87 D2 93 87 19 C9 7D 93 6F 7C  
1CA0 9A 67 C9 7C B7 F0 7C 2F 67 7D 2F 6F 23 78 EE 80  
1CB0 47 C9 7C AA F2 B8 87 EB E7 C9 FF DA E0 87 E5 CF  
1CC0 3D 08 DF 44 4D E1 71 23 70 C9 C3 E0 87 CF 3B 04  
1CD0 F1 C3 2B 83 CF OD 04 F1 C3 1B 83 C9 EF FE OD C8  
1CE0 D5 11 75 81 97 CD 7F 88 D1 1A F5 97 12 2A C1 89  
1CF0 E5 7E 23 B6 D1 CA 81 81 7E R7 FA 68 85 CD ED 88  
1D00 1B F1 12 3E 3F D7 97 CD 7F 88 C3 81 81 D5 11 7B  
1D10 81 C3 E4 87 D7 11 37 8F CD 85 89 CA 18 88 FE 7E  
1D20 CA 42 88 FE 0A CA 18 88 B7 CA 18 88 FE 5C CA 4F  
1D30 88 D7 12 13 FE OD C2 3C 88 C3 AA 89 7B FE 87 C2  
1D40 18 88 7B FE 37 CA 4F 88 1B 3E 5F D7 C3 18 88 CD  
1D50 0E 00 3E 5E C3 14 88 7C B7 FA 66 81 11 D7 89 E5  
1D60 2A D5 89 2B E7 E1 D8 1A 95 47 13 1A 9C DA 74 88  
1D70 1B C9 13 13 1A FE OD C2 74 88 13 C3 5F 88 47  
1D80 1A 13 R8 C8 D7 FE OD C2 80 88 C9 CF 22 0F 3E 22  
1D90 CD 7F 88 FE OD E1 CA 1B 83 23 23 23 E9 CF 27 05  
1DA0 3E 27 C3 90 88 CF 5F 08 8D D7 D7 C1 88 09 88  
1DB0 C9 D5 11 0A 00 D5 42 OD CD A3 87 F2 E1 88 06 2D  
1DC0 OD C5 CD 86 87 7B B1 CA D2 88 E3 2D E5 60 69 C3  
1DD0 C2 88 C1 OD 79 R7 FA DF 88 3E 20 D7 C3 D3 88 7B  
1DE0 D7 5D 7B FE 0A D1 C8 C6 30 D7 C3 E2 88 1A 6F 13  
1DF0 1A 67 13 0E 04 CD R1 88 3E 20 D7 97 CD 7F 88 C9

D1E00 1FFF

1E00 E7 C8 1A 02 13 03 C3 09 89 C1 11 89 79 93  
1E10 C8 1B 2B 1A 77 C3 09 89 E1 22 CD 89 C1 E1 22 C9 89 7C B5 CA  
1E20 32 89 E1 22 CB 89 E1 22 CF 89 E1 22 CF 89 E1 22  
1E30 D1 89 C5 C9 21 AF 8F CD A6 87 C1 39 D2 OD 88 2A  
1E40 C9 89 7C B5 5A 89 2A D1 89 E5 2A CF 89 E5 2A  
1E50 CD 89 E5 2A CB 89 E5 2A C9 89 E5 C5 C9 C2 62 89  
1E60 F1 C9 F1 C5 D5 E5 32 BF 89 00 00 4F CD C4 05 3A  
1E70 BF 89 00 00 FE OD C2 7E 89 0E 0A CD C4 05 3A BF  
1E80 89 E1 D1 C1 C9 C5 D5 E5 DB 01 E6 02 CA 81 89 DB  
1E90 00 E6 7F FE 0F C2 A2 89 3A C0 89 2F 32 C0 89 C3  
1EA0 88 89 FE C9 DB FF AC A5 CA 83 89 F7 DB FF 6F C9 00  
1EB0 D3 FF C9 DB FF AC A5 CA 83 89 F7 DB FF 6F C9 00  
1EC0 FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
1ED0 00 00 00 00 01 D7 89 FF 7F 7F 20 50 4C 5A 2D  
1EE0 41 53 20 54 49 4E 59 20 42 41 53 49 43 20 35 2D  
1EF0 38 35 20 20 OD FF FF FF FF FF FF FF FF FF FF 3E  
1F00 FF 32 C0 89 3E 0C D7 97 11 D8 89 CD 7F 88 3A 07  
1F10 20 32 83 81 3D 32 19 81 32 26 81 32 A5 81 32 E5  
1F20 81 32 AB 85 32 05 87 32 17 88 32 36 89 21 84 81  
1F30 22 01 81 C3 84 81 29 00 69 73 0F C3 23 B7 02 E0  
1F40 CA 4F 88 D7 C9 FF FF FF FF FF FF FF FF FF FF 90  
1F50 E3 EF BE C3 2F 81 3E OD F5 3A C0 89 B7 C3 5D 89  
1F60 CD 16 86 E5 C3 D2 85 7C BA C0 7D BB C9 41 4E  
1F70 1A FE 20 C0 13 C3 28 00 F1 CD CD 87 C3 E0 87 47  
1F80 EF D6 40 D8 C3 03 81 F0 67 E8 7D 88 3D 01 2F D0  
1F90 2F 43 82 B8 7E 88 6F 99 DF EC 2E F8 CE C4 9F 6C  
1FA0 21 07 20 11 4F 8A 1A 77 23 13 3E 40 BD C2 A6 8A  
1FB0 21 FF 89 22 01 81 FF FF FF FF FF FF FF FF FF FF  
1FC0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
1FD0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
1FE0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
1FF0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

APPENDIX - ILIST OF ADDITIONAL SUB-ROUTINES IN KEYBOARD DISPLAYMONITOR

1.	0040 - 0048	Character out
2.	0050 - 005D	Line Mode
3.	006C - 0082	Key Test Loop
4.	0087 - 00AE	Screen Dump on Printer
5.	00B0 - 00B9	Local Mode
6.	00C4 - 00EB	Serial Printer Output
7.	0100 - 012D	Display Test
8.	0130 - 0144	Display Printable Character
9.	0145 - 0151	Next Line
10.	0152 - 0171	Scroll
11.	0172 - 018A	Cursor
12.	018E - 019E	Blank Last Line
13.	019F - 01A9	'CR'
14.	01AA - 01BA	'LF'
15.	01BB - 01C9	'TAB'
16.	01CD - 01D5	'HOME'
17.	01D6 - 01D9	'RIGHT'
18.	01E0 - 01F0	'UP'
19.	01F1 - 01FA	Left, Backspace

20.	01FB - 01FF	ESC
21.	0350 - 0370	INITIALIZE
22.	0590 - 05A8	Character In from Keyboard
23.	05A8 - 05B6	Test Print Monitor
24.	05C4 - 05CD	Charcter out, display on screen
25.	05DB - 05EA	Keyboard Call
26.	1600 - 1630	Keyscan (Call)
27.	1631 - 164D	CTRL code pressed
28.	164E - 145E	Codes above 40 (H) or equal
29.	165F - 168F	Alpha
30.	1670 - 1682	Keyscan (Main)
31.	16B3 - 16B6	No Key
32.	16B6 - 16BD	Return for Cursor Control
33.	1700 - 1708	Add 20
34.	1708 - 170F	New Code for SHF & D
35.	1710 - 1717	Substract 40 (H)
36.	1718 - 171F	Substract 20 (H)

Table : 16C0 - 16FF

Code Table for Keyscan

02

APPENDIX - J

This Programme was copied to Extra RAM Located at

C000

It Runs at 0000 Location only.

```

•LC000 C03F      INITIALIZATION AND RESTART ADDRESSES
C000 31A020      LP SP,20A0      ;INIT STACK POINTER
C003 C35003      JP 0350        ; JUMP TO INIT ROUTINE
C006 FF          RST 38H
C007 FF          RST 38H
C008 C30820      JP 2008        ;RST 1
C008 FF          ;ADDRESSES IN PAGE 0
C00C FF          RST 38H      ;POINT TO LOCATIONS IN
C00C FF          RST 38H      ;RAM AT 2008(H) AND UP
C00D FF          RST 38H      ;TO BE USED BY TINY BASIC
C00E 3E0D        LD A,0D
C010 C31020      JP 2010        ;USED BY TINY BASIC
C013 FF          RST 38H      ;RST 2
C014 FF          RST 38H
C015 FF          RST 38H
C016 FF          RST 38H
C017 FF          RST 38H
C018 C31820      JP 2018        ;RST 3
C01B FF          RST 38H
C01C FF          RST 38H
C01D FF          RST 38H
C01E FF          RST 38H
C01F FF          RST 38H
C020 C32020      JP 2020        ;RST 4
C023 FF          RST 38H
C024 C32420      JP 2024        ;TRAP (8085)
C027 FF          RST 38H
C028 C32820      JP 2028        ;RST 5
C02B FF          RST 38H
C02C C32C20      JP 202C        ;RST 5.5 (8085)
C02F FF          RST 38H
C030 C33020      JP 2030        ;RST 6
C033 FF          RST 38H
C034 C33420      JP 2034        ;RST 6.5 (8085)
C037 FF          RST 38H
C038 C33820      JP 2038        ;RST 7
C03B FF          RST 38H
C03C C33C20      JP 203C        ;RST 7.5 (8085)

```

```

* .LC0C40 C04C 'CHAR OUT '
C040 F5 'SAVE STATUS
C041 D801 IN A,(01)
C043 E604 AND 04 'OUT BUFF FULL ?
C045 CA4100 JP Z,0041 'WAIT IF SO
C048 F1 POP AF 'RESTORE STATUS
C049 D300 OUT (00),A 'OUT CHAR
C04B C9 RET
*
.LOCC050 C069 'LINE MODE '
C050 D801 IN A,(01) '8251 STATUS FLAG
C052 E602 AND 02 'INPUT BUFF
C054 CA6C00 JP Z,006C 'NO NEW CHAR ? TEST
C057 D800 IN A,(00) 'AGAIN, ELSE 'INPUT
C059 AF LD C,'A 'CHAR TO C REG
C05A CDC405 CALL 05C4 'DISPLAY IT
C05D C36C00 JP 006C 'JUMP TO KEY TEST
C060 FF RST 38H
C061 FF RST 38H
C062 FF RST 38H
C063 FF RST 38H
C064 FF RST 38H
C065 FF RST 38H
C066 C36620 JP 2066 'NMI JMP IN PAGE (20)H

.LC060C C087
C06C CD7201
C06F CDD805
C072 DA5000
C075 CDAB05
C078 CD4000
C07B D5
C07C 11FF7F
C07F CDF105
C082 D1
C083 C35000
C086 FF
* *
.LC087 C0AF
C087 E5
C088 D5
C089 C5
C08A 00
C08B 00
C08C 00
C08D 2100F0
C090 1418
C092 1E40
C094 0E0A
C096 CDC400
C099 0E0D
C09B CDC400
C09E 4E
C09F CDC400
C0A2 23
C0A3 1D
C0A4 C29E00
C0A7 15
C0AB C29200
C0AC C1
C0AD D1
C0AE E1
C0AF C9
* .LC0B0 C0BB
C0B0 CD9005
C0B3 4F
C0B4 CDC405
C0B7 C3B000
C0BA FF

'KEY TEST LOOP'
CALL 0172 'CURSOR
CALL 05DB 'KEYBD TEST
JP C,0050 'NOKEY ,JMP
CALL 05AB 'ON ANY KEY,TEST CTRL 0 ,P
CALL 0040 'KEY PRESSED,OUTPUT
PUSH DE
LD DE,7FFF 'DELAY
CALL 05F1
POP DE
JP 0050 'JMP LOOP
RST 38H

'PRINT SCREEN '
PUSH HL 'SAVE
PUSH DE
PUSH BC
NOP
NOP
NOP
LD HL,F000 'SCREEN MEM POINTER
LD D,1B '24 LINES
LD E,40 '64 CHAR PER LINE
LD C,0A CR
CALL 00C4 'CALL PRINT
LD C,0D 'LF
CALL 00C4 'PRINT
LD C,(HL) 'CHAR TO PRINT
CALL 00C4 'PRINT IT
INC HL 'NEXT CHAR
DEC E 'FULL LINE OVERT
JP NZ,009E 'IF NOT DO IT
DEC D 'FULL PAGE OVERT ?
JP NZ,0092 'PRINT REMAINING
POP BC 'LINES

'LOCAL '
CALL 0590 'READ KEYBOARD
LD C,'A 'DATA IN C REG
CALL 05C4 'DISPLAY IT
JP 00B0 'LOOP BACK
RST 38H

```

```

* .LC0C4 COEC 'PRINT OUT'
C0C4 F3          ;DISABLE INT
C0C5 C5          ;SAVE
C0C6 D5          ;XMIT '1' MARK
C0C7 CDF000     ;BIT COUNT
C0CA 0607       ;ONE BIT DELAY
C0CC 110202     ;DELAY VIA Z0R1(H)
C0CF CDEA07     ;DATA IN A
C0D2 79         ;LSB IN CARRY BIT
C0D3 1F         ;SAVE RESR OF DATA
C0D4 4F         ;NO CARRY XMIT '1'
C0D5 I4F000    ;CARRY XMIT '0'
C0D8 DCFB00     ;
C0DB 05         ;DO ALL BITS
C0DC F2CC00    ;STOP BIT
C0DF CDFB00     ;
C0E2 110403    ;2 BIT DELAY
C0E5 CDEA07     ;
C0E8 D1         ;
C0E9 C1         ;ENABLE INT
C0EA FB         ;
C0EB C9         ;

*
.LC0F0 COFB
C0F0 DB0A      'PORT C BIT PC2 USED FOR SERIAL
C0F2 F604      DATA OUT TO PRINTER'
C0F4 D30A      IN A,(0A) ;XMIT '1'
C0F6 C9        OR 04
C0F7 FF        OUT (0A),A
               RET
.LC0F8 COFF
C0FB DB0A      IN A,(0A) ;XMIT '0'
C0FA E6FB     AND FB
C0FC D30A     OUT (0A),A
C0FE C9       RET

```



```

*
.LC152 C172 'SCROLL'
C152 C5 PUSH BC ;SAVE
C153 D5 PUSH DE
C154 E5 PUSH HL
C155 1100F0 LD DE,F000 ;FIRST LOC OF SCREEN
C158 2140F0 LD HL,F040 ;FIRST CHAR OF 2ND LIN
C15B 01C005 LD BC,05C0 ;DISPLACEMENT OF LAST LIN
C15E 7E LD A,(HL) ;COPY 2ND LINE TO 1ST LIN
C15F 12 LD (DE),A ;
C160 13 INC,DE ;
C161 23 INC HL ;
C162 08 DEC BC ;TILL 24TH LINE
C163 78 LD A,B ;
C164 B1 OR C ;
C165 C25E01 JP NZ,015E ;
C168 C08E01 CALL 018E ;BLANK 24TH LINE
C16B E1 POP HL ;
C16C 21C0F5 LD HL,F5C0 ;HL POINTS TO 24TH LINE
C16F D1 POP DE ;
C170 C1 POP RC ;
C171 C9 RET ;
*
.LC172 C18C 'CURSOR:0E &OF ARE 6845 REOS'
C172 F5 PUSH AF ;SAVE
C173 3E0E LD A,0E ;REG OE
C175 3200FB LD (F800),A ;
C178 3AB920 LD A,(20B9) ;DATA AT 20B9(H)
C17B 3201FB LD (F801),A ;IN OE REG
C17E 3E0F LD A,0F ;OF REG
C180 3200FB LD (F800),A ;
C183 3AB820 LD A,(20B8) ;DATA AT 20B8(H)
C186 3201FB LD (F801),A ;IN OF REG
C189 F1 POP AF ;
C18A C9 RET ;
C18B FF RST 38H ;
*
.LC18E C19F 'BLANK LAST LINR'
C18E 21C0F5 LD HL,F5C0 ;POINT TO LAST LINE
C191 014000 LD RC,0040 ;64 CHAR
C194 3E20 LD A,20 ;SPACE OR BLANK
C196 77 LD (HL),A ;
C197 23 INC HL ;NEXT LOC
C198 08 DEC BC ;64 CHAR OVER
C199 78 LD A,B ;
C19A B1 OR C ;
C19B C29401 JP NZ,0194 ;IF NO THEN DO IT
C19E C9 RET ;

```

```

'DISPLAY TEST'
PRINTABLE DATA ABOVE 20(H)
CP 20 ;
JP NC,0130 ;DATA ABOVE 20(H)
CP OD ;CR
JP Z,019F ;LF
CP OA ;
JP Z,01AA ;TAB
CP O9 ;
JP Z,01BB ;BACK SPACE
CP OB ;
JP Z,01F1 ;UP
CP OB ;
JP Z,01E0 ;RIGHT
CP OC ;
JP Z,01D6 ;HOME
CP 1E ;
JP Z,01CD ;ESCAPE
CP 1B ;
JP Z,01FB ;
RET ;
RST 38H ;
*
.LC130 C145 'DISPLAY PRINTABLE CHARACTERSS'
C130 F5 PUSH AF ;SAVE
C131 E5 PUSH HL ;
LD HL,(20B8) ;CURSOR ADDR
C132 2AB820 LD (HL),A ;DISPLAY DATA AT
C135 77 INC HL ;THIS LOC, THEN NEXT
C136 23 LD A,'L ;LO ADDR
C137 7D AND 3F ;64 LOC OVER?
C138 E63F NOP ;
C13A 00 NOP ;
C13B 00 NOP ;
C13C CC4B01 CALL Z,014B ;IF YES THEN JUMP
C13F 22B820 LD (20B8),HL ;OTHERWISE STORE
C142 E1 POP HL ;
C143 F1 POP AF ;ADDR AND RESTORE
C144 C9 RET ;
*
.LC145 C152 'NEXT LINE'
C145 C5 PUSH BC ;SAVE
C146 011800 LD RC,001B ;ADD 24 TO ADDR POINTER
C149 09 ADD HL,BC ;POINTER
C14A C1 POP BC ;
C14B 7C LD A,H ;HI ADDR
C14C FEF6 ;POINTS TO 24TH LINE?
C14E CC5201 CALL Z,0152 ;THEN CALL SCROLL
C151 C9 RET ;

```

```

*
.LC152 C172 'SCROLL'
C152 C5 PUSH BC ;SAVE
C153 D5 PUSH DE
C154 E5 PUSH HL
C155 1100F0 LD DE,F000 ;FIRST LOC OF SCREEN
C158 2140F0 LD HL,F040 ;FIRST CHAR OF 2ND LIN
C15B 01C005 LD BC,05C0 ;DISPLACEMENT OF LAST LIN
C15E 7E LD A,(HL) ;COPY 2ND LINE TO 1ST LIN
C15F 12 LD (DE),A ;
C160 13 INC,DE ;
C161 23 INC HL ;
C162 08 DEC BC ;TILL 24TH LINE
C163 78 LD A,B ;
C164 B1 OR C ;
C165 C25E01 JP NZ,015E ;
C168 C08E01 CALL 018E ;BLANK 24TH LINE
C16B E1 POP HL ;
C16C 21C0F5 LD HL,F5C0 ;HL POINTS TO 24TH LINE
C16F D1 POP DE ;
C170 C1 POP RC ;
C171 C9 RET ;
*
.LC172 C18C 'CURSOR:0E &OF ARE 6845 REOS'
C172 F5 PUSH AF ;SAVE
C173 3E0E LD A,0E ;REG OE
C175 3200FB LD (F800),A ;
C178 3AB920 LD A,(20B9) ;DATA AT 20B9(H)
C17B 3201FB LD (F801),A ;IN OE REG
C17E 3E0F LD A,0F ;OF REG
C180 3200FB LD (F800),A ;
C183 3AB820 LD A,(20B8) ;DATA AT 20B8(H)
C186 3201FB LD (F801),A ;IN OF REG
C189 F1 POP AF ;
C18A C9 RET ;
C18B FF RST 38H ;
*
.LC18E C19F 'BLANK LAST LINR'
C18E 21C0F5 LD HL,F5C0 ;POINT TO LAST LINE
C191 014000 LD RC,0040 ;64 CHAR
C194 3E20 LD A,20 ;SPACE OR BLANK
C196 77 LD (HL),A ;
C197 23 INC HL ;NEXT LOC
C198 08 DEC BC ;64 CHAR OVER
C199 78 LD A,B ;
C19A B1 OR C ;
C19B C29401 JP NZ,0194 ;IF NO THEN DO IT
C19E C9 RET ;

```

```

'DISPLAY TEST'
PRINTABLE DATA ABOVE 20(H)
CP 20 ;
JP NC,0130 ;DATA ABOVE 20(H)
CP OD ;CR
JP Z,019F ;LF
CP OA ;
JP Z,01AA ;TAB
CP O9 ;
JP Z,01BB ;BACK SPACE
CP OB ;
JP Z,01F1 ;UP
CP OB ;
JP Z,01E0 ;RIGHT
CP OC ;
JP Z,01D6 ;HOME
CP 1E ;
JP Z,01CD ;ESCAPE
CP 1B ;
JP Z,01FB ;
RET ;
RST 38H ;
*
.LC130 C145 'DISPLAY PRINTABLE CHARACTERSS'
C130 F5 PUSH AF ;SAVE
C131 E5 PUSH HL ;
LD HL,(20B8) ;CURSOR ADDR
C132 2AB820 LD (HL),A ;DISPLAY DATA AT
C135 77 INC HL ;THIS LOC, THEN NEXT
C136 23 LD A,'L ;LO ADDR
C137 7D AND 3F ;64 LOC OVER?
C138 E63F NOP ;
C13A 00 NOP ;
C13B 00 NOP ;
C13C CC4B01 CALL Z,014B ;IF YES THEN JUMP
C13F 22B820 LD (20B8),HL ;OTHERWISE STORE
C142 E1 POP HL ;
C143 F1 POP AF ;ADDR AND RESTORE
C144 C9 RET ;
*
.LC145 C152 'NEXT LINE'
C145 C5 PUSH BC ;SAVE
C146 011800 LD RC,001B ;ADD 24 TO ADDR POINTER
C149 09 ADD HL,BC ;POINTER
C14A C1 POP BC ;
C14B 7C LD A,H ;HI ADDR
C14C FEF6 ;POINTS TO 24TH LINE?
C14E CC5201 CALL Z,0152 ;THEN CALL SCROLL
C151 C9 RET ;

```

```

*
.LC19F C1AA
'CR'
PUSH AF
LD A,(20B8) ;CURSOR ADDR
AND C0 ;LOWER 6 BITS ZERO
LD (20B8),A ;IT MEANS LEFT MOST
POP AF
RET
*
.LC1AA C1BA
'LINEFEED'
PUSH AF
PUSH HL
PUSH DE
LD HL,(20B8) ;
LD DE,0040 ;ADD 64 TO CURSOR
ADD HL,DE ;ADDRESS
CALL 014B ;TEST FOR LAST LINE
JP 01EA ;STORE AND RET
*
.LC1BB C1CB
'TAB'
PUSH HL
LD HL,(20B8) ;
LD A,L ;LO ADDR
ADD A,08 ;ADD 08
AND F8 ;3 LO BITS ZERO
LD L,A ;MEANS TAB
LD (20B8),HL ;
POP HL
RET
RST 38H
*
'HOME'
*
.LC1CD C1D6
C1D6 E5
C1D6 E5
C1CE 2100F0
C1D1 22B820
C1D4 E1
C1D5 C9
*
.LC1D6 C1E0
C1D6 E5
C1D7 2AB820
C1DA 23
C1DR 22B820
C1DE E1
C1DF C9
PUSH HL
LD HL,(20B8) ;ADD +1 TO
INC HL
LD (20B8),HL ;CURSOR ADDR
POP HL
RET
*
.LC19F C1AA
'CR'
PUSH AF
LD A,(20B8) ;CURSOR ADDR
AND C0 ;LOWER 6 BITS ZERO
LD (20B8),A ;IT MEANS LEFT MOST
POP AF
RET
*
.LC1AA C1BA
'LINEFEED'
PUSH AF
PUSH HL
PUSH DE
LD HL,(20B8) ;
LD DE,0040 ;ADD 64 TO CURSOR
ADD HL,DE ;ADDRESS
CALL 014B ;TEST FOR LAST LINE
JP 01EA ;STORE AND RET
*
.LC1BB C1CB
'TAB'
PUSH HL
LD HL,(20B8) ;
LD A,L ;LO ADDR
ADD A,08 ;ADD 08
AND F8 ;3 LO BITS ZERO
LD L,A ;MEANS TAB
LD (20B8),HL ;
POP HL
RET
RST 38H
*
'HOME'
*
.LC1CD C1D6
C1D6 E5
C1D6 E5
C1CE 2100F0
C1D1 22B820
C1D4 E1
C1D5 C9
*
.LC1D6 C1E0
C1D6 E5
C1D7 2AB820
C1DA 23
C1DR 22B820
C1DE E1
C1DF C9
PUSH HL
LD HL,(20B8) ;ADD +1 TO
INC HL
LD (20B8),HL ;CURSOR ADDR
POP HL
RET
*
.LC19F C1AA
'UP'
PUSH AF
PUSH HL
PUSH DE
LD HL,(20B8) ;CURSOR ADDR
LD DE,FFC0 ;SUB 64 IN 2'S COMPLEMENT
ADD HL,DE ;
LD (20B8),HL ;STORE
POP DE
POP HL
POP AF
RET
*
.LC1F1 C1FB
'LEFT:BACKSPACE'
PUSH HL
LD HL,(20B8) ;CURSOR ADDR
DEC HL ;-1
LD (20B8),HL ;STORE
POP HL
RET
*
.LC1FB C1FF
'ESC'
LD C,24
JP 05C4
*
.LC1E0 C1F1
C1E0 F5
C1E1 E5
C1E2 D5
C1E3 2AB820
C1E6 11C0FF
C1E9 19
C1EA 22B820
C1ED D1
C1EE E1
C1EF F1
C1F0 C9
*
.LC1F1 C1FB
C1F1 E5
C1F2 2AB820
C1F5 2B
C1F6 22B820
C1F9 E1
C1FA C9
*
.LC1FB C1FF
C1FB 0E24
C1FD C3C405
*

```

CASSETTE TAPE I/O  
 \* .LC200 C215  
 C200 DR0A ;100 USEC PULSE\*  
 C202 F602 IN A,(OA)  
 C204 R30A OR 02 ;FC2 IS MADE '1'  
 C206 D5 PUSH DE ; SAVE 'D'E'  
 C207 1619 ;DELAY 100 USEC  
 C209 15 DEC D ;  
 C20A C20P02 JP NZ,0209 ;BCLK 4 MHZ  
 C20D D1 POP DE  
 C20E DR0A IN A,(OA)  
 C210 E63D AND 3D  
 C212 D30A OUT (OA)\*A ;FC2 IS MADE '0'  
 C214 C9 RET  
 \* \*  
 .LC215 C221 ;100 USEC PULSE, 400 USEC DELAY,  
 C215 CD0002 CALL 0200  
 C218 D5 PUSH DE ;100 USEC PULSE  
 C219 166B LD D,6B  
 C21B 15 DEC D C:400 USEC DELAY  
 C21C C21R02 JP NZ,021B  
 C21F D1 POP DE  
 C220 C9 RET  
 \* \*  
 .LC221 C22F ;ZERO',100 USEC PULSE 900USEC DELAY  
 C221 CD0002 CALL 0200 ;100 USEC PULSE  
 C224 D5 PUSH DE ;900 USEC DELAY  
 C225 16FB LD D,6B  
 C227 15 DEC D  
 C228 C22702 JP NZ,0227  
 C22B D1 POP DE  
 C22C C9 RET 38H  
 C22D FF RST 38H  
 C22E FF ONE BIT, BIT WRITE  
 \* \*  
 .LC230 C250  
 C230 79 LD A ,C ;DATA FROM C TOA  
 C231 17 RLA ;MSB IN CARRY  
 C232 4F LD C ,A  
 C233 DA3902 JP C,0239 ; CARRY =1 'ONE'  
 C236 C32102 JP 0221 ; CARRY =0 'ZERO'  
 C239 CD1502 CALL 0215 ; 'ONE', 100U PULSE &  
 C23C CD1502 CALL 0215 ;400U DELAY TWICE  
 C23F C9 RET  
 C240 060B LD B,0B ;BYTE WRITE,8 BIT IN B  
 C242 CD3002 CALL 0230 ;BIT WRITE  
 C245 05 DEC B  
 C246 C24202 JP NZ,0242 ;ALL BITS OVER?  
 C249 11A400 LD DE,00A4 ;IF YES 1 MSEC GAP  
 C24C CDF105 CALL 05F1 ;EQUIL TO ONE BIT TIME  
 C24F C9 RET  
 \* \* \* \*  
 \* .LC250 C279  
 C250 060B LD B,0B ;BIT COUNT  
 C252 0E00 LD C,00 ;DATA INITIALISE  
 C254 1600 LD D,00 ;TIME COUNT INITIALISE  
 C256 14 INC D ;COUNT IN D REG  
 C257 DROA IN A,(OA) ;TEST PC4  
 C259 E610 AND 10  
 C25B C25602 JP NZ,0256 ; IF IT IS '1' THEN COUNT  
 C25E 14 INC D ;ELSE IF '0'  
 C25F DROA IN A,(OA) ; TEST PC4 AGAIN  
 C261 E610 AND 10  
 C263 C45E02 JP Z,025E ; IF IT IS '0' THEN COUNT  
 C266 05 DEC B ;ONE CYCLE RISE TIME TO NXT ONE  
 C267 CA7902 JP Z,0279 ;ALL BITS OVER THEN JUMPOVER  
 C26A 7A LD A ,D ; COUNT IN D, BRING IT TO A  
 C26B FE53 CP 53 ; COMPARE TO 53, '>'0', '<'IS '1',  
 C26D CDB802 CALL 028B ; SAVE AND SHIFT  
 C270 D25402 JP NC,0254 ;IF NO CARRY THEN MORE BITS  
 C273 CDF002 CALL 0290 ;WAIT  
 C276 C35402 JP 0254 ;REPEAT  
 \* \*  
 .LC279 C28B ;MEASURE TIME DELAY,  
 C279 7A LD A ,D ;COUNT IN A REG  
 C27A FE66 CP A6 ; COMPARE TO A6 (H)  
 C27C DDB802 CALL 028B ; LAST BIT '0', THEN ALL OVER  
 C27F D28502 JP NC,0285 ;ELSE WAIT FOR PULSE  
 C282 CDF002 CALL 0290  
 C285 79 LD A ,C ;GET DATA BACK IN A REG  
 C286 C9 RET  
 C287 FF RST 38H  
 \* \*  
 .LC288 C28F ;SAVE AND SHIFT, COMPARE AND INSERT CARRY,  
 C288 F5 PUSH AF ; SAVE A REG  
 C289 79 LD A ,C ;DATA IN C REG TO A REG  
 C28A 17 RLA ; CARRY TO LSB  
 C28B 4F LD C ,A ;ONE MORE BIT COMPILED  
 C28C F1 POP AF ; RESTORE A REG  
 C28D C9 RET  
 C28E FF RST 38H  
 \* \*  
 .LC290 C29F ;WAIT,  
 C290 DR0A IN A,(OA) ;STATUS OF PC4  
 C292 E610 AND 10  
 C294 C29002 JP NZ,0290 ; WAIT AT 'ONE'  
 C297 DR0A IN A,(OA) ;STATUS OF PC4  
 C299 E610 AND 10  
 C29B CA9702 JP Z,0297 ;WAIT AT 'ZERO'  
 C29E C9 RET  
 \* \* \* \*  
 \* \* \* \*  
 \* \* \* \*

CASSETTE TAPE I/O  
 \* .LC200 C215  
 C200 DR0A ;100 USEC PULSE\*  
 C202 F602 IN A,(OA)  
 C204 R30A OR 02 ;FC2 IS MADE '1'  
 C206 D5 PUSH DE ; SAVE 'D'E'  
 C207 1619 ;DELAY 100 USEC  
 C209 15 DEC D ;  
 C20A C20P02 JP NZ,0209 ;BCLK 4 MHZ  
 C20D D1 POP DE  
 C20E DR0A IN A,(OA)  
 C210 E63D AND 3D  
 C212 D30A OUT (OA)\*A ;FC2 IS MADE '0'  
 C214 C9 RET  
 \* \*  
 .LC215 C221 ;100 USEC PULSE, 400 USEC DELAY,  
 C215 CD0002 CALL 0200  
 C218 D5 PUSH DE ;100 USEC PULSE  
 C219 166B LD D,6B  
 C21B 15 DEC D C:400 USEC DELAY  
 C21C C21R02 JP NZ,021B  
 C21F D1 POP DE  
 C220 C9 RET  
 \* \*  
 .LC221 C22F ;ZERO',100 USEC PULSE 900USEC DELAY  
 C221 CD0002 CALL 0200 ;100 USEC PULSE  
 C224 D5 PUSH DE ;900 USEC DELAY  
 C225 16FB LD D,6B  
 C227 15 DEC D  
 C228 C22702 JP NZ,0227  
 C22B D1 POP DE  
 C22C C9 RET 38H  
 C22D FF RST 38H  
 C22E FF ONE BIT, BIT WRITE  
 \* \*  
 .LC230 C250  
 C230 79 LD A ,C ;DATA FROM C TOA  
 C231 17 RLA ;MSB IN CARRY  
 C232 4F LD C ,A  
 C233 DA3902 JP C,0239 ; CARRY =1 'ONE'  
 C236 C32102 JP 0221 ; CARRY =0 'ZERO'  
 C239 CD1502 CALL 0215 ; 'ONE', 100U PULSE &  
 C23C CD1502 CALL 0215 ;400U DELAY TWICE  
 C23F C9 RET  
 C240 060B LD B,0B ;BYTE WRITE,8 BIT IN B  
 C242 CD3002 CALL 0230 ;BIT WRITE  
 C245 05 DEC B  
 C246 C24202 JP NZ,0242 ;ALL BITS OVER?  
 C249 11A400 LD DE,00A4 ;IF YES 1 MSEC GAP  
 C24C CDF105 CALL 05F1 ;EQUIL TO ONE BIT TIME  
 C24F C9 RET  
 \* \* \* \*  
 \* \* \* \*

```

*
*
* LC2A0 C2AD
*
LD D,80 ;LEADER'
LD C,E ;128 TIMES RECORD
PUSH DE ; CONTENTS OF E REG
CALL 02A0 ;BYTE WRITE
POP DE
DEC D
JP NZ,02A2
RET
*
* LC2AD C2BB
*
C2AD C5 ;SAVERC, HL IS DATA POINTER
C2AE 4E ;LD C,(HL);BC HAS NO OF BYTES
C2AF CD4002 ;CALL 0240 ;CALL BYTE WRITE
C2B2 23 INC HL
C2B3 C1 POP BC
C2B4 0B DEC BC
C2B5 7B LD A,B
C2B6 B1 OR C
C2B7 C2AD02 JP NZ,02AD;ALL BYTES OVER?IF NOT
C2BA C9 RET
*
* LC2BB C2C8
*
C2BB C5 ;PUSH BC ;SAVE BC
C2BC 1E00 LD E,00 ;NULLS
C2BE CD4002 CALL 02A0 ;RECORDED N TIMES
C2C1 1EFF LD E,FF ;FF'S RECORDED
C2C3 CD4002 CALL 02A0 ; N TIMES
C2C6 C1 POP EC
C2C7 C9 RET
*
* LC2C8 C2DC
*
C2C8 D5 ;PUSH DE
C2C9 111000 LD DE,0010 ;DELAY COUNT IN DE REG PAIR
C2CC D5 ;PUSH DE
C2CD 11FFFF LD DE,FFFF ; DOUBLE LOOP
C2D0 CD105 CALL 05F1
C2D3 D1 POP DE
C2D4 1B DEC DE
C2D5 7A LD A,D
C2D6 B3 OR E
C2D7 C2CC02 JP NZ,02CC
C2DA D1 POP DE
C2DB C9 RET
*
* LC2A0 C2AD
*
LD D,80 ;LEADER'
LD C,E ;128 TIMES RECORD
PUSH DE ; CONTENTS OF E REG
CALL 02A0 ;BYTE WRITE
POP DE
DEC D
JP NZ,02A2
RET
*
* LC2AD C2BB
*
C2AD C5 ;SAVERC, HL IS DATA POINTER
LD C,(HL);BC HAS NO OF BYTES
CALL 0240 ;CALL BYTE WRITE
INC HL
POP BC
DEC BC
LD A,B
OR C
JP NZ,02AD;ALL BYTES OVER?IF NOT
RET
*
* LC2BB C2C8
*
PUSH BC ;SAVE BC
LD E,00 ;NULLS
CALL 02A0 ;RECORDED N TIMES
LD E,FF ;FF'S RECORDED
CALL 02A0 ; N TIMES
POP EC
RET
*
* LC2C8 C2DC
*
PUSH DE
LD DE,0010 ;DELAY COUNT IN DE REG PAIR
PUSH DE
LD DE,FFFF ; DOUBLE LOOP
CALL 05F1
POP DE
DEC DE
LD A,D
OR E
JP NZ,02CC
POP DE
RET
*
* LC2A0 C2FB
*
C2DC D5 ;PUSH DE ;SAVE
C2DD 1620 LD D,20 ; SEARCH 32 FF'S
C2DF D5 ;PUSH DE
C2E0 CD5002 CALL 0250 ; BYTE READ
C2E3 D1 POP DE
C2E4 FEFF CP FF ; IS IT FF
C2E6 C2DD02 JP NZ,02DD ; IF NOT THEN REPEAT
C2E9 15 DEC D ;IF YES THEN
C2EA C2DF02 JP NZ,02DF ;CONTINUE TILL 32 FF'S
C2ED CD5002 CALL 0250 ; BYTE READ
C2F0 FEFF CP FF ; IS IT FF
C2F2 CAED02 JP Z,02ED ; WAIT TILL ALL FF'S OVER
C2F5 D1 POP DE
C2F6 C9 RET
C2F7 FF RST 3BH
*
* LC2FB C2FF
*
C2FB DB0A ;'MOTOR OFF'
C2FA E6FE IN A,(0A) ;FC0 MADE '0'
C2FC D30A AND FE
C2FE C9 OUT (0A),A
RET
*
* LC300 C30E
*
C300 C5 ;PUSH BC ;HL MEM POINTR BC BYTE
C301 CD5002 CALL 0250 ; COUNTER, READ ONE BYTE
C304 C1 POP BC ;
C305 77 LD (HL),A ;LOAD FIRST BYTE IN RAM
C306 23 INC HL ;ADVANCE POINTER
C307 0B DEC RC ;MEELCK COUNT REDUCED
C308 7B LD A,B
C309 B1 OR C
C30A C20003 JP NZ,0300 ;FULL BLOCK READ?
C30D C9 RET ; THEN ALL DONE
*
* LC30E C313
*
C30E 3EBA ;LD A,BA ; CODE 88(H)
C310 D30B OUT (0B),A ;FOR CASSETTE I/O
C312 C9 RET
*
* LC313 C31B
*
C313 DB0A ;'MOTOR ON'
C315 F601 IN A,(0A) ;PC1 IS MADE '1'
C317 D30A OR 01
C319 C9 OUT (0A),A
C31A FF RST 3BH

```

```

*
.LC330 C38F
INITIALIZE
LD C,06 #6 BYTES
LD HL,039A #AT 039A
CALL 0390 #DELAY 0.5 MSEC
LD A,(HL) #BYTE ONE
OUT (01),A #TO CONTROL OF 8251
INC HL #NEXT BYTE
DEC C #BYTE COUNT
JP NZ,0355 #SEND ALL 6 BYTES
LD A,8A #83 8255 INIT,PA OUT PB OUT
OUT (0B),A #PCLO OUT PCHI IN
CALL 03A3 #INIT 6845
LD HL,F000 #BLANK SCREEN
LD (208B),HL
LD BC,0600
CALL 0194
CALL 0172 #CURSOR BLINK
JP 03E0 #REST IS MONITOR
RST 3BH
C379 FF

.LC390 C39A
,DELAY 0.5 MSEC
PUSH DE
LD DE,003F
CALL 05F1
POP DE
RET
RST 3BH
*
.DOC39A C39F #8251 INIT BYTES
C39A 00 00 00 40 CE 05
*
.LC590 C5A7
'CI: CHAR IN '
C590 CD7201 #CURSOR
C593 CDD805 #CALL KEYBD
C596 DA9305 #WAIT FOR KEY PRESS
C599 CDA805 #CALL TEST CTRL O P
C59C F5 #PUSH AF
C59D D5 #PUSH DE
C59E 11FF7F #LD DE,7FFF
C5A1 CDF105 #CALL 05F1
C5A4 D1 #POP DE
C5A5 F1 #POP AF
C5A6 C9 #RET

*
.LC3AB C5B7
'TEST CTRL O & P FOR MONITOR JMP OR PRINT
PUSH AF #SAVE
PUSH AF
CP 10 #CTRL P ?
CALL Z,0087 #PRINT SCREEN
POP AF #ELSE TEST FOR
CP 0F #CTRL O ?
JP Z,03FA #IF YES JMP MONITOR
POP AF
RET

*
.LC5C4 C5CF
'CHAR OUT:DISPLAY'
LD A,C #DATA IN C REG
AND 7F #KILL MSB
CALL 0100 #DISPLAY
CALL 0172 #CURSOR BLINK
RET
RST 3BH

*
.LC5DB C5E4
'KEYBD '
PUSH HL #SAVE
PUSH DE
PUSH BC
CALL 1600 #CALL KEYSKAN
POP BC
POP DE
POP HL

*
C5A8 F5
C5A9 F5
C5AA FE10 #CTRL P ?
C5AC CD8700
C5AF F1
C5B0 FE0F
C5B2 CFAA03
C5B5 F1
C5B6 C9

*
C5C4 79
C5C5 E67F
C5C7 CD0001
C5CA CD7201
C5CD C9
C5CE FF

*
C5DB E5
C5DD D5
C5DD C5
C5DE CD0016
C5E1 C1
C5E2 D1
C5E3 E1

```

```

*
.LD600 D631
D600 00
D601 00
D602 00
D603 CD7016
D606 DA2D16
D609 7A
D60A E688
D60C CA1616
D60F 2E37
D611 2C
D612 17
D613 D21116
D616 2616
D618 7D
D619 F6C0
D61B 6F
D61C 7E
D61D 4F
D61E FE40
D620 D25116
D623 7A
D624 E602
D626 CA3116
D629 79
D62A 97
D62B 4F
D62C 79
D62D 00
D62E 00
D62F 00
D630 C9
*
.LD631 D64E
D631 79
D632 FE21
D634 DA4E16
D637 7A
D638 00
D639 E644
D63B CA2C16
D63E 79
D63F FE30
D641 CA0817
D644 DC0017
D647 79
D648 D610
D64A 4F
D64E C32C16
;
;
;
;
;CALL KEY SCAN
;NO KEY PRESSED RET
;KEY PRESSED,CHECK
;FOR CURSOR KEYS
;IF YES,THEN POINT
;TABLE AT 10F8(FH)
;ADVANCE & TEST
;FOR KEY PRESSED
;
;TABLE HI
;KEY CODE IN ACC
;KEYSCAN LO IN 'L'
;POINT TO TABLE ENTRY
;KEY CODE IN ACC
;IN C REG
;
;JUMP TO ALPHA
;TEST FOR CTRL KEY
AND 02
;IF NOT PRESSED JUMP
;ELSE IT WAS WRONG
;NOTHING TO SEND
;
;CODE IN 'A'
;CTRL NOT PRESSED'
LD A ,C
CP 21
;CODE >20(H)?
JP C ,164E
LD A ,D
;IT IS
;ELSE
AND 44
;TEST FOR SHIFT KEY
JP Z ,162C
;IF NOT PRESSED SEND AS
LD A ,C
CP 30
;
JP Z ,1708
;=30(H),SEND NEW CODE
CALL C ,1700;<30(H),ADD 20(H) TO GET
LD A ,C
;SHIFT,
SUB 10
;SUBTRACT 10(H) FOR
LD C ,A
;PUNCTUATION
JP 162C
;SEND IT NOW

'KEYSCAN'
;
;
;
;CALL KEY SCAN
;NO KEY PRESSED RET
;KEY PRESSED,CHECK
;FOR CURSOR KEYS
;IF YES,THEN POINT
;TABLE AT 10F8(FH)
;ADVANCE & TEST
;FOR KEY PRESSED
;
;TABLE HI
;KEY CODE IN ACC
;KEYSCAN LO IN 'L'
;POINT TO TABLE ENTRY
;KEY CODE IN ACC
;IN C REG
;
;JUMP TO ALPHA
;TEST FOR CTRL KEY
AND 02
;IF NOT PRESSED JUMP
;ELSE IT WAS WRONG
;NOTHING TO SEND
;
;CODE IN 'A'
;CTRL NOT PRESSED'
LD A ,C
CP 21
;CODE >20(H)?
JP C ,164E
LD A ,D
;IT IS
;ELSE
AND 44
;TEST FOR SHIFT KEY
JP Z ,162C
;IF NOT PRESSED SEND AS
LD A ,C
CP 30
;
JP Z ,1708
;=30(H),SEND NEW CODE
CALL C ,1700;<30(H),ADD 20(H) TO GET
LD A ,C
;SHIFT,
SUB 10
;SUBTRACT 10(H) FOR
LD C ,A
;PUNCTUATION
JP 162C
;SEND IT NOW

*
.LD64E D65F
D64E 3F
D64F C9
D650 00
D651 FE7B
D653 DA5F16
D656 7A
D657 E644
D659 C41817
D65C C32C16
*
.LD65F D670
D65F 7A
D660 E647
D662 CA2C16
D665 E602
D667 C41017
D66A CD1B17
D66D C32C16
*
CODES ABOVE 40(H) OR EQUAL
CCF
RET
NOP
CP 7B
;CODE<7A(H)
JP C ,165F
LD A ,D
AND 44
CALL NZ ,171B
JP 162C
'ALPHA'
LD A ,D
AND 47
;TEST FOR CAPS LOCK
;AND SHIFT
JP Z ,162C
;IF NOT PRESSED THEN
AND 02
;SEND AS IT IS,IF CTRL
CALL NZ ,1710
;THEN SUBTRACT 40(H)
CALL 171B
;SUBTRACT 20(H) AND
JP 162C

```

LD670 D684

```

D670 0100FF      'KEYSCAN'
D673 110000      LD BC,FF00
D676 37          LD DE,0000
D677 79          SCF
D678 17          LD A,C
D679 17          RLA
D679 4F          LD C,A
D67A 2F          CPL
D67B D308        OUT (08),A
D67D CD9003      CALL 0390
D680 DB09        IN A,(09)
D682 2F          CPL
D683 57          LD D,A
D684 E6B8        AND B
D686 C2BB16      JP NZ,16BB
D689 79          LD A,C
D68A 17          RLA
D68B 4F          LD C,A
D68C DB          RET C
D68D 79          LD A,C
D68E 2F          CPL
D68F D308        OUT (08),A
D691 CD9003      CALL 0390
D694 DB09        IN A,(09)
D696 2F          CPL
D697 E6FE        AND FE
D699 C8916       JP Z,1689
D69C 04          INC B
D69D 1F          RRA
D69E D29C16      JP NC,169C
D6A1 B7          OR A
D6A2 C2B416      JP NZ,16B4
D6A5 79          LD A,C
D6A6 1C          INC E
D6A7 17          RLA
D6A8 D2A616      JP NC,16A6
D6AB 7B          LD A,E
D6AC 3D          DEC A
D6AD 07          RLCA
D6AE 07          RLCA
D6AF 07          RLCA
D6B0 B0          OR B
D6B1 6F          LD L,A
D6B2 AF          XOR A
D6B3 C9          RET
  
```

LD684 D6BF

```

D684 97          SUB A
D685 37          SCF
D686 C9          RET
D687 FF          RST 38H
D688 FF          RST 38H
D689 FF          RST 38H
D68A FF          RST 38H
D68B 37          SCF
D68C 3F          CCF
D68D C9          RET
D68E FF          RST 38H
  
```

\* DD6C0 D6FF 'CODE TABLE FOR KEYSKAN'

```

D6C0 00 38 39 30 3A 3D 60 1B 00 31 32 33 34 35 36 37
D6D0 00 79 75 69 6F 70 7B 0D 00 61 73 71 77 65 72 74
D6E0 00 68 6A 6B 6C 3B 7D 0A 00 7A 7B 20 63 64 66 67
D6F0 00 6E 6D 2C 2E 2F 76 62 0A FF 0B 08 0C 0C FF FF
  
```

\* \*  
\* DVX  
\* LD700 D705

```

D700 79          LD A,C
D701 C620        ADD A,20
D703 4F          LD C,A
D704 C9          RET
  
```

\* LD708 D70F 'NEW CODE FOR SHF AND 0'

```

D708 3E7E        LD A,7E
D70A 4F          LD C,A
D70B C32C16     JP 162C
D70E FF          RST 38H
  
```

\* LD710 D717 'SUBR SUBTRACT 40 (H)'

```

D710 79          LD A,C
D711 D640        SUB 40
D713 4F          LD C,A
D714 C9          RET
D715 FF          RST 38H
D716 FF          RST 38H
  
```

\* LD718 D71E 'SUBTRACT 20(H)'

```

D718 79          LD A,C
D719 D620        SUB 20
D71B 4F          LD C,A
D71C C9          RET
D71D FF          RST 38H
  
```

!C=00,B=00-1  
!DE=0000  
!CARRY =1  
!00 IN ACC  
!CARRY IN LSB  
!STORE IN C  
!COMPLEMENT TO OUTPUT  
!FE CODE FOR SCAN  
!DELAY 0.5 MSEC  
!READ PORT FOR RET LINE  
!COMPLEMENT  
!SAVE IN D , TEST FOR  
!CURSOR KEYS,IF PRESSED  
!RETURN VIA SET CARRY  
!ELSE SCAN NEXT LINE  
!  
!ALL LINES OVER ?  
!THEN CARRY =1,RETURN  
!  
!OUTPUT MORE  
!NEXT SCAN LINES  
!DELAY  
!READ RETURN LINES  
!AND COMPLEMENT  
!KILL BIT 0 TO AVOID CAPS LOCK  
!NOKEY FOUND, REPEAT  
!NEXT ROW,COUNT B  
!TO GET COLUMN NO IN B  
!  
!MORE THAN ONE KEY AT A TIME ?  
!THEN ABORT,RET WITH CARRY=1  
!GET ROW NUMBER  
!IN E REG  
!  
!E TO ACC  
!  
!MULTIPLY BY B  
!  
!  
!  
!MULTIPLY . . . ROW AND COLUMN  
!CODE TABLE ENTRY  
!FLAGS RESET  
!

```

LD BC,FF00
LD DE,0000
SCF
LD A,C
RLA
LD C,A
CPL
OUT (08),A
CALL 0390
IN A,(09)
CPL
LD D,A
AND B
JP NZ,16BB
LD A,C
RLA
LD C,A
RET C
LD A,C
CPL
OUT (08),A
CALL 0390
IN A,(09)
CPL
AND FE
JP Z,1689
INC B
RRA
JP NC,169C
OR A
JP NZ,16B4
LD A,C
INC E
RLA
JP NC,16A6
LD A,E
DEC A
RLCA
RLCA
RLCA
OR B
LD L,A
XOR A
RET
  
```

'KILL CONTENTS OF ACC,NO KEY FOUND'  
'KILL CONTENTS OF ACC'  
'SET CARRY'  
'  
'RETURN FOR CURSOR CONTROL'  
'CARRY =0'  
'CODE TABLE FOR KEYSKAN'

APPENDIX - K

DC300 C5FF

C300 C5 CD 50 02 C1 77 23 78 B1 C2 00 03 C9 3E BA  
C310 D3 08 C9 DB 0A F6 01 D3 0A C9 FF 00 00 00 00  
C320 00 3A F1 20 E6 08 CA 2D 03 FB C3 31 03 37 D2 31  
C330 03 21 E9 20 F9 D1 C1 F1 2A F4 20 F9 2A F2 20 E5  
C340 2A EF 20 C9 FF FF FF FF FF FF FF FF FF FF FF  
C350 0E 06 21 9A 03 CD 90 C3 7E D3 01 23 0D C2 55 03  
C360 3E BA D3 08 A3 03 21 00 F0 22 8B 20 01 00 06  
C370 CD 9A 01 CD 72 01 C3 E0 03 FF FF FF FF FF FF  
C380 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
C390 D5 11 3F 00 CD F1 05 D1 C9 FF 00 00 40 CE 05  
C3A0 00 00 00 21 C0 03 06 0F 7E 32 00 F8 23 7E 32 01  
C3B0 F8 23 05 C2 AB 03 C9 FF FF FF FF FF FF FF FF  
C3C0 00 56 01 40 02 43 03 AB 04 26 05 00 06 18 07 20  
C3D0 08 00 09 07 0A 40 08 07 0C 30 0D 00 0E 30 0F 00  
C3E0 22 EF 20 E1 22 F2 20 F5 E1 22 ED 20 21 00 00 39  
C3F0 22 F4 20 21 ED 20 F9 C5 D5 00 21 8C 07 06 14 4E  
C400 CD C4 05 23 05 C2 FF 03 21 A0 20 F9 0E 2E CD F8  
C410 05 C3 14 04 CD 1F 06 CD F8 05 79 01 06 00 21 AE  
C420 07 BE CA 2D 04 23 0D C2 21 04 C3 C0 0F 21 A0 07  
C430 09 09 7E 23 66 4F E9 0E 02 CD 5B 06 D1 E1 CD EB  
C440 05 7C CD C7 06 7D CD C7 06 0E 20 CD F8 05 7E CD  
C450 C7 06 CD 0A 06 D2 5E 04 CD EB 05 C3 08 04 23 7D  
C460 E6 0F C2 49 04 C3 3E 04 CD 26 06 D2 7D 0A 7A FE  
C470 0D C2 11 06 21 F2 20 71 23 70 C3 83 04 7A FE 0D  
C480 C2 11 06 C3 1B 03 0E 01 CD 5B 06 3E FF 32 FD 20  
C490 D1 CD 1F 06 4F CD F8 05 79 FE 1B CA C7 04 CD 79  
C4A0 07 DA 91 04 CD 5E 07 D2 C1 04 CD BB 05 4F CD 3F  
C4B0 07 3A CD 20 B7 C2 B9 04 13 EE FF 32 FD 20 C3 91  
C4C0 04 CD 3A 07 C3 11 06 CD 3A 07 CD EB 05 C3 08 04  
C4D0 0E 03 CD 5B 06 C1 E1 D1 E5 62 6B 7E 60 69 77 03  
C4E0 7B B1 CA 08 04 13 E1 CD A0 06 D2 08 04 C3 DB 04  
C4F0 CD 26 06 C5 E1 7A FE 20 CA 00 05 FE 2C C2 08 04  
C500 7E CD C7 06 0E 2D CD F8 05 CD 26 06 D2 10 05 71  
C510 23 C3 F5 04 CD 1F 06 4F CD F8 05 79 FE 0D C2 27  
C520 05 CD EA 06 C3 08 04 4F CD 1B 07 C5 E1 0E 20 CD  
C530 F8 05 79 32 FD 20 3A FD 20 FE 20 CA 43 05 FE 2C  
C540 C2 08 04 7E B7 C2 4E 05 CD EB 05 C3 08 04 E5 5E  
C550 16 20 23 46 D5 D5 E1 C5 7E CD C7 06 F1 F5 B7 CA  
C560 67 05 2B 7E CD C7 06 0E 2D CD F8 05 CD 26 06 D2  
C570 87 05 7A 32 FD 20 F1 E1 B7 CA 7E 05 70 2B 71 11  
C580 03 00 E1 19 C3 36 05 7A 32 FD 20 D1 D1 C3 7F 05  
C590 CD 72 01 CD DB 05 DA 93 05 CD AB 05 F5 D5 11 FF  
C5A0 7F CD F1 05 D1 F1 C9 FF F5 F5 FE 10 CC 87 00 F1  
C5B0 FE 0F CA F9 03 F1 C9 FF FF FF FF 79 D6 30 FE 0A  
C5C0 FB D6 07 CA 09 79 E6 7F CD 00 01 CD 72 01 C9 FF  
C5D0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  
C5E0 16 C1 D1 E1 C9 FF FF FF FF FF FF FF FF FF FF  
C5F0 C9 1B 7A B3 C2 F1 05 C9 41 3E 1B B8 C2 01 06 0E

C000 C2FF

C000 31 A0 20 C3 50 03 FF FF C3 08 20 FF FF FF 3E 0D  
C010 C3 10 20 FF FF C3 18 20 FF FF FF FF FF FF FF  
C020 C3 20 20 FF C3 24 20 FF C3 28 20 FF C3 2C 20 FF  
C030 C3 30 20 FF C3 34 20 FF C3 38 20 FF C3 3C 20 FF  
C040 F5 DB 01 E6 04 CA 41 00 F1 D3 00 C9 FF FF FF FF  
C050 DB 01 E6 02 CA 6C 00 4F CD C4 05 C3 6C 00  
C060 FF FF FF FF FF C3 66 20 FF FF FF CD 72 01 CD  
C070 DB 05 DA 50 00 CD AB 05 CD 40 00 D5 11 FF 7F CD  
C080 F1 05 D1 C3 50 00 FF E5 D5 C3 00 00 21 00 F0  
C090 16 18 1E 40 0E 0A CD C4 00 0E 0D CD C4 00 4E CD  
C0A0 C4 00 23 1D C2 9E 00 15 C2 92 00 C1 D1 E1 C9 FF  
C0B0 CD 90 05 4F CD C4 05 C3 B0 00 FF FF FF FF FF FF  
C0C0 FF FF FF F3 C5 D5 CD F0 00 06 07 11 02 02 CD  
C0D0 EA 07 79 1F 4F DA F0 00 DC F8 00 05 F2 CC 00 CD  
C0E0 F8 00 11 04 03 CD EA 07 D1 C1 FB C9 FF FF FF FF  
C0F0 DB 0A F6 04 D3 0A C9 FF DR 0A E6 F8 D3 0A C9 FF  
C100 FE 20 D2 30 01 FE 08 CA F1 01 FE 0B CA EA 01 FE 0C  
C110 09 CA BB 01 FE 08 CA F1 01 FE 0B CA EA 01 FE 0C  
C120 CA D6 01 FE 1E CA CD 01 FE 1B CA FB 01 C9 FF FF  
C130 F5 E5 2A B8 20 77 23 7D E6 3F 00 00 CC 4B 01 22  
C140 B8 20 E1 F1 C9 C5 01 18 00 09 C1 7C FE F6 CC 52  
C150 01 C9 C5 D5 E5 11 00 F0 21 40 F0 01 C0 05 7E 12  
C160 13 23 0B 7B B1 C2 5E 01 CD 8E 01 E1 21 C0 F5 D1  
C170 C1 C9 F5 3E 0E 32 00 F8 3A B9 20 32 01 FB 3E 0F  
C180 32 00 F8 3A B8 20 32 01 F8 F1 C9 FF FF FF 21 C0  
C190 F5 01 40 00 3E 20 77 23 0B 7B B1 C2 94 01 C9 F5  
C1A0 3A B8 20 E6 C0 32 B8 20 F1 C9 F5 E5 D5 2A B8 20  
C1B0 11 40 00 19 CD 4B 01 C3 EA 01 FF E5 2A B8 20 7D  
C1C0 C6 08 E6 FB 6F 22 B8 20 E1 C9 FF FF FF E5 21 00  
C1D0 F0 22 B8 20 E1 C9 E5 2A B8 20 23 22 B8 20 E1 C9  
C1E0 F5 E5 2A B8 20 11 C0 FF 19 22 B8 20 D1 E1 F1  
C1F0 C9 E5 2A B8 20 2B 22 B8 20 E1 C9 0E 24 C3 C4 05  
C200 DB 0A F6 02 D3 0A D5 16 19 15 C2 09 02 D1 DB 0A  
C210 E6 3D D3 0A C9 CD 00 02 D5 16 6B 15 C2 1B 02 D1  
C220 C9 CD 02 D5 16 FB 15 C2 27 02 D1 C9 FF FF FF  
C230 79 17 4F DA 39 02 C3 21 02 CD 15 02 CD 15 02 C9  
C240 06 08 CD 30 02 05 C2 42 02 11 A4 00 CD F1 05 C9  
C250 06 08 0E 00 16 00 14 DB 0A E6 10 C2 56 CD 14 DB  
C260 0A E6 10 CA 5E 02 05 CA 79 02 7A FE 53 CD 8B 02  
C270 D2 5A 02 CD 90 02 C3 54 02 7A FE A6 CD 8B 02 D2  
C280 85 02 CD 90 02 79 C9 FF F5 79 17 4F F1 C9 FF FF  
C290 DB 0A E6 10 C2 90 02 DB 0A E6 10 CA 97 02 C9 FF  
C2A0 16 80 4B D5 CD 04 02 D1 15 C2 A2 02 C9 C5 4E CD  
C2B0 40 02 23 C1 0B 7B B1 C2 AD 02 C9 C5 1E 00 CD A0  
C2C0 02 1E FF CD A0 02 C1 C9 D5 11 10 00 D5 11 FF FF  
C2D0 CD F1 05 D1 1B 7A B3 C2 CC 02 D1 C9 D5 16 20 D5  
C2E0 CD 50 02 D1 FE FF C2 DD 02 15 C2 DF 02 CD 50 02  
C2F0 FE FF CA ED 02 D1 C9 FF DB 0A E6 FE D3 0A C9 FF



IC900 CBFF

C900 7E FE 40 DA 26 09 FE DD CA 80 09 FE ED CA 3A 09 FE  
C910 FE CB CA 32 09 FE DD CA 80 09 FE ED CA 3A 09 FE  
C920 FD CA BA 09 D6 80 21 FF OF 4F A7 C4 D0 08 CD A0  
C930 0A C9 3E 00 32 B3 20 C3 B7 09 23 7E 22 BE 20 21  
C940 25 15 06 C4 D0 08 CD A0 0A 0A 0A 0A 0A 0A 0A  
C950 00 00 06 C4 D0 08 CD A0 0A 0A 0A 0A 0A 0A  
C960 04 23 0C 05 CA 6D 09 BE C2 61 09 37 C9 37 C9  
C970 21 98 0E 85 6F 7E FE 24 CC 23 0A 4F CD F8 05 C2  
C980 23 7E E5 21 A0 0B 06 29 CD 5C 09 D2 F4 08 E1 22  
C990 BE 20 7E FE CB C2 AF 0B E5 F5 23 23 00 7E FE 36  
C9A0 CA F3 08 E6 07 FE 06 C2 F3 08 F1 E1 23 7E 32 BA  
C9B0 20 3E FF 32 B5 20 00 23 7E FE 40 DA F6 09 E5 F5  
C9C0 E6 60 IF 1F 1F 1F 1F 1F 1F 1F 1F 1F 1F 1F C5  
C9D0 CD 90 0A 0E 20 CD F8 05 F1 E6 07 CD 70 09 E1 23 22  
C9E0 30 4F CD 0A 0E 20 CD F8 05 F1 E6 07 CD 70 09 E1 23 22  
CA00 08 F5 E6 38 1F 1F 1F 21 5F 0A 4F A7 C4 D1 08 CD  
CA10 90 0A 0E 20 CD F8 05 F1 E6 07 CD 70 09 E1 23 22  
CA20 BE 20 C9 0E 28 CD F8 05 3A B3 20 21 50 0A A7 4F  
CA30 C4 D0 08 CD 90 0A 3A B3 20 87 CA 48 0A 0E 28 CD  
CA40 F8 05 3A B4 20 CD C7 06 3E 29 C9 FF 4C 44 20 0D  
CA50 48 4B 4C 0D 49 5B 0D 49 5B 0D 49 5B 0D 49 5B 0D  
CA60 52 4C 43 0D 52 52 43 0D 52 43 0D 52 43 0D 52 43 0D  
CA70 53 4C 41 0D 53 52 41 0D 45 52 41 0D 53 52 4C 0D  
CA80 42 49 54 0D 52 45 53 0D 53 45 0D 45 52 4C 0D  
CA90 23 7E FE 0D CB 4F CD FB 05 C3 90 0A FF FF FF FF  
CAA0 23 7E FE 0D C2 AF 0A 2A BE 20 23 22 BE 20 C9 FE  
CAB0 24 C2 DE 0A 3A B3 20 00 E5 21 50 0A A7 4F C4 D0  
CAC0 08 CD 90 0A E1 00 00 00 00 00 00 00 00 00 00 0A  
CAD0 0E 28 CD F8 05 3A B4 20 CD C7 06 C3 A0 0A FE 3F  
CAE0 C2 0C 0R 23 7E E5 2A BE 20 23 FE C2 01 0B 23  
CAF0 7E CD C7 06 22 BE 20 E1 C3 A1 0A 4F CD F8 05  
CR00 0A 7E CD C7 06 22 BE 20 E1 C3 A1 0A 4F CD F8 05  
CB10 C3 0A 0A F5 FE 76 CA 41 0B 21 4B 0A CD 90 0A F1  
CB20 F5 E6 38 1F 1F 1F CD 70 09 0E 20 CD F8 05 0E 2C  
CB30 CD F8 05 F1 E6 07 CD 70 09 2A BE 20 23 22 BE 20  
CB40 C9 CD EB 08 F1 C3 F8 08 F5 E6 38 1F 1F 4F 21  
CB50 70 0B A7 C4 D0 08 CD 90 0A 2B 7E FE 41 C2 65 0B  
CB60 0E 2C CD F8 05 C3 33 0B FF FF FF FF FF FF FF FF  
CB70 41 44 44 20 41 2C 0D 41 44 43 20 41 2C 0D 41 44  
CB80 42 20 0D 53 42 43 20 41 2C 0D 41 44 43 20 41 2C 0D 41 44  
CB90 4F 52 20 0D 4F 52 20 0D 43 50 20 0D FF FF FF FF  
CBA0 09 19 21 22 23 29 2A 2B 34 35 36 39 46 4E 56 5E  
CBB0 66 66 71 72 73 74 75 76 77 78 79 86 86 96 9E A6  
CBC0 AE B6 BE CB E1 E3 E5 E9 F9 FF FF FF FF FF FF FF  
CBD0 00 00 02 02 00 01 01 01 01 01 01 01 01 01 01 01  
CBF0 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01

IC600 CBFF

C600 24 CD C4 05 3E 0D B8 C2 0F 06 0E 0A CD C4 05 4B  
C610 C9 0E 2A CD F8 05 CD EB 05 C3 08 04 37 3F C9 CD  
C620 90 05 E6 7F 4F C9 E5 21 00 00 1E 00 CD 1F 06 4F  
C630 CD F8 05 CD 79 07 D2 45 06 51 E5 C1 7B B7 C2  
C640 32 07 CA 1C 06 CD 5E 07 D2 11 06 CD 8R 05 1E FF  
C650 29 29 29 06 00 4F 09 C3 2C 06 2E 03 79 E6 03  
C660 C8 67 CD 26 06 D2 11 06 C5 2D 25 CA 77 06 7A FE  
C670 0D CA 11 06 C3 62 06 7A FE 0D C2 11 06 01 FF FF  
C680 7D B7 CA 8A 06 C5 2D C2 85 06 C1 D1 E1 CD A0 06  
C690 D2 95 06 54 5D E3 D5 E5 3D F8 E1 E3 C3 99 06  
C6A0 C5 47 E5 7A R3 CA C1 06 23 7C R5 CA C1 06 E1 D5  
C6B0 3E FF AA 57 3E FF AB 5F 13 7D 83 7C 8A D1 78 C1  
C6C0 C9 E1 78 C1 C3 32 07 E5 F5 0F 0F 0F 0F 0F 0F 0F 0F  
C6D0 CD E2 06 CD F8 05 F1 E6 0F 4F CD C2 06 CD F8 05  
C6E0 E1 C9 21 B4 07 06 00 09 4E C9 21 C4 07 4E 79 FE  
C6F0 C2 F7 06 CD EB 05 C9 CD F8 05 0E 3D CD F8 05 23  
C700 5E 16 20 23 1A CD C7 06 7E B7 CA 12 07 1B 1A CD  
C710 C7 06 0E 20 CD F8 05 23 C3 ED 06 21 C4 07 11 03  
C720 00 7E B7 CA 11 06 B9 CA 2E 07 19 C3 21 07 23 44  
C730 4D C9 37 C9 3A FD 20 B7 C0 0E 00 CD 3F 07 C9 D5  
C740 E1 79 E6 0F 4F 3A FD 20 B7 C2 52 07 7E E6 F0 B1  
C750 77 C9 7E E6 0F 4F 79 0F 0F 0F 0F 0F 0F 0F 0F 0F  
C760 30 FA 1C 06 FE 39 FA 32 07 CA 32 07 FE 41 FA 1C  
C770 06 FE 47 F2 1C 06 C3 32 07 79 FE 2C CA 32 07 FE  
C780 0D C4 32 07 FE 20 CA 32 07 C3 1C 06 0D 0A 5A 2D  
C790 38 30 20 52 4E 4D 20 41 50 52 20 38 36 20 0D 0A  
C7A0 00 00 14 05 F0 04 D0 04 86 04 68 04 37 04 44 47  
C7B0 49 4D 53 58 30 31 32 33 34 35 36 37 38 39 41 42  
C7C0 43 44 45 46 41 EE 00 42 EC 00 43 EB 00 44 EA 00  
C7D0 45 E9 00 46 ED 00 49 F1 00 48 F0 00 4C EF 00 4D  
C7E0 F0 01 53 F5 01 50 F3 01 00 00 E5 2A B1 20 29 1D  
C7F0 C2 EE 07 EB E1 CD F1 05 C9 FF C3 C4 05 C3 90 05  
C800 7E F5 CD C7 06 F1 E5 FE 40 DA 1A 0A FE C0 D2 18  
C810 08 06 07 CD A0 08 E1 C9 D6 80 21 60 15 00 85 6F  
C820 7E F5 E6 F0 1F 1F 1F 47 F1 E6 07 C4 50 08 48  
C830 78 A7 CA 42 0B E1 E5 23 7E C5 CD C7 06 C1 05 C2  
C840 37 08 3E 03 A9 87 3C 47 CD A0 08 E1 C9 06 00 C9  
C850 32 R3 20 FE 04 C2 5B 08 06 01 C9 2A BE 20 FE 03  
C860 C2 79 08 23 7E 21 E0 15 06 06 2B 04 23 05 CA 77  
C870 08 BE C2 6C 08 06 02 04 C9 23 23 7E 32 B4 20 2R  
C880 7E 00 21 A0 08 06 29 CD 5C 09 D2 4D 08 79 21 D0  
C890 0R 85 6F 46 04 C9 FF FF 42 43 44 45 46 4C 24 41  
C8A0 0E 20 C5 CD F8 05 C1 05 C2 A0 08 C9 FF F1 E1 FE  
C8B0 E1 D2 01 09 FE 39 CA 01 09 FE 34 DA 01 09 F5 23  
C8C0 7E 32 B4 20 22 BE 20 3E FF 32 B5 20 F1 C3 01 09  
C8D0 2B 23 7E FE 0D C2 D1 08 0D C2 D1 08 C9 E5 F5 21  
C8E0 8R 0A CD 90 0A F1 E1 C9 E5 F5 21 5A 0A CD 90 0A  
C8F0 F1 E1 C9 F1 E1 CD DD 08 2A BE 20 23 22 BE 20 C9

\* DE00 CFFF  
.DGE00 CFFF

CE00 3E 80 D3 87 3E 80 D3 86 CD 27 0E 23 13 0B 78 D3 84 7B D3 85 7A E6  
 CE10 1F F6 80 D3 86 E6 1F D3 86 C9 F5 F6 20 D3 86 D5 11 FF 23  
 CE20 DB 86 E6 1F D3 86 C9 F5 F6 20 D3 86 D5 11 FF 23  
 CE30 CD F1 05 D1 F1 D3 86 C9 F5 F6 20 D3 86 D5 11 FF 23  
 CE40 3E 90 D3 87 7B D3 85 7A E6 1F CD 38 0E D2 D1 C9  
 CE50 8A 77 23 13 0B 78 B1 C2 44 0E C9 FF FF FF FF  
 CE60 31 A0 20 3E 60 32 BB 20 CD 40 OF CD 00 0E CF FF  
 CE70 31 A0 20 3E 40 C3 65 0E 31 A0 20 3E 40 C3 75 0E  
 CE80 31 A0 20 3E 40 C3 65 0E 20 CD FA 07 7E CD C7 06 C1  
 CE90 C5 1E 10 CD ER 05 0E 20 CD FA 07 7E CD C7 06 C1  
 CEA0 CD 8B 0E C8 C5 23 1D 7B FE 0B CA 96 0E 7B A7 CA  
 CEB0 91 0E C3 9B 0E FF FF FF FF FF FF FF FF FF FF  
 CEC0 C5 1E 10 0E 7F CD FA 07 1D C2 C3 0E C1 C9 FF FF  
 CED0 CD E1 0E 07 07 07 07 07 07 07 07 07 07 07 07 07 07  
 CEE0 0E CD 1F 06 CD 5E 07 D2 E1 0E CD BB 05 C9 0E 02  
 CEF0 CD 5R 06 D1 E1 22 BE 20 EB 22 C0 20 C3 34 0C FF  
 CF00 3E 80 D3 83 CD 7A 0F F6 02 E6 72 F5 D3 82 7B D3  
 CF10 81 7E D3 80 F1 CD 21 0F 23 0B 78 B1 C8 13 C3 04  
 CF20 0F F5 F6 01 D3 B2 D5 11 FF 23 CD F1 05 D1 F1 D3  
 CF30 82 C9 FF FF FF FF FF FF FF FF FF FF FF FF FF  
 CF40 2A BE 20 5E ER 2A C0 20 CD 51 0F 2A BC 20 EB E1  
 CF50 C9 7B 2F 5F 7A 2F 13 19 E5 C5 E1 C1 C9 FF FF  
 CF60 3E 90 D3 83 CD 7A 0F E6 70 D3 82 7B D3 81 DB 80  
 CF70 77 23 13 0B 78 B1 C2 64 0F C9 7A 17 17 C9  
 CF80 31 A0 20 CD 40 OF CD 00 0E CF 31 A0 20 CD 40 OF  
 CF90 CD 00 0E CF 31 A0 20 CD 40 OF CD 60 0F CF 31 A0  
 CFA0 20 CD 40 OF CD 0E CD 90 0E CD 0E CD 31 A0  
 CFB0 20 CD 40 OF CD 0E CD 0E CF FF FF FF FF FF FF  
 CFC0 01 07 00 21 F8 0F RE CA D2 0F 23 0D C2 C3 84 0F  
 CFD0 11 06 21 EB 0F 09 7E 23 66 6F E9 01 02 0A 06  
 CFE0 0E 02 04 0A 0C 06 0E FF 00 00 A9 0C CA 0D BA 0F  
 CFF0 A0 0D EE 0E 97 0C C2 0C 43 46 4C 50 52 57 42 FF

DCC00 CFFF

CC00 3E 00 32 B3 20 32 B5 20 CD EB 05 7C CD C7 06 7D  
 CC10 CD C7 06 0E CD 00 08 0E 20 CD A0 06 D2 FD 08  
 CC20 CD 00 09 2A C0 2D EB 2A BE 20 CD A0 06 D2 FD 08  
 CC30 C9 2A RE 20 E5 CD FD 0B E1 22 RE 20 CD EB 05 C3  
 CC40 08 04 CD 1F 06 CD F8 05 CD BR 05 E6 OF C9 FF FF  
 CC50 CD 42 0C 17 17 17 47 C5 CD 42 0C C1 80 C9 FF  
 CC60 C5 CD ER 05 7A CD C7 06 C1 C9 0E 02 CD 5R 06 D1 E1 E5 EB  
 CC70 05 1A CD C7 06 C1 C9 0E 02 CD 5R 06 D1 E1 E5 EB  
 CC80 CD 51 0F E1 C9 FF CD 77 0C C5 CD 50 0C 47 CD 1F  
 CC90 06 79 FE 0D 7B C1 C9 CD 86 0C C2 11 06 57 72 23  
 CCA0 0R 7B B1 C2 9E 0C C3 08 04 CD 86 0C C2 11 06 57  
 CCB0 BE CC F0 0C CA 08 04 23 0B 78 B1 CA 08 04 7A C3  
 CCC0 B0 0C 0E 03 CD 5R 06 C1 D1 E1 C5 E5 EB CD 51 0F  
 CCD0 E1 D1 1A BE C4 F0 0C C2 08 04 23 13 0B 78 B1 C2  
 CCE0 D2 0C C3 08 04 FF FF FF FF FE 20 C1 ER E1 C9  
 CCF0 E7 ER C5 CD 60 0C CD 1F 06 79 FE 20 C1 ER E1 C9  
 CD00 CD 70 0D FE 40 DA OF 0D FE C0 DA 2B 0D D6 80 E5  
 CD10 21 60 15 85 6F 7E E1 F5 E6 0F C2 3A 0D F1 E6 F0  
 CD20 CA 2B 0D FE 20 CA 57 0D CD 70 0D D5 E5 2A C0 20  
 CD30 ER E1 CD A0 06 D1 D2 00 0D C9 FE 04 C2 43 0D F1  
 CD40 C3 2B 0D FE 03 C2 5D 0D F1 CD 70 0D C5 E5 CD 65  
 CD50 0B E1 05 C1 CA 2B 0D CD 7D 0D C3 2B 0D F1 CD 70  
 CD60 0D FE 21 CA 57 0D FE 22 CA 57 0D FE 2A C3 D0 0D  
 CD70 7E 12 13 23 C9 E1 C1 CD 7D 0D C3 37 0D 7E 23 E5  
 CD80 66 6F D5 ER CD F0 0D DA 9A 0D ER D1 09 7D 12 13  
 CD90 7C 12 13 E1 23 C9 F1 C3 37 0D ER D1 C3 8D 0D FF  
 CDA0 0E 03 CD 5R 06 0E 03 CD 5B 06 D1 E1 22 BC 20 E1  
 CDB0 22 BA 20 ER CD 51 0F D1 E1 22 C0 20 E1 22 BE 20  
 CDC0 CD 00 0D C3 08 04 FF FF FF CD 77 0C C3 A4 0F  
 CDD0 CA 57 0D C5 E5 CD 82 08 E1 05 CA E5 0D 05 C1 C4  
 CDE0 70 0D C3 2B 0D C1 C3 2B 0D FF FF FF FF FF FF  
 CDF0 2A BA 20 CD A0 06 DB 2A BC 20 CD A0 06 3F C9 FF

D000 D2FF

D000 4E 4F 50 0D 4C 44 20 42 43 2C 3F 3F 0D 4C 44 20  
D010 2B 42 43 29 2C 41 0D 49 4E 43 20 42 0D 4C 44 20 42 2C 3F  
D020 43 20 42 0D 44 45 43 20 42 0D 4C 44 20 42 2C 3F  
D030 0D 52 4C 43 41 0D 45 5B 20 41 46 2C 41 46 27 0D  
D040 41 44 44 20 24 2C 42 43 0D 4C 44 41 20 2C 2B 42  
D050 43 29 0D 44 45 43 20 42 43 0D 49 4E 43 20 43 0D  
D060 44 45 43 0D 4C 44 20 43 2C 3F 0D 4C 44 20 42 52 43  
D070 41 0D 44 4A 4A 5A 20 3F 0D 4C 44 20 44 45 2C 3F  
D080 3F 0D 4C 44 20 2B 44 45 29 2C 41 0D 49 4E 43 20  
D090 44 45 0D 4E 43 20 44 0D 44 45 43 20 44 0D 4C  
D0A0 44 20 44 2C 3F 0D 52 4C 41 0D 4A 52 20 3F 0D 41  
D0B0 44 44 20 24 20 2C 44 45 0D 4C 44 20 41 2C 2B 44  
D0C0 45 29 0D 44 45 43 20 44 45 0D 49 4E 43 20 45 0D  
D0D0 44 45 43 20 45 0D 4C 44 20 45 2C 3F 0D 52 43 41  
D0E0 0D 4A 52 20 4E 5A 2C 3F 0D 4C 44 20 24 2C 3F 3F  
D0F0 0D 4C 44 20 2B 3F 3F 29 2C 24 0D 49 4E 43 20 24  
D100 0D 49 4E 43 20 48 0D 44 45 43 20 48 0D 4C 44 20  
D110 48 2C 3F 0D 44 41 0D 4A 52 20 5A 2C 3F 0D 41  
D120 44 44 20 24 2C 24 0D 4C 44 20 24 2C 2B 3F 3F 29  
D130 0D 44 45 43 20 24 20 0D 49 4E 43 20 4C 0D 44 45  
D140 43 20 4E 43 2C 3F 0D 4C 44 20 4C 2C 3F 0D 43 50 4C 0D 4A  
D150 52 20 4E 43 2C 3F 0D 4C 44 20 53 50 2C 3F 3F 0D  
D160 4C 44 20 2B 3F 3F 29 2C 41 0D 49 4E 43 20 53 50  
D170 0D 49 4E 43 20 2B 24 29 0D 44 45 43 20 2B 24 29  
D180 0D 4C 44 20 2B 24 29 2C 3F 0D 53 43 46 0D 4A 52  
D190 20 43 2C 3F 0D 41 44 44 20 24 2C 53 50 0D 4C 44  
D1A0 20 41 2C 2B 3F 3F 29 0D 44 45 43 20 53 50 0D 49  
D1B0 4E 43 20 41 0D 44 45 43 20 41 0D 4C 44 20 41 2C  
D1C0 3F 0D 43 43 46 0D 52 45 54 20 4E 5A 0D 50 4F 50  
D1D0 20 42 43 0D 4A 50 20 4E 5A 2C 3F 3F 0D 4A 50 20  
D1E0 3F 3F 0D 43 41 4C 4C 20 4E 5A 2C 3F 3F 0D 50 55  
D1F0 53 4B 20 42 43 0D 41 44 44 20 41 2C 3F 0D 52 53  
D200 54 20 4F 0D 52 45 54 20 5A 0D 52 45 54 0D 4A 50  
D210 20 5A 2C 3F 3F 0D 40 0D 43 41 4C 4C 20 5A 2C 3F  
D220 3F 0D 43 41 4C 4C 20 3F 3F 0D 41 44 43 41 2C  
D230 0F 0D 52 53 54 20 3B 0D 52 45 54 20 4E 43 0D 50  
D240 4F 50 20 44 45 0D 4A 50 20 4E 43 2C 3F 3F 0D 4F  
D250 55 54 20 2B 3F 29 2C 41 0D 43 41 4C 4C 20 4E 43  
D260 2C 3F 3F 0D 50 55 53 4B 20 44 45 0D 53 55 42 20  
D270 3F 0D 52 53 54 20 31 30 4B 0D 52 45 54 20 43 0D  
D280 45 5B 5B 0D 4A 50 20 43 2C 3F 3F 0D 49 4E 20 41  
D290 2C 2B 3F 29 0D 43 41 4C 4C 20 3F 0D 43 2C 3F 0D 40  
D2A0 0D 53 42 43 20 41 2C 3F 0D 52 53 54 20 31 3B 4B  
D2B0 0D 52 45 54 20 50 4F 0D 50 4F 50 20 24 0D 4A 50  
D2C0 20 50 4F 2C 3F 3F 0D 45 5B 20 2B 53 50 29 2C 24  
D2D0 0D 43 41 4C 4C 20 50 4F 2C 3F 0D 50 55 53 4B  
D2E0 20 24 0D 41 4E 44 20 3F 0D 52 53 54 20 32 30 4B  
D2F0 0D 52 45 54 20 50 45 0D 4A 50 20 2B 24 29 0D 4A

D030 D5FF

D300 50 20 50 45 2C 3F 3F 0D 45 5B 20 44 45 2C 24 0D  
D310 43 41 4C 4C 20 50 45 2C 3F 3F 0D 40 5B 4F 52  
D320 20 3F 0D 52 53 54 20 32 3B 4B 0D 52 45 54 20 50  
D330 0D 50 4F 50 41 46 0D 4A 50 20 50 2C 3F 3F 0D  
D340 4A 49 0D 43 41 4C 4C 20 50 2C 3F 3F 0D 50 55 53  
D350 4B 20 41 46 0D 4F 52 20 3F 0D 4C 44 20 53 50 2C 24 0D  
D360 4B 0D 52 45 54 20 4D 0D 4C 44 20 53 50 2C 24 0D  
D370 4A 50 20 4D 2C 3F 3F 0D 45 49 0D 43 41 4C 4C 20  
D380 43 38 4B 0D 4F 49 4E 20 42 2C 2B 43 29 0D 4F 55  
D390 54 20 2B 43 29 2C 42 0D 53 42 43 20 4B 4C 2C 42  
D3A0 43 0D 4C 44 20 2B 3F 3F 29 2C 42 43 0D 4E 45 47  
D3B0 0D 52 45 54 4E 0D 49 4D 20 4F 0D 4C 44 20 49 2C  
D3C0 0D 52 45 54 4E 0D 49 4D 20 4F 0D 4C 44 20 49 2C  
D3D0 41 0D 49 4E 20 43 2C 2B 43 29 0D 4F 55 54 20 2B  
D3E0 43 29 2C 43 0D 41 44 43 20 4B 4C 2C 42 43 0D 4C  
D3F0 44 20 42 43 2C 2B 3F 29 0D 52 45 54 49 0D 49  
D400 4E 20 44 2C 2B 43 29 0D 4F 55 54 20 28 43 29 2C  
D410 44 0D 53 42 43 20 4B 4C 2C 44 45 0D 4C 44 20 2B  
D420 3F 29 2C 44 45 0D 49 4D 20 31 0D 4C 44 20 41  
D430 2C 49 0D 49 4E 20 45 2C 2B 43 29 0D 4F 55 54 20  
D440 2B 43 29 2C 45 0D 41 44 43 20 4B 4C 2C 44 45 0D  
D450 49 4E 20 48 2C 2B 43 29 0D 4F 55 54 20 28 43 29  
D460 2C 48 0D 53 42 43 20 4B 4C 2C 44 45 0D 4C 44 20  
D470 2C 48 0D 53 42 43 20 4B 4C 2C 44 45 0D 4C 44 20  
D480 29 2C 4C 0D 41 44 43 20 4B 4C 2C 44 45 0D 4C 44  
D490 2D 4C 4C 44 49 52 0D 43 50 49 4E 49 4E 49 4E 49  
D4A0 44 0D 53 42 43 20 4B 4C 2C 53 50 0D 4C 44 20 2B  
D4B0 3F 3F 29 2C 53 50 0D 49 4E 20 41 2C 28 43 29 0D  
D4C0 4F 55 54 20 2B 43 29 2C 41 0D 41 44 43 20 4B 4C  
D4D0 2C 53 50 0D 4C 44 20 53 50 2C 2B 3F 3F 29 0D 4C  
D4E0 44 49 0D 43 50 49 0D 49 4E 49 4E 49 4E 49 4E 49  
D4F0 0D 4C 44 49 52 0D 43 50 49 52 0D 49 4E 49 52 0D  
D500 4F 54 49 52 0D 43 50 44 52 0D 49 4E 44 52 0D 4F  
D510 54 44 52 0D 43 50 44 52 0D 49 4E 44 52 0D 4F  
D520 4B 4D 50 51 52 53 56 57 58 59 5A 5B 5E 60 61 62  
D530 67 68 69 6A 6F 72 73 7B 79 7A 7E 80 A1 A2 A3 A8  
D540 A9 AA AB B0 B1 B2 B3 B8 B9 BA BB FF FF FF FF FF  
D550 00 20 00 00 00 10 00 00 00 00 00 00 00 00 10 00  
D560 10 20 20 00 00 00 10 00 00 00 00 00 00 00 10 00  
D570 10 20 20 00 00 00 10 00 00 00 00 00 00 00 10 00  
D580 10 20 20 00 00 00 10 00 00 00 00 00 00 00 10 00  
D590 10 20 20 00 00 00 10 00 00 00 00 00 00 00 10 00  
D5A0 00 00 20 20 00 00 10 00 00 00 00 00 00 00 10 00  
D5B0 00 00 20 10 20 00 10 00 00 00 00 00 00 00 10 00  
D5C0 00 00 20 00 20 00 10 00 00 00 00 00 00 00 10 00  
D5D0 00 00 20 00 20 00 10 00 00 00 00 00 00 00 10 00  
D5E0 43 4B 53 5B 73 7B 7F FF FF FF FF FF FF FF FF FF FF  
D5F0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

DD600-D71F

D600 00 09 00 CD 70 16 DA 2D 16 7A E6 B8 CA 16 16 2E  
D610 37 2C 17 D2 11 16 26 16 7D F6 C0 6F 7E 4F FE 40  
D620 D2 51 16 7A E6 02 CA 31 16 79 97 4F 79 00 00 00  
D630 C9 79 FE 21 DA 4E 16 7A 00 E6 44 CA 2C 16 79 FE  
D640 30 CA 0B 17 DC 00 17 79 D6 10 4F C3 2C 16 3F C9  
D650 00 FE 7B DA 5F 16 7A E6 44 CA 18 17 C3 2C 16 7A  
D660 E6 47 CA 2C 16 E6 02 C4 10 17 CD 18 17 C3 2C 16  
D670 01 00 FF 11 00 00 37 79 17 4F 2F D3 08 CD 90 03  
D680 DB 09 2F 57 E6 B8 C2 BB 16 79 17 4F DB 79 2F D3  
D690 08 CD 90 03 DB 09 2F E6 FE CA B9 16 04 1F D2 9C  
D6A0 16 B7 C2 B4 16 79 1C 17 D2 A6 16 7B 3D 07 07 0F  
D6B0 30 6F AF C9 97 37 C9 FF FF FF FF 37 3F C9 FF FF  
D6C0 00 38 39 30 3A 3D 60 1B 00 31 32 33 34 35 36 37  
D6D0 00 79 75 69 6F 70 7B 0D 00 61 73 71 77 65 72 74  
D6E0 00 68 6A 6B 6C 3B 7D 0A 00 7A 7B 20 63 64 66 67  
D6F0 00 6E 6D 2C 2E 2F 76 62 0A FF 0B 08 0C FF FF  
D700 79 C6 20 4F C9 FF FF FF FF 3E 7E 4F C3 2C 16 FF FF  
D710 79 D6 40 4F C9 FF FF FF FF 79 D6 20 4F C9 FF FF FF

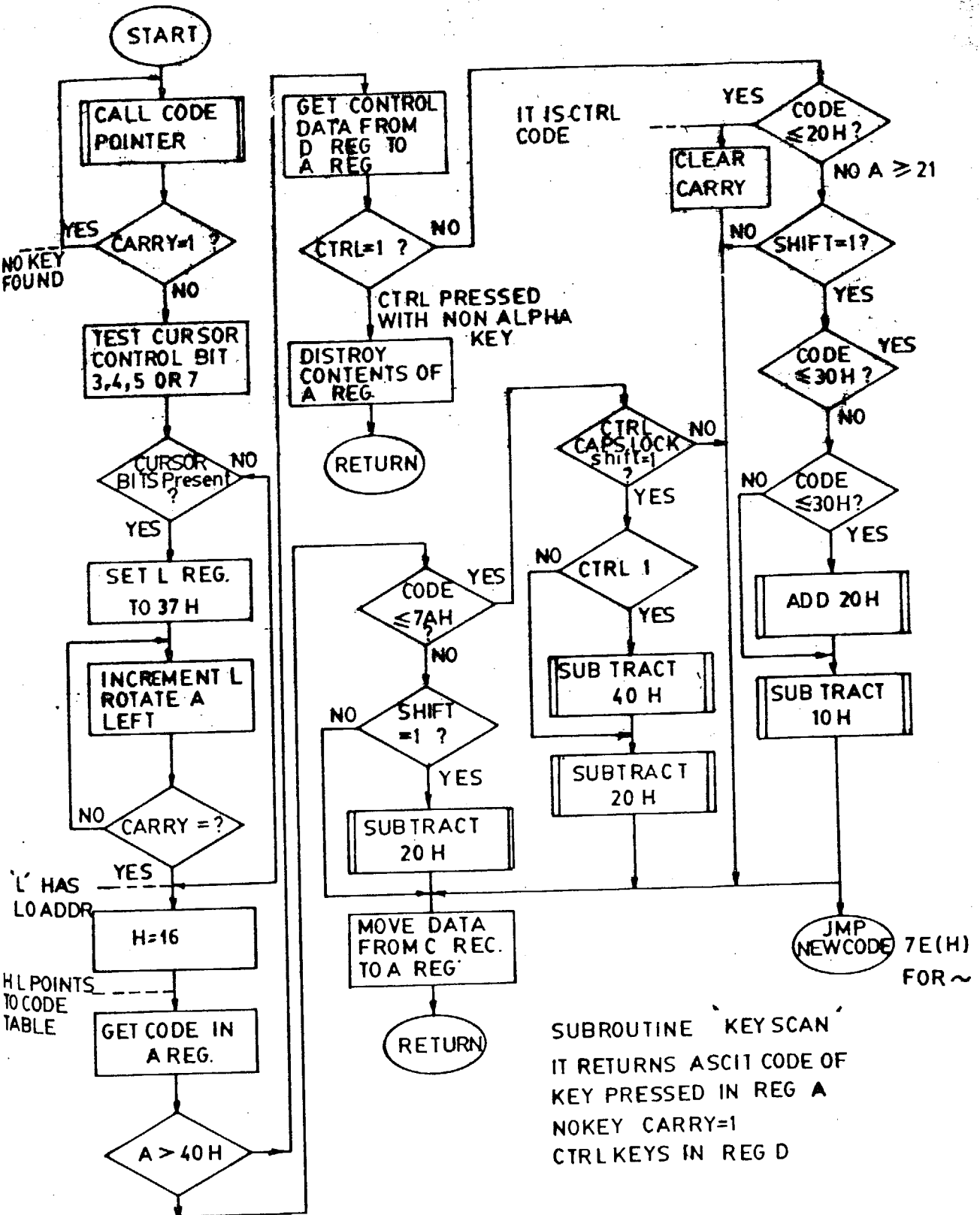
D3100 33FF

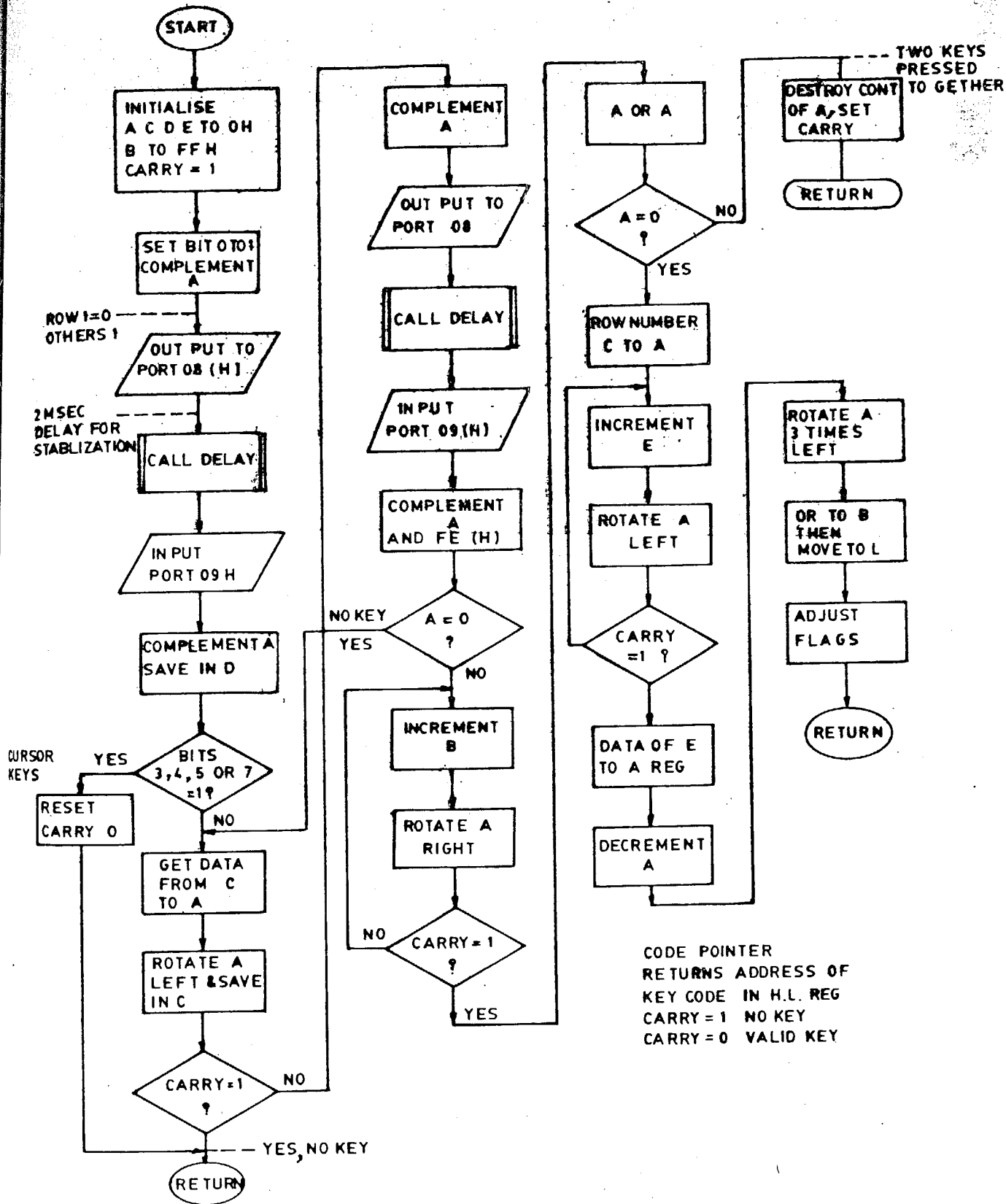
'CONTENTS OF CHARACTER GENERATOR'

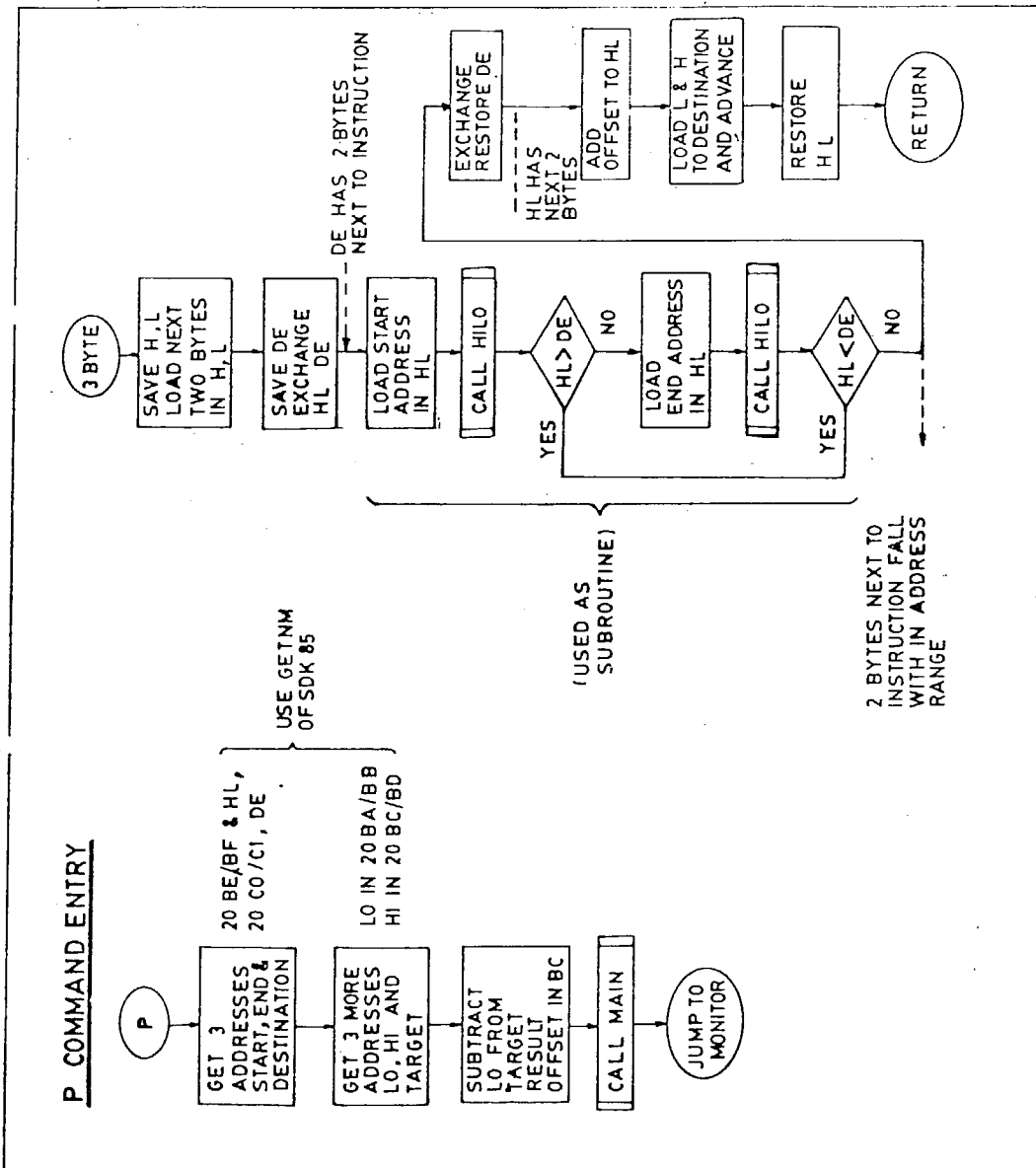
3100 00 00 00 00 00 00 00 00 00 00 00 00 10 10 10 00 00 10 00  
3110 2B 2B 00 00 00 00 00 00 00 00 28 28 FF 28 FF 28 2B 00  
3120 10 1C 30 18 14 38 10 00 00 00 24 08 10 20 42 00 00  
3130 10 2B 2B 30 54 4B 34 10 10 10 00 00 00 00 00 00  
3140 08 10 20 20 20 10 08 00 20 10 08 08 08 10 20 00  
3150 10 54 38 10 38 54 10 00 00 10 10 7C 10 10 00 00  
3160 00 00 00 00 00 10 10 20 00 00 00 3C 00 00 00  
3170 00 00 00 00 00 30 30 00 00 04 08 10 20 40 00  
3180 38 44 4C 54 64 44 38 00 10 30 10 10 10 10 38 00  
3190 3B 44 04 38 40 40 7C 00 7C 04 08 18 04 44 38 00  
31A0 08 18 2B 48 7C 08 08 01 7C 40 78 04 04 44 38 00  
31B0 1C 20 40 78 44 44 38 00 7C 04 08 10 20 20 00  
31C0 3B 44 44 38 44 44 38 00 38 44 44 3C 04 08 70 00  
31D0 00 00 10 00 00 10 00 00 00 00 00 00 10 10 20  
31E0 08 10 20 40 20 10 08 00 00 00 7C 00 00 00 00  
31F0 20 10 08 04 08 10 20 00 38 44 04 08 10 00 10 00  
3200 3B 44 5C 54 5C 40 38 00 10 28 44 44 7C 44 44 00  
3210 78 24 24 38 24 24 78 00 38 44 40 40 40 44 38 00  
3220 78 24 24 24 24 24 24 78 00 7C 40 40 70 40 40 7C 00  
3230 7C 40 40 70 40 40 40 40 40 40 3C 40 40 5C 44 3C 00  
3240 44 44 44 7C 44 44 00 38 10 10 10 10 10 38 00  
3250 0E 04 04 04 04 44 38 00 44 48 50 60 50 48 44 00  
3260 40 40 40 40 40 40 40 7C 00 44 6C 54 44 44 44 00  
3270 44 64 54 4C 44 44 44 00 38 44 44 44 44 44 38 00  
3280 78 44 44 78 40 40 40 40 3C 40 40 44 54 48 34 00  
3290 78 44 44 78 50 48 44 40 00 3C 40 40 38 04 04 78 00  
32A0 7C 10 10 10 10 10 10 10 10 44 44 44 44 44 44 38 00  
32B0 44 44 44 44 44 28 10 00 44 44 44 44 54 6C 44 00  
32C0 44 44 28 10 28 44 44 00 44 44 44 28 10 10 10 00  
32D0 7C 04 08 10 20 40 7C 00 38 20 20 20 20 38 00  
32E0 00 40 20 10 08 04 00 00 1C 04 04 04 04 04 1C 00  
32F0 10 28 44 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
3300 20 10 08 00 00 00 00 00 00 00 00 3C 04 3C 44 3C 00  
3310 40 40 58 64 44 44 78 00 00 00 7C 20 21 20 1C 00  
3320 04 04 34 4C 44 44 44 3C 00 00 1C 24 3D 21 1D 00  
3330 1C 24 20 70 20 20 20 20 00 00 3D 44 44 3C 04 3C  
3340 40 40 58 64 44 44 44 44 44 44 10 10 10 10 09 00  
3350 04 00 04 04 04 25 24 18 40 40 44 48 50 68 44 00  
3360 11 10 10 10 10 10 09 00 00 00 6C 54 54 44 44 00  
3370 00 00 58 64 44 44 44 44 44 44 00 18 24 24 18 00  
3380 00 00 78 44 44 78 40 40 40 00 38 44 44 3C 04 04  
3390 00 00 58 64 40 40 40 40 00 00 1C 20 18 04 38 00  
33A0 10 10 38 10 10 10 09 00 00 00 44 44 44 4C 32 00  
33B0 00 00 44 44 44 44 28 10 00 00 44 44 54 7C 28 00  
33C0 00 00 44 28 10 28 44 00 00 44 44 44 44 3C 04 1C  
33D0 00 00 7C 08 10 20 7C 00 0C 10 10 20 10 10 0C 00  
33E0 10 10 10 10 10 10 10 10 10 10 60 10 10 0B 10 10 60 00  
33F0 00 00 00 54 58 00 00 00 00 00 00 00 00 00 00 00 00 00 00

NOTE: 1. PROM CONTENTS COPIED TO 3000 (H) BEFORE READING  
2. PAGE ONE OF PROM CONTAINS ALL FF

APPENDIX







**P COMMAND ENTRY**

P

GET 3 ADDRESSES START, END & DESTINATION

20 BE/BF & HL, 20 CO/CI, DE

GET 3 MORE ADDRESSES LO, HI AND TARGET

LO IN 20 BA/BB HI IN 20 BC/BD

SUBTRACT LO FROM TARGET RESULT OFFSET IN BC

CALL MAIN

JUMP TO MONITOR

USE GETNM OF SDK 85

DE HAS 2 BYTES NEXT TO INSTRUCTION

HL > DE

HL < DE

EXCHANGE RESTORE DE

ADD OFFSET TO HL

LOAD L & H TO DESTINATION AND ADVANCE

RESTORE HL

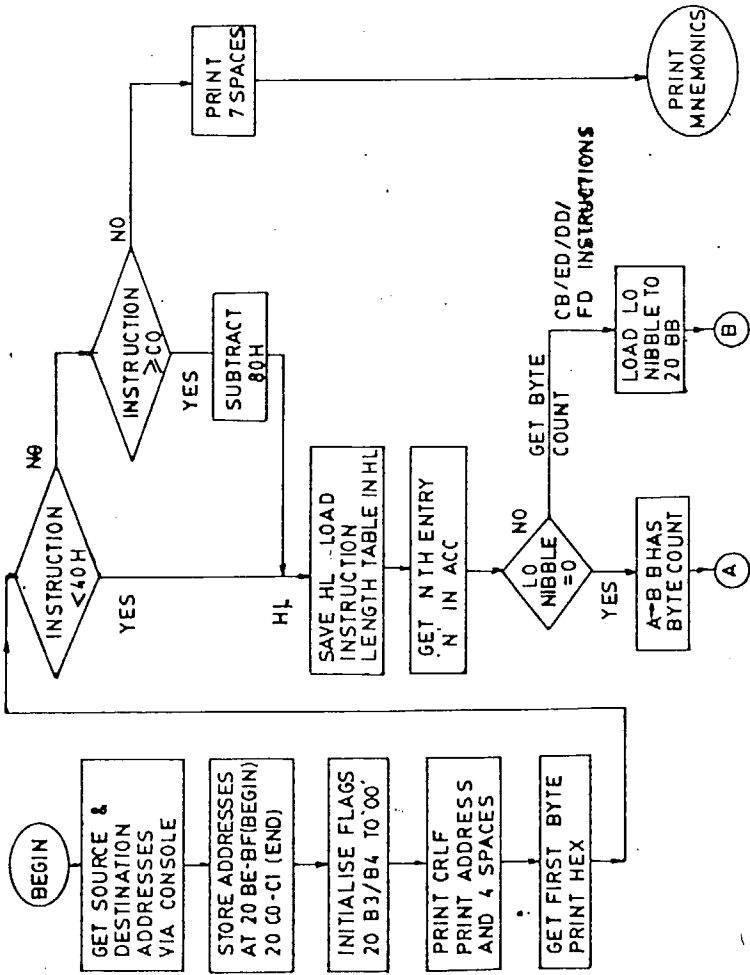
RETURN

2 BYTES NEXT TO INSTRUCTION FALL WITH IN ADDRESS RANGE

(USED AS SUBROUTINE)

# FLOW CHART OF Z-80 DISASSEMBLER

HL HAS CURRENT SOURCE ADDRESS  
 DE HAS DESTINATION ADDRESS  
 BC HAS OFFSET TO BE ADDED





APPENDIX - MZ-80 DISSEMBLER

1. GET THE STARTING & FINAL ADDRESSES TO BE LISTED
2. STORE THEM AT 20BE-BF, 20C0-C1 RESPECTIVELY  
INITIALISE THE FLAGS AT 20B3-B4 TO ZERO
3. PRINT ADDRESS :  
  - PRINT CRLF
  - PRINT CONTENTS OF 20BE-BF
4. PRINT SPACES :  
  - PRINT 4 SPACES
5. PRINT HEX CODE :  

CHECK THE DATA, FIND LENGTH & TYPE CODE FROM TABLES

  - I) 40CH) SEARCH TABLE
  - II) COCH) SUBTRACT 80, SEARCH TABLE
  - III) 40-C0 SINGLE BYTE INSTRUCTION

NOTE : HI BYTE FROM SEARCH TAB GIVES LENGTH -1  
 LOG BYTE GIVES TYPE CODE (CB/ED/DD/FD)  
 STORE TYPE CODE AT 20B3-B4  
 PRINT SOME SPACES AFTER PRINTING CODE
6. PRINT MNEOMONICS :  
  - I) 40CH) SEARCH TABLE (ASCII)
  - II) COCH) SUBTRACT 80, SEARCH TABLE (ASCII)
  - III) 40-C0 SEARCH RESPECTIVE TABLE (ASCII)

## NOTE :

- 1) H, L CONTAINS START ADDRESS OF ASCII STRING TO BE PRINTED END BY 00CH)
- 2) \$ IN THE STRING CHECKS FOR CONTENT OF 20B3 IF IT IS 00 IT PRINTS HL
  - 01 IT PRINTS IX
  - 02 IT PRINTS IY
- 3) IN THE STRING PRINTS '+' AND CONTENTS OF 20B4 AND THEN)
- 4) ? IN THE STRING PRINTS NEXT BYTE
  - ?? IN THE STRING PRINTS NEXT 2 BYTES LAST ONE FIRST
7. INCREMENT CONTENTS OF 20BE-BF BY LENGTH OF PRINTED HEX CODES
8. IF (20BE-BF) IS LESS THAN (20C0-C1) THEN GO TO STEP 2
9. COME TO MONITOR MODE.

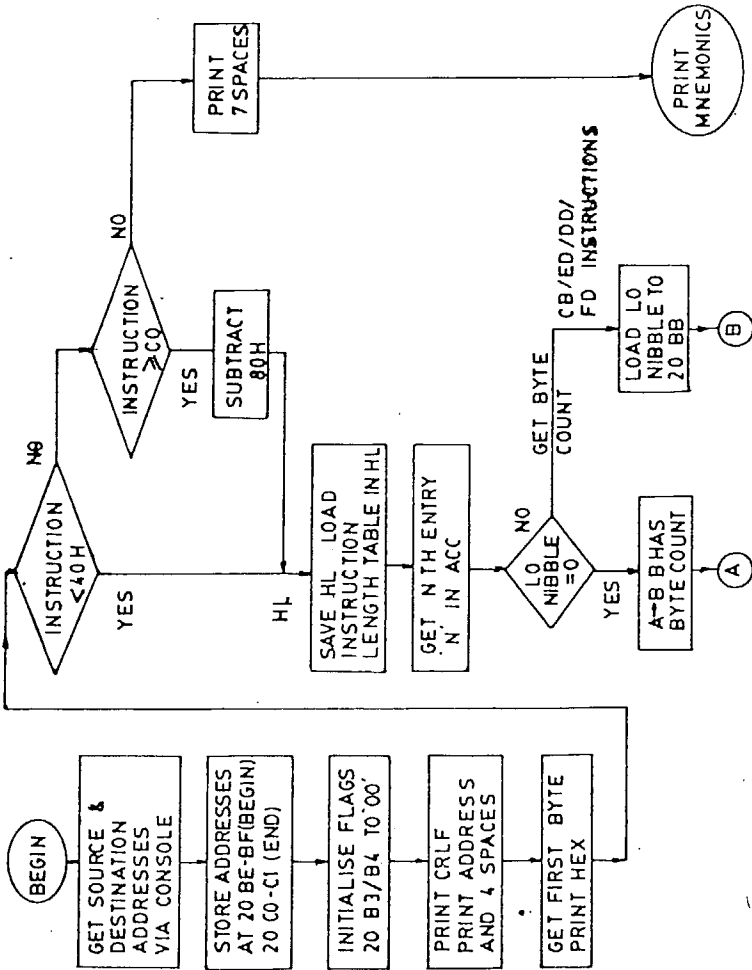
-----

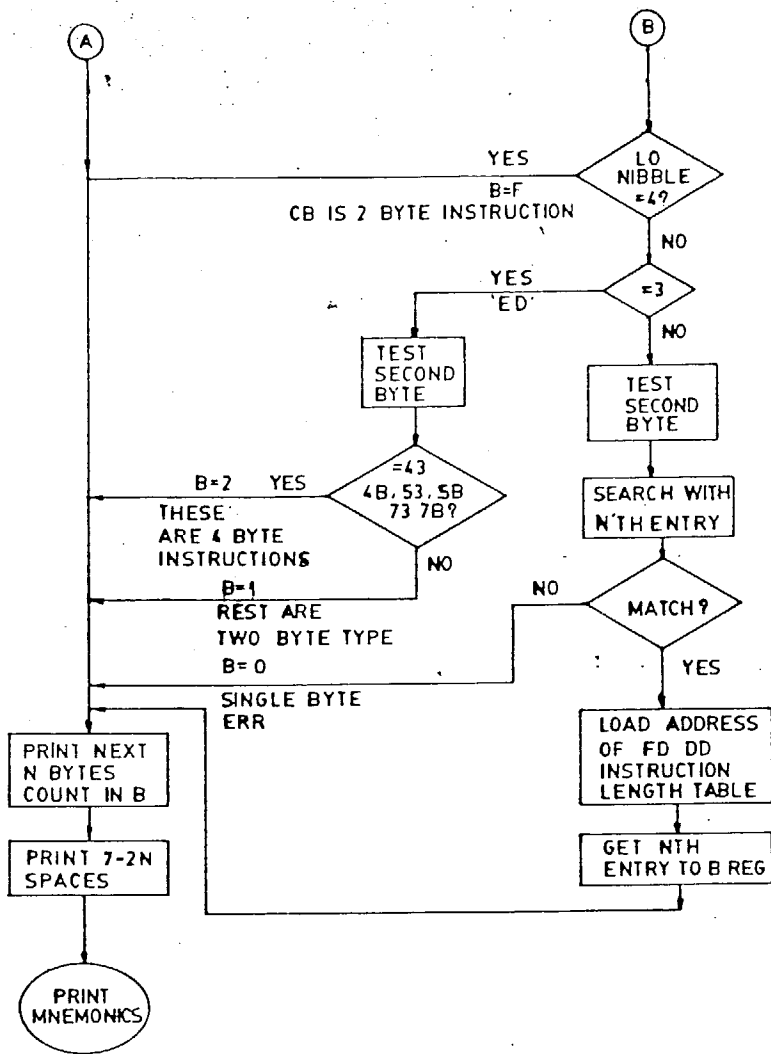
APPENDIX M

FLOW CHART OF Z-80 DISASSEMBLER

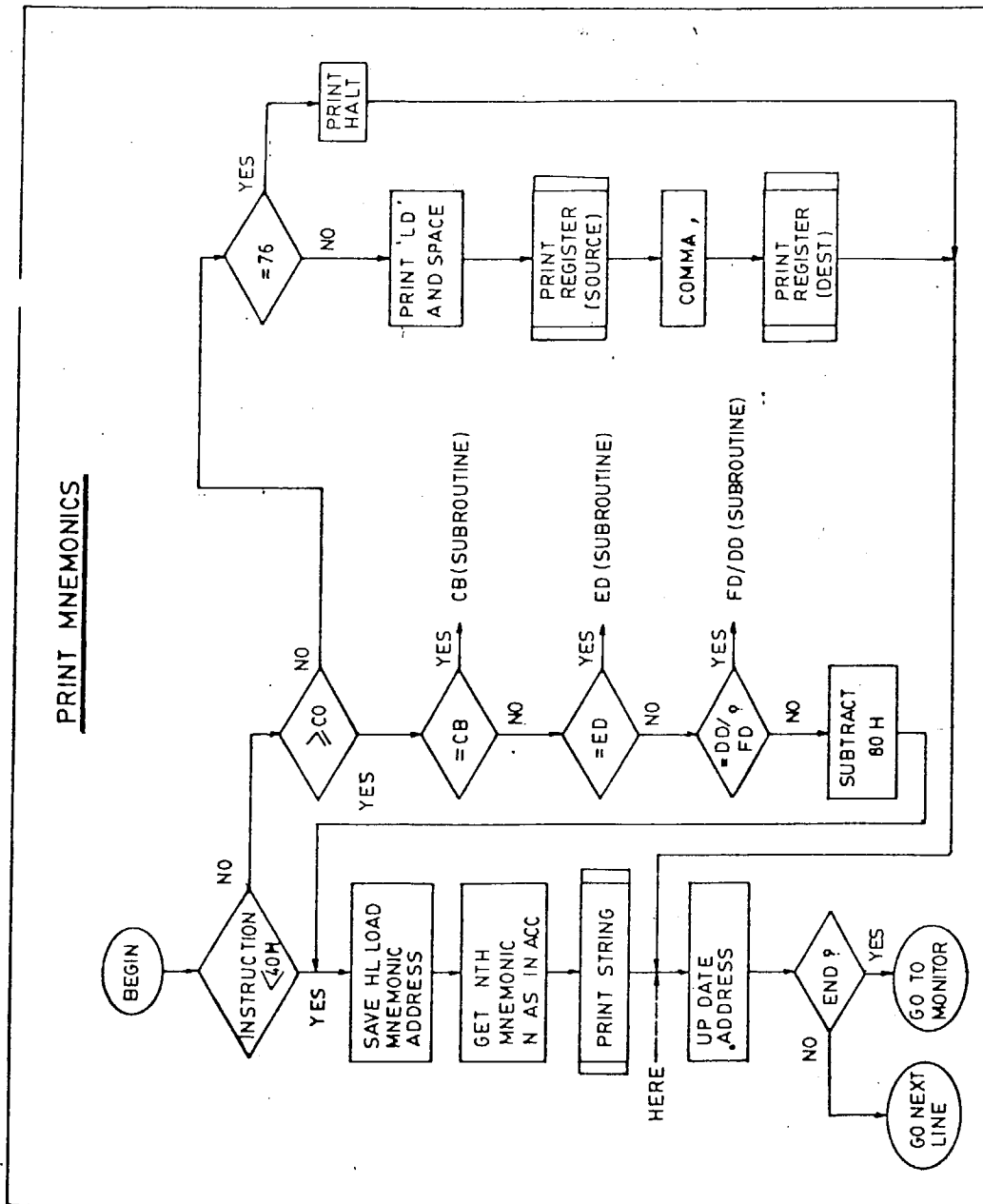
FLOW CHART OF Z-80 DISASSEMBLER

HL HAS CURRENT SOURCE ADDRESS  
 DE HAS DESTINATION ADDRESS  
 BC HAS OFFSET TO BE ADDED



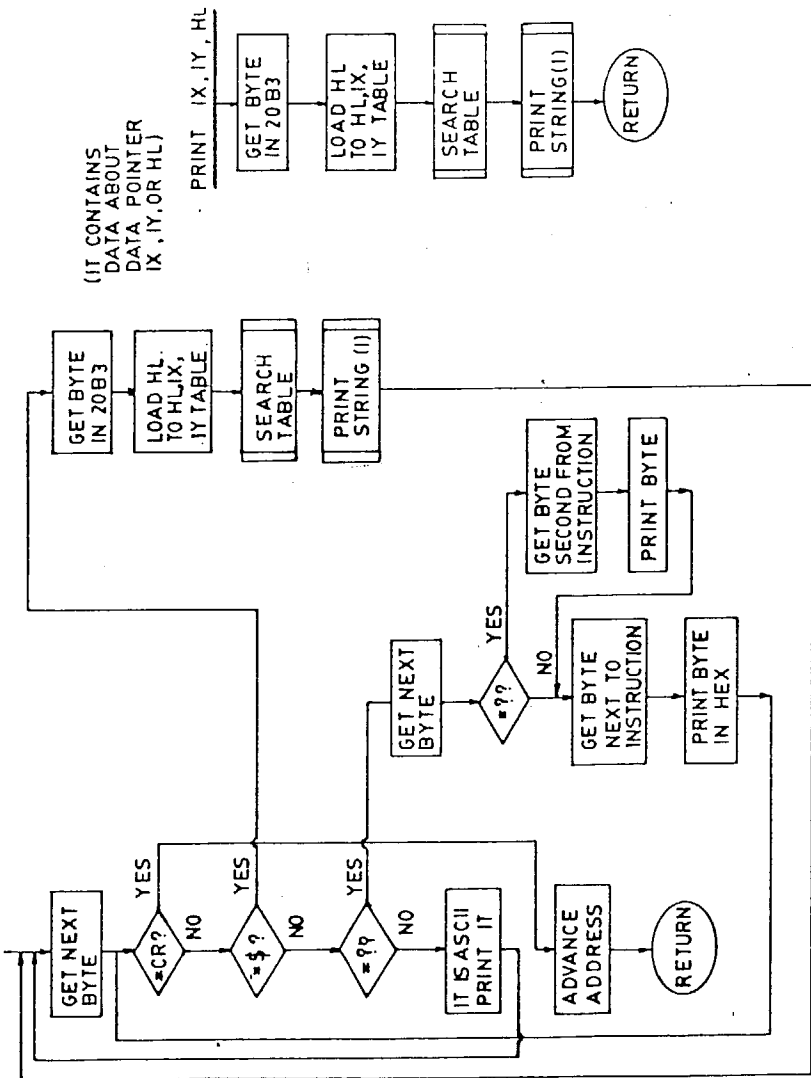


PRINT MNEMONICS

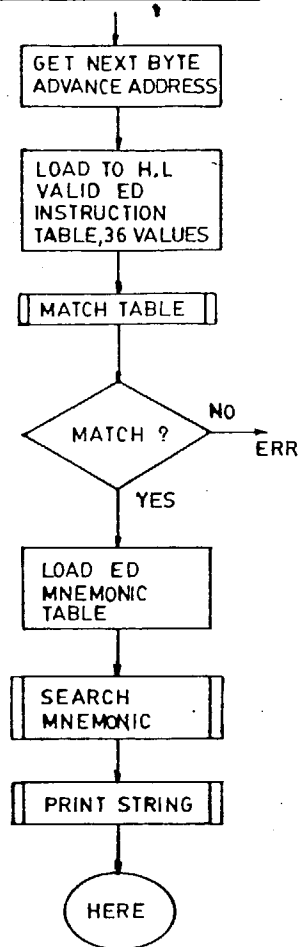


PRINT STRING (WITH VARIATIONS)

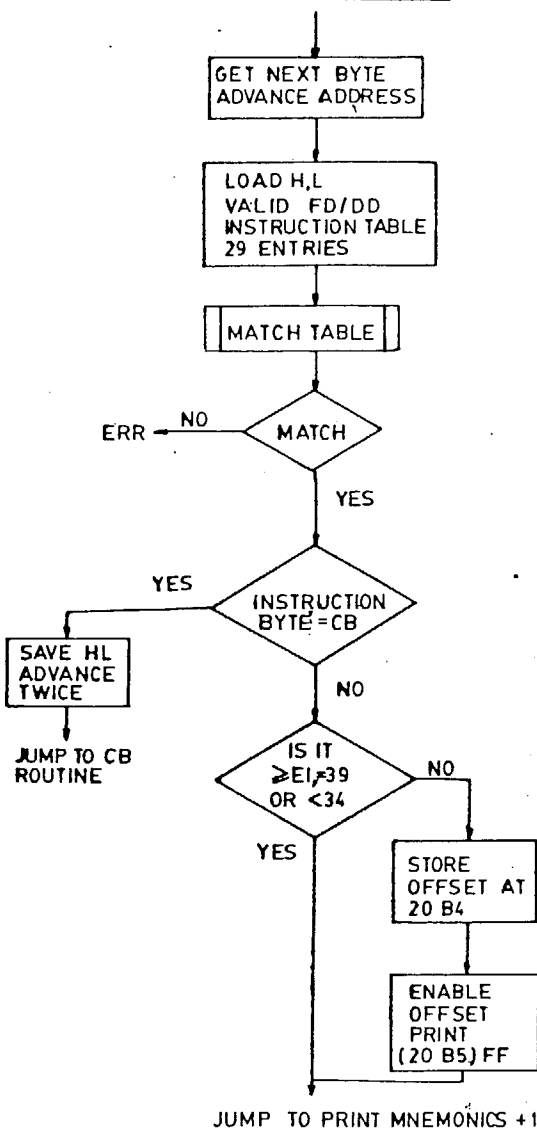
PRINT STRING (WITH VARIATIONS)



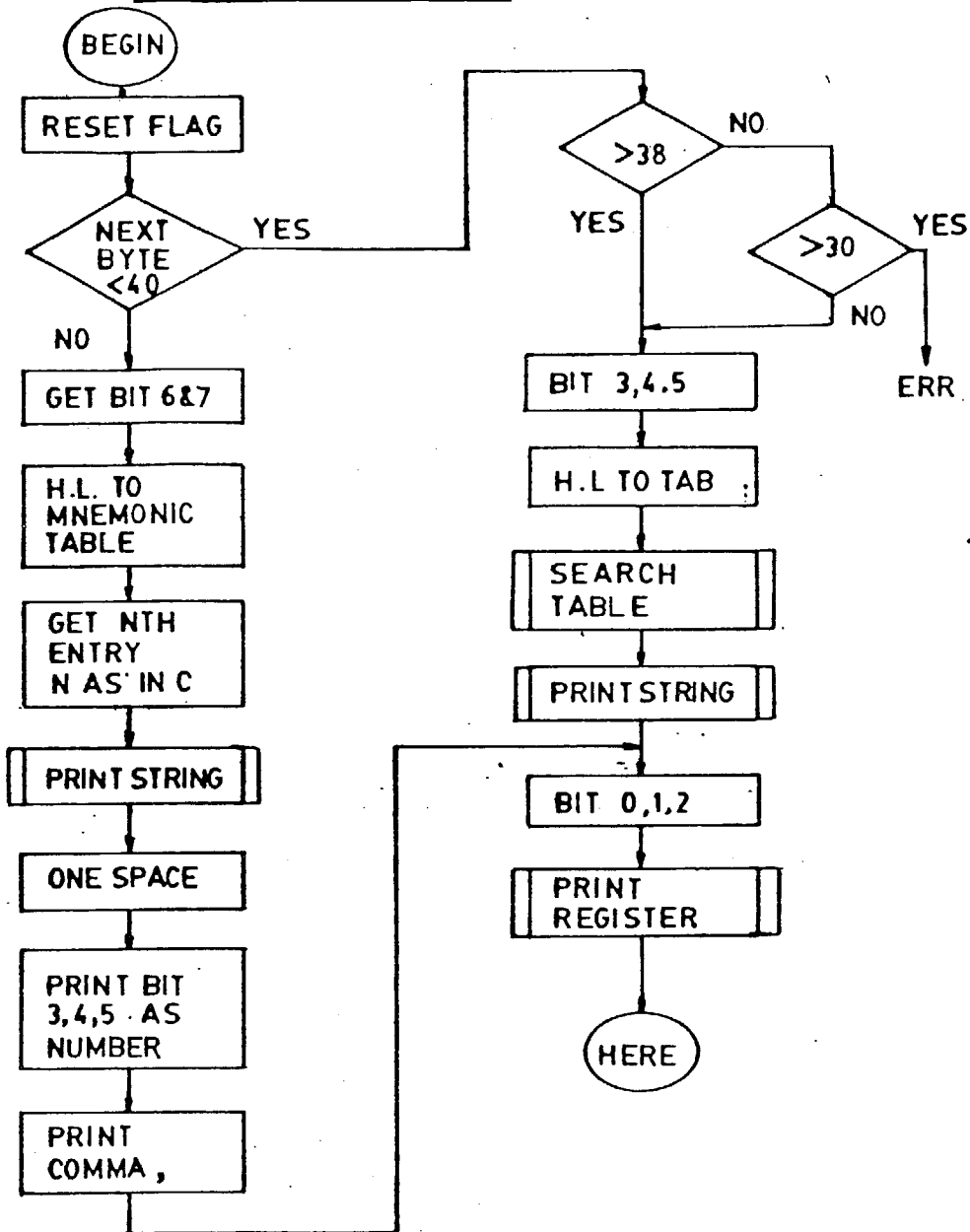
ED INSTRUCTIONS



FD/DD INSTRUCTIONS



## CB INSTRUCTIONS





MNEMONICS OF INSTRUCTIONS 00 -3F AND CO-FF

DD00 4E 4F 50 0D 4C 44 20 42 43 2C 3F 3F 0D 4C 44 20  
 DD10 28 42 43 29 2C 41 0D 49 4E 43 20 42 43 0D 49 4E  
 DD20 43 20 42 0D 44 45 43 20 42 0D 4C 44 20 42 2C 3F  
 DD30 0D 52 4C 43 41 0D 45 58 20 41 46 2C 41 46 27 0D  
 DD40 41 44 44 20 24 2C 42 43 0D 4C 44 41 20 2C 28 42  
 DD50 43 29 0D 44 45 43 20 42 43 0D 49 4E 43 20 43 0D  
 DD60 44 45 43 20 43 0D 4C 44 20 43 2C 3F 0D 52 52 43  
 DD70 41 0D 44 4A 4E 5A 20 3F 0D 4C 44 20 44 4E 2C 3F  
 DD80 3F 0D 4C 44 20 28 44 45 29 2C 41 0D 49 4E 43 20  
 DD90 44 45 0D 49 4E 43 20 44 0D 44 45 43 20 44 0D 4C  
 DDA0 44 20 44 2C 3F 0D 52 4C 41 0D 4A 52 20 3F 0D 41  
 DDB0 44 44 20 24 20 2C 44 45 0D 4C 44 20 41 2C 28 44  
 DDC0 45 29 0D 44 45 43 20 44 45 0D 49 4E 43 20 45 0D  
 DDD0 44 45 43 20 45 0D 4C 44 20 45 2C 3F 0D 52 52 41  
 DDE0 0D 4A 52 20 4E 5A 2C 3F 0D 4C 44 20 24 2C 3F 3F  
 DDF0 0D 4C 44 20 28 3F 3F 29 2C 24 0D 49 4E 43 20 24  
 DD00 0D 49 4E 43 20 4B 0D 44 45 43 20 48 0D 4C 44 20  
 DD10 48 2C 3F 0D 44 41 41 0D 4A 52 20 5A 2C 3F 0D 41  
 DD20 44 44 20 24 2C 44 20 24 0D 4C 44 20 24 2C 2B 3F 3F 29  
 DD30 0D 44 45 43 20 24 20 0D 49 4E 43 20 4C 0D 44 45  
 DD40 43 20 4E 43 2C 3F 0D 4C 44 20 53 0D 4C 0D 4A  
 DD50 52 20 4E 43 2C 3F 29 2C 41 0D 49 4E 43 20 53 50  
 DD60 4C 44 20 28 3F 3F 29 2C 41 0D 49 4E 43 20 28 24 29  
 DD70 0D 49 4E 43 20 28 24 29 2C 3F 0D 53 46 0D 4A 52  
 DD80 0D 4C 44 20 28 24 29 2C 3F 0D 53 46 0D 4A 52  
 DD90 20 43 2C 3F 0D 41 44 44 20 24 2C 53 50 0D 4C 44  
 DDA0 20 41 2C 28 3F 3F 29 0D 44 45 43 20 53 50 0D 49  
 DDB0 4E 43 20 41 0D 44 45 43 20 41 0D 4C 44 20 41 2C  
 DDC0 20 42 43 0D 4A 50 20 4E 5A 2C 3F 0D 5A 0D 50 55  
 DDD0 3F 0D 43 41 4C 4C 20 4E 5A 2C 3F 0D 52 53  
 DDE0 53 48 20 42 43 0D 41 44 44 20 41 2C 3F 0D 50 55  
 DDF0 54 20 4F 0D 52 45 54 20 5A 0D 52 45 54 0D 4A 53  
 DD00 20 5A 2C 3F 3F 0D 40 0D 43 41 4C 4C 20 5A 2C 3F  
 DD10 3F 0D 43 41 4C 4C 20 3F 0D 41 44 43 20 41 2C  
 DD20 3F 0D 52 53 54 20 38 0D 52 45 54 20 4E 43 0D 50  
 DD30 4F 50 20 44 45 0D 4A 50 20 4E 43 2C 3F 0D 4F  
 DD40 55 54 20 28 3F 29 2C 41 0D 43 41 4C 4C 20 4E 43  
 DD50 2C 3F 3F 0D 50 55 53 48 20 44 45 0D 53 55 42 20  
 DD60 3F 0D 52 53 54 20 31 30 48 0D 52 45 54 20 43 0D  
 DD70 45 58 58 0D 4A 50 20 43 2C 3F 3F 0D 49 4E 20 41  
 DD80 2C 28 3F 29 0D 43 41 4C 4C 20 43 2C 3F 0D 4F 0D  
 DD90 0D 53 42 40 41 2C 3F 0D 52 53 54 20 31 38 48  
 DD00 0D 52 45 54 20 50 4F 0D 50 4F 50 20 24 0D 4A 50  
 DD10 20 50 4F 2C 3F 0D 45 58 20 28 53 50 29 2C 24  
 DD20 0D 43 41 4C 4C 20 50 4F 2C 3F 0D 50 55 53 48  
 DD30 20 24 0D 41 4E 44 20 3F 0D 52 53 54 20 32 30 4B  
 DD40 0D 52 45 54 20 50 45 0D 4A 50 20 28 24 29 0D 4A

DD300 D394

DD300 50 20 50 45 2C 3F 3F 0D 45 58 20 44 45 2C 24 0D  
 DD310 43 41 4C 4C 20 50 45 2C 3F 0D 40 0D 58 4F 52  
 DD320 20 3F 0D 52 53 54 20 32 38 48 0D 52 45 54 20 50  
 DD330 0D 50 4F 50 20 41 46 0D 4A 50 20 50 2C 3F 0D  
 DD340 44 49 0D 43 41 4C 4C 20 50 2C 3F 0D 50 55 53  
 DD350 4B 20 41 46 0D 4F 52 20 3F 0D 52 53 54 20 33 30  
 DD360 48 0D 52 45 54 20 4D 0D 4C 44 20 53 50 2C 24 0D  
 DD370 4A 50 20 4C 2C 3F 3F 0D 45 49 0D 43 41 4C 4C 20  
 DD380 4D 2C 3F 3F 0D 40 0D 43 50 20 3F 0D 52 53 54 20  
 DD390 33 38 4B 0D FF

\* INSTRUCTIONS OF ED PRIFIX:MNEMONICS'

DD395 D3D523

DD395 49 4E 20 42 2C 28 43 29 0D 4F 55  
 DD3A0 5A 20 28 43 29 2C 42 0D 53 42 43 20 48 4C 2C 42  
 DD3B0 43 0D 4C 44 20 28 3F 3F 29 2C 42 43 0D 4E 45 47  
 DD3C0 0D 52 45 54 4E 0D 49 4D 20 4F 0D 4C 44 20 49 2C  
 DD3D0 41 0D 49 4E 20 43 2C 28 43 29 0D 4F 55 54 20 28  
 DD3E0 43 29 2C 43 0D 41 44 43 20 48 4C 2C 42 43 0D 4C  
 DD3F0 44 20 42 43 2C 28 3F 29 0D 52 45 54 49 0D 49  
 DD400 4E 20 44 2C 28 43 29 0D 4F 55 54 20 28 43 29 2C  
 DD410 4A 0D 53 42 43 20 48 4C 2C 44 45 0D 4C 44 20 28  
 DD420 3F 3F 29 2C 44 45 0D 49 4D 20 31 0D 4C 44 20 41  
 DD430 2C 49 0D 49 4E 20 45 2C 28 43 29 0D 4F 55 54 20  
 DD440 28 43 29 2C 45 0D 41 44 43 20 48 4C 2C 44 45 0D  
 DD450 4C 4A 20 44 45 2C 28 3F 3F 29 0D 49 4D 20 32 0D  
 DD460 49 4E 20 44 45 2C 28 43 29 0D 4F 55 54 20 28 43 29  
 DD470 2C 4B 0D 53 42 43 20 48 4C 2C 48 4C 0D 52 52 44  
 DD480 0D 49 4E 20 4C 2C 28 43 29 0D 4F 55 54 20 28 43  
 DD490 29 2C 4C 0D 41 44 43 20 48 4C 2C 48 4C 0D 52 4C  
 DD4A0 4A 0D 53 42 43 20 48 4C 2C 53 50 0D 4C 44 20 28  
 DD4B0 3F 3F 29 2C 53 50 0D 49 4E 20 41 2C 28 43 29 0D  
 DD4C0 4F 55 54 20 28 43 29 2C 41 0D 41 44 43 20 48 4C  
 DD4D0 2C 53 50 0D 43 50 49 0D 53 50 2C 28 3F 29 0D 4C  
 DD4E0 4A 49 0D 43 50 49 0D 49 4E 49 0D 4F 55 54 49 0D  
 DD4F0 4C 4A 44 0D 43 50 44 0D 49 4E 44 0D 4F 55 54 44  
 DD500 0D 4C 44 49 52 0D 43 50 49 52 0D 49 4E 49 52 0D  
 DD510 4F 54 49 52 0D 43 50 44 52 0D 49 4E 44 52 0D 4F  
 DD520 54 44 52 0D

'54 VALID INSTRUCTIONS OF CB PRIFIX'

DD525 D55A

D525 40 41 42 43 44 45 46 47 48 49 4A  
D530 4B 4D 50 51 52 53 56 57 58 59 5A 5B 5E 60 61 62  
D540 67 68 69 6A 6F 72 73 78 79 7A 7B 80 A1 A2 A3 AB  
D550 A9 AA AB B0 B1 B2 B3 B8 B9 BA BE

'INSTRUCTION LENGTH AND TYPE TABLE USED

BY PMOVE AND DISSEMBLER

DD560 D5DF

D560 00 20 00 00 00 00 10 00 00 00 00 00 00 00 10 00  
D570 10 20 00 00 00 00 10 00 10 00 00 00 00 00 10 00  
D580 10 20 20 00 00 00 10 00 10 00 20 00 00 00 10 00  
D590 10 20 20 00 00 00 10 00 10 00 20 00 00 00 10 00  
D5A0 00 00 20 20 20 00 10 00 00 00 20 04 20 20 10 00  
D5B0 00 00 20 10 20 00 10 00 00 00 20 10 20 01 10 00  
D5C0 00 00 20 00 20 00 10 00 00 00 20 00 20 03 10 00  
D5D0 00 00 20 00 20 00 10 00 00 00 20 00 20 02 10 00

'6 INSTRUCTIONS OF EDXX TYPE OF 4 BYTE LENGTH'

DD5E0 D5E6

D5E0 43 4B 53 5B 73 7B FF

PRINT HEX CODE

```

0800 7E LD A,(HL);
0801 F5 PUSH AF;
0802 CDC706 CALL 06C7; PRINT 1ST BYTE
0805 F1 POP AF;
0806 E5 PUSH HL;
0807 FE40 CP 40; < 40 (H) ?
0809 DA1A08 JP C,081A; THEN JUMP ACCORDINGLY
080C FEC0 CP C0; >= CO (H) ?
080E D21808 JP NC,0811; THEN JUMP ACCORDINGLY
0811 0607 LD B,07; 7 SPACES FOR 1 BYTE INSTRUCTIONS
0813 CDA008 CALL 08A0;
0816 E1 POP HL;
0817 C9 RET;
*
.L0818 084D
0818 D680 SUB 80; FOR 'CO - FF
081A 216015 LD HL,1560; TAB BEGINNING NO. OF BITS
081D 00 NOP;
081E 85 DD A,L;
081F 6F LD L,A;
0820 7E LD A,(HL); POINT TO CORRESPONDING ENTRY
0821 F5 PUSH AF; GET ENTRY
0822 E6F0 AND F0; SAVE
0824 1F RRA;
0825 1F RRA;
0826 1F RRA;
0827 1F RRA;
0828 47 LD B,A; FIRST NIBBLE (HI) IN B
0829 F1 POP AF;
082A E60F AND OF; LOW NIBBLE
082C C45008 CALL NZ,08; IF IT IS ZERO CB/DD/ED/FD
082F 48 LD C,B; SUBROUTINE GIVES LENGTH
0830 78 LD A,B; ELSE IT IS IN B FROM TABLES B OR 9
0831 A7 AND A;
0832 CA4208 JP Z,0842; ZERO MEANS SINGLE BYTE INSTRUCTION
0835 E1 POP HL;
0836 E5 PUSH HL; C & B HAVE BYTE COUNT - 1
0837 23 INC HL;
0838 7E LD A,(HL);
0839 C5 PUSH BC;
083A CDC706 CALL 06C7; NEXT BYTE PRINTED
083D C1 POP BC;
083E 05 DEC B; ALL BYTES OVER
083F C23708 JP NZ,0837; ELSE REPEAT
0842 3E03 LD A,03;
0844 A9 XOR C;
0845 87 DD A,A; 6+1 SPACES FOR SINGLE BYTE
0846 3C INC A; 0+1 SPACES FOR 3 BYTES
0847 47 LD B,A;
0848 CDA008 CALL 08A0; PRINT SPACES
084B E1 POP HL;
084C C9 RET;

L084D 0850
084D 0600 LD B,00; INITIALISE B TO ZERO
084F C9 RET;
*
.L0850 085B
0850 32B320 LD (20B3),A; CB/DD/ED/FD
0853 FE04 CP 04; STORE COUNT IN FLAG
0855 C25B08 JP NZ,085B; IF 4 THEN JUMP MOR 'CB' ROUTINE
0858 0601 LD B,01; CB IS 2 BYTE INSTRUCTION
085A C9 RET;
*
.L085B 790879
085B 2ABE20 LD HL,(20BE); ED/DD/FD
085E FE03 CP 03; IF FLAG = 3 IT IS ED ELSE JUMP
0860 C27908 JP NZ,0879; NEXT BYTE
0863 23 INC HL;
0864 7E LD A,(HL); ED 6 BYTE INSTRUCTIONS
0865 21E015 LD HL,15E0; INSTRUCTIONS OF EDXX TYPE
0868 0606 LD B,06;
086A 2B DEC HL;
086B 04 INC B;
086C 23 INC HL;
086D 05 DEC B; NO MATCH THEN JUMP B=01
086E CA7708 JP Z,0877;
0871 BE CP (HL);
0872 C26C08 JP NZ,086C; IF MORE, COMPARISON NEEDED
0875 0602 LD B,02; IF MATCH FOUND IT IS
0877 04 INC B; 2+1 BYTE INSTRUCTION
0878 C9 RET;
*
.L0879 0896
0879 23 INC HL; DD/FD
087A 23 INC HL;
087B 7E LD A,(HL); DATA
087C 32B420 LD (20B4),A; IT MUST BE OFFSET
087F 2B DEC HL; GO BACK TO INSTRUCTIONS (SUBGROUP)
0880 7E LD A,(HL);
0881 00 NOP;
0882 21A00B LD HL,08A0; FD/DD INSTRUCTION TABLE
0885 0629 LD B,29; TAB 10 HAS 29 VALID INSTRUCTIONS
0887 CD5C09 CALL 095C;
088A D24D08 JP NC,084D; NO MATCH (ILLEGAL BYTE)
088D 79 LD A,C;
088E 21D00B LD HL,0BDO; FD/DD INSTRUCTION LENGTHS
0891 85 DD A,L; DD/FD INSTRUCTIONS
0892 6F LD L,A;
0893 46 LD B,(HL); IT IS BROUGHT TO B
0894 04 INC B;
0895 C9 RET;

```

```

LD C,20      SPACES
LD D,20      POP AF
PUSH BC      POP HL
CALL 05FB   CALL 05FD
POP BC      CP E1
DEC B       JP NC,0901
JP NZ,08A0  JP Z,0901
RET         CP 34
           JP C,0901
           PUSH AF
           INC HL
           LD A,(HL)
           ; ELSE
           ; GET OFFSET
           LD (20B4),A
           ; STORE
           LD (20BE),HL
           ; ADJUST ADDRESS
           LD A,FF
           ; PRINT OFFSET PRINT
           LD (20B5),A
           ; FLAG TO FF
           POP AF
           JP 0901
           ; GO FOR PRINTING MNEMONICS

*
.L08D0 08D0
08AD F1    POP AF
08AE E1    POP HL
08AF FEE1  CP E1
08B1 D20109 JP NC,0901
08B4 FE39  CP 39
08B6 CA0109 JP Z,0901
08B9 FE34  CP 34
08BB DA0109 JP C,0901
08BE F5    PUSH AF
08BF 23    INC HL
08C0 7E    LD A,(HL)
08C1 32B420 LD (20B4),A
08C4 22BE20 LD (20BE),HL
08C7 3EFF  LD A,FF
08C9 32B520 LD (20B5),A
08CC F1    POP AF
08CD C30109 JP 0901
*
.L08D0 08DD
08D0 2B    DEC HL
08D1 23    INC HL
08D2 7E    LD A,(HL)
08D3 FE0D  CP OD
08D5 C2D108 JP NZ,08D1
08D8 OD    DEC C
08D9 C2D108 JP NZ,08D1
08DC C9    RET
*
.L08DD 08E8
08DD E5    PUSH HL
08DE F5    PUSH AF
08DF 21B80A LD HL,08BB
08E2 CD900A CALL 0A90
08E5 F1    POP AF
08E6 E1    POP HL
08E7 C9    RET
*
.L08EB 08F3
08E8 E5    PUSH HL
08E9 F5    PUSH AF
08EA 215A0A LD HL,0A5A
08ED CD900A CALL 0A90
08F0 F1    POP AF
08F1 E1    POP HL
08F2 C9    RET

LD C,20      SPACES
LD D,20      POP AF
PUSH BC      POP HL
CALL 05FB   CALL 05FD
POP BC      CP E1
DEC B       JP NC,0901
JP NZ,08A0  JP Z,0901
RET         CP 34
           JP C,0901
           PUSH AF
           INC HL
           LD A,(HL)
           ; ELSE
           ; GET OFFSET
           LD (20B4),A
           ; STORE
           LD (20BE),HL
           ; ADJUST ADDRESS
           LD A,FF
           ; PRINT OFFSET PRINT
           LD (20B5),A
           ; FLAG TO FF
           POP AF
           JP 0901
           ; GO FOR PRINTING MNEMONICS

*
.L08AD 08D0
0900 7E    LD A,(HL)
0901 FE40  CP 40
0903 DA2609 JP C,0926
0906 FE80  CP 80
0908 DA130B JP C,0B13
090B FE00  CP C0
090D DA4808 JP C,0B48
0910 FE00  CP CB
0912 CA3209 JP Z,0932
0915 FE00  CP DD
0917 CAB009 JP Z,0980
091A FE00  CP ED
091C CA3A09 JP Z,093A
091F FE00  CP FD
0921 CAB009 JP Z,0980
0924 D680  SUB 80
0926 21FF0F LD HL,OFFF
0929 4F    LD C,A
092A A7    AND A
092B C4D008 CALL NZ,08D0
092E CDA00A CALL 0AA0
0931 C9    RET
*
.L0932 093A
0932 3E00  LD A,00
0934 32B320 LD (20B3),A
0937 C3B709 JP 09B7
*
.L093A 095C
093A 23    INC HL
093B 7E    LD A,(HL)
093C 22BE20 LD (20BE),HL
093F 212515 LD HL,1525
0942 0636  LD B,36
0944 CD5C09 CALL 095C
0947 D2DD08 JP NC,08DD
094A 79    LD A,C
094B 219413 LD HL,1394
094E A7    AND A
094F 4F    LD C,A
0950 00    NOP
0951 00    NOP
0952 C4D008 CALL NZ,08D0
0955 CDA00A CALL 0AA0
0958 22BE20 LD (20BE),HL
095B C9    RET

LD A,(HL)
LD (20BE),HL
LD HL,1525
LD B,36
CALL 095C
JP NC,08DD
LD A,C
LD HL,1394
AND A
LD C,A
NOP
NOP
CALL NZ,08D0
CALL 0AA0
LD (20BE),HL
RET

LD A,(HL)
LD (20BE),HL
LD HL,1525
LD B,36
CALL 095C
JP NC,08DD
LD A,C
LD HL,1394
AND A
LD C,A
NOP
NOP
CALL NZ,08D0
CALL 0AA0
LD (20BE),HL
RET

LD A,(HL)
LD (20BE),HL
LD HL,1525
LD B,36
CALL 095C
JP NC,08DD
LD A,C
LD HL,1394
AND A
LD C,A
NOP
NOP
CALL NZ,08D0
CALL 0AA0
LD (20BE),HL
RET

LD A,(HL)
LD (20BE),HL
LD HL,1525
LD B,36
CALL 095C
JP NC,08DD
LD A,C
LD HL,1394
AND A
LD C,A
NOP
NOP
CALL NZ,08D0
CALL 0AA0
LD (20BE),HL
RET

LD A,(HL)
LD (20BE),HL
LD HL,1525
LD B,36
CALL 095C
JP NC,08DD
LD A,C
LD HL,1394
AND A
LD C,A
NOP
NOP
CALL NZ,08D0
CALL 0AA0
LD (20BE),HL
RET

```

MATCH TABLE

L095C 0970

```

095C 0E00 LD C,00
095E 2B DEC HL
095F 0D DEC C
0960 04 INC B
0961 23 INC HL
0962 0C INC C
0963 05 DEC B
0964 CA6D09 JP Z,096D ; NO MATCH, THEN ERR
0967 RE CP (HL) ; ELSE TEST NEXT BYTE
0968 C26109 JP NZ,0961 ; REPEAT IF MATCH NOT FOUND
096B 37 SCF
096C C9 RET
096D 37 SCF
096E 3F CCF
096F C9 RET
. *
.L0970 0980
0970 219808 LD HL,0989 ; PRINT REGISTER
0973 85 DD A,L ; H,L TO REGISTER CODES TAB 6
0974 7E LD L,A ; H,L TO CORRESPONDING ASCII
0975 6E LD A,(HL) ;
0976 FE24 CP 24 ; IS IT 24 ($)
0978 CC230A CALL Z,0A23 ; IT COULD BE HL,IX OR IY ; TEST IT
097B 4F LD C,A ; BEFORE PRINTING
097C DFB05 CALL 05FB ; END
097F C9 RET
. *
.L0980 09BE
0980 23 INC HL
0981 7E LD A,(HL)
0982 E5 PUSH HL
0983 21A00B LD HL,08A0 ; FD/DD TAB
0986 0629 LD R,29 ;
0988 CD5C09 CALL 095C ; TABLE MATCH
098B D2F408 JP NC,08F4 ;
098E E1 POP HL ;
098F 22BE20 LD (20BE),HL ; STORE ADDRESS POINTER
0992 7E LD A,(HL) ; IS NEXT BYTE CB
0993 FECB CP CB ;
0995 C2AF08 JP NZ,08AF ; IF NOT BYPASS FOR CR
0998 E5 PUSH HL ;
0999 F5 PUSH AF ;
099A 23 INC HL ;
099B 23 INC HL ;
099C 00 NOP ;
099D 7E LD A,(HL) ; DUMMY HAS SUB INSTRUCTION
099E FE36 CP 36 ; IS IT 36
09A0 CAF308 JP Z,08F3 ; THEN BYPASS TO ERR & RESTORE
09A3 E607 AND 07 ; ELSE TABLE BITS 0,1,2
09A5 FE06 CP 06 ;
09A7 C2F308 JP NZ,08F3 ; IF NEITHER X6 OR XE THEN FALSE
09AA F1 POP AF ; PRINT ERR
09AB E1 POP HL ;

```

```

09AC 23
09AD 7E
09AE 32B420
09B1 3EFF
09B3 32B520
09B6 00
09B7 23
09B8 7E
09B9 FE40
09BB DAF609
09BE E5
09BF F5
09C0 E6C0
09C2 1F
09C3 1F
09C4 1F
09C5 1F
09C6 1F
09C7 1F
09C8 217F0A
09CB 4F
09CC A7
09CD C4D008
09E0 CD900A
09D3 0E20
09D5 CDFB05
09DB F1
09D9 F5
09DA E638
09DC 1F
09DD 1F
09DE 1F
09DF C630
09E1 4F
09E2 CDFB05
09E5 0E2C
09E7 CDFB05
09EA F1
09EB E607
09ED CD7009
09F0 E1
09F1 23
09F2 22BE20
09F5 C9
. *

```

```

0993 FECB
0995 C2AF08
0998 E5
0999 F5
099A 23
099B 23
099C 00
099D 7E
099E FE36
09A0 CAF308
09A3 E607
09A5 FE06
09A7 C2F308
09AA F1
09AB E1
09AC 23
09AD 7E
09AE 32B420
09B1 3EFF
09B3 32B520
09B6 00
09B7 23
09B8 7E
09B9 FE40
09BB DAF609
09BE E5
09BF F5
09C0 E6C0
09C2 1F
09C3 1F
09C4 1F
09C5 1F
09C6 1F
09C7 1F
09C8 217F0A
09CB 4F
09CC A7
09CD C4D008
09E0 CD900A
09D3 0E20
09D5 CDFB05
09DB F1
09D9 F5
09DA E638
09DC 1F
09DD 1F
09DE 1F
09DF C630
09E1 4F
09E2 CDFB05
09E5 0E2C
09E7 CDFB05
09EA F1
09EB E607
09ED CD7009
09F0 E1
09F1 23
09F2 22BE20
09F5 C9
. *

```

```

0993 FECB
0995 C2AF08
0998 E5
0999 F5
099A 23
099B 23
099C 00
099D 7E
099E FE36
09A0 CAF308
09A3 E607
09A5 FE06
09A7 C2F308
09AA F1
09AB E1
09AC 23
09AD 7E
09AE 32B420
09B1 3EFF
09B3 32B520
09B6 00
09B7 23
09B8 7E
09B9 FE40
09BB DAF609
09BE E5
09BF F5
09C0 E6C0
09C2 1F
09C3 1F
09C4 1F
09C5 1F
09C6 1F
09C7 1F
09C8 217F0A
09CB 4F
09CC A7
09CD C4D008
09E0 CD900A
09D3 0E20
09D5 CDFB05
09DB F1
09D9 F5
09DA E638
09DC 1F
09DD 1F
09DE 1F
09DF C630
09E1 4F
09E2 CDFB05
09E5 0E2C
09E7 CDFB05
09EA F1
09EB E607
09ED CD7009
09F0 E1
09F1 23
09F2 22BE20
09F5 C9
. *

```

```

.L09F6 0A23
09F6 E5      ; CB < 40 INSTRUCTIONS
09F7 FE3B    ;
09F9 D2010A  ; > 3B(H)
09FC FE30    ; JUMP OVER
09FE D2F408  ; > 30(H)
0A01 F5      ; ERR. ILLEGAL CODES 30-37(H)
0A02 E63B    ;
0A04 1F      ; BITS 3,4,5
0A05 1F      ;
0A06 1F      ;
0A07 215F0A  ; H,L TO TAB 13
0A0A 4F      ;
0A0B A7      ;
0A0C C4D108  ; CALL NZ,08D1
0A0F CD900A  ; CALL 0A90
0A12 0E20    ; LD C,20
0A14 CDFB05  ; CALL 05FB
0A17 F1      ; POP AF
0A18 E607    ; AND 07
0A1A CD7009  ; CALL 0970
0A1D E1      ; POP HL
0A1E 23      ; INC HL
0A1F 22BE20  ; LD (20BE),HL
0A22 C9      ; RET

.L0A23 0A4F
0A23 0E28    ; LD C,28
0A25 CDFB05  ; CALL 05FB
0A28 3AB320  ; LD A,(20B3)
0A2B 21500A  ; LD HL,0A50
0A2E A7      ; AND A
0A2F 4F      ; LD C,A
0A30 C4D008  ; CALL NZ,08D0
0A33 CD900A  ; CALL 0A90
0A36 3AB320  ; LD A,(20B3)
0A39 B7      ; DD A,A
0A3A CA480A  ; JP Z,0A48
0A3D 0E2B    ; LD C,2B
0A3F CDFB05  ; CALL 05FB
0A42 3AB420  ; LD A,(20B4)
0A45 CDC706  ; CALL 06C7
0A48 3E29    ; LD A,29
0A4A C9      ; RET
0A4B FF      ; RST 38H
0A4C 4C      ; LD C,H
0A4D 44      ; LD B,H
0A4E 200D    ; JR NZ,0D

PRINT REGISTER HL/IX/IY
; '(
; '(
; CODE AT 20B3(IE) 0,1,2 REP. HL IX IY
; H,L TO TAB 12 (2 ASCII CHAR)
AND A
LD C,A
CALL NZ,08D0
CALL 0A90
LD A,(20B3)
; CODE AT 20B3
; DOUBLE IT
; IF IT IS ZERO, IT WAS HL, BUMP OVER
; PRINT '+'
; PRINT OFFSET FOR IX,IY ONLY
; ')
; ')

```

LOA90 OA9C

```

OA90 23      ; PRINT STRING
OA91 7E      LD A,(HL)
OA92 FE0D    CP OD
OA94 C8      RET Z
OA95 4F      LD C,A
OA96 CDF805 CALL 05FB
OA99 C3900A JP OA90
*

```

LOA00 OAAF

```

OAA0 23      ; PRINT WITH VARIATIONS
OAA1 7E      LD A,(HL)
OAA2 FE0D    CP OD
OAA4 C2AFOA JP NZ,OAAF
OAA7 2ABE20 LD HL,(20BE)
OAAA 23      INC HL
OAA8 22BE20 LD (20BE),HL
OAAE C9      RET

```

LOAF OADE

```

OAF FE24      ; IS IT 24($)
OAB1 C2DE0A JP NZ,OADE
OAB4 3AB320 LD A,(20B3)
OAB7 00      NOP
OAB8 E5      PUSH HL
OAB9 21500A LD HL,0A50
OABC A7      AND A
OABD 4F      LD C,A
OABE C4D008 CALL NZ,0BD0
OAC1 CD900A CALL OA90
OAC4 E1      POP HL
OAC5 00      NOP
OAC6 00      NOP
OAC7 00      NOP
OAC8 00      NOP
OAC9 3AB520 LD A,(20B5)
OACC A7      AND A
OACD CAA00A JP Z,0AA0
OADO OE2B    LD C,2B
OAD2 CDF805 CALL 05FB
OAD5 3AB420 LD A,(20B4)
OADB CDC706 LD A,(20B7)
OADE C3A00A JP OA90
*

```

LOAE OAORO1

```

OAE FE3F      ; IS IT '?'
OAE0 C20C0B JP NZ,OB0C
OAE3 23      INC HL
OAE4 7E      LD A,(HL)
OAE5 E5      PUSH HL
OAE6 2ABE20 LD HL,(20BE)
OAE9 23      INC HL
OAEA FE3F    CP 3F
OAEB C30108 JP NZ,OB01
OAEF 23      INC HL
OAF0 7E      LD A,(HL)
OAF1 CDC706 CALL 06C7
OAF4 2B      DEC HL
OAF5 7E      LD A,(HL)
OAF6 CDC706 CALL 06C7
OAF9 23      INC HL
OAFB 22BE20 LD (20BE),HL
OAFD E1      POP HL
OAFE C3A00A JP OA90
*

```

LOB01 OB0A

```

OB01 7E      LD A,(HL)
OB02 CDC706 CALL 06C7
OB05 22BE20 LD (20BE),HL
OB08 E1      POP HL
OB09 C3A10A JP OAA1
*
LOB0C OB13
OB0C 4F      LD C,A
OB0D CDF805 CALL 05FB
OB10 C3A00A JP OA90
*

```

LOB13 OB41

```

OB13 F5          PUSH AF
OB14 FE76       CP 76
OB16 CA410B     JP Z,OB41
OB19 21480A     LD HL,0A48
OB1C CD900A     CALL 0A90
OB1F F1         POP AF
OB20 F5         PUSH AF
OB21 E63B      AND 3B
OB23 1F        RRA
OB24 1F        RRA
OB25 1F        RRA
OB26 CD7009     CALL 0970
OB29 0E20      LD C,20
OB2B CDF805     CALL 05F8
OB2E 0E2C      LD C,2C
OB30 CDF805     CALL 05F8
OB33 F1        POP AF
OB34 E607      AND 07
OB36 CD7009     CALL 0970
OB39 2ABE20     LD HL,(20BE)
OB3C 23        INC HL
OB3D 22BE20     LD (20BE),HL
OB40 C9        RET
  *
.LOB41 OB48

```

```

OB41 CDE808     CALL 08E8
OB44 F1        POP AF
OB45 C3F808     JP 08F8
  *
.LOB48 OB650B68
OB48 F5        PUSH AF
OB49 E638      AND 38
OB4B 1F        RRA
OB4C 1F        RRA
OB4D 1F        RRA
OB4E 4F        LD C,A
OB4F 21700B     LD HL,0B70
OB52 A7        AND A
OB53 C4D008     CALL NZ,08D0
OB56 CD900A     CALL 0A90
OB59 2B        DEC HL
OB5A 7E        LD A,(HL)
OB5B FE41      CP 41
OB5D C2650B     JP NZ,0B65
OB60 0E2C      LD C,2C
OB62 CDF805     CALL 05F8
OB65 C3330B     JP 0B33
  *

```

LOBFD OB0C31

```

OBFD 2ABE20     LD HL,(20BE)
OC00 3E00      LD A,00
OC02 32R320     LD (20R3),A
OC05 32R520     LD (20R5),A
OC08 CDEB05     CALL 05EB
OC09 7C        LD A,H
OC0C CDC706     CALL 06C7
OC0F 7D        LD A,L
OC10 CDC706     CALL 06C7
OC13 0E20      LD C,20
OC15 CDF805     CALL 05F8
OC18 CD0008     CALL 0800
OC1B 0E20      LD C,20
OC1D CDF805     CALL 05F8
OC20 CD0009     CALL 0900
OC23 2AC020     LD HL,(20C0)
OC26 EB        EX DE,HL
OC27 2ABE20     LD HL,(20BE)
OC2A CDA006     CALL 06A0
OC2D D2FD0B     JP NC,0BFD
OC30 C9        RET
  *
.LOC31 OC42

```

```

OC31 2ABE20     LD HL,(20BE)
OC34 E5        PUSH HL
OC35 CDFD0B     CALL 0BFD
OC38 E1        POP HL
OC39 22BE20     LD (20BE),HL
OC3C CDEB05     CALL 05EB
OC3F C30B04     JP 040B
  *
.LOC80 0 *

```

10

10

10

10

10

10

10

10

10

10

10

10

10

10

10

10

10

10



R C D E H L \$ A REGS

'TABLES HL,IX,IY'

.DOA4C 0A5B

0A4C 4C 44 20 0D LD (CR)  
0A50 4B 4B 4C 0D 49 58 0D 49 59  
H L CR IX CR IY

'OTHER TABLES'

DOA5B 0A8F

0A5B 4B 41 4C 54 0D  
0A60 52 4C 43 0D 52 52 43 0D 52 4C 20 0D 52 52 20 0D  
0A70 53 4C 41 0D 53 52 41 0D 45 52 0D 53 52 4C 0D  
0A80 42 49 54 0D 52 45 53 0D 53 45 54 0D 45 52 52 0D

HALT (CR)

R L C CR R C CR R L CR R R CR  
S L A CR S R A CR E R R CR S R L CR  
B I T CR R E S CR S E T CR E R R CR

'ARITHMETIC INSTRUCTIONS'

DOB70 0B9F

0B70 41 44 44 20 41 2C 0D 41 44 43 20 41 2C 0D 53 55  
0B80 42 20 0D 53 42 43 20 41 2C 0D 41 4E 44 20 0D 5B  
0B90 4F 52 20 0D 4F 52 20 0D 43 50 20 0D FF FF FF

A D D A , CR A D C A , CR S U  
B CR S B C A , CR A N D CR X  
O R CR O R CR C F CR

'VALID INSTRUCTIONS OF FD/DD PREFIX'

DOB80 0BC8

0BA0-09 19 21 22 23 29 2A 2B 34 35 36 39 46 4E 56 5E  
0BB0 66 6E 70 71 72 73 74 75 76 77 7E 86 8E 96 9E A6  
0BC0 AE B6 BE CR E1 E3 E5 E9 F9

'LENGTH OF FD/DD INSTRUCTIONS'

DOBDO 0BFF

0BD0 00 00 02 02 00 00 02 00 01 01 02 00 01 01 01 01  
0BE0 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01  
0BF0 01 01 01 02 00 00 00 00 00 FF FF FF 2A BE 20

APPENDIX - NTINY BASIC INTERPRETER COMMANDS

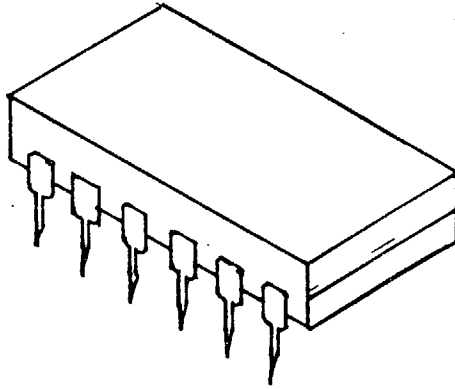
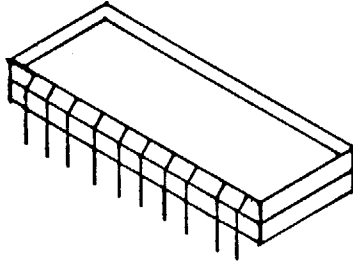
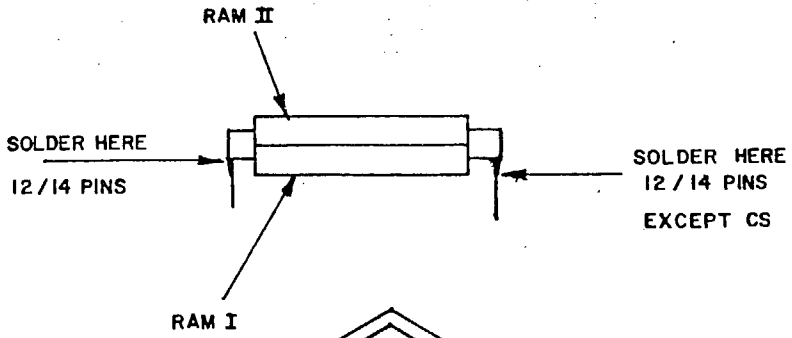
NEW	-	DELETES OLD PROGRAMS
STOP	-	END STATEMENT FOR EACH PROGRAM
RUN	-	RUN PROGRAM
LOAD	-	LOAD A PROGRAM FROM CASSETTE AND DISPLAYS THE NAME
SAVE	-	STORES A NAMED PROGRAM ON A CASSETTE TAPE RECORDER. NAME CAN HAVE ANY NUMBER OF CHARACTERS, BUT ONLY FIRST SIX ARE RETAINED
LIST	-	LIST ALL SAVED LINES
PRINT	-	PRINT COMMAND
FOR	-	CONTROL COMMAND
NEXT	-	CONTROL COMMAND
REM	-	REMARK, THIS LINE IS IGNORED BY TINY BASIC INTERPRETER
IF	-	CONTROL COMMAND
INPUT	-	DATA INPUT COMMAND
LET	-	DEFINES VARIABLES
RND	-	RANDOM NUMBER
ABS	-	ABSOLUTE VALUE
GOTO	-	CONTROL COMMAND
GOSUB	-	SUBROUTINE CALL

RETURN - AND RETURN  
 SIZE - NUMBER OF BYTES LEFT  
 OUT I, J, K, L) IT OUTPUTS EXPRESSION J TO OUTPUT PORT I  
 AND MAY BE REPEATED AS IN DATA 'L' TO  
 PORT K.  
 INP (I) - DATA FROM PORT I IS READ.  
 WAIT I, J, K IT READS STATUS OF PORT I, EXCLUSIVE OR'S  
 THE RESULT WITH K, IN CASE K IS NOT GIVEN  
 IT XOR'S WITH 0, AND'S WITH J AND RETURNS  
 WHEN THE RESULT IS NON-ZERO (IT WOULD BE  
 USED FOR CHECKING PARTICULAR BITS OF  
 PORTS).  
 POKE I, J , LIKE OUTPUT EXCEPT THAT IT INSERTS DATA  
 (I, J, K, L) J IN THE MEMORY LOCATION I.  
 PEEK I - IT GETS THE VALUE FROM LOCATION I  
 USR (I (I, J)) IT CALLS A MACHINE CODE PROGRAM LOCATED  
 AT I, J PARAMETER, WHICH IS OPTIONAL IS A  
 VALUE IN H, L. THE VALUE OF FUNCTION  
 SHOULD BE RETURNED IN H, L.

APPENDIX - 0HOW TO MOUNT TWO RAMs PIGGYBACK ONE ON THE OTHER ?

The signals on the pins of RAMs in a single system are identical except for their chip enable or chip select. Therefore two or even three RAMs can be stacked together with all pins soldered to corresponding pins except for chipselect lines. Now only one socket is needed and, of course, a wire from the each additional RAMs chip select line.

APPENDIX O



APPENDIX - PLIST OF COMPONENTS FOR MAIN CARD

<u>S No</u>	<u>DESCRIPTION</u>	<u>QTY</u>	<u>SNo.</u>	<u>DESCRIPTION</u>	<u>QTY</u>
1.	I/C. No.1 7400	1	26.	I/C. No.26 6264	1
2.	I/C. No.2 74191	1	27.	I/C. No.27 74138	1
3.	I/C. No.3 7493	1	28.	I/C. No.28 74244	1
4.	I/C. No.4 7493	1	29.	I/C. No.29 74245	1
5.	I/C. No.5 7404	1	30.	I/C. No.30 6264	1
6.	I/C. No.6 74165	1	31.	I/C. No.31 Z-80	1
7.	I/C. No.7 2716	1	32.	I/C. No.32 6264	1
8.	I/C. No.8 7486	1	33.	I/C. No.33 74244	1
9.	I/C. No.9 74175	1	34.	I/C. No.34 7404	1
10.	I/C. No.10 74191	1	35.	I/C. No.35 6264	1
11.	I/C. No.11 74374	1	36.	I/C. No.36 7404	1
12.	I/C. No.12 74157	1	37.	I/C. No.37 7400	1
13.	I/C. No.13 74157	1	38.	I/C. No.38 7400	1
14.	I/C. No.14 74157	1	39.	CRYSTAL 8 MHZ	1
15.	I/C. No.15 6116	1	40.	RESISTANCE 47K	2
16.	I/C. No.16 6845	1	41.	RESISTANCE 10K	2
17.	I/C. No.17 74245	1	42.	RESISTANCE 3K	5
18.	I/C. No.18 IS TEST		43.	RESISTANCE 1K	2

SOCKET FOR RxD & TxD  
& PA0 TO PA7 PORT 1

19.	I/C. No.19	74245	1	44.	RESISTANCE	410E	1
20.	I/C. No.20	IS TEST	1	45.	RESISTANCE	330E	1
SOCKET FOR PB & PC PORT							
21.	I/C. No.21	7430	1	46.	RESISTANCE	100E	3
22.	I/C. No.22	8255	1	47.	RESISTANCE	68E	1
23.	I/C. No.23	2764	1	48.	DIODE	IN 4001	2
24.	I/C. No.24	74138	1	49.	CONDENSOR	10 MICRO F	1
25.	I/C. No.25	8251	1	50.	CONDENSOR	1 MICRO F	2

LIST OF COMPONENTS FOR POWER SUPPLY RS232C INTERFACE &  
BATTERY BACK UP

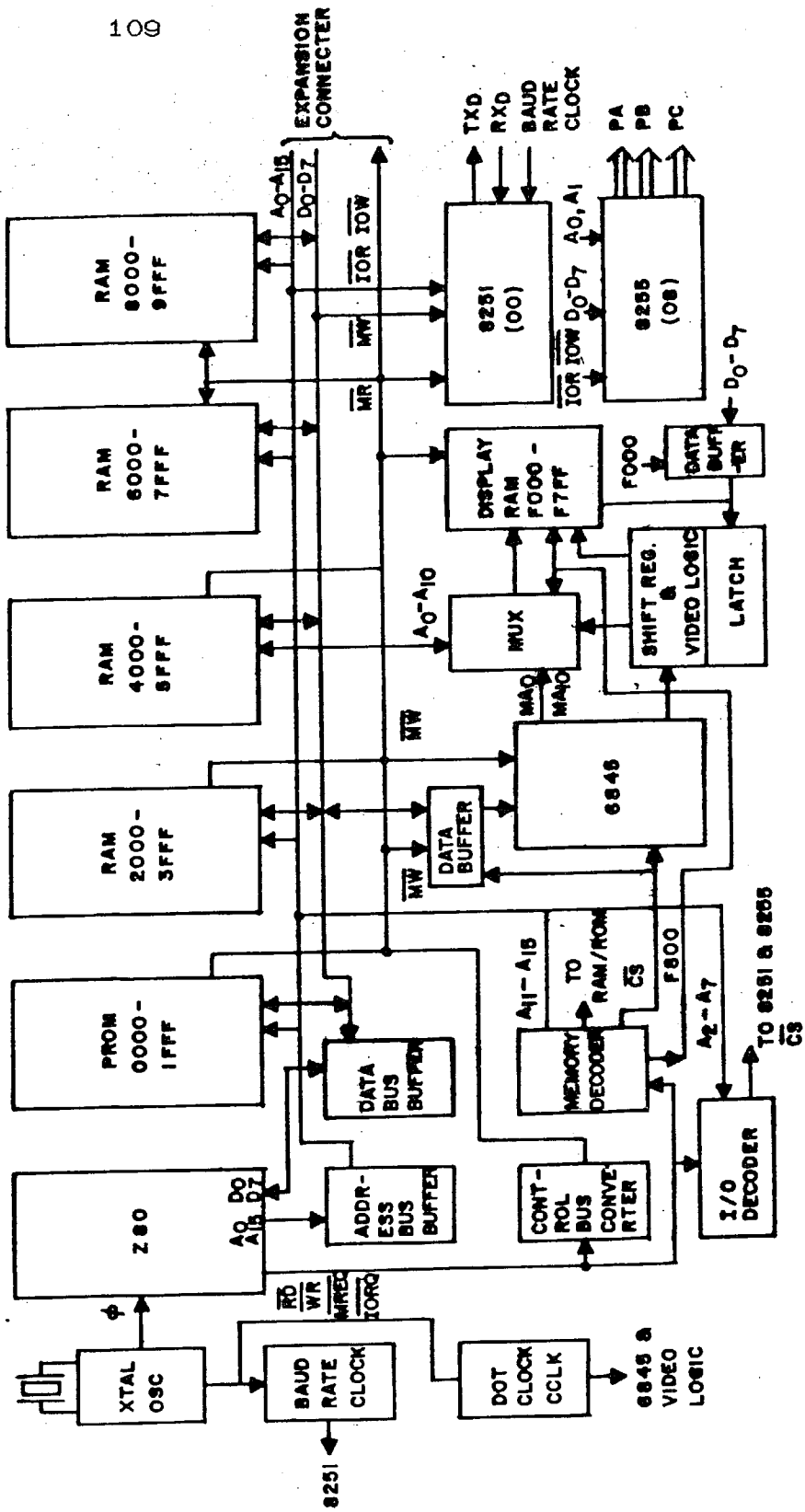
	QTY
1. STEP DOWN TRANSFORMER 9-0-9 VOLTS A/C. 50 HZ AND 1 AMPS	ONE
2. DIODE IN 4001	SIX
3. DIODE IN 4148	TWO
4. ZENER DIODE FZ 4.3 VOLTS	ONE
5. TRANSISTOR 2N2222	TWO
6. THREE TERMINAL REGULATOR 7805	ONE
7. I/C FOR RS232C INTERFACE 1488	ONE
8. I/C FOR RS232C INTERFACE 1489	ONE
9. CONDENSOR 4700 MICROFARAD      16 VOLTS	ONE
10. CONDENSOR 1000 MICROFARAD      16 VOLTS	TWO
11. CONDENSOR      10 MICROFARAD      16 VOLTS	ONE
12. CONDENSOR      .1 MICROFARAD      50 VOLTS	THREE
13. RESISTANCE      220 K OHMS	ONE
14. RESISTANCE      100 K OHMS	ONE
15. RESISTANCE      47 K OHMS	ONE
16. RESISTANCE      10 K OHMS	ONE
17. RESISTANCE      1 K OHMS	ONE
18. RESISTANCE      330 OHMS	ONE
19. BATTERY 1.5 VOLTS UM - 3	TWO
20. 25 PIN MALE CONNECTORS FOR TTY1 & TTYII	TWO



APPENDIX - QCONNECTOR OF KEYBOARD CARD

PIN No. 1	GND
PIN No. 2	NC
PIN No. 3	NC
PIN No. 4	NC
PIN No. 5	R0
PIN No. 6	R1
PIN No. 7	R2
PIN No. 8	R3
PIN No. 9	R4
PIN No. 10	R5
PIN No. 11	R6
PIN No. 12	R7
PIN No. 13	+5 V
PIN No. 14	+5 V
PIN No. 15	C7
PIN No. 16	C6
PIN No. 17	C5
PIN No. 18	C4
PIN No. 19	C3
PIN No. 20	C2
PIN No. 21	C1
PIN No. 22	C0

BLOCK DIAGRAM OF Z80A BASED - MICRO COMPUTER



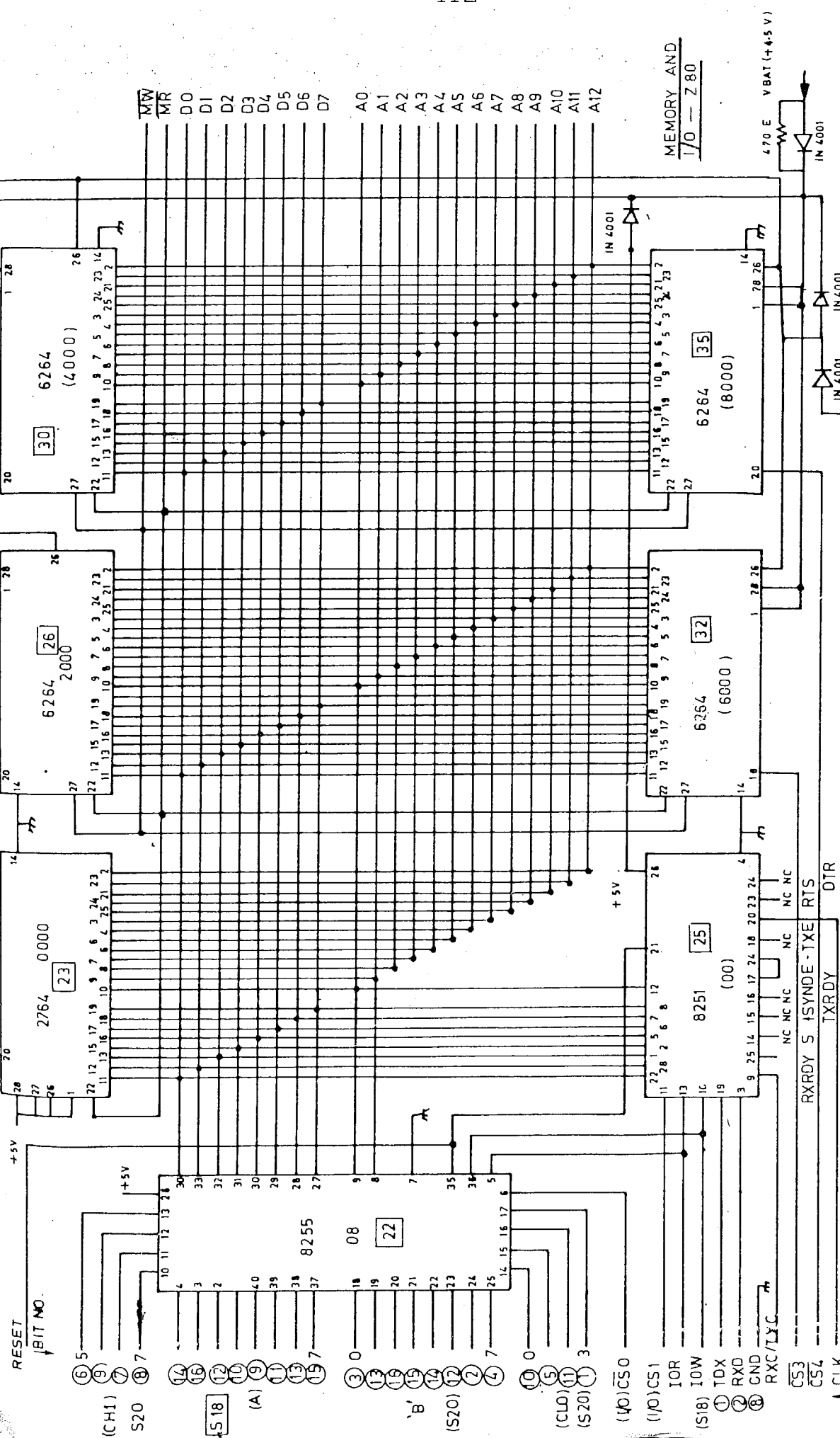
RNM JUN 87

FIG. 1



POWER FAIL

Fig. 4



NUMBER IN CIRCLE INDICATE PIN NUMBER ON I.C. SOCKETS I C 18 AND I C 20  
 NUMBER IN SQUARES INDICATE IC NUMBER ON PCB. **FIG. 4**

R.N. MISRA.

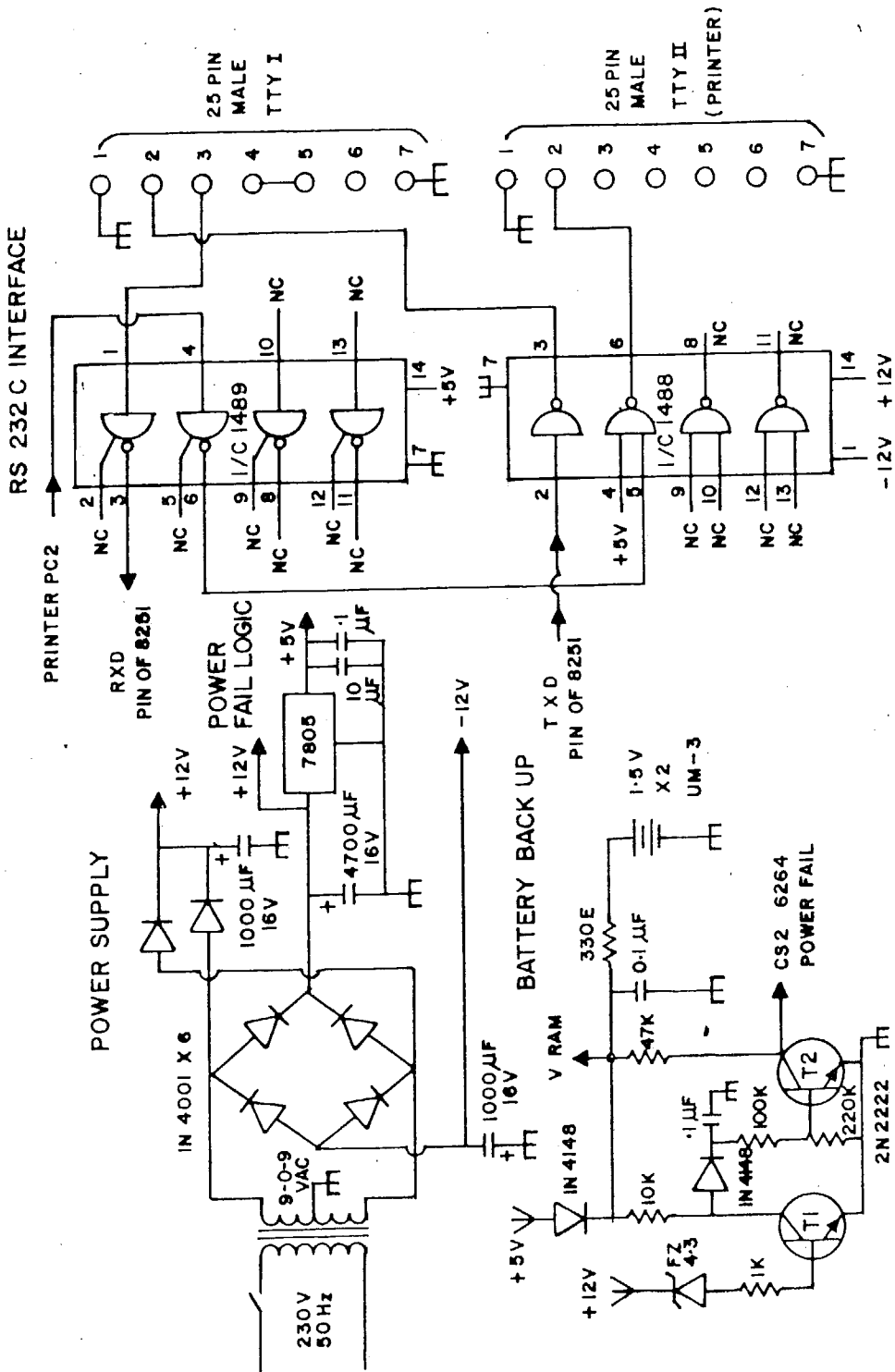


FIG.5 POWER SUPPLY RS 232 C INTERFACE & BATTERY BACK UP

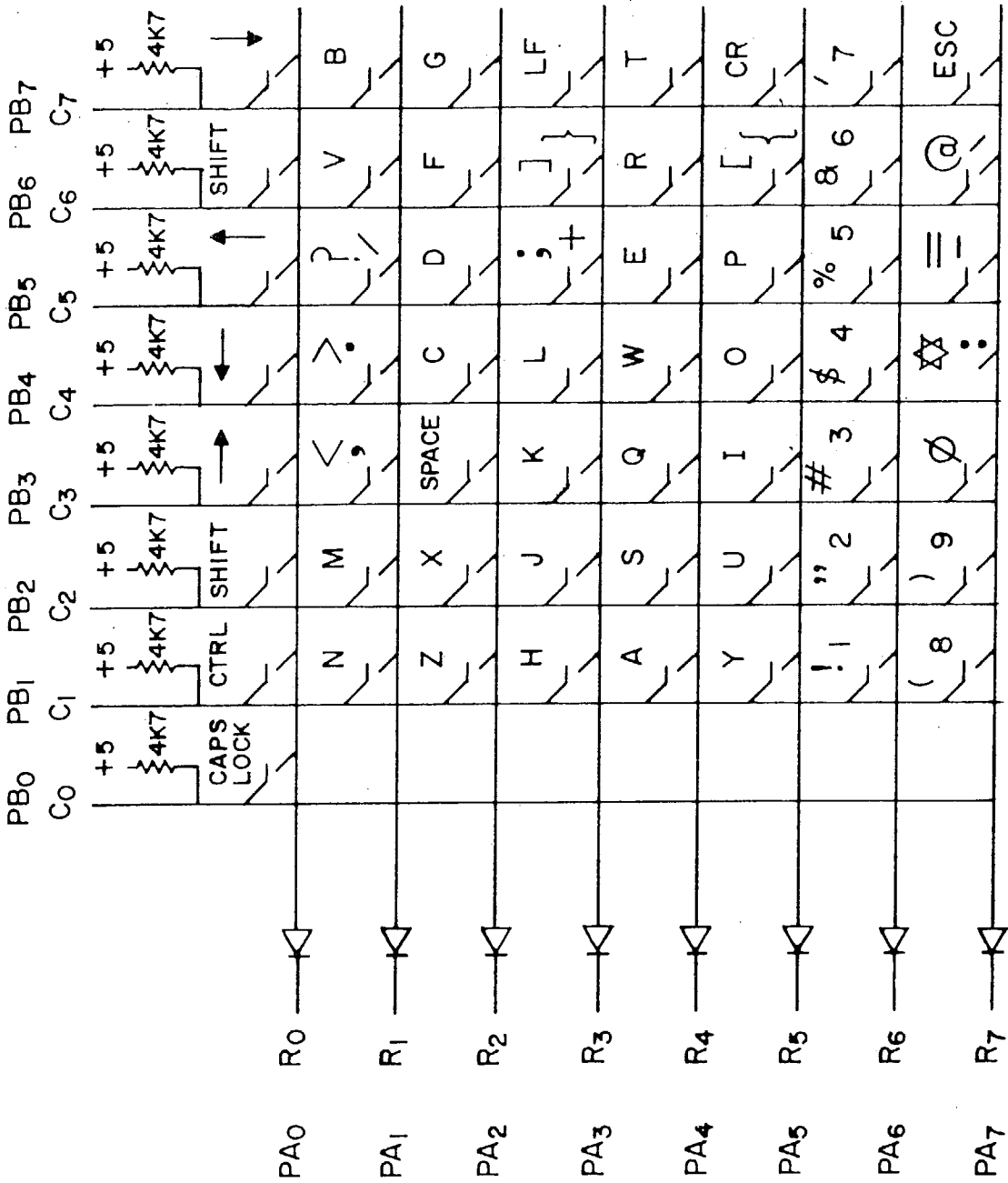


FIG. 6 KEY BOARD ORGANIZATION

# BLOCK DIAGRAM OF CLOCK GENERATOR

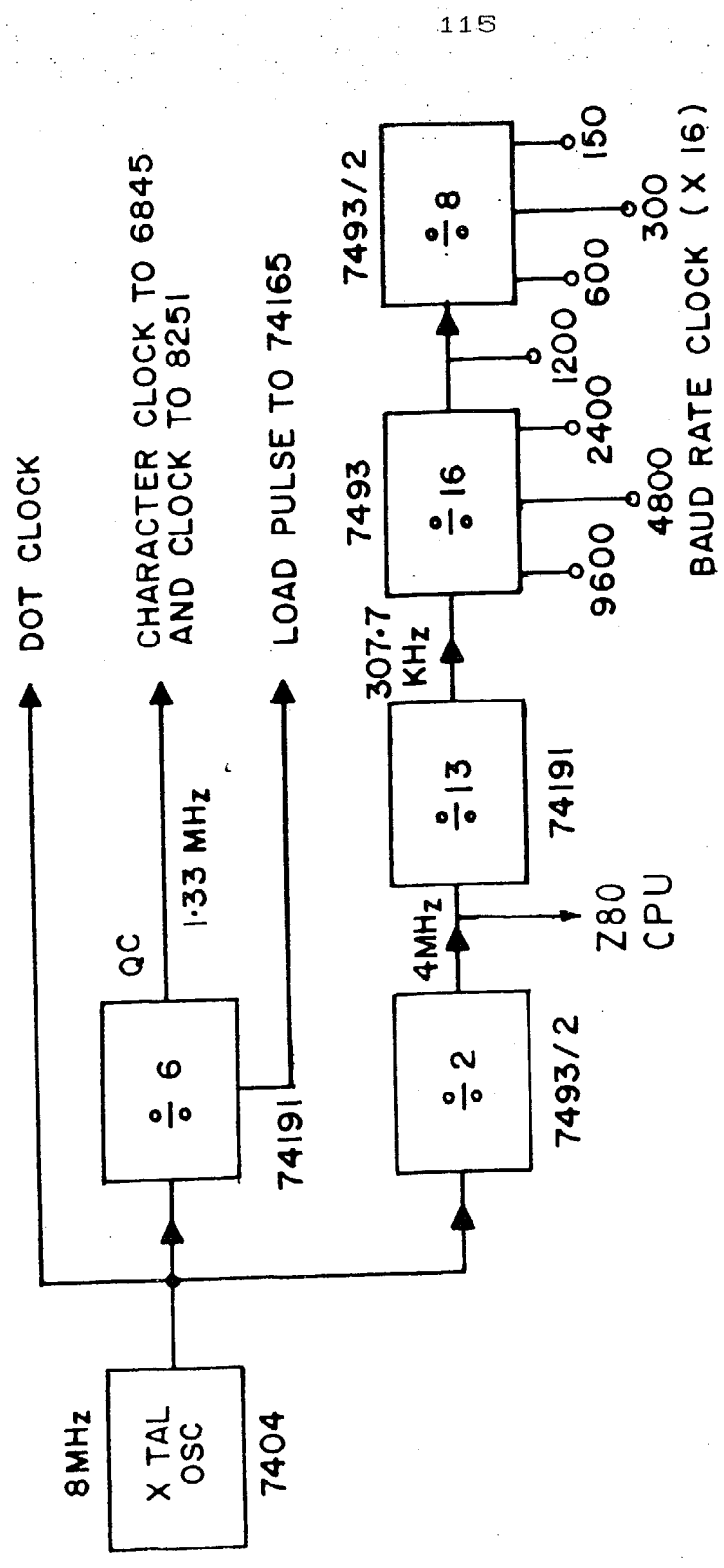
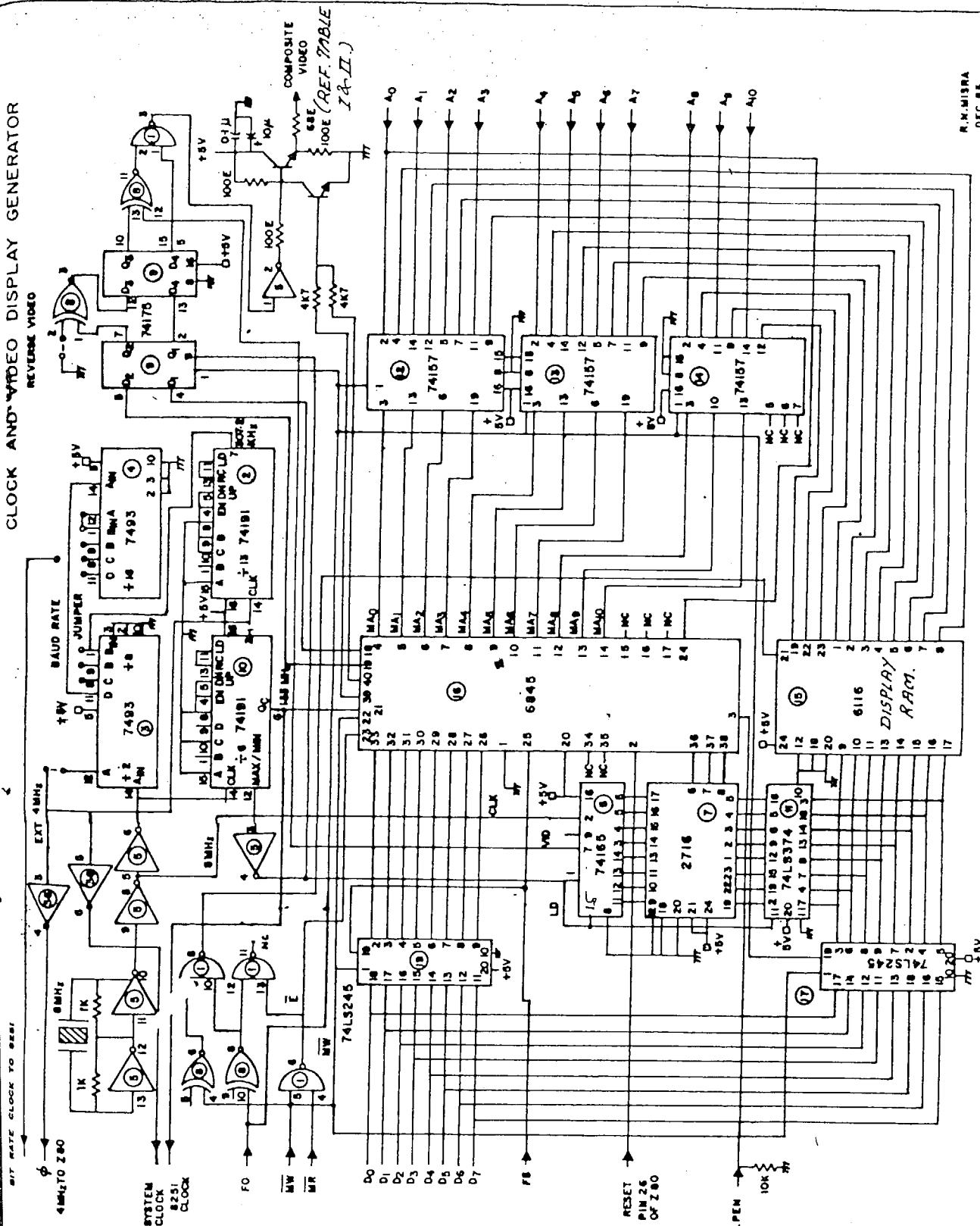


FIG. 7 A

CLOCK AND VIDEO DISPLAY GENERATOR



R.K. MISRA  
DEC. 85

FIG. NO.-7 B





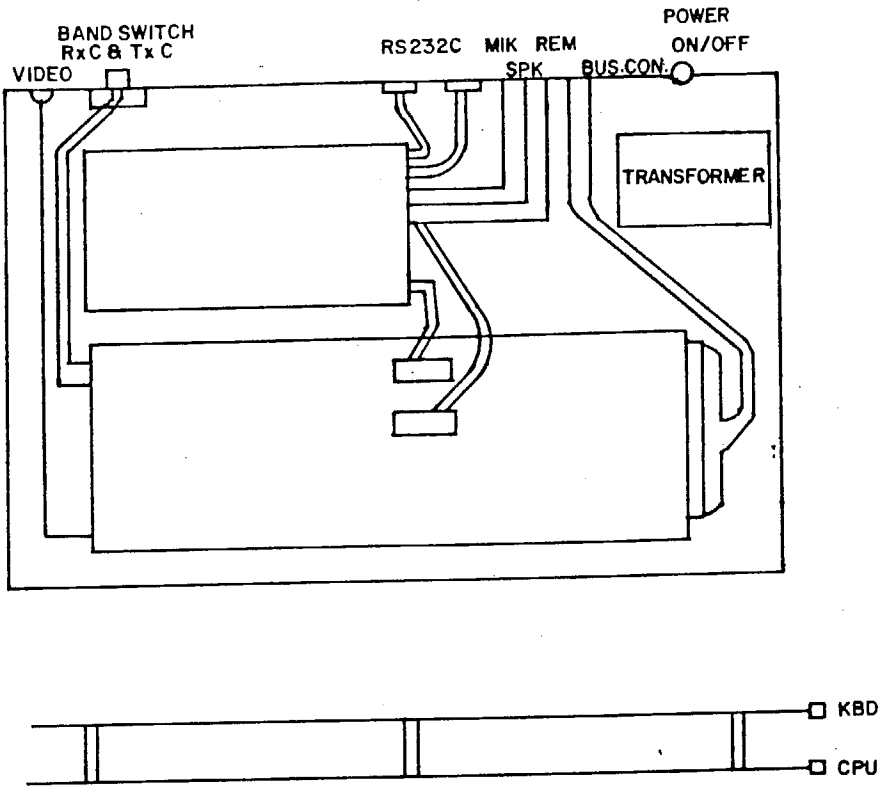
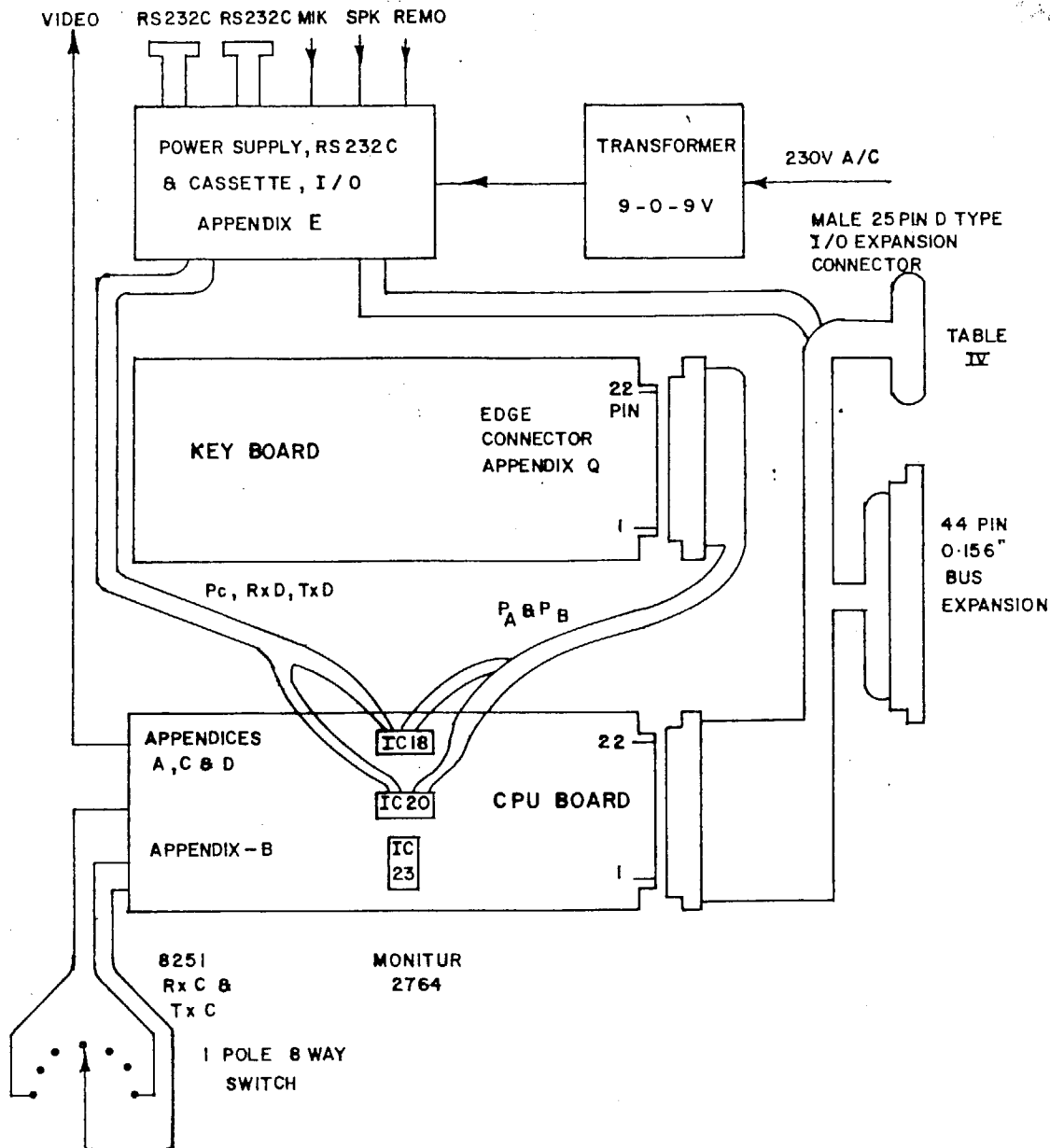


FIG. NO. 9



INTERCARD CONNECTION DIAGRAM

FIG. NO. 10

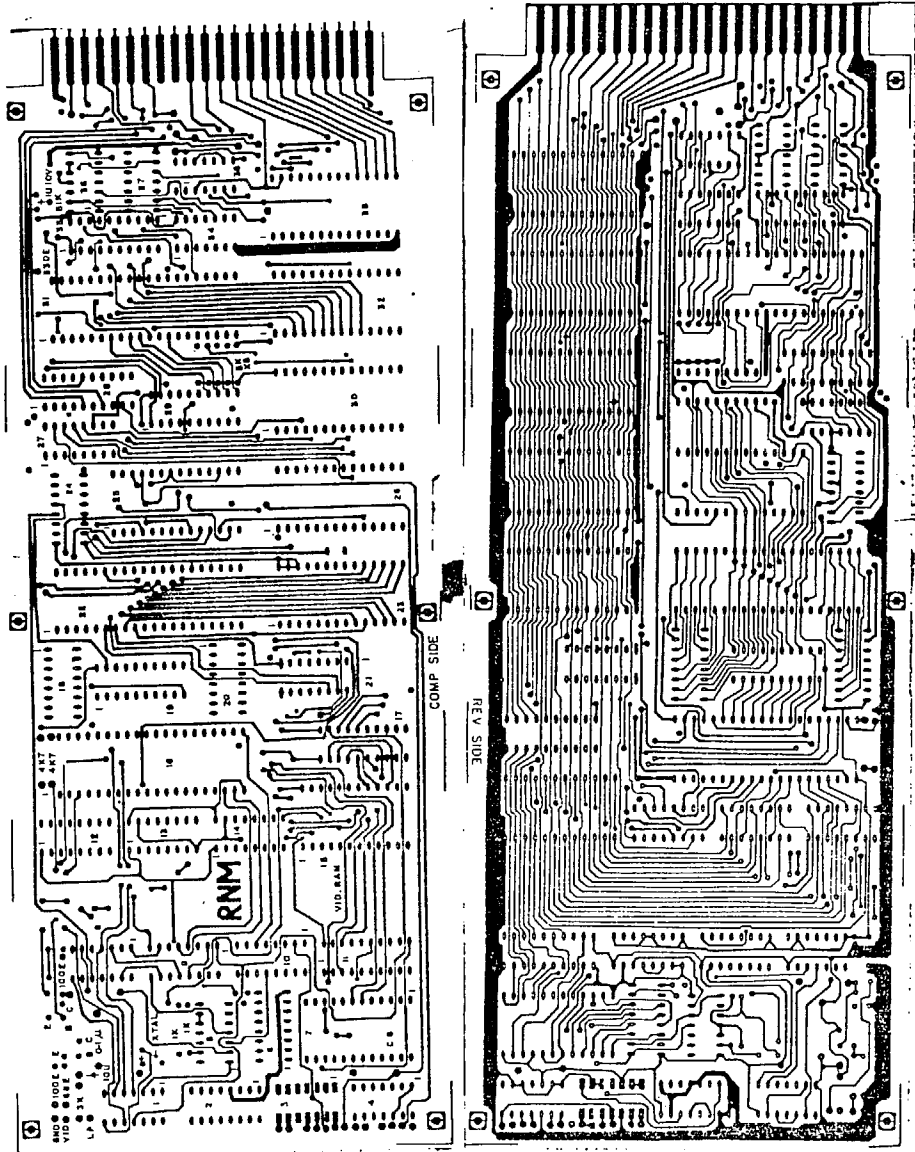


FIG. NO. II

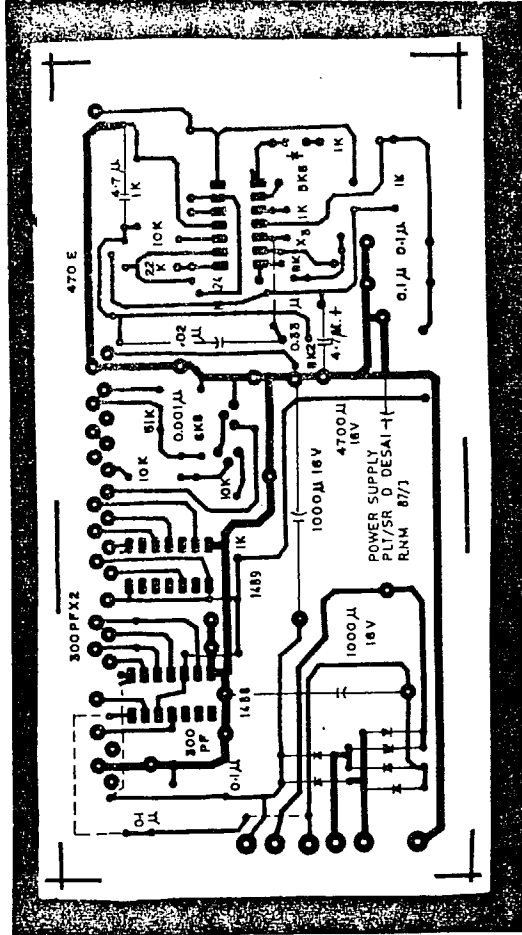


FIG. NO. 12