B. Electrochem 3(1) January-Februay 1987 pp 79-83

MICROPROCESSOR BASED LIFE CYCLE TESTER FOR BATTERIES

M PANDIAMMAL, K MUTHU, K R RAMAKRISHNAN and Y MAHADEVA IYER
Central Electrochemical Research Institute, Karaikudi-623 006

Life cycle testing of batteries requires sophisticated instrumentation for measurement of various parameters to evaluate the performance of batteries in the charge/discharge mode in a production line and logging the measured values for carrying out data processing either with the help of a computer connected off line or on line. The parameters normally measured are voltage, current (charging/discharging), time, number of cycles etc. The measurements are carried out usually under constant current condition. This paper deals with a microprocessor based charging unit for life cycle testing of storage cells and the developed software is fused with the help of the EPROM programmer whose design is also incorporated in this paper.

LIFE CYCLE TESTER FOR BATTERIES

The present system described here is a microprocessor based battery life cycle tester, with software for fixing the charge/discharge cycle and multiplexing of cells upto 256 nos. The unit is easily expandable with proper addition of peripherals to evaluate any parameter. Software modification can be easily carried out unlike in other rigid modules, to meet any specific testing demand. Since the development of the storage cells being an ongoing project, the present system described here enables acquiring and logging of a large number of data for further processing.

SYSTEM DESCRIPTION

The proposed system consists of scanning a series of individual cells which are either in changing mode or discharge mode. The cells are charged with constant current chargers or charging through resistors or SCRs. The cells are also discharged under constant current or through a resistor or through SCRs. The individual cell voltages scanned are logged in RAMs which are expanded upto the level required using the bus expansion facility of SDK 85 kit. The set high voltage (E1) is compared every time during charging and the cells are stopped for charging at a high level upto 3 volts and reversed to discharging mode. Every charge and discharge cycle is logged and recorded. The timing is obtained for charge and discharge cycle from an 8253 timer if constant time charge/discharge mode is carried out, or if the limits are exceeded during charging, the timer reading is logged for reaching the voltage limits of charge and discharge. The independent timer thus enables more flexibility to the microprocessor to attend to service calls as well as sampling only. A detailed schematic diagram for a simple cell charge and discharge cycle is given in Fig.1 and the flow chart of the developed program is given in Fig.2.

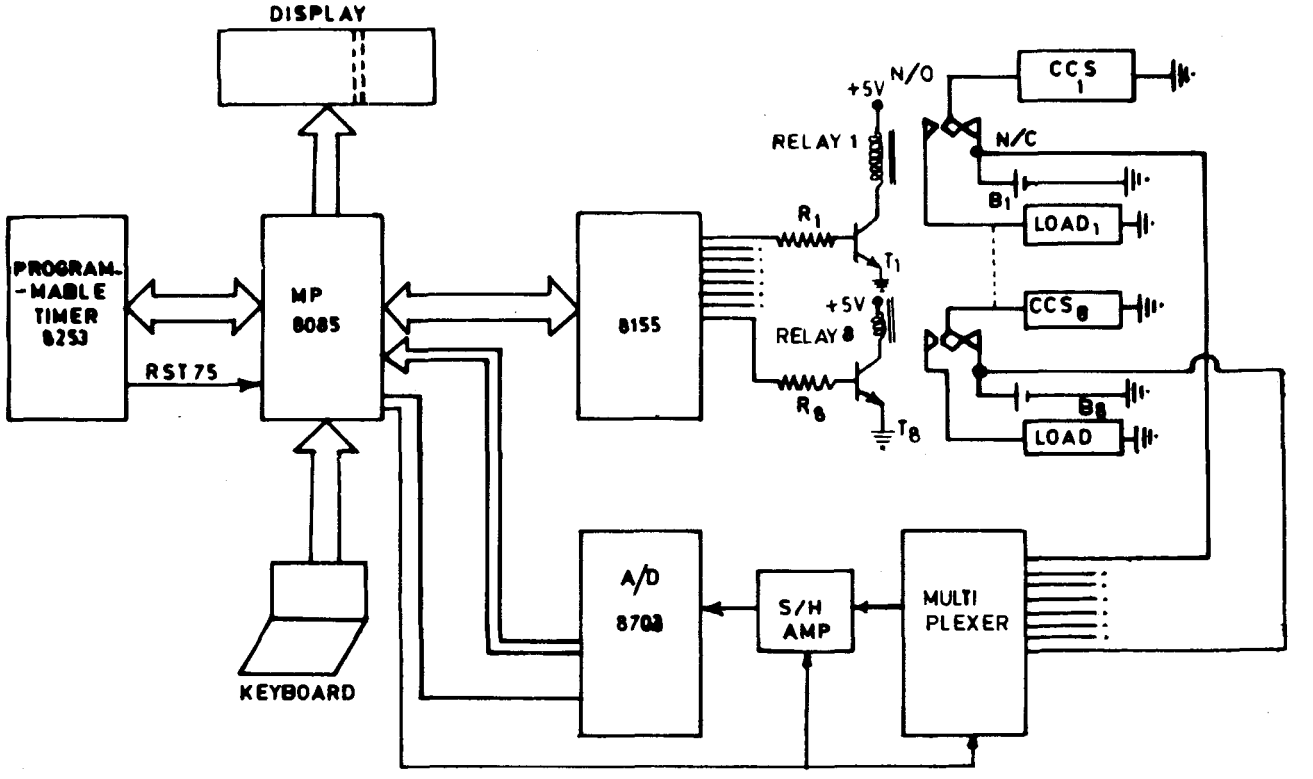

Fig.1: Battery tester schematic

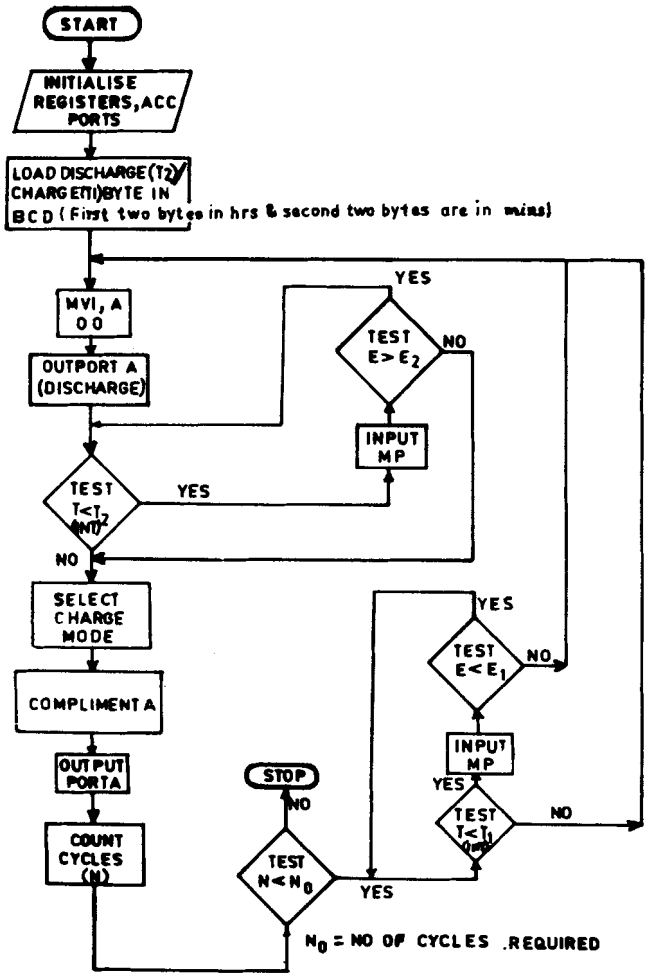

Fig.2: Flow chart for constant time mode

FUNCTIONAL DESCRIPTION

When the instrument is switched on, it changes the mode of testing to discharging mode. Then the discharge time and charge time in hours and minutes are fed into the microprocessor through keyboard. Then the microprocessor will measure the cell voltage for every 1 sec. through A/D convertor 8703 and compares the lower limit of the cell voltage (E2) i.e., upto 1V. After discharge time is over, it will switch over to charging mode. The battery will be changed to charging mode, even before discharge time is over if the cell voltage comes below lower limit. In the charging mode a d.c. power supply at 1 amp current rating is connected to charge the battery (E1), as long as the battery voltage is below the upper limit (E1) upto 3V. For one cycle, one discharge and one charge operations are taken.

The battery will be switched over between discharge and charge modes till it reaches no. of cycles (No) for which the battery is to be put under testing. Once it reaches the required no. of cycles, then the system will come to stop. In the discharging mode, the discharge current is 1 amp. A D.C. resistor is used as load. In the charging mode also, the charging is maintained at 1 amp. through electronic resistor.

The system is built around SDK 85 with bus expansion for additional memory upto 8K bytes [1,3]. A programmable timer 8253 is used in hardware interrupt mode. Most part of the microprocessor time is used for cell voltage monitoring and comparing with set limits (E1, E2, T1 & T2). The counter 0 and counter 1 are used in mode 3 and counter 2 is used in mode 0. The bytes stored in counters are in BCD form. The output of counter 0 is clock input to counter 1 and output of counter 1 is clock input to counter 2 and output of counter 2 is used for RST 7.5 interrupt. The clock output from microprocessor is given to counter 0 clock input. The charge/discharge time is fed into microprocessor in BCD form. That time byte is converted into number i.e. minutes (say 1 hour ten minutes = 70 minutes). This time in minutes in BCD form is stored in counter 2 of 8253. The period of clock input to counter 2 is made equivalent to 1 minute, i.e. frequency of the clock input to counter 2 is 1/60 Hz. So a pulse output will appear at the output pin of counter 2 which will interrupt the microprocessor at the end of TC (terminal count) of counter 2, i.e. at the end of charge/discharge time. At location 20 CE a microprocessor instruction is written to a subroutine program which changes the mode of the battery. Multiplexer 16S, A/D converter 8703 and sample/hold amplifier are used to select batteries and to monitor the battery voltage. Port A of 8255 is used for outputting signals to electronic resistors. The 6 digit display available at the kit will display the cell voltage in address field and display the charge/discharge mode in the data field.

The test system described in this paper is capable of carrying out life cycle testing of nickel-zinc cells to a minimum of one to a maximum of 8 Nos. It could be expanded to 256. The instrument can also be easily expanded to record data such as temperature, state of charge, capacitance, internal resistance etc. for which work is being carried out in this lab.

The specifications are as follows for the testing system.

Mode: 1. Constant time charging/discharging
2. Fixed number of cycles of charging/discharging
3. Endurance mode, voltage sensing mode and recording maximum number of cycles

Charging cutoff voltage (E1)	: upto 3 volts
Discharging cutoff voltage (E2)	: upto 1 volt
Charging/discharging current for each cycle	: 500 mA to 10 A
Max. No. of cells to be connected	: 8 (256)

Programme listing can be supplied on request.

EPROM PROGRAMMER

In a computer or in a microprocessor based system memories are required for storing informations like data, instructions etc. [2,4].

ROM or EPROM is basically a non-volatile memory. This means, even when the power to the memory is switched off, the stored contents are retained. The contents of ROM are either

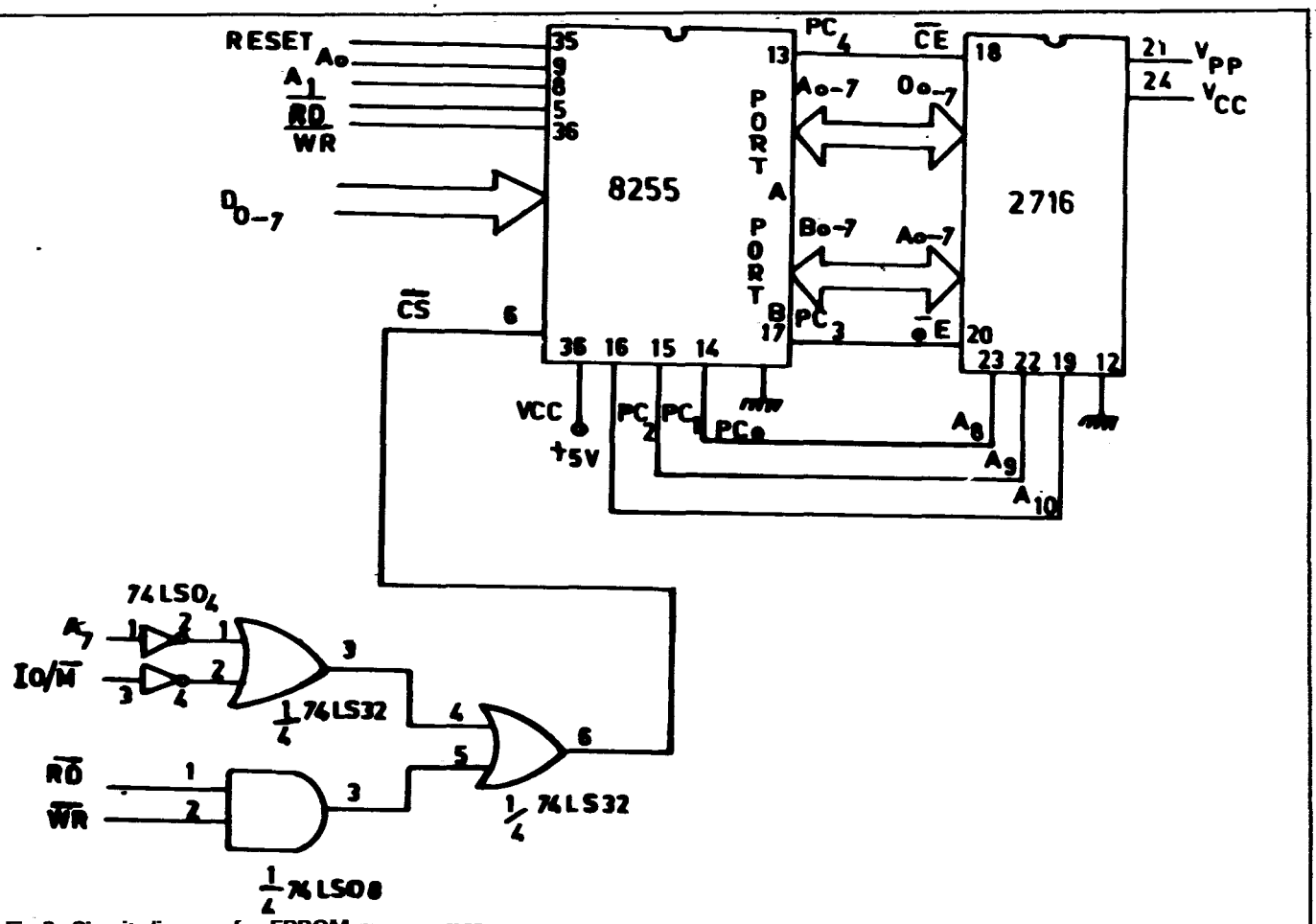

Fig.3: Circuit diagram for EPROM programmer

blown by the manufacturers at the time of manufacturing or by the user with the aid of special programmers like the one presented here. To meet the changing requirements of the user, Erasable and Reprogrammble Read Only Memories - EPROM chips are available.

The programmer consists of 8255 PPI chip, the memory chip to be programmed and a push button switch to connect a + 25V or + 5V supply to the program pin of 2716. It is shown in Fig. 3. The developed programmer unit has a Euro edge connector for connecting the unit with the SDK 85 microprocessor kit during programming and program reading [5].

The flow chart of the monitor listing to enable carrying out programming and reading the EPROM is given in figs. 4 and 5 respectively.

PROGRAMMING

For programming the following steps are followed in sequence.

1. + 5V is applied to Pin No. 24 of 2716
2. + 25 V is applied to Pin No.21 of 2716
3. Load the program in RAM (8155) from starting address 2000
4. Load the informations (to be blown into 2716) in RAM location starting from 8000
5. Execute the program (Press, Go, 2000, EXE)
6. Remove + 25V supply
7. Remove + 5V supply

Now the information has been transferred from RAM to EPROM and stored.

READING THE INFORMATION FROM EPROM

After programming, it is necessary to verify the contents of EPROM. For this the following steps should be followed in sequence.

1. Apply + 5V to Pin No. 24 of 2716
2. Apply + 5V to Pin No. 21 of 2716
3. Load the program in RAM from starting address 2050
4. Execute the program (Go, 2050, EXE)
5. Now the address will appear at the address field and the data will appear at the data field of the display.

The software in the read mode provides facility to either increment or decrement the address by the use of separate incrementing (Next) and decrementing (EXE) keys, so as to enable forward and backward search of the stored data in EPROM.

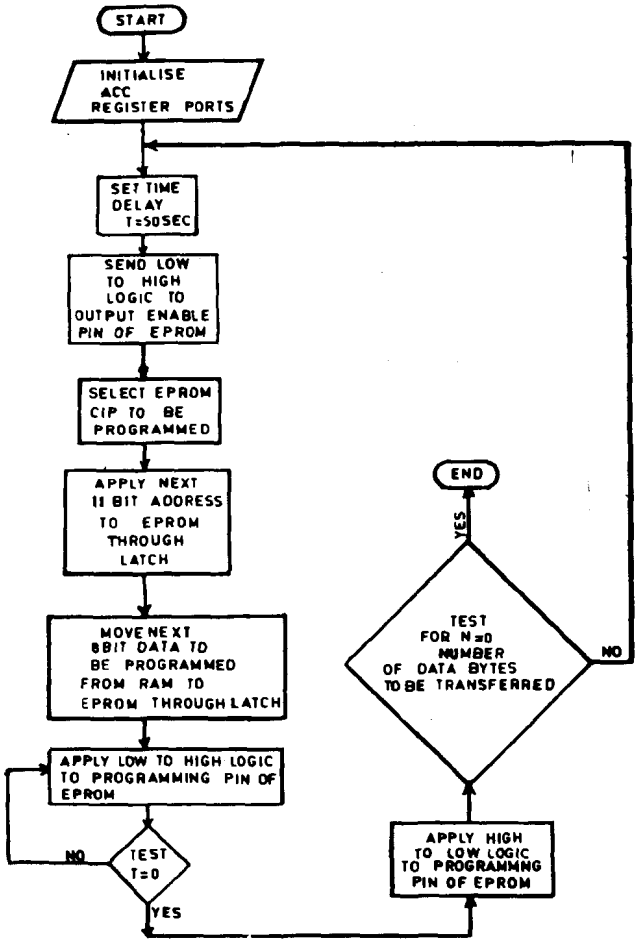

Fig.4 Flow diagram for programming

The EPROM programmer is easy to construct. The circuit can be extended for programming of multiple 2716 EPROMs in parallel and also for loading different data in the different chips as well. In the latter case except for CE/PGM all like inputs of the parallel 2716s may be common. A TTL level program pulse applied to the 2716s CE/PGM input with VPP at 25V will program a particular chip, while a low level CE/PGM input inhibits all other 2716 from being programmed. Using this EPROM programmer, a microprocessor based monitor and testing systems for batteries has been designed and fabricated.

REFERENCES

1. A P Mathur, *Introduction to microprocessors,* Tata-McGraw Hill (1985)
2. A.P. Malvino, *Digital Computer electronics,* Tata-McGraw Hill (1985)
3. *MCS-85 (Intel) User's Manual,* September 1978
4. *8085 Microprocessor and the common peripheral LSIs that are compatible with 8085,* IISc, Bangalore
5. *SDK System design kit user's manual*

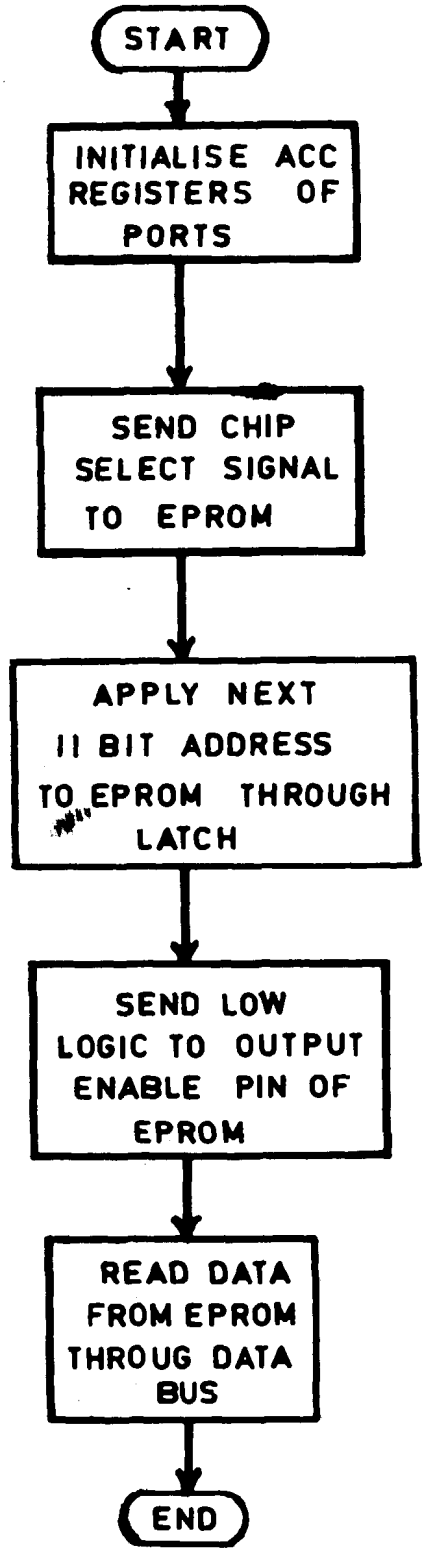

Fig.5: Flow diagram for program reading

EPROM PROGRAMMER

Software

Locations	*OP Code*	*Source Statement*
2000	31 C2 20	Initialise stack pointer
2003	21 X_2X_1 Y_2Y_1	Initialise HL register pair X_2 X_1 Low Address byte } of Y_2 Y_1 High Address byte } RAM from where data is to be transferred to EPROM
2006	11 A_2 A_1 OB_1	Initialise DE register pair A_2A_1 Low Address byte } of OB_1 High Address byte } EPROM to where DATA is to be transferred B_1 is 0 to 7
2009	01 M_1 M_2 ON_2	Initialise BC register pair M_1 M_2 DATA IN HEX } ON_2 DATA IN HEX × } No. of bytes to be transferred N_2 is 0 to 7
200C	3E 80	MVI, A - 80 is the code number for making port A, port B and port C as output ports in zero mode
200E	D3 83	83 is the address of command register of 8255
2010	7A	MOV A,D
2011	E6 07	ANI to unmask page numbers
2013	F6 08	ORI to give High Logic to output Enable pin of EPROM
2015	32 A0 20	STA, store this information at RAM Location 20 A0 for later use
2018	D3 82	OUT 82
201A	7B	MOV A,E
201B	D3 81	OUT 81
201D	7E	MOV A,M - Move Data to be programmed to ACC
201E	D3 80	
2020	3A A0 20	LDA, Retrieve the data stored at 20A0
2023	F6 10	ORI
2025	D3 82	OUT 82, Gives a Highlogic to programming pin of EPROM
2027	D5	PUSH D
2028	11 00 10	Initialise DE register pair with 00, 10 D = 10 E = 00
202B	CD F1 05	Delay subroutine in SDK-85 monitor
202E	D1	POP D
202F	3A A020	LDA, Load ACC with the contents of 20A0
2032	D3 82	OUT 82, one data is now programmed at EPROM
2034	23	INXH
2035	13	INX D
2036	OB	DCX B
2037	78	MOV A,B
2038	B1	ORA, C
2039	C2 10 20	JNZ, not all datas stored Jump to 2010
203 C	CF	RST 1 DISPLAY 8085 on the SDK 85 display to indicate that programming is over.

Software for reading the programmed data

Locations	*OP Code*	*Source statement*
2050	31 C2 20	Initialise stack pointer
2053	3E 08	Unmask RST 5.5
2055	30	SIM
2056	3E 90	Make port A of 8255 as input port and ports B & C as output ports
2058	D3 83	OUT 83
20 5A	11 X_1 X_2 Y_1 Y_2	INITIALISE DE Register pair X_1X_2 Low address byte } of EPROM Y_1Y_2 High address byte } from (page number) where contents are to be read
20 5D	7B	MOV A,E
20 5E	D3 81	OUT 81
2060	7A	MOV A,D
2061	E6 07	ANI, TO unmask page number and to apply
2063	D3 82	Low logic to output Enable pin of EPROM
2065	DB 80	IN 80
2067	D5	PUSH D
2068	CD 6E 03	CALL UPDDT
206B	D1	POPD
206C	D5	PUSH D
206D	CD 6303	CALL UPDAD
2070	FB	EI
2071	CD E7 02	RDKBD
2074	D1	POPD
2075	FE 11	CPI (Next Button)
2077	C2 7B 20	JNZ
20 7A	13	INX D
20 7B	FE 10	CPI (EXEC Button)
20 7D	C2 5D 20	JNZ
20 80	1B	DCX D
2081	C3 5D 20	