PROGRAMMING EPROM's WITH A SMALL COMPUTER

Hardware-software system interfaces with computer for easy, sophisticated programming from keyboard, cassette or disk

BY JOHN DOOLITTLE AND SLOBODAN TKALCEVIC

Part 1

WHY pass up advanced digital construction projects because they involve EPROM (Erasable Programmable Read Only Memory) programming? If a small computer such as the TRS-80 is available, then the EPROM programmer described here can expand your digital design work inexpensively. This modest-cost peripheral device can program EPROMs with data loaded from a keyboard, cassette, or diskette. Its other operating modes verify programmed EPROMs against the programming buffer, list the EPROM or programming buffer contents to the monitor screen, copy the contents of one EPROM to another, and check that an EPROM is completely erased before re-programming.

EPROMs are most commonly used for storing machine-language programs to control microprocessors or for storing look-up tables where a large number of inputs and outputs are needed (for example, converting one binary

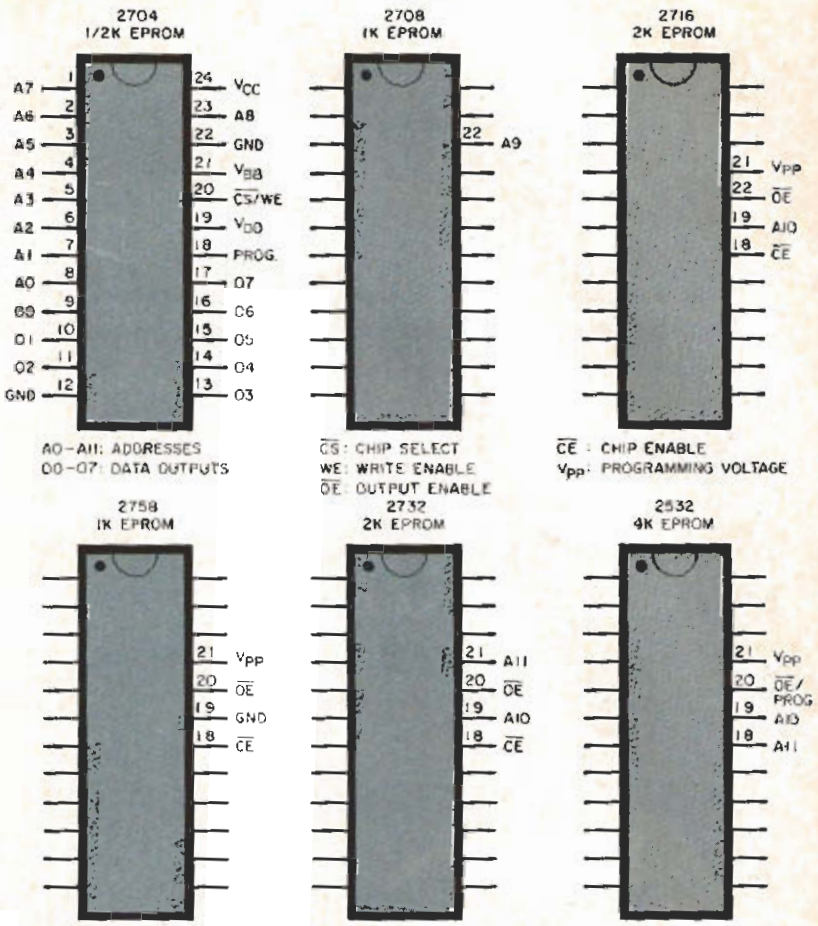
code to another, defining mathematical functions, or controlling sequential circuits). Usually, microprocessor control programs that are to be stored in EPROM are written in assembly language and then converted into machine-compatible binary code. While this can be done by manually resolving the necessary op-codes and addresses, a more efficient approach is to use a small computer that supports a text editor and assembler. Once the assembly language text is entered and assembled into binary code, it can then be written directly to EPROM using the peripheral programmer.

An EPROM programmer interfaced with a computer benefits from system software support (such as an editor or assembler), and can use the logic and computational resources of the computer to control its operation. By making address generation and programming-pulse sequencing into software tasks, the hardware component count can be reduced to a bare minimum.

This EPROM programmer can program 2704-, 2708-, 2758-, 2716-, 2732-, and 2532-type EPROMs, giving the user a selection of ½K, 1K, 2K, or 4K by 8-bit devices. The design includes five power supplies to provide the voltages neeessary to program all these memory chips. While the printed circuit directly interfaces with the TRS-80 Model I expansion bus using a 40-conductor ribbon cable, its design can be easily adapted for any computer that allows access to the CPU data and address lines.

Working with EPROMs. There are basically only two operating modes for EPROMS: Read and Program. Most EPROMs are byte-addressable, which means that 8 bits of memory contents can be accessed by specifying a unique byte address. The data lines remain electrically disconnected, or "three-stated," until the chip is enabled. This allows several memories or I/O devices to share the same data bus in turn. Once enabled by supplying a *chip select* signal, data may be read from the device simply by providing the desired address. Valid data becomes available on the output lines a short time after the address is specified. This brief delay, or "access time," allows the address lines to settle to constant values.

Before programming, all bit locations of an EPROM are in the one state as a result of exposure to ultraviolet light. Data is placed in the device by entering zeros at the required bit locations of any byte addressed. The Program mode is activated by supplying a write-enable signal and a programming pulse



EPROM	READ				PROGRAM			
PIN	18	19	20	21	18	19	20	21
2704, 2708	0	12	0	−5	25.5 ⎍	12	12	−5
2758, 2716	0	A10	0	5	5 ⎍	A10	5	25.5
2732	0	A10	0	A11	5 ⎍	A10	25.5	A11
2532	A11	A10	0	0	A11	A10	5 ⎍	25.5

Fig. 1. Pin designations and programming signals for the EPROM chips that are programmable with this project.

to the appropriate pins. The duration and amplitude of the programming pulse must be within specified limits. Some EPROMs (2758, 2716, 2732, and 2532) can be programmed in one pass because a relatively long pulse is supplied for each byte addressed. Other types (2704 and 2708) demand that a short programming pulse be repeated many times with a delay inserted between pulses. (This allows for adequate heat dissipation.) Whatever the EPROM type, it is normal to make many passes through the entire memory until each affected bit location has been pulsed for a total time exceeding a specified minimum.

Verification after programming is done by reading back the contents of all programmed bit locations and comparing them with the original data. If modifications to the EPROM data are deemed necessary, and involve changing a 1 to a 0, they are easily made by reprogramming with the altered data. Of course, corrections that involve changing a 0 to a 1 can only be made by first erasing the EPROM.

All of the EPROMs considered here are 24-pin DIP packages which have identical pin configurations except for the assignment of pins 18 through 21. The pin designations and programming signals for all chips programmable with

TABLE I-PORT ADDRESSES	
8255 Port	CPU I/O Port Address
A	F8H
B	F9H
C	FAH
Control	FBH

the EPROM programmer are shown in Fig. 1. The similarity in pinouts means that a general circuit can be designed to accommodate the various types with only four lines being unique to each EPROM. The connection of these four lines can be made most conveniently by routing the affected signals through "personality" modules wired specifically for each EPROM type.

Circuit Operation. The schematic diagram of the EPROM programmer appears in Figs. 2 and 3. Computer interfacing is greatly simplified by designing the circuit around a general-purpose programmable I/O device such as the Intel 8255A Programmable Peripheral Interface (PPI), shown logically in Fig. 4. This chip communicates with the CPU through the eight lines of the data bus and provides for 24 I/O lines that can be connected to the computer data bus under program control.

PARTS LIST

C1,C2—1000-μF, 16-V electrolytic
C3—1000-μF, 50-V electrolytic
C4—22-μF, 15-V electrolytic
C5,C6,C8—0.1-μF, 20-V disc ceramic
C7—0.1-μF, 50-V disc ceramic
D1 through D6—1N4004 diode
F1—0.5-A fuse and holder
IC1—8255 programmable peripheral interface
IC2—74LS30 8-input NAND gate
IC3—74LS32 quad 2-input OR gate
IC4—74LS04 hex inverter
IC5,IC6—75451 dual peripheral AND driver
IC7,IC8—723 precision voltage regulator
IC9—LM320-5 negative 5-V regulator
IC10—LM340-5 positive 5-V regulator
IC11—LM340-12 positive 12-V regulator
K1—24-V, 700-ohm 4pdt relay (Potter-Brumfield R10-E1-W4-V700, with socket 27E213, or similar)
LED1—Green light-emitting diode
LED2,LED3—Red light-emitting diode
The following are ¼-W, 5% resistors unless otherwise noted:
R1,R4,R6,R7,R10,R17—2200 ohms
R2,R11—10,000 ohms
R3—5600 ohms
R5,R9—10-ohm, ½-W
R8,R12—10,000-ohm potentiometer
R13—220 ohms
R14,R15,R16—330 ohms
S1—Spst switch
SO1—24-pin, zero-insertion-force socket (Textool 224-3344 or similar)
T1—10-20 CT-40 CT, 300-mA transformer (Triad F-9IX)
Misc.—Four 16-pin DIP header sockets with covers, suitable enclosure, 40-conductor ribbon cable with connectors, line cord, mounting hardware, etc.

Note: The following are available from Parhelion Inc., Box 3602, Stanford, CA 95405: complete kit including drilled and etched pc board, enclosure, and driver program listing for $134.95, plus $5 for postage and handling. Also available separately are an etched and drilled pc board for $21.95, plus $2 for postage and handling; driver program listing for $10.95; driver program on cassette for $19.95, plus $1 for postage and handling; driver program on diskette for $21.95, plus $1 for postage and handling. California residents, please add 6½% sales tax.

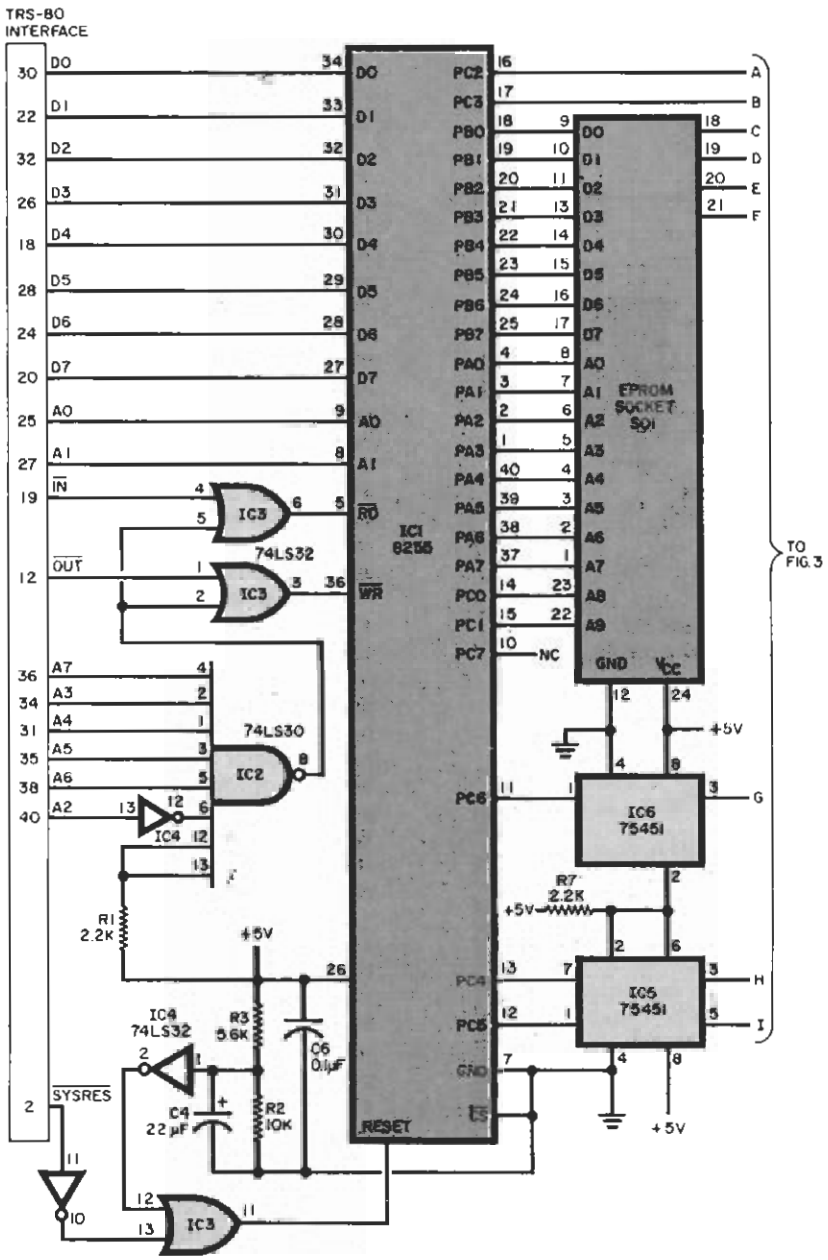


Fig. 2. Interfacing to the computer is provided by a general-purpose programmable I/O device, the 8255A.

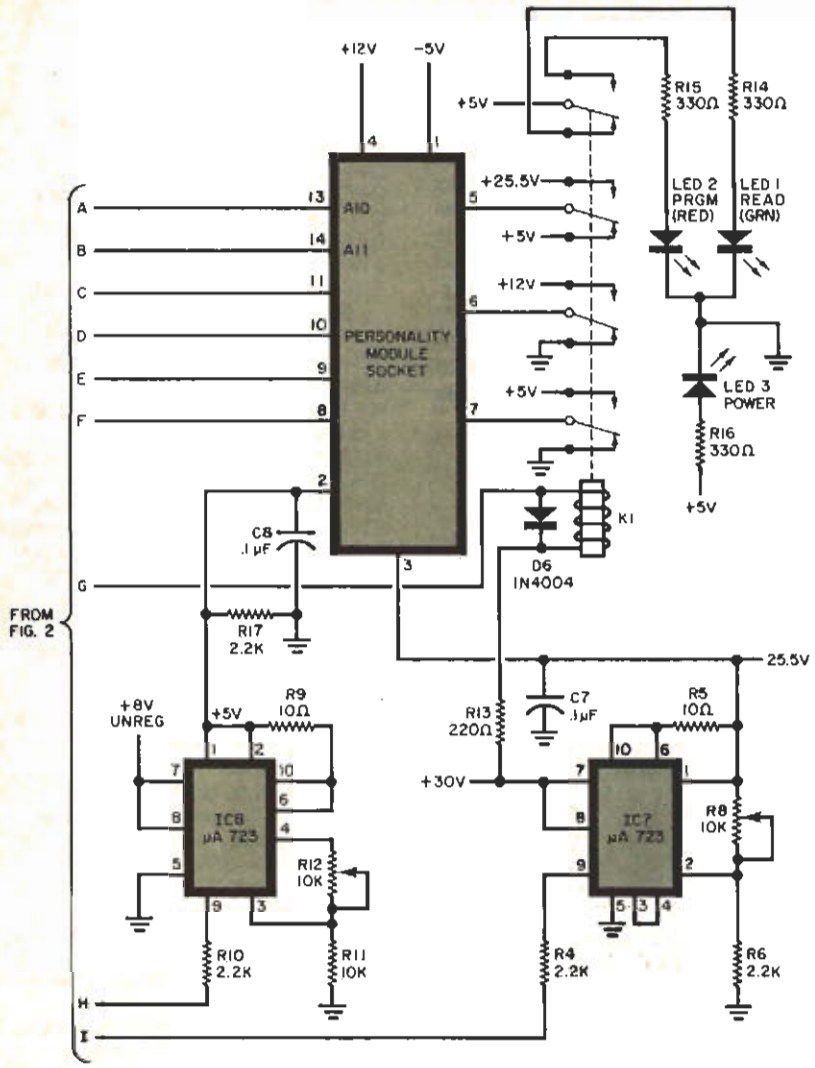


Fig. 3. Precision voltage regulators IC8 and IC9 are used to supply 5- and 25.5-V programming pulses.

TABLE II—CONTROL BYTE FUNCTIONS

Control byte	8255 response	Function
00	reset PC0	reset EPROM add. A8
01	set PC0	set EPROM add. A8
02	reset PC1	reset EPROM add. A9
03	set PC1	set EPROM add. A9
04	reset PC2	reset EPROM add. A10
05	set PC2	set EPROM add. A10
06	reset PC3	reset EPROM add. A11
07	set PC3	set EPROM add. A11
08	reset PC4	turn off 25.5V
09	set PC4	turn on 25.5V
0A	reset PC5	turn off 5V
0B	set PC5	turn on 5V
0C	reset PC6	relay PROGRAM mode
0D	set PC6	relay READ mode
0E	reset PC7	not used
0F	set PC7	not used
80	all 8255 ports in output mode	PROGRAM mode
82	8255 ports A&C output, B input	READ mode

In this application, the 8255A's I/O lines are organized into two eight-bit ports (A and B) and a special port (C) with eight lines that can be set or reset independently—without disturbing the other bits of the port. The lower half of port C is used in conjunction with port A to supply up to 12 bits of address information to the EPROM from the CPU data bus. The upper half of port C controls relay *K1* and the switchable power supplies during the programming sequence.

Port B is used for data transfer between the EPROM and the CPU. An additional 8225A port (Control) receives a control byte which configures separately ports A, B, and C for input or output as required. The 8255A remains in a specified configuration until a new control byte is sent by the computer.

Connection of the A, B, C, or Control ports to the computer data bus is accomplished by using "port-addressed I/O." Computer address lines A0 and A1 supply the Port Select signals while addresses A2 through A7 are combined through 8-input NAND gate *IC2*, allowing the $\overline{IN}$ and $\overline{OUT}$ control signals from the TRS-80 to pass to *IC1*. Address line A2 is inverted by an element within *IC4* so that unique I/O ports can be specified without conflicting with the TRS-80 assigned port address of FFH for the cassette recorder. The resulting TRS-80 I/O port addresses for the 8255A's A, B, C, and Control ports are listed in Table I.

When power is first applied, the 8255A automatically resets via the $\overline{SYSRES}$ signal, causing all ports initially to be in the input mode. The reset condition is maintained for about 200 ms by *R2* and *C4*. The 8255A will also be reset whenever the TRS-80 system reset button is operated. This initializes the unit to a known state and protects any EPROM left in the programming socket from inadvertent pulses which might result from an undefined initial state.

As shown in Fig. 5, supply voltages are provided so that several EPROM types are programmable with this circuit. Constant −5, 5, and 12 V are delivered by three-terminal regulators, while 5- and 25.5-V programming pulses, which must be switched under precise control of the software, are supplied by using precision voltage regulators *IC7* and *IC8* (Fig. 3). Frequency compensation (pin 9) is connected through analog switch *IC5* to ground. When a logic 1 is supplied by the 8255A control line (PC4 or PC5), the compensation is allowed to float, turning on the precision regulator. A logic ZERO

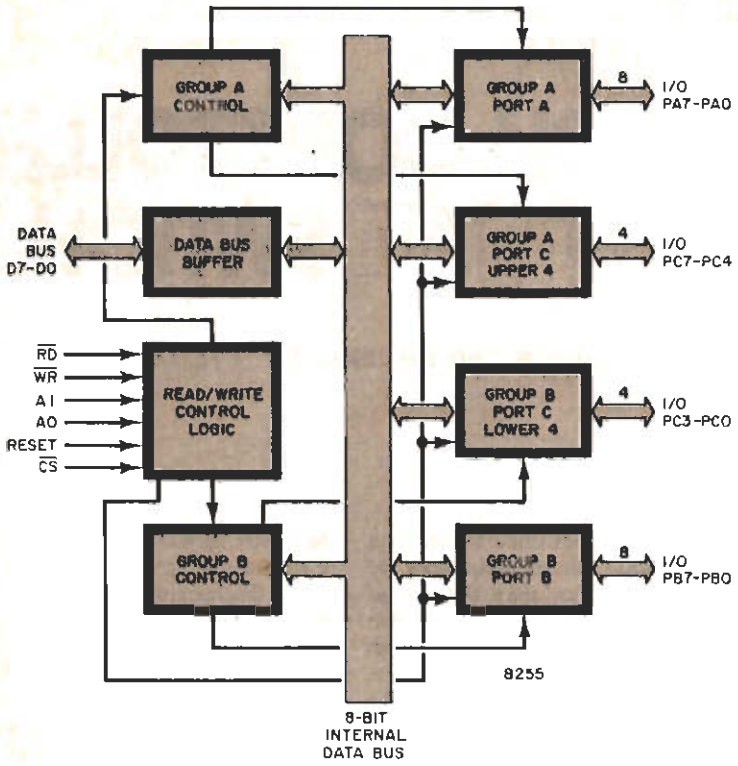


Fig. 4. Internal logic arrangement of the Intel 8255A.

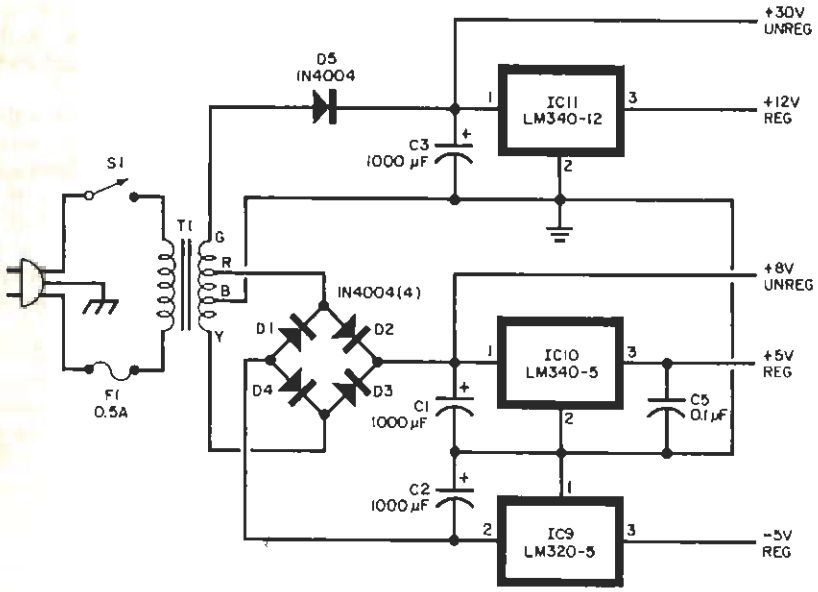


Fig. 5. Power supply provides several voltages.

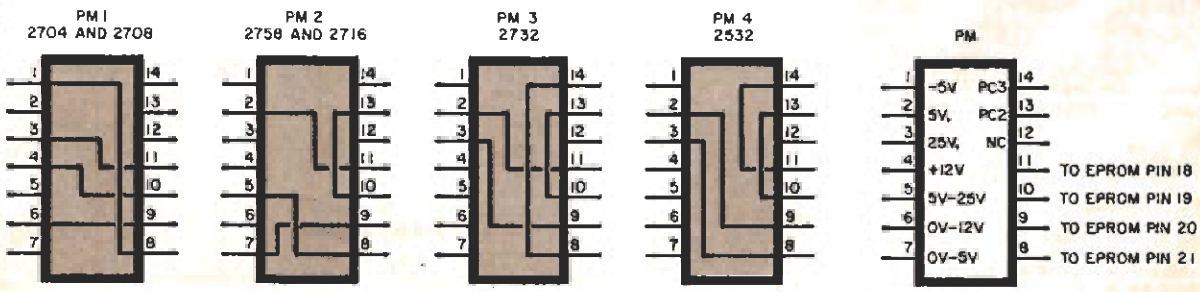


Fig. 6. Wiring diagrams for personality modules to accommodate the various types of EPROMS.

causes the compensation pin to be grounded, turning the precision regulator off.

Input to the 5- and −5-V regulators is taken from a full-wave rectifier (*D1* through *D4* in Fig. 5), driven by the lower half of the power transformer's secondary winding. The center-tap of that section is grounded, causing the bridge to give both positive and negative outputs of about 8 V. A half-wave rectified input to the 12- and 25.5-V regulators is supplied at about 30 V by the top three quarters of the transformer secondary winding. (All currents are well below the maximum allowed for regulator operation without heat sinks.)

The voltages supplied to some of the EPROM pins depend on whether the device is in the Read or Program mode. Switching between two voltages is done in this case by relay *K1* (Fig. 3), which is activated when a 0 on the 8255A control line (PC6) causes current to flow from the 30-V unregulated supply through voltage-dropping resistor *R13*, the relay coil, and analog switch *IC6* to ground. Diode *D6* protects *IC6* from voltage spikes caused by relay switching. The relay also switches the *LED1* (READ) and *LED2* (PROGRAM) mode indicators.

Personality Modules. Connections to pins 18 through 21 of the different EPROMs are unique to each type and must be routed through appropriate "personality" modules. Wiring diagrams for these personality modules are shown in Fig. 6. These can be made using standard 14-pin headers that will plug directly into the personality module socket of Fig. 3.

In Part 2 of this article, we will present construction plans and software information for the EPROM programmer. ◇