Programmer for fusible-link bipolar PROMs

The problem: How do I configure an array of memory chips of differing capacities without resorting to multiple decoders and a rat's nest of jumpering? Or, how would I get a 7-segment display to show more than the usual 0-9 digit information?

The solution: A bipolar PROM! They're fast. They're cheap. They're easy to program. They're cheap. They don't take up a lot of expensive pc board real estate. And they're cheap!

Chuck Simmers
Field Application Engineer, Intel Corp., NSW

HAVE YOU EVER been designing a project and wished you had a custom decoder or simple PLA (programmable logic array) to reduce the chip count and make the circuit less complex? How about using a seven-segment LED display to show more than the usual zero-to-nine digit information? All these applications and many more can be addressed by one versatile integrated circuit called the *Schottky Bipolar Programmable Read Only Memory*.

They are produced by many manufacturers in pin-compatible forms and are widely available through normal sources. Although they range in sizes up to 4K x 8 (ten inputs, eight outputs), for our purposes we will be discussing only the smallest one, the 32 x 8 (five inputs, eight outputs), known under the generic number of 74LS188/288. Table 1 gives a cross reference of the more popular, pin-compatible parts. All have similar or identical specifications.

The bipolar PROM programmer described in this article is a by-product of a micro-controller design to be presented in later issues of ETI.

Bipolar PROMs — a lot more than just a memory chip

Pin-out and logic diagram are given in Figure 1. Some suggested uses are shown in Figure 2. Figure 2(a) illustrates the original application of a memory decoder. With the appropriate program in PROM, the memory banks can be any size from 2048 to 32K bytes, in any mixture, and may be located on any 2K boundary.

Figure 2(b) shows the PROM used as a LED segment driver. With 32 different characters, these PROMs can be cascaded to produce custom 14 and 16-segment 'star-burst' display drivers.

The device is used as a liquid crystal display driver in Figure 2(c) wth address line four and the LCD's backplane being driven from a low frequency oscillator. The lower half of the PROM is programmed with 'positive' segment data while the upper half is 'negative'. The result is a 180° phase relationship between the backplane and the decoded segment information.

As in Figure 2(d), any eight different functions, based upon any 32 input conditions, or the states of five inputs, can be realised with these chips. Keep in mind the 25-35 ns propagation delay time when designing your circuits and calculate this into any timing equations.

These Schottky PROM memories are organised as 32 words by eight bits. An enable input is provided to control the output states. When the enable input is in the low state, the outputs present the contents of the selected word. If the enable input is raised to the high level, it causes all eight outputs to go to the 'off' or high impedance state.

Since these devices are of a Schottky-clamped type, they have very fast access times, in the range of 25-35 ns maximum.

They are available in both open-collector (74LS188) and totem-pole versions (74LS288). PROMs are shipped from the distributor with *lows* in all locations. A high may be programmed into any selected location by using the following programming instructions. (NOTE: Once programmed, it is impossible to go back to a low state.)

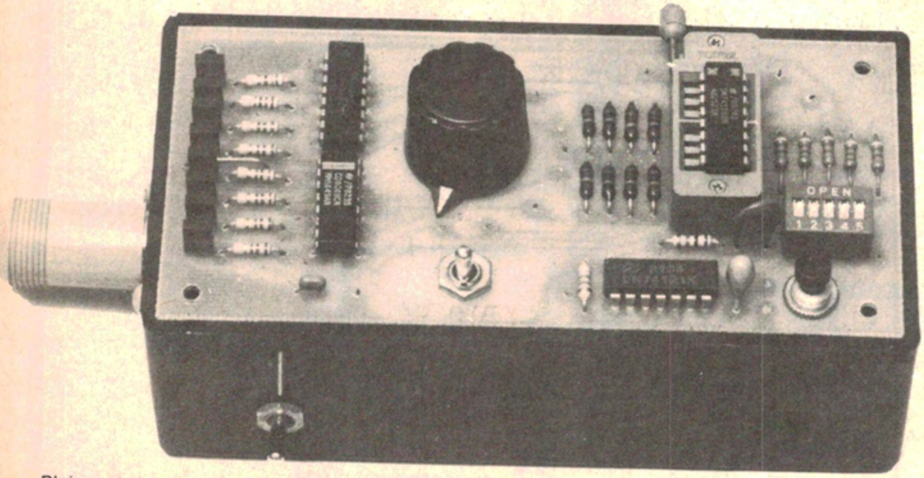
In order to generate a high level on the outputs, the PROMs must be programmed in the following manner:

1. Address/Inputs and chip-enable pins must be driven from normal TTL logic levels during both programming and verification.

2. Programming will occur at a selected address when Vcc is held at 10.5 V, the appropriate output is held at 10.5 V, and the chip is enabled. To achieve these conditions and the appropriate sequence, the following procedure must be followed:

(a) Select the desired word by applying a high or low level to the address/inputs. Disable the chip by taking the enable input to a high level.

(b) Increase Vcc to 10.5 V. Since Vcc supplies the current to program the fuse as well as the Icc of the device at programming voltage, it must be capable of supplying 750 mA at 11.0 V.



Plain and simple. No, it doesn't come in a spray can, but you'll find it very useful, nonetheless! The pc board for the programmer simply replaces the lid on a small jiffy box.

(c) Select the output where a high level is desired by raising that output pin to 10.5 V. It is critical that only one output at a time be programmed since the internal circuitry can only supply programming current to one bit at a time. Outputs not being programmed must be left floating or tied to a high impedance source of at lease 20k Ohm.

(d) Enable the device by taking the chip enable to a low level for a duration of 10 us.

(e) Verify that the bit is programmed by first removing the programming voltage of 10.5 V and returning Vcc to 5 V. The chip must be enabled to sense the states of the outputs. Steps (b), (c) and (d) are repeated about ten times or until the output is verified.

(f) Following verification, apply about five more programming pulses to the bit to ensure that the fuse is completely blown.

(g) Repeat steps (a) through (f) for each bit to be programmed to a high level. After all selected bits are programmed, verification of the entire contents of the memory should be performed.

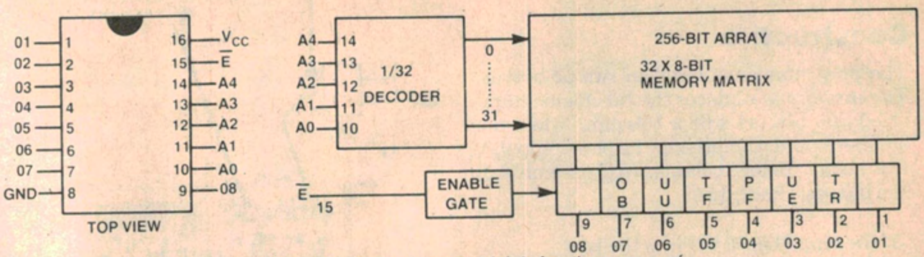


Figure 1. Standard pinout (left) and logic diagram (right) for the range of pin-compatible bipolar PROMs listed in Table 1.

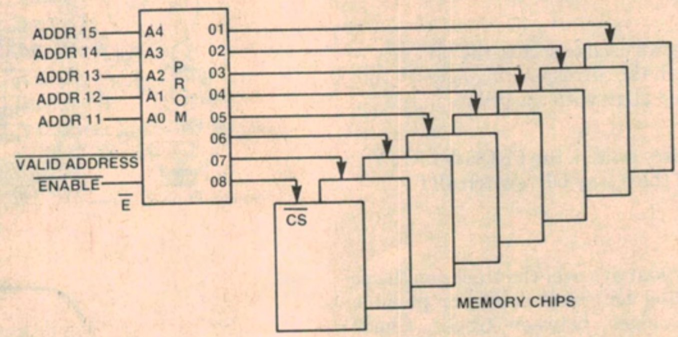


Figure 2(a). This project was designed so that I could blow a PROM for the application illustrated here — a memory address decoder.

TABLE 1.

Pin-compatible fusible-link bipolar PROM types

Type No.	Speed	Manufacturer
74S188	22 ns	National Semiconductor
74S288	22 ns	National Semiconductor
77S188	12 ns	National Semiconductor
77S288	12 ns	National Semiconductor
27S18		A.M.D.
27S19		A.M.D.
6330-1		M.M.I.
6331-1		M.M.I.
82S23		Signetics
82S123		Signetics

The circuit

Two voltage regulators provide the appropriate programming and verifying voltages needed. The ground of one of the regulators is 'lifted' about 5 V above ground to obtain the 10 V for programming. SW1 is a 4PDT type and is used to switch between programming and verifying modes. PB1, a momentary-action normally-open pushbutton, enables the one-shot 74121 to enable the chip for about 10 us when in 'PROG' mode.

A 9-position rotary switch (SW3), is used to select the output that is to be set high. Replacing this switch with a bank of single-pole switches is not recommended because of the possibility, or temptation, to blow more than one fuse at a time.

SW4 through SW8 are the address/input select switches. The two CMOS 4049 hex buffers were chosen for output drivers for their ability to tolerate input voltages up to 15 V with a 5 V Vcc supply.

Lastly, the LEDs are used for visual display and verification of the PROM contents after programming. Simple, right?

The circuit could easily be expanded to accommodate larger PROMs, but I think I would be inclined to explore a more automatic approach by adding ports and driving transistors to a microprocessor-based system.

A4	A3	A2	A1	A0	a 07	b 06	c 05	d 04	e 03	f 02	g 01	DISPLAY
0	0	0	0	0	1	1	1	1	1	1	0	0
0	0	0	0	1	0	1	1	0	0	0	0	1
0	0	0	1	0	1	1	0	1	1	0	1	2
0	0	0	1	1	1	1	1	1	0	0	1	3
0	0	1	0	0	0	1	1	0	0	1	1	4
0	0	1	0	1	1	0	1	1	0	1	1	5
0	0	1	1	0	1	0	1	1	1	1	1	6
0	0	1	1	1	1	1	1	0	0	0	0	7
0	1	0	0	0	1	1	1	1	1	1	1	8
0	1	0	0	1	1	1	1	1	0	1	1	9
0	1	0	1	0	1	1	1	0	1	1	1	A
0	1	0	1	1	0	0	1	1	1	1	1	b
0	1	1	0	0	1	0	0	1	1	1	0	C
0	1	1	0	1	0	1	1	1	1	0	1	d
0	1	1	1	0	1	0	0	1	1	1	1	E
0	1	1	1	1	1	0	0	0	1	1	1	F
1	0	0	0	0	0	0	0	0	0	0	1	—
1	0	0	0	1	1	0	0	1	1	1	1	

TABLE 2. Logic table for the application illustrated in Figure 2(c). Note that, beyond address 10000, the display complement is generated.

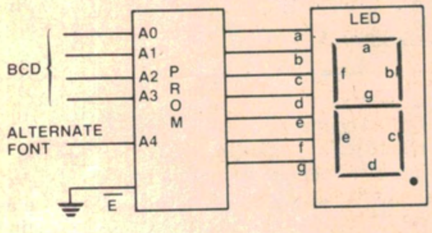


Figure 2(b). A 7-segment LED display driver. You can get more than just 0-9 you know!

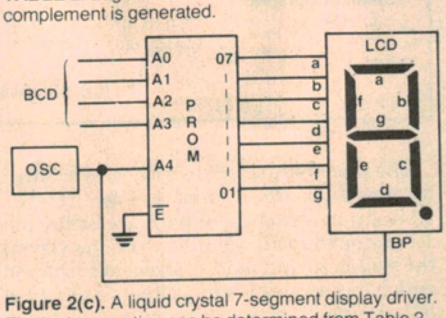


Figure 2(c). A liquid crystal 7-segment display driver. The logic operation can be determined from Table 2.

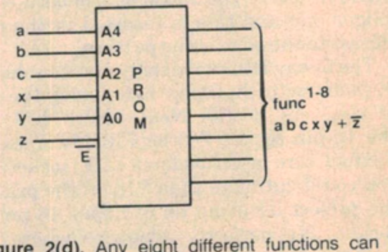


Figure 2(d). Any eight different functions can be realised, based on any 32 different input conditions or the states of five inputs.

Construction

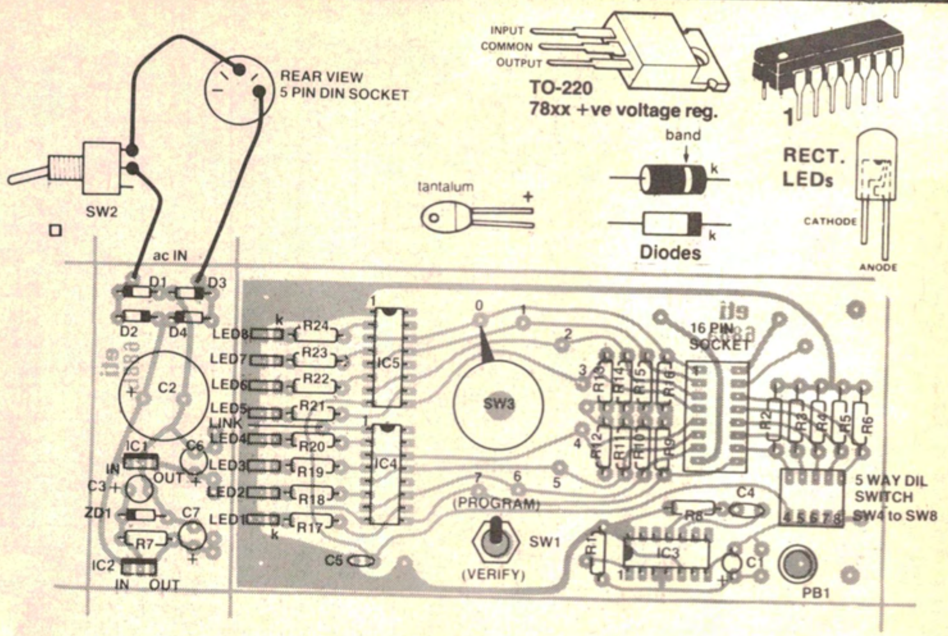
The programmer is built on two pc boards. The larger one replaces the lid of a medium-sized jiffy box (41 x 68 x 130 mm), while the smaller board (comprising the power supply), fits inside. Before commencing assembly of the boards, check that:

(i) the small board fits into the box vertically;

(ii) the larger board fits in place of the box's lid;

(iii) the screws used to secure the lid pass through the corresponding holes in the pc board and align with the pillars in the box; and

(iv) the rotary switch, the PROG/VERIFY switch and the 5-way DIL switch all fit correctly.



Also run your eye over the tracks on the pc board looking for small cracks or possible shorting 'bridges' between tracks. Check also that the component holes are all drilled and the correct size.

Build the smaller board first. There are two alternatives to choose from here, depending on whether you have an ac or a dc plugpack. I used a 12 Vdc/1 A plugpack and did not install the four rectifier diodes. As the plugpack I used powers the office MicroBee, which has a 5-pin DIN plug output, I installed a 5-pin DIN socket on the end of the programmer's case and wired pin 1 via a small toggle switch to the diode cathode pad nearest the 2200 uF filter capacitor (C2). This switch was mounted on the side of the jiffy box, about 10 mm up from the bottom and 20 mm in from the end with the DIN socket.

If you are using an ac plugpack or an externally-mounted transformer, you should install the four diodes and wire the ac input via a switch, as shown in the overlay and wiring diagram.

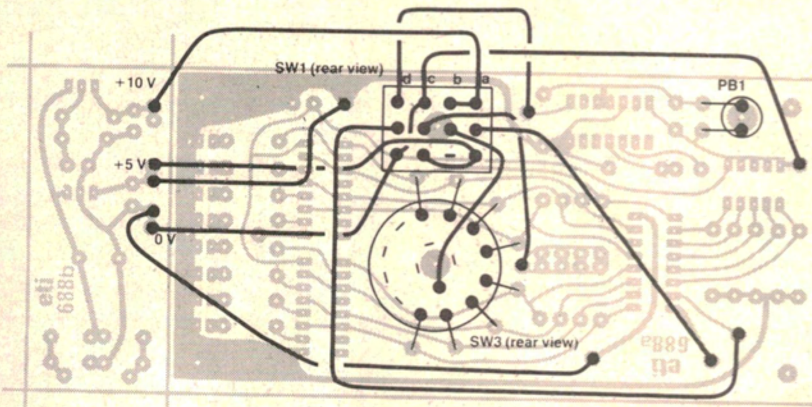
The components can be assembled on the power supply board in any order, just take care that they are correctly oriented.

When you've completed it, apply power and check that the +5 V and +10 V rails are present (measured with respect to the 0 V rail).

The large pc board can be tackled next. Start by installing the link between the positions of R20 and R21. Install all the resistors and capacitors next (make sure you get C1 the right way round).

Follow by soldering the eight rectangular LEDs in place. The anode lead of each is the longer one and this is inserted in the hole closest to the edge of the pc board.

The 5-way DIL switch may now be soldered in place with the 'open' side facing the row of resistors R2-R6. Next install IC3 and the 16-pin socket for the PROMs. I used a Textool 'zero insertion force' (ZIF) socket, but you could cut more than $10 off the price of the project by using an ordinary 16-pin IC socket. It's perfectly adequate where the programmer is not in everyday or every-week use.



688a AND 688b VIEW OF COPPER SIDE

PARTS LIST — ETI-688

Resistors	all ¼ W, 5%
R1	47k
R2-6	4k7
R7	1k
R8	10k
R9-16	22k
R17-24	220R
Capacitors	
C1	10u/25 V RB electro.
C2	2200u/25 V RB electro.
C3, 6, 7	4u7/10 V tant.
C4	1n5 greencap
C5	100n ceramic or greencap
Semiconductors	
D1-D4	1N4001, 1N4002, EM401, EM402 etc
IC1, IC2	uA7805, LM7805
IC3	74121
IC4, IC5	4049
LED1-8	rectangular red LEDs
ZD1	5V1/1 W zener
Miscellaneous	
PB1	miniature momentary action pushbutton
SW1	three-pole, two position min. toggle switch
SW2	DPDT min. toggle switch
SW3	single pole, 12-position C&K 'Lorlin' rotary switch, set for 8-pos. operation.

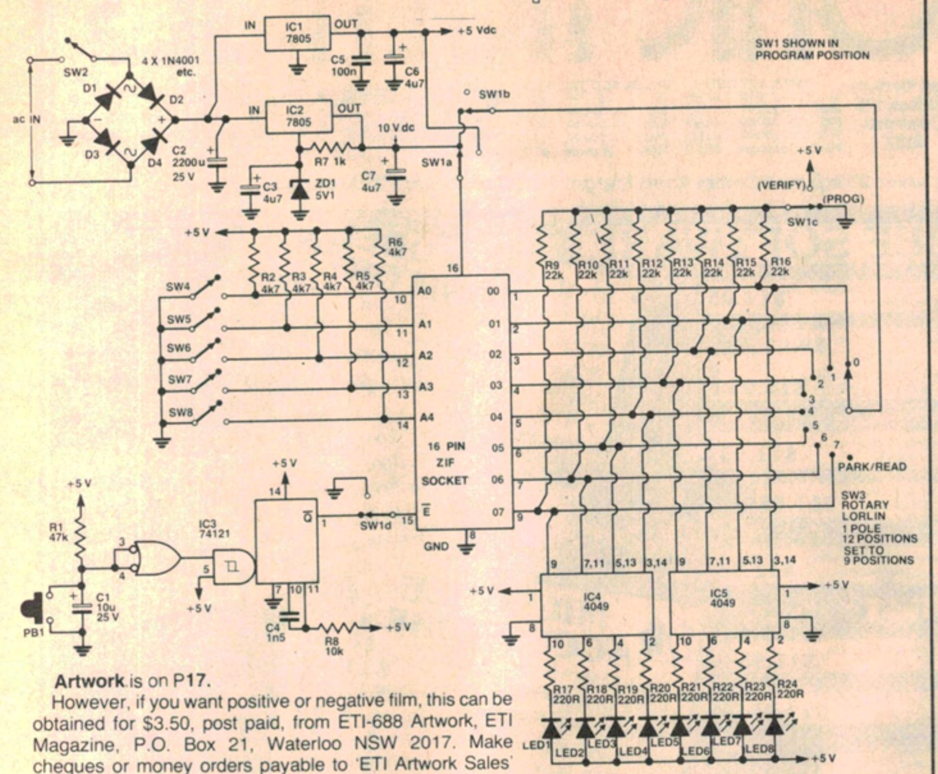
ETI-688 pc board; zippy box 48 x 68 x 130 mm; 16-pin ZIF socket; ac plugpack to suit (see text) etc.

Price estimate $38 — $42

The rotary switch, SW3 can now be screwed to the pc board. I used a C&K 'Lorlin' 12-position switch here with the index pin set to stop the action at nine positions. Orient the switch so that pin 1 is adjacent to the pad which has a track leading to pins 3 and 14 of IC5. Using eight short lengths of 22 guage tinned copper wire, wire each of the contacts, one to eight, to the respective pads on the pc board, as per the underside wiring diagram.

Now install the pushbutton, PB1 and wire its contacts to the board with tinned copper wire too.

Install the PROG/VERIFY switch, SW1,



Artwork is on P17.

However, if you want positive or negative film, this can be obtained for $3.50, post paid, from ETI-688 Artwork, ETI Magazine, P.O. Box 21, Waterloo NSW 2017. Make cheques or money orders payable to 'ETI Artwork Sales' and ensure that you ask for a positive or negative, as required.

HOW IT WORKS — ETI-688

There's not much to it, just power supply, programming pulser and simple readout.

Two supply rails are necessary: +5 Vdc and +10 Vdc. The 5 V rail is provided by a 7805 (IC1) three-terminal regulator from the capacitor-input bridge rectifier (D1-D4, C2). The 10 V supply is provided by 'jacking-up' the reference terminal of another 7805 (IC2) by 5 V using a zener (ZD1).

The programming pulse is provided by triggering a 74121 one-shot multivibrator, IC3. When PB1 is pressed, the NOR gate inputs of IC3 (connected as an inverter), pins 3 and 4, are pulled low and the inverter triggers the following Schmitt trigger, which sets off a 10 microsecond pulse from the Q-bar output of the 74121.

When SW1 is in the PROGRAM position, SW1d couples this pulse to the enable pin of the PROM in the 16-pin PROM socket. Capacitor C1 debounces PB1.

Switches SW4-SW8 pull the address pins of the PROM socket low when closed, resistors R2-R6 pull the address pins high when the switches are open.

Resistors R9-R16 provide pull-ups or pull-downs on the output pins according to how the PROG/VERIFY switch, SW1, is set. The rotary switch, SW3 selects the output line to be programmed. Two 4049 CMOS buffers drive the indicator LEDs which let you know what's happening.

Capacitors C5, C6 and C7 bypass the supply rails and ensure the regulators remain stable.

and tighten it in place. The remaining wiring, to this switch and the power supply board, should be done with plastic insulated, light hookup wire (10 x 0.2 mm). The underside wiring diagram shows the details. Do this wiring carefully and check it when you've finished.

Lastly, install the two CMOS ICs, IC4 and IC5. Don't handle their pins, either use an appropriate IC insertion tool or only handle the devices with your thumb and forefinger grasping the ends of the package. Solder pins 1 and 8 first. Digital CMOS ICs are pretty robust in general and it's unlikely you'll damage them. If you prefer, IC sockets can be used.

Well. All that's left is to do a 'trial run' to see if it works and to get familiar with the procedure.

A typical programming session

A) With all power off, insert a PROM into the programming socket.

B) Set SW1 to the VERIFY position and turn the main power on. At this time all the LEDs should be on, indicating all fuses intact.

C) Set switches SW4 to SW8 to conform to the appropriate input condition.

D) Set switch SW3 to the output to be programmed.

E) Throw SW1 to the PROG position.

F) Touch PB1 a few times to burn the bit. A bit cannot be over-programmed.

G) Put SW1 back into the VERIFY position to check that the bit is in fact programmed, which the LEDs will indicate.

H) Repeat steps C through G until done.

This is rather tedious and boring, and it would become a rather large pain if you were doing more than two or three chips at a time, or if the PROMs being used were of a larger variety. But, I've found 32 bytes isn't too bad. Just get a cup of coffee, find a comfortable chair, and take your time. ●