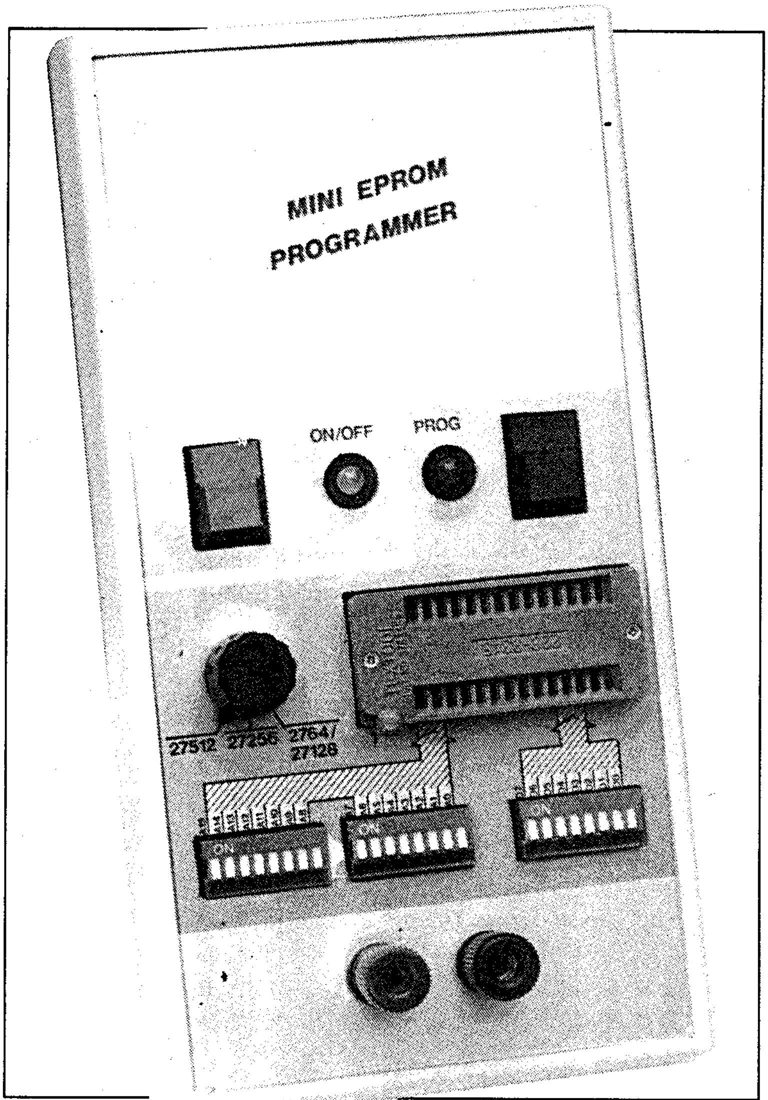
MINI EPROM PROGRAMMER

J. Ruffell

Many constructors have good reasons for preferring a low-cost EPROM programmer with manual data and address to a full-blown programmer operating under computer control. First, they may not have a computer; second, they do not mind spending some time on programming small amounts of data; and third, they object to the expenditure on an instrument that is only occasionally called upon.

The programmer is definitely not intended for loading huge amounts of data into EPROMs. Even if you could manage to program bytes faultlessly at a rate of one per second, it would take more than 18 hours to load all 65,536 bytes (64 Kbytes) in a Type 27512 EPROM, the largest the present programmer can handle. Moreover, it is a common misunderstanding to associate EPROMs with microcomputer systems that require large amounts of data. As has been shown in a number of projects in this magazine over the past few months, there are quite a few occasions where no more than, say, 256 or 512 bytes are involved, such as where EPROMs function as programmable look-up or conversion tables.



Although the circuit diagram of the mini EPROM programmer (Fig. 1) looks crowded, it should be noted that most components go into the power supply to ensure that the high programming voltage is not applied to the EPROM until the 5 V supply voltage is present — the reverse sequence would have disastrous consequences. Also note that an EPROM must never be removed from the working programmer, because the order in which the programming voltage and the supply voltage are removed might just be wrong.

How it works

Although the supply circuit already looks quite crowded in places, it is not complete without an external regulated power supply to furnish the programming voltage plus about 750 mV. The input voltage to the circuit may be adjusted by measuring the programming voltage at point Pv. The programming voltage is determined by the EPROM type and its manufacturer. Regulator IC_3 reduces the programming voltage to 5 V which is used to power the EPROM and the programmer circuit. Transistor T_1 prevents the programming voltage being applied to the EPROM before the 5 V supply voltage. The transistor is controlled by a monostable multivibrator (MMV), IC_{1a}. After this has been triggered, its Q output remains high until it is reset via its $\overline{CLR}$ input. This input is connected to the rest of the circuit in a manner to ensure that IC_{1a} is reset if the +5 V supply voltage disappears, or when there is no supply voltage at all. The latter condition may appear superfluous, but the programming voltage is immediately disconnected from the relevant EPROM pin when the programmer is switched off with S_1. The 5 V supply voltage, however, remains present for a short while because the electrolytic capacitors take some time to discharge. Conversely, IC_{1a} can not be set until the +5 V supply voltage is present. The SET input of the MMV is formed by trigger input B, which is connected to network R_3-C_5. The voltage on C_5 takes a few seconds to rise to a level that enables IC_{1a} to be set, and, consequently, the programming voltage to be applied to the EPROM. Capacitor C_{10} is required only if the $\overline{CLR}$ input of IC_{1a} is erroneously actuated by input voltage fluctuations. The value of C_{10} should be between 100 pF and 10 nF and must be determined empirically. In general, the capacitor must be kept as small as possible.

Bi-colour LED D_7 indicates the status of the EPROM programmer. The LED turns red (T_3 off; T_4 on) if the 5 V supply voltage is present, and green if both the 5 V and the programming voltage are present.

The circuit around IC_{1b} (also a MMV) is a programming pulse generator. When S_2 is actuated, a single 50-ms programming pulse is generated. An intelligent programming algorithm with variable programming pulse length is, of course, not feasible in a simple circuit like this. Even if it were available, the total programming time would not be reduced because the data and addresses are set manually, which takes much longer than 50 ms in any case.

The bulk of the signals in the programmer circuit emanates from DIP switches and associated pull-up resistors. Addresses are set with S_4 and S_5, data with S_6. Switch S_3 is used to select the EPROM type. The connections it makes are in ac-

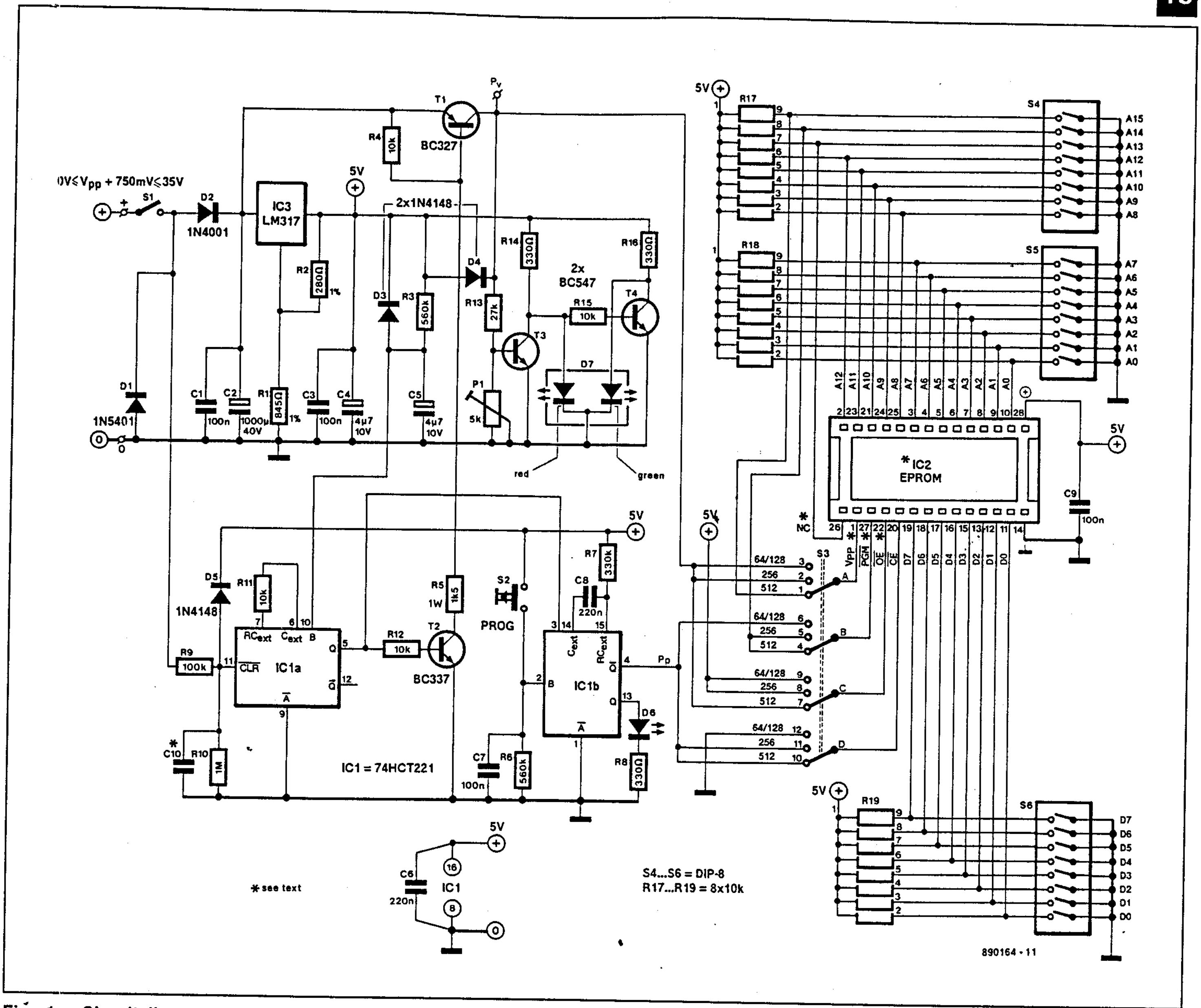


Fig. 1. Circuit diagram of the budget EPROM programmer, the larger part of which consists of the power supply with automatic Vpp delay.

cordance with the EPROM data listed in Table 1.

Construction

The printed-circuit board for the programmer is shown in Fig. 3. Start the construction by fitting the wire links. Next, fit the resistors and capacitors. Each of the three single-in-line (SIL) resistor arrays may be replaced by eight vertically mounted, discrete resistors whose upper terminals are cut short and commoned by a horizontal wire that goes into the hole provided for the +5 V connection of the array.

The semiconductors are fitted next, with the exception of the LEDs. IC3 does not need a heat-sink, and is bolted straight on to the PCB. Mount rotary switch S3, but do not cut its spindle as yet.

Be sure to mount the components that protrude from the front panel at the correct height above the board. This involves the LEDs, on/off switch S1, programming switch S2, the data/address DIP switches and the ZIF (zero-insertion force) socket

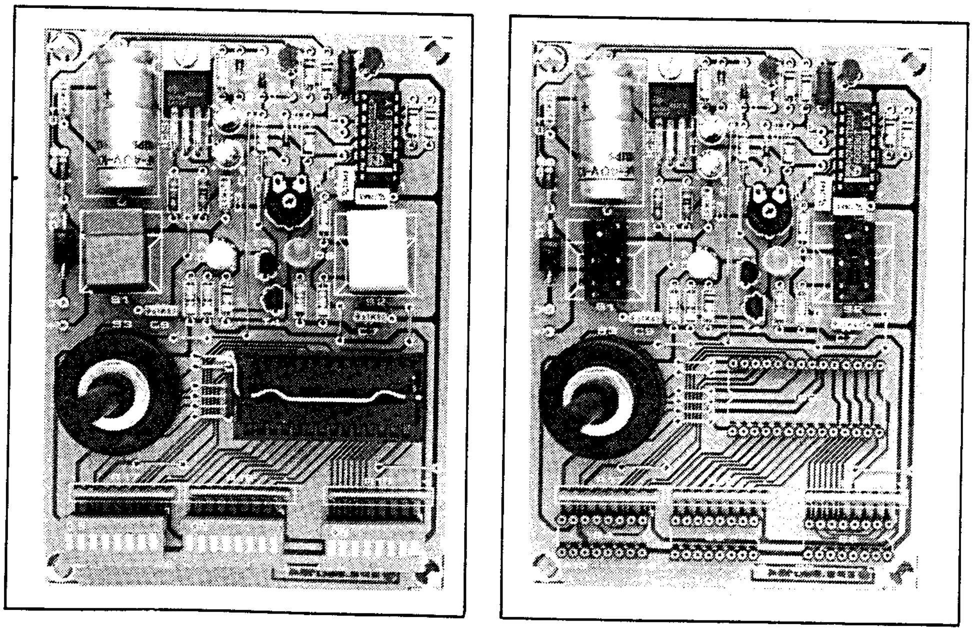


Fig. 2. As shown on these photographs of the assembled PCB, IC sockets are perfect for mounting the push-buttons and the DIP switch blocks. SIL strips are used for the ZIF socket. The height of the prototype enclosure did not require low-profile IC sockets to be stacked.

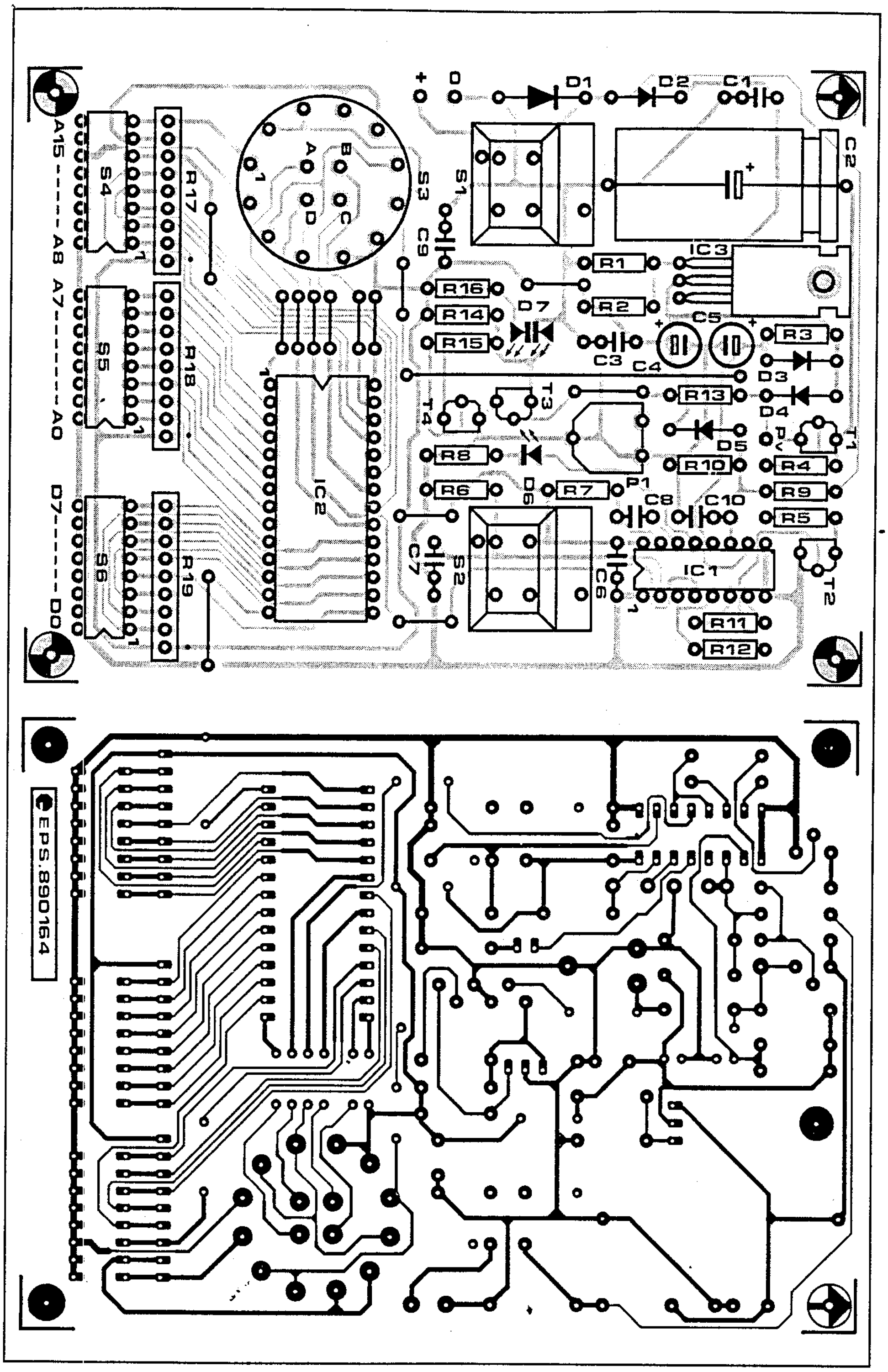


Fig. 3. Track layout and component mounting plan of the single-sided PCB for the mini EPROM programmer. Start the construction with fitting the wire links.

Parts list

Resistors:
R1 = 845Ω; 1%
R2 = 280Ω; 1%
R3;R6 = 560k
R4;R11;R12 = 10k
R5 = 1k5; 1 W
R7 = 330k
R8;R14;R16 = 330Ω
R9 = 100k
R10 = 1M0
R13 = 27k
R15 = 10k
R17;R18;R19 = 8-resistor; 9-pin SIL resistor array 10k
P1 = 5k preset H

Capacitors:
C1;C3;C7;C9 = 100n
C2 = 1000µ; 40 V
C4;C5 = 4µ7; 10 V; radial
C6;C8 = 220n

Semiconductors:
D1 = 1N5401
D2 = 1N4001
D3;D4;D5 = 1N4148
D6 = LED
D7 = 3-terminal bi-colour LED
T1 = BC327
T2 = BC337
T3;T4 = BC547
IC1 = 74HCT221
IC2 = EPROM to be programmed
IC3 = LM317

Miscellaneous:
S1 = self-locking push-button; ITW Type 61-20204000 +.
S2 = momentary action push-button; ITW Type 61-10204000 +.
S3 = PCB-mount 4-pole 3-way rotary switch.
S4;S5;S6 = 8-way DIP switch block.
28-way IC socket (ZIF type preferred).
PCB Type 890164 (see Readers Services page).

We regret that the front-panel foil for this project is not available ready-made.

+ ITW Switches • Division of ITW Ltd. • Norway Road • Hilsea • PORTSMOUTH PO3 5HT. Telephone: (0705) 694971.

for the EPROM. A good way of achieving the correct height for the DIP switches is to use wire-wrap sockets or simply three or four stacked low-profile IC sockets (see Fig. 2). Much depends on the enclosure used.

Figure 4 shows a suggested lay-out for the front panel of the programmer. Make a photocopy of this drawing and use it as a template to cut and drill the metal or ABS front-panel of your enclosure.

Setting up

Connect the external power supply and adjust it to an output that results in +10.0 V at point Pv (you may have to wait

Pin	2764	27128	27256	27512
1	Vpp	Vpp	Vpp	A15
22	$\overline{OE}$	$\overline{OE}$	$\overline{OE}$	$\overline{OE}$/Vpp
26	n.c.	A13	A13	A13
27	PGM	PGM	A14	A14

Signal	2764	27128	27256	27512
$\overline{OE}$	H	H	H	
$\overline{OE}$/Vpp				Pv
Vpp	Pv	Pv	Pv	
$\overline{CE}$	L	L	⊔	⊔
$\overline{PGM}$	⊔	⊔		

Table 1. EPROM programming data as set by the EPROM type switch on the front panel.

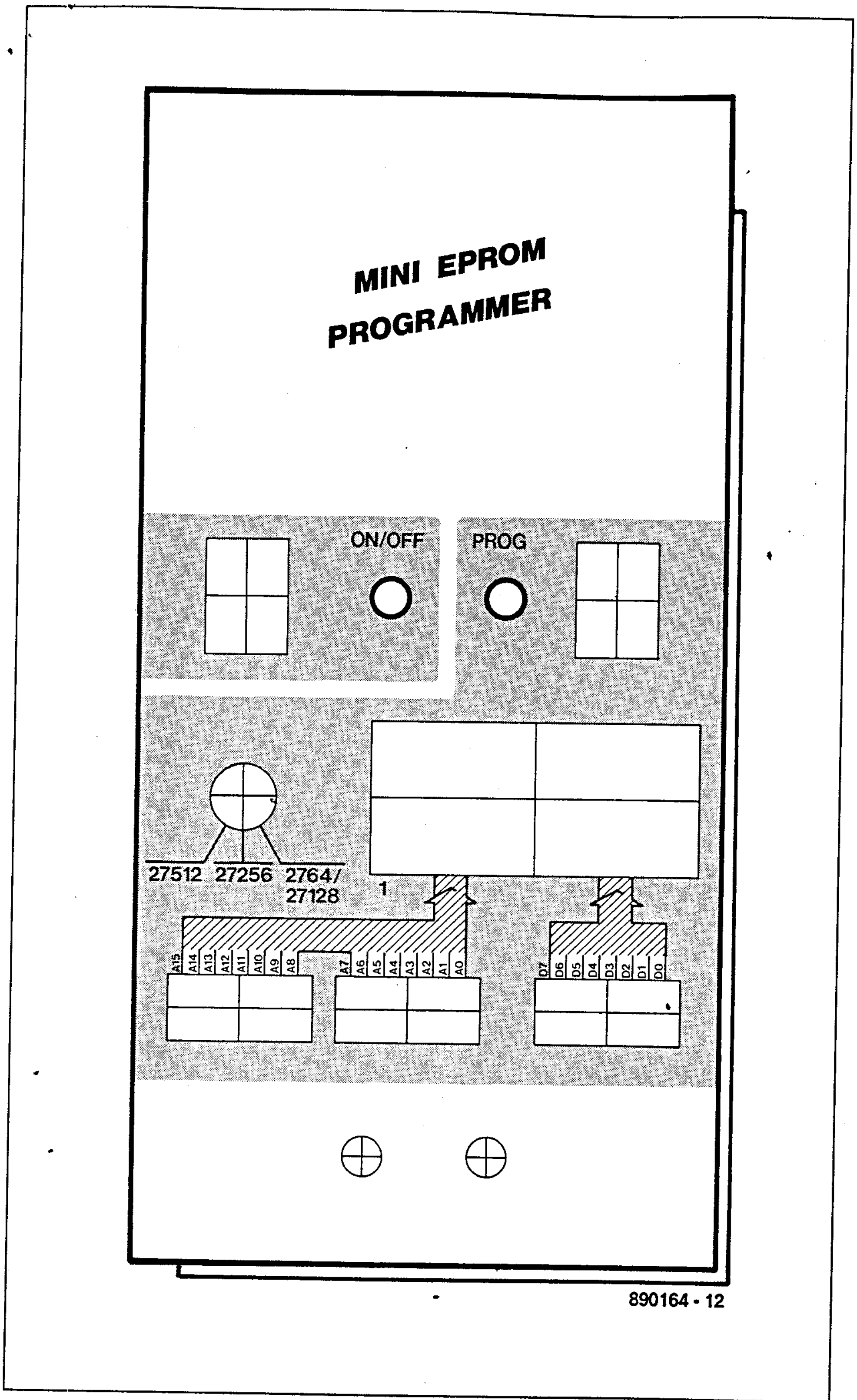


Fig. 4. Suggested front-panel layout shown at true size for easy reproduction.

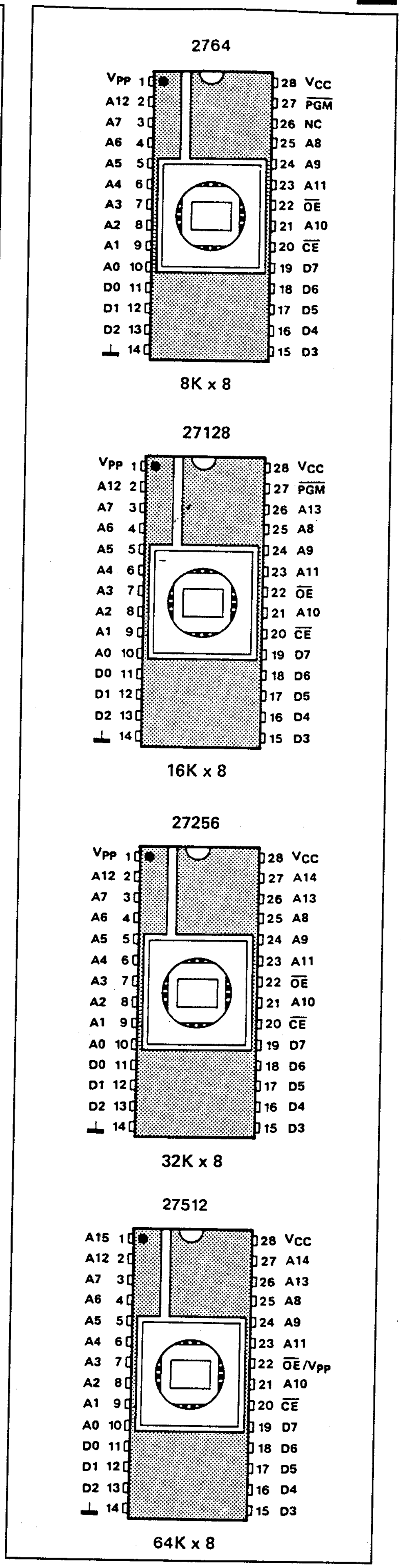


Fig. 5. Pin-outs of the four members of the 27xxx family that can be programmed.

a second or two until T_1 is turned on). Adjust P_1 until the status LED changes colour. This completes the adjustment procedure.

Do's and don'ts

There are a few basic rules to keep in mind when using the mini EPROM programmer:

- Before inserting an EPROM, always check the programming voltage at point Pv and adjust your power supply to set the correct value for the device to be programmed. Next, set the EPROM type on S_3.
- Never insert or remove an EPROM with the programmer switched on.
- And, finally, *think* before actuating the programming switch!