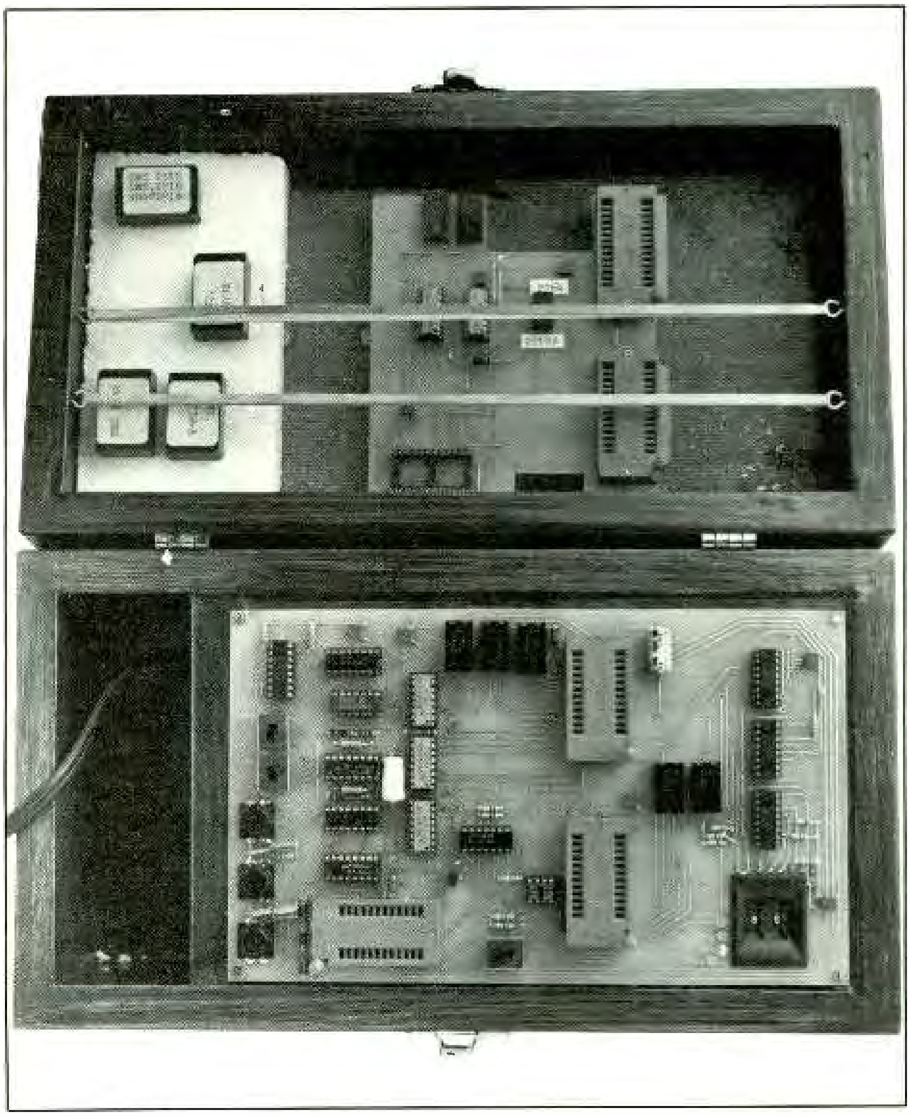
# A Stand-Alone EPROM Programmer

## Part 1

***Read a program or copy programs from one EPROM to another with this versatile Programmer that can be easily upgraded as needed to accommodate new EPROMs as they become available***

**By W. Schopp**

Most inexpensive EPROM programmer/readers are designed to accommodate a single type of EPROM, generally using a personal computer as the controller. If you want a more versatile programmer/reader that can handle a variety of EPROMs of different storage capacities, it's very costly. The Stand-Alone EPROM Programmer to be described offers an alternative to both types of programmers at only a moderate cost. With it, you can program and read most popular EPROMs with capacities up to 32K in the basic unit presented here. Additionally, our Programmer will handle EPROMs with capacities ranging up to 128K with a retrofit add-on adapter that we'll discuss next month.



### *General Information*

An address is the only information needed to access or program data into an EPROM at a specific location. Putting a code consisting of 1s and 0s on the EPROM's address lines finds the location of the desired data, which then appears on the EPROM's data lines and is similarly presented in the form of 1s and 0s.

To be able to address the large

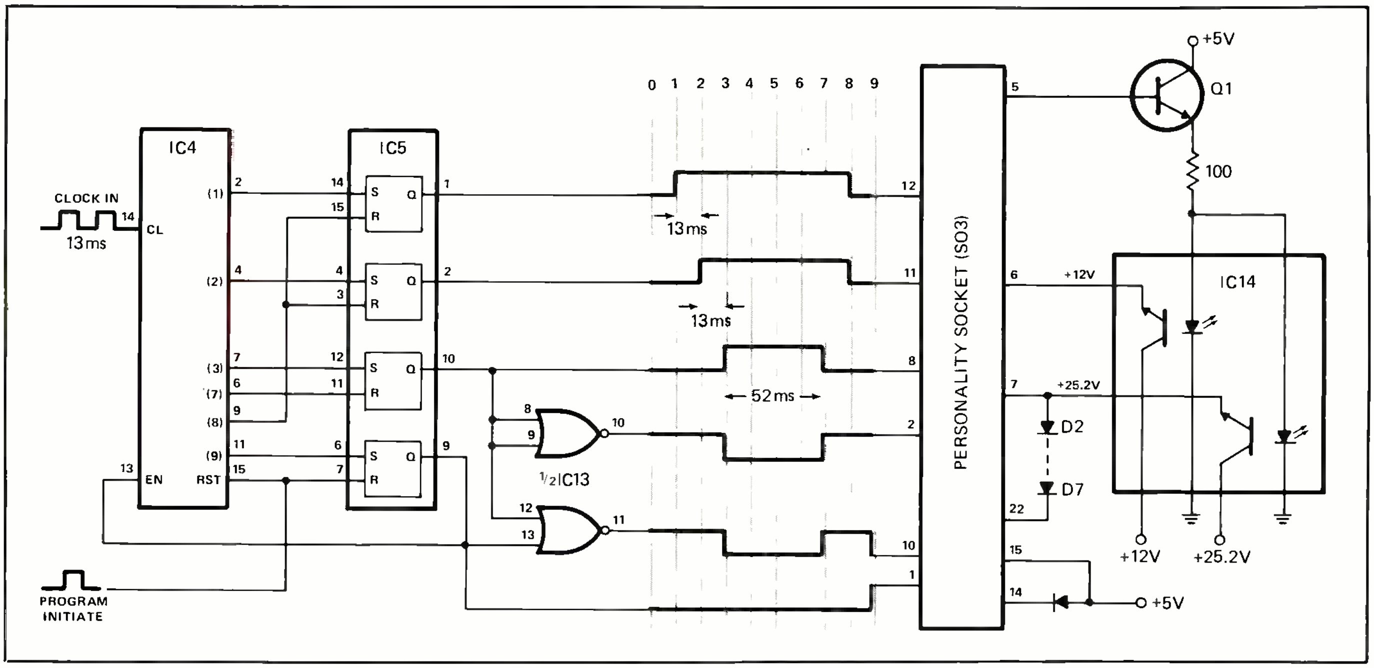



*Fig. 1. A sequence generator is the heart of the EPROM Programmer.*

number of memory cells in some logical and repeatable order, a three-stage counter is used. Outputs from the counters are connected to the EPROM's address lines. Counting from all 0s to all 1s on the 12 lines of this basic project gives you 4096 different combinations of 0s and 1s. This binary information is read on LED hexidecimal displays. The binary information is decoded by the special displays and is presented in hex format for readings ranging from 000 to FFF.

Two more LED hex displays read the data output after programming has taken place. This method of display verifies that the program being placed in an EPROM has been accepted. Three pushbutton switches provide for counter reset, single-step counter advance and program initiation. Slide switches on the Programmer's main board allow you to select count direction, manual or automatic mode of operation and to select the program source data.

## About the Circuit

All waveforms and voltages for programming a variety of EPROMs are available at the project's PERSONALITY socket. Personality plugs for the various EPROMs connect the waveforms and programming pulses to the correct pins of the type of EPROM being programmed. These personality plugs are wired by you on 24-pin headers with covers and are individually labeled according to the type of EPROM with which they are to be used.

The heart of the Programmer is the sequence waveform generator shown in Fig. 1. This circuit provides all the waveforms and programming voltages needed to program 16K 2716, 32K 2732, 64K 2764 and 128K 27128 EPROMs. The generator is made up of half the buffers in hex inverter *IC1*, decade counter *IC4* and latch *IC5*.

Clock pulses of 13 ms duration are generated by *IC1* and applied to the clock input of *IC4*, which feeds pulses in the correct sequence to the set-reset latches in *IC5*. The Q outputs of these latches produce the timed waveform shown. Initiating the program cycle, the counter runs through one complete cycle of waveforms and stops until the program is reinitiated.

A good understanding of this sequence generator will allow you to make up personality modules for new EPROMs, using the waveforms and technical information available from the manufacturers for each type of EPROM. Hence, the Programmer will not become obsolete as new EPROMs become available.

The waveforms produced by the sequencing generator are used to turn on the EPROM's chip-enable (CE) and output-enable (OE) functions. Optoisolator *IC14* (see Part 3 of Fig. 2) keys the higher programming voltages needed to program the EPROM. The appropriate waveform is connected to pin 5 through the personality plug to turn on *IC14* and make the higher programming voltages available at pins 6, 7 and 22 of *SO3* (see Part 2 of Fig. 2).

Pin 14 of *SO3* provides +5 volts and allows the higher programming voltages to be superimposed on it at the EPROM's pin without feeding back into the 5-volt supply. The 22-volt supply required by some EPROMs is derived from the 25-volt

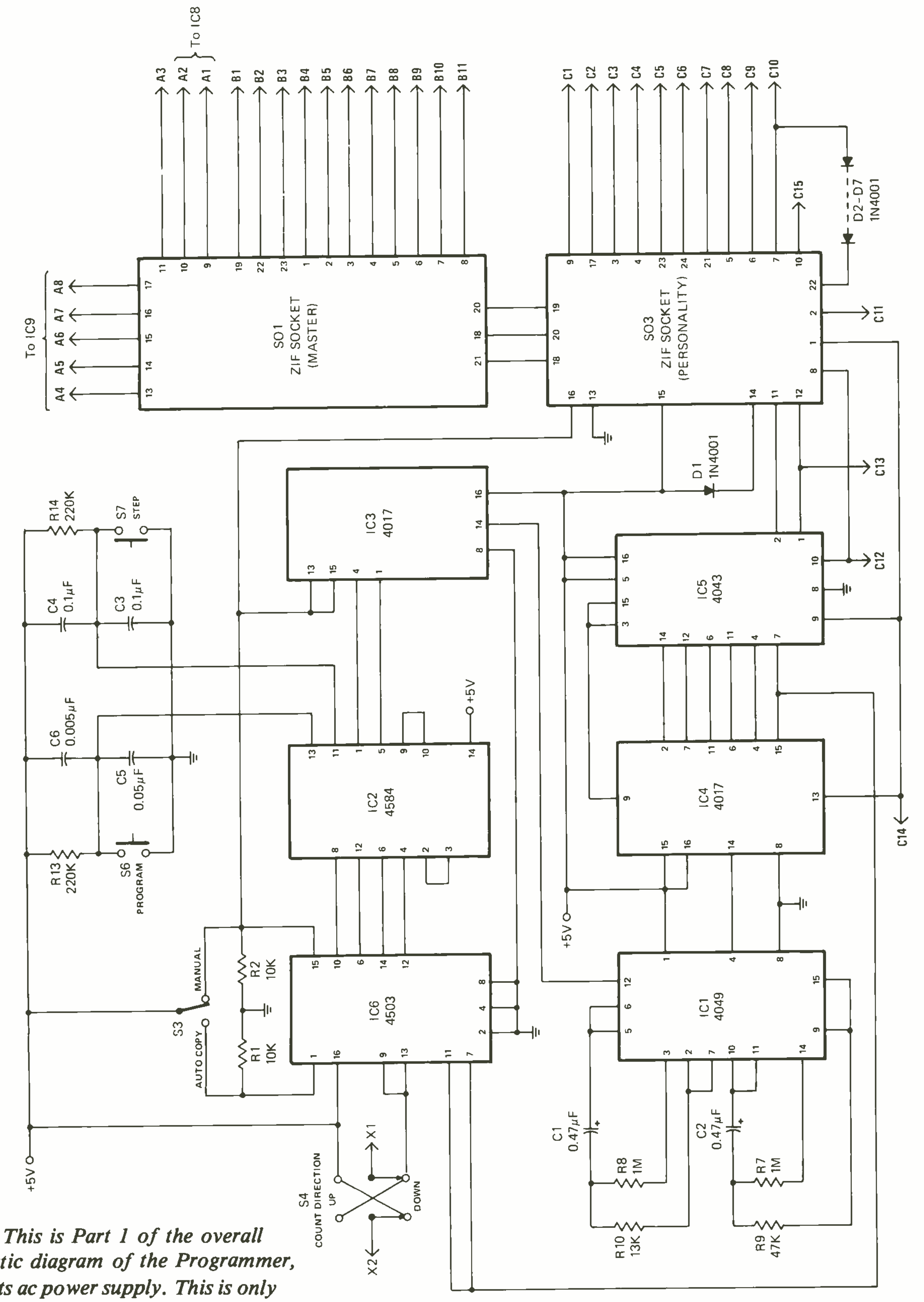



*Fig. 2. This is Part 1 of the overall schematic diagram of the Programmer, minus its ac power supply. This is only the first of three parts.*

supply by dropping 0.6 volt for each of the *D2* through *D7* diodes between pins 7 and 22 of *SO3*. Future expansion connections are also brought to PERSONALITY socket *SO3*. These connections include reset, carry up and carry down lines.

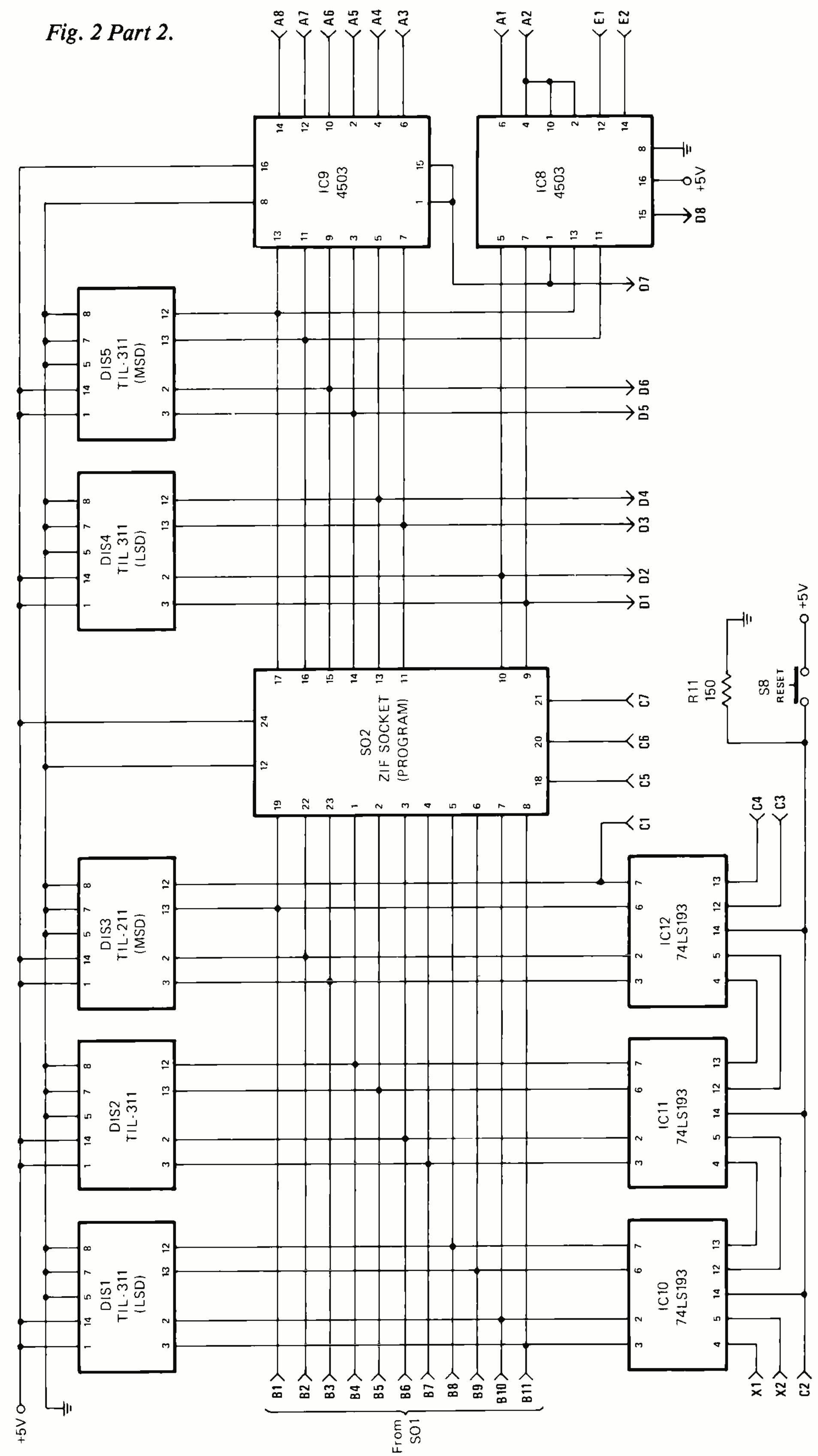
The balance of the Programmer is straightforward, as shown in the three Parts that make up Fig. 2. (Note: Because this Programmer's circuit is fairly complicated, though relatively inexpensive, to be able to present it in its entirety, the very large schematic diagram had to be divided into the three Parts that make up Fig. 2.—Editor.)

Pushbutton switches *S6* and *S7* are debounced by Schmitt trigger buffer *IC2*. The half of *IC1* not used in the sequence generator produces 48-ms pulses for the automatic program mode. These pulses go to decade counter *IC3*, which alternately produces step and program pulses. Tristate buffer *IC6* selects manual inputs from *S6* and *S7* or automatic step and program pulses from *IC3*. Switch *S3* controls *IC6*'s tristate operation.

Counters *IC10, IC11* and *IC12* have outputs that are tied directly to the address inputs of MASTER and PROGRAM sockets *SO1* and *SO2*. The outputs of each counter are read with hex displays *DIS1, DIS2* and *DIS3*. The data lines of the EPROM being programmed are read on *DIS4* and *DIS5*. The LED numeric displays are isolated from the input program data by *IC7, IC8* and *IC9*. These ICs read data only after it has been programmed into the EPROM. The tristate buffers momentarily make the connection between the data to be programmed and the EPROM being programmed. All buffers are normally in the tristate (float) mode and are opened briefly during the programming cycle by the waveform from pin 12 of *SO3*.

Switch *S5* selects *IC7* and half of *IC8* to select the input data from manual programming switches *S1*

Fig. 2 Part 2.



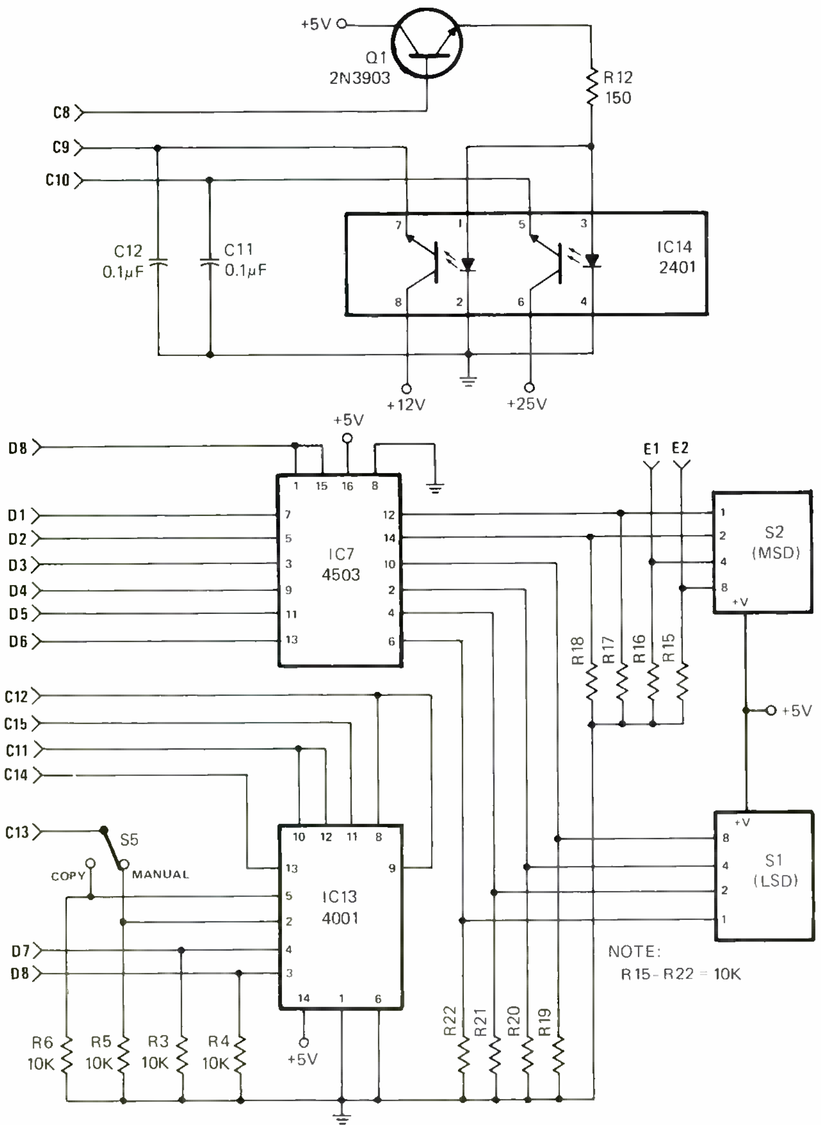



*Fig. 2 Part 3.*

## PARTS LIST

**Semiconductors**

D1 thru D13—1N4001 rectifier diode
DIS1 thru DIS5—TIL-311 LED hexidecimal display (Texas Instruments; available from Jameco)
IC1—CD4049 hex inverting buffer
IC2—CD4585 hex Schmitt trigger
IC3,IC4—CD4017 decade counter
IC5—CD4043 quad R/S flip-flop
IC6 thru IC9—CD4503 tristate hex buffer
IC10,IC11,IC12—74LS193 4-bit up/down counter
IC13—CD4001 quad 2-input NOR gate
IC14—NEC 2401 dual optical isolator (available from Digi-Key)
IC15—7824 +24-volt regulator
IC16—7812 +12-volt regulator
IC17—7805 +5-volt regulator
Q1—2N3903 or equivalent transistor

**Capacitors** (25-volt)

C1,C2—0.47-μF tantalum
C3 thru C6—0.05-μF disc
C7,C8—100-μF electrolytic
C9 thru C17—0.1-μF disc
C18 thru C23—1,000-μF, 50-volt electrolytic

**Resistors** (¼-watt, 5% tolerance)

R1 thru R6,R15 thru R22—10,000 ohms
R7,R8—1 megohm
R9—47,000 ohms
R10—13,000 ohms
R11,R12—150 ohms
R13,R14—220,000 ohms

**Switches**

S1,S2—Hex thumbwheel type (Unimax No. SF-52 plus end plates; available from Jameco)
S3,S4,S5—Miniature pc-type dpdt slide type (No. 1201 M2QE; available from M&Q Components)
S6,S7,S8—Normally-open spst pushbutton type (Panasonic No. EVQ-PAR11K; available from Digi-Key)

**Miscellaneous**

T1—24-volt, 180-mA center-tapped transformer (EWC No. EL-24-180A18; available from Digi-Key as Part No. T104-ND)
T2—12-volt, 350-mA center-tapped transformer (EWC No. El-12-350A6; available from Digi-Key as Part No. T101-ND)
Double-sided main printed-circuit board with plated-through holes and two single-sided power-supply pc boards (see text); 24-pin ZIF sockets (3); sockets for all other ICs (except IC15, IC16 and IC17) and displays; materials for personality modules (see text); suitable enclosure; machine hardware; hookup wire; solder, etc.

**Note:** A ready-to-wire main board with plated-through holes is available for $22.00, including P&H, from Electronic Enterprises, 3305 Pestana Way, Livermore, CA 94550.

**Addresses of Suppliers**

**C&K Components**
15 Riverdale Ave.
Newton, MA 02158-1082

**Digi-Key Corp.**
P.O. Box 677
Thief River Falls, MN 56701
1-800-344-4539

**Jameco Electronics**
1355 Shoreway Rd.
Belmont, CA 94002
415-592-8097

and *S2* or selects *IC9* and the other half of *IC8* to accept program data from the EPROM plugged into MASTER socket *SO1*. Only "read" voltages are present on *SO1* so that no alterations can be accidentally made to the EPROM being copied.

Not shown in Fig. 2 are *C7* and *C8*, which provide extra on-board dc filtering, and *C9, C10* and *C13* through *C17*, which provide electrical noise bypassing from the + dc lines to ground in various parts of the circuit.

Figure 3 is the schematic diagram of the Programmer's three-voltage ac-operated power supply. Each of the output lines from this power supply is fully regulated and filtered. Note that 24-volt regulator *IC15* is raised 1.2 volts above ground by the voltage dropped across diodes *D10* and *D11*. This gives an actual output from this supply of 25.2 volts.

Wiring details for personality modules for a variety of popular EPROMs are shown in Fig. 4.

## Construction

Because this project makes extensive use of integrated circuits, sockets and other miniature components, printed-circuit wiring is the recommended method of construction. Though it is possible to wire the Programmer using Wire Wrap hardware and wire, the many wire runs that must be made can easily result in wiring errors. The basic project is wired on three boards—one large one for the programming circuits and two smaller ones for the power supplies.

To keep the project as compact as possible, a double-sided pc board with plated-through holes is actually needed for the main board. The actual-size etching-and-drilling guides for both sides of this board are shown in Fig. 5. The actual-size guides for the two power supply boards are shown in Fig. 6.

If you cannot have the double-sided board with plated-through holes made locally, you can purchase it

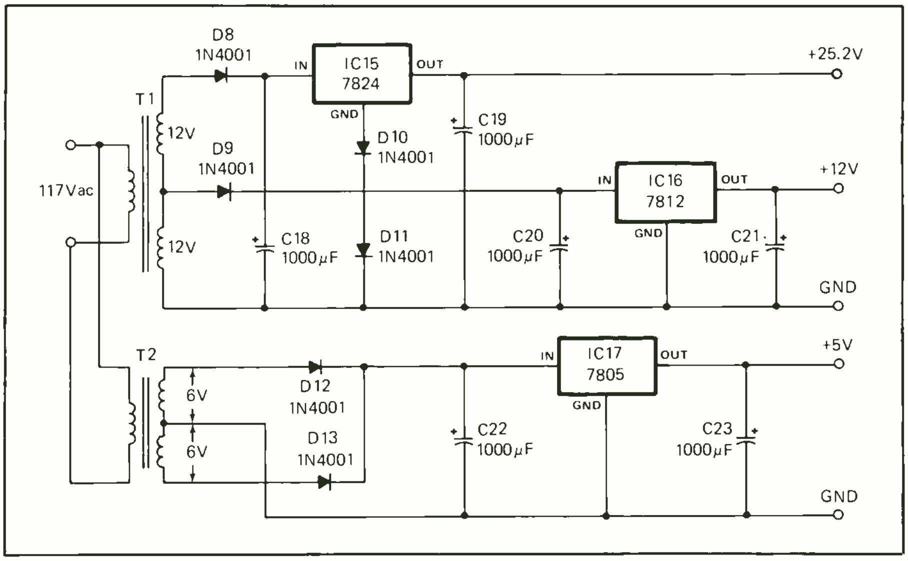



*Fig. 3. Three dc voltages are required by the Programmer for powering the circuitry and programming EPROMs.*

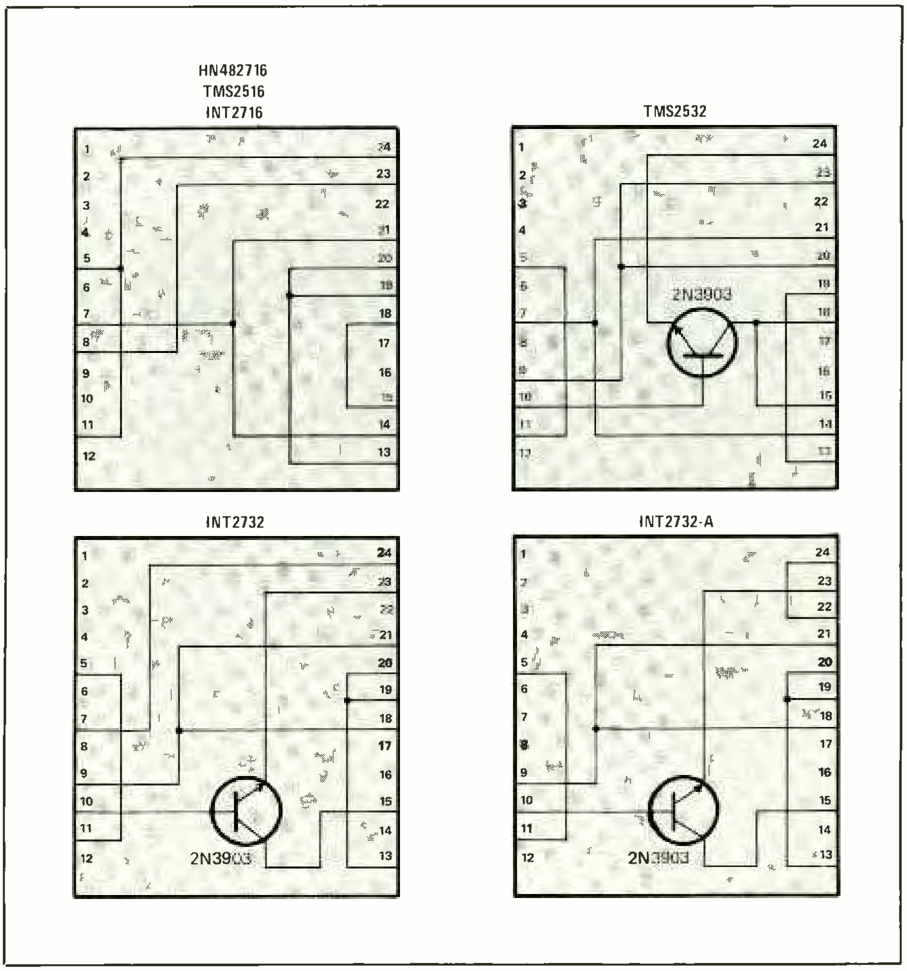
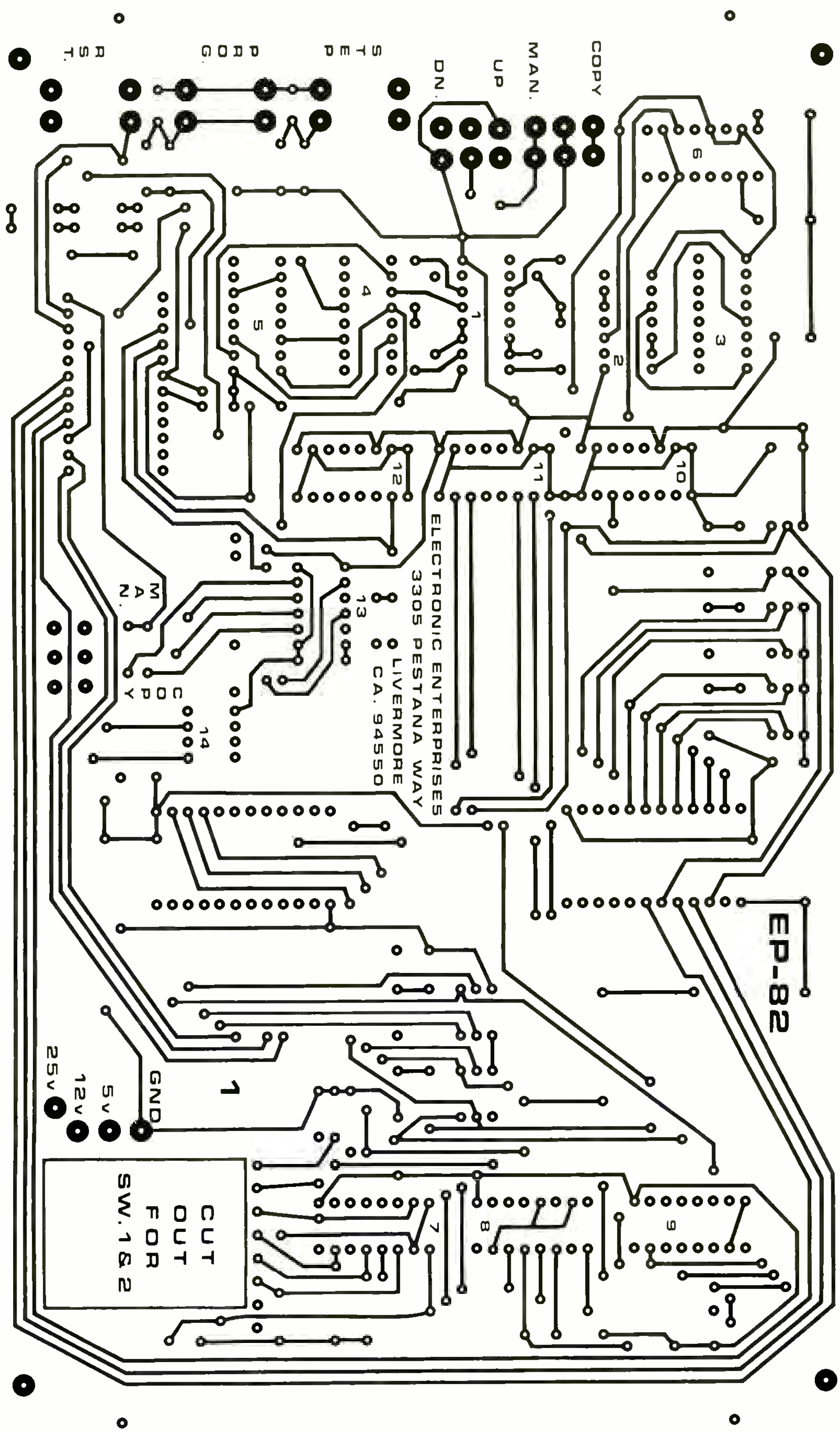



*Fig. 4. Different personality modules are required for different types of EPROMs. Wiring for four such modules inside plug-in headers is shown here for popular 16K and 32K EPROMs.*

*Fig. 5. Actual-size etching-and-drilling guides for both sides of project's main printed-circuit board.*

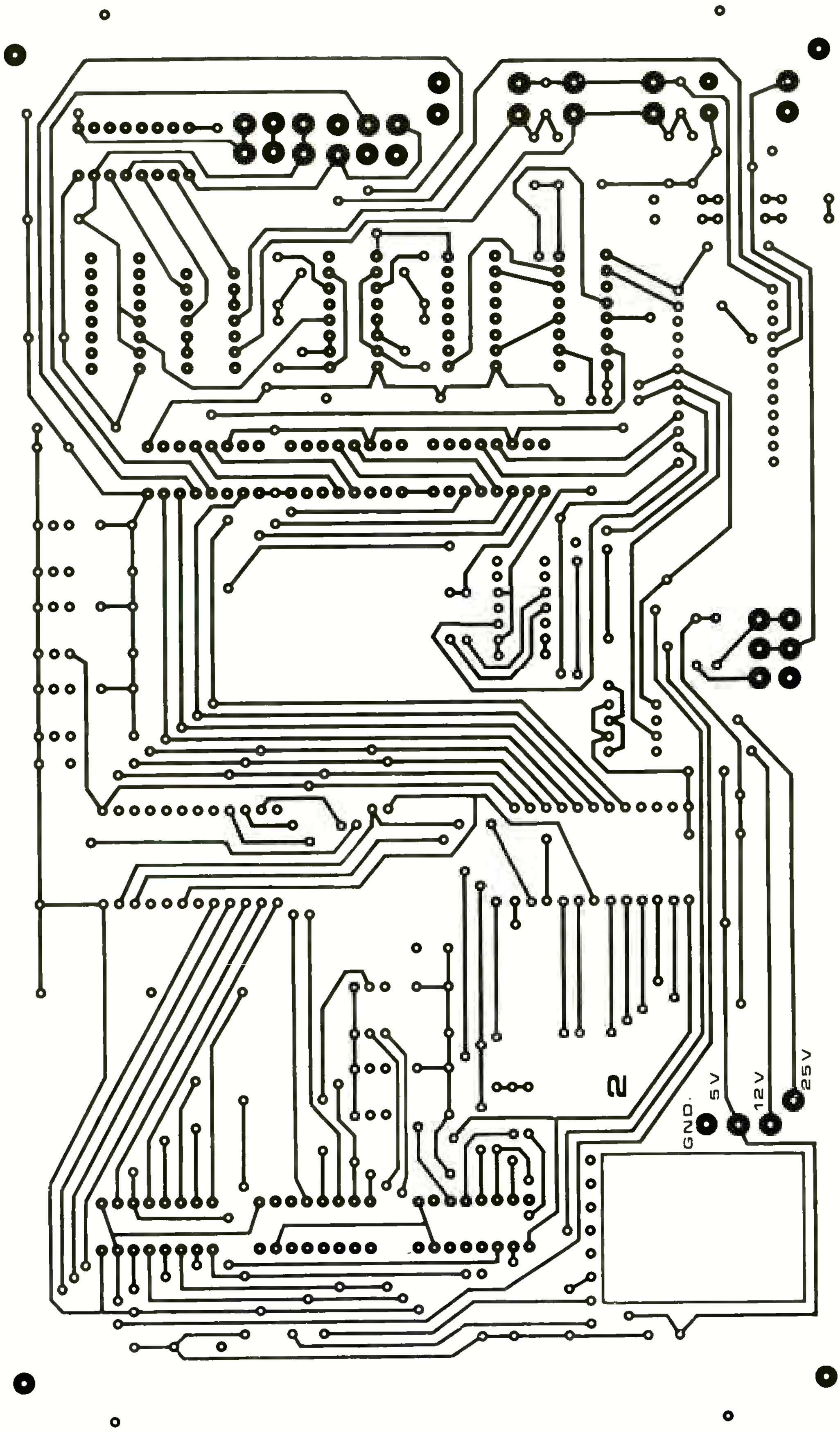



ready to wire from the source given in the Note at the end of the Parts List. This is the easiest—and recommended—approach because all you have to do is install the components and solder them into place, rather than running the risk of having the top and bottom foil patterns being misregistered and having to go through a complicated and tedious procedure to make do. (The power-supply boards are not commercially available from the same source as the main board. Therefore, you must either fabricate them yourself or have them made locally.)

You can, of course, fabricate all three boards yourself—even the double-sided main board—using single-sided techniques. A couple of tricks when working with the double-sided main board will yield a properly operating project. Before you do anything after fabricating your boards, carefully read and do the following.

For the main board, the easiest way around the plated-through dilemma is to install the ICs directly and solder their pins to the pads on both sides of the board. If you wish to socket the ICs, use Molex Soldercons or Wire Wrap sockets to provide soldering access to the pin/pad junctions on the top as well as on the bottom of the board.

Soldercons are easier to work with because you install them and solder their pins to the board's pads just as you do for the pins of the ICs themselves. If you use Wire Wrap sockets, you must space them about ¼″ above the top surface of the board to have soldering access and work very carefully with a fine conical or needle point soldering tip and fine-wire solder to avoid creating solder bridges between the closely spaced soldering pads. No matter




what method you use to install the ICs, Wire Wrap zero-insertion-force (ZIF) sockets should be used for *SO1, SO2* and *SO3*.

The LED hex displays present a problem because their pins are molded into cases with no pin/pad soldering access when they are directly installed on the board. For these devices, you must use Soldercons or Wire Wrap sockets. Also, note that several pins that are not needed have been left off the hex displays by the manufacturer and that for these no holes are provided on the board. So when using Soldercons omit those for which no holes are provided and when using sockets clip the associated pins flush with the bottoms of the sockets.

Pc-mount slide, toggle and push-button switches generally have shouldered lugs whose pins plug into the board holes and provide limited access for soldering on the top side of the board. So you should have few problems in this area, except if you use keyboard-type switches for *S6, S7* and *S8*. If you do find yourself in difficulty here, you have the option of locating the switches off the board —on a separate panel—and running hookup wires between their lugs and the appropriate holes on the board. Thumbwheel switches *S1* and *S2* can mount on the board regardless of whether or not your board has plated-through holes.

Small components—like resistors, capacitors, diodes and the transistor—offer no obstacle in terms of whether or not your main board has plated-through holes. Because these components have no hidden leads or pins, it is a simple matter to solder all leads to all visible pads on both sides of the board.

If you are using a main board that does not have plated-through holes, refer to Fig. 7 and do the following first. With a soft black pencil, mark the locations of every point indicated by a large solid black dot. Each of these locations indicates a point

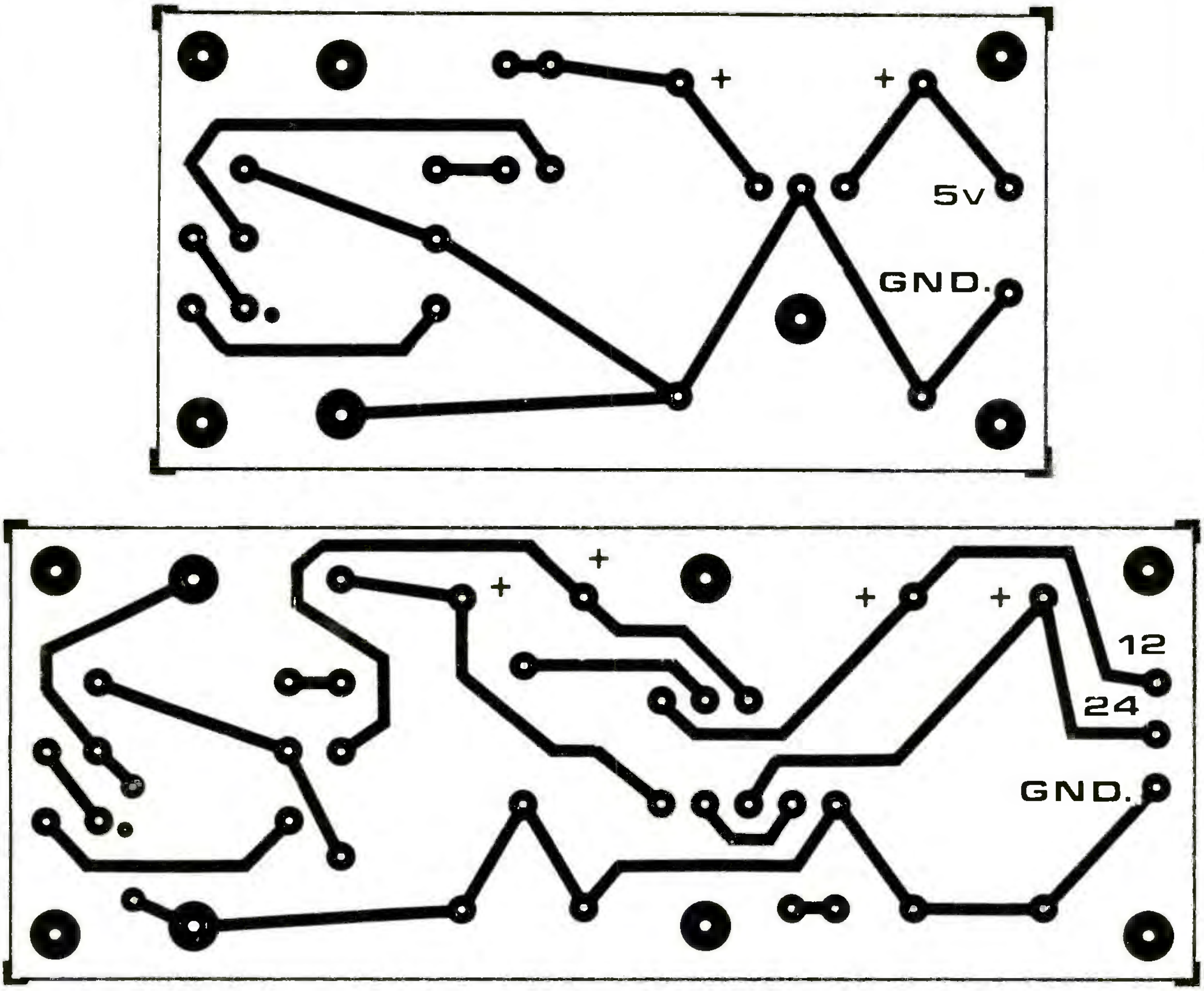



*Fig. 6. Actual-size etching-and-drilling guides for power supply boards.*

where the conductor pattern on one side continues on the other side of the board and, hence, requires a wire jumper to bridge the pads for the hole so that the conductor pattern is indeed continuous and the project operates properly.

Use solid bare 22-gauge wire to make the jumpers. Pass the end of the wire through one hole until it protrudes through the hole on the other side of the board by about ⅛" and solder the short end to the pad on that side of the board. Flip over the board and solder the wire to the pad on the other side. Allow the connection to cool, and then clip the wire close to the board. Repeat this for all the holes you marked. There are quite a number of these bridging conductors; so work carefully to make sure you get every one of them. If you miss just one and discover it after installing the component, you may have to destroy the component or an expensive socket to be able to rectify your oversight.

Once the bridging conductors are installed, you can proceed to populate the main board. Start with installation of the sockets—*not* the ICs themselves—so that you have unobstructed soldering access to their

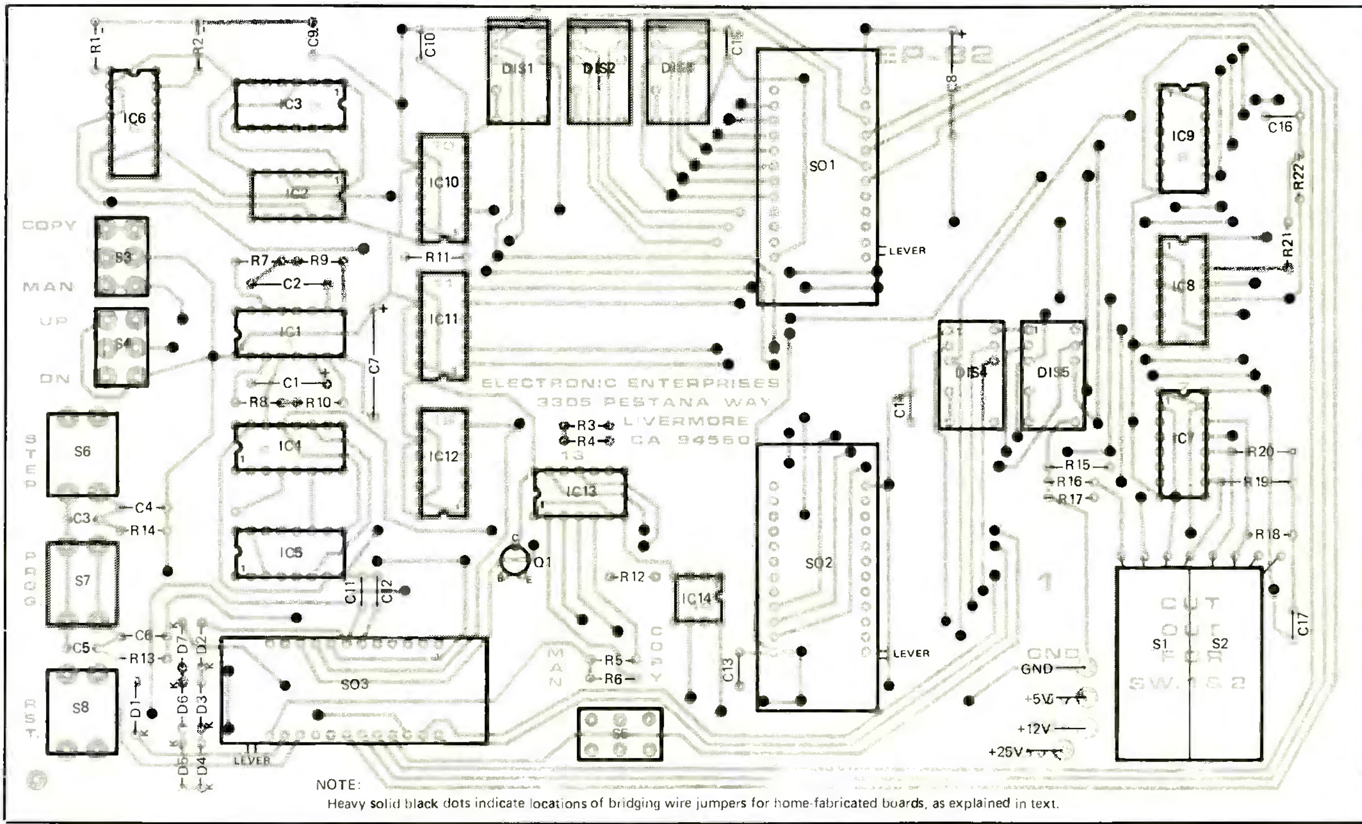



*Fig. 7. Wiring diagram for main board.*

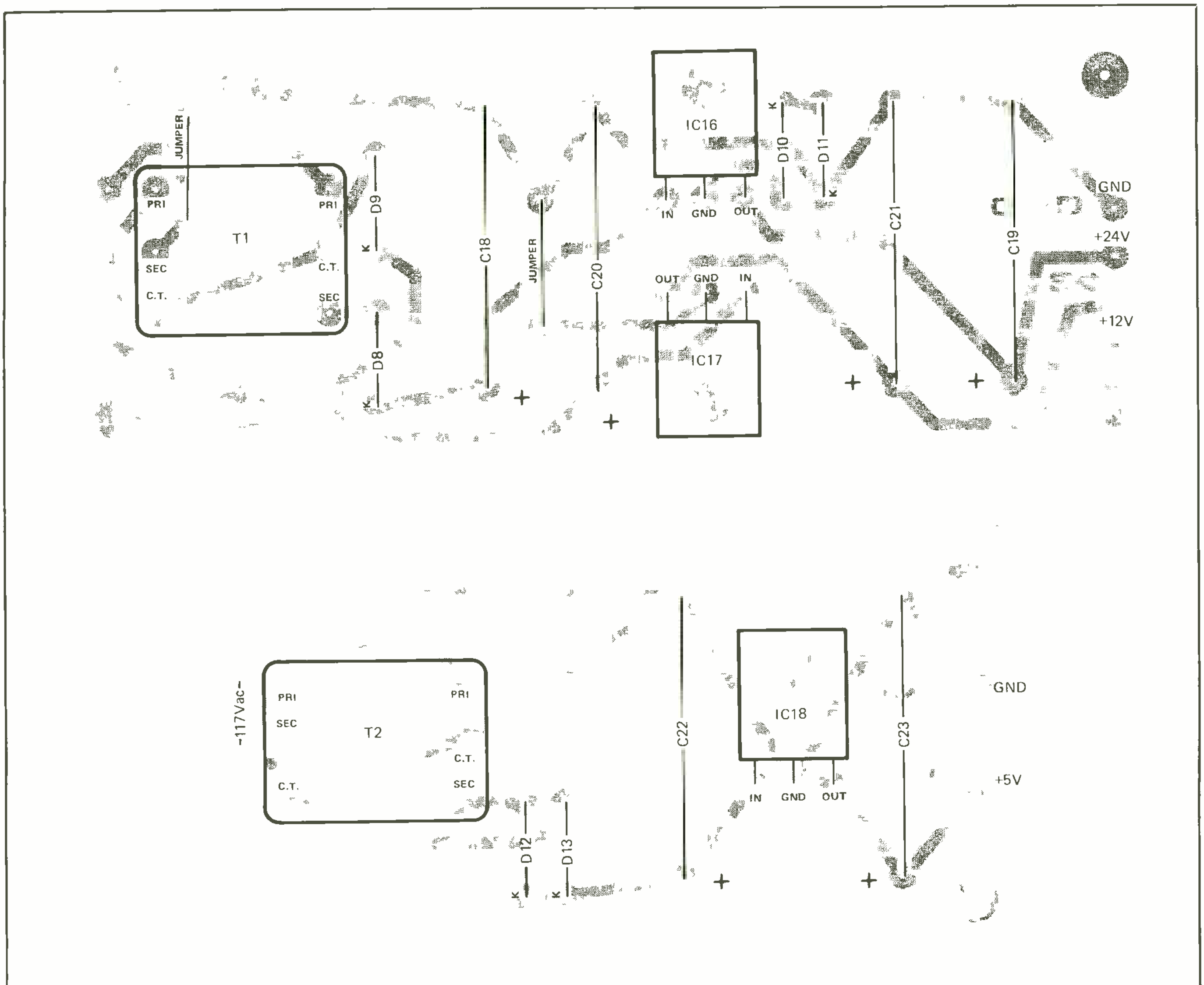



*Fig. 8. Wiring diagram for power supplies.*

pin/pad connections on the top of the board. Use only enough solder to assure a good electrical joint on the component side of the board at each connection point.

After installing the sockets, mount the transistor and diodes in their respective locations, taking care to properly orient them according to Fig. 7. Use soldering heat judiciously to avoid damaging these components. Then install and solder into place the capacitors (observe polarity where required). After soldering the leads of all components to the pads on the bottom, flip over the board and do the same for all leads on the top of the board.

Thumbwheel switches *S1* and *S2* drop into the cutout area in the lower-right of the board. Connections from their lugs to the appropriate holes in the board are made with insulated hookup wire with ¼" of insulation removed from both ends.

When all components, except the ICs and displays, have been installed, carefully inspect the board for poor soldering and for solder bridges, especially between the closely spaced IC and socket pads with respect to the latter. Reflow the solder at any suspect connection and remove excess solder to clear away any solder bridges you discover.

If you have purchased the commercial plated-through-hole board, all you have to do is stuff it with the components and sockets according to the wiring details in Fig. 7. Soldering of each connection need be done on the bottom side of the board only because the plating completes the circuit and draws solder to the top-of-the-board connections by capillary action. Use solder judiciously to avoid creating solder bridges between the closely spaced conductors of the foil pattern. Do not install the displays and ICs in their sockets yet.

Now wire the power-supply boards, referring to Fig. 8 for details. These single-sided boards require soldering on only one side. Make sure you properly orient the rectifier diodes and electrolytic capacitors and double check that the correct transformer is installed on each board before soldering any leads or pins to the copper pads.

You can house your Programmer in any enclosure of your choice that is large enough to accommodate the two power-supply and main boards and provides a panel on which to mount any switches you decide to locate off the board. The prototype of the Programmer project was housed inside a home-made wooden enclosure with a hinged top, as shown in the lead photo. The flip-up top was made deep enough so that it does not interfere with the circuitry—including the retrofit expansion add-on board that will be described next month. It also has room for handy storage of the personality modules.

Only after the Programmer is installed in its enclosure (do not forget to wire the power supplies to the main board), should you install the LED hex displays and integrated circuits. Make sure as you install these devices in their sockets that you orient them as shown in Fig. 7 and that no pins fold under or overhang the sockets. Also, practice safe handling when working with the ICs, since these devices can be damaged by static electricity.

## *Using the Programmer*

Plug the EPROM Programmer's line cord into an ac outlet and set AUTO/MANUAL switch *S3* to MANUAL. In this position, nothing happens until a pushbutton switch is pressed. Set COUNT DIRECTION switch *S4* to UP and MANUAL/COPY switch *S5* to MANUAL to accept programming data from thumbwheel switches *S1* and *S2*. Press and release RESET switch *S8* to set the counter to 000. Momentarily set *S3* to AUTO and note that the counter starts incrementing sequentially. Return *S3* to MANUAL and press and release *S8* to reset the counter to 000.

Select the proper personality module for the EPROM to be programmed and plug it into PERSONALITY socket *SO3*. Plug the EPROM to be programmed into PROGRAM socket *SO2*. The Programmer is now ready to run a manual-entry program.

Enter the program data for the first address by setting *S1* and *S2*.

The LED hex displays should read FF if there is no program data at that address. Push the PROGRAM button and, as you release it, notice that the program data entered is displayed on *DIS4* and *DIS5*. This completes your first entry.

Press and release STEP button *S7* to advance the counter by one step and program in your next data entry. Then keep programming in data at each successive address according to the program you wish to burn into the EPROM.

You can reverse count direction at any time to check entered program data at any previous address by setting COUNT DIRECTION switch *S4* to DOWN. It is a good idea to record on paper each step of your program as it is entered so that if you stop in the middle of a programming job you can return to where you left off when you resume programming.

Some EPROMs will instantly read programmed data, while others will have a short lag between the time the data is programmed and when the data appears in the display. This delay is in the display circuitry and only when in the manual mode and you wish to see a piece of data.

To read data from an EPROM that has already been programmed, plug into the PERSONALITY socket the appropriate personality module and plug the EPROM into the PROGRAM socket. Set *S1* and *S2* to read location FF to insure that the existing program does not become altered should the PROGRAM button accidentally be pressed during the read operation. Step through the EPROM's addresses with the manual STEP button. The data for each address will appear in *DIS4* and *DIS5*.

To copy an EPROM, set *S3* to MANUAL to stop the action. Set *S4* to UP and *S5* to COPY to accept data from the programmed EPROM. Plug the EPROM to be copied into the MASTER socket and the EPROM to be programmed into the PROGRAM socket. Set *S3* to AUTO to start the Programmer copying at a rate of 2.5 to 3 addresses per second. This is a slow enough speed for you to see the data being displayed to verify that programming is taking place.

## *Coming Next Month*

This completes the information on constructing and using our basic 32K EPROM Programmer. Next month, we will show you how to upgrade the basic Programmer so that you can program and read 64K 2764 and 128K 27128 EPROMs with a plug-in add-on board that retrofits on the main programmer board.