

Easy Computerized Control and Monitoring

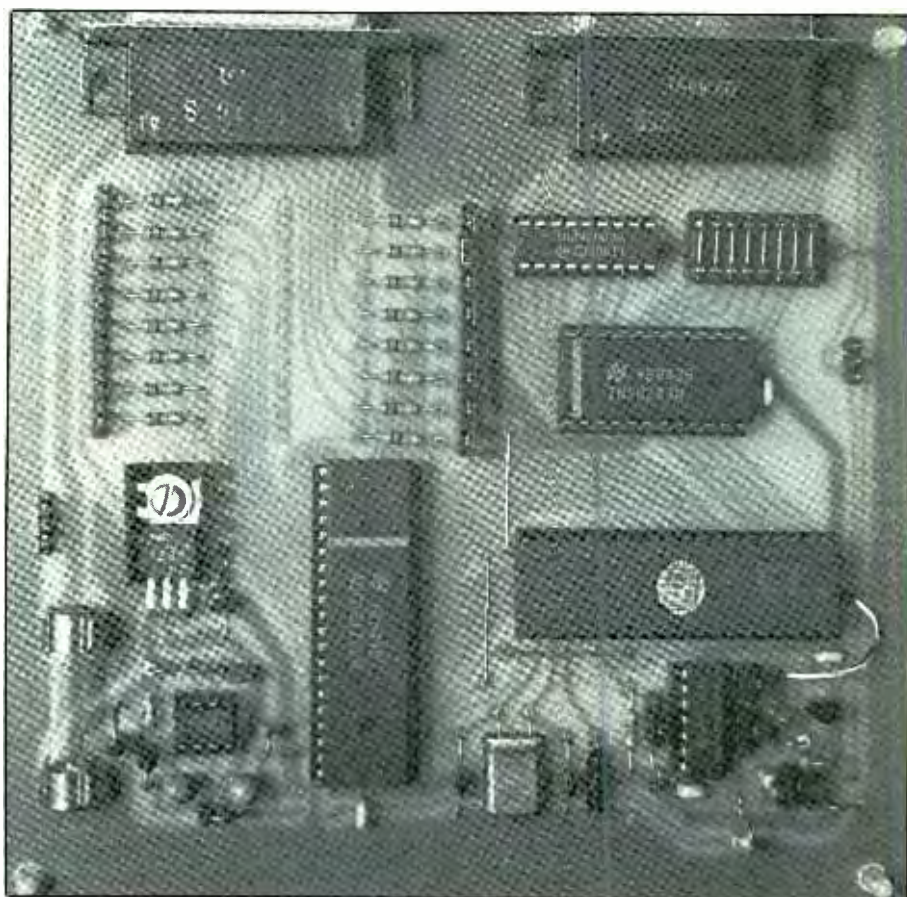
An economical way to monitor sensors and actuate electrically operated devices using a computer, external controller and a BASIC language interpreter

By Daniel N. Eggert

Personal computers can be used for a wide variety of things other than processing words and crunching numbers. For example, they can be used as monitoring and controlling devices for a home or business security system or as climate controllers. Model railroaders can use their computers to automate their layouts. Computers can also be used during the Christmas season to control lighting displays. In fact, there is almost no end to what you can control with a personal computer.

How you go about achieving control of electrically powered devices and appliances with a computer is rarely as simple as thinking up applications. And if you go the commercial hardware/software route, you can almost certainly bet on fairly high prices for the hardware and software needed to accomplish such control. With the hardware controller described here and a BASIC language interpreter, however, the cost can be kept reasonably low.

Unlike many commercial controllers, the one described here does not occupy a slot in your computer. Instead, it works through one of the serial ports on your computer. Connection between controller and computer is via a simple three-conductor cable that carries the receive-data, transmit-data and ground lines. Ad-



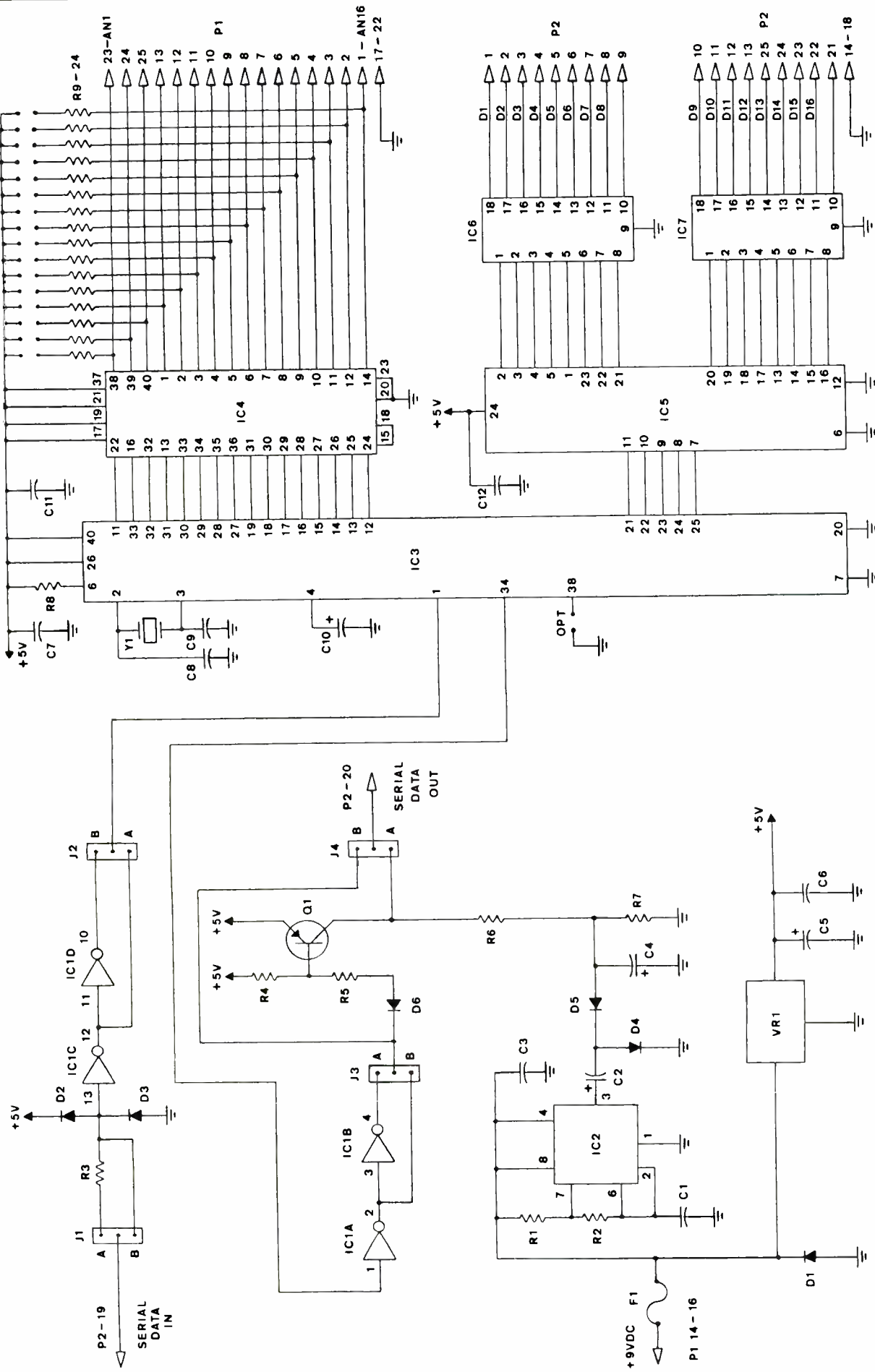
ditionally, the project has its own ac-operated power supply. Thus, the controller can be located remotely from the computer, closer to any sensors that must be monitored and devices that must be actuated.

Accompanying the controller project is a test fixture and an operating BASIC program for it. This test fix-

ture can be used as an aid to understanding programming and operation of the controller.

Many input/output configurations are possible with the 16 eight-bit analog inputs and 16 digital I/O

Fig. 1. Complete schematic diagram of controller circuitry.



lines on the controller. The analog inputs can also be used as digital inputs in applications where a large number of digital inputs are required. The 16 digital/O lines can be split into two eight-bit ports, each of which can be used as either input or output.

The controller is designed to respond to 10 input/output commands that can easily be generated within a BASIC program. Detailed examples on how to generate BASIC command statements are included here to help you get the project up and running.

About the Circuit

Use of an 8748 single-chip microcomputer that takes the place of many individual logic chips that would otherwise have been required made it possible to produce the simple circuit design shown in Fig. 1. Microcomputer chip *IC3* is the heart of the circuit. One of the functions of this chip is to emulate a UART configured to transmit and receive at 1,200 baud with no parity, eight data bits and one stop bit. It waits for an eight-bit command byte, receives the byte and then decodes and executes the command represented by the byte.

Another function of *IC3* is to control 16-channel multiplexed eight-bit analog-to-digital (A/D) converter *IC4*. The analog inputs are connected to the controller via plug *P1*. Pull-up resistors *R9* through *R24* are provided for the analog input lines. These resistors can be jumpered to the +5-volt bus when an analog input is used as a simple digital input with a switch or relay-contact closure. Two commands read the eight upper and eight lower analog inputs as though they were digital inputs.

Input/output expander *IC5* is also controlled by *IC3*. The digital lines from *IC5* are split into two eight-bit ports. Each eight-bit port can be either an input or an output. If the first eight digital I/O lines are to be used as outputs, *IC6* must be installed in the controller circuit. Integrated cir-

Semiconductors

D1,D4,D5—1N4004 silicon rectifier diode
 D2,D3,D6—1N914 or similar silicon signal diode
 IC1—74LS04 hex inverter
 IC2—555 timer
 IC3—8748 single-chip microcomputer (must be programmed—see Note below)
 IC4—ADC0817 multiplexed 8-bit analog-to-digital converter
 IC5—8243 input/output expander
 IC6,IC7—ULN2803A Darlington array
 Q1—2N3906 or similar silicon pnp transistor
 VR1—7805 +5-volt fixed regulator

Capacitors

C1—0.01- μ F ceramic disc
 C2,C4,C5—10- μ F, 15-volt electrolytic
 C3,C6,C7,C11,C12—0.1- μ F ceramic disc
 C8,C9—22-pF ceramic disc
 C10—10- μ F, 15-volt electrolytic

Resistors (1/4-watt, 10% tolerance)

R1—1,500 ohms
 R2,R4,R9 thru R24—10,000 ohms
 R3—2,200 ohms
 R5—2,700 ohms
 R6—470 ohms
 R7,R8—4,700 ohms

Miscellaneous

F1—1/4-ampere slow-blow fuse
 P1,P2—DB-25S right-angle, pc-mount connector
 Y1—6.000-MHz crystal with wire leads
 Printed-circuit board or perforated board with holes on 0.1-inch centers

cuits *IC6* and *IC7* are Darlington transistor arrays whose output transistors are capable of sinking 500 milliamperes and will withstand up to 50 volts in the off condition. Maximum continuous power dissipation of a single Darlington pair is 1 watt, and total power dissipation for the complete IC is 2.25 watts. The Darlington arrays can directly interface to light-emitting diodes, relays and solid-state switching modules.

Each Darlington pair in *IC6* and *IC7* has an internal clamping diode,

PARTS LIST (Controller)

and suitable Wire Wrap or soldering hardware (see text); suitable enclosure (see text); sockets or Molex Soldercons for all DIP ICs; 50 single-row male header pins; 20 shorting socket jumpers; two 16-pin male header plugs; 9-volt dc, 250-mA plug-in power supply and matching chassis-mount jack; pc clips for *F1*; machine hardware; hookup wire; solder; etc.

Note: The following items are available from D. Eggert, 3527 E. Edgemore Dr., Appleton, WI 54915: Silk-screened pc board, \$13; programmed 8748 microcomputer chip, \$14; 8740 assembly-language listing, \$5. Also available is a 5/8-inch diskette containing the test fixture BASIC program and machine-language call for TRS-80 Model I for \$5.00. Add \$1 per order for P&H. A home security system BASIC program on 5/8-inch diskette can be obtained for \$5 from B. Pelon, 1609 Grant St., Little Chanutte, WI 54140.

Parts Suppliers

Digi-Key Corp.

701 Brooks Ave. S.
 Thief River Falls, MN 567-1-0677
 1-800-344-4539

Jameco Electronics

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

JDR Microdevices

110 Knowles Dr.
 Los Gatos, CA 95030
 1-800-538-5000

and all eight of these diodes have a common connection at pin 10 of the IC. The common leads of the clamping diodes are available at pins 9 and 21 of plug *P2* and can be used when driving relay coils.

To use eight digital I/O lines as inputs, a 16-pin header plug with jumpers from pin 1 to pin 18, pin 2 to pin 17, pin 3 to pin 16, etc., is installed in the circuit in place of *IC6* or *IC7*. The jumpers connect the I/O lines directly to plug *P2*, which is the digital I/O connector in this circuit. Installation

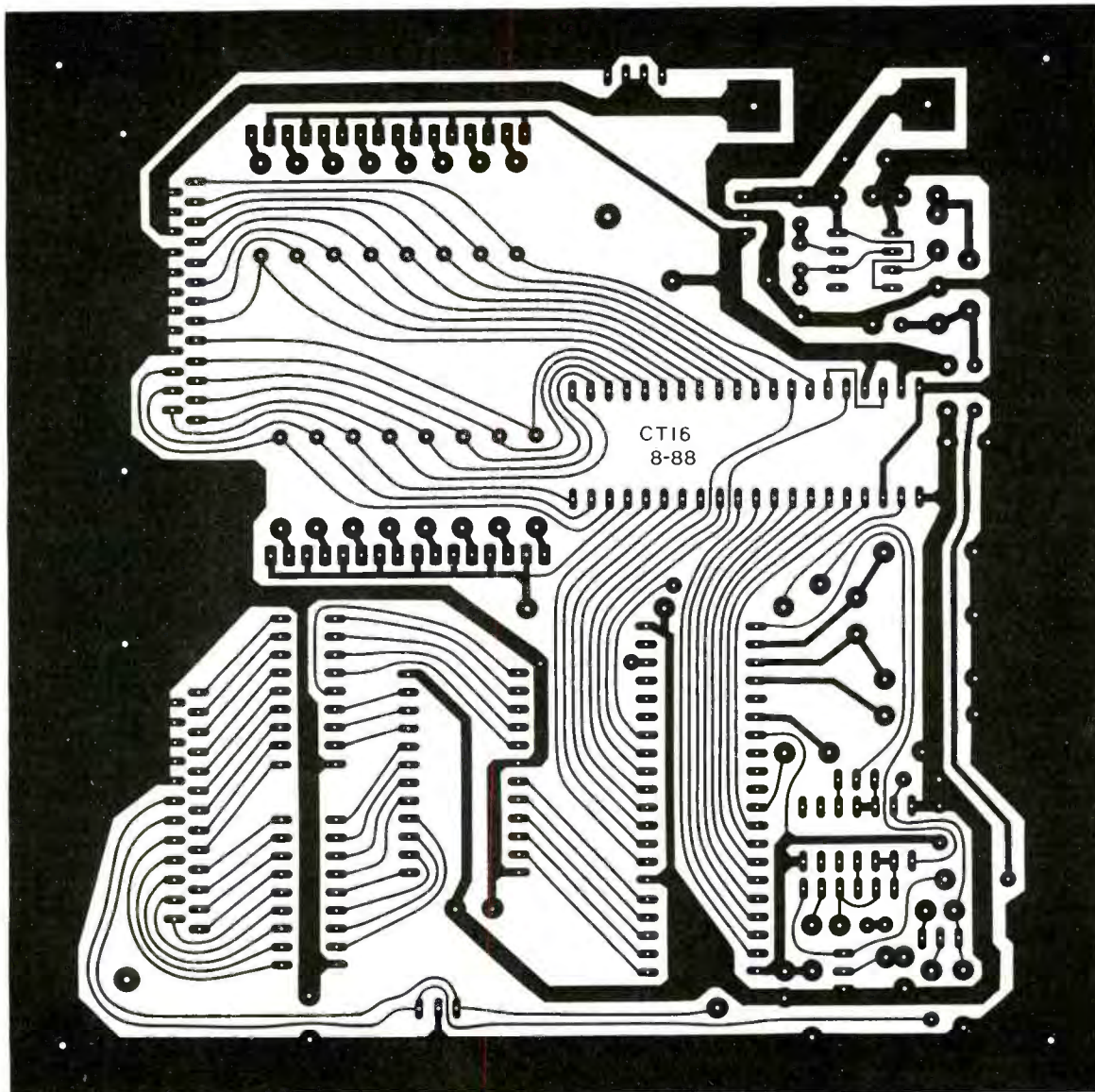


Fig. 2. Actual-size etching-and-drilling guides for top and bottom of double-sided printed-circuit board required for assembling project.

of this plug is in the uppermost portion of the IC socket, which leaves socket pins 9 and 10 unoccupied.

The serial interface circuitry is simple but versatile. Jumper options are provided in the circuit for receive and transmit data-line signal polarity and for TTL or RS-232 interfacing. When *J1* and *J4* are jumpered to position A, the interface is configured for RS-232 communication. When they are jumpered to position B, the interface is configured for TTL communication. Jumpers *J2* and *J3* per-

mit selection of signal polarity. For a typical RS-232 interface configuration, all four jumpers should be in position A.

The bipolar RS-232 output is +5 volts. The negative voltage for the RS-232 output option is generated by the *IC2* 555 timer circuit. The 555 timer is configured here to operate as a high-frequency astable multivibrator. Frequency of oscillation for this circuit is determined by the values selected for *C1*, *R1* and *R2*. Capacitor *C2* is charged through diode *D4*

when pin 3 of *IC2* goes high. When pin 3 goes low, *C4* partially charges through *D5*, diverting some of the charge from *C2*. After powering up the circuit, a few oscillator cycles are required to fully charge *C4*.

Serial data output at pin 34 of *IC3* goes to pin 1 of *IC1* for buffering and optional signal inversion. RS-232 output driver transistor *Q1* receives its input from pin 2 or pin 4 of *IC1*, which pin depending on option jumper *J3*. Diode *D6* in the input circuit of *Q1* ensures that the transistor

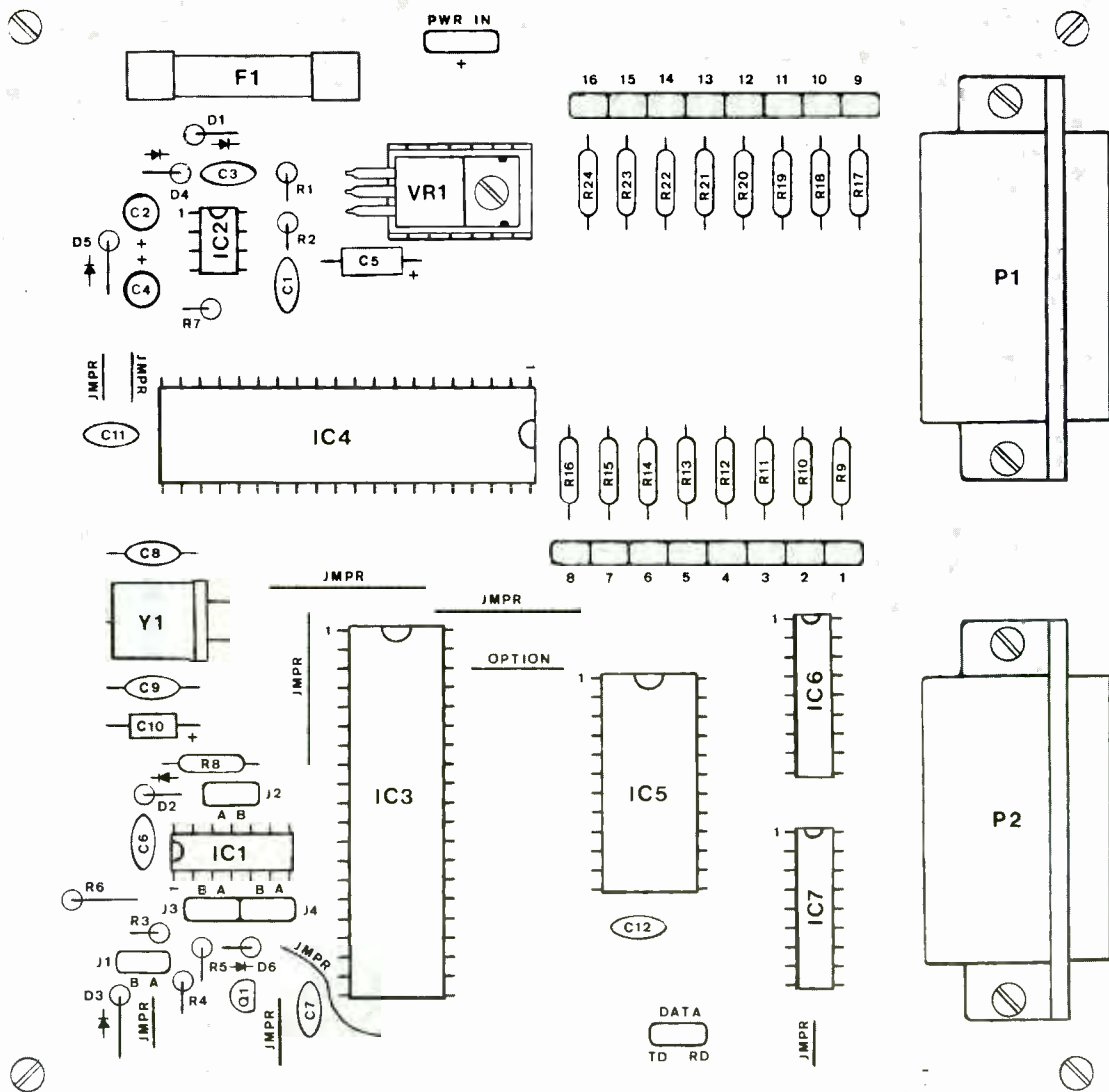


Fig. 3. Wiring diagram for pc board.

gets cut off when the output of *IC1* goes high. You can replace *D6* with a high-efficiency light-emitting diode and change the value of *R5* to 1,800 ohms if you wish to have a convenient activity indicator.

If option jumper *J4* position A is selected, the serial output data will be RS-232 compatible. Selecting position B for this jumper makes the output data TTL compatible.

Received serial data enters the circuit at pin 9 of *P2* and is routed directly to option jumper *J1*. Diodes

D2 and *D3* and resistor *R3* make up a limiting circuit that converts the bipolar RS-232 input signal to a TTL level when option jumper is in position A. Option jumper *J1* is installed at position B when the incoming serial data is TTL compatible. The serial input is then buffered by *IC1* and is optionally inverted by jumper *J2*, after which it goes to pin 1 of *IC3*, where it is processed.

If you wish to expand the I/O capabilities of the project, you can use two controller boards in parallel. In

this event, the serial receive and transmit data lines must be connected in parallel and an option jumper must be connected from pin 38 of *IC3* to ground in one of the controller circuits. With this arrangement, when the most-significant bit (MSB) in the command byte is a logic 1, the controller with the option jumper installed will execute the command. When the MSB in the command byte is a logic 0, the controller without the jumper will execute the command.

Fuse *F1* and rectifier diode *D1* were designed into the circuit to prevent damage to the circuit in the event that the polarity of the input power is accidentally reversed. Fixed 5-volt regulator *VRI* easily handles the modest 90 milliamperes of current drawn by the circuit.

Construction

Though you can build this circuit on perforated board that has holes on 0.1-inch centers using suitable Wire Wrap or soldering hardware, a printed-circuit board is highly recommended. You can fabricate your own board using the actual-size etching-and-drilling guide in Fig. 2 or purchase a ready-to-wire board from the source given in the Note at the end of the Parts List.

When you have the pc board ready to be wired, place it in front of you in the orientation shown in Fig. 3. Begin populating the board by installing and soldering into place the various jumpers, plugs and sockets for all DIP ICs. Do not install the ICs in their respective sockets until after preliminary voltage checks have been made.

Continue populating the board by installing and soldering into place the resistors, then the capacitors, diodes and transistor. Make certain before you solder any leads to the copper pads on the bottom of the board that the electrolytic capacitors and diodes are properly oriented and that the transistor is properly based.

Note in Fig. 3 that some resistors and all diodes mount on-end. The resistors that mount on-end are *R1* through *R7*. All other components mount flat against the top surface of the board.

If you decided to use the transmit-activity LED in your controller, omit diode *D6* and use an 1,800-ohm resistor for *R5*. Install the LED via 3-inch-long hookup wires attached to each lead and insulated with small-diameter heat-shrinkable tubing.

Make sure you install the LED in the *D6* holes in the same polarity as indicated for the diode before soldering the wires to the copper pads on the bottom of the board.

Bend the leads of the crystal at a 90-degree angle to one side of its case. Plug the leads into the appropriate holes in the board and push the crystal flat against the top surface of the board. Using heat judiciously, solder the leads of the crystal to the copper pads on the bottom of the board.

Install and solder into place the two clips for the fuse. When the clips cool, plug into them the fuse. Bend the pins of the voltage regulator at a 90-degree angle toward the rear of the case at the point where they just begin to widen. As mentioned above, there is no need to use a heat sink for the voltage regulator. However, if you decide to use one, use heat-transfer compound between the heat sink and regulator. Plug the pins of the regulator into the appropriate holes in the board and secure it in place with a 4-40 × ¼-inch machine screw, nut and lockwasher. Then carefully solder the pins of the regulator into place.

Finish up wiring the circuit-board assembly by installing and soldering into place the various wire jumpers. Regardless of how you configure the circuit, nine such jumpers must be installed. Use solid bare hookup wire or cut-off resistor leads for the short jumpers and insulated solid hookup wire for the long ones. Route the jumper wire that passes the lower-left corner of *IC3* as shown. Now install and solder into place any other jumper wires that will be required when the circuit is put into service.

As mentioned above, you can increase the I/O capabilities of the controller by using two circuit-board assemblies in parallel. If you are doing this, install the OPTION jumper wire shown in the center of Fig. 3 on only one board.

For the time being, place the head-

ers on the position A jumper pins for *J1* through *J4* so that they will not become lost.

Using heat judiciously, pool a small blob of solder on the top surface of the case of the crystal in line with the unoccupied hole in the circuit board. Pass one end of a bare hookup wire through the hole from the solder side of the board and bend it over the case of the crystal. Again using heat judiciously, solder the end of this wire to the case of the crystal using the solder you previously pooled on it. Then solder the other end of the wire to the copper pad on the bottom of the board and clip away any excess.

Except for installation of the DIP ICs in their respective sockets, this completes wiring of the circuit-board assembly.

You can use any type of enclosure that will comfortably accommodate the circuit-board assembly. If you can obtain one, a case that measures roughly 6½ inches square by about 1½ inches deep (2½ inches deep if you plan on housing two controller assemblies in the same enclosure) is ideal. Also, though you can use an all-metal enclosure for the project, it is better to use one that is mostly or all plastic to make it easier for you to make the cutouts for plugs *P1* and *P2*.

Begin machining the enclosure by placing the circuit-board assembly inside it and marking on its floor where to drill the four mounting holes for the board. Mark the locations of the slots for *P1* and *P2* (for both controllers if you built two and are housing them in the same enclosure). Also mark the location in which to drill the jack for the power supply you will be using. Locate this jack near where *F1* will be when the circuit-board assembly is mounted inside the enclosure.

Remove and set aside the circuit-board assembly. Then drill the four holes through the floor and jack hole through the side wall. Use whatever tools are necessary to carve out the two connector slots. Also, if you

have decided to use the LED activity indicator, drill a hole for it in a location where the LED will be easily seen with the project in operation. Replace the circuit-board assembly inside the enclosure and check that the connector slots are large enough and accurately located.

Before actually mounting the circuit-board assembly inside the enclosure, double check all component installations (except the DIP ICs, which should not be installed at this time) for appropriate values or types and orientations. When you are satisfied that all component installations are correct, turn over the circuit-board assembly and carefully examine it for missed solder connections, poorly soldered connections and inadvertent solder bridges, especially between closely spaced IC pads and conductors. Solder any missed connections and reflow the solder on any connection that appears suspicious. Use desoldering braid or a vacuum-type desoldering tool to remove solder bridges.

When you are certain that the circuit-board assembly is properly wired and soldered, mount it inside the enclosure using 1/4-inch metal spacers and 4-40 machine hardware or plastic pc board mounting clips. Then install IC2 in its socket; make sure it is properly oriented and that no pins overhang the socket or fold under between IC and socket.

Now prepare the cables that will interconnect the project with your computer. Prepare as many cables as needed, depending on whether you built one or two controller circuit-board assemblies. You can make each cable from scratch or start with already made up cables. Wiring details for the cables are shown in Fig. 4, one for using a DB-9S connector that goes to an AT or compatible computer and the other for using a DB-25S connector that goes to PC/XT or compatible computer. Only three conductors are required in either case, one for transmit, one for

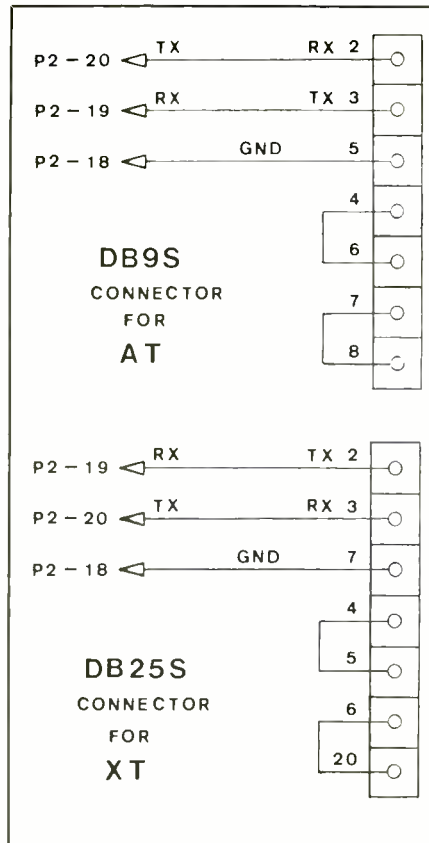


Fig. 4. Wiring diagrams for cable-to-connector hookups for DB-9 (upper) and DB-25 (lower) serial connectors.

receive and one for the common ground return. Choose the wiring arrangement that suits the particular computer you will be using with the controller project.

Command Summary

Figure 5(A) graphically illustrates the bit positions of the variables in the following example BASIC statements. The first statement is an example of how command 1 is used:

```
PRINT #1,CHR$((CH-1)*4) + CN)
```

“Command Number” CN is bits 0 and 1 of the output command byte. Command Number 0 is not used. If CN = 0, the controller will ignore the command if only one controller board is used and the OPTION jumper is not installed. If CN = 1, the con-

troller will respond by sending back to the computer an analog input using “CH” as analog “CHannel” numbers 1 through 16.

In the above sample statement, 1 is subtracted from the channel number (CH) to convert it to a four-bit value from 0 to 15. It is also multiplied by 4 so that it occupies bit positions 2 through 5 in the output command byte. For command number 2, the variable OS is added to the BASIC statement in the following manner:

```
PRINT#1,CHR$((OS*64)
+ ((CH-1)*4) + CN)
```

If CN = 2, the controller changes the state of a digital output using CH as digital output channel numbers 1 through 16.

OS is the “Output State.” If OS = 0, the digital output addressed by CH is turned off, and if OS = 1, it will be turned on. In the above statement, OS is multiplied by 64 so that it occupies bit position 6 in the output command byte.

After digital output command 2 is sent, the controller sends back to the computer an “ACK” code (6) to ACKnowledge that the command was received and executed. The above statement can be used for both commands 1 and 2. The OS variable is not used by analog input command 1, but it can remain in the statement. OS must be either 0 or 1.

The variable “MX” is now added to the BASIC statement as follows:

```
PRINT#1,CHR$(MX*128)+
(OS*64)+((CH-1)+CN)
```

Before decoding any commands, the controller always checks bit 7, which is the most-significant bit (MSB), to see if the command is to be executed or ignored. This depends on whether or not the jumper from pin 38 of IC1 to ground is installed.

MX is multiplied by 128 so that it occupies position 7 in the output command byte. MX must be either 0

$$((MX * 128) + (OS * 64) + ((CH - 1) * 4) + CN) \quad ((MX * 128) + ((SC - 1) * 4) + CN)$$



Fig. 5. Bit positions for variables in sample BASIC statements used in text.

or 1 if it is included in the statement.

The remaining eight commands will use the following BASIC statement example:

```
PRINT#1,CHR$((MX*128)+
(SC-1)*4)+CN)
```

Figure 5(B) graphically illustrates the bit positions of the variables in this statement. Variable SC, the "Sub Command," replaces CH in the previous statements and is sub-command statement numbers 1 through 8.

CN must equal 3 for all of the following eight sub-commands. In the above statement, 1 is subtracted from sub-command SC to convert it to a three-bit value from 0 to 7. It is also multiplied by 4 so that it occupies positions 2, 3 and 4 of the output command byte.

If SC = 1, the controller responds by reading the lower eight digital I/O lines as inputs and sends them to the computer as an eight-bit byte. If SC = 2, the controller reads and sends the upper eight lines.

If SC = 3, the controller responds by outputting the next byte that it receives to the lower eight digital I/O lines. If SC = 4, the controller outputs the next byte received to the upper eight I/O lines.

For sub-commands 3 and 4, the controller acknowledges the command byte and data byte by sending an ACK code (6) back to the computer after receiving each byte.

If SC = 5, the controller responds

by reading the lower eight analog inputs as if they were digital inputs and sends them to the computer as an eight-bit byte. If SC = 6, the controller reads and sends the upper eight analog inputs.

For sub-commands 5 and 6, if an analog input is greater than 2 volts, it will be a logic 1; if it is less than or equal to 2 volts, it will be a logic 0.

If SC = 7, the controller responds by reading the lower eight analog inputs and sends them to the computer as a string of eight consecutive bytes. If SC = 8, the controller reads and sends the upper eight analog inputs.

The ACK code (6) that is sent back to the computer when command 2 and sub-commands 3 and 4 are executed may appear to be a waste of time. However, this is the only way to test for proper communication between the computer and controller. It is not necessary to input the ACK code after execution of these commands.

If the ACK code is not input, you must generate some amount of delay to allow it time to send the ACK code and get ready to receive the next byte. It cannot receive and transmit data simultaneously as your computer can! Maximum reliable operating speed can be obtained by inputting the ACK code before sending the next output command byte to the controller.

The following decimal values shown in parentheses can be used as the output command byte instead of using the above sample statement for

the sub-commands: SC1 = (3), SC2 = (7), SC3 = (11), SC4 = (15), SC5 = (19), SC6 = (23), SC7 = (27) and SC8 = (31). If two controller cards are used to expand the I/O capabilities of the project, the decimal value 128 must be added to this value if the sub-command is intended for the controller that has the OPTION jumper installed in it.

Test Fixture

The test fixture mentioned at the beginning of this article is shown in Fig. 6 plugged into P1 and P2 on the controller board. Schematic details of the circuitry for the test fixture are shown in Fig. 7. References to P1 and P2 in Fig. 5 indicate where the various connections are to be made to the analog input and digital output lines on the controller circuit-board assembly via the DB-25 connectors.

Power for the analog and digital inputs of the test fixture is obtained from the same 9-volt dc source used for powering the controller circuitry. Voltage regulator VRI reduces the incoming 9 volts to the required 5 volts dc. **Caution:** Never apply a potential that is greater than 5 volts dc to any analog or digital input line in the controller. Otherwise, one or more solid-state devices in the project is almost certain to result.

You can assemble this circuit on a rigid 1/8-inch thick plastic panel or 16-gauge metal panel, using all-new components or components you may already have on hand. Switches S1

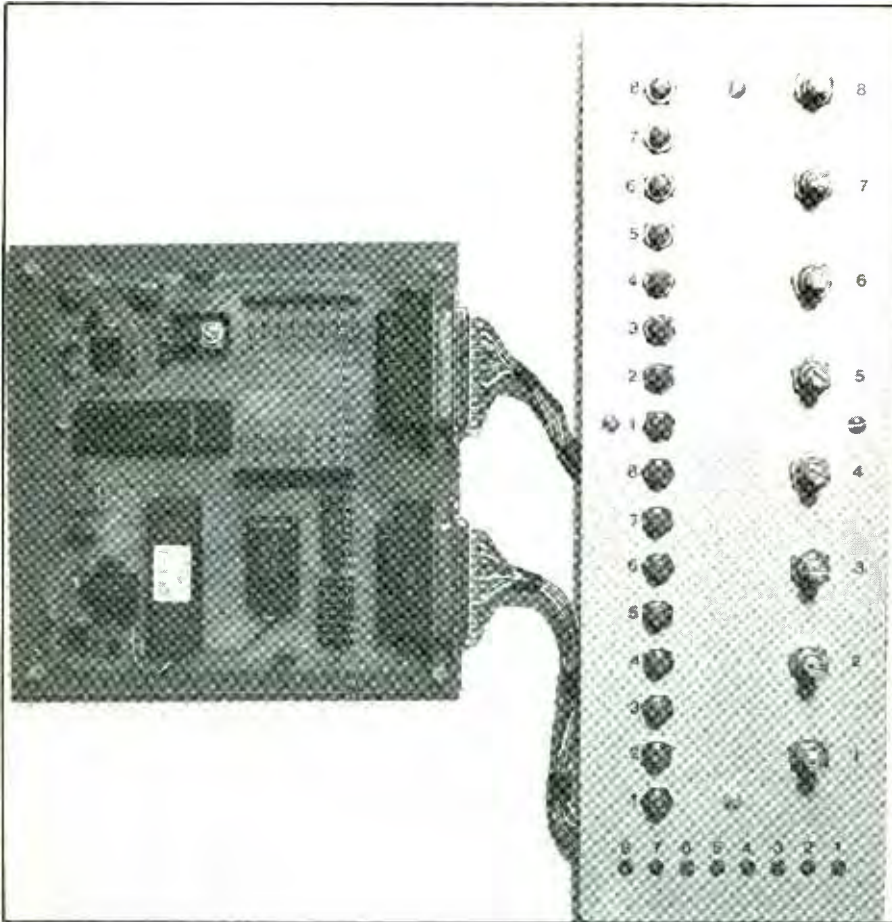


Fig. 6. Wired project and test fixture shown connected together. Not shown is three-conductor serial-in, serial-out and ground cable that goes to computer.

through *S16* are ordinary normally-open, momentary spst pushbutton types. Use any type of visible light-emitting diodes for *D1* through *D8*. Also, almost any type of panel-mount potentiometers can be used for *R17* through *R24*. Resistance values are not critical to circuit performance, but it is recommended that you limit value selections to the range from 1,000 to 50,000 ohms.

As you can see in Fig. 7, the circuitry for the test fixture is simple and straightforward. No circuit board—printed-circuit or otherwise—is needed to wire together the circuit. In fact, you mount all components either directly on the metal or plastic panel or, as in the case of resistors *R1* through *R8* and the cathode connections of light-emitting diodes *D1*

through *D8*, via suitable terminal strips secured to the panel. Use terminal strips that have no lugs electrically connected to their mounting tabs.

Machine the plastic or metal panel as needed. That is, drill the mounting holes for the pots, switches and LEDs, making the latter just large enough to accommodate the LEDs with a bit of pressure. Also, drill holes for mounting any terminal strips you will be using and the voltage regulator and four holes with which to mount the assembly.

After machining a metal panel, deburr all holes to remove sharp, ragged edges. Scrub the panel to remove dirt, grease and grime and thoroughly dry it. Then use a dry-transfer lettering kit or a tape labeler to label each switch, LED and control hole as

shown. If you use dry-transfer lettering, protect it from abrasion with two or more light coats of clear acrylic spray. Allow each coat to dry before spraying on the next.

When the panel is ready, mount the pots, switches and LEDs in their respective holes and the terminal strips near the LEDs. Use the hardware supplied with the switches and pots to mount these components. If any LEDs do not remain in place by friction fit, use a small daub of fast-setting epoxy cement on each.

If you use a plastic panel, place a small heat sink between it and *VR1*. If the panel is metal, the panel itself will be the heat sink, but use heat-transfer paste and an insulator between the case of the regulator and panel and appropriate insulating hardware to mount the regulator.

When all components have been physically mounted in place on the panel, wire them together exactly according to Fig. 7, using point-to-point wiring. Check off each lead and conductor run as you make it, and make certain that the LEDs and *CI* are properly polarized and that appropriate connections are made to voltage regulator *VR1*.

Start wiring the circuit with *R1* through *R8*, connecting one end of each of these resistors to its own separate terminal-strip lug. Do not solder any of these connections just yet. Terminate the free leads of all eight resistors in a single terminal-strip lug. Also crimp one end of an appropriate length of hookup wire at this lug (do not solder the connection yet). If a single lug will not accommodate all these leads, the wire and one more wire, use two lugs, bridging them with a length of wire.

Slip a 1-inch length of small-diameter heat-shrinkable tubing onto the free end of the wire connected to the common lug of *R1* through *R8*. Crimp and solder the wire to the INPUT (+9V) lead of *VR1*. Slide the tubing over the connection and shrink it into place.

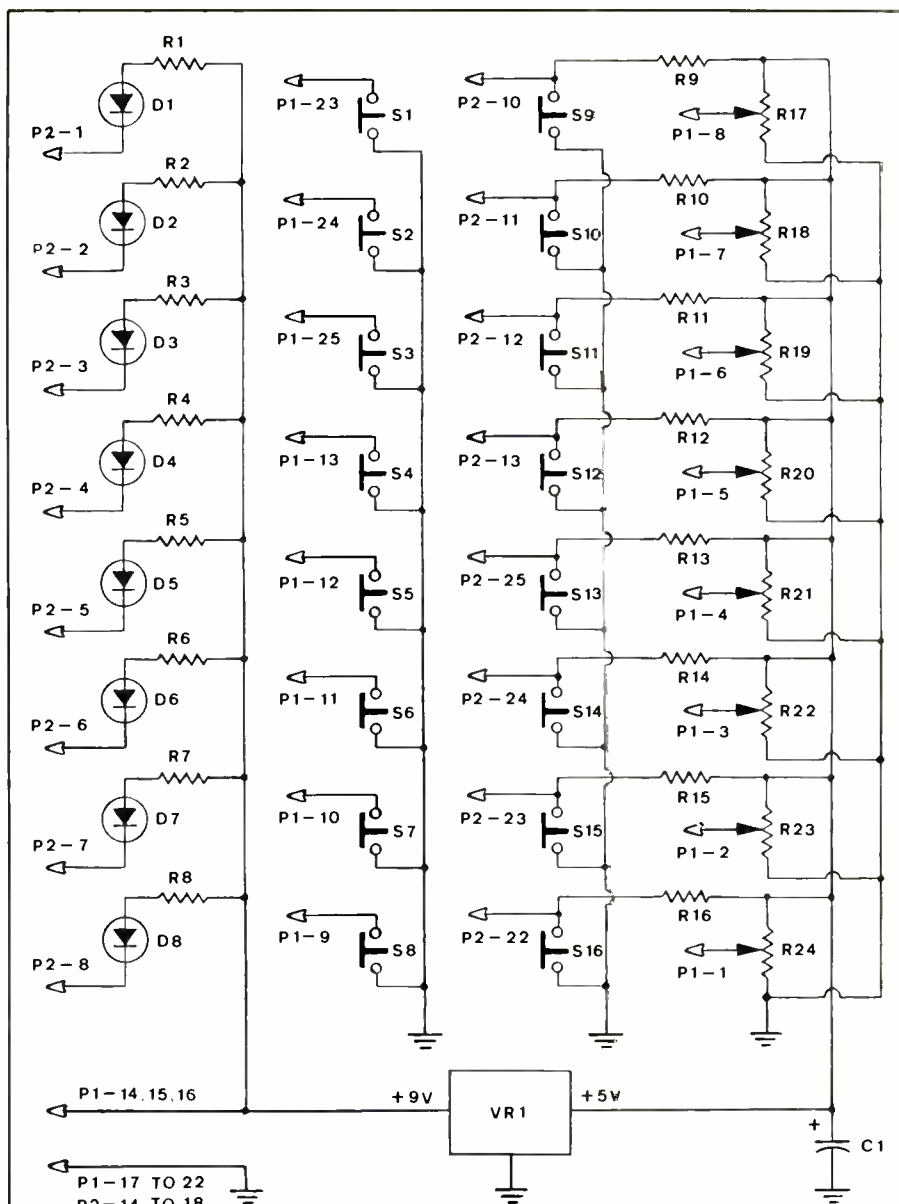
Interconnect one lug of *S1* through *S8* with a continuous length of bare hookup wire, terminating the end of this wire in a chassis grounding lug. Solder the connections at all switch lugs but not at the grounding lug. Do the same for one lug of *S9* through *S16*, and repeat with one lug of the potentiometers (use lug 3, the one on the right when viewing the pots from the rear, for this run). Crimp the negative (-) lead of *C1* to the lug. Again, do not solder the connection in both cases.

Crimp but do not solder one end of an appropriate-length hookup wire to the grounding lug. Slip a 1-inch length of heat-shrinkable tubing over the free end of this wire and crimp and solder the wire to the OUTPUT (+5V) lead of the voltage regulator. Slide the tubing over the connection and shrink into place.

Crimp and solder one lead of *R9* through *R16* to lug 1 of the potentiometers as shown in Fig. 7. Lengthen the other leads of the resistors as follows. First trim these leads to 1/2 inch in length and form a small hook in the stub. Crimp and solder appropriate-length hookup wires to the stubs. Slip a 1-inch length of heat-shrinkable tubing over the free ends of the wires. Slide the tubing over the connections to completely cover them and shrink into place. Crimp but do not solder the free ends of these wires to the unoccupied lugs of switches *S9* through *S16*.

Clip the cathode lead of each LED to a length of 1/2 inch and form a small hook in the remaining stubs. Crimp and solder one end of appropriate-length hookup wires to the stubs. Slip a 1-inch length of heat-shrinkable tubing over the free end of each wire. Slide the tubing up over the soldered connections and flush against the bottom of the LED cases and shrink into place.

Crimp and tack-solder the free ends of the cathode wires to separate terminal-strip lugs. Then repeat the entire operation for the anode leads



PARTS LIST (Test Fixture)

Semiconductors

D1 thru D8—Light-emitting diode
 VR1—7805 + 5-volt fixed regulator

Capacitors

C1—10- μ F, 15-volt electrolytic

Resistors (1/4-watt, 10% tolerance)

R1 thru R8—470 ohms
 R9 thru R16—10,000 ohms
 R17 thru R24—1,000-ohm to 50,000-ohm panel-mount potentiometer (see text)

Miscellaneous

S1 thru S16—Momentary-action push-button switch (see text)
 Two DB-25P connectors; connector to match serial I/O connector on computer (see text); terminal strips (see text); grounding chassis lug; rigid plastic or metal sheet (see text); small-diameter heat-shrinkable tubing; 1/2-inch spacers; machine hardware; stranded hookup wire; solder; etc.

Fig. 7. Schematic diagram of test-fixture circuitry.

of the LEDs, this time terminating the free ends of the wires at the terminal strip lugs to which *R1* through *R8* are connected.

You need two short 25-conductor cables, each terminated at one end in a DB-25 connector to mate with *P1* and *P2* on the controller circuit-board assembly. You also need another cable with three conductors and terminated in a DB-25 connector that plugs into the serial I/O connector on your computer. You can make these cables using stranded hookup wire and appropriate connectors.

When making the cables, strip $\frac{1}{4}$ inch of insulation from only one end of each wire. Tightly twist together the fine conductors that are exposed and sparingly tin with solder. You need 35 wires that are 30 inches in length and three wires that are 36 inches in length. Prepare both ends of the latter three wires as described.

For the following steps, use the 30-inch-long wires. Crimp and solder the prepared end of eight wires to the lugs to which the cathodes of the LEDs are connected. Similarly, crimp and solder the prepared end of eight more wires to the unoccupied lugs of *S1* through *S8*, another eight wires to the lugs of *S9* through *S16* to which *R9* through *R16* are connected, and yet another eight wires to the center lugs of the potentiometers.

Now crimp the prepared ends of the three remaining wires to the lugs to which all eight leads of *R1* through *R8* are connected and solder the connection. Then crimp and solder the prepared end of the remaining two 30-inch-long wires to the chassis grounding lug.

After making all connections, neatly bundle together the wires for *P1* and use plastic cable ties, waxed lacing cord or electrical tape to secure the bundle, ending about 6 inches from the unprepared end. Do the same for the wires for *P2*. Trim the wires in both bundles all to the same length, leaving at least 3 inches of un-

bundled length in both cases.

Strip $\frac{1}{8}$ inch of insulation from the end of each wire in both bundles. Tightly twist together the exposed conductors and sparingly tin with solder. Use heat judiciously to minimize damage to the insulation.

Use your soldering iron to carefully heat the pin cups on both DB-25 connectors that will mate with *P1* and *P2* on the controller board and flow a small amount of solder into each. Make sure you get solder into only the cups and do not create solder bridges between the closely-spaced cups. To terminate each wire from the cables, simply heat the cup and plug into it the free end of the wire, remove the heat and allow to cool undisturbed.

Begin wiring the connectors by tack-soldering lengths of bare hookup wire to the pins 14, 15 and 16 posts and the pins 17 through 22 posts of the connector that will plug into *P1*. Position these wires near where the posts enter the plastic form that holds them in place on the DB-25 connector that will plug into *P1* on the controller board. Do the same for the pins 14 through 18 posts for the connector that will plug into *P2* on the controller board.

Retrieve the three 36-inch-long wires you prepared. Connect one end of these wires to pins 17, 18 and 19 of the *P2* connector. Use electrical tape to secure these three wires together in a neat bundle along their entire length to within 3 inches of the free ends. Using the same procedure detailed above, terminate the free ends in a DB-25 or other connector that mates with the serial I/O connector on your computer. Terminate each wire at the appropriate serial data in, serial data out and ground connector pin for your computer. Use an ohmmeter or audible signal tracer to identify which wire goes to which pin in all cases.

Solder the ends of the cables to their respective connector pin cups. Again, make sure you do not create

unwanted solder bridges between the closely-spaced cups. Also, make sure that each connection is electrically and mechanically secure. When you are finished, mount the test-fixture assembly on a metal or plastic plate, using $1\frac{1}{2}$ -inch spacers and suitable-length machine screws, nuts and lockwashers.

Using the Test Fixture

After you double-check your work on the test fixture and are satisfied that it is correctly wired, plug the DB-25 connectors at the ends of its cable into their respective connectors on the controller board and the serial I/O connector on your computer. Plug the 9-volt dc power supply into its jack on the project and turn on both computer and controller.

Boot up your computer and load into memory the BASIC language interpreter. Then key into the computer the program listing given elsewhere in this article. Save the program to disk and then recall and RUN it. As the program runs, it will display on the video monitor screen of the computer three vertical columns of video inputs. The column on the left displays analog inputs 1 through 8. These are fed in as digital inputs using sub-command 5 and are displayed as 0s and 1s. The decimal value of all eight inputs combined is displayed just below this column.

If normally-open switches are used for *S1* through *S8* in the test fixture, all 1s will be displayed on-screen in the column on the left. Pressing any of these switches to close its contacts causes the position in the column for that input to become a 0.

The middle on-screen column displays analog inputs 9 through 16. These are input using command 1. Each input is displayed as a decimal value that is between 0 and 255, corresponding to the input potential of from 0 to 5 volts. You can change the values displayed for these inputs by varying the settings of the potentiometer.

**ORDER
FROM
JAN
FOR
CRYSTAL
CLEAR
COMMUNICATION**



- YOUR RELIABLE SOURCE SINCE 1965
- QUICK DELIVERY
- PRICED RIGHT
- EMERGENCY ORDER SERVICE AVAILABLE

FOR FREE CATALOG,
CALL OR WRITE:

JAN CRYSTALS

P.O. BOX 06017

FORT MYERS, FL 33906

VISA

(813) 936-2397

MasterCard

TOLL-FREE: 1-800-237-3063

IN FLORIDA: 1-800-226-XTAL
FAX ORDERS: 1-813-936-3750

CIRCLE 25 ON FREE INFORMATION CARD

**LEARN VCR
CLEANING/MAINTENANCE/REPAIR
EARN UP TO \$1000 A WEEK, WORKING
PART TIME FROM YOUR OWN HOME!**



Secrets
Revealed!

NO Special
Tools or
Equipment
Needed.

**EARN UP TO \$80
AN HOUR AND
MORE!**

**THE MONEY MAKING OPPORTUNITY
OF THE 1990'S**

IF you are able to work with common small hand tools, and are familiar with basic electronics (i.e. able to use voltmeter, understand DC electronics)

IF you possess average mechanical ability, and have a VCR on which to practice and learn . . . then we can teach YOU VCR maintenance and repair!

FACT: up to 90% of ALL VCR malfunctions are due to simple MECHANICAL or ELECTRO-MECHANICAL breakdowns!

FACT: over 77 million VCRs in use today nationwide! Average VCR needs service or repair every 12 to 18 months!

Viejo's 400 PAGE TRAINING MANUAL (over 500 photos and illustrations) and AWARD-WINNING VIDEO TRAINING TAPE reveals the SECRETS of VCR maintenance and repair— "real-world" information that is NOT available elsewhere!

Also includes all the info you'll need regarding the BUSINESS-SIDE of running a successful service operation!

FREE INFORMATION
CALL TOLL-FREE 1-800-537-0589

Or write to:

Viejo Publications
3540 Wilshire BL. STE 310
Los Angeles, CA 90010 Dept ME

CIRCLE 44 ON FREE INFORMATION CARD

Test Fixture BASIC Program Listing

```

100 CLS: PLC=0: OPEN"COM1:1200,N,8,1" AS #1
200 PRINT#1,CHR$(19);: I=ASC(INPUT$(1,#1))
300 LOCATE 16,14: PRINT I," ";
400 COL=15: ROW=7: GOSUB 2700
500 PRINT#1,CHR$(7);: I=ASC(INPUT$(1,#1))
600 LOCATE 16,46: PRINT I," ";
700 TI=I: COL=47: ROW=7: GOSUB 2700
800 CN=1: ROW=14: COL=30
900 FOR CH=9 TO 16
1000 PRINT#1,CHR$(((CH-1)*4)+CN);: I=ASC(INPUT$(1,#1))
1100 LOCATE ROW,COL: PRINT I," ";
1200 ROW=ROW-1: NEXT CH
1300 PLC=PLC+1: IF PLC=256 THEN PLC=0
1400 PRINT#1,CHR$(11);: I=ASC(INPUT$(1,#1))
1500 PRINT#1,CHR$(PLC);: I=ASC(INPUT$(1,#1))
1600 LOCATE 16,30: PRINT PLC," ";
1700 IF TI (<) 126 THEN GOTO 200
1800 PRINT#1,CHR$(11);: I=ASC(INPUT$(1,#1))
1900 PRINT#1,CHR$(0);: I=ASC(INPUT$(1,#1))
2000 FOR T = 1 TO 1000: NEXT T
2100 CN=2: FOR OS = 1 TO 0 STEP -1
2200 FOR CH = 1 TO 8
2300 PRINT#1,CHR$((OS*64)+((CH-1)*4)+CN);
2400 FOR T = 1 TO 200: NEXT T
2500 I=ASC(INPUT$(1,#1)): NEXT CH: NEXT OS
2600 GOTO 200
2700 BV=128: FOR BIT=1 TO 8
2800 IF (I-BV)=0 THEN GOSUB 3300 ELSE GOSUB 3400
2900 NEXT BIT
3000 FOR X=1 TO 8
3100 LOCATE ROW,COL: PRINT SW(X);
3200 ROW=ROW+1: NEXT X: RETURN
3300 SW(BIT)=1: I=I-BV: BV=BV/2: RETURN
3400 SW(BIT)=0: BV=BV/2: RETURN

```

meters on the test fixture. The decimal value displayed below the middle column is the program loop counter and increments one step each time the main program loops back to the beginning.

The program loop count value is also output to digital I/O lines 1 through 8, using sub-command 3. The loop count is displayed in binary by the eight LEDs that are controlled by these lines in the test fixture.

The right column displays digital I/O lines 9 through 16 and are input using sub-command 2. Switches S9 through S16 on the test fixture correspond to this column. Their conditions are displayed in the same manner as described for the left column.

Analysis of the test program is as follows:

- Line 100 clears the screen, sets the program loop counter to 0 and opens COM1 for communication at 1,200 baud with no parity, eight data bits and one stop bit.
- Line 200 outputs sub-command 5 and then pauses for the input character from the controller.
- Line 300 prints the received input character.
- Line 400 sets up the screen display location variables and then executes a subroutine at line 2700 that displays each individual bit of the eight-bit input character as a 0 or a 1 on the screen in a vertical column.
- Lines 500, 600 and 700 are identical

to lines 200, 300 and 400, except that sub-command 2 is used and the inputs are displayed in the right column on the screen. This input character is also saved as variable "TI" to be tested later in the program.

- Line 800 sets the command to 1 and sets up screen location variables before executing lines 900 through 1200, which inputs and displays analog inputs 9 through 16 in the center column on the screen.

- Line 1300 increments the program loop counter.

- Lines 1400 and 1500 use sub-command 3 to output the program loop count to digital I/O lines 1 through 8.

- Line 1600 prints the program loop count below the middle column on the screen.

- Line 1700 tests variable "TI," which is the saved value of digital I/O lines 9 through 16. If switches *S9* and *S16* on the test fixture are pressed and held closed simultaneously (these switches are normally open), the routine at line 1800 will begin executing.

- Lines 1800 through 2700 make up a subroutine that clears digital I/O lines 1 through 8 with sub-command 3 and then uses sub-command 2 to sequence these lines on and then off.

Hardware Interfacing

You can use the test fixture whose circuitry is shown in Fig. 7 as a simple guide when interfacing the controller to your unique applications. For example, you can replace *S1* through *S16* with the contacts of relays or magnetic reed switches. Potentiometers *R17* through *R24* could be replaced with simple resistance voltage-divider networks for monitoring potentials that do not exceed 5 volts dc. (Remember the warning against applying potentials greater than 5 volts dc to the analog or digital lines!)

You can use a thermistor as one of the resistive elements in any voltage-divider network to monitor tempera-

ture. Alternatively, a photocell can be used in any network to detect changes in light levels.

A small relay coil or solid-state switch module can be substituted for light-emitting diode *DI* and current-limiting resistor *R1*. Solid-state switch modules typically isolate the input from the load for protection and can be obtained for both ac and dc switching applications.

As you experiment with this project, you will undoubtedly come up with other interfaces to suit particular applications. There is no limit to what you can do with this project, depending on your imagination and technical expertise.

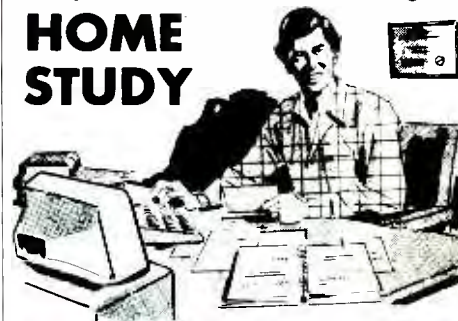
Although this project was designed primarily for use with IBM PC/XT/AT and compatible computers, the controller circuit-board assembly can be used with any computer that has an RS-232 serial interface. To this end, the project was originally conceived for use on a Radio Shack TRS-80 Model I-compatible system.

On other popular home computers, a major obstacle that might be encountered is that the BASIC programming language structure might permit a string to be only input from a serial port. From the foregoing, it should be obvious that your system must be able to input and output one character at a time via the serial port for the project to operate properly. The character must also be eight bits in length.

For computers that have the above limitation, a machine-language call can be used to perform the serial input/output function. For the TRS-80 Model I, a short machine-language user call can be used to output the command byte first and then pause for the serial input character before returning to BASIC. If no character is received within a specified period of time, on the order of milliseconds, the routine would return with a value that is greater than a decimal 255, indicating that a communication failure has occurred. **ME**

Put Professional Knowledge and a
COLLEGE DEGREE
in your Technical Career through

**HOME
STUDY**



Add prestige and earning power to your technical career by earning your Associate or Bachelor degree through directed home study.

Grantham College of Engineering awards accredited degrees in
electronics and computers.

An important part of being prepared to *move up* is holding the right college degree, and the absolutely necessary part is knowing your field. Grantham can help you both ways—to learn more and to earn your degree in the process.

Grantham offers two degree programs—one with major emphasis in **electronics**, the other with major emphasis in **computers**. Associate and bachelor degrees are awarded in each program, and both programs are available completely by correspondence.

No commuting to class. Study at your own pace, while continuing on your present job. Learn from easy-to-understand lessons, with help from your Grantham instructors when you need it.

Write for our free catalog (see address below) or telephone us at (213) 493-4421 (no collect calls) and ask for our "degree catalog."

Accredited by
the Accrediting Commission of the
National Home Study Council

GRANTHAM
College of Engineering
10570 Humbolt Street
Los Alamitos, CA 90720