Programmable Thermostat

Build your own versatile controller.

TREVOR COX

Most people turn down the thermostat at night to save on heating bills and to sleep better. Most people now and then also forget to turn it down. And no one likes to wake up to a cold house in the morning. Offices and multi-user buildings such as meeting halls stand to save even more by turning down the heat when no one is using the building. Yet more often than not, it is impossible to give that responsibility to one person, with the result that the building will be kept heated for no reason. If the thermostat itself knew that everyone went home at 5:00, or that it should have the gymnasium warmed up for the aerobics class in the morning, energy would be saved and life would seem just that little bit easier. In this age where we can program our telephones, TVs, ovens, lamps, computers, and compact disc players, why not program our thermostats as well?

Let's look at what sort of electronics is necessary to make a Programmable Thermostat. It needs a switched output to tell the furnace or heater to turn on and off. Some way of sensing the room temperature is required. A keyboard to enter the time, to program it, and to manually change the temperature setting should be provided, and hopefully these things could be done simply. Of course it needs a display to show the room temperature and the temperature setting, and maybe act as a clock as well. But there is a lot more to this than just those things. All the temperature settings with the corresponding time they come into effect must be stored somewhere.

Hysteresis (the difference between the temperature the room is allowed to drop before the heat comes on, and the temperature it rises to before the heat shuts off) must be taken into account. The circuit has to see when it is time to change the thermostat setting and then change it. In short, this is a very complex task, and we would like it to fit in a compact case which can mount unobtrusively on a wall, and because we may want to build one as a hobbyist project, it should not be too complicated or expensive.

The Design

If you are trying to visualize this thermostat circuitry made in a conventional way, you may be thinking of a very complex digital design. To build this project from standard logic gate ICs would not meet our criteria, as it would be large, complicated and expensive. If we were making these by the thousands, we could decide to have a custom-designed IC built. Usually commercial designs for devices like this take that route, custom chips called Application Specific Integrated Circuits (ASICs) are designed that put almost all of the necessary circuitry on a single chip. But our budget is not quite that large here so we will have to come up with another idea.

Programmable Circuitry

It may have crossed your mind that a computer would have no trouble with this task. It would have no trouble keeping track of the time, displaying the temperature and other information on its screen, and all of the other functions. While we would not want to hook up an IBM XT to the furnace, perhaps we could build
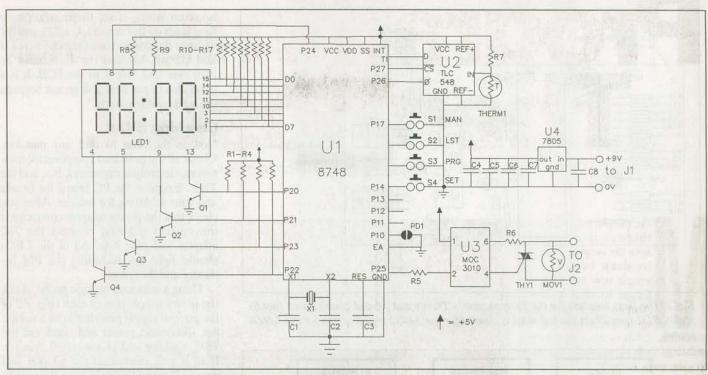
Fig. 1. The schematic for the Programmable Thermostat.

a simple computer into the thermostat. It would require a microprocessor IC, one chip for data storage, another for program storage, a couple ICs to interface to the digital display, another to interface to the temperature sensor, and maybe one or two more that would be necessary to keep it all working together. While this would do the job, it would not fit in a small case very well, it is rather complex, and it is still a bit pricey.

Fortunately, there is another alternative, a chip that combines almost all of that circuitry in one. Single-chip microcontrollers are economical, easy to use, and the one we will use in this project includes program memory, data memory, input and output capabilities, and a timer, all on a single 40 pin IC. It is programmed in its own assembly language, not too different from the lowest level programming language for any computer. (Some microcontrollers even have a built-in BASIC interpreter.) Intel Corporation's 8748 contains a built in EPROM, so that it may be programmed with a special PROM programmer (these are widely available) and then erased by exposing the clear window on the top of the IC to ultraviolet light, in order to revise the program.

The Circuit
Once you have the programmed microcontroller, all that remains is to add the few associated parts and put them

together on a printed circuit board. The thermostat program was written with the PC board design in mind, to make it as simple as possible. Although more expensive microcontrollers are available which include an analog to digital converter (ADC) on chip, this design uses an separate ADC chip. The ADC takes the voltage at the thermistor which is proportional to the temperature, and converts this to a binary number. This number is then sent serially into the microcontroller to reduce the number of necessary wires (see Radio Shack's data sheet for how this is done).

Also, to conserve wires and reduce current requirements, the LED display is multiplexed. This means that only one digit is on at a time, but the digits are scanned so fast that it looks like they are all on. U3 in the schematic diagram is an opto-isolator, used to isolate the current switched by THY1 from the low voltage used by the electronics.

The Components
Because most readers will not have access to an 8748 programmer, an 8748 is available from the supplier in the parts list with the thermostat program already programmed in. This program is about 600 instructions long, so a discussion of it is beyond the scope of this article. Use a triac such as Radio Shack's 276-1000, unless you are using the project in a high

power application (this is discussed further when we talk about installation). The value for MOV1 (metal-oxide varistor) also depends on the application, it used to protect the triac from voltage spikes. Try to find one rated for the voltage the triac has to switch (24V for a standard furnace). If one is not available use a varistor rated for a higher voltage. X1 may be either a 1.8432MHz or 3.6864MHz crystal. The AC adapter need not be the one specified in the parts list; however, it must provide at least six volts and 300 milliamperes. The thermistor used must be exactly the same as the Radio Shack device specified. This lack of flexibility is the reason that no calibration is required.

Building It
A PC board pattern (single-sided) is included here; you can make your own or order from the supplier in the parts list. Sockets must be used with U1 and U2 as they are especially sensitive to heat and ESD (static electricity). If possible, use a grounded IC insertion tool when putting these two chips into their sockets. The first step is to short the copper pads labelled PD1 with a solder bridge if you are using a 1.8432MHz crystal for X1.

Then, follow this sequence in building the project: Wire links (there are five), IC sockets, switches, resistors, capacitors, transistors, crystal, display module (connect to PCB with 15 pieces of short, solid
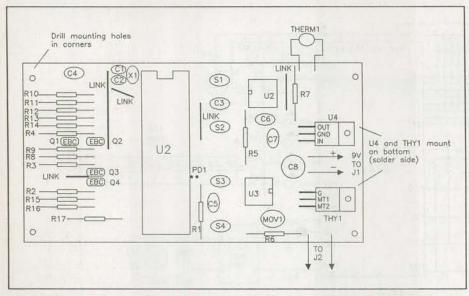
Programmable Thermostat

Fig. 2. The parts location for the Programmable Thermostat printed circuit board (not to scale). PD1 (pin 27 on the foil side) is shorted for the 1.8432 crystal., open for the 3.6864 crystal.
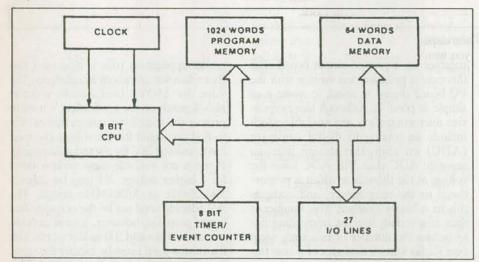

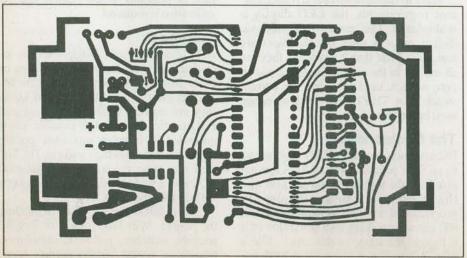
Fig. 3. The organization of the processor buss.

Fig. 4. The printed foil side.

hook-up wire), triac, thermistor (leave long leads on this device), U4, U3, and the wires to the connectors. Do not insert U1 and U2 yet. U4 and the Triac must be bent over and bolted to the PCB. It is a good idea to put heat sink grease between the metal surfaces here.

Checking It Out

Now is the time to find any mistakes. Check all the polarized components; transistors, electrolytic capacitors, ICs, and the Triac. Examine the PC board for breaks and your soldering for bridges. After you check that the power adapter connector is the correct polarity, connect the AC adapter and plug it in. All of the LEDs should light up, including the PM indicator and the colon.

Using a voltmeter or logic probe, check the power supply pins of each chip. All of the positive supply pins should be 5 volts; if not, disconnect power and check out the 7805 regulator and its associated circuitry. If all is well, disconnect the adapter and insert U1 and U2. Make sure they are in the right way. Reconnect power, and 12:00 should flash on the display. At this point you can try programming it and setting the time, an ohmmeter connected across the output will indicate if the furnace would be on or off.

Putting It Together

The PC board should be mounted to the front of the cabinet using bolts with spacers. Because the components which require holes in the case are mounted on the PC board, the hole locations can be worked out from the PCB layout. Drill four holes for the bolts and four holes for the pushbuttons in the cabinet face. The pushbuttons with their mounting nuts will also help hold the PCB in place. The length of the spacers and bolts depends on the height of the pushbuttons. Drill a small hole for the thermistor in the side of the case (the thermistor must be largely outside of the case in order to be accurately at room temperature).

A rectangular opening for the display must be cut and filed smooth. For improved visibility, a red filter should be epoxied in place behind this opening. If you made the wires to the display the correct length, the display will be touching the filter when the PCB is bolted in. It may be epoxied in place to the filter. The connector for the thermostat output should be mounted to the back of the case if you want to make this connection in the wall. Mount the AC adapter connector on the

back or side of the case depending on how you plan to install the project. The Programmable Thermostat is now ready for final testing.

Wiring It In

So far it has been assumed that the Programmable Thermostat will be used to replace a low-voltage furnace thermostat. The thermostat in most central heating systems switches a 24 volt circuit. If you wish to use this project with a baseboard heater or any system where the thermostat must directly switch the power to the heater, you will have to use a larger Triac and a large heatsink. If you do this, make sure that the Triac is rated for a load a little larger than you expect the heater to be, and that U3 and R8 are appropriate for driving the Triac's gate. You must also use a larger varistor for MOV1. Alternatively, you could use the Triac to drive a large relay.

Follow all safety precautions if you install the project yourself. Determine which wires were switched by the old thermostat, either by looking at a wiring diagram or examining the circuit. If you are unsure about how the wiring works, consult the Electrical Code or a book on do-it-yourself heating installation. The two wires which will turn on the furnace when they are shorted must be connected to the two screw terminals on the back of the Programmable Thermostat. After drilling appropriate holes in the case, it can be screwed to the electrical box of the old thermostat. The AC adapter should be plugged into a nearby outlet. The wire from it can be run neatly up the wall or you may wish to have it enter the wall near the outlet and connect to the thermostat inside the electrical box.

Going Further

Microcontrollers are a great way to simplify complicated projects. They are easy to design with once you learn to program in assembly language. You can get someone else to burn in the program or buy a EPROM programmer yourself. The author started programming microcontrollers using a very simple (two ICs) homemade EPROM programmer which plugged into a Commodore VIC-20 computer. If you like programming and experimenting with electronics, microcontrollers are the perfect way to combine the two. If you would like more information on microcontrollers, you can contact the company mentioned in the parts list.

Using the Thermostat

If you have just plugged in the thermostat, "12:00" will be flashing on the display. Otherwise, it will be in its normal display mode, alternating between the clock and thermometer displays. Using the thermostat is similar to setting a digital watch.

1) To set the clock:

—Press SET/ADV. The hour digits will flash.

—Press SET/ADV to advance the hour display. The dot in the top left corner of the display indicates PM.

—Press any other button when the hour is correct. The minute digits will now flash.

—Use SET/ADV again to set the minutes, then any other key to return to the normal display.

2) To program the thermostat:

—Press PRG/CLR to tell the thermostat you wish to add a program. If there are already eight settings entered, your request will be ignored.

—Advance the hours and minutes the same way you did to set the clock.

—A two digit number will now flash on the display. This is the temperature setting you would like to come into effect at the time you just set. Use the SET/ADV button to change this setting.

—Press any button other than SET/ADV to return to the normal display mode.

—Every day when the clock display reaches the time you just set, the thermostat setting will automatically change to the programmed one.

3) To list the programmed settings:

—Press LST. If there are no programmed settings, your request will be ignored. Otherwise, a setting will be displayed on the display, alternating between time and temperature.

—Press LST again to go on to the next setting. The settings will not be displayed in any particular order. If there are no more settings to be listed, it will return to the normal display.

4) To erase (clear) a setting:

—While the setting you wish to erase is being displayed (see LST instructions above) press the SET/CLR button. That setting will then be lost from the memory.

5) To manually change the thermostat setting:

—Press MAN. It is now just like a standard thermostat. The "S" in the leftmost digit indicates that the display is the current thermostat setting. Pressing SET/ADV will change this setting.

—Press MAN again to return to the normal display mode. ∎