

CIRCUIT CELLAR®

THE MAGAZINE FOR COMPUTER APPLICATIONS

FEATURE ARTICLE

Noel Rios

Multi-Channel Programmable Timer

How often have you wished you could preset your TV, radio, and lights to go on and off at particular times? Well, Noel has a multi-channel programmable timer to suit your needs. You could even use it to periodically recharge batteries.



In many situations, you need a gadget to turn on and off at a prescribed time.

One example is the radio. In the morning you need to hear the most important news and traffic tips before you go to work. Another is the TV, as much as important as the radio in bringing the news. The list goes on and on.

It would be especially convenient if you could turn most of these gadgets on and off automatically! And, that's not all! What about having a maximum of four devices with four different on and off times. With the multi-channel programmable timer, you can connect a maximum of four fluorescent or incandescent lamps to it, as long as it does not exceed the solid-state relay current rating. For example, you can connect a lampshade to the gadget, along with a TV and a stereo.

Equally important, you can use the programmable timer to automatically set the charging time of rechargeable

devices like lamps, drills, screwdrivers, batteries (particularly NiCd batteries or lead-acid batteries), and so on. Simple chargers do not have timers or charge-termination circuits in them to shut off the charger after some time, to prevent them from heating up and overcharging.

Another use for the programmable timer is to sequence a number of devices to automatically turn on or off at the same time.

CIRCUIT DESCRIPTION

The heart of the circuit is an 80C31 (U1) microcontroller (see Figure 1). Couple this with a Dallas DS1287 real-time clock with nonvolatile RAM and some logic parts, and you have a hefty programmable timer.

The 80C31 microcontroller manipulates the real-time clock, reads the switches, compares the settings, and gives outputs according to the settings. It is a versatile part that has been popular for many years because of its price and availability. It also has an efficient instruction set suited for embedded control. It has 64 KB of program memory and 64 KB of external RAM. In this project, a 27C64 is used, which takes up 8 KB of the program area.

The DS1287 and 74HC244 occupy the external RAM, known as the MOVX space. All of the parts use CMOS so the current consumption is minimal, enabling us to use a smaller, more inexpensive power transformer.

Using external memory, two of the ports (P0 and P2) are used as address and data buses, which are also multiplexed. This leaves us with 14 I/O ports that are used for the LCD, which is controlled by Port P3. This is easier than using an 8-bit port, because bit manipulation is possible. And, Port P1 is used for output to the eight channels.

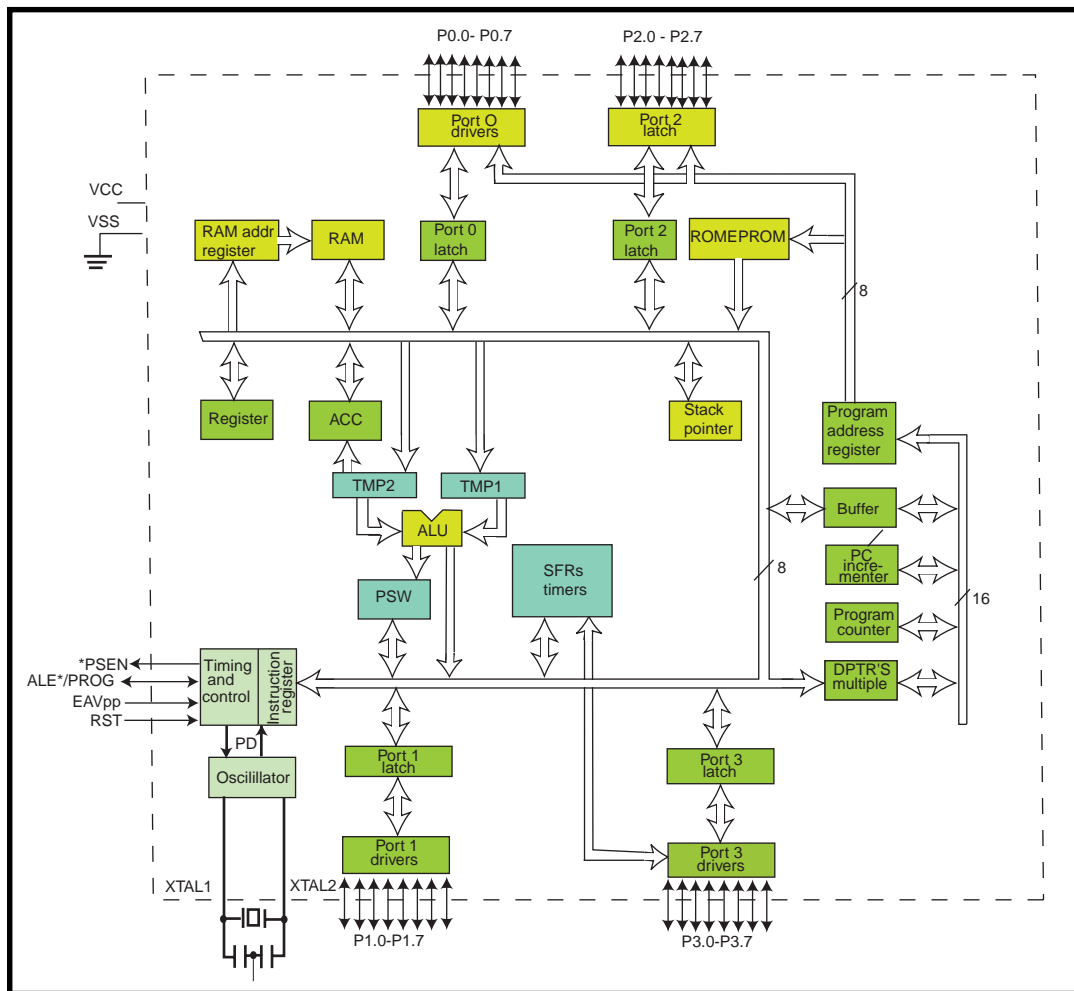


Figure 1—The 80C31 is an 8-bit CMOS computer having 128 KB RAM, four ports, two timers, a serial port, and two external interrupt sources. It can access separate 64 KB of program and data memory.

The DS1287 (U4) is a versatile device (see Figure 2). It has the crystal, logic and protection circuits, and energy source all in a single package. This eliminates several additional parts that would normally be needed for this project. The DS1287 is a real-time clock with nonvolatile RAM, which is used in our project to store the time to turn on and turn off. Because it is nonvolatile, it is not necessary to reset the time in case of a power outage. The same principle goes for the real-time clock (a hardware implementation, which frees the microcontroller to do other tasks). The clock continues to function even if you turn the gadget off.

The 1 × 16 LCD module in this project uses a HD44780 controller/driver. It contains a CPU, RAM, and ROM to control the LCD glass. A 4-bit operation is chosen, so only six wires will be used to talk to the LCD

module, and the R/W pin is grounded, so it will always be in write mode. In this configuration, the busy flag is not read, and a fixed delay is used for proper timing. In a 4-bit configuration, only the upper nibble is used, and the lower nibble is left unconnected.

A 74HC138 (U5), one of eight decoders, is used to map the DS1287 and 8-bit port to the MOVX space. (The DS1287 occupies memory locations 00H to 7FH.) The 74HC32 is a quad, or gate, used as a glue logic for the 8-bit port, which is composed of the 74HC244. (The address space allocated to the 8-bit input port for the keypad is 80H.) The 74HC240 (U9) provides the inverting function and gives enough driving current to drive the solid state relays. We need to invert the outputs of Port 1 after a reset.

The DS1232 (U8)—a supervisory IC that monitors the power fed to the

circuit—generates a reset if the V_{CC} goes out of tolerance (4.5 V or below) or if the microcontroller freezes (see Figure 3). This watchdog function works because pin 7 (/ST) of the DS1232 is connected to ALE (address latch enable) line, which is constantly active when it is fetching programs and data. This pin must be toggled every now and then—if not, it will generate a reset.

Solid state relays (SSR) are used for controlling the outputs, because they present a simple interface to the CMOS circuits. No level translation is necessary. Also, the input is optically isolated from the output. In short, the main board is not fried if something goes wrong in the SSR.

(Fuses are used to protect the SSRs, in case of a short circuit or overload.)

The power source consists of a 6-V 200-mA transformer, rectifier diodes D2 to D5, and capacitors C5 and C6. The regulator used is a LM2931 low dropout regulator (LDO)—a wise choice, because it consumes a mere 25 mA. So, we can use a 6-V transformer and a higher operating efficiency for the power section.

One thing you might ask is why I did not connect the keypad to the same I/O lines used by the LCD. It is possible, but because the LCD is in the clock mode, it is being updated constantly. Therefore, there is a greater chance that the I/O line will be shorted to ground if a key is pressed.

CONSTRUCTION AND ASSEMBLY

I used point to point wiring to connect the parts, although wire

wrapping is also an alternative to construct the circuit solder and connect the IC sockets to the prototyping board. The layout for the IC sockets is grouping the parts together, so the wires will not become tangled.

Always note the position of the power pins—pin 40 (V_{CC}) and pin 20 (Gnd) in 40-pin DIP, and pin 16 (V_{CC}) and pin 8 (Gnd) in 16-pin DIP. These reference pins are useful in numbering the power pins so that you will not be confused.

Start with the microcontroller 80C31 (U1) from left to right. Wire the microcontroller to the octal transparent latch 74HC373 (U2)—they should be close together. Use different colors each for the data bus, address bus, and the control bus (/RD, /WR, PSEN). Next, wire the '27C64 EPROM (U3), the DS1287 real time clock (U4), the 74HC138 (U5), and the two 74HC244s (U8 and U6), consecutively. After connecting the ICs together, assemble the power-supply section. (Note the orientation of the LM2931 LDO regulator.) Lastly, wire the LCD, taking note that pin 1 of the LCD is ground, and pin 2 is V_{CC} .

CHECKING AND TROUBLESHOOTING

In checking digital and microcontroller circuits, the following are important when using a voltmeter and logic probe.

- POWER is present
- CLOCK is present
- activity in the data bus
- Activity in the address bus (check transparent latch U2)
- program data is read /OE and /CS of

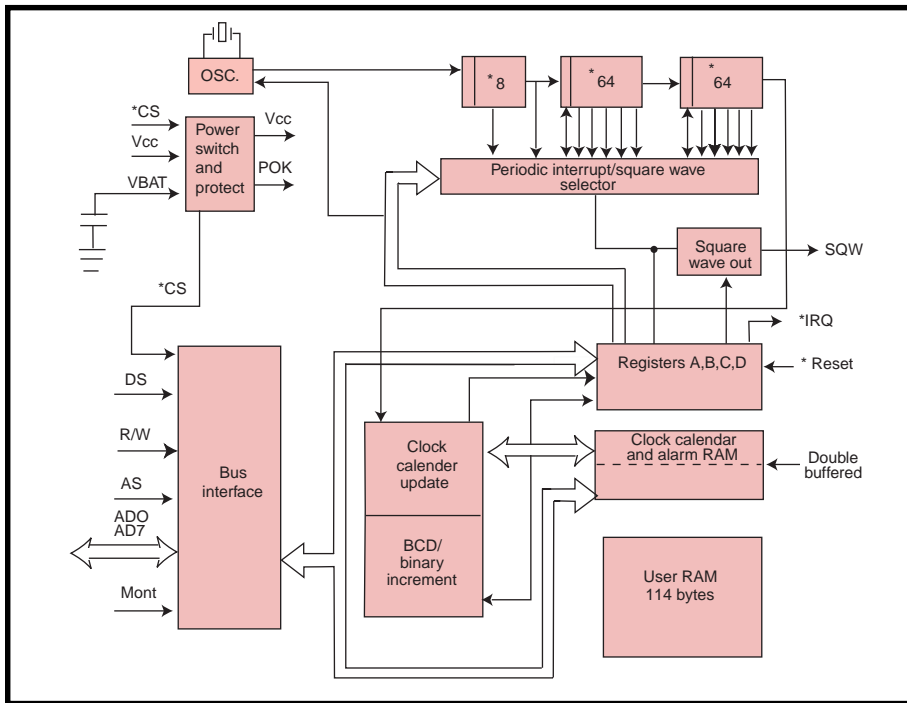


Figure 2—The DS1287 real-time clock contains all the necessary logic circuitry to be connected directly to the microprocessor/microcontroller bus without any glue logic. The battery and power switch/write-protect circuitry make the internal RAM nonvolatile.

- EPROM pins are active
- data is read and written /RD and /WR are active

Do not wire the solid-state relay until you check the connections between the pins of all the ICs, using a multimeter or continuity tester. So it will be easy to troubleshoot later, don't mount the circuit board to a suitable enclosure, until you first check the circuit. Do not plug in the ICs, until after you check the power.

After wiring the power supply section, check the DC supply at the supply pins of the ICs. Pins 40, 28, 24, 16, and 14 should read 5 V. Pins 20, 14, 12, 8, and 7 should be at ground potential. It's not enough for them to

be at zero potential, they must also be connected to ground.

After Power is OK. Turn off the device and plug in the 80C31 microcontroller. Pin 18 should be active or pulsing. This means clock is present. Turn off the project and plug in the EPROM. After powering up, you'll see activity or pulsing in the data bus and address bus. It is fetching program memory by checking the /OE and /CS of the EPROM.

Also, check the LCD display, it should read "PC Timer V1.0". If all seems fine, plug in the remaining ICs one at a time after powering off and see which part, when inserted, causes it to malfunction. If the project ceases to display the message in the LCD,

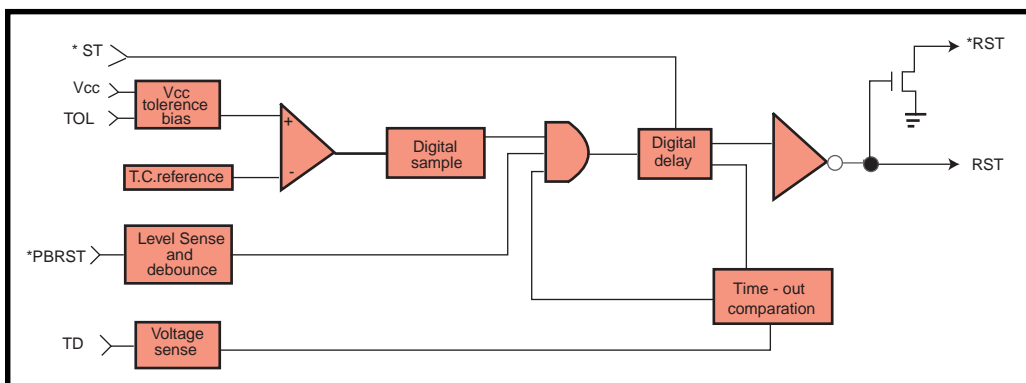


Figure 3—The DS1232 micromonitor features a digital delay for the necessary time delay so a capacitor and resistor is not needed for a power-on reset. It also monitors the supply to make sure it is within tolerance and also checks if the processor is running the program properly.

the most probable cause is a wrong connection in that IC.

In accessing the DS1287 and 8-bit input port for the keypad, pins 15 and 14 of the 74HC138 (U5) should be pulsing. If they're not, check if pins 1, 2, and 3 are pulsing. Also, to decode the memory locations properly, the address lines connected to the 74HC138 (U5) should be correct.

If the keypad cannot be accessed, try checking if the /RD (pin 10 of U1) is connected to pin 2 of 74HC32 (U7). Also, check if pins 1 and 19 of U6 are pulsing.

If the circuit board is functioning properly, place it in a suitable enclosure. The enclosure should provide holes for the switches, LCD, fuse holder, convenience outlet, and the power plug. And finally, wire the solid state relay, fuse, and convenient outlet. Check the connections and check for shorts or faults.

SETTING UP THE DEVICE

After powering up, the LCD should display "Timer V1.0" and will start displaying the time. Usually, a new DS1287 has its oscillator turned off, so you will see that the time is stopped. To set the clock, press the Adjust key, and the hour field will start blinking. To move to the minute setting, press the Select button, and the minute field will blink. While in each field, press the Up button to increment the value, and press the Down button to decrement the value. Pressing Adjust again will bring you to the main screen, or the clock with an asterisk after the time.

To scroll through the channels, press the Select button while in the main screen or Clock mode. It will then display the on and off time (in hour and minutes). To adjust the on and off times, press the Adjust button, and the hour of the on time will blink. To move to the minute setting, press the Select button. To move to the hour setting of the off time, press the Select button again. Pressing Adjust once more will bring you to the Scroll Channels mode.

You will then be able to scroll through channels 0 to 7 by pressing

the Up or Down buttons to move back and forth between them. Pressing the Select button once more will bring you back to the Clock mode. It is important to remember that you must be in the Clock mode, and not in the Adjust mode or scrolling through the channels, for the timer to control the devices plugged into it.

Note that pressing Reset will turn on or off all devices in a short time. So, the Reset button is a good way to shut off all the channels at once. Also, in a power failure, on return of power, the devices on all channels must be set on again for safety reasons.

TIME TO GO

There are many uses for the multi-channel programmable timer, and it is only limited to the imagination of the user. The circuit currently designed is for four outlets, but you can expand it to eight if you have eight solid-state relays. Although some provide a supervisory circuit and some safety features are included in the software, this device is not recommended in cases where a catastrophic event will occur if the circuit board ceases to function. ☠

Noel a. Rios is an electronics and communications engineer and has worked for semiconductor electronics companies like Allegro, IMI, and ASTEC, to name a few. His interests include power conversion, embedded control, GPIB control, as well as test and measurement. You can reach him at narios@philonline.com.

SOFTWARE

The .asm, .bin, .hex and schematic files are available for download on the *Circuit Cellar* Web site.

SOURCE

DS1287
Dallas Semiconductor
(972) 371-4448
Fax: (972) 371-3715
www.dalsemi.com

Circuit Cellar, the Magazine for Computer Applications. Reprinted by permission. For subscription information, call (860) 875-2199, subscribe@circuitcellar.com or www.circuitcellar.com/subscribe.htm.