Q & A

READERS' QUESTIONS, EDITORS' ANSWERS

CONDUCTED BY MICHAEL A. COVINGTON, N4TMI

Gently Falling Frequency

Q *I need a design for an oscillator circuit, preferably 555-based, that will start out at approximately 500 Hz and decay to 0 Hz over a period of eight to ten seconds while maintaining an amplitude high enough to drive other digital ICs. Any help you could give would be greatly appreciated.—J. A. S., Smyrna, GA*

A This is not easy to do with a 555 (I won't say impossible, because next month I'll probably hear from someone who has done it!). Figure 1 shows how to do it with an LM331N voltage-controlled oscillator. The frequency of oscillation is set by R3, C4, and the control voltage at pin 7. When you apply power, C1 and C2 charge, with most of

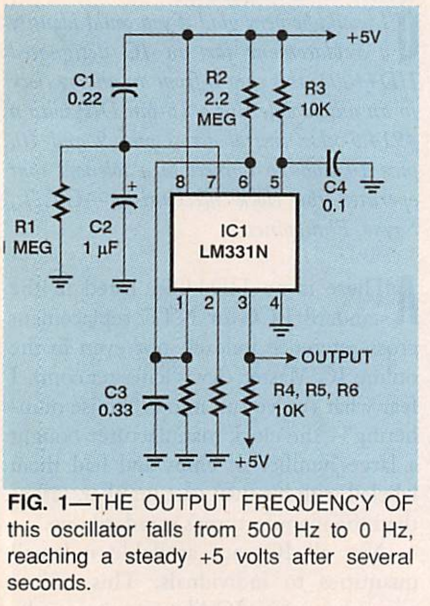

FIG. 1—THE OUTPUT FREQUENCY OF this oscillator falls from 500 Hz to 0 Hz, reaching a steady +5 volts after several seconds.

the voltage ending up on C1; pin 7 reaches about 1 volt and the oscillator runs at 500 Hz. Immediately, C2 starts discharging through R1, and the frequency drops over a period of several seconds, until oscillation stops completely with the output high (+5V). Connected to a speaker, this gives a nice "Eeeeeeeeuuuuwwww...pop...pop...pop" sound effect, and it's fully compatible with TTL and CMOS logic circuits. If you're not driving logic chips, the supply voltage doesn't have to be exactly 5 volts.

The purpose of R2, which you won't find in most LM331N circuits, is to bias the voltage-controlled oscillator slightly so that the oscillation will definitely stop when the input voltage falls to zero, rather than continuing at a very low frequency. Resistor R6 is the pull-up resistor for the open-collector outputs. For the functions of the other components, see the LM331N data sheet (available online at www.nsc.com).

Another way to get a falling frequency is to program a microcontroller to toggle an output bit at a steadily decreasing rate. Figure 2 shows *pseudocode* (an English-like outline of a computer program) indicating how this might be done. We'd appreciate feedback from readers as to whether pseudocode is a good way of documenting microcontroller software; the actual assembly code would be much longer and harder to read and would apply to only one type of CPU.

The key idea is to delay N milliseconds between output transitions, where N is a steadily increasing number, and then stop altogether when N gets high enough. As shown, the program takes 10 seconds to stop, but the frequency decrease is nonlinear; the frequency drops very rapidly at first, then slowly trails off. Instead of just adding 2 to N each time, you might want to compute something like:

If N>8 then N:=N/8 else N:=N+1

so that the increase in N is proportional to N itself. Note that N/8 is easy to compute in binary; all you do is shift the number 3 bits to the right. However, if you start with a number less than 8, you get zero, and in that situation, it's necessary to add 1 rather than 0, or N will never change and you'll be stuck. Or you can compute a more complicated function and make the frequency fall exactly the way you want. You could even use a memory lookup table for a series of steadily increasing time delays.

```
   N := 1;
A: Delay N milliseconds;
   Toggle output bit;
   N := N + 2;
   If N < 200 then go to A;
   Stop.
```

FIG. 2—HERE'S THE PSUEDOCODE for a microprocessor program to generate a signal whose frequency gradually falls to zero.

Since it only needs one output pin, this application might be a good job for one of the new low-cost 8-pin PIC microcontrollers (PIC12C508 and the like). Microchip Technology has just announced an 8-pin PIC with an on-board voltage regulator so that it doesn't need a 5-volt supply. Information is available online from www.microchip.com or by writing to Microchip Technology, Inc., 2355 W. Chandler Blvd., Chandler, AZ 85224.

Mystery Outlet

Q *I have a digital electric alarm clock that began gaining time. I wrote it off to a lightning hit we took a few weeks ago and bought a new one. I plugged it in and it, too, ran about 25% fast. I then plugged both clocks into other outlets in the same room and they both are working correctly. I am completely baffled. Do you have any ideas?—R. E. S., Watkinsville, GA*

A Two words: *electrical noise.* As shown in Fig. 3, a line-powered digital clock keeps time by counting cycles of the 60-Hz AC power. (This is normally more