design ideas

Edited by Bill Travis and Anne Watson Swager

Preheat starter for electronic ballast

Arthur E Edang, Don Bosco Technical College, Mandaluyong City, Phillipines

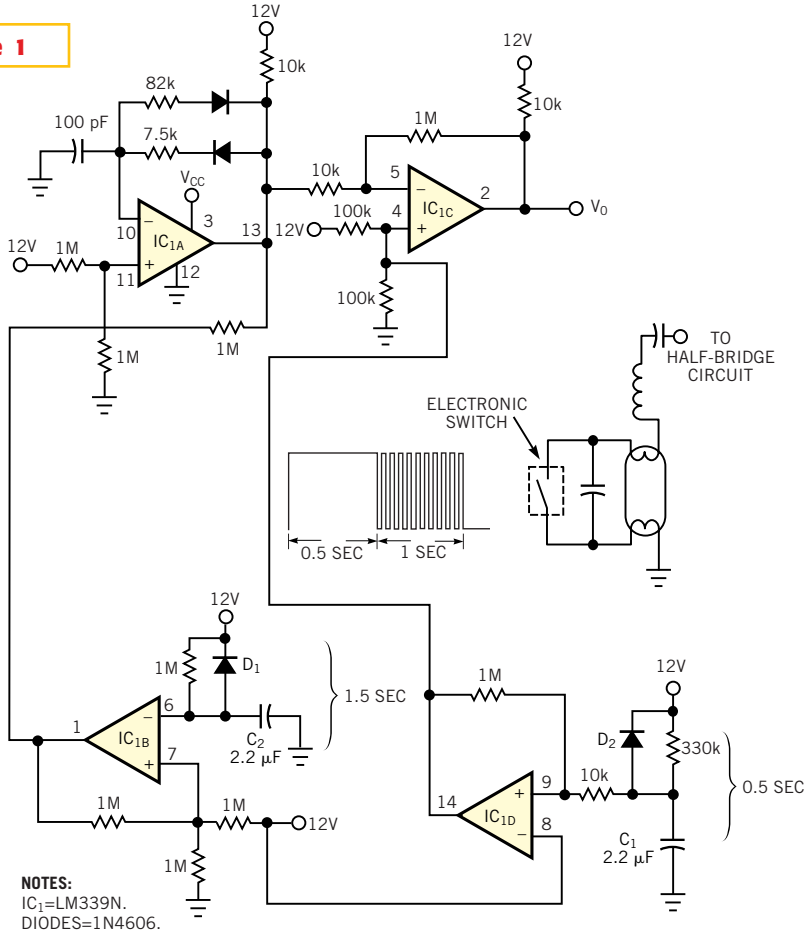
Electronic ballasts for fluorescent lamps use various techniques to turn on the bulbs. The design usually involves a compromise between turn-on voltage and lamp life because the two are inversely related. One way to reach a reasonable compromise is to initially allow a momentary inrush current to warm the filaments, followed by a series of interrupted short circuits across the lamp that generate the required high voltage to trigger the fluorescent. With a preheated filament, the necessary strike potential reduces to half.

The trigger circuit in **Figure 1** controls the electronic switch across the bulb. At start-up, IC_{1D}'s output is low as C_1 and C_2 charge toward V_{CC}. IC_{1D}'s low output pulls IC_{1C}'s inverting input low, which causes V_O to clamp high. A high level at V_O closes the switch and forces current through the filaments. After approximately 0.5 sec, IC_{1D}'s output changes state and allows IC_{1C} to accept the high-frequency signal at its noninverting input. IC_{1A} is a square-wave oscillator, which causes V_O to be a high-frequency-pulse series that lasts for approximately 1 sec. At the end, C_2 reaches a high enough voltage to force IC_{1B} to pull down IC_{1A}'s noninverting pin to ground. With a grounded IC_{1A} output, V_O clamps low.

The high-frequency switching strikes the preheated lamp. In case the bulb fails to start, the circuit turns off and then on again. Residual charges on the capacitors discharge through D_1 and D_2 to ensure precise timing.

Is this the best Design Idea in this issue? Vote at www.ednmag.com/ednmag/vote.asp.

Figure 1



This trigger circuit generates a high-frequency-pulse series to strike a preheated lamp.

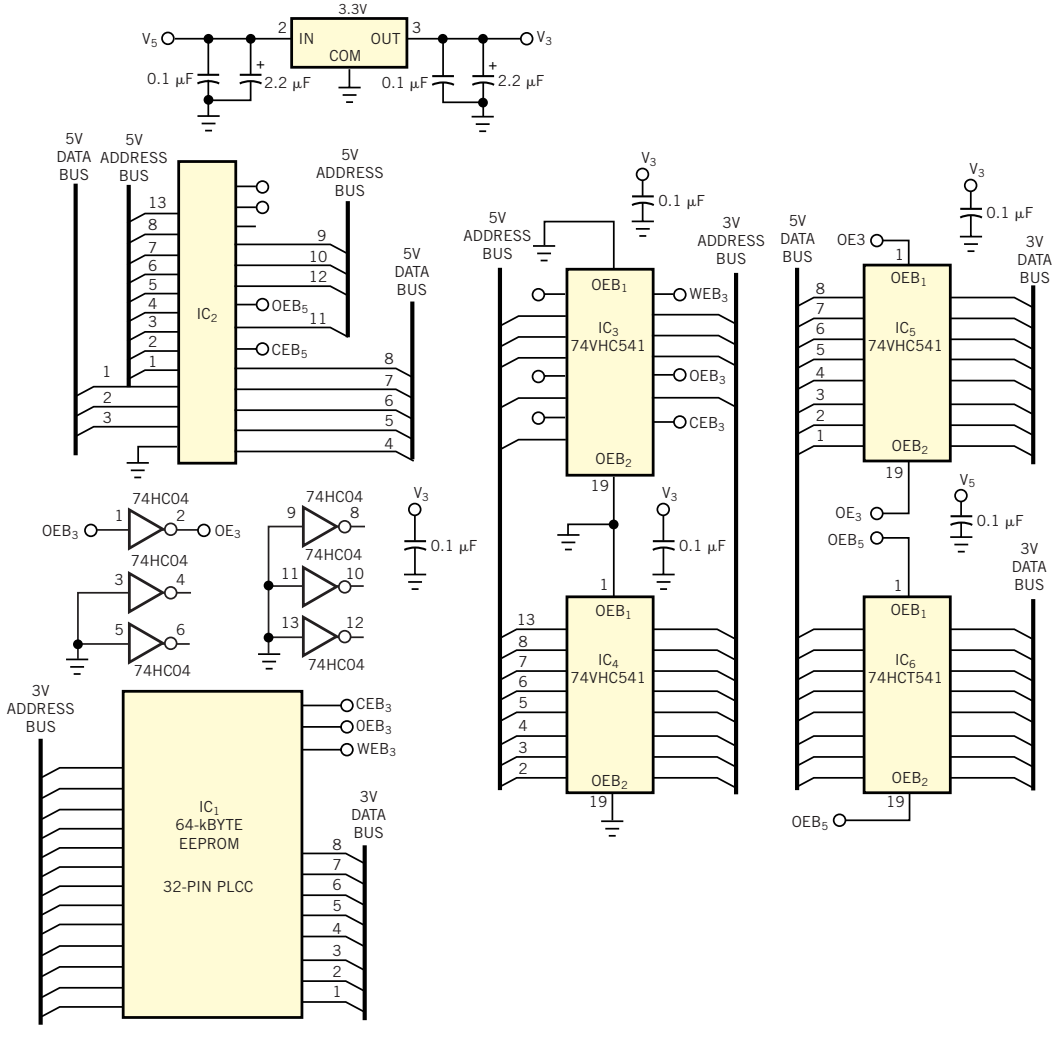
Low-cost circuit programs EEPROMs

Jarrod Eliason, Ramtron, Colorado Springs, CO

When you migrate to 3.3V system supplies, you must usually replace your old, reliable EEPROM programmer with a new, overly flexible and expensive universal programmer. We could not find a 3.3V programmer for less than $1000. For less than $100, the circuit in **Figure 1** extends the functional life of any 5V EEPROM programmer. You can apply the circuit to any bidirectional 5 to 3.3V level-translating application. The key to the circuit lies in choosing the correct logic families. The 74VHC and 74LVC families handle the 5-to-3.3V conversion better than previous logic families, such as the 74HC series. The 74HC family accommodates 3.3V operation, but the input-protection diodes clamp the input voltage within a diode drop of V_{DD} (**Figure 2a**). So, applying 5V to the input of a 74HC part powered from 3.3V results in much undesired current. An external resistor could limit this current, but this fix would impact bus speed. The 74VHCT and 74LVC families do not use a reverse-biased diode to V_{DD} (**Figure 2b**), so the input voltage can safely rise to 5.5V, regardless of the supply level.

The 74HCT family handles the 3.3-to-5V conversion. This 5V CMOS logic family uses input switching levels skewed to accommodate TTL-level inputs. The low and high levels are 0.8 and 2.4V, respectively, in comparison with the typical CMOS levels of 1.5 and 3.5V. Because the inputs receive high levels of 3.3V at most, CMOS-optimized 74HC logic would not guarantee recognition of logic 1 inputs. On the other hand, to a 74HCT powered from 5V, a 3.3V input level represents a solid logic 1. We selected the tristatable

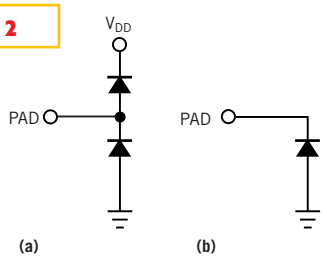
Figure 1



For less than $100, this circuit adapts a 5V EEPROM programmer for 3.3V operation.

buffer function for the EEPROM-programmer level translation. The circuit in **Figure 1** programs a 3.3V, 64-kbit EEPROM, using a 5V programmer. For the address and control pins, the output-enable pin of the 74VHC chips is constantly active. For the bidirectional data bus, the OEB5 and OE3 signals control the in/out selection. When OEB5 is low and OE3 is high, a read operation takes place, and the EEPROM has control of the data bus. When OEB5 is high and OE3 is low, a write operation takes place, and the programmer drives the data bus.

Figure 2



The 3.3V-powered 74HC-logic inputs are not amenable to 5V inputs (a); 74HVC and 74LVC inputs have no such problem (b).

A 28-pin DIP socket, IC_2, connects to the 5V EEPROM programmer. The circuit uses an additional adapter to interface to the 32-pin PLCC target device, IC_1. The 74VHC and 74LVC logic parts are not readily available in DIP form, so you can use SOIC-to-DIP adapters for breadboarding. If the 74HCT541 is not available, you can use the alternate-pinout 241 or 244.

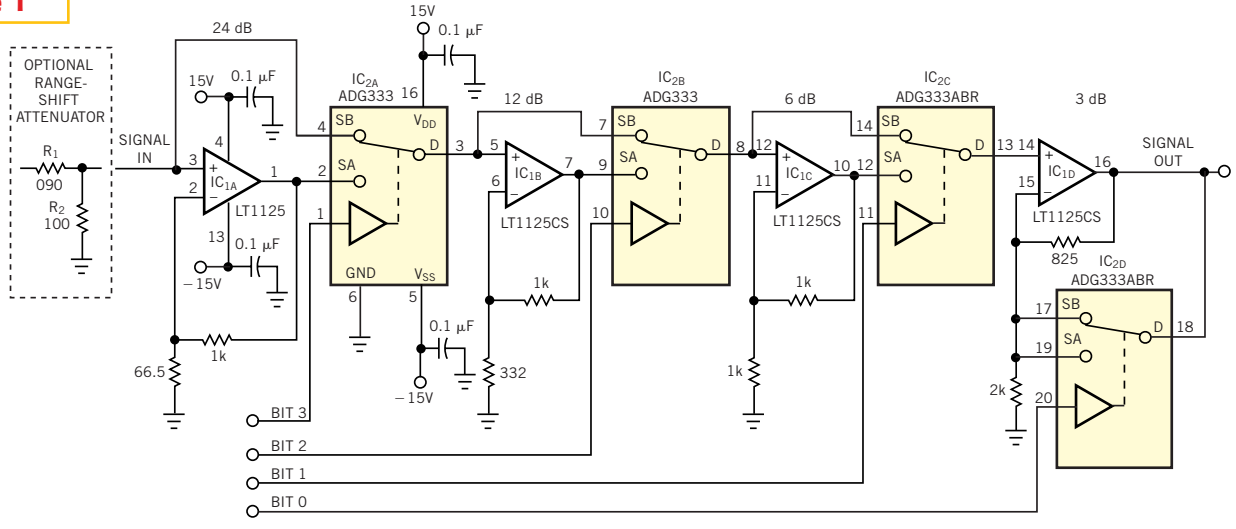
Is this the best Design Idea in this issue? Vote at www.ednmag.com/ednmag/vote.asp.

Circuit yields ultralow-noise VGA

Dale Ouimette, California Institute of Technology, Pasadena, CA

A NUMBER OF SINGLE-CHIP VGAs variable-gain amplifiers are available today. Unfortunately, they all have drawbacks, such as high noise, 55V limit, low input impedance, or nonlinear gain/frequency characteristics. The circuit in **Figure 1** is a 16-step, ultralow-noise VGA that solves many of these problems. IC_1 is a low-noise quad op amp, and IC_2 is a quad SPDT CMOS switch. The stages switch in successive multiplication (gain) factors using a TTL binary code. The values shown provide 0- to 45-dB gain in 3-dB steps. For best low-noise performance, the higher gain stages precede the lower gain stages. The circuit exhibits approximately 3 nV/$\sqrt{\text{Hz}}$, referred to the input, for most gain settings. The highest noise is 4.5 nV/$\sqrt{\text{Hz}}$ at a gain of 9 dB. Distributing the total gain across multiple stages increases the overall bandwidth. The output stage has a different configuration to yield a low-output-impedance output driver

Figure 1



This VGA offers ultralow noise, a wide dynamic range, and high bandwidth.

at all gain settings.

If you need to remotely control the gain, you must concern yourself with ground loops that can compromise the low-noise characteristics of the circuit. One solution is to place optoisolators in the four digital-control lines, so that no ground connection exists between the two ends of the cable except through the power supply. The method you use is an analog differential-control voltage using an ADC to generate the 4 bits. **Figure 2** shows a circuit that performs this function well. IC_1 is a differential receiver, and IC_2 is an 8-bit ADC. In some applications, you could get away with using only the ADC, because it already has a differential input. However, you must take care not to exceed the narrow common-mode range of the ADC's input. A more robust solution is to place a differential receiver in front of the ADC, as shown. R_1 and C_1 form a lowpass filter for the control voltage to the ADC. The 4 high-order bits from the ADC control the CMOS switches. As shown, the ADC operates in a self-clocking mode and needs no other controls.

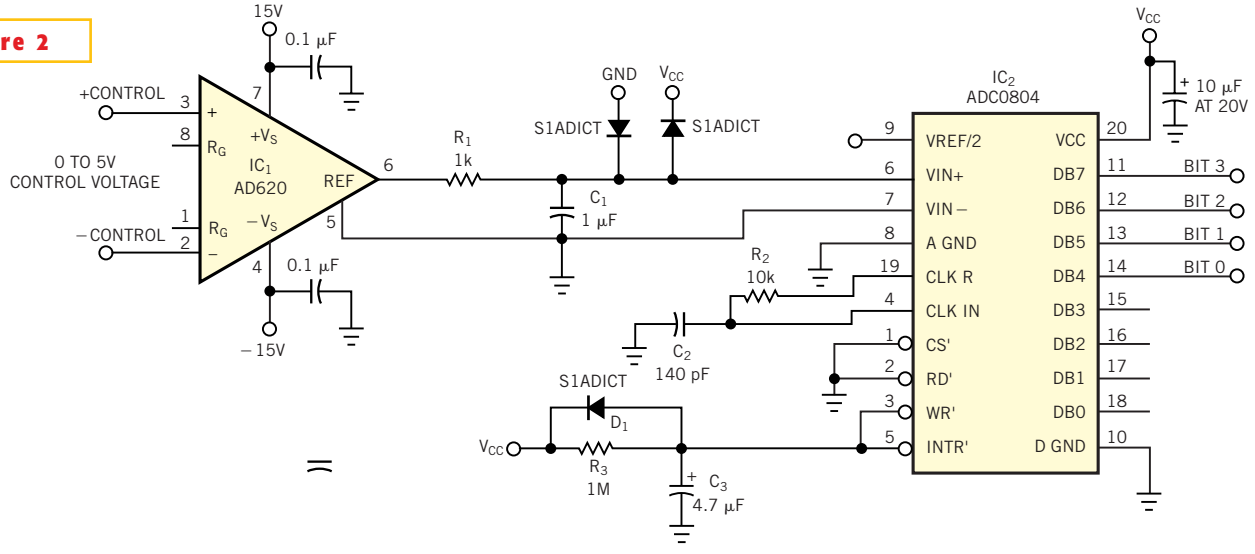
TABLE 1–PERFORMANCE VERSUS GAIN

Step	Gain (dB)	V/V	Noise (referred to input) (nV/√Hz)	3-dB bandwidth (MHz)
0	0	1	3.1	10.5
1	3	1.4	3.8	7.7
2	6	2	4.4	5.1
3	9	2.8	4.5	4.6
4	12	4	3.6	2.7
5	15	5.6	3.6	2.7
6	18	7.9	3.7	2.6
7	21	11.2	3.7	2.6
8	24	15.8	3	0.88
9	27	22.4	3	0.89
10	30	31.6	3	0.94
11	33	44.7	3	0.96
12	36	63.1	3	0.97
13	39	89.1	3	0.97
14	42	125.9	3	1.04
15	45	177.8	3	1.02

R_2 and C_2 control the sampling frequency, approximately 640 kHz for the values shown. D_1, R_3, and C_3 provide power-up initialization for the ADC's clocking function. The control-voltage steps are 310 mV apart, providing ample noise immunity. **Table 1** shows the performance of the overall circuit with analog control. You can use R_1 and R_2 in **Figure 1** to shift down the overall gain range with little sacrifice of noise characteristics. You can obviously alter the individual gain stages to yield other ranges and step sizes, such as 0 to 30 dB in 2-dB steps. At the expense of circuit simplicity, you could replace the quad op amp with four ultra-low-noise op amps, such as the LT1128 or AD797. This replacement lowers the noise to approximately 1.4 nV/√Hz. You could also increase the number of stages, thereby providing a wider dynamic range, finer gain steps, or both. The benefits of this circuit over commercially available single-chip VGAs include ultra-low noise, high bandwidth, ±13V range, high input impedance, ground-loop immunity, and user-defined dynamic range and step size.

Is this the best Design Idea in this issue? Vote at www.ednmag.com/ednmag/vote.asp.

Figure 2



An ADC controls the gain-setting codes for the circuit in Figure 1.

Sequential channel selector simplifies software

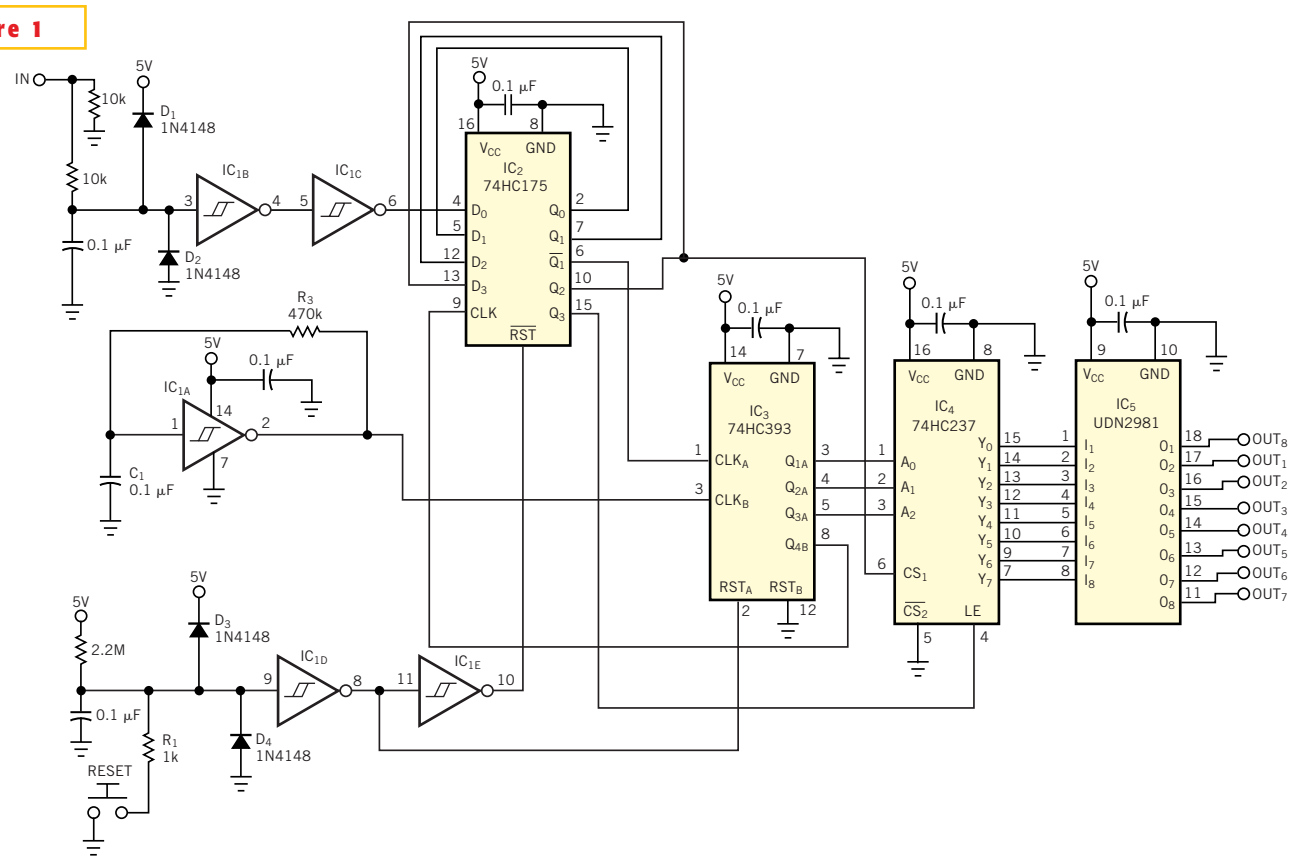
Alex Knight, Cummins Engine Co, Columbus, IN

An efficient but powerful circuit is useful for a variety of applications with limited I/O and for which you want to use one input to sequentially select a different output channel (**Figure 1**). When the software changes the state of only one input, the circuit sequentially selects one output channel at a time for test purposes. Because the test-application environment is potentially harsh, the circuit must have relatively high noise immunity and transient protection at the inputs. You must also be able to reset the circuit to resynchronize the hardware with a test program after any interruption in testing.

Although the resulting circuit may seem simple and standard, it is distinctly robust. The delayed reset signals at IC_2's Pin 1 and IC_3's Pin 2 return the counter and flip-flop ICs to their initial state so that OUT_1 is the first channel active at the first count. The power-on and switch-activated reset circuit includes R_1, D_3, and D_4 to protect against ESD that could arc over the switch contacts when someone first touches the switch. The IN signal input circuit has similar transient protection with R_2, D_1, and D_2. A simple RC oscillator generates the clock signal at IC_2's Pin 9, and the second four-stage binary-ripple counter, IC_3, divides this clock by 16. The oscillator frequency is approximately 21 Hz, but you can change R_3 and C_1 to produce the desired frequency, which is approximately $1/R_3C_1$. You can also use a potentiometer in place of R_3 to make the frequency adjustable. Keep in mind that the flip-flop clock-cycle period should be much less than the expected active and inactive periods of the IN signal but long enough to produce adequate debouncing of the input signal to maintain good noise immunity. The circuit serves a low-speed application, so the clock at IC_2's Pin 9 is 1.3 Hz.

The circuit filters and buffers the IN signal before sending it to the flip-flop

Figure 1



A robust circuit uses one input to sequentially select one output channel at a time.

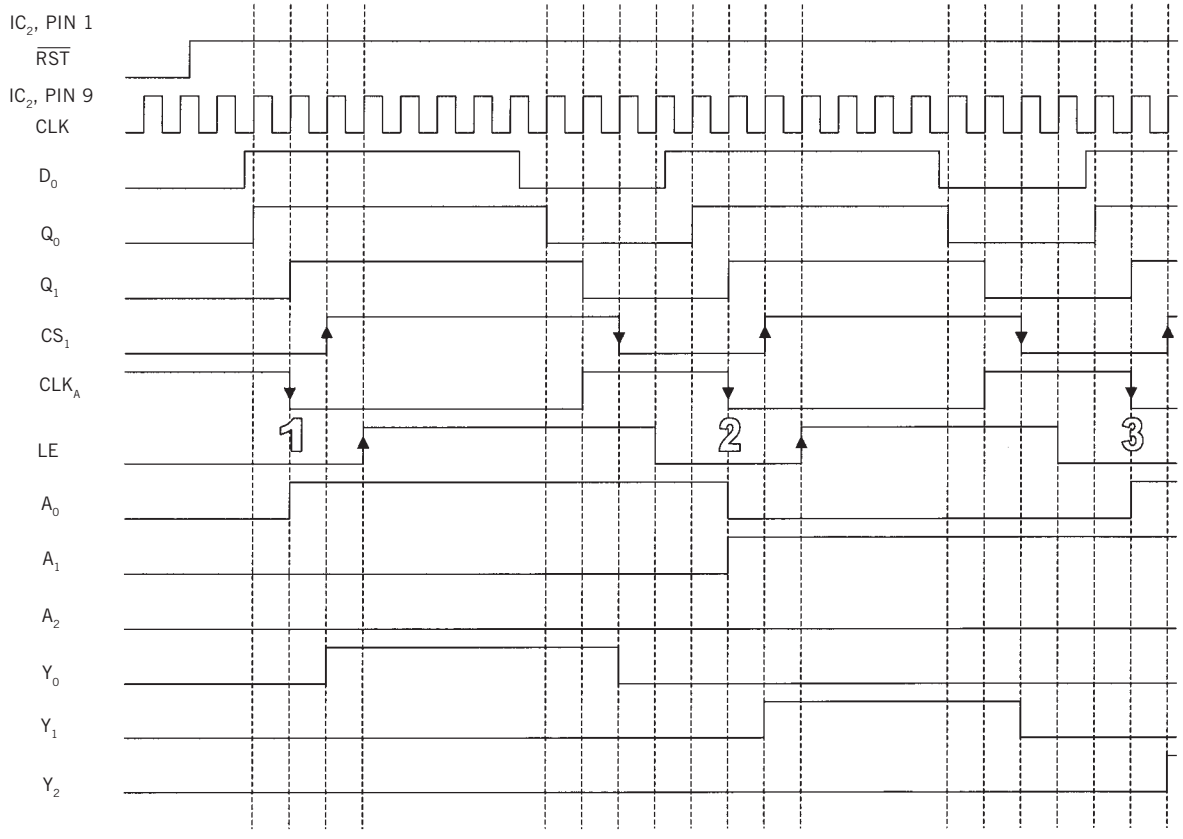
input at IC_2's pin 4. The Schmitt inverter, IC_1 with its built-in hysteresis and the cascaded flip-flop circuit provide high immunity to noise, and the cascaded flip-flop ignores any glitches on the input signal that occur asynchronously to the flip-flop clock signal's positive-going transitions. The circuit uses the $\overline{Q_1}$ output signal as the CLK_A clock input to the first four-stage binary ripple counter, IC_3. Negative-going transitions increment the counter as the timing diagram indicates at counts 1, 2, and 3 (**Figure 2**). The circuit uses the Q_2 output to select the active-high CS_1 chip-select input of IC_4's one-of-eight decoder, which allows plenty of time for the ripple counter outputs to stabilize, even at high flip-flop clock speeds. These outputs do not simultaneously change states. With CS_1 high, the positive-going Q_3 output signal at the LE input of the decoder (IC_4, pin 4) latches the output channel that the state of the A_0-to-A_2 address inputs select. Latching the output channel ensures you that any subsequent noise-induced counter-output state changes will not affect the output-channel states. While CS_1 is low, the Y_0-to-Y_7 outputs from IC_4 are also low. This design maintains a similar off-time for all of the output channels, as reflected in the input signal, although the circuit delays any change of state for each of the outputs by approximately two cycles of the flip-flop clock period.

IC_5 can drive loads that sink as much as 350 mA at room temperature, such as relays, solenoids, dc motors, and lamps. This eight-channel source-driver IC is unnecessary if CMOS outputs suffice as the channel-select signals. The IC_5 source voltage can climb to 35V if you add a separate supply. IC_5 has internal diodes on all of the outputs to clamp inductive spikes.

The circuit includes a switch for generating a reset signal, which you can use in addition to or instead of an external reset signal. The input can also be an external analog signal or non-TTL, as long as you properly compensate for any dc offset necessary to work at the switching thresholds of the Schmitt inverter. You can cascade additional ripple-counter stages and add decoders and output drivers to select from more output channels.

Is this the best Design Idea in this issue? Vote at www.ednmag.com/ednmag/vote.asp.

Figure 2



Negative-going transitions increment the counter at counts 1, 2, and 3.

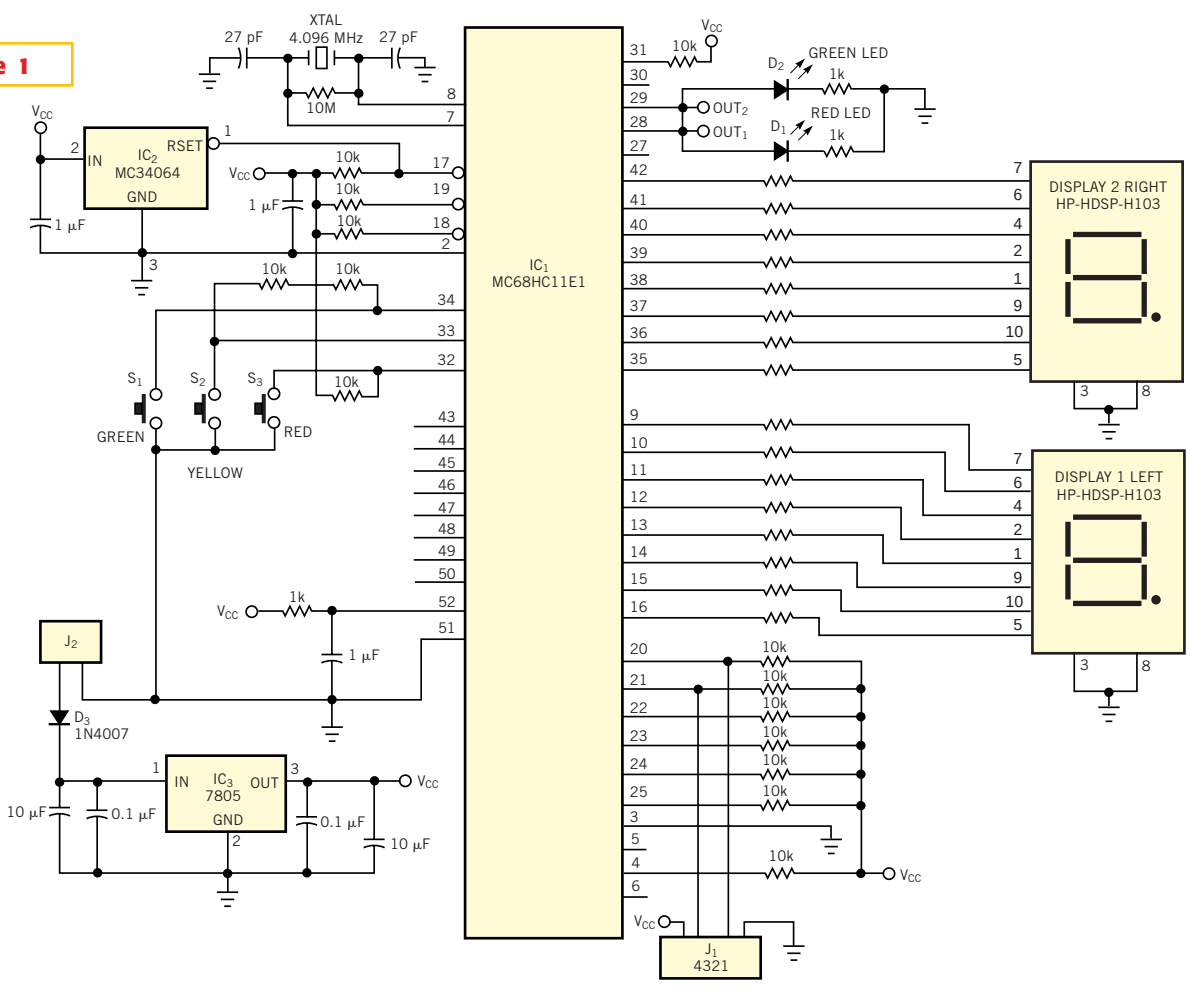
µC provides timer function

Tito Smailagich, ENIC, Belgrade, Yugoslavia

The circuit in Figure 1 is a µC-based programmable timer with two output channels. The first channel, activated by pressing the red switch, S_3, has a red LED at its output. This channel is active until it reaches its desired time-out point. The second output channel connects to a green LED and is active after a preselected time-out period. The second channel remains active until the next depression of the red switch. You can deactivate both channels at any time by pressing the green switch, S_1. In normal mode, the display shows the current remainder of the desired time in seconds. The display decrements by 1 until it reaches 0. The timebase in seconds derives from the main oscillator of the µC, which generates a real-time interrupt every 8 msec. The µC multiplies the 8 msec by 125, yielding a timebase of 1 sec. You program the desired time interval by pressing the yellow switch, S_2; the display shows the programmed value. If you need to change the programmed value, pressing the red switch decrements the value by 1 until it reaches 0, after which it starts with 99. If you need to put the timer interval into memory, press the green switch, and the µC writes the value in its internal EEPROM. You can download the software for the MC68HC11E1 µC from *EDN*'s Web site, www.ednmag.com. Click on "Search Databases" and then enter the Software Center to download the file for Design Idea #2629.

Is this the best Design Idea in this issue? Vote at www.ednmag.com/ednmag/vote.asp.

Figure 1



An MC68HC11 µC provides flexible timing functions.