

# Ethernet 10BaseT simulator jig yields zero emissions

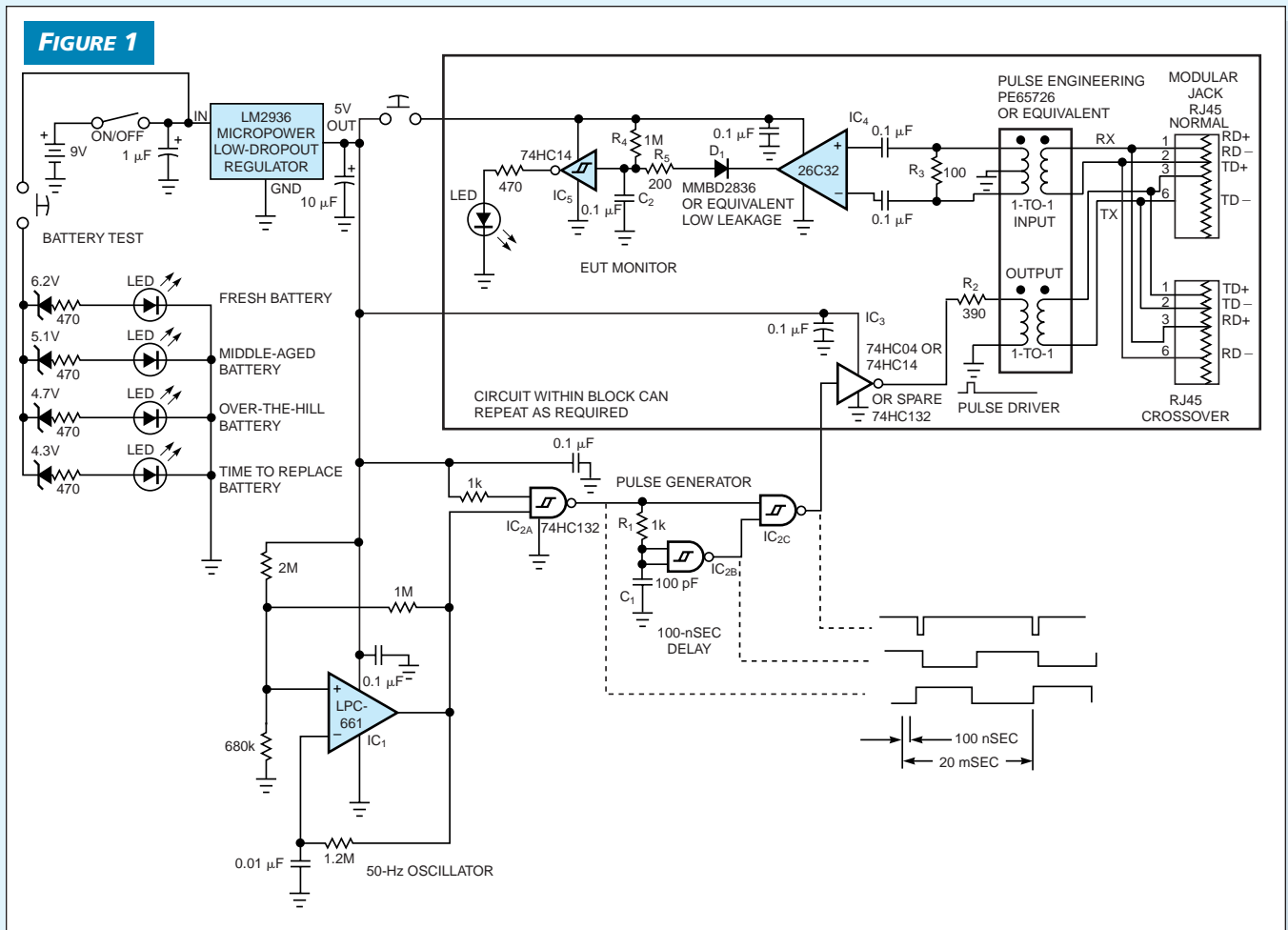
GLEN CHENIER, FUJITSU NETWORK COMMUNICATIONS, RICHARDSON, TX

A test jig (Figure 1) is a valuable tool because it evaluates RF emissions from Ethernet unshielded-twisted-pair (UTP) 10BaseT LAN-interface devices without contaminating the measured results with its own RF emissions. When an RF-emissions-measurement lab tests a multiport, UTP 10BaseT Ethernet device for compliance with FCC-radiated emission limits, the test is meaningful only if the device transmits data packets from all 10BaseT ports. To enable this transmission, the 10BaseT ports must receive a steady stream of link test pulses from attached 10BaseT devices. Unfortunately, the attached devices commonly radiate from their attached cables on the same frequencies as the equipment under test (EUT). This problem makes EUT performance evaluation and any trial fixes difficult, if not impossible. The solution is to

eliminate the radiated noise from the ancillary equipment.

A test generator outside the measurement area can avoid interfering emissions from the generator and its cables by generating the packet data and feeding it over optical fiber to an EUT port. But the EUT 10BaseT ports still need a source of link test pulses to retransmit the data packets. To evaluate EUT emissions from the cables, these sources must connect to the EUT by UTP cables of at least 1m. Only emission-free link-test-pulse sources provide an accurate evaluation of the EUT. The emission-free requirement is difficult to meet when the link-test-pulse sources are 10BaseT devices with noisy 20-MHz clock oscillators and digital circuitry.

The circuit in Figure 1 generates the required link-test pulses without RF emissions. The pulse must have a width of 60



A useful test jig, which includes an equipment-under-test (EUT) monitor and pulse-driver circuit, evaluates RF emissions from Ethernet unshielded-twisted-pair LAN-interface devices without contaminating the measured results with its own RF-emissions.

to 130 nsec with a repetition frequency of 42 to 125 Hz. Pulse amplitude should be 500 mV to 3V. This test jig outputs pulses of about 1V, but you can easily change this level by changing the value of  $R_2$ .

For convenience, the jig operates from a 9V battery. The use of CMOS devices results in a current drain of less than 100  $\mu$ A. According to battery specifications, this current drain translates to an expected battery life of several thousand hours. The duty cycle of the link test pulses is 100 nsec/20 msec=0.0005%, so little battery energy is necessary to drive the EUT receivers. A pushbutton battery-test feature using zener diodes and a four-LED bar-graph display ensures that the battery is capable of the day's testing. To extend battery life, a pushbutton EUT-monitor circuit verifies that the EUT is working properly and is actually transmitting data packets from each port you test.

The frequency of the master oscillator ( $IC_1$ ) is 50 Hz, which spaces the link-test pulses at 20-msec intervals. This low frequency has low harmonic energy in the 30-MHz and higher portion of the RF spectrum; thus, it achieves the primary design goal.  $IC_1$ , a micropower LPC661, has low power consumption, yet its slew rate is fast enough that the input of the following 74HC132 Schmitt trigger stage ( $IC_{2A}$ ) is between the rails for a minimum time.

You can eliminate  $IC_1$  and use  $IC_{2A}$  as the oscillator, but the power consumptions of HCMOS gates and Schmitt triggers tend to rise drastically when the input voltages are not at supply rails. A Schmitt-trigger CMOS oscillator has a constant sawtooth centered linearly between its trip points at its input. The measured difference in the jig's total power drain is 6.5 mA with the HCMOS oscillator and 85  $\mu$ A with the LPC661 oscillator. Long battery life is a secondary design goal.

The pulse generator uses a 100-nsec RC-delay line ( $R_1$ ,  $C_1$ ), and a Schmitt-trigger buffer ( $IC_{2B}$ ) to present the 50-Hz square wave and a 100-nsec delayed version of this square wave to the inputs of NAND gate  $IC_{2C}$ . Thus,  $IC_{2C}$ 's output consists of 100-nsec-wide pulses at 20-msec intervals. The pulse-driver gate ( $IC_3$ ) inverts these pulses to drive the output transformer through a 390 $\Omega$  current-limiting resistor ( $R_2$ ). The transformer is a 1-to-1 Ethernet transformer. You can use other types of transformers with built-in filters; however, the low

pulse-repetition rate does not require a bandlimiting filter. Changing the value of  $R_2$  varies the nominally 1V output's amplitude.

Two RJ45 jacks connect the simulator to the EUT. The test jig includes both normal and crossover-wired jacks so that you always have the jack you need for the available UTP cable.

The EUT-monitor circuit includes the 100 $\Omega$  termination ( $R_3$ ) for the EUT-transmit pair. Keep the leads in this area short and symmetrical to avoid reradiating the data packets from the EUT and causing false emission readings. Even with  $V_{CC}$  off,  $IC_4$  has a high input impedance to avoid generating and reradiating harmonics of the data packets. When you depress the EUT-monitor test button, any input data activity causes the  $IC_4$ 's output to toggle at the packet data rate. The negative-going data pulses discharge  $C_2$ , resulting in a high output from  $IC_5$ 's Schmitt-trigger inverter, which in turn lights the LED.

$R_5$ , a 200 $\Omega$  resistor, slows the capacitor's discharge time constant enough to ensure that any overshoot or ringing from the EUT's positive-going, individual link test pulses do not light the LED. Only actual data packets have a pulse density sufficient to light the LED. Once discharged, the low-leakage diode,  $D_1$ , and the 1M  $R_4$  increase the capacitor's recharge time (the period after the end of the packet when  $IC_4$ 's output is high again). The lengthy recharge time extends the LED's on-time to 100 msec for maximum visibility.

You can add any number of pulse-driver and EUT-monitor sections to the basic circuit, depending on the number of simulator ports and the limits of the LM2936 regulator when all LEDs are on. If you wish to power the EUT-monitor circuit from a separate higher power regulator, place the pushbutton at the additional regulator input so that the regulator's quiescent-current drain does not unnecessarily load the battery. If you don't wish to incorporate the EUT-monitor circuit, you still need to place the 100 $\Omega$  termination resistor across the EUT-transmit pair, but you will no longer need the input transformer. (DI #2235) EDN

To Vote For This Design, Circle No. 347

## Digital potentiometer autonulls op amp

STEPHEN WOODWARD, UNIVERSITY OF NORTH CAROLINA, CHAPEL HILL, NC

Op-amp applications that need the highest possible dc accuracy are generally best served by CMOS chopper-stabilized amplifiers, such as the LTC1050. But high-speed, low-noise applications may require high-performance rockets, such as the 700-MHz LT1226. So what to do for applications that need it *all*? Sometimes, a composite topology in which a

bipolar amplifier provides gain-bandwidth and a CMOS chopper acts as an offset-nulling servo can do the job. Such arrangements can successfully null out offset-voltage errors. But these circuits can get messy if you also need bias-current-related error correction. The circuit in **Figure 1** offers an error-cancellation method that handles both error sources.



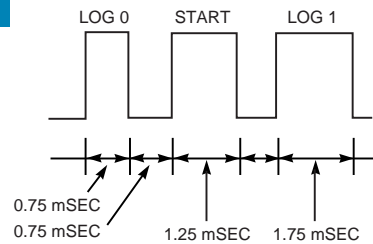
# Single wire connects microcontrollers

ABEL RAYNUS, ARMITRON INTERNATIONAL, MELROSE, MA

Low-cost  $\mu$ Cs, such as Motorola's 68HC705 Series, offer great simplicity at the expense of some useful functions—notably, serial data transmission. Unlike their predecessors, these  $\mu$ Cs do not have serial communication interfaces (SCIs), serial peripheral interfaces (SPIs), or simple serial I/O ports (SIOPs). This method describes how you can overcome this deficiency by creating an asynchronous serial interface through  $\mu$ C software. The most obvious way to effect the interface is to use pulse-width coding to differentiate the start pulse and the logic 1 and 0 pulses. You can use any value of pulse-width ratio, depending on your design objectives. This application uses the 1-to-2-to-3 ratio, slightly modified for easy programming. So, logic 0, start, and logic 1 have widths of 0.75, 1.25, and 1.75 msec, respectively (Figure 1).

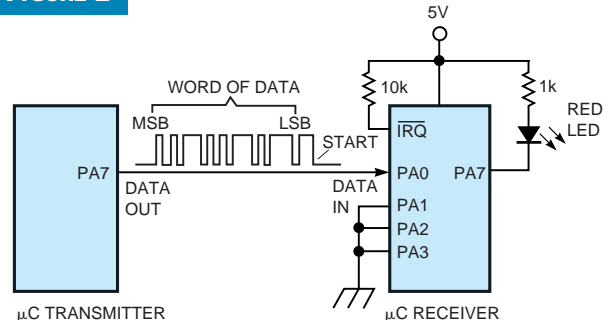
This example uses 68HC705J1A  $\mu$ Cs. The information to transmit accumulates in the output data register of the transmitting  $\mu$ C (Figure 2). The data-transmission subroutine (Listing 1 shows a fragment) generates the start pulse, which is followed by an 8-bit data word that reflects the state of the

FIGURE 1



Pulse-width ratios provide a convenient way to transmit serial data between  $\mu$ Cs lacking communications amenities.

FIGURE 2



With the help of some  $\mu$ C software, a simple one-wire connection provides serial communications between low-cost  $\mu$ Cs.

output data register. This pulse sequence goes, LSB-first, to the data-in input of the receiving  $\mu$ C. In the  $\mu$ C transmitter, you can use any output pin as data out. In the  $\mu$ C receiver, you can use any one of the four lower PortA pins (PA0 through PA3) as data in.

You should program the input pins as positive-edge, external-interrupt inputs. Because pins PA0 to PA3 combine in a logic-OR operation in the  $\mu$ C, you should connect the unused pins to ground to avoid false interruption. You should disable the  $\overline{\text{IRQ}}$  pin by connecting it to 5V. The external-interrupt subroutine (Listing 2) restores the data word, which can generate the proper response according to your design objectives. Listing 3 shows a fragment of the receiver routine. The program's watchdog utility lights a red LED to indicate that the communication link between the  $\mu$ Cs is broken or that it received the wrong sequence. The method also applies to

## LISTING 1—TRANSMITTER PROGRAM FRAGMENT

```

1 *****
2 * TRANSMITTER PROGRAM FRAGMENT *
3 *****
4 * Generates the sequence of start pulse and 8-bit word
5 * of data according to the content of register REG.
6 *****
7 *nclist
8 %include "std-j1a.asm"
9 %list
10 * I/O PORTS
11 data_out equ 7 ;prtA
12 * VARIABLES
13 org RAM
14 REG rmb 1 ;output data register
15 num rmb 1 ;bit test register
16 * INITIALIZATION
17 org MOR
18 fcb $00
19 org ROM
20 %init rsp ;reset stack pointer to $ff
21 lda #$80 ; pA7 as output
22 sta ddrA
23 clr prtA
24 clr REG
25 clr num
26 * DATA TRANSMISSION SUBROUTINE
27 idx #125T ;start pulse (1.25ms)
28 jsr pulse
29 lda #$01 ;0-bit test prepare
30 sta num
31 w1 lda REG ;is tested bit = 0?
32 and num
33 beq w2
34 lda #175T ;logic1 pulse (1.75ms)
35 jsr pulse
36 w3 clr
37 lsl num ;0 -> C-carry bit
38 dcc w2 ;go to next tested bit
39 ;is it NOT a last bit?
39 w2 ;return from DATA TRANSMISSION
40 w2 idx #75T ;log0 pulse (0.75ms)
41 jsr pulse
42 bra w3
43 *****
44 dly01x lda #2 ;Delay =0.01*x [ms]
45 rep0 decA
46 bne rep0
47 decx
48 bne dly01x
49 rts ;return from dly01x
50 *****
51 pulse bset data_out,prtA ;width = x
52 bsr dly01x
53 bclr data_out,prtA
54 idx #64T
55 jsr dly01x ;return from pulse
56 rts ;return from unused interrupts
57 *****
58 un rti ;return from unused interrupts
59 org VECTORS
60 fdb un ;Timer Interrupt unused
61 fdb un ;External Interrupt unused
62 fdb un ;SWI unused
63 init ;set restart address

```

wireless applications with minor modifications. You can download the complete listings from EDN's Web site, [www.ednmag.com](http://www.ednmag.com). At the registered-user area, go into the Software Center to download the files from DI-SIG, #2265. (DI #2265). EDN

To Vote For This Design, Circle No. 349

## LISTING 2—EXTERNAL-INTERRUPT SUBROUTINE

```
ART10.ASM          Assembled with IASM   04/09/1998  11:26  PAGE 2

032E 3FC1          57 * EXTERNAL INTERRUPT SUBROUTINE
0330 3FC0          58 ExtInt: clr      T
0332 000011        50 e0 brset  data,prtA,e1 ;High level ?
0335 01C31C        61 brclr  WF,flag,e2  ;WF=0?
0338 B6C0          62 lda     W
033A A101          63 cmp    #1          ;W=1?
033C 2724          64 beq    log0
033E A103          65 cmp    #3          ;W=3?
0340 2728          66 beq    log1
0342 11C3          67 bclr  WF,flag    ;0 -> WF
0344 2041          68 bra   e3
0346 3C11          69 e1   inc  T
0348 B6C1          70 lda  T
034A A137          71 cmp  #55T
034C 26E4          72 bne  e0
034E 3FC1          73 clr  T
0350 3CC0          74 inc  W
0352 20DE          75 bra  e0
0354 B6C0          76 e2   lda  W
0356 A102          77 cmp  #2          ;W=2?
0358 262D          78 bne  e3
035A 10C3          79 bset  WF,flag    ;1 -> WF
035C A6FE          80 lda  #5fe       ;0 -> 0-bit of num
035E B7C4          81 sta  num
0360 2025          82 bra  e3
0362 B6C5          83 log0 lda  reg    ;put 0 into given bit of
0364 B4C4          84 and  num       ;reg. without changing
0366 E7C5          85 sta  reg       ;of the rest of its bits.
0368 200C          86 bra  e4
036A B6C4          87 log1 lda  num    ;put 1 into given bit of
036C B7C6          88 sta  mem       ;reg. without changing
036E B4C5          89 and  reg       ;of the rest of its bits
0370 33C6          90 com  mem
0372 B8C6          91 eor  mem
0374 B7C5          92 sta  reg
0376 99          93 e4   sec        ;1 -> Carry bit
0377 39C4          94 rol  num       ;go to the next bit
0379 250C          95 bcs  e3        ;is it NOT the last bit?
037B 11C3          96 bclr  WF,flag    ;0 -> WF word process flag
037D 17C3          97 bclr  WDF,flag
;0*****
```

## LISTING 3—RECEIVER-PROGRAM FRAGMENT

```
1
2 * RECEIVER PROGRAM FRAGMENT
3 *****
4 * Transfers the received serial data
5 * into content of the register WORD.
6 *****
7 *nolist
8 $include "std-ja.asm"
9 *list
10
11 org MOR ;pos.edge Ext.Interrupt
12 fcb $24 ; on pA0 - pA3 enable
13
14 *I/O PORTS
15
16 data equ 0 ;prtA Data Input pin
17 RedLED equ 7 ;prtA red LED output pin
18
19 *Specific equates
20
21 WF equ 0 ;Word processing flag
22 WDF equ 1 ;signal preense flag
23 WDF equ 3 ;watch-dog flag
24
25 * VARIABLES
26
27 org RAM
28 W rmb 1 ;pulse width counter
29 T rmb 1 ;time (0.5 ms) counter
30 wdc rmb 1 ;watch-dog counter
31 flag rmb 1 ;flag register
32 num rmb 1 ;register to form word
33 reg rmb 1 ;temporary word register.
34 mem rmb 1 ;memory register
35 WORD rmb 1 ;final received word register.
36
37 * INITIALIZATION
38
39 org ROM
40 init rsp ;reset stack pointer to $ff
41 lda #5f0 ;pA0 - pA3 as input
42 sda dda ;pA4 - pA7 as output
43 jsr in_set ;go to initial set
44
45 bset IRQ5,ISCR ;ExtInt enable
46 bset TOIE,TSCR ;TOF interrupt enable
47 cll ;interrupt enable
48 brset WDF,flag,m1 ;WDF=1? No data-in?
49 brset WF,flag,m0 ;WF=1? wait for word end
50
51 ml jsr in_set
52 bra m0
53 *****
54 in_set clr prtA ;set red LED on
55 sf clrx ;start to clear
56 a0 clrx RAM,x ; 8 variables in RAM
57
58 incx
59 cpx #8T
60 blo a0
61 rts ;return from in_set
62 *****
63 TOFint inc wdc
64 tst wdc ;wdc / 0 ?
65 bne t0
66 bset WDF,flag ;1 -> WDF
67 t0 bset TOFR,TSCR ;TOF reset
68 rti ;return from TOFint
```

# Photo-flash charger minimizes parts count

STEVEN CHENETZ, MICREL SEMICONDUCTOR, SAN JOSE, CA

Photo-flash and strobe devices operate by discharging a high-voltage capacitor into a bulb. Charging the capacitor from a battery or other low-voltage source requires a step-up dc/dc converter to boost the voltage, typically to 300V. One way to generate the high voltage is to use a flyback converter. The circuit in **Figure 1** provides a simple and reliable way to charge a high-voltage capacitor. The flyback converter performs two functions: It boosts the low-voltage input and provides isolation between the input (battery) and output (high voltage). Its main components are the power transformer; the output diode; the output capacitor; and the MIC3172 controller chip, which combines the switching transistor, voltage regulator, and control logic.

The transformer stores energy when the internal transistor

of the MIC3172 turns on, allowing current to flow through the transformer's primary. When the transistor turns off, the stored energy flows through the output rectifying diode and into the capacitor. The voltage across the capacitor increases with each switching cycle until it reaches the preset voltage. The resistive divider  $R_2/R_2/R_3$  and the 1.24V reference in the IC determine the preset output voltage:  $V_{OUT} = V_{REF} (R_1 + R_2 + R_3) / R_3$ .

Once the capacitor voltage reaches the preset value, the MIC3172 stops switching. Current flow in the output components cause the capacitor to discharge. The MIC3172 provides occasional energy pulses that keep the capacitor fully charged. When the capacitor discharges into the bulb, the charging process repeats.  $D_1$  and  $D_2$  clamp any voltage spikes

on the collector of the MIC3172 switch node, caused by leakage inductance on the transformer. When the IC's internal transistor turns off, the voltage across the transformer's primary approximately equals the output voltage divided by the turns ratio. The voltage at the transistor collector node (Pin 7) equals the reflected voltage plus the input voltage, plus the voltage spike caused by the leakage energy in the transformer:

$$V_{SW} = (V_{OUT}/N) + V_{IN} + V_{LEAKAGE}$$

The collector-node voltage must always be less than 65V. The zener-diode voltage is set greater than the maximum reflected voltage at the transformer primary. For **Figure 1**, the reflected voltage is 10V. The zener diode is a 12V device, approximately 20% greater than the reflected voltage. The maximum reverse voltage across  $D_2$  equals the maximum input voltage. This diode must be an ultrafast or Schottky device, to prevent excessive losses in the diode.

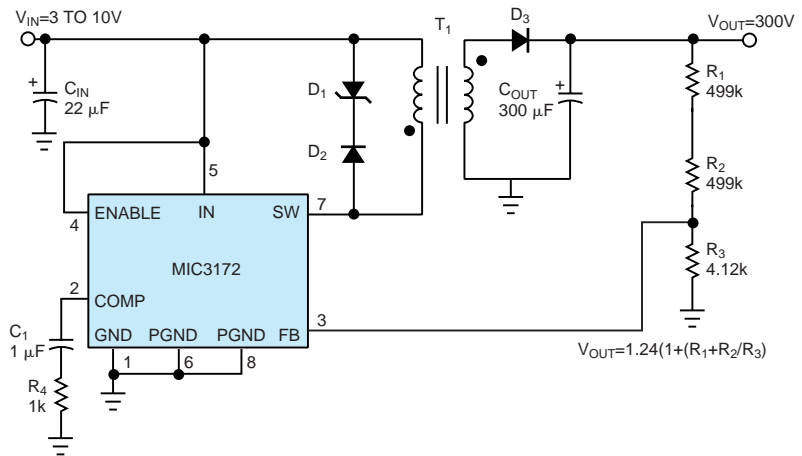
The energy stored in the capacitor is  $0.5CV^2$ . The output power that the flyback converter requires to charge the capacitor in a period  $T$  is  $(0.5CV^2)/T$ . The following formula gives the approximate charging time for the converter circuit:

$$T_{CHARGE} = \frac{C_{OUT} V_{OUT}^2}{2V_{IN} I_{PK} \eta}$$

where  $I_{PEAK}$  is the peak current level of the MIC3172 control chip (typically, 1.8A);  $D$  is the maximum duty cycle (approximately 0.6); and  $\eta$  is the efficiency of the flyback converter (0.5).

Charging a 300- $\mu$ F capacitor to 300V from a 5V input requires  $(300 \mu\text{F} \times 300\text{V}^2) / (2 \times 5\text{V} \times 0.6 \times 0.5) = 5$  sec. For the circuit in **Figure 1**, the output voltage is potentially lethal. At 300V, the energy in the output capacitor is 27J, more than enough to ruin an otherwise good day. When you lay out the circuit, be sure to provide adequate spacing between the high- and low-voltage sections. The power transformer, such as the Coiltronics CTX04-13770, must have the proper spacing and insulation between the high-voltage secondary and low-voltage primary.

The circuit uses two resistors,  $R_1$  and  $R_2$ , in the upper section of the output to reduce voltage stress, because most commonly available resistors are rated at 200 to 300V—too close to the limit for reliable, long-term operation. If  $R_1$  or  $R_2$  should open or if  $R_3$  shorts, the converter runs open-loop at its maximum duty cycle. This failure mode boosts the voltage far above the preset limit and causes the output capacitor to vent. The circuit in **Figure 2** provides overvoltage protection.

**FIGURE 1**


NOTES: IC=MICREL MIC3172BM BOOST-CONVERTER IC.

$C_{IN}$ =AVX TPS SERIES TANTALUM.

$T_1$ =1-TO-30 STEP-UP FLYBACK TRANSFORMER; LPRI=35  $\mu$ H;

COILTRONICS P/N CTX04-13770.

$D_1$ =MOTOROLA 1SMB5927BT3, 12V ZENER.

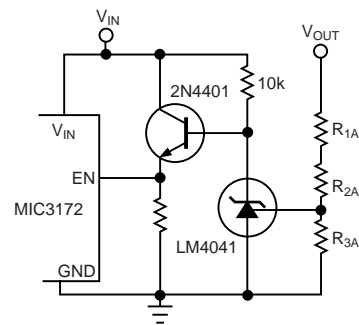
$D_2$ =MOTOROLA MBR5130LT3, 1A, 30V.

$D_3$ =MOTOROLA MURS160T3, 1A, 600V.

$R_1$  TO  $R_4$ =METAL-FILM RESISTORS, 1206 (200V RATED).

$C_1$ =CERAMIC CAPACITOR, X7R SERIES.

Generate 300V from a low-voltage source, using this simple, low-parts-count circuit.

**FIGURE 2**


Avoid the exploding-capacitor syndrome by using this overvoltage-protection circuit with the circuit of Figure 1.

Be sure that the resistor divider for the overvoltage circuit is separate from the voltage-regulation divider. Set the overvoltage level 15% higher than the output-voltage setting, and make sure it does not exceed the capacitor's voltage rating. (DI #2266).

**EDN**

To Vote For This Design, Circle No. 350

# Generate an analog signal with a $\mu\text{C}$

THOMAS SCHMIDT, MICROCHIP TECHNOLOGY, CHANDLER, AZ

Applications requiring D/A conversion abound, including dual-tone generation, motor-speed control, and offset-voltage generation for a sensor or for battery charging. Most designers believe the D/A converter must be either an integrated module in a  $\mu\text{C}$  or an external component; however, a simpler approach is possible. You can generate an analog signal by using a low-cost  $\mu\text{C}$ , thereby eliminating the need for external components and thus reducing board space and overall system cost. The RC network in **Figure 1** provides an easy way to convert a digital signal into an analog voltage. The RC network, a lowpass filter, connects to an I/O pin of the  $\mu\text{C}$ .

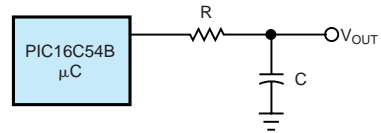
To generate an analog signal, the  $\mu\text{C}$  charges the capacitor via the resistor. The  $\mu\text{C}$  uses PWM to charge the capacitor. The voltage across the capacitor is the analog voltage. When the PWM signal is high, the capacitor charges. When the PWM signal is low, the capacitor discharges. By varying the duty cycle, you can generate a sine wave or any other analog signal. The  $\mu\text{C}$  in **Figure 1** is a low-cost, 8-bit RISC controller. The PWM signal, generated in software, drives the RC lowpass filter connected to one of the  $\mu\text{C}$ 's I/O lines.

**Listing 1** gives the software code. The PWM routine requires only three general-purpose registers. One register contains the value of the period; the other contains the duty cycle. The program starts by initializing the PWM output pin and the period and duty-cycle registers. The initial duty cycle is 50%. It's assumed in the initialization routine that the RC network connects to pin RA1 of the  $\mu\text{C}$ . After initialization, the main subroutine calls the routine in which the PWM signal is generated. In this example, the main routine calls only the PWM\_Signal routine. You could easily implement other functions—a keypad or a seven-segment display, for example—just by adding call instructions for subroutines.

The PWM implementation requires a software counter. The register counter stores the software counter. The counter increments each time the program calls the PWM\_Signal routine. Each time the counter increments, the program checks to see if the register's value is greater than or equal to the duty cycle. If this condition is true, the program sets the value at the port pin to logic 0. This action signifies that the time for the duty cycle has elapsed. After this time elapses, the routine checks to see if the time for the period is over. If the value of the counter is less than the value of the duty cycle, the PWM signal remains high.

If the value of the counter is greater than the value of the duty cycle, the program compares the counter with the value of the period register. If the value of the counter equals the value of the period register, the period for the PWM signal is over, and the next period starts. The performance of this PWM implementation depends on the number of times the program calls the PWM function from the main routine—the more calls, the higher the resolution. To generate a sine wave,

FIGURE 1



A simple RC network connected to a low-cost  $\mu\text{C}$ 's output can generate analog signals of any desired resolution.

## LISTING 1—ROUTINE FOR ANALOG-SIGNAL GENERATION

```

list p=pic16c54B, r=hex
#include <pic5x.inc>
#define PWM_PORT PORTA
#define PWM_PIN 0

DutyCycle EQU 0x09
Counter EQU 0x0A
Period EQU 0x0B

ORG 0x00
Begin call Initialize_PWM
Main call PWM_Signal
goto Main

Initialize_PWM
PORTA ; reset PORTA
movlw b'00001110'
tris PORTA ; Set PWM pin for output

movlw 80
movwf DutyCycle ; initialize duty cycle
movwf FF
movwf Period ; initialize period register
clrf Counter ; reset counter
bsf PORTA, PWM_PIN ; set PWM signal to high
retlw 0

PWM_Signal
incf Counter,f ; Increment the counter
movf DutyCycle,w
subwf Counter,w ; compare duty cycle against
; period
btfss STATUS,C ; is period < duty cycle
; duty cycle is greater than
; period, therefore PWM signal
; remains high

bcf PORTA, PWM_PIN ; time of duty cycle
; elapsed
movf Period,w ; compare the counter
; against the period

subwf Counter,w
btfss STATUS,Z ; if the counter ==
; period,
; period is not over
retlw 0 Counter ; reset the counter
bsf PORTA, PWM_PIN ; Set PWM_Pin to high
retlw 0

ORG 0x1FF
Reset goto Begin
END

```

for example, you can store the values for the duty cycle in a look-up table. The values in the look-up table depend on the values and tolerances of the resistor and capacitor and on the desired resolution of the sine wave. You can download the **listing** from EDN's Web site [www.ednmag.com](http://www.ednmag.com). At the registered-user area, go into the Software Center to download the files from DI-SIG, #2268. (DI #2268). EDN

To Vote For This Design, Circle No. 351

## Design Idea Entry Blank

Entry blank must accompany all entries. \$100 Cash Award for all published Design Ideas. An additional \$100 Cash Award for the winning design of each issue, determined by vote of readers. Additional \$1500 Cash Award for annual Grand Prize Design, selected among biweekly winners by vote of editors.

To: Design Ideas Editor, EDN Magazine  
275 Washington St, Newton, MA 02158

I hereby submit my Design Ideas entry.

Name \_\_\_\_\_

Title \_\_\_\_\_

Phone \_\_\_\_\_ Fax \_\_\_\_\_

E-mail \_\_\_\_\_

My e-mail address may be published Yes \_\_\_\_ No \_\_\_\_

Company \_\_\_\_\_

Address \_\_\_\_\_

Country \_\_\_\_\_ ZIP \_\_\_\_\_

Design Idea Title \_\_\_\_\_

Social Security Number \_\_\_\_\_  
(US authors only)

Entry blank must accompany all entries. (A separate entry blank for each author must accompany every entry.) Design entered must be submitted exclusively to EDN, must not be patented, and must have no patent pending. Design must be original with author(s), must not have been previously published (limited-distribution house organs excepted), and must have been constructed and tested. Fully annotate all circuit diagrams. Please submit software listings and all other computer-readable documentation on a IBM PC disk in plain ASCII.

Exclusive publishing rights remain with Cahners Publishing Co unless entry is returned to author, or editor gives written permission for publication elsewhere.

In submitting my entry, I agree to abide by the rules of the Design Ideas Program.

Signed \_\_\_\_\_

Date \_\_\_\_\_

Your vote determines this issue's winner. Vote now, by circling the appropriate number on the reader inquiry card.

**CIRCLE NO. 9**