design ideas

Edited by Bill Travis and Anne Watson Swager

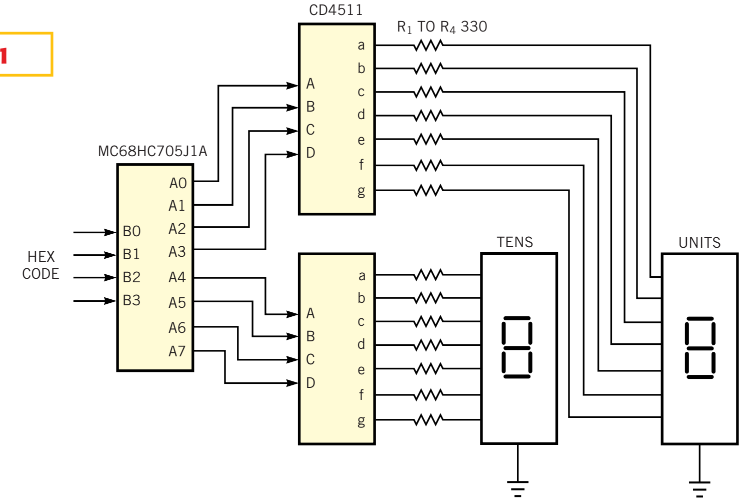
μC visualizes hex code

Abel Raynus, Armatron International, Melrose, MA

IN μC SYSTEMS, information exchanges usually use the hexadecimal 1-2-4-8 format; output data also appear in this format. Reading the hex code is not a problem; several LEDs connected to the output lines can display the answer. The problem arises when you wish to observe the output data. Many engineers are unfamiliar with hex code and prefer to observe data in the common decimal format. If the value of the output data is less than 10, you can use an ordinary BCD-to-seven-segment decoder to visualize the hex code on an LCD or an LED display. But what do you do if the value of the output data is greater than 10? Unfortunately, no decoders can transform hex code into two seven-segment codes. Of course, you could configure such a decoder using a number of logic gates, but another simple and inexpensive option is available. The key to this option is using a low-end μC to transform the hex code into two BCD codes. In **Figure 1**, the data displayed ranges from 0 to 15. Thus, you need only 4-bit hex code, using four input and eight output μC lines.

Figure 1 uses the approximately $1 Motorola MC68HC705J1A, with 14 I/O pins. The input lines Pin B0 to Pin B3 re-

Figure 1



You can visualize hex code in BCD format by using this simple circuit.

LISTING 1—HEX-TO-BCD CONVERSION

```
                    * HEX CODE TO 2 BCD CODE TRANSFER*
 0000               $pagewidth 160
 0000               $include "std-jla.asm" ;standard frame for
MC68HC705J1A
                    *VARIABLES
 00C0                       org    RAM
 00C0               M1      rmb    1
                    *INITIALIZATION
 07F1                       org    MOR;
 07F1      20               fcb    %00100000;resistor osc
 0300                       org    ROM
 0300 [02] A600     init    lda    #%00 ; prtB as input
 0302 [04] B705             sta    ddrB
 0304 [02] A6FF             lda    #$ff ; prtA as output
 0306 [04] B704             sta    ddrA
 0308 [05] 3F00             clr    prtA ; 0 -> prtA
                    *****************************************************
 030A [03] B601     main    lda    prtB
 030C [02] A40F             and    #%00001111 ; extract data from pB0-pB3
 030E [04] B7C0             sta    M1
 0310 [02] A109             cmp    #$9        ; data > 9?
 0312 [03] 220A             bhi    h1
 0314 [02] A600             lda    #$00       ;set 00 to pA4-pA7 (tens)
 0316 [04] B700     h2      sta    prtA
 0318 [03] BAC0             ora    M1
 031A [04] B700             sta    prtA       ;set data to pA0-pA3
(units)
 031C [03] 20EC             bra    main
 031E [02] A00A     h1      sub    #$a        ;(data-10) -> Acc
 0320 [04] B7C0             sta    M1
 0322 [02] A610             lda    #$10       ;set 1 to pA4-pA7 (tens)
 0324 [03] 20F0             bra    h2

*****************************************************
 07FE                       org    VECTORS+6
 07FE      0300             fdb    init
```

ceive the 4-bit hex code to be displayed. The μC-assembly program in **Listing 1**, converts the hex code into two BCD codes, which appear on lines Pin A0 to Pin A3 (units) and Pin A4 to Pin A7 (tens). These outputs drive the two standard BCD-to-seven-segment decoders, which, in turn, drive the common-cathode LED displays. You can use the same method for an expanded data range, but you need more I/O lines, decoders, and displays. For example, the 8-bit hex code covers the data range 0 to 255, but it needs eight input lines, 12 output lines, and three decoders and displays. You can download **Listing 1** and the "include" file in line 3 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 #2518. (DI #2518)

To Vote For This Design,
Circle No. 301

Simple circuit times bathroom fan

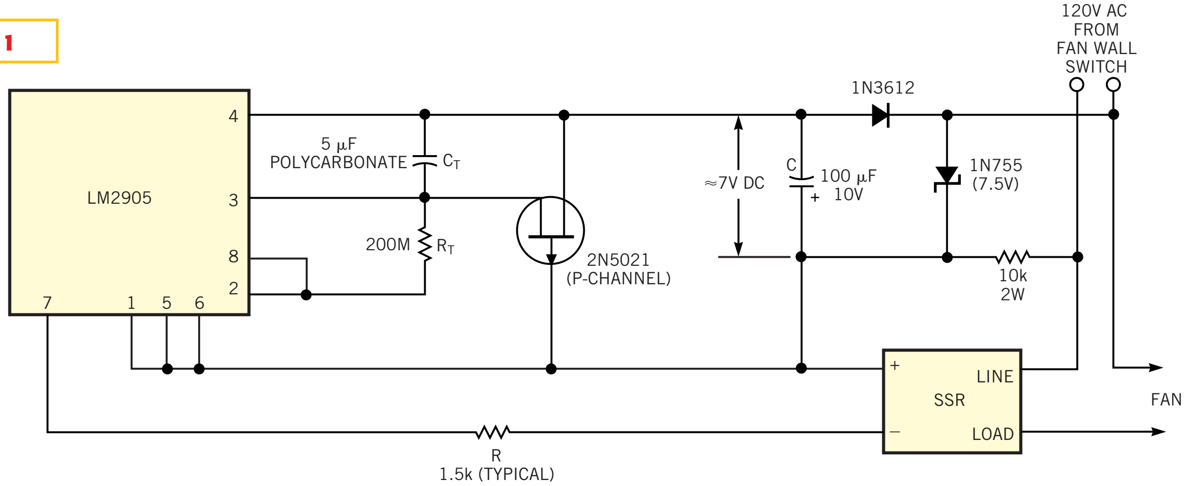
Maxwell Strange, Goddard Space Flight Center, Fulton, MD

Forget to turn off the ceiling fan in your bathroom? Install the simple timer in **Figure 1**. It's located out-of-sight in the fan unit, and you turn it on via the wall switch. The circuit costs virtually nothing, using "junk" parts. When ac power appears, a simple rectifier develops approximately 7V across filter capacitor C. This voltage powers the LM2905 analog timer and simultaneously triggers it via trigger-input Pin 1; output Pin 7 then goes low, turning on the solid-state relay and fan. Low-leakage capacitor C_T and resistor R_T set the time delay; for the values shown, the delay is 1000 sec, or approximately 17 minutes. At time-out, Pin 7 goes high, turning off the relay. For most fans, this relay can be a small, pc-mountable unit rated at as little as 0.5A ac load current. These relays typically turn on reliably at 3V. You can use resistor R to drop excess voltage, thereby reducing loading on the power supply.

The p-channel JFET turns on when the power switches off, rapidly discharging C_T to allow immediate recycling. The 10-kΩ series resistor makes the circuit inherently safe; in a worst-case failure, line voltage appears across this resistor and develops a harmless 1.4W. You can easily adapt this circuit to other applications. You can change the timing or make it linearly adjustable, and you can program the solid-state relay to turn on instead of off at time-out, by connecting Pin 8 of the timer to Pin 4 instead of Pin 2. You can also reverse the action of the solid-state relay by reversing the control inputs. (DI #2520)

To Vote For This Design,
Circle No. 302

Figure 1



Gain control of your bathroom fan by using this simple time-out circuit.

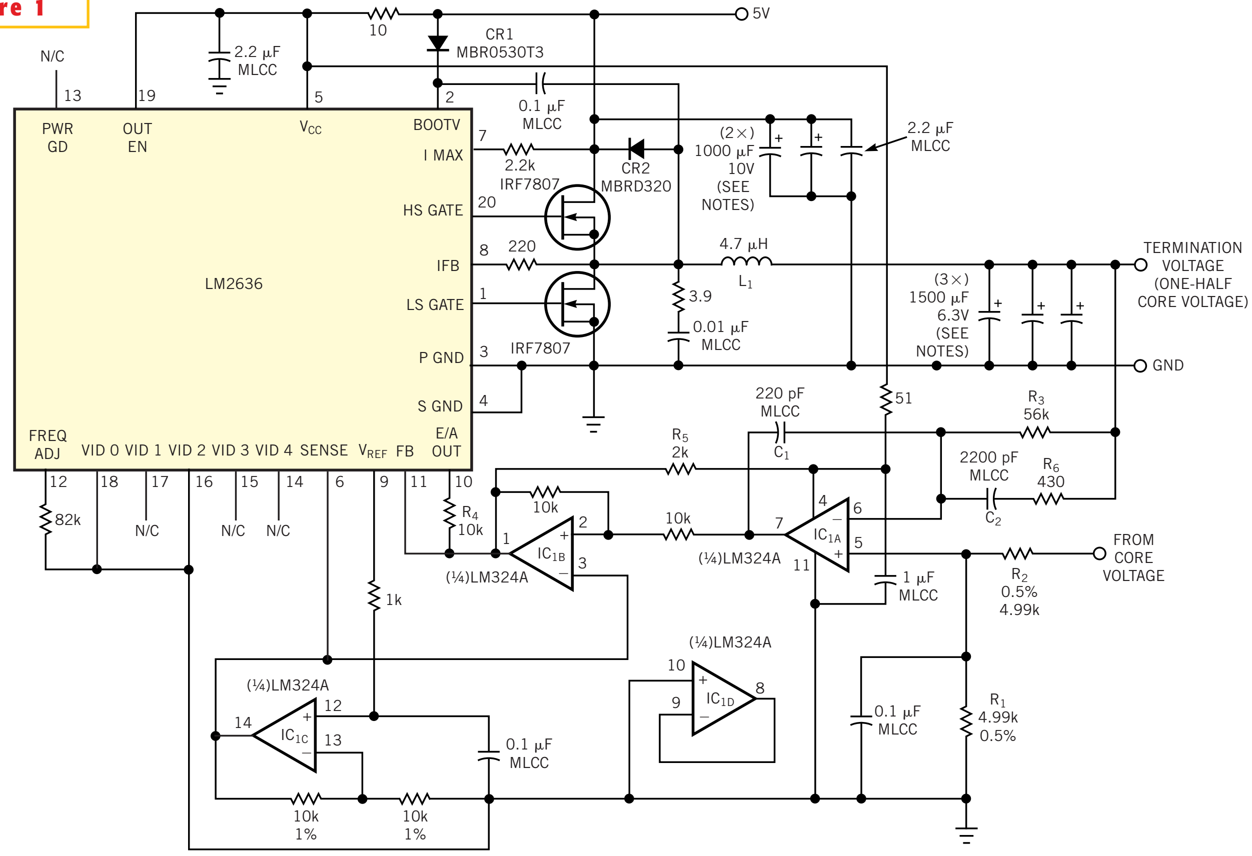
Termination supply tracks one-half core voltage

Chester Simpson, National Semiconductor, Santa Clara, CA

Modern µPs typically require core voltages of 2 to 3V. They also require "termination" voltages that, for best performance, should equal half the core voltage. The problem is, core voltages vary among CPUs, and some systems even incorporate variable-core-voltage supplies that allow systems to adjust the voltage on the fly, thereby increasing or reducing CPU operating speed to optimize power consumption. A termination-voltage power supply that maintains an output of one-half V_{CORE} over a range of core voltages is highly desirable, as the core voltage can vary without upsetting the termination-voltage set point. The circuit in **Figure 1** is a 6A power-supply design that generates a termination voltage regulated to 1/2 $V_{CORE.}$ The circuit targets applications in which the core voltage is approximately 1.8 to 3.6V.

An LM2636 synchronous-rectifier controller switching at 300 kHz provides an efficient power converter that operates from a 5V input. Because the LM2636 is designed to operate at a fixed output voltage (as determined by control bits 14 through 18), the circuit in **Figure 1** uses a different control scheme to force the regulated output to track at one-half V_{CORE}. Resistors R_1 and R_2 halve the core voltage, and this voltage serves as the reference in error amplifier IC_{1A}. The amplifier compares the one-half V_{CORE} reference with the termination output voltage obtained through R_3 and adjusts its output to lock the termination voltage at one-half V_{CORE}. In this way, the core-voltage signal sets the termination voltage. IC_{1B} is a unity-gain inverter that corrects the phase of the feedback signal that goes to the input of the LM2636's inter-

Figure 1



NOTES:
USE LOW-ESR ALUMINUM ELECTROLYTIC CAPACITOR SUITABLE FOR 300-kHz SWITCHING APPLICATIONS.
UNLESS OTHERWISE SHOWN, RESISTOR TOLERANCES ARE 5%.
MLCC DESIGNATES A CERAMIC CAPACITOR WITH AN X7R OR X5R TEMPERATURE RANGE.

Optimize your CPU's performance by feeding it a termination voltage that's exactly half the core voltage.

nal error amplifier. R_4 sets the gain of the internal amplifier to unity. R_5 forces a soft start on turn-on and also eliminates overshoot. IC_{1C} amplifies the 1.23V internal reference to approximately 2.5V, which sets the operating point for the error amplifier, IC_{1A}.

The 2.5V also connects to the sense input of the LM2636, which would normally sense the regulated output voltage. Because the termination output voltage must be variable (to track the core voltage), a fixed 2.5V goes to the sense pin, and the control pins 16 and 18 are grounded. These connections program the internal DAC for a 2.5V output. This scheme prevents the LM2636's internal error-detection circuitry from shutting down the part in response to an undervoltage or overvoltage condition. R_3, R_6, C_1, and C_2 provide loop compensation. (DI #2517)

To Vote For This Design,
Circle No. 303

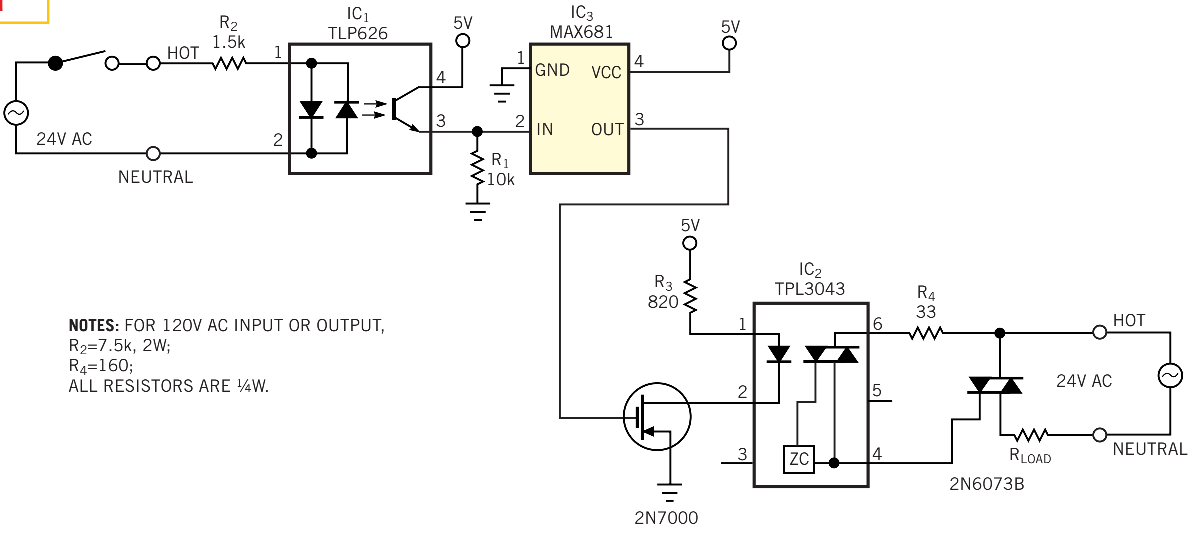
Switch debouncer isolates input and output

Phill Leyva, Maxim Integrated Products, Sunnyvale, CA

Modern programmable-logic controllers (PLCs) for automated process-control systems have either 16 or 32 inputs and accept ac voltages of 24 to 120V. A single circuit (a relay for isolation and an RC network with a Schmitt trigger to debounce signals to the processor) can debounce all the PLC signals in sequence. However, this approach slows real-time data processing. Such debounce circuits also produce delay times that change with relay wear and capacitor aging. In the PLC program, you can use a debounce timer for each input, but this technique increases the program-scan time and ties up valuable timers. The solid-state, electrically isolated circuit in **Figure 1** debounces single inputs without slowing the PLC module. Optoisolators IC_1 and IC_2 provide electrical isolation for the ac sources at the input and output. IC_3 is a CMOS switch debouncer whose output (a 4V logic high) appears following a fixed 40-msec delay. A 63-kΩ pullup resistor, connected internally between IN and V_{CC}, forms a voltage divider with R_1. R_1's value ensures a logic low of less than 0.8V at IN when IC_1's optotransistor (an emitter follower) is off.

The two LEDs in IC_1, which illuminate the phototransistor on alternate half cycles of the ac-input current, rectify this current. Most optoisolator applications set the current-transfer ratio (CTR) to more than 10 to ensure an accurate reproduction of the input signal. The circuit in **Figure 1**, however, sets the CTR to less than 1, which ensures that the emitter follower does not turn off as the ac current goes to zero twice in each cycle. R_2 biases the emitter follower such

Figure 1



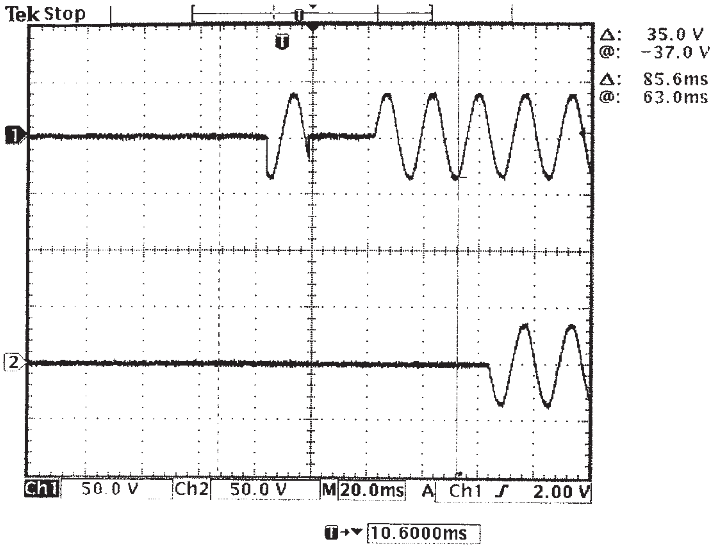
NOTES: FOR 120V AC INPUT OR OUTPUT,
R_2=7.5k, 2W;
R_4=160;
ALL RESISTORS ARE ¼W.

This debouncer circuit allows an isolated ac voltage to control a separately isolated ac source.

that IC_3's IN signal remains greater than the high level, 2.4V, during these zero crossings. This action eliminates the capacitor normally found in debouncing circuits. IC_3's OUT pin drives the n-channel MOSFET, and R_3's value is such that the resulting current flow (in the MOSFET and LED) is approximately 5 mA. When the MOSFET turns on, the LED activates IC_2's zero-crossing triac driver. Thus, when the power triac turns on, an ac source connected to the output drives R_{LOAD} with as much as 4A (**Figure 2**). Turning on the triac at zero crossings eliminates EMI and reduces the turn-on stress in the triac. R_4 limits current into the triac driver (IC_2) to 1A. Each source can be either 24V ac (as shown) or 120V ac. (DI #2521)

To Vote For This Design,
Circle No. 304

Figure 2



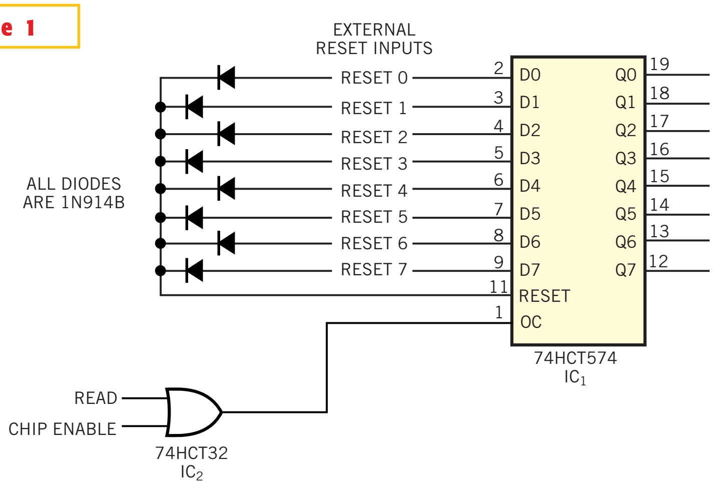
An ac switch (top trace) turns on and then briefly bounces off; the output of the associated debouncer circuit turns on cleanly (bottom trace).

Circuit detects reset source

Shyam Tiwari, Sensors Technology Pvt Ltd, Gwalior, India

Embedded single-board computers do not rely only on power-on resets; they often use multiple sources to reset CPUs to prevent CPUs from being locked into endless loops. However, in most cases, CPUs start from the starting points of memories to fetch their first code, because they have no way of knowing what generated the reset. The circuit in **Figure 1** allows a CPU to know the source of a reset. It stores in an 8-bit latch reset-control input data that the CPU can read. If an input frequently resets the CPU, the CPU can then report the error source to the user, using LEDs or other indicators. In the circuit of **Figure 1**, an experiment combined eight independent reset-signal sources into a wired-OR single output. The 74HCT574 stores the 8-bit reset data at the rising edge of the signal. The latch records no other signal if the signal appears after the rising edge of the first signal. If the CPU finds data that resembles a reset signal in the latch that the signal does not reset, then the reset pulse is too narrow to effect a reset. The CPU recognizes this signal after reading the information from the latch. The output of the latch is a tristate structure; the CPU reads the output using active chip-enable and read-input signals. (DI #2516)

Figure 1



This circuit detects the source of a CPU-reset signal.

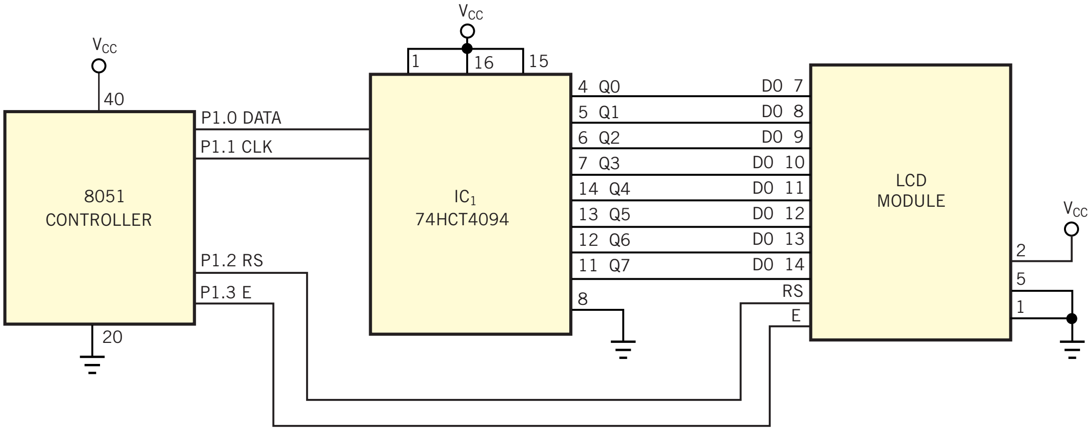
To Vote For This Design,
Circle No. 305

Interface LCD with ease

Bharat Mehta, Space Applications Center, Ahmedabad, India

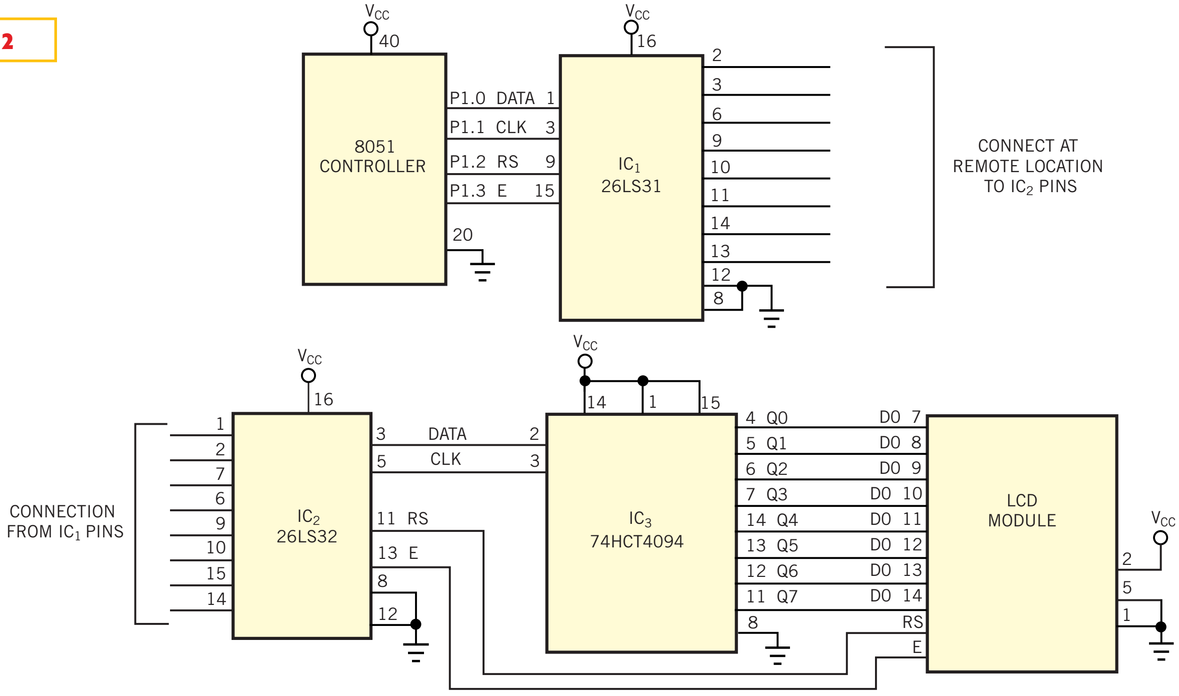
Alphanumeric LCDs find use in various embedded systems. In these systems, μCs often lack I/O resources, so you must make optimum use of the available resources. LCDs have a bus-oriented interfacing structure, which consumes 11 I/O lines (eight data lines and three control lines). You cannot always afford to spare this many I/O lines. Most applications need to write the data to the LCD and then read from it. In these situations, you can omit one control pin (of three) for reading data and save one additional I/O-port pin. The circuit in **Figure 1** saves I/O pins. It uses an 8051

Figure 1



Save μC I/O pins by effecting a serial-to-parallel conversion with a shift register.

Figure 2



Extend Figure 1's scheme to long distances by using line-driver and -receiver ICs.

µC, and only four I/O pins (P1.0 to P1.3) drive the LCD, instead of the 11 pins other displays require. Data transfers in a serial mode through the P1.0 port of the µC. The µC interfaces to Pin 2 of IC_1, a 74HCT4094 8-bit shift register. Serial data advances on every clock pulse and transfers to the shift register. The register converts the serial data to parallel data, available on the output pins Q0 to Q7. The P1.1 port of the µC provides the clock.

The data bus, D0 to D7, of the LCD module connects to the shift register's outputs. Software carries out the data transformation and displays the result on the LCD (**Listing 1**). The same design can drive various types of alphanumeric LCDs—for example, single-line-to-multiline types, with different character lengths on each line. You can configure the circuit to send data over a long distance for remote LCD readouts (**Figure 2**). You can transmit serial data through a differential line-driver IC such as the 26LS31 (IC_1). You can feed the output of IC_1 to either twisted-pair wires or a parallel pair of wires for transmission to a remote location. At the other end of the remote location, you can retrieve the data through a differential line receiver such as the 26LS32 (IC_2). The output of IC_2 drives IC_3, which drives the LCD as in **Figure 1**.

Listing 1, written in 8051 assembly code, provides the sample text "Hello, EDN Reader" on a single-line, 16-character LCD. You can download **Listing 1** 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 #2519. (DI #2519)

TO VOTE FOR THIS DESIGN,
CIRCLE NO. 306

LISTING 1—SAMPLE DATA DISPLAY

```
temp1:  reg     20h
Dat:    reg     P1.0
Clk:    reg     P1.1
RS:     reg     P1.2
ENA:    reg     P1.3

;;;;;;;;;;;;;;;;  Main progarm   ;;;;;;;;;;;;;;

        org     0000h
        jmp     start

        org     30h

start:  call    INT
        mov     dptr,#title
        mov     r0,#10h         ;counter for  data transfer

loop1:  clr     a
        movc    a,@a+dptr
        mov     temp1,a
        djnz    r0,loop2
        call    dly
        jmp     start

loop2:  call    sr_tx
        setb    RS      ;This part transfer data to LCD.
        setb    ENA
        call    delay   ;small delay
        clr     ENA
        clr     RS
        inc     dptr
        jmp     loop1

title:  db "Hello,EDN Reader"

;;;;;;;;;;;;;;;;;  Main program End  ;;;;;;;;;;;

;------  Sub-Routine  programs  --------
;;; initialization of LCD screen.

INT:    mov     sp,#60h
        clr     RS
        clr     ENA
        mov     temp1,#38h
        call   disp2
        mov     temp1,#01h
        call    disp2
        mov     temp1,#02h
        call    disp2
```

```
        mov     temp1,#06h
        call    disp2
        ret
;;;;;;;;;;;;;;;;;;;;;;;;;;;

disp2:  call    sr_tx


        setb    ENA
        mov     r3,#0ffh
        djnz    r3,$
        clr     ENA

        ret

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; program to transfer data serially from I/O

sr_tx:  mov    r1,#08h          ;counter
s1:     mov     a,temp1
        clr     c
        rlc     a
        mov     temp1,a
        mov     Dat,c
        setb    Clk
        nop
        nop
        clr     Clk
        djnz    r1,s1

        ret
;;;;;;;; Delay Sub-Routine   ;;;;;;;;;;;;;;;
; this delay program is a flexible loop , any time delay can
; be adjusted by changing the value of registers

dly:    mov     r5,#0fh
d1:     acall   mdly
        djnz    r5,#d1
        ret

mdly:   mov     r6,#0ffh
mdly1:  call    delay
        djnz    r6,mdly1
        ret

delay:  mov     r7,#0ffh
        djnz    r7,$

  ret
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

end
```

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 02458

I hereby submit my Design Ideas entry.

Name ______________________________

Title ______________________________

Phone ______________________________

E-mail ________________ **Fax** __________

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 text and listings by e-mail to b.travis@cahners.com or send a disk.

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.