Arrange LEDs as seven-segment displays

Charaf Laissoub, Valeo Engine and Electrical Systems, Créteil, France

When you need to drive three seven-segment LED displays, you typically need 10 I/O lines—and that's without a decimal point. You might think that you cannot accomplish that task without a binary-to-seven-segment decoder or a serial-to-parallel shift register (**Reference 1**). Many previous Design Ideas have shown how to maximize the number of LEDs you drive with a minimum number of I/O lines (**references 2** through **5**). This Design Idea shows how you can build a circuit that drives 21 LEDs, thus forming three pseudo-seven-segment displays.

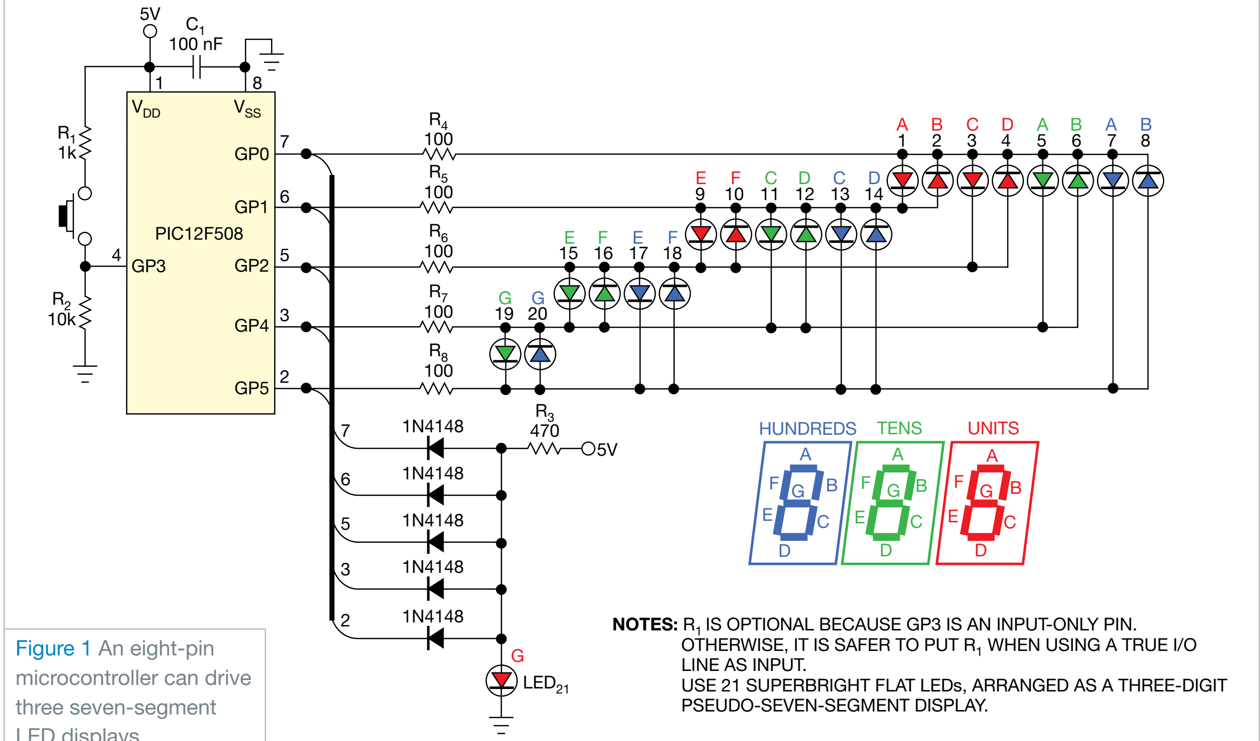

5V
C1 100 nF
R1 1k
R2 10k
PIC12F508
VDD VSS GP0 GP1 GP2 GP3 GP4 GP5
R4 100 R5 100 R6 100 R7 100 R8 100
1N4148
R3 470
5V
LED21
HUNDREDS TENS UNITS
NOTES: R1 IS OPTIONAL BECAUSE GP3 IS AN INPUT-ONLY PIN. OTHERWISE, IT IS SAFER TO PUT R1 WHEN USING A TRUE I/O LINE AS INPUT.
USE 21 SUPERBRIGHT FLAT LEDs, ARRANGED AS A THREE-DIGIT PSEUDO-SEVEN-SEGMENT DISPLAY.

Figure 1 An eight-pin microcontroller can drive three seven-segment LED displays.

The circuit in **Figure 1** modifies the circuit in a previous Design Idea (**Reference 6**). It adds the 21st LED, but it modifies the assembler code to use just 98 words without the main routine. **Listing 1**, the assembler code, is available with the online version of this Design Idea at www.edn.com/110526dia. It can also suit any of a Microchip (www.microchip.com) baseline or midrange PIC microcontroller's eight pins.

THE CIRCUIT ADDS THE 21st LED, BUT IT MODIFIES THE ASSEMBLER CODE TO USE JUST 98 WORDS WITHOUT THE MAIN ROUTINE.

You can adapt this code for another type of microcontroller, such as those from Atmel (www.atmel.com) or STMicroelectronics (www.st.com), using the following steps:

1. Build a look-up table of 10 values for seven-segment coding (see **table** "Code7Segment" in **Listing 1**).
2. Build a look-up table of 3×7 values to store the successive configurations for I/O lines, each configuration containing only one high output and one low output to drive one LED at a time, for each digit (see **table** "Cfg2LinesOut" in **Listing 1**).
3. Build a look-up table of 3×7 values to store the successive high and low state for the I/O lines that are acting as outputs to light only one LED at a time for each digit (see **table** "Light1LED" in **Listing 1**).
4. The subroutine DispDigit rotates to the right seven times, through Carry flag, and the seven-segment code of a digit. It then calls the subroutine LEDon each time you set Carry.
5. The subroutine LEDon activates the LED related to the I/O configuration code, which you can extract from **table** "Cfg2LinesOut," and lights it according to the high or low state code, which you extract from **table** "Light1LED." The subroutine ends by a jump to a critical 1- to 3-msec delay subroutine. Increasing this delay increases the flicker effect, and decreasing this delay dims the LED.
6. Cycle digits of units, tens, and hundreds through steps 4 and 5.

For the PIC10F2xx series, which contains only three I/O lines, **Figure 2** shows an example of driving one digit, and **Listing 2** shows the corresponding assembler code. You can access **Listing 2** from the Web version of this Design Idea at www.edn.com/110526dia. EDN

REFERENCES

1 Anonymous, "Microcontroller provides low-cost analog-to-digital conversion, drives seven-segment displays," *EDN*, May 10, 2007, pg 80, http://bit.ly/hrcp8g.

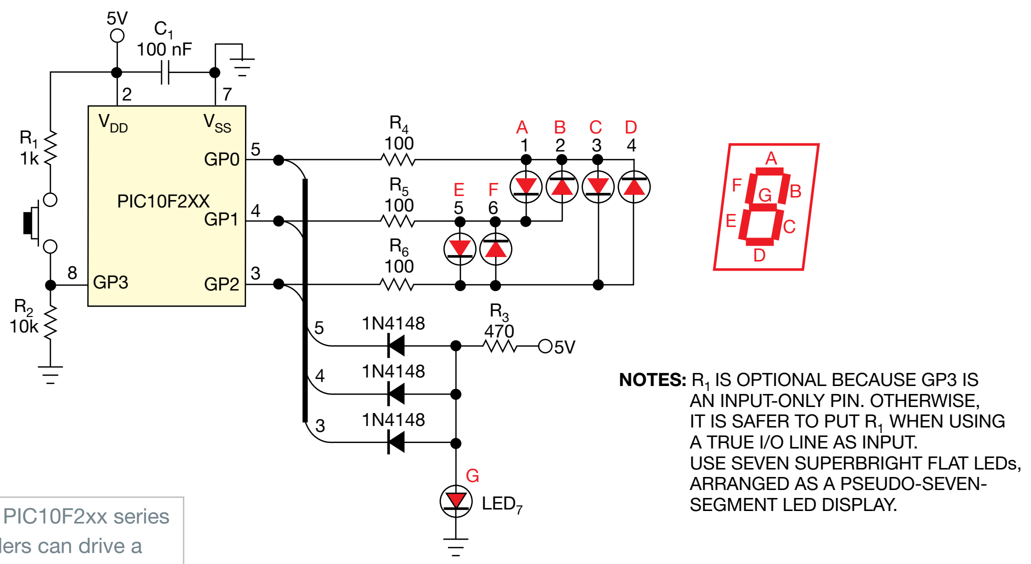
2 Raynus, Abel, "Squeeze extra outputs from a pin-limited microcontroller," *EDN*, Aug 4, 2005, pg 96, http://bit.ly/gX723N.

3 Jayapal, R, PhD, "Microcontroller's single I/O-port line drives a bar-graph display," *EDN*, July 6, 2006, pg 90, http://bit.ly/fjb0MU.

4 Lekic, Nedjeljko, and Zoran Mijanovic, "Three microcontroller ports drive 12 LEDs," *EDN*, Dec 15, 2006, pg 67, http://bit.ly/dRIIBN.

5 Gadre, Dhananjay V, and Anurag Chugh, "Microcontroller drives logarithmic/linear dot/bar 20-LED display," *EDN*, Jan 18, 2007, pg 83, http://bit.ly/hJCs3j.

6 Benabadji, Noureddine, "PIC microprocessor drives 20-LED dot- or bar-graph display," *EDN*, Sept 1, 2006, pg 71, http://bit.ly/g7ZIQY.



NOTES: R1 IS OPTIONAL BECAUSE GP3 IS AN INPUT-ONLY PIN. OTHERWISE, IT IS SAFER TO PUT R1 WHEN USING A TRUE I/O LINE AS INPUT.
USE SEVEN SUPERBRIGHT FLAT LEDs, ARRANGED AS A PSEUDO-SEVEN-SEGMENT LED DISPLAY.

Figure 2 The PIC10F2xx series microcontrollers can drive a seven-segment display with three pins.