BY GAURANG KAVAIYA • MICROCHIP TECHNOLOGY

Don't pay for level translators in systems using multiple power-supply voltages

BECAUSE LOWER-POWER-SUPPLY-VOLTAGE VERSIONS OF ICs BECOME AVAILABLE AT DIFFERENT TIMES, MANY OF TODAY'S SYSTEMS MUST USE SEVERAL VOLTAGES. EVEN SO, YOU CAN OFTEN AVOID THE EXPENSE OF LEVEL TRANSLATORS.

From its early days, the semiconductor industry's mantra has been smaller, better, faster, and cheaper. Today's handheld computing devices are more powerful than the first computers, which required separate buildings. Manufacturers have so far achieved the simultaneous dramatic improvements in cost, performance, and speed by reducing the size of the individual transistors that make up semiconductor devices. This size reduction has had an interesting consequence. As transistors shrink, their operating voltage also shrinks. The most popular power-supply voltage in embedded systems used to be 5V. Now, though, most components in typical embedded systems are moving toward lower supply voltages to take advantage of the industry's newest trends. On the other hand, some system components take longer than others to evolve. Therefore, during the transition to lower voltage, components of a system often require different supply voltages. This situation creates challenges for embedded-system designers. One approach is to use level translators, although they can be costly. This article discusses some low-cost ways to interface a 3.3V microcontroller to a 5V peripheral.

If you are moving your 5V design to 3.3V, the first things to look for are 3.3V versions of the 5V devices. In most cases, you'll find equivalent devices that operate from 3.3V. Usually, 3.3V devices are available at the same—or lower—cost. If you can't find an alternative device that runs at 3.3V, you may need to use two supply voltages. This article's main focus is on designs that use two supply-voltage rails (**Figure 1**).

When you bring together people from different parts of the world, you need to ensure that they can communicate with each other. The same is true when you bring together the 5 and 3V worlds. You must first understand the logic-voltage levels and input/output structure. For inputs, you need to consider high and low input voltage. When you interface a 3.3V system with a 5V device, high input voltage tends to be a bigger problem than low input voltage. Still, you can't ignore the low-input-voltage spec. To ensure proper logic detection, the driving device must produce an output voltage higher than the receiving device's minimum-high-input-voltage spec. However, voltages that are too high aren't good, either.

Almost all CMOS devices have some kind of ESD (electrostatic-discharge) protection on all I/O pins. The most common implementation of this protection uses clamping diodes from the I/O pins to V_{DD} and V_{SS}. Generally, this arrangement translates into a maximum input-voltage spec of $V_{DD}+0.3V$ and a minimum voltage spec of $V_{SS}-0.3V$. If you exceed these specs, the protection diodes may start conducting. If the input terminal has no series resistor, the result can be very high current through these diodes, potentially causing latch-up. You don't want to cre-

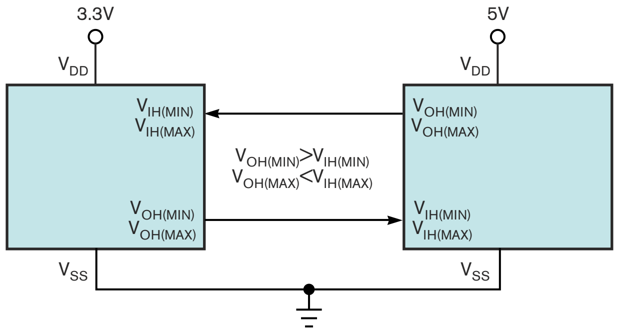

Figure 1 In a two-rail system, devices with different logic levels must communicate.

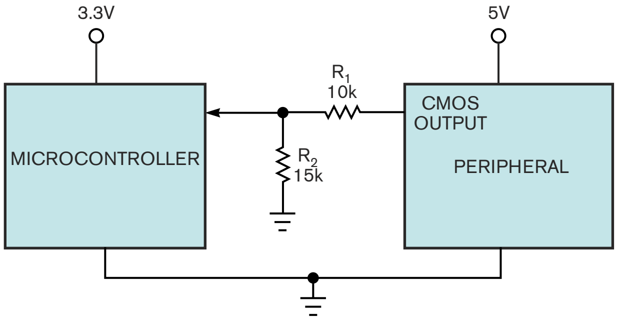

Figure 2 A resistive divider brings a 5V signal in range for a 3.3V input.

ate this condition. If the voltage is high enough—that is, a 5V input in a 3.3V system—you need high series resistance to reduce the clamp current to a safe value. Also, if the resistance is large enough, the low pin and pc-board capacitances may be significant. That is, the RC time constant can cause signal delay. Many manufacturers recommend against the use of ESD-protection diodes for signal clamping. Therefore, a series resistor isn't the best option for feeding a 5V signal to a 3.3V device.

If you look at the logic levels of standard CMOS devices, most of them tend to have minimum high-input voltage of 0.7 or $0.8V_{DD}$. Maximum low-input voltage tends to be around 0.2 or $0.3V_{DD}$. For 5V logic, these levels translate into a high-input voltage of 3.5 or 4V and a maximum low-input voltage of 1 or 1.5V. At lower load currents, most CMOS devices produce output voltages close to the rail with a 0.1 or 0.2V drop. As the load current increases, high output voltage tends to be lower. In this scenario, you must consider the load current to determine the high output voltage.

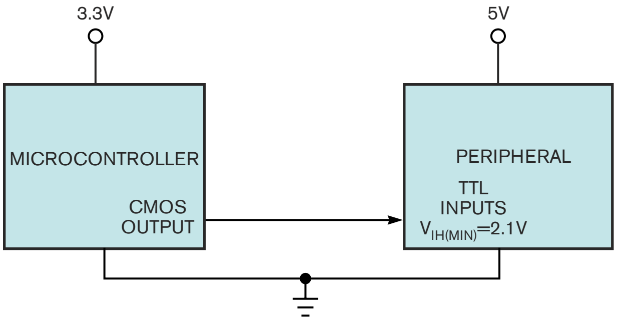

Figure 3 A 5V CMOS device can directly drive TTL inputs.

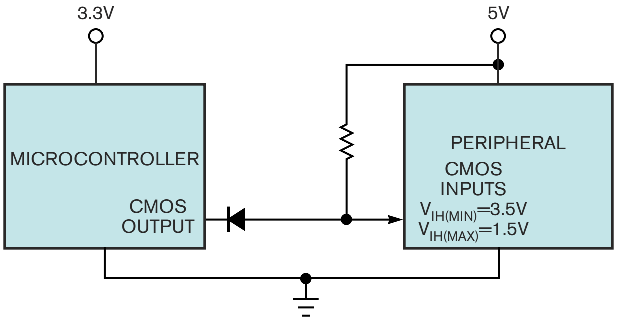

Figure 4 A diode and a resistor bring a 3.3V device output in range for a 5V input.

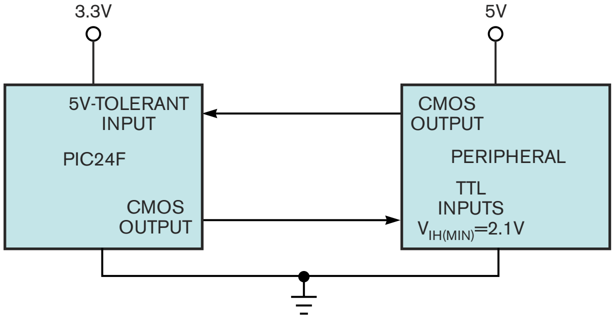

Figure 5 This simple interface works with 5V devices that have CMOS inputs.

RESISTIVE DIVIDER

Compared with a resistor in series with the input, a better option is to use a resistive divider to reduce 5V signals to a range suitable for 3.3V inputs (**Figure 2**). Select the resistor values to account for all tolerances. The following formulas assist in the calculation: With the 5V supply at its minimum value, $(R_2/(R_1+R_2))\times V_{OH(MIN)}>V_{IH(MIN)}$. With the 5V supply at its maximum value, $(R_2/(R_1+R_2))\times V_{OH(MAX)}<V_{IH(MAX)}$. You should also consider resistor tolerances in these calculations.

TTL-INPUT DEVICES

A simpler approach is to use a 5V device with TTL inputs (**Figure 3**). The minimum high-input voltage for a TTL device is 2.1V at drain-to-drain voltage of 5V. Most 3.3V devices can provide a much higher high-output-voltage level, even at high load currents. In this case, swap the peripheral for an equivalent device that has TTL-compatible inputs. If you search carefully, you should be able to find a similar device with TTL inputs. **Table 1** lists some examples. If you are using a standard digital-logic family that must run at 5V, you can find equivalent devices that have TTL inputs. For example, instead of the 74HC family, you can use the 74HCT family. If you need a level translator, use an HCT or a VHCT type of digital buffer. In most situations, this TTL-input-buffer approach is less expensive than the use of dedicated level translators.

DIODE CREATES 0.6V POSITIVE SHIFT

The output high-voltage level of a device operating at 3.3V is slightly lower than the input high-voltage, 0.7V at 3.5V, of a CMOS device operating at 5V. One simple way around this problem is to use a diode to provide the required voltage shift.

The circuit in **Figure 4** shifts the output by approximately 0.6V in the positive direction. Shifting this voltage by 0.6V brings it in range for a 5V CMOS input. The same amount of shift applies to the logic-low signal. Maximum low-voltage input for the CMOS is approximately 1.5V, so the shifted signal does not violate the input low-voltage spec. You need to consider a few things regarding this configuration, though. When the 3.3V device produces a logic-zero level, it increases the current the device draws from the power supply. You should also look at the 3.3V device's low-output-voltage spec at this current. Typically, the higher the sink current, the higher the input low voltage. In this case, you must take care to avoid violating the input-low-voltage spec. If the CMOS low-output voltage is higher, you

TABLE 1 DEVICES WITH CMOS INPUTS AND TTL-INPUT-EQUIVALENT DEVICES

Function	Device with CMOS input	Device with TTL input
16×2-character alphanumeric LCD	Many variants	Devices such as LCM-S01602DTR/M
MOSFET driver	LM5100	LM5101
Vacuum-fluorescent-display driver	Many variants	Devices such as MM58342
CAN (controller-area-network) transceiver	PCA82C50	MCP2551
LIN (local-interconnect-network) transceiver	NCV7380	MCP201

should consider increasing the pullup-resistor value. If the resistor value is too high, the diode bias current will be too low, and the diode may switch too slowly.

5V-TOLERANT INPUTS

Devices such as Microchip Technology's (www.microchip.com) new PIC24 family of 16-bit microcontrollers offer a unique feature to simplify the 5V interface. The devices offer 5V- or 5.5V-tolerant inputs, even when the device operates from a drain-to-drain voltage of 3.3V or less (**Figure 5**). These devices do not use clamp diodes that connect to the drain-to-drain voltage; they use a different mechanism to provide ESD protection. This feature is important for the 5V interface because it allows you to directly connect a 5V output to a 3.3V device without a resistor divider. If you revisit the example of **Figure 3**, you will see that this feature allows for a seamless 5V interface.

Some microcontrollers further enhance this feature by providing the option of generating a 5V output with an external 5V pullup resistor. The 3.3V device drives a 3.3V output, but it can tolerate a 5V input. The digitally controlled open-drain-output capability on these pins allows you to pull this pin to 5V without violating any specs. This feature supports a simple interface to 5V devices with CMOS inputs (**Figure 6**).

CAPACITANCE CAN LIMIT SPEED

When you use a pullup-resistor configuration, you must consider the maximum switching frequency and the capacitance of the connection between the two devices. Doing so helps determine the rise and fall rate of the signal on this port pin and the resistor value that is appropriate for the application. Consider the following equation:

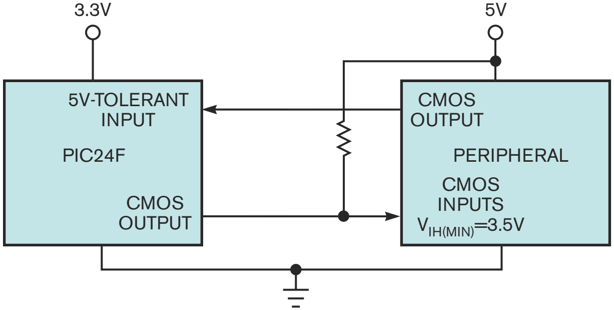

Figure 6 A pullup resistor on an open-drain output generates a 5V signal.

$$\text{RISE/FALL TIME} = \tau \ln\left(\frac{PV_{DD}}{PV_{DD}-PV_{IH(MIN)}}\right),$$

where τ is the RC time constant, R×C, PV_{DD} is the supply voltage of the peripheral, and $PV_{IH(MIN)}$ is the peripheral's minimum input high voltage. If you use a 1-kΩ pullup resistor, a 10-pF parasitic capacitance for the pin and pc board, the 5V supply voltage of the peripheral, and a 3.5V minimum input high voltage for the peripheral, the resulting rise and fall time will be approximately 12 nsec. If the minimum acceptable pulse width for this rise and fall time is 50 nsec, the maximum output frequency is 20 MHz, which is fast enough for most peripheral interactions.

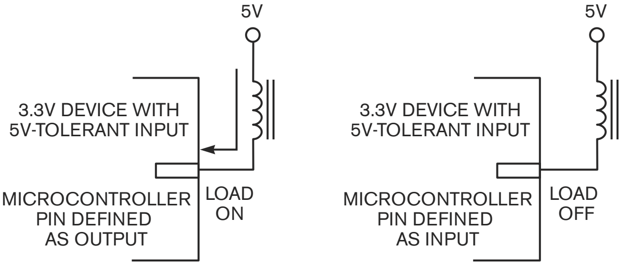

Figure 7 This circuit configuration drives low-impedance loads.

This configuration has a side effect: When the microcontroller drives the logic low, the extra current flows through a pullup resistor. The pullup resistor trades off speed for power-supply current. You need to select a compromise value for your application that provides the required speed and current consumption for the application.

Some may say that you can't use this kind of configuration to drive a low-impedance load. If you want to drive a 5V relay, what should you do? Fortunately, the previously described configuration is also helpful for driving low-impedance loads, such as relays. **Figure 7** shows the circuit-configuration information. To drive the load, define the pin as an output and drive it low. The only limiting factor is the device's current-sinking capability. To turn off the load, define the pin as an input, which turns off the load and applies 5V to the input. The pin tolerates 5V, so this operation is valid. In other words, you need to maintain a logic low on the output latch and toggle the input/output control register to turn the load on and off.

You now have an effective way to bridge 5 and 3.3V power-supply rails. You can devise similar low-cost, intelligent approaches for bridging other supply-voltage combinations during the transition to new, lower voltages. It is likely that most devices will soon move to lower voltages, eliminating the need for bridging. In the meantime, however, these methods should help you to lower your system costs by taking advantage of the newest semiconductor-industry trends.

AUTHOR'S BIOGRAPHY

Gaurang Kavaiya leads the microcontroller-systems applications group supporting most PIC24, PIC18, and PIC16 products at Microchip Technology (Chandler, AZ). He has 10 years of experience in various areas of embedded-system design and has done assembly- and C-language programming and embedded-hardware design. He has also worked on Microchip's nanoWatt Technology power-management products, both as a manager and an engineer.