Application Bulletin 214
May 1993

# Interfacing Sensors with a Microprocessor

This application note addresses a method for monitoring multiple sensor channels with a single analog-to-digital channel. The application might be an eight-channel reflective or a transmissive bar code reader. By using an A/D converter an analog level of the sensor can be measured. This analog voltage can provide more information than a digital sensor.

This system would be controlled by one of the single chip microprocessors like the Motorola 68HC11. Referring to Figure 1 on the next page, let's review the details of operation. The sensors will be configured to operate one channel at a time. This is accomplished by illuminating the LEDs with a microprocessor and connecting the sensors in parallel. The LEDs will be illuminated one by one, and the resulting phototransistor current will be measured by the microprocessor analog-to-digital converter.

This type of operation consumes very little current. Each LED is only on long enough to collect a sensor reading. The microprocessor can also be programmed to have sleep periods and operate only long enough to check all the sensors. The net average current consumption would be less than 25 mA without modulating the CPU on time. This approach is ideal for applications requiring battery operation or those trying to minimize power supply requirements.

When connecting several sensors in parallel as shown, the leakage current of each sensor and its associated solder connections must be very low (200 nA). The ambient light incident on all the sensors must also be low. If these conditions are not met, the leakage currents or off currents (noise) may exceed the on current (signal). Minimizing the off state currents will also produce the greatest dynamic range for the sensing system.

The OPAMP shown is one of a CMOS series produced by National Semiconductor. It is a single supply device with an output that can be driven rail-to-rail. Any OPAMP with equivalent characteristics is acceptable. You may even find it desirable to use one with a balance adjustment which can be used to adjust OPAMP offset and any quiescent noise or leakage out of the sensing system.

Another consideration is the analog-to-digital range and resolution. If the sensor conditions vary, attempt to balance them by adjusting each LED current-limiting resistor. Keep in mind that if you have an eight-bit A/D you will have 256 different possible sensor levels. Make sure the sensor off state falls at the low end of the A/D and the on state falls near the upper end by picking good A/D reference voltages and carefully selecting the proper gain of the final amplifier or buffer. This will probably require prototyping and testing of your circuit to select the proper values.

With regard to LED current, the 7445 shown in Figure 1 is rated for operation with 80 mA of sink current. However, try to limit the LED current to about 40 mA. Normally a lower LED drive current is recommended but this system will be pulsing the LEDs at a low duty cycle, so heating and LED degradation will not be a long term factor in most applications. If you have problems with ambient light, operate the LEDs at a higher current level. Also, if you find you can work with 10 mA or less and you have the extra CPU outputs, you may want to drive the LEDs directly from the CPU and save a chip. Check the specification of your CPU. Most CPUs have a particular eight-bit port which is rated to sink more current.

Timing is another consideration. If the 68HC11 is clocked at 8 MHz, the A/D will complete a single conversion in 32 clock cycles or 16 µs. A few more instructions will be needed to turn on the LED and to allow the signal to stabilize so a channel read time of about 30 µs will be needed. All of these factors can be adjusted and fine tuned for a particular design.

The 68HC11 includes a serial port option. This port can be used to communicate with another computer or CPU for sensor status and control.

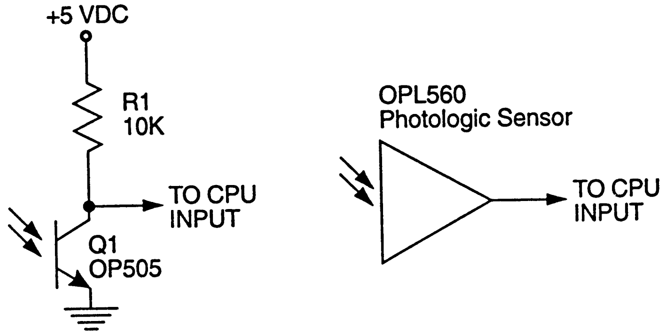

Figure 2

Another application might be to have eight different sensor locations operate as digital inputs. This can be accomplished by using phototransistors and a pull-up resistor or a Photologic sensor as shown in Figure 2. These devices would be connected directly to a CPU input. This approach would be much faster but it will not provide as much information about the state of the sensor channel.

Once you have introduced an CPU into your sensing application, the door is open for all sorts of possibilities. For instance, you may want to dynamically adjust the LED drive current for sensing conditions. Another possibility would be to monitor the signals while watching for a particular sensor signature which signals the occurrence of an event. Finally, you may want to add a temperature sensor to the system and use this data to factor out the temperature effects on the sensor.

**Bob Stricklin**
**Technical Marketing Specialist**

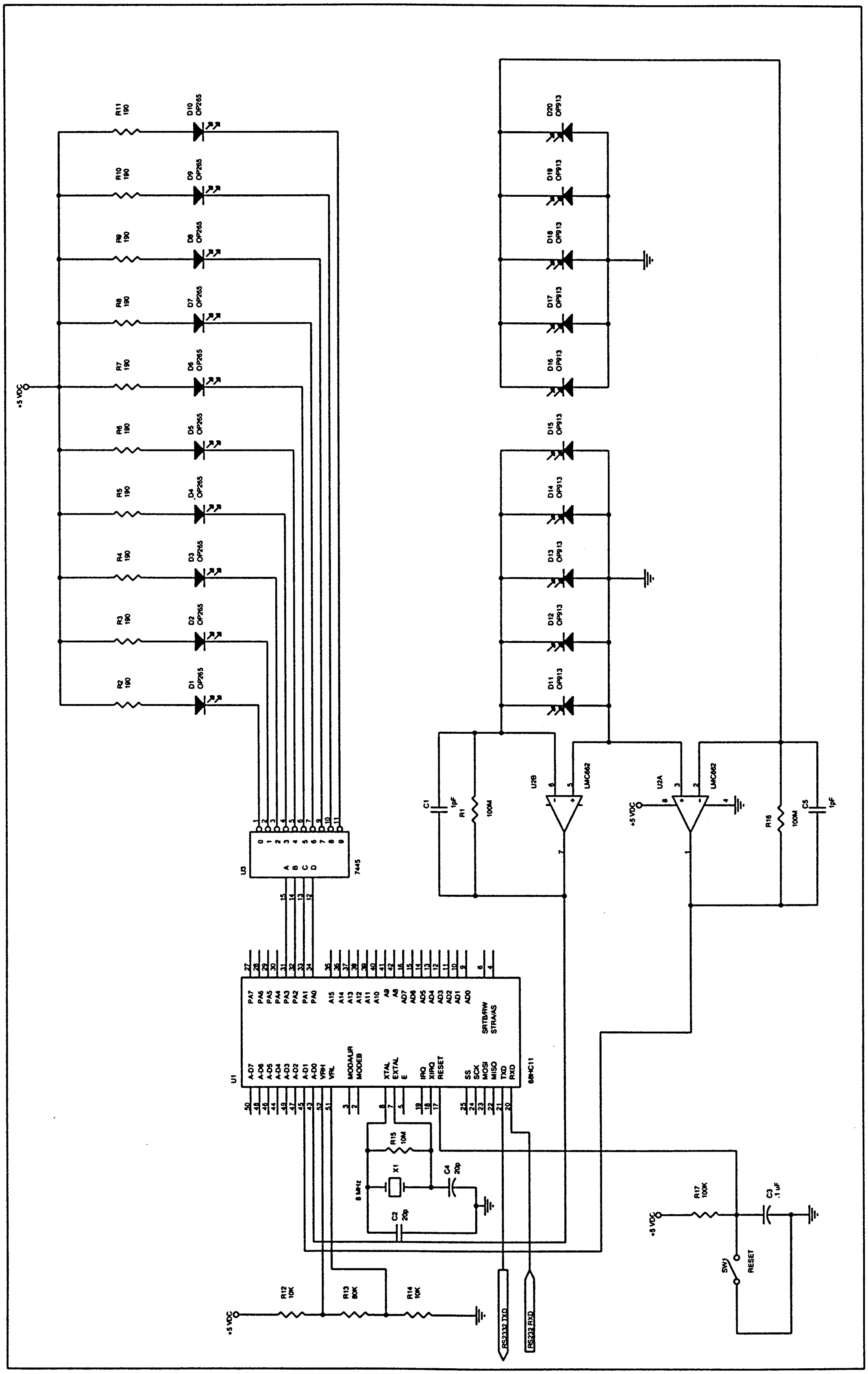
+5 VDC
R2 190
R3 190
R4 190
R5 190
R6 190
R7 190
R8 190
R9 190
R10 190
R11 190
D1 OP265
D2 OP265
D3 OP265
D4 OP265
D5 OP265
D6 OP265
D7 OP265
D8 OP265
D9 OP265
D10 OP265
D11 OP913
D12 OP913
D13 OP913
D14 OP913
D15 OP913
D16 OP913
D17 OP913
D18 OP913
D19 OP913
D20 OP913
U3
7445
U2B
LMC662
U2A
LMC662
C1 1pF
R1 100M
R16 100M
C5 1pF
U1
68HC11
R15 10M
X1
8 MHz
C4 20p
C2 20p
R17 100K
C3 .1 uF
SW1
RESET
R12 10K
R13 80K
R14 10K
RS232 TXD
RS232 RXD

Figure 1