Interface a serial 12-bit ADC to a PC

Davinder Oberoi - July 10, 2003

Over the years, IC manufacturers have devised various ways of effecting interfaces and paying special attention to reducing the number of ICs' interface-I/O pins. The MAX187 is one such device, a 12-bit A/D converter. You can create an interface to this ADC using serial data-communications techniques. Analog-to-digital conversion and data transfer from MAX187 require only three digital-I/O lines. You can create a simple interface between the MAX187 and a PC using the computer's Centronics printer port (Figure 1). You enable or disable the MAX187 by setting the SHDN pin (Pin 3) to high or low level, respectively. If you leave this pin open, then the internal reference (4.096V) becomes disabled, and you must apply an external voltage reference to REF (Pin 4). Otherwise, this pin connects to a 4.7-µF bypass capacitor, C1. The digital data from the MAX187 transfers to the processing unit one bit at a time by using an external clock at SCLK (Pin 8).

A complete data transfer requires 13 external clock pulses. The first clock pulse's falling edge latches the first data bit (the MSB) at the DOUT pin (Pin 6). The output data bit changes at the falling edge of the next external clock, and you can read the serial data bits until the appearance of the falling edge of the next clock cycles. The analog-to-digital conversion starts when the ADC's CS pin (Pin 7) goes low. This pin must remain in the low state until the complete cycle of conversion and subsequent serial-data transfer has taken place. A change of state in the DOUT pin from low to high level indicates the EOC (end-of-conversion) status. Then, serial 12-bit data is available for transfer. Software controls the MAX187's operation. The software should be able to generate all the control signals for successful conversion and also should be able to detect the EOC status. It should also be able to generate 13 external clock pulses to read serial 12-bit data and convert it into parallel data.

The software for the MAX187's operation is in Turbo C++, Version 3.0 (click here to download). In the code, Port defines the Centronics port of the PC to which the MAX187 interfaces. Write Port defines the port for initiating the analog-to-digital conversion and generating the external clock pulses. Read Port defines the port for reading the EOC and serial data from the ADC. After pulling CS and SCLK low, the EOC loop checks for the EOC status. If a valid EOC does not appear, this loop remains operational. As soon as a valid EOC appears, the first of the 13 clock cycles appears, which latches the first data bit (MSB). After this action, the routine calls a subroutine (get_adc()). The subroutine generates the rest of the external clock cycles to read the 12 bits of serial data. The function also converts the received serial data into parallel data (adc_val). It converts by multiplying the previous data by two by shifting adc_val to the left by one bit and adding one to the parallel data if the serial bit's value is one. Once the parallel data is available, the function returns the value and displays it on the screen.