DESIGN NOTES

Number 35 in a series from Linear Technology Corporation June, 1990

12-Bit 8-Channel Data Acquisition System Interfaces to IBM PC Serial Port

Guy Hoover and William Rempfer

IBM PCs Collect Analog Data

IBM PC compatibles can be found just about everywhere. In those instances where a PC is not already in place, battery operated portables are readily available. This makes the PC a good choice for controlling a data acquisition system. Typically, such data acquisition systems have been expensive. Using dedicated A/D cards or IEEE-488 controllers and instruments, these systems tie up slots in the PC and are not readily transportable from one machine to another. As an alternative, the schematic of Figure 1 shows a 12-bit, 8-channel data acquisition system that connects to the serial port of the PC. This system uses an LTC1290, a reference, a handful of other low cost components and requires 12 lines of BASIC to transfer data into the PC. If only ten bits of resolution are required the LTC1290 can be replaced with an LTC1090. Additionally, if the LTC1090 is used, the system can be powered directly from the PC serial port with the option shown.

Two Glue Chips Provide the Interface

The control and status lines of the PC serial port are used to send data to and receive data from the LTC1290. Due to incompatible data formats the Rx and Tx lines are not used. The LTC1290 is a 12-bit, 8-channel data acquisition system on a chip. ACLK of the LTC1290 controls the A/D conversion rate while SCLK controls D_{IN} and D_{OUT} data rates. While $\overline{CS}$ is low D_{IN} is clocked into the LTC1290 and D_{OUT} is clocked out in a synchronous full duplex format. While $\overline{CS}$ is high the conversion requested by the last D_{IN} word is performed.

A simple RC oscillator is used to generate ACLK. The DTR pin of the PC serial port is used to form SCLK. The DTR signal is also fed into the CLR and D inputs of a 74C74 so that on the first falling SCLK the Q output of the 74C74 drives the $\overline{CS}$ of the LTC1290 low. Between data transfers DTR is held high to charge C2 which provides the unregulated V^+ if the RS232

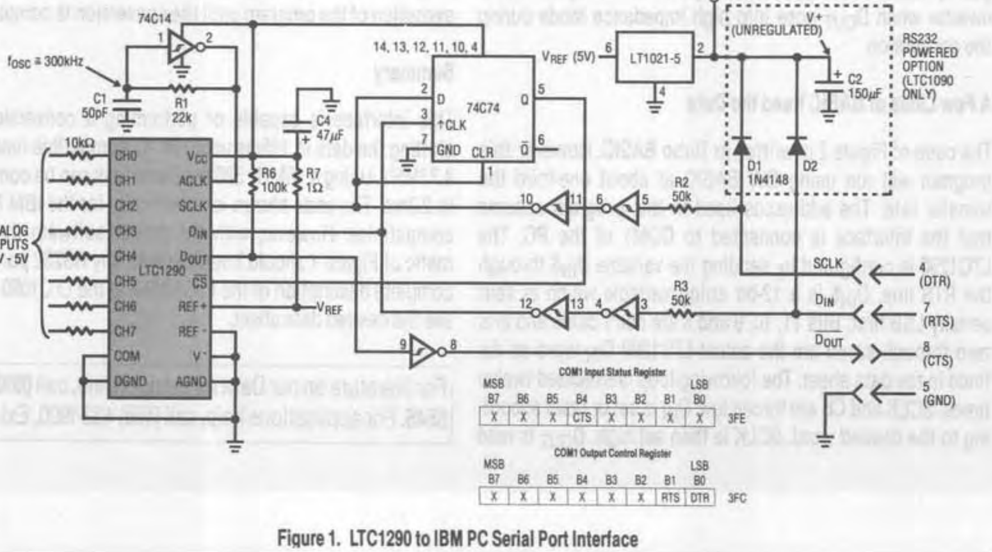

Figure 1. LTC1290 to IBM PC Serial Port Interface

```
10    '        LTC1290 TO RS232 IBM PC TRANSFER PROGRAM
20    '                      BY GUY HOOVER
30    '                 LINEAR TECHNOLOGY CORP
40    '                         1/4/90
50    '  &H3FC IS THE ADDRESS IN HEX OF THE RS232 OUTPUT CONTROL REGISTER
60    '  &H3FE IS THE ADDRESS IN HEX OF THE RS232 INPUT STATUS REGISTER
66    '   "111101110001"  CH0 \
67    '   "111101110011"  CH1  \
68    '   "111101111001"  CH2   \
69    '   "111101111011"  CH3    \ DIN WORDS FOR CH0-CH7 SINGLE ENDED
70    '   "111101110101"  CH4    / UNIPOLAR, MSB FIRST AND 12 BITS
71    '   "111101110111"  CH5   /
72    '   "111101111101"  CH6  /
73    '   "111101111111"  CH7 /
74  DIN$="111101111111"          'DIN$ IS SENT LSB FIRST.
75                               'THE MSB MUST BE A 1 SO THAT DIN IS NORMALLY HIGH
80                               'THIS  DIN  WORD  CONFIGURES  THE LTC1290 FOR CH7
90                               'WITH RESPECT TO COM, UNIPOLAR, MSB FIRST AND
100                              '12 BITS
110  B=2048                      'B IS SCALE FACTOR FOR DOUT. B=512 FOR LTC1090
120 VOUT=0                       'VOUT IS DECIMAL REPRESENTATION OF LTC1290 DOUT
140 FOR I=1 TO 12                'LOOP TWELVE TIMES
145  OUT &H3FC,(&HFE AND INP(&H3FC))                ' SCLK AND CS GO LOW
150                                                 ' DIN IS SHIFTED OUT
160  IF MID$(DIN$,13-I,1) ="0" THEN OUT &H3FC,(&HFD AND INP(&H3FC)) ELSE
     OUT &H3FC,(&H2 OR INP(&H3FC))
180  OUT &H3FC,(&H1 OR INP(&H3FC))                  ' SCLK GOES HIGH
210  IF (INP(&H3FE) AND 16) = 16 THEN D=0 ELSE D=1' READ DOUT
220  VOUT=VOUT+(D*B):B=B/2                          ' SCALE EACH BIT AND SUM BITS
250 NEXT I                                          ' GO THROUGH LOOP AGAIN
260 OUT &H3FC,(&HFD AND INP(&H3FC))                 ' DIN GOES LOW
270 OUT &H3FC,(&H2 OR INP(&H3FC))                   ' DIN AND CS GO HIGH
287  'FOR J=1 TO 20: NEXT J                         ' MAKE CS HIGH FOR 52 ACLKS
```

Figure 2. Turbo BASIC Code for LTC1290 to IBM PC Serial Port Interface

powered option is used. V^+ is fed into the LT1021 reference which provides a regulated +5V for the LTC1290 and the 74C devices. The RTS pin drives the D_{IN} input of the LTC1290 and the CLK input of the 74C74. During a data transfer, RTS (D_{IN}) changes state only when DTR (SCLK) is low so the 74C74 output ($\overline{CS}$) stays low. After the transfer is completed, RTS is toggled while DTR is high causing the Q output ($\overline{CS}$ of the LTC1290) to go high. D_{OUT} of the LTC1290 goes through an inverter which drives the CTS input of the serial interface. The pull-up resistor on D_{OUT} prevents power consumption in the inverter when D_{OUT} goes into high impedance mode during the conversion.

A Few Lines of BASIC Read the Data

The code of Figure 2 is written in Turbo BASIC. However, this program will run using GW BASIC at about one-third the transfer rate. The addresses used in this program assume that the interface is connected to COM1 of the PC. The LTC1290 is configured by sending the variable D_{IN}\$ through the RTS line. D_{IN}\$ is a 12-bit string variable which is sent serially LSB first. Bits 11, 10, 9 and 8 are don't cares and bits zero through seven are the actual LTC1290 D_{IN} word as defined in the data sheet. The following loop is executed twelve times. SCLK and $\overline{CS}$ are forced low. D_{IN} is set or reset according to the desired word. SCLK is then set high. D_{OUT} is read one bit at a time and multiplied by a weighting variable B, to produce a variable that ranges from 0 to 4095 (0 to 1023 for the LTC1090). The variable B is initialized to 2048 (512 for the LTC1090) and divided by two after each bit. The last time through the loop SCLK is high and D_{IN} is cycled low then high. This causes $\overline{CS}$ to return high at which time the requested conversion is performed. $\overline{CS}$ must remain high for 52 ACLK cycles, typically 175μs with the RC oscillator shown. This is not a problem except for the fastest of PCs where a simple FOR...NEXT loop as in line 287 can be used to delay execution of the program until the conversion is complete.

Summary

This interface is capable of performing a conversion and shifting the data in 185ms using an XT compatible running at 4.77MHz. Using a 16MHz 386 the same task can be completed in 2.3ms. The code shown is specifically for the IBM PC and compatibles. However, with the proper software the schematic of Figure 1 should interface with any RS232 port. For a complete description of the LTC1290 and the LTC1090 please see the desired data sheet.

For literature on our Data Acquisition line, call **(800) 637-5545**. For applications help, call (408) 432-1900, Ext. 453.