Serial Code Conversion between BCD and Binary

XAPP 029 October 27, 1997 (Version 1.1)

Application Note by Peter Alfke and Bernie New

Summary

Binary-to-BCD and BCD-to-binary conversions are performed between serial binary values and parallel BCD values.

Xilinx Family

XC3000A / XC3100A

Demonstrates

Serial Arithmetic

Introduction

The FPGA architecture with its powerful function generators evenly interspersed between flip-flops lends itself very well to serial code conversion. Data is entered into a register in one format, and retrieved from the same register in a different format. A common application of this technique is converting binary data to BCD, and BCD to binary.

Operating Description

Binary-to-BCD Conversion

Binary-to-BCD conversion is performed in a modified shift register that successively doubles its BCD contents. As shown in Figure 1, the binary data is shifted into the converter serially, MSB first. Subsequent bits are entered into the shift register to fill the LSB vacated by the doubling. The conversion is complete when all bits of the binary input have been entered, at which time the BCD result is available in parallel form. Each input bit will have been doubled and redoubled to regain its original binary weight, but in BCD format.

To remain a valid BCD number when doubled, a BCD digit of 5 or greater must not just be shifted, but must be converted into the proper BCD representation of its doubled value; along with a 1 being shifted into the next higher digit, a 5 is converted into a 0, a 6 into a 2, a 7 into a 4, an 8 into a 6, and a 9 into an 8.

The binary-to-BCD converter requires three CLBs for each BCD digit in the output, Figure 2. To start a new conversion, $\overline{\text{INIT}}$ should be asserted at the time the binary MSB is applied to the converter input. $\overline{\text{INIT}}$ clears all bits except the LSB which is loaded.

BCD-to-Binary Conversion

BCD-to-binary conversion reverses the process described above, Figure 3. BCD data is parallel loaded into a modified shift register that successively halves its contents. The equivalent binary value is obtained serially, LSB first, from the LSB of the shift register.

To divide by 2, data in the shift register is shifted towards the LSB. However, when a bit shifts across a digit boundary, its weight in the lower digit is 5. This value is added to the shifted digit using carry-save adders associated with bits 0 and 2. The conversion is complete when all bits of the binary output have been generated.

The BCD-to-binary converter requires three CLBs per digit, Figure 4. A new conversion is started by applying the BCD data and asserting the $\overline{\text{LD}}$ control to load the data. The MSB of each digit is loaded into the carry flip-flop of the bit-2 adder; the carry of the bit-0 adder is cleared.

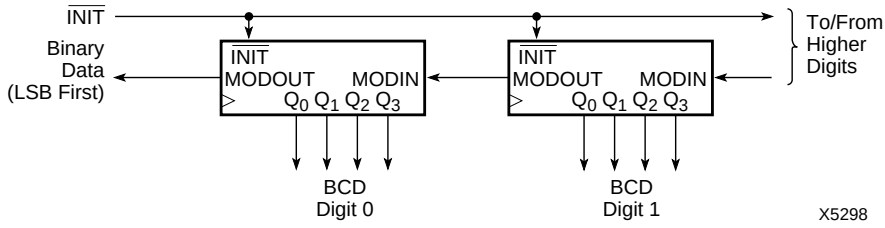

Figure 1: Binary-to-BCD Converter

Note: Supporting design files are available on the Xilinx AppLINX CD-ROM and on the Xilinx WebLINX web site under the names XAPP029V (VIEWlogic) and XAPP029O (OrCAD).

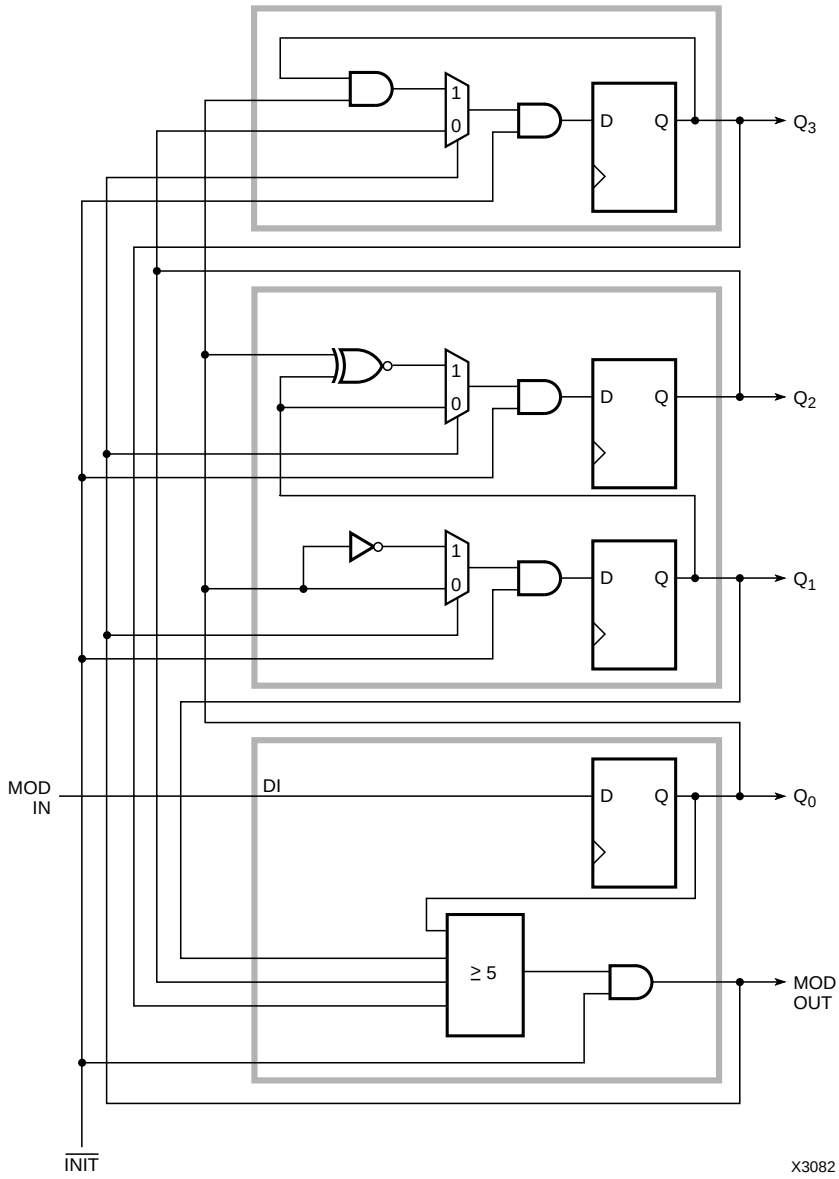

Figure 2: Binary-to-BCD Converter (Three CLBs per BCD Digit)

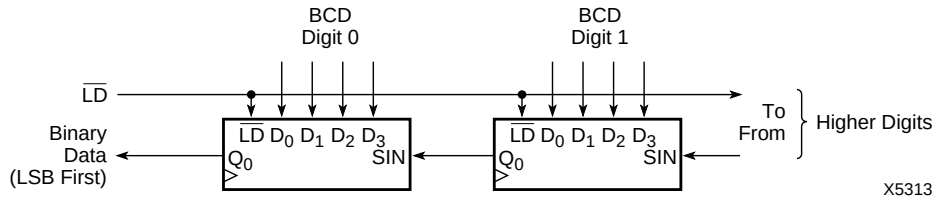

Figure 3: BCD-to-Binary Converter

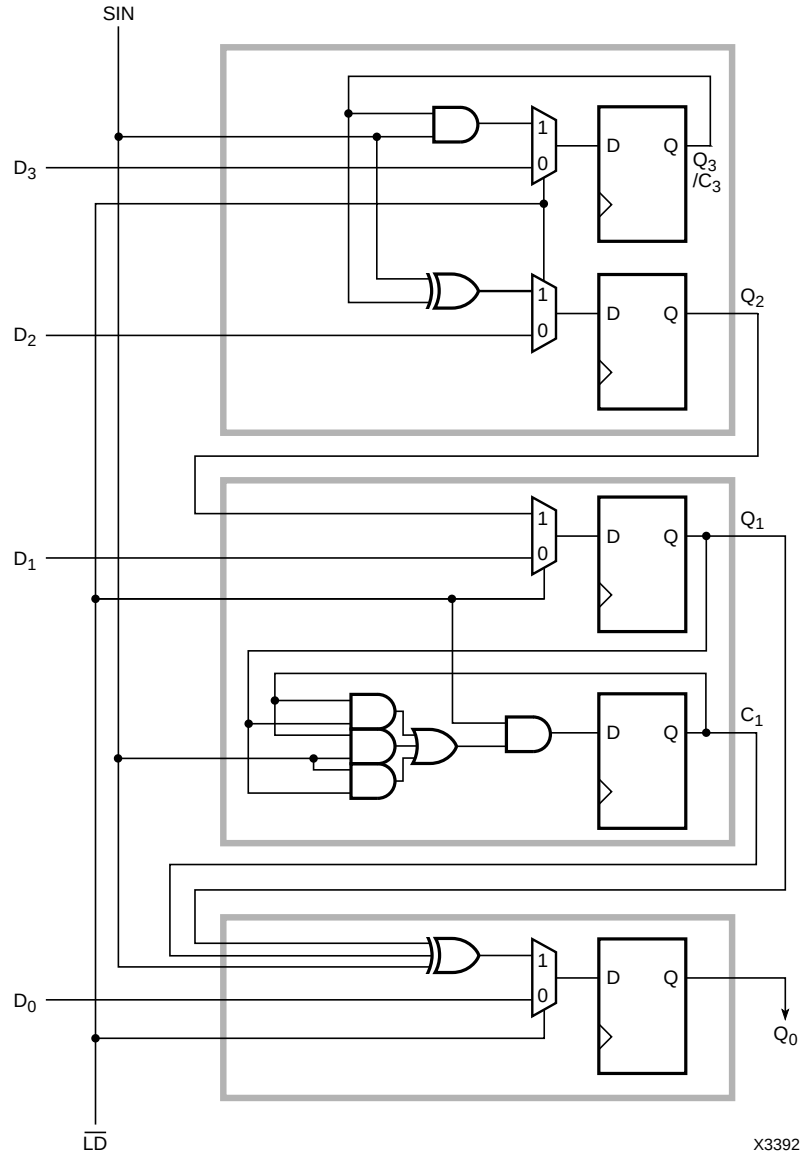

Figure 4: BCD-to-Binary Converter (Three CLBs per BCD Digit)