COE 200, Term 042

Fundamentals of Computer Engineering

HW# 7

Q.1. Design a Combinational circuit that accepts a 3-bit number and generates as output a binary number equal to the square of the input number.

Q.2. Design a combinational circuit that detects an error in the representation of a decimal digit in BCD. In other words, obtain a logic diagram whose output is equal to 1 when the inputs contain any one of the six unused bit combinations in the BCD code.

Q.3. Design a combinational circuit that accepts a 4-bit number and generates a 3-bit binary number output that approximates the square root of the number. For example, if the square root is 3.5 or larger, give a result of 4. If the square root is < 3.5 and >= 2.5, give a result of 3.

Q.4. Design two simplified combinational circuits that generate the 9's complement of (a) a BCD digit and (b) an excess-3 digit. Compare the gate and literal counts of the two circuits. Assume in both cases that input combinations not corresponding to decimal digits give don't care outputs.

Q.5. Design a combinational circuit that receives a 4-bit signed number in 2's complement representation and returns the absolute value of the number i.e., the output returned should be 3-bit.

Q.6. Assume that you have a 4-bit binary adder. Use this adder to design a circuit that decrements a 4-bit number by 1.

Q.7. It is required to design a 4-bit ripple-borrow subtractor to find the subtraction X-Y for the two unsigned numbers, $X=X_3\text{-}X_0$, and $Y=Y_3\text{-}Y_0$. Design a 1-bit full subtractor and show how it can be used to construct the 4-bit subtractor.

Q.8. It is required to design a combinational circuit that compares two n-bit numbers to see if they are equal or not. Design a circuit that has three inputs and one output that can be used for each of the n bits, such that the circuit is connected in cascade by carry-like signals. One of the inputs to each cell is a carry input, and the single output is a carry output.