EE5423 Computer Arithmetic (Spring 2016)
Project #2 – Due 3/22/2016

Goal: Compare performance of an unsigned shift-and-add multiplier with one using a shift-over technique.

1. Write a program to simulate the process of an n-bit ($n \times n$) shift-and-add **synchronous** multiplier.

   - Use an n-bit RCA for the adding process.
   - Assume that the process of a one-bit shift, a multiplexing, and a unit 2-level logic in the RCA, each takes a delay of $2d$.
   - Assume that it is synchronized with a clock of a period exactly long enough for all the delays in one shift-and-add cycle.
   - Note: in this part, since it is a synchronous process, there is no need to simulate the RCA process, but instead to simply account for its worst-case delay.

2. Write a program to simulate the process of an n-bit ($n \times n$) shift-and-add **asynchronous** multiplier with shift over 0's/1's.

   - Fast shift-over-0's/1's is taken, whenever applicable, without the adding process. That is, a shift-over situation does not incur an adding process. Assume a delay of $2d$ for such a case.
   - Use a CCA (carry completion adder) of sufficient size for the adder/subtractor to detect the completion of the adding/subtracting process so as to initiate the next shift-and-add cycle. Note: you can use either your design from Project#1 or the code provided on the class website.
   - Assume that the process of a one-bit shift, a multiplexing, and a unit 2-level logic in the CCA, each takes a delay of $2d$.
   - Assume that there is another delay of $2d$ from the necessary logic for the asynchronous process to trigger the next cycle.

3. Results required include:

   - the program listings for both techniques,
   - a plot showing delays of the technique #1 versus n, ($\forall n, 2 \leq n \leq 32$, without using simulation, (Note: Use a log scale for y-axis.)
   - From a simulation run of 10,000 times on the technique (2), a plot showing its performance compared to the results from (1); also a table to show all the corresponding numbers,
   - a demonstration of step-by-step (in delay of $2d$) intermediate results for the second technique for an 12-bit multiplication of x4b7 and x6c9. Clearly indicate what exactly transpires in each $2d$ delay.