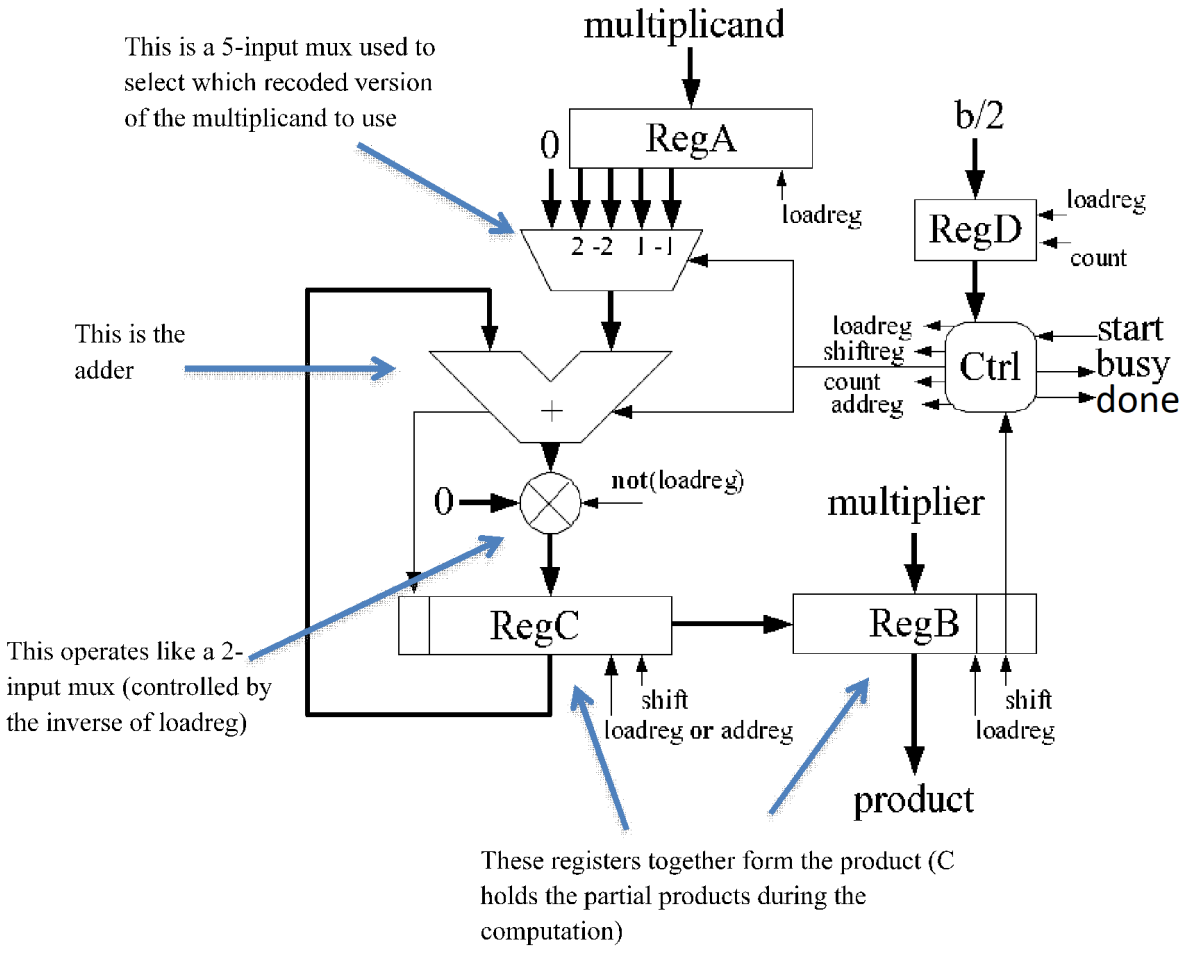
ECE3270 Digital System Design
Lab 5: Bit-pair Recoded Multiplier

Lab Overview: The purpose of this lab is to implement a Bit-pair Recoded fixed point multiplier in VHDL. The top level entity should be a structural design, and all lower level components should be kept completely **modular** and **generic**. The multiplier should be tested in ModelSim with appropriate testbench(es) that test for all cases (+/- and varying ranges). You will map your top level entity to the board which will use a shortened version of the multiplier. You will complete a full report using the LaTeX template and include discussion about the clock cycle latency of one multiplication. You will submit all edited files to the assign server as Assignment 5.

Part I

You are to design an 8-bit shift-add multiplier using bit-pair recoding as discussed in class. Each component in the design should be a separate module that you will connect in your top-level structural entity. Part 2 will require that you set the design 4-bits, so a **modular** and **generic** design will be **required**.
Model your design based on the datapath below. Processing begins when the *start* signal is high and the *busy* signal is low (note, *busy* and *done* are set by the state machine, **Ctrl**). The following cycle loads the *multiplicand* and the *multiplier* into their corresponding registers and the *busy* signal becomes high (multiplication is in progress). The machine will complete add/shifts until the product is complete, at which point the *done* signal becomes high for **1 clock cycle**, guaranteeing valid output. Then the *busy* signal becomes low, after which the machine can begin again (whenever *start* is high as well). You will need to include a copy of the state machine diagram generated by Quartus in your report.



Part II

You are to map an instance of your component for use on the board. As you only have 10 input switches, you will use a data width of 4, leaving you space for other input signals. You may choose where to map your signals, as long as your output is displayed on the LEDs.

Rubric	
Report - Proper format - All sections included - Valid images where applicable - Proper grammar, punctuation, and spelling	**50%** - 10% - 30% - 5% - 5%
Demo - Live Demonstration o Includes working code and answering questions from the TA - Proper modular design o Thoughtful I/O and Signal names included - Comments o Thoughtful comments, not English translations of code	**40%** - 30% - 5% - 5%
Proper Assign Server Code Submission	**10%**