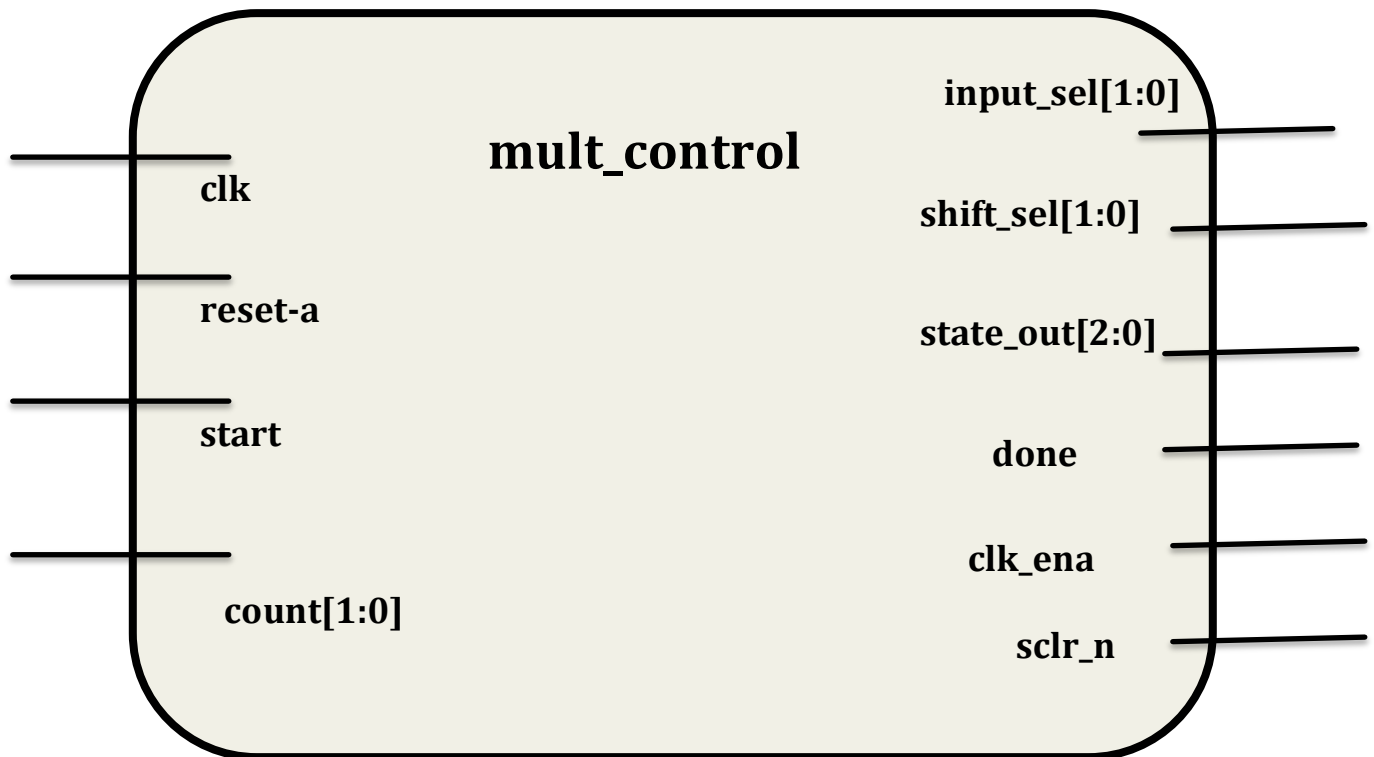


**ECE 176 – Computer Aided Engineering in Digital Design
Fall 2014**

**Examine State a Machine Implementation
Assignment 9**

Objectives:

Examine a State Machine Implementation



You have now completed building all of the components necessary to build the 8x8 multiplier, except for the controlling state machine. Due to time, the controlling state machine has been written for you and is located in Blackboard. The name of the file is `mult_control.v`. You may open the file in the Quartus II text editor for review.

This state machine will manage all the operations that occur within the 8x8 multiplier using 6 defined states:

idle, lsb, mid, msb, calc_done and **err**. See the state diagram in next page for more definition of its behavior.

The state machine in the **LSB** state multiplies the lowest 4 bits of the two 8-bit multiplicands, $((a[3..0] * b[3..0]) * 2^0)$. This intermediate result is saved in an accumulator.

The state machine in the **MID** state performs cross multiplication $((a[3..0] * b[7..4]) * 2^4)$ and $((a[7..4] * b[3..0]) * 2^4)$. This is done in successive clock cycles. The products of both multiply operations are added to the content of the accumulator as they are completed and clocked back into the accumulator.

The state machine in the **MSB** state multiplies the highest 4 bits of the two 8-bit multiplicands $((a[7..4] * b[7..4]) * 2^8)$. This product is added with the content of the accumulator and clocked back into the accumulator.

This result is the final product:

$$\begin{aligned} \text{result}[15..0] &= a[7..0] * b[7..0] \\ &= ((a[7..4] * b[7..4]) * 2^8) \\ &\quad + ((a[7..4] * b[3..0]) * 2^4) \\ &\quad + ((a[3..0] * b[7..4]) * 2^4) \\ &\quad + ((a[3..0] * b[3..0]) * 2^0) \end{aligned}$$

The state machine in the **CALC_DONE** state asserts the **done_flag** output to indicate the final product has been calculated and is ready for reading by downstream logic.

The state machine in the **ERR** state indicates incorrect inputs have been received.

There are two inputs to the state machine: **start** and **count**. The **start** signal is asserted for once clock cycle to begin an 8x8 multiply operation on the next clock cycle. The **start** signal must only be asserted for one clock cycle. The **count** signal is used by the state machine to track the multiplication cycles.

The outputs of **mult_control** control the various other blocks in the design.

mult_control state machine state diagram

