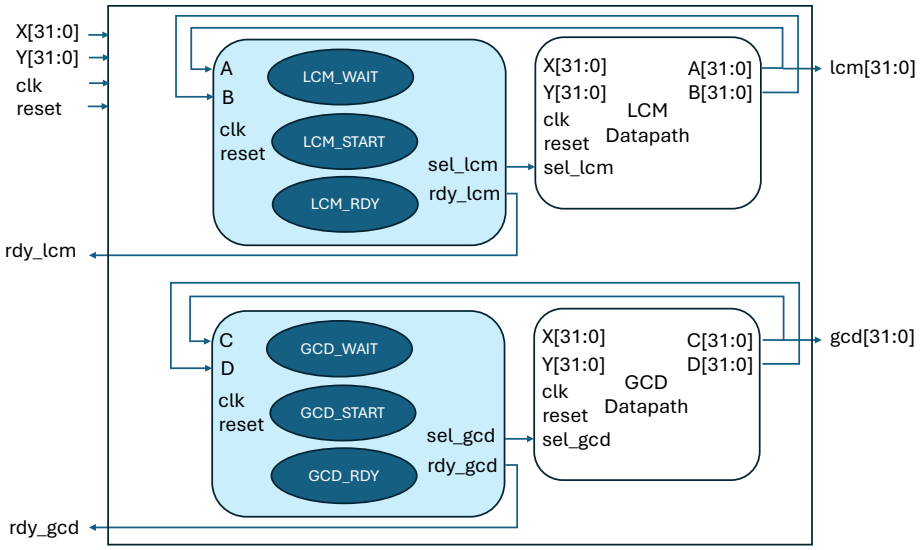
ECE 3401 Digital Systems Design – Spring 2025

Programming Assignment 2:
State Machines for LCM and GCD Digital Design

Due March 10, 2025 (Friday) @ 11:59 PM on HuskyCT

This programming assignment extends the programming assignment 1 with state machine designs for controlling the LCM and GCD datapaths. The figure below shows the module level diagram for the design. The LCM and GCD datapaths should adopt your working PA1 codes.



For the LCM datapath control, the state machine uses the LCM_WAIT, LCM_START and LCM_RDY states. The A, B register outputs are also available for the state machine to control this datapath. When asynchronous reset is asserted the state machine must go to LCM_WAIT state. After reset is de-asserted, the LCM datapath sequences to compute its output and the state machine must be in LCM_START state in this duration. However, when LCM computation completes, the state machine must go to LCM_RDY, and stay in this state until reset is asserted again. The LCM datapath control must derive the sel_lcm logic using the states and the inputs A and B. When the LCM output is ready, it must be available on the lcm output, and the rdy_lcm flag must be asserted.

For the GCD datapath control, the state machine uses the GCD_WAIT, GCD_START and GCD_RDY states. The C, D register outputs are also available for the state machine to control this datapath. When asynchronous reset is asserted the state machine must go to GCD_WAIT state. After reset is de-asserted, the GCD datapath sequences to compute its output and the state machine must be in GCD_START state in this duration. However, when GCD computation completes, the state machine must go to GCD_RDY, and stay in this state until reset is asserted again. The GCD datapath control must derive the sel_gcd logic using the states and the inputs C and D. When the GCD output is ready, it must be available on the gcd output, and the rdy_gcd flag must be asserted.

You will use separate VHDL modules for the register bank (dff.vhd), the ALU (alu.vhd), and the overall PA2 module (pa2.vhd). You are given top-level modules, and you are expected to write the architecture for each module. You will be graded on the design of these modules and their functionality.

DESIGN VERIFICATION

We have provided a fully functional testbench1.vhd that computes the LCM and GCD of several two number tuples. The reset mechanism ensures that the state machines will be cleared after each tuple finishes its computations by tracking the rdy_lcm and rdy_gcd flags. The report statements are provided to display the design outputs for each tuple computation.

DELIVERABLES

Please submit the following report saved as a single PDF:

1. Your code for each module. Make sure to submit the following VHDL files for your submission.
   alu.vhd
   dff.vhd
   pa2.vhd

2. Submit screenshots of the following: Your output waveform from 0ns to 700ns. The clk, reset, X, Y, rdy_lcm, lcm, rdy_gcd, and gcd state should all be clearly visible. The data format should also be unsigned decimal radix for all of these signals. Format all the waveforms as described.

3. In case your design is not fully functional, you will need to schedule a code review with the TA.