1. (10 pts) On the diagram below, complete the timing diagram for the Y output for all clock cycles.
2. (15 pts) On the waveforms below, complete the waveforms for State, ld, en, and Q. The FSM is controlling an UP counter.
3. (20 pts) For the figure below:

a. Give the maximum register-to-register delay. Show your work.
   \[ Tcq + \text{mult delay} + \text{add delay} + \text{tsu} = 3 + 18 + 7 + 1 = 29 \text{ ns} \]

b. Modify the diagram to add one level of pipelining but still maintain the same functionality. Add
   the pipeline stage in the place that will improve the register-to-register delay the most. Compute
   the new maximum register-to-register. Assume that adding a pipeline registers to any functional
   unit (adder or multiplier) breaks the combinational delay path in the unit exactly in half.
   \[ Tcq + \frac{1}{2} \text{ mult delay} + \text{add delay} + \text{tsu} = 3 + 9 + 7 + 1 = 19 \text{ ns} \]

c. With the pipeline stage added, complete the ‘Q’ waveform shown below. Input registers
   change values as shown, assume Reg Q is loaded every clock cycle. All waveforms represent
   register outputs.
4. (15 pts) For the figure below:

a. Compute the maximum setup time on pin A
   \[ \text{Buff delay} + 2 \times \text{(gate delays)} + \text{Tsu} - \text{clk buff delay} = 1.5 + 2(1.0) + 1.5 - 2.0 = 3 \text{ ns} \]

b. Compute the minimum hold time on pin A.
   \[ \text{Clk buff delay} + \text{Thd} - \text{min path A (buff delay)} = 2 + 1 - 1.5 = 1.5 \text{ ns} \]

All times in ns.
DFF timings: C2Q = 0.5, Tsu = 1.5, Thd = 1.0
Non-clk I/O Buffer delay: 1.5
Clk Buffer delay = 2.0
All other gate delays: 1.0
5. (10 pts) Draw the GATE LEVEL logic generated for the VHDL code shown below (A, B, Y are all single-bit signals). Your schematic must be composed of GATES (i.e, nands, nors, ands, ors, xors, etc…).

\[
Y \leq A \text{ when } (S = '1') \text{ else } B;
\]

![Diagram of logic circuit](attachment:image.png)

6. (10 pts) We looked a couple of SRAM based FPGAs from Xilinx and Altera. What was the basic mechanism for implementing a combinational logic function? Give one other feature/function that was included in both of the basic cells from Xilinx and Altera.

Both Xilinx and Altera use 4-input Look Up tables (LUTs). Both basic cells had a DFFs, also had fast carry logic.

7. (10 pts) Name two functions of a PLL (Phase Locked Loop)

*Internal/External Clock synchronization (eliminates skew from I/O buffer delay)*
*Clock multiplication*