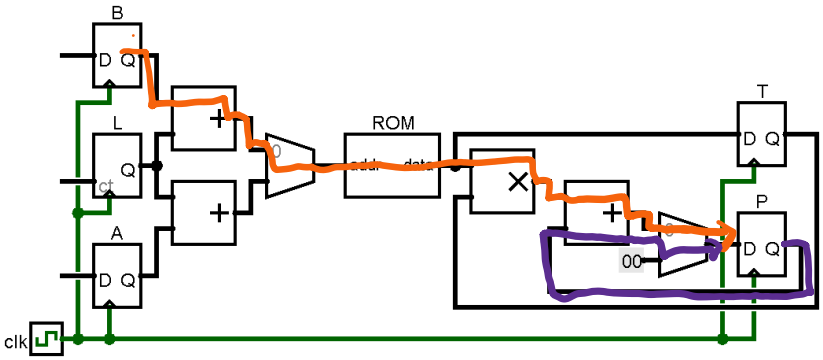
Design of Digital Circuits and Systems, Quiz 5

Timing: STA, Pipelining, CDC

Solution Outlines

Static Timing Analysis:



The longest (**orange**) and shortest (**purple**) data paths are shown on the circuit diagram above. The clock network delays are ordered $t_{clkB} > t_{clkL} > t_{clkA}$, which is why the longest path starts at B instead of L or A. The longest and shortest paths both have register P as their destination register for the clock paths.

setup slack = min clock path – max data path

$$= (T + t_{\text{clkP,min}} - t_{\text{su,max}}) - (t_{\text{clkB,max}} + t_{\text{CO,max}} + 7 \cdot t_{\text{wire,max}} + 2 \cdot t_{+,\text{max}} + 2 \cdot t_{\text{MUX,max}} + t_{\text{ROM,max}} + t_{\times,\text{max}})$$

$$= (175 + 2 - 17) - (7 + 12 + 7 \cdot 0 + 2 \cdot 25 + 2 \cdot 15 + 19 + 50) = \textbf{– 8 ns}$$

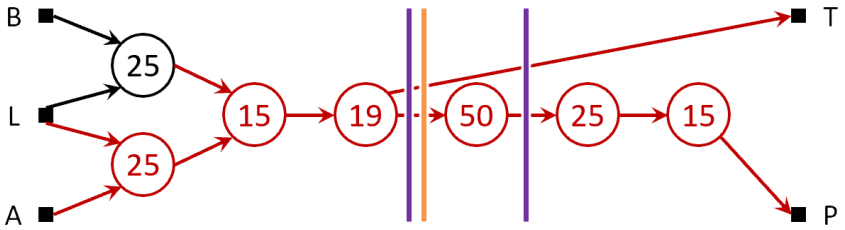
hold slack = min data path – max clock path

$$= (t_{\text{clkP,min}} + t_{\text{CO,min}} + 3 \cdot t_{\text{wire,min}} + t_{+,\text{min}} + t_{\text{MUX,min}}) - (0 + t_{\text{clkP,max}} + t_{\text{h,max}})$$

$$= (2 + 9 + 3 \cdot 0 + 20 + 11) - (0 + 6 + 8) = \textbf{28 ns}$$

Pipelining:

The DFG and optimal cutsets (for **2-stage** and **3-stage** pipelines) look like:

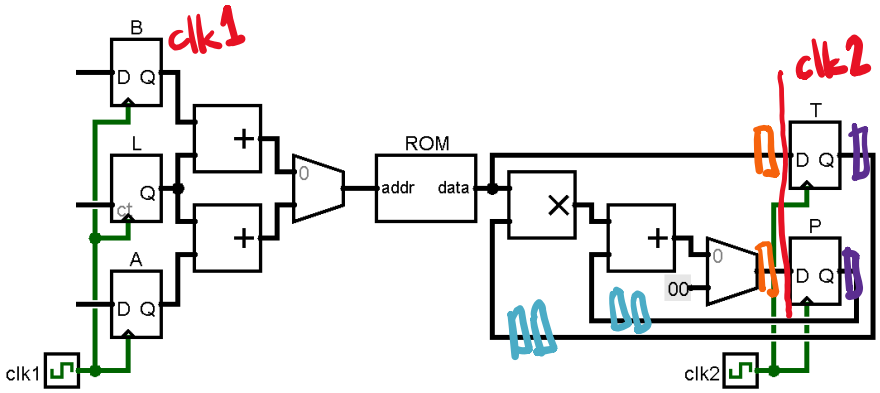


The key is to try to balance the stages: the total delay of just combinational components is 149 ns.

The 2-stage critical path gives us a T_{min} of $t_{\text{CO}} + t_{\times} + t_{+} + t_{\text{MUX}} = 12 + 50 + 25 + 15 =$ **102 ns**. Since it takes 2 clock cycles for the inputs to reach the output, latency = $2 \cdot T_{\text{min}} + t_{\text{CO}} =$ **216 ns**.

The 3-stage critical path gives us a T_{min} of $t_{\text{CO}} + t_{+} + t_{\text{MUX}} + t_{\text{ROM}} = 12 + 25 + 15 + 19 =$ **71 ns**. Since it takes 3 clock cycles for the inputs to reach the output, latency = $3 \cdot T_{\text{min}} + t_{\text{CO}} =$ **225 ns**.

Clock Domain Crossing:



The keys to this question are (1) a clock domain is defined by *sequential* elements (*i.e.,* registers, flip-flops), and (2) a clock domain crossing is defined by a path that has source and destination registers controlled by *different* clocks. So even though it might appear that the feedback paths cross back from clock domain 2 to 1, the destination register in both cases is register P, meaning that these don't create clock domain crossings!

For **registered crossings**, we need to add the **two registers** shown above in **orange** (both connected to clk1). The crossings from clk1 to clk2 (*i.e.*, from the outputs of registers B, L, and A to the inputs of registers T and P) all come from combinational logic elements – the ROM and rightmost MUX – so they should be registered.

For **2-FF synchronizers**, we need to add the **two registers** shown above in **purple**. The crossings from clk1 to clk2 already to go registers in the receiving domain, so only one additional register attached to clk2 needs to be added. Since the problem said to ignore part A, there are no crossings from clk2 to clk1.

- *If* the circuit also had the registered crossings attached to clk1 (**orange**), then this would make the feedback paths clock domain crossings from clk2 to clk1, in which case they would each need two registers added attached to clk1 (shown in **aqua**), bringing the total up to six registers.