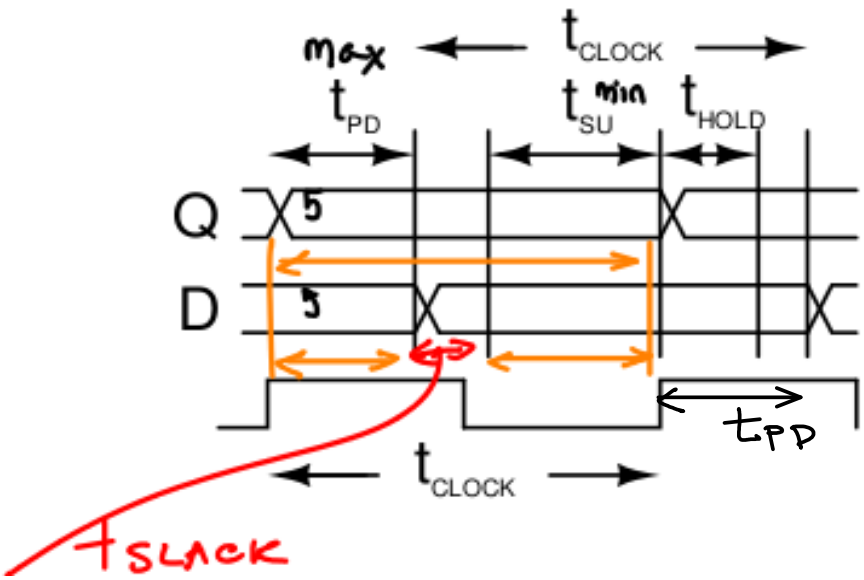
RTL Design

Exercise 1: Mark the slack time in the diagram above.



Exercise 2: If the clock period (t_{CLOCK}) is known, how are the minimum setup (t_{SU}) and hold (t_{HOLD}) times related to the minimum and maximum propagation delays (t_{PD})?

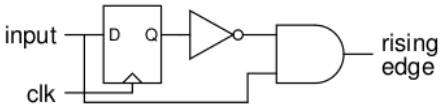
$$t_{clock} = t_{su}(min) + t_{slack} + t_{PD}(max)$$

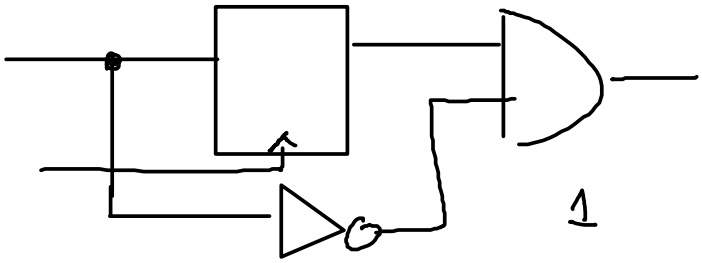
Exercise 3: Which of these requires the most time and effort? Least? Which gives the designer most control over the cost and performance of the design? Least? Which produce(s) designs that are portable to different implementation technologies (FPGAs, ASICs)? Which allow the same design to meet a variety of speed/area targets?

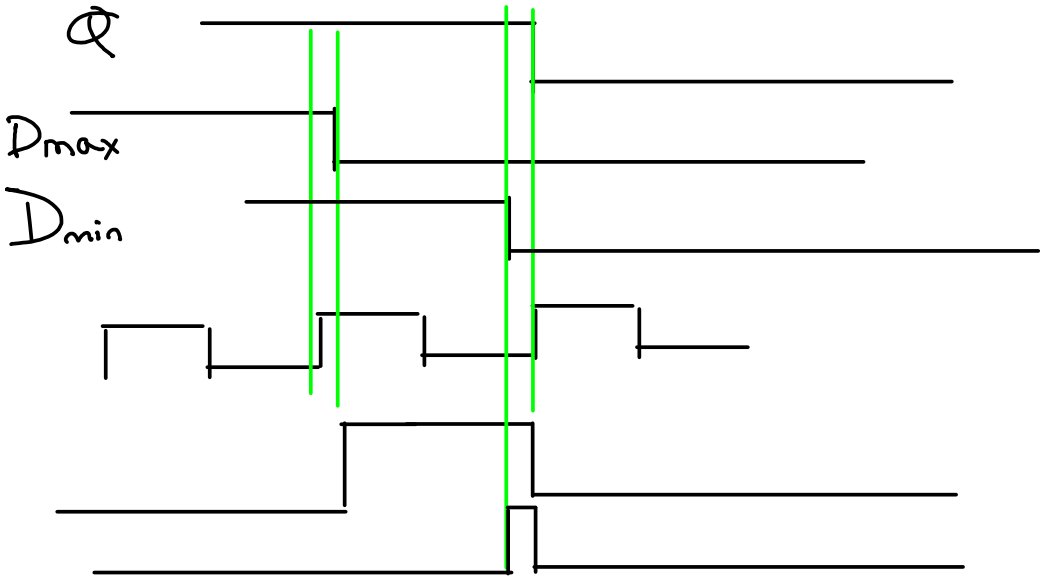
	Time & Effort:	Control	Portability & Flexibility	
most:	structural	structural	HLS	
least:	HLS	HLS	structural	

Exercise 4: How would you design a falling-edge detector? For how many clock cycles is the detector output true?

falling edge detector

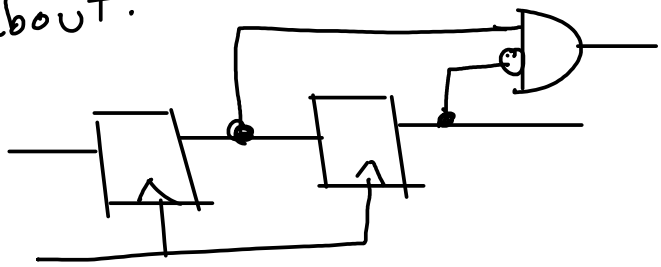


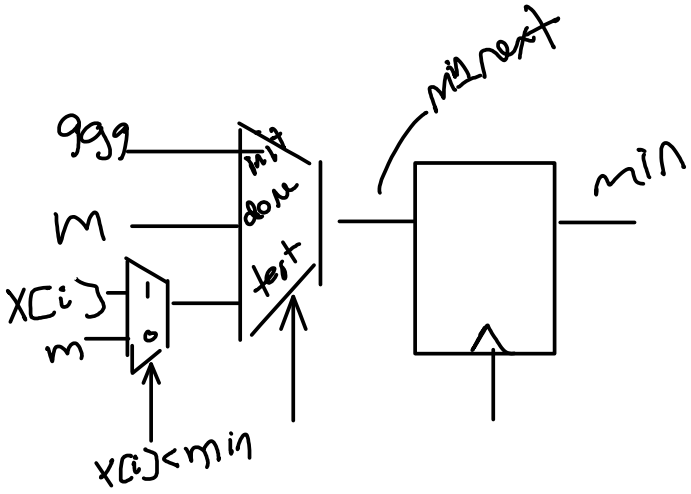


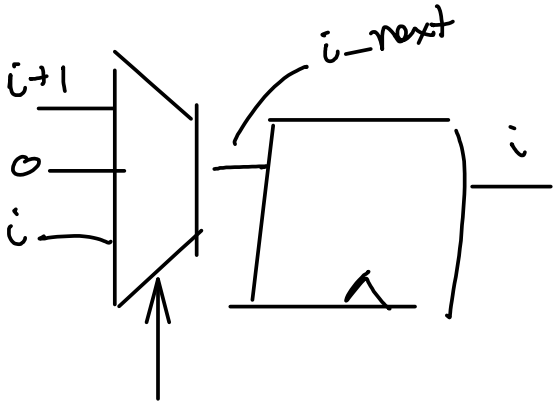


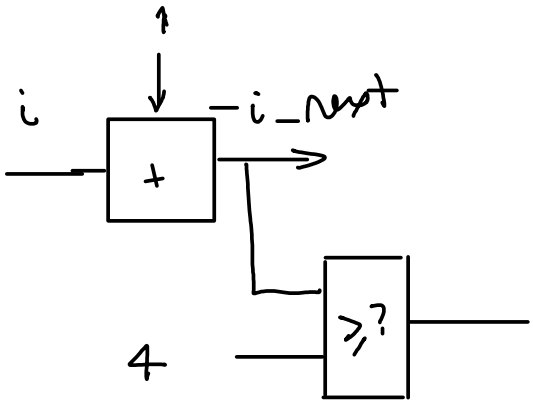
o/p true for 0 → 1 clock cycle

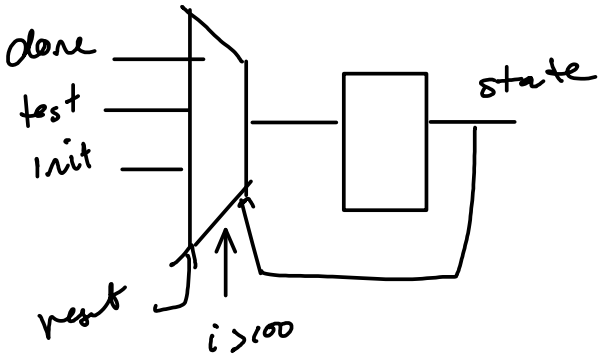
what about:



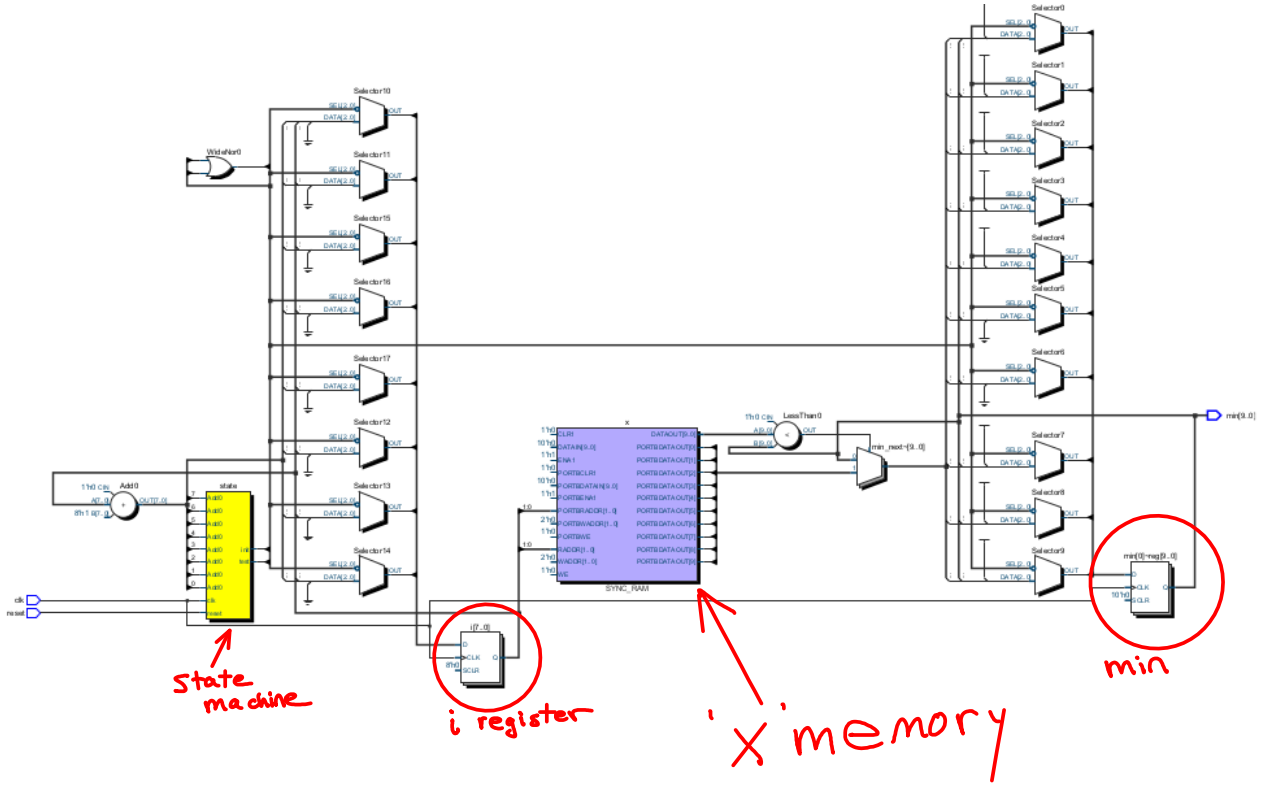

min_next
ggg
init
M
done
test
X[i]
1
0
m
min
X[i]<min


i_next
i+1
0
i
i


1
i
i_next
+
4
>=?


done
test
init
state
reset
i>100

Exercise 5: Find the following blocks in the schematic: the min register, the x[] memory block, the i register, the state state machine.



Exercise 6: Each number in the Fibonacci sequence is the sum of the previous two. Write an algorithm to compute the values of the sequence that are less than 100 and store them. Include the first two (both 1). What registers are required? What states? Draw the state transition diagram for the controller. Write the Verilog code.