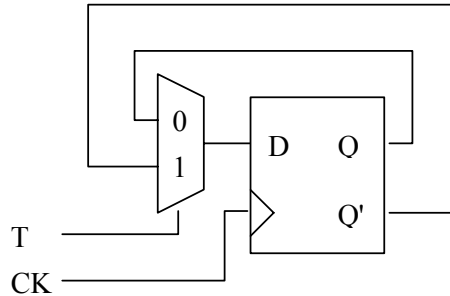


**Homework #3**

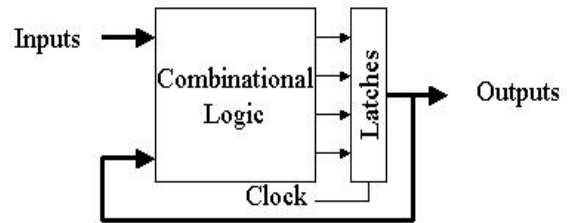
Due: Friday, September 28, 2001

1. Show how to implement a T flip-flop starting with a D flip-flop. The T flip-flop is a FF that toggles when the input is 1 and holds its state when the input is 0.

*Solution*



2. In the sequential circuit pictured the flip-flops have setup times of 15ns, propagation delays of 10 ns and hold times of 5ns.

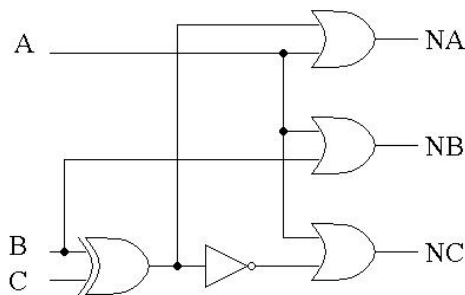


- a. Assuming no propagation delay through the combinational logic block, what is the maximum allowable frequency for the clock signal?
- b. Assuming a propagation delay of 60 ns through the combinational logic, how does your answer to part (a) change?

*Solution*

- a)  $T - t_p > t_{su} \Rightarrow T > 15\text{ns} + 10\text{ns} \Rightarrow f_{\text{max}} = 1/25\text{ns} = 40\text{MHz}$
- b)  $T - t_p - t_D > t_{su} \Rightarrow T > 15\text{ns} + 10\text{ns} + 60\text{ns} \Rightarrow f_{\text{max}} = 1/85\text{ns} = 11.76\text{MHz}$

3. You are given a sequential circuit that has the following circuit to compute the next state:



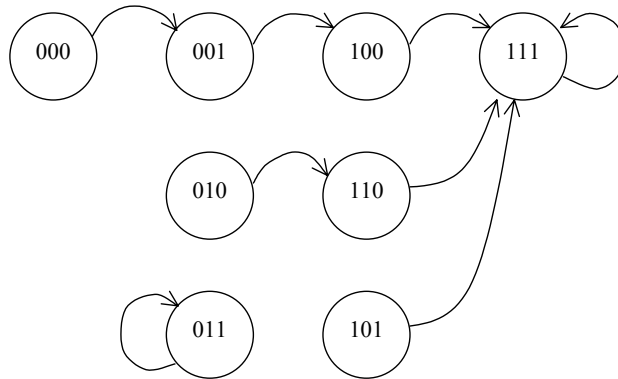
- a. Write out the state transition table for the above circuit.
- b. Draw the state diagram.

Solution

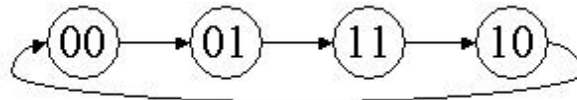
a)

A	B	C	NA	NB	NC
0	0	0	0	0	1
0	0	1	1	0	0
0	1	0	1	1	0
0	1	1	0	1	1
1	0	0	1	1	1
1	0	1	1	1	1
1	1	0	1	1	1
1	1	1	1	1	1

b)



4. A Gray code is a sequence of bit patterns in which only one bit changes from one element to the next element. Consider the following FSM for a two bit Gray code:



- Write out the state transition table.
- Assuming that you have JK flip flops, write out the equations for the next state decoder.
- Draw the schematic for the circuit.
- Suppose that we want to be able to start and stop the counter based on an enable signal. How would your schematic change if you wanted to add this input to your counter?
- Consider the addition of a second control signal, called "D" for direction. If this signal is high then we will proceed through the above state diagram as shown, from left to right. If D is unasserted, then we will transition in the opposite direction. Write out the equations for the new state transitions. Do not include the enable signal in this circuit.
- What use could a counter like this have?

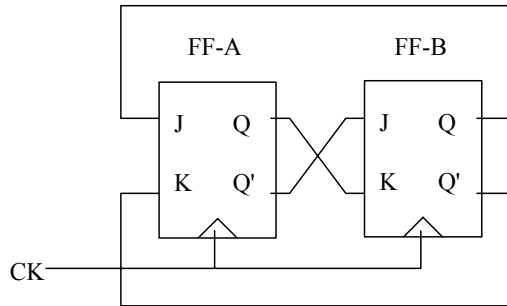
Solution

a)

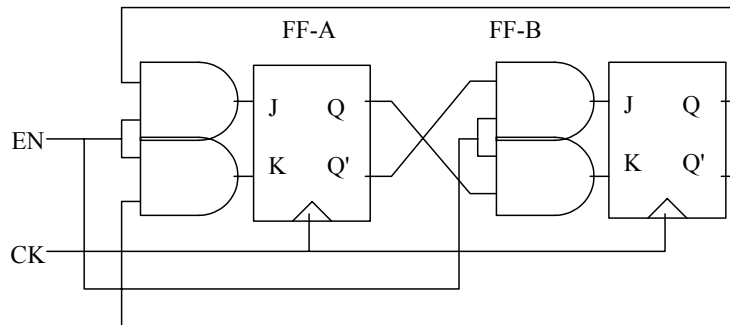
A	B	NA	NB
0	0	0	1
0	1	1	1
1	0	0	0
1	1	1	0

b)  $NA = B \Rightarrow JA = B, KA = B'$   
 $NB = A' \Rightarrow JB = A', KB = A$

c)



d)



e)

D	A	B	NA	NB
0	0	0	0	1
0	0	1	1	1
0	1	0	0	0
0	1	1	1	0
1	0	0	1	0
1	0	1	0	0
1	1	0	1	1
1	1	1	0	1

$$NA = D'B + DB' = D \oplus B$$

$$NB = D'A' + DA = (D \oplus A)'$$

f) An up/down counter can have many applications. For example, suppose a communication system where we send a packet and receive an acknowledge for every packet that was sent. Every time we send a packet we count up, and every time we receive an

acknowledge we count down. If the count reaches certain predetermined value N, it means that we sent N packets which have not been acknowledged yet, therefore we stop sending packets to avoid an overflow on the other side.

A Gray code counter has the property that consecutive values differ in only one bit. Since every time a bit changes some current is drained from the power source, a Gray code counter would have less power consumption than a binary counter. It is also used to code signals for error detection purposes, and even to solve the "Towers of Hanoi" puzzle!

5. Shifters are normally used to shift data in a circular pattern (the data that shifts out at one end of the shifter is shifted back into the other end), or as a logic shift (fill the shifted positions with 0's) or an arithmetic shift (propagate the high-order sign bit to the right or shift in 0's to the left). For example, if a 4-bit register contains the data 1110, the effects of the six kinds of shifts are the following:

Circular shift right: 1110 becomes 0111.  
 Circular shift left: 1110 becomes 1101  
 Logical shift right: 1110 becomes 0111  
 Logical shift left: 1110 becomes 1100  
 Arithmetic shift right: 1110 becomes 1111  
 Arithmetic shift left: 1110 becomes 1100

Show how to wire the universal shift register introduced in class (section 2 p. 32) to perform the following types of shifts:

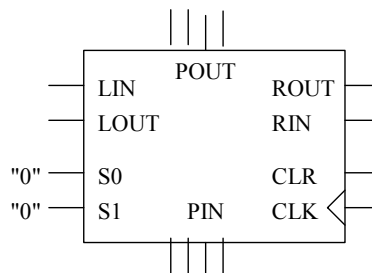
- a. Hold (don't shift)
- b. Circular shift right
- c. Logical shift left
- d. Arithmetic shift right

Suppose that you want to implement a shift register unit that can switch between the functions implemented above (3a – 3d). This shift register should have two control inputs S1 and S0 that will select the shift register's function. The mapping should be as follows: S1, S0 = 00 is "hold," S1, S0 = 01 is "circular shift right," etc.

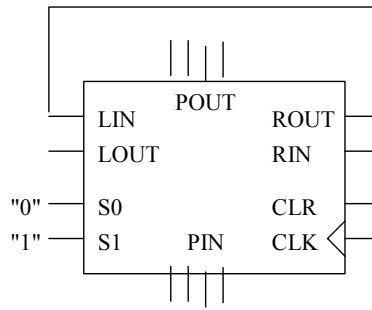
- e. Show the combinational logic (equations or schematics) to decode the control signals S1, S0 into the appropriate functional behavior for the shifter.

*Solution*

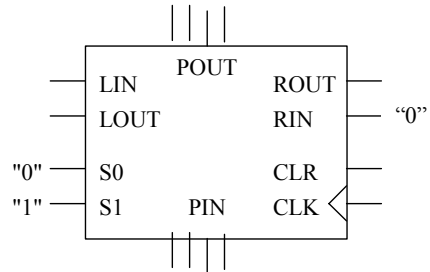
a)



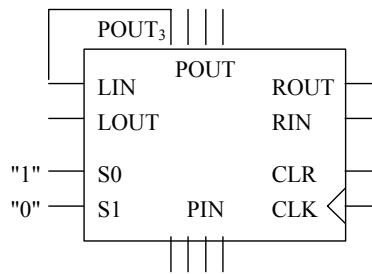
b)



c)



d)



e)

	S1	S0	us0	us1	LIN	RIN
hold	0	0	0	0	X	X
circ. right	0	1	0	1	ROUT	X
logic. left	1	0	1	0	X	0
arith. right	1	1	0	1	POUT <sub>3</sub>	X

(us0 and us1 are the control inputs of the universal shift register)

$$us0 = S1 S0'$$

$$us1 = S0$$

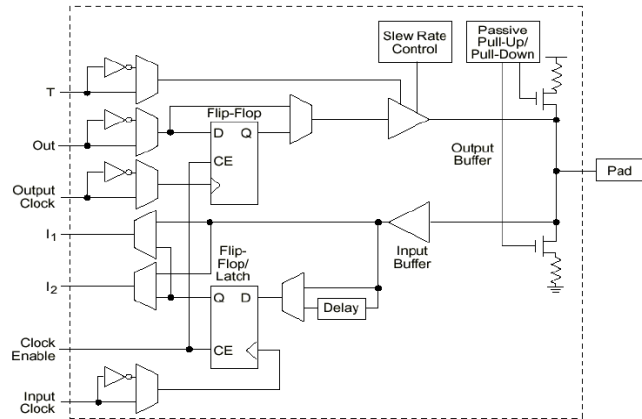
$$LIN = S1' S0 ROUT + S1 S0 POUT_3$$

Or using the don't care conditions to simplify:

$$LIN = S1' ROUT + S1 POUT_3$$

$$RIN = 0 \text{ (the other conditions are don't care)}$$

6. In the Xilinx Input Output Block (IOB), the pad can be either an input or an output. How do we choose which function we want a pad to have? How do we keep an output from coming back in as an input?



### *Solution*

From a user point of view, we select the function of the pad by inserting a specific symbol in our schematic (i.e., IPAD or OPAD) and assigning a pin number to it. What actually happens in the Xilinx device, is that during the configuration process the multiplexors and buffers are programmed according to the function selected. If the pad is to be used as an output, then the output buffer is enabled (the 3-state control signal T is set to active if it is not used explicitly in our design), and the Out signal is connected to some internal signal via the interconnection matrix. If the pad is to be used as an input, then the output buffer is disabled, and the input is connected to some internal CLB via the interconnection matrix (either I1 or I2 ). If the input or output is to be registered, then the multiplexors will route the output of the flip-flops to the proper destination.

We keep the output from coming back as an input by simply not connecting the signals I1 and I2 to the rest of the circuit.