EECS 151/251 A Discussion 10

March 22, 2024

Berkeley
UNIVERSITY OF CALIFORNIA

Content

- Questions with common mistakes:
  - Q3
  - Q5
  - Q6
  - Q11

Question 3

3 FPGA LUT Circuit [12pts]

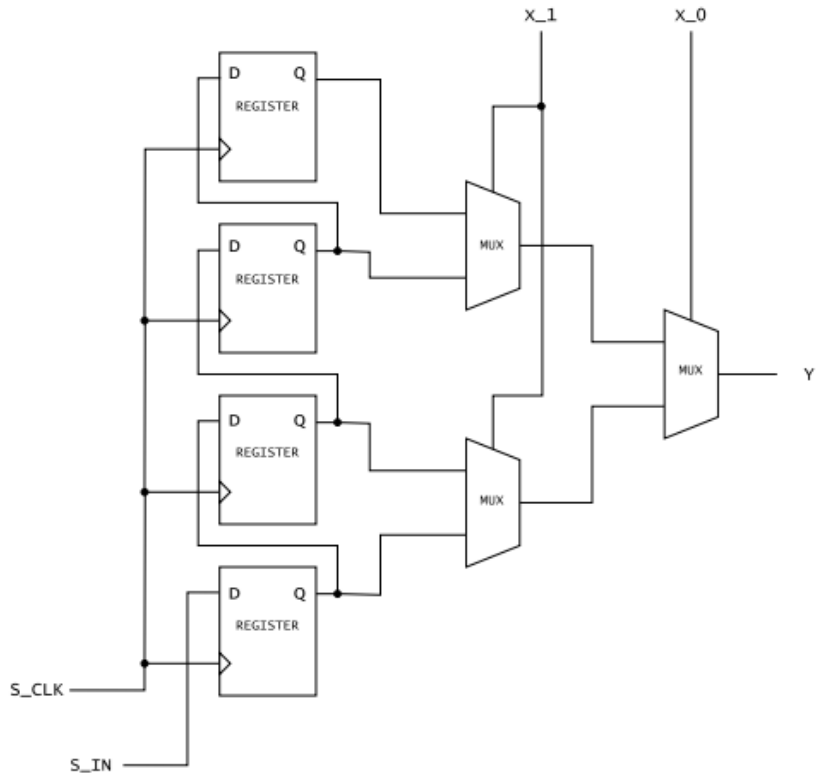
1. In the box below and using the indicated port signals, complete the drawing of the internal circuitry of a 2-LUT, using FFs, 2-to-1 multiplexors, and simple logic gates as needed. S_{in} is a bit-serial data input port and S_{clk} is a clock signal to be used for configuring the LUT function. x1, are x0 are the LUT data inputs and y is the LUT output.

2. How many distinct logic functions can this LUT implement?

Question 3

Solution:

1. Here is the corresponding diagram:



Note that S_{in} is a one bit signal and S_{clk} is only used for configuration.

2. A 2-LUT can implement 2^{2^2} = 16 functions.

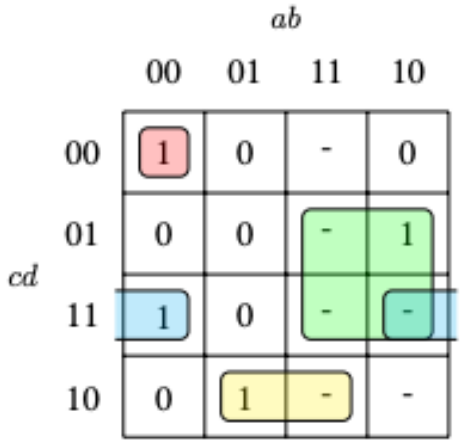
Question 5

5 Combinational Logic Design [10pts]

Recall that Binary Coded Decimal (BCD) is a number representation that uses the binary encodings 0000–1001 to represent the 10 decimal digits. Derive two combinational logic functions that each accept a BCD digit and outputs a 1 iff that digit is evenly divisible by 3. (Yes, 0 is divisible by 3.) One expression should be in SOP form and the other is POS, and both should be minimized.

Question 5

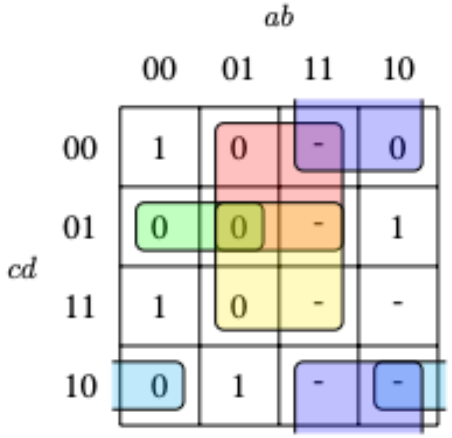
SOP:



Boolean expression:

$$f = \bar{a}\bar{b}\bar{c}\bar{d} + \bar{b}cd + ad + bc\bar{d}$$

POS:



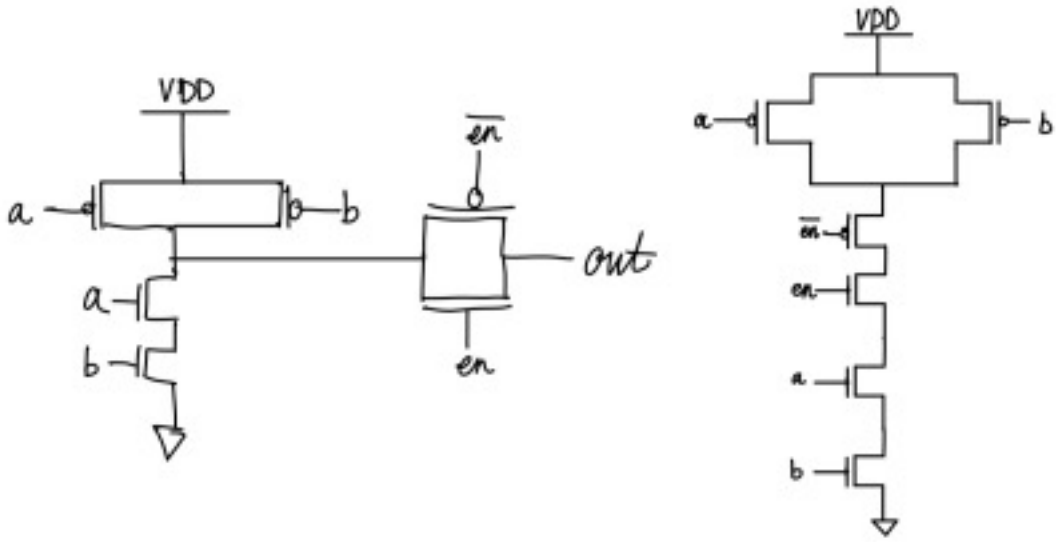
Boolean expression:

$$f = (a + c + \bar{d})(\bar{b} + c)(\bar{b} + \bar{d})(\bar{a} + d)(b + \bar{c} + d)$$

Question 9

9 Tri-state Buffers [10pts]

1. For some application you need a tri-state buffer, but you would like to combine it with a NAND function. Draw a transistor level circuit that would achieve both functions with the minimal number of transistors. Label the inputs and outputs.



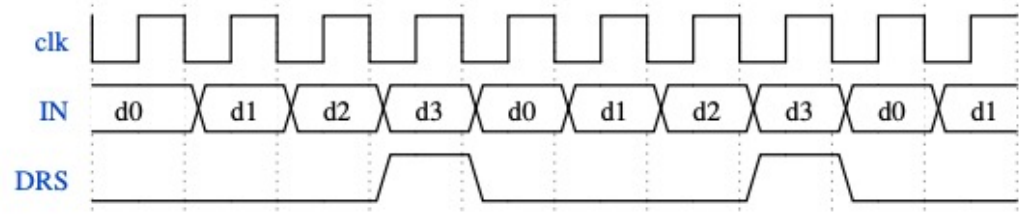
Question 11

11 Circuit Design [12pts]

In lecture we presented a parallel-to-serial converter. It could be used, for instance, for sending words over a wire or a wireless link, one bit at a time. Your task here is to design a circuit for receiving the bits, a *serial-to-parallel converter*. You are tasked to design a circuit that adheres to the following specifications, using FFs, multiplexors, and simple logic gates as needed.

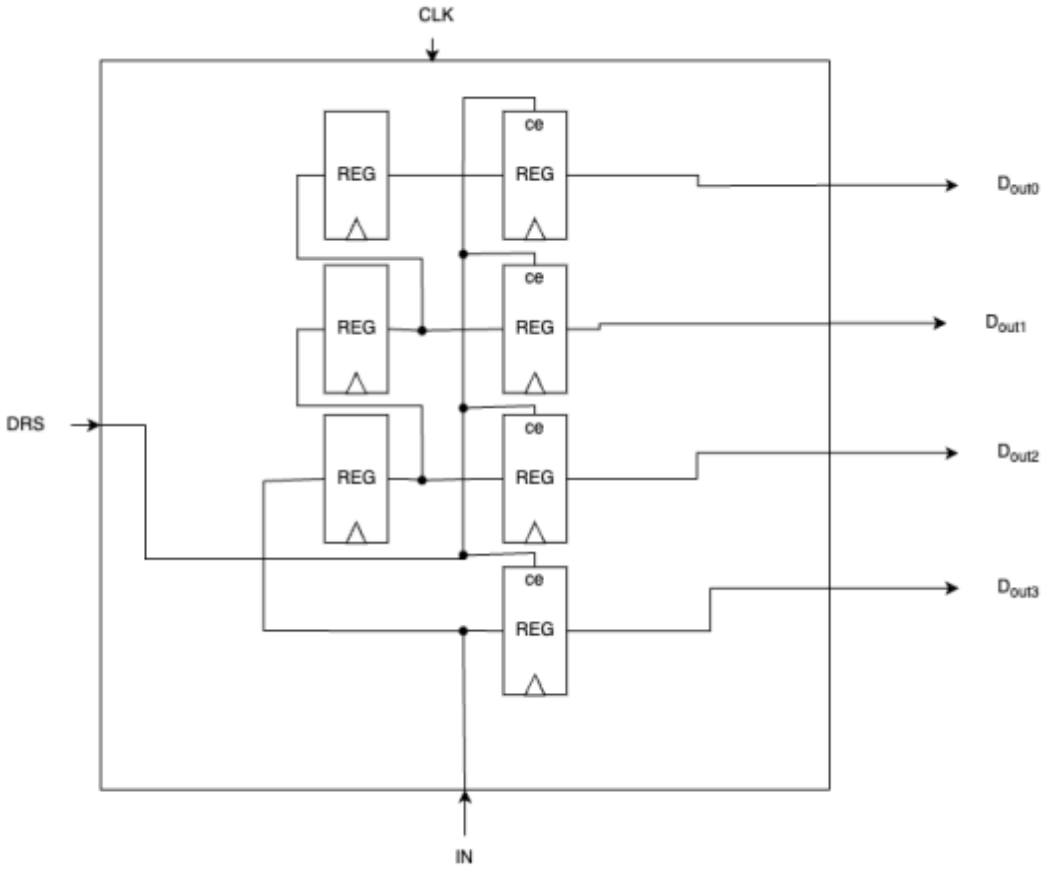
1. Your circuit will receive the bits of each word (4-bit words in this case), LSB first, one per clock cycle as shown below and must collect up the bits and present them to the external interface in word form.
2. The external interface supplies a "data request signal (DRS)" every four clock cycles that your circuit should use to provide the received bits to the interface. The output needs to remain stable until the next occurence of the DRS signal.

The input waveform will look like the following:



Neatly draw your circuit in the box provided on the next page:

Question 11

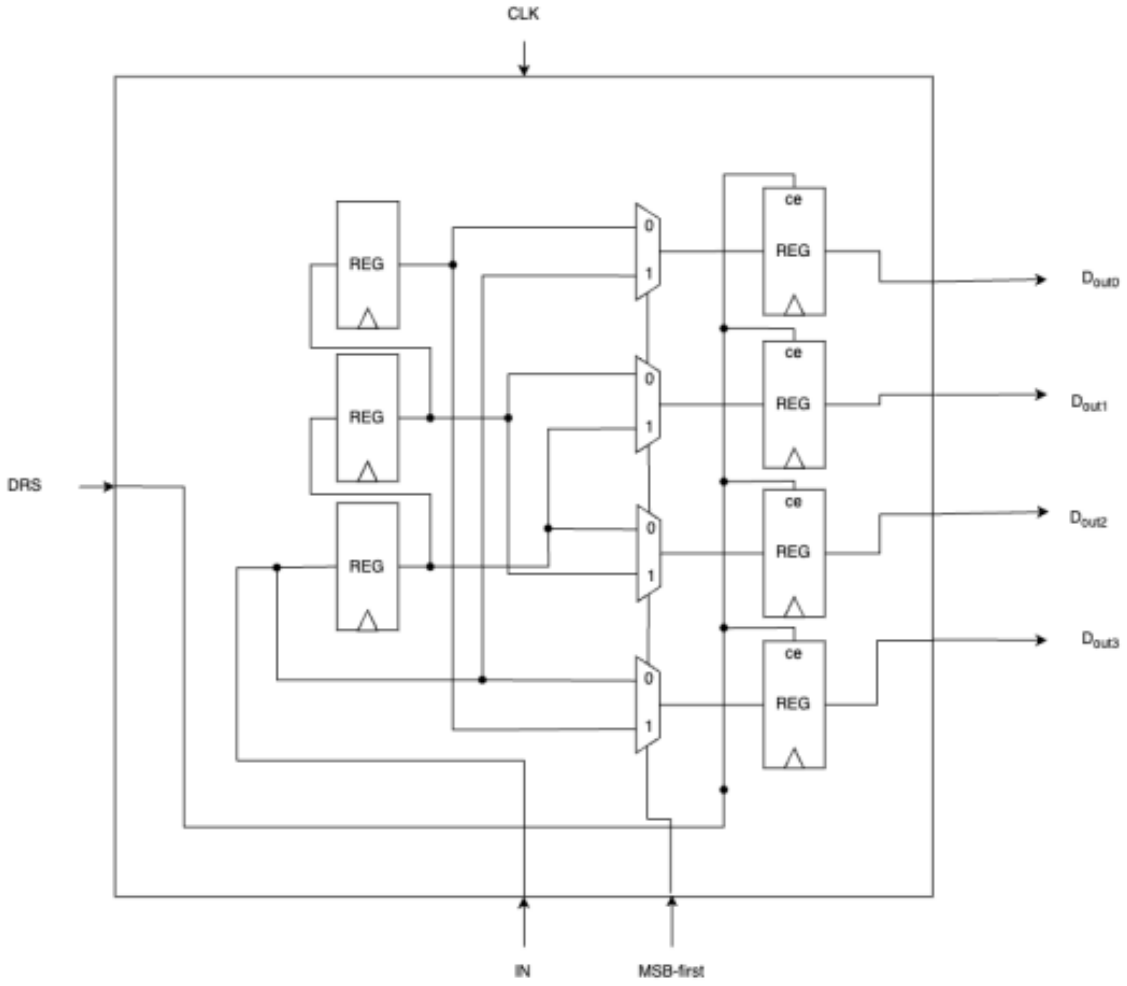

CLK
REG
ce
REG
REG
ce
REG
REG
ce
REG
REG
ce
REG
DRS
IN
D_{out0}
D_{out1}
D_{out2}
D_{out3}

Question 11

Now suppose, the external interface wants to be able to get the received word either MSB-first or LSB-first (big vs. little endian). The external circuit will additionally send an "MSB-first" signal every four clock cycles. If the MSB-first signal is high, then the first bit received in the stream (d0) should be considered the MSB, and the last bit (d3) the LSB. If the MSB-first signal is low, then the last bit received is the MSB and the first bit is the LSB.

Draw the updated circuit below. You may choose to abstract your answer from above and add additional circuitry, or to redraw it with modifications as needed.

Question 11


CLK
DRS
REG
REG
REG
0
1
0
1
0
1
0
1
ce
REG
ce
REG
ce
REG
ce
REG
D_{out0}
D_{out1}
D_{out2}
D_{out3}
IN
MSB-first