CSE260 Practice Sheet

Q: Are all of these enough to get full marks in the exam?

A: NO. This is a practice sheet. Meaning, you can practice all you want using the questions from this sheet. However, doing well in exams depends upon your ability to understand a question, formulate an answer, and express it correctly. You see, these are humane skills which cannot be guaranteed from completing a practice sheet only. But yeah, Best of luck anyways.

- Convert the following binary numbers to equivalent decimal numbers.
  (a) $(101110001001)_2$
  (b) $(11011.101)_2$
- Convert the following decimal number to equivalent binary numbers: $(4195)_{10}$
- Convert the following decimal number to equivalent binary numbers: $(3785.65625)_{10}$
- Convert the following decimal number to equivalent binary numbers: $(4785.150263)_{10}$ [for infinite fractional part, just do 6-7 steps and use dots for the rest]
- Convert the following decimal number to equivalent base 5 numbers: $(4123)_{10}$
- Convert the following decimal number to equivalent hexadecimal numbers: $(513)_{10}$
- Convert the following decimal number to equivalent base 9 numbers: $(813)_{10}$
- Perform the following base conversions
  a) $(29)_{12} = (?)_7$
  b) $(10110111)_5 = (?)_4$
- Perform addition, subtraction and multiplication for the pair of following base-9 numbers. Verify your results by converting the problem into decimal.
  412
  134
- Subtract 13 from 27 in 7 bits using 2's complement number system and justify whether there is an overflow or not.
- Subtract 45 from 98 in 12 bits using 2's complement number system and justify whether there is an overflow or not.
- Add 13 with 27 in 6 bits using 2's complement number system and justify whether there is an overflow or not.
- Add -57 with 63 in 11 bits using 2's complement number system and justify

whether there is an overflow or not.

- Perform the following arithmetic operations using 13-bit two's complement and one's complement systems . State if there is an overflow in each case.
  - a) 91 - 499
  - b) 379 + 98

- You are a computer engineer and you want to buy two 8 GB DDR4 RAMs. Each RAM costs $(1C2)_{16}$ dollars. You also want to buy a graphics card RTX4070Ti which costs $(10010110000)_2$ dollars. However, you don't have that much money with you and you are afraid to ask your parents about it. Suddenly, one of your generous friends agreed to give you the money you need. He decided to give you $(4064)_8$ dollars. How much will you have left after buying those components? (Show the answer in decimal)
- Simplify the following boolean expression to minimum number of literals:
  $(A + B)\,(A + \overline{B})\,(\overline{A} + C)$
- Find the complement of the following expression:
  (x' + y + z')(x' + y')(x + z')
- Draw the following functions using NAND gates only:
  F(A,B,C,D) = (A'B'CD' + A'D + (B+D'))
  NB: Please draw horizontally on your script.
  NB: You can't simplify the above functions and then draw using NAND gate. You have to draw based on the function given in question
- Draw the following functions using NOR gates only:
  F(A,B,C,D) = (AB'C'D' + AD + (B+D'))
  NB: Please draw horizontally on your script.
  NB: You can't simplify the above functions and then draw using NAND gate. You have to draw based on the function given in question
- Find out SOP and POS for the following:
  1. F(A,B,C) = AB+BC'
  2. F(A,B,C,D) = A+ B'CD'
  3. F(A,B,C,D,E) = AB+CDE
- **Simplify** the following boolean equation using the laws of boolean algebra and implement the simplified function using only NOR gates:
  F(a,b,c,d) = Σ(8, 9, 10, 11, 13, 15)

- Find the simplified function using K-map:
  1. F(a,b,c,d) = Σ(8, 9, 10, 11, 13, 15)
  2. F(a,b,c,d) = Σ(0, 1, 2, 11, 13, 15)
  3. F(a,b,c) = Σ(0,3,5,7)
  4. F(a,b,c,d) = Σ(8, 9, 10) + d(0,1,2)
  5. F(a,b,c,d) = Σ(1, 9, 10) + d(0,8,11,12)
  6. F(a,b,c,d) = π(1, 9, 10) + d(0,8,11,12)
  7. F(a,b,c,d) = Σ(0, 1, 2,3,4,5,6,7,8,9) + d(10,11,12,13,14)
  8. F(a,b,c,d) = Σ(0, 1, 2,3,4,5,6,7,8,9) + d(10,11,12,13,14,15)
  9. F(a,b,c,d) = Σ( ) + d(4,5,9)
  10. F(a,b,c,d) = Σ( ) + d( )

- Find the simplified function using tabulation method:
    11. $F(a,b,c,d) = \Sigma(8, 9, 10, 11, 13, 15)$
    12. $F(a,b,c,d) = \Sigma(0, 1, 2, 11, 13, 15)$
    13. $F(a,b,c) = \Sigma(0,3,5,7)$
    14. $F(a,b,c,d) = \Sigma(8, 9, 10) + d(0,1,2)$
    15. $F(a,b,c,d) = \Sigma(1, 9, 10) + d(0,8,11,12)$
    16. $F(a,b,c,d) = \pi(1, 9, 10) + d(0,8,11,12)$
    17. $F(a,b,c,d) = \Sigma(0, 1, 2,3,4,5,6,7,8,9) + d(10,11,12,13,14)$
    18. $F(a,b,c,d) = \Sigma(0, 1, 2,3,4,5,6,7,8,9) + d(10,11,12,13,14,15)$

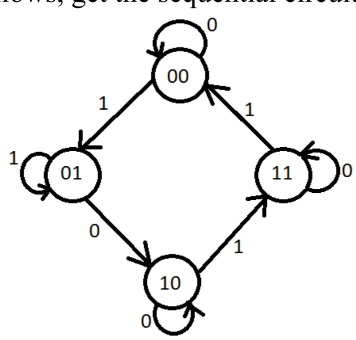
- K-Maps Scenario Based Problems:
  https://docs.google.com/document/d/1eJ2ItIDALeGNAVn1iUGjXkjEJCbmhoYA0LH2RHIP8rc/edit

  For all the following: Make sure that your circuit is efficient, meaning you should use the lowest number of components. You may use external gates if required.

- Build an adder cum subtractor (4 bits)
- Draw the block diagram of a 20 bits parallel adder.
- Build a 13 person voting system using full and parallel adders.
- Build a 8 person voting system using full and parallel adders.
- Build a 15 person voting system using full and parallel adders.
- Consider A is a 4 bit number. Design A-3 using a 4 bit parallel adder. Use external gates if required.
- Consider A is a 4 bit number. Design A+ 3 using a 4 bit parallel adder. Use external gates if required.
- Consider two numbers: 7 and 5. You can only calculate addition and subtraction between those two numbers. Design a circuit that can perform the above calculations based upon the user's intention.
- Design a full adder using two half adders. You must use two NOR gates and no OR gates.
- Design a full adder using two half adders. You must use three NAND gates and no OR gates.
- Build a circuit that implements the 2's complement number system (3 bits) using encoder(s) and decoder(s).
- Design an Octal to Binary encoder.
- Design a 4x2 priority encoder prioritizing MSB.
- Design a 4x2 priority encoder prioritizing LSB.
- Implement the following boolean function using a single 16:1 mux .
  $F(A,B,C,D) = \sum(0,1,2,7,8,10,11,13, 15)$. Use external gates if required.
- Implement the following boolean function using a single 8:1 mux.
  $F(A,B,C,D) = \sum(0,1,2,7,8,10,11,13, 15)$. Use external gates if required.
- Implement the following boolean function using a single 4:1 mux.
  $F(A,B,C,D) = \sum(0,1,2,7,8,10,11,13, 15)$. Use external gates if required.
- Implement the following boolean function using both 4:1 and 2:1 mux in a single circuit
  $F(A,B,C,D) = \sum(0,1,2,7,8,10,11,13, 15)$. Use external gates if required.
- Implement the following boolean function using **a)** 4x16 decoder(s) only **b)** 2x4 decoder(s) only
  $F(A,B,C,D,E) = \sum(0,1,2,7,8,10,11,13, 15,18,21,24,25)$. Use external gates if required.

- Build a full adder using encoder(s) and decoder(s).
- Build a BCD to Excess-3 code converter using encoder(s) and decoder(s).
- Build a BCD to Excess-5 code converter using encoder(s) and decoder(s).
- Design the circuit diagram for a 3x8 decoder.
- Design the circuit diagram for a 2x4 decoder.
- Design the circuit diagram for a 4:1 mux.
- Design a 8:1 mux using both 4:1 mux and 2:1 mux in a single circuit.
- Design AND, OR, NOT gate using 4:1 mux.
- Design AND, OR, NOT gate using 2:1 mux.
- Design a circuit that can demonstrate the usage of ENABLE pin in decoders.
- Design a D FF using SR FF.
- Design a T FF using JK FF.
- Design a D FF using JK FF.
- Design a SR FF.
- Write down the excitation table for SR, D, JK and T FF.
- Write down the characteristics table for SR, D, JK and T FF.

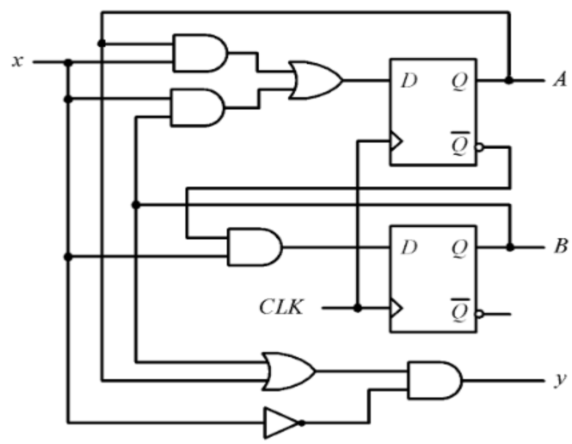
- Given the state diagram as follows, get the sequential circuit using SR flipflop.



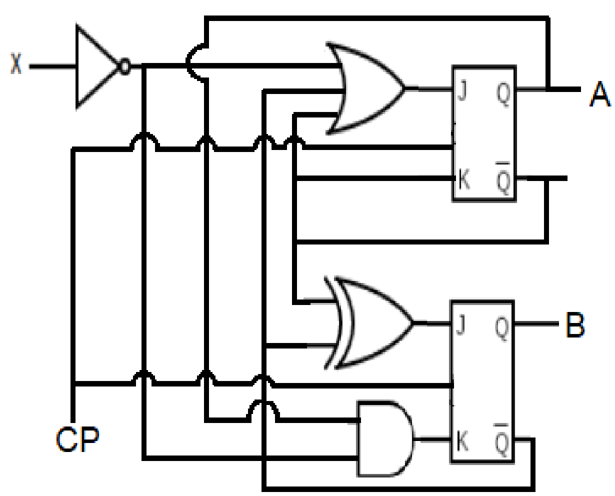
- Implement the following counter using T flip flop
  CSE110 -> CSE111 -> CSE220 -> CSE221 -> CSE331 -> CSE221 -> CSE321 -> CSE110
- 3->4->6->10->12->13->15->3
  1. Implement the given counter using JK flip-flop.
  2. Implement the given counter using T flip-flop.

  NB: For states not given in question, please move to the initial state as per question.
- Implement 4 bit up counter using JK flip flop
- Implement 4 bit down counter using JK flip flop

- Implement 2 bit up/down counter using D flip flop
- Implement 3 bit up/down counter using T flip flop
- Implement 2 bit up/down counter using SR flip flop
- Implement 3 bit up/down counter using JK flip flop
- Implement the following counter using T FF: Green->Orange->Yellow->Red->Yellow->Orange->Yellow->Green
- Implement the following counter using JK FF: Green->Yellow->Red->Yellow->Green
- Implement the following counter using JK FF: 1->2->3->5->7->11->13->1
- Implement the following counter using SR FF: 2->3->6->8->10->12->2
- Draw the state diagram for the given circuit.



- Draw the state diagram for the given circuit.



- How many address lines do we need for a 64 MB RAM with 16 bit/words?
- Design a 2048x32 RAM showing internal details.
- Design a 4096x16 RAM showing internal details.
- What is the capacity for a 2^16 x 16 RAM in Gigabytes?
- Draw the block diagram of a binary cell.
- Draw and explain the functionalities of MAR and MBR.
- What type of information can you get from the “2048x32” part of a 2048x32 RAM?
- How many types of RAM are there?

- What is static RAM?
- What is dynamic RAM?
- Convert 8,796,093,022,208 Bits to Gigabytes.
- Convert 3 Terabytes to bits.