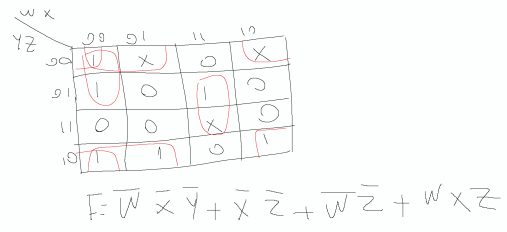
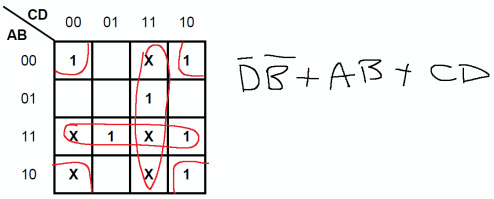


CS223 – DIGITAL DESIGN
HOMEWORK # 2

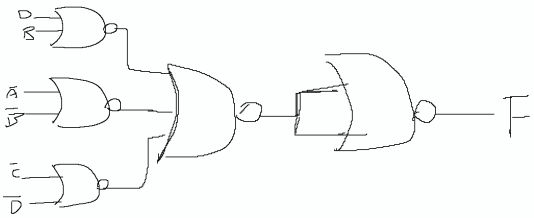
Q1. Given the logic function $\boldsymbol{F(w,x,y,z) = \sum m(0,1,2,6,10,13) + \sum d(4,8,15)}$ where $\boldsymbol{m}$ represents minterms of the function and $\boldsymbol{d}$ represents don't cares. Find the minimal (optimal) Boolean expression for $\boldsymbol{F}$ using the Karnaugh map method.

Q2. Find a minimal Boolean expression for the function given in the following Karnaugh map. Implement the function $\boldsymbol{F}$ using only NOR gates assuming that the complements of input variables are available.

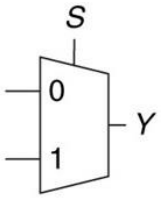


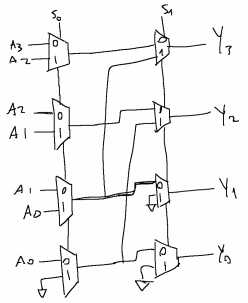
$\bar{D}\bar{B} + AB + CD$

AB \ CD	00	01	11	10
00	1		X	1
01			1	
11	X	1	X	1
10	X		X	1

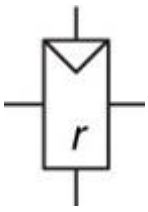


Q3. Design a 4-bit left shifter using only eight 2:1 multiplexers. The shifter accepts a 4-bit input $\boldsymbol{A_{3:0}}$ and a 2-bit shift amount $\boldsymbol{S_{1:0}}$. It produces a 4-bit output $\boldsymbol{Y_{3:0}}$. Write the truth table and sketch the circuit schematic using the multiplexer symbol given below.

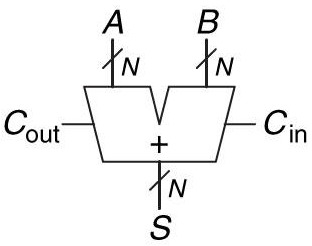




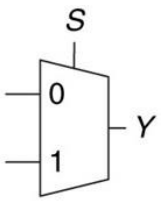
Q4. You will design an 8-bit synchronous Up/Down counter. The inputs are CLK, Reset, and Up. When Reset is 1, the output bits $\boldsymbol{C_{7:0}}$ are all 0. Otherwise, when **Up = 1**, the circuit **counts up**, and when **Up = 0**, the circuit **counts down**. You can use only one 8-bit **register** with reset input, one 8-bit **adder**, and one 8-bit 2:1 **multiplexer**. Sketch the circuit schematic of your design using the component symbols given below.



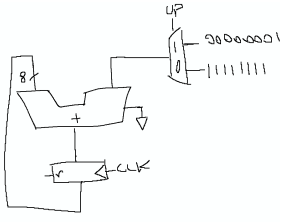
Register



Adder



Multiplexer



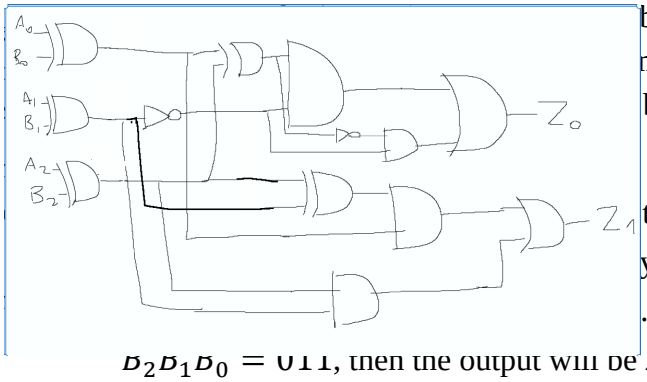
$F = \bar{B}C + CD + A\bar{D} + \bar{A}B\bar{C}$

$F = (A + \bar{B} + \bar{C} + D)(A + B + C)(\bar{A} + C + \bar{D})$

Q5. For the logic function $\boldsymbol{F = BD(\bar{A} + C) + A(C + \bar{D}) + \overline{(B + \bar{C})(A + \bar{B} + C + D)}}$

(a) Write the truth table.

(b) Write the function in Sum-of-Products (SOP) and Product-of-Sums (POS) canonical forms.

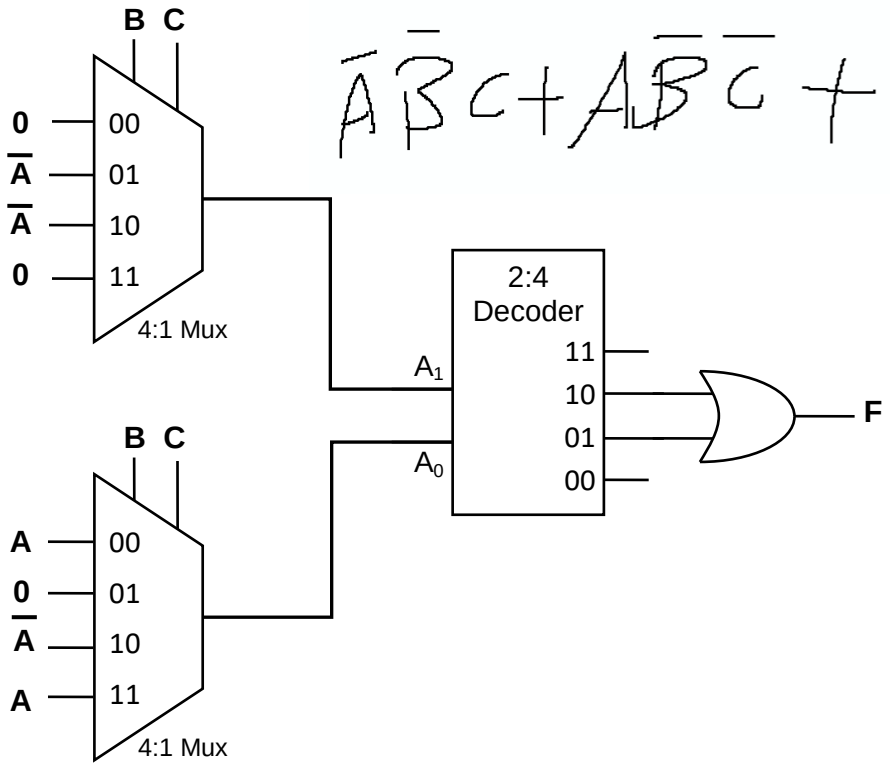


binary numbers as the number of bit positions that the
mple, ween 001 and 100 is 2 since they
betw s 3 since they differ in three-bit
000 er in one-bit position.

take umbers ($A_2A_1A_0$ and $B_2B_1B_0$)
y nu utput. The output will give the
. For example, if the inputs are $A_2A_1A_0 = 101$ and $B_2B_1B_0 = 011$, then the output will be $Z_1Z_0 = 10$ indicating that the distance is equal to 2.

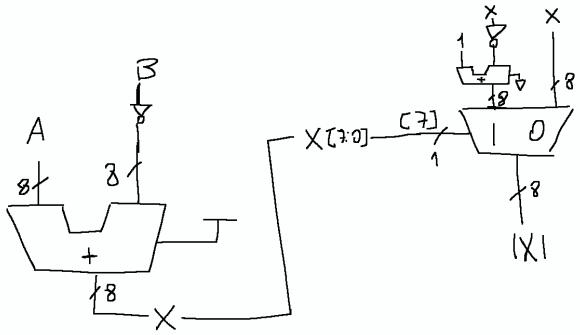
Draw the logic diagram of your circuit. You can use **only** AND, OR, NOT, and XOR gates.

Q7. Find the minimized Boolean expression for the logic function performed by the following circuit (inputs are $\boldsymbol{A, B, C}$ and the output is $\boldsymbol{F}$).



$\bar{A}\bar{B}C + A\bar{B}\bar{C} + ABC$

Q8. Design a circuit that wil



presented by using two's complement. Use e 8-bit numbers in two's complement form. l logic gates. *Show your design clearly by dr s.*

Q9. The state transition diagram of an FSM is given below. The FSM receives two inputs **A**, **B** and it has one output **Y**.

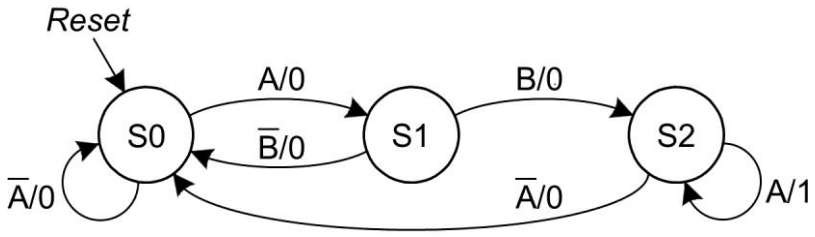
(a) Write the state transition and output table using binary state encodings.

(b) Write the minimized Boolean equations for the next-state logic and output logic.

$S_1' = \bar{S}_1 S_0 B + S_1 \bar{S}_0 A$

$S_0' = \bar{S}_1 \bar{S}_0 A$

$Y = S_1 A$

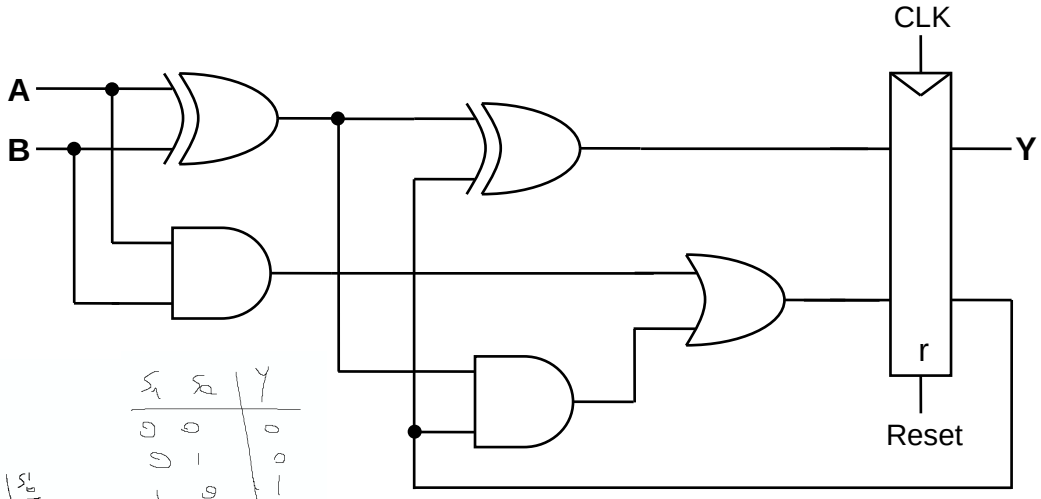


S_1	S_0	A	B	S_1'	S_0'	Y
0	0	0	X	0	0	0
0	0	1	X	0	1	0
0	1	X	0	0	0	0
0	1	X	1	1	0	0
1	0	0	X	0	0	0
1	0	1	X	1	0	1

Q10. Analyze the Finite State Machine (FSM) given below.

(a) Write the state table and output table for the FSM using binary state encodings.

(b) Sketch the state transition diagram of the FSM.



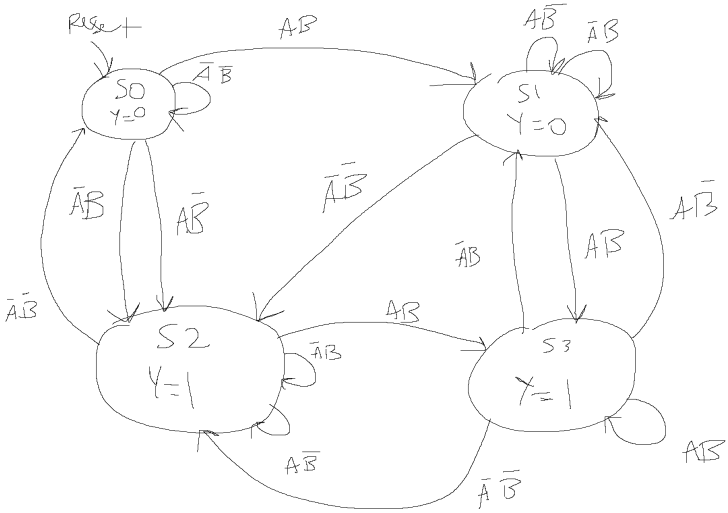
$S_1' = S_0 \oplus (A \oplus B)$

$S_0' = S_0 (A \oplus B) + \bar{A} B$

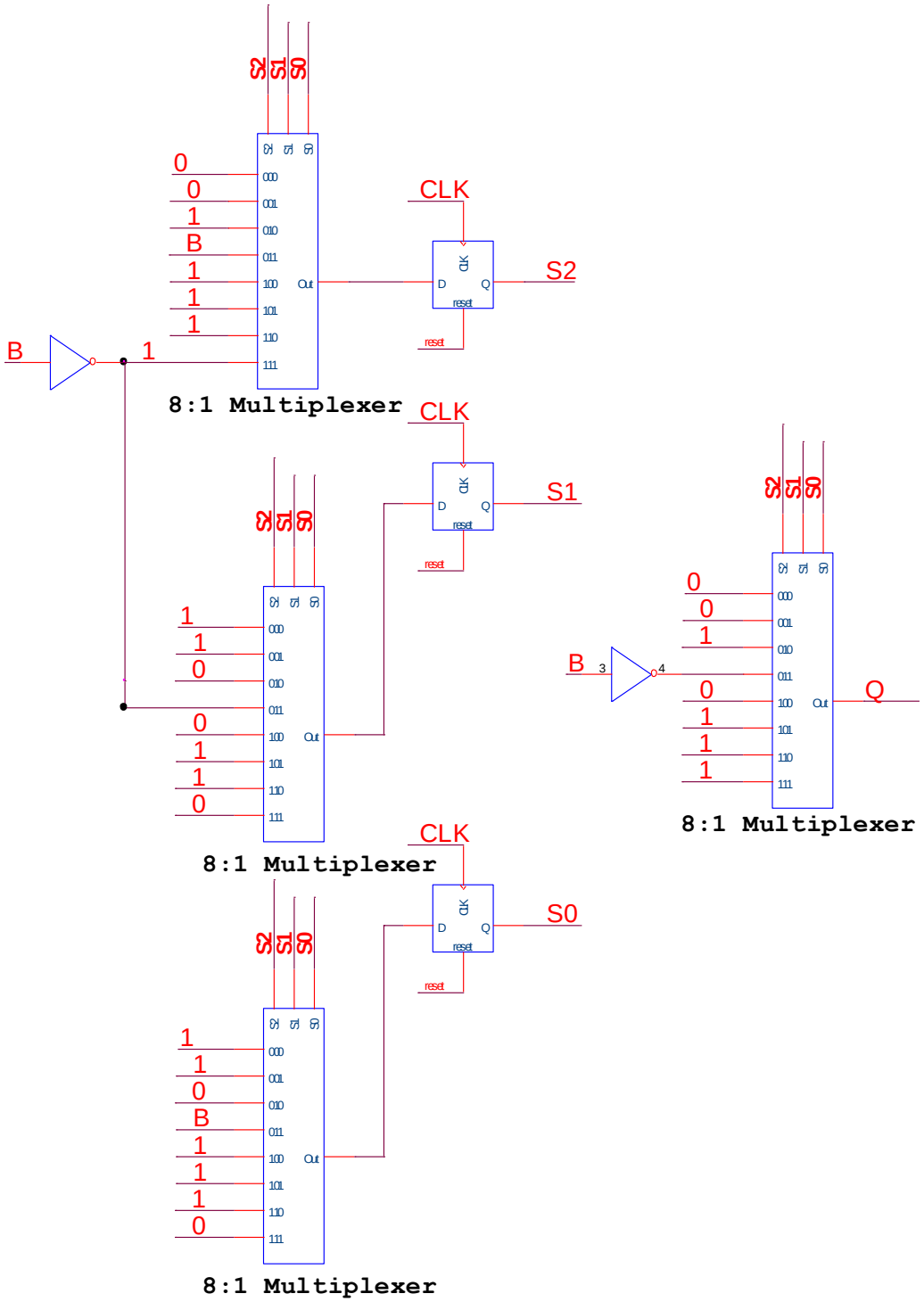
$Y = S_1$

S_1	S_0	Y
0	0	0
0	1	0
1	0	1
1	1	1

S_1	S_0	A	B	S_1'	S_0'
0	0	0	0	0	0
0	0	0	1	1	0
0	0	1	0	1	1
0	1	0	0	1	0
0	1	0	1	1	0
0	1	1	0	0	0
0	1	1	1	1	0
1	0	0	0	0	0
1	0	0	1	1	0
1	0	1	0	1	1
1	0	1	1	0	1
1	1	0	0	1	0
1	1	0	1	1	0
1	1	1	0	1	1
1	1	1	1	1	1



Q11. For the sequential circuit (FSM) given below (**B** is the input, **Q** is the output)



(a) Write the Boolean equations for the next-state and output logic in sum-of-products form.

$S'_2 =$ $S_2' = \bar{S}_1 S_0 + S_1 \bar{S}_0 + S_2 \bar{S}_1 + S_2 \bar{B} + \bar{S}_2 S_0 B$

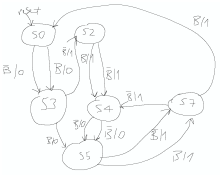
$S'_1 =$ $S_1' = \bar{S}_2 \bar{S}_1 + \bar{S}_1 S_0 + \bar{S}_2 S_1 \bar{B} + S_2 S_1 \bar{S}_0$

$S'_0 =$ $S_0' = \overline{S_1} + S_2 \bar{S}_0 + \overline{S_2} S_1 B$

$Q =$ $Q = S_1 \bar{S}_0 + S_1 \bar{B} + S_2 S_0$

(b) Write the state transition and output tables given below.

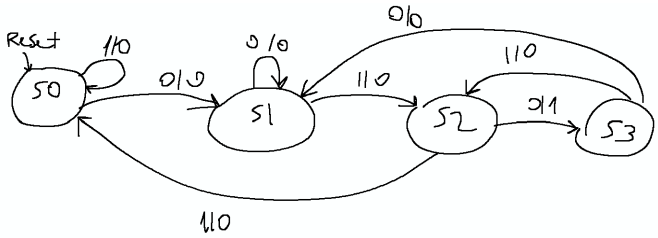
S_2	S_1	S_0	B	S_2'	S_1'	S_0'	Q
0	0	0	0	0	1	1	0
0	0	0	1	0	1	1	0
0	0	1	0	1	1	1	0
0	0	1	1	1	1	1	0
0	1	0	0	1	0	0	1
0	1	0	1	1	0	0	1
0	1	1	0	0	1	0	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	1	0
1	0	0	1	1	0	1	0
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	1
1	1	0	0	1	1	1	1
1	1	0	1	1	1	1	1
1	1	1	0	1	0	0	1
1	1	1	1	0	0	0	1



(c) Draw the state transition diagram for this FSM. Disregard (don't draw) any states that cannot be entered.

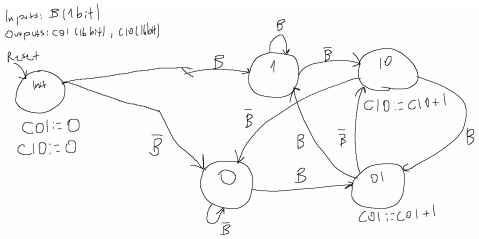
(d) Analyze the finite state machine and in one or two sentences, tell what it does. You should summarize the overall function of the FSM. Do not go into details like "when in this state, if such-and-so happens, then this happens...", etc.

Q12. Design a **Mealy** Fir

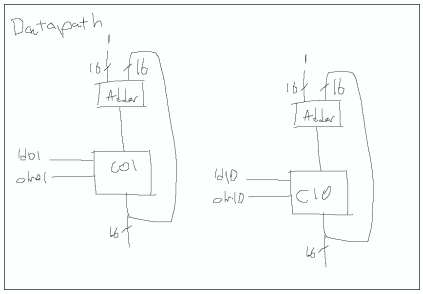


of 1's and 0's at the input and outputs 1 v puts 0. Draw the state transition diagram o ole for the FSM using binary state encodin

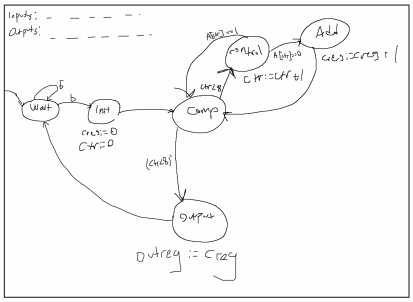
Q13. Design a system that counts t bit input B and always outputs on 16 are initially 0. An event is a 01 is 0 (C10 is incremented by 1). unt rc alue of either C10 or C01 is

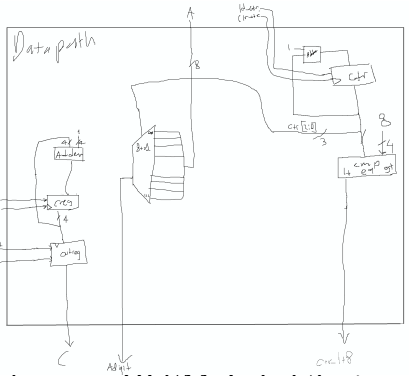


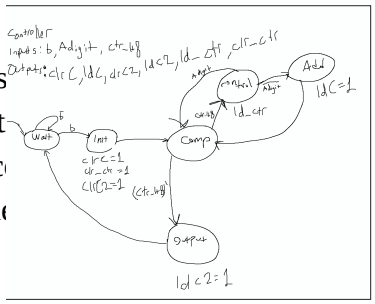
behavi transition diagram.



(b) Create a datapath for your HLSM. Label the input/output signals of all components.







nts it input **A**

nit ts when a

ec nd then it

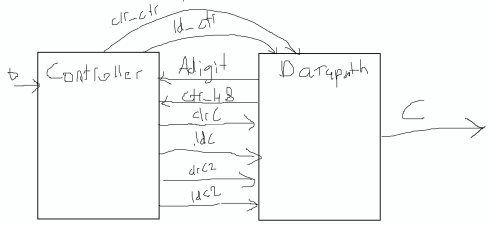
th b changes

w the state transition diagram

(b) Create a datapath for your HLSM. Label the inpu

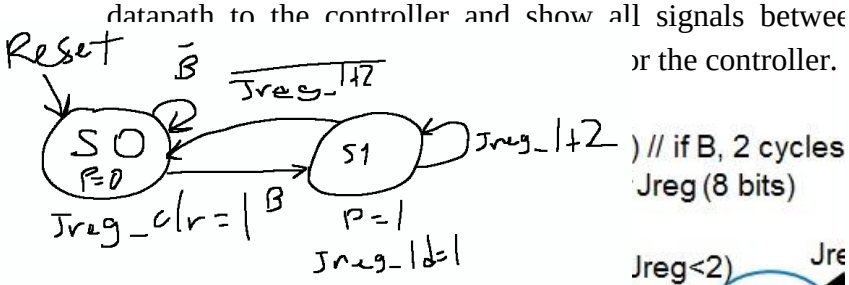
(c) Connect the datapath to the controller and show a

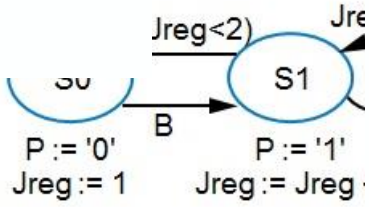
(d) Derive the FSM and draw the state transition diag

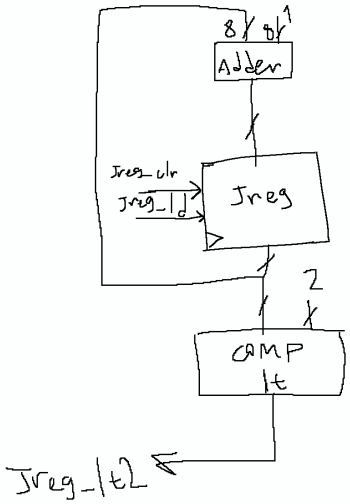


Q15. For the HLSM given below, complete the RTL design process. Create a datapath with each component clearly labeled and show the connections he datapath to the controller and show all signals betwee he

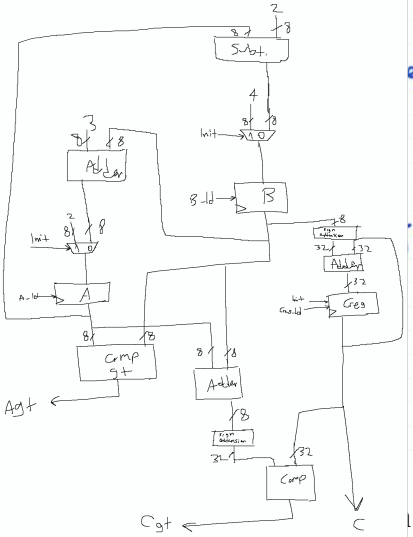
r the controller.

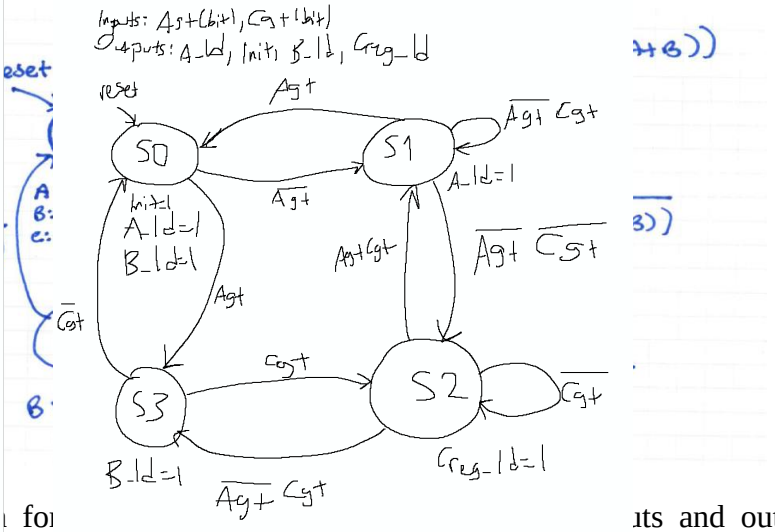






Q16. The state transition diagram for an HLSM is given below. The output **C** (32 bit) comes from a register of the same name (Creg). A and B are 8-bit local storage registers.



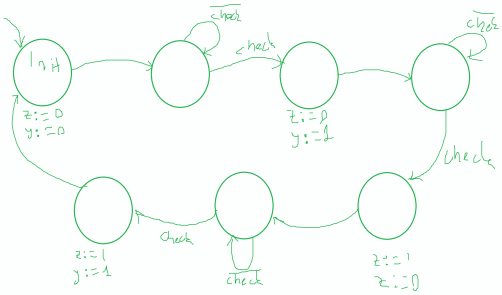


for its and outputs of the datapath. For each component in the datapath, clearly identify it to show its function, name its inputs an

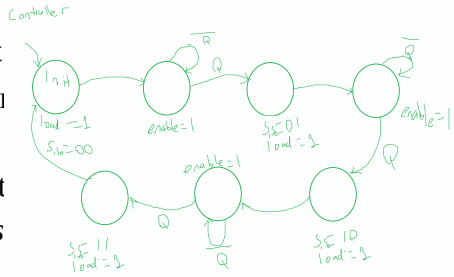
S1	S0	Agt	Cgt	S'1	S'0
0	0	0	x	0	1
0	0	1	x	1	1
0	1	0	0	1	0
0	1	0	1	0	1
0	1	1	x	0	0
1	0	0	1	1	1
1	0	x	0	1	0
1	0	1	1	0	1
1	1	x	0	0	0
1	1	x	1	1	0

(b) Derive the FSM and draw the state transition diagram. Write the state transition and output tables using binary state encodings (Hint: You can use a shorthand notation Agt for A > B and Cgt for C > A+B).

(c) Write the minimized Boolean equations for the next state and output of FSM.

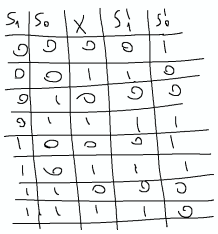


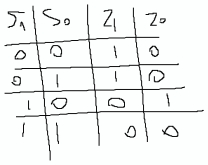
clock input **zy = 01** fo ock signal, also use a t *mber of bits*

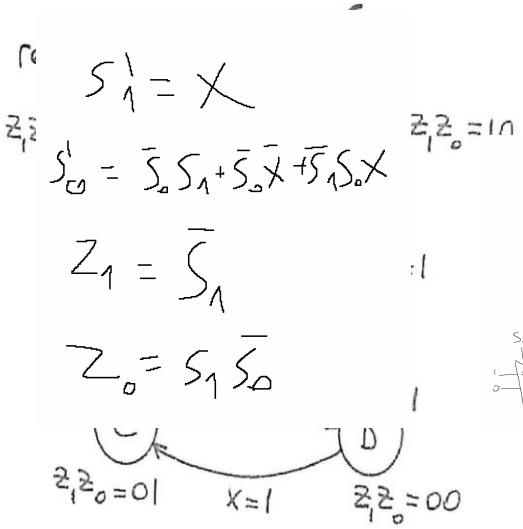


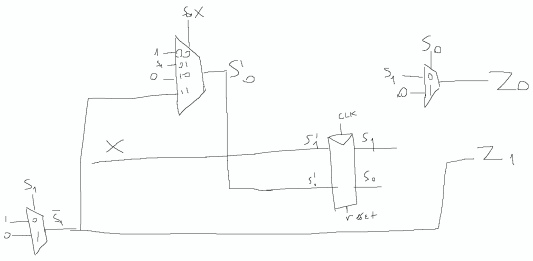
ate for ip- *by*

Q18. The state transition diagram of an FSM is given below. You will design the controller for this FSM using only two 4:1 multiplexers and two 2:1 multiplexers.







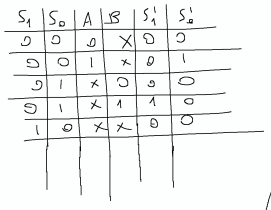


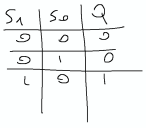
(a) Write the state transition table and output table for the FSM using binary state encodings.

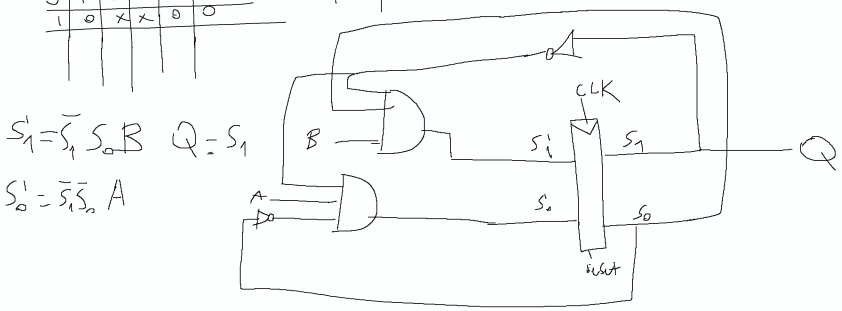
(b) Write the Boolean equations for the next state and output of FSM.

(c) Sketch the circuit schematic of FSM using state register, two 4:1 multiplexers, and two 2:1 multiplexers. Don't use any logic gate. You can use the logic levels 1 and 0 as input.

Q19. Design a Finite State Machine (FSM) using the state transiton diagram given below







(a) Using bina for the FSM. Write Bool

(b) Sketch a sc

Q20. A register file has 16 registers each having 16 bits. The registers RF[0] to RF[15] contain positive or negative integers. Design an HLSM which will find the sum of absolute values of the 16 integers stored in RF[0] to RF[15]. The negative numbers are in 2's complement form. The sum is going to be stored in a separate **SUM** register. Upon reset, the HLSM will be in the initial state waiting for the **GO** signal. When the operation is completed the HLSM will return to its initial state and wait for a GO signal to start over again. *Show your design clearly*.

(a) Capture the system behavior as an HLSM and draw the state transition diagram.

(b) Design the datapath for this HLSM. Specify each component in the datapath clearly (i.e. name of the component, input signals, output signals, number of bits, etc.).

(c) Connect the datapath to the controller (FSM) and show the signals between the controller and the datapath.

(d) Derive the FSM and draw the state transition diagram.