Logic Design ANS

Final Exam

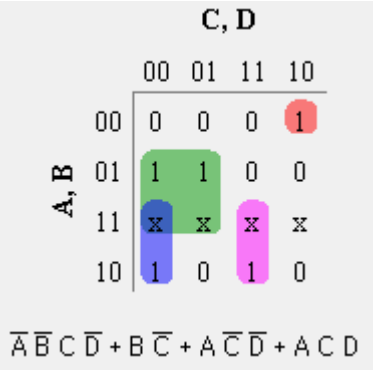
2017. 12. 6

ID : Name :

Problem 1
The logic under has 4-input and 1-ouput named A, B, C, D and F respectively. Find minimum Sum of Product form of output F.

A	B	C	D	F
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	X
1	1	0	1	X
1	1	1	0	X
1	1	1	1	X

Answer)



Problem 2 (QM method)

Use the Quine-McCluskey method to find the minimum sum-of-products form for the following truth table.

A	B	C	D	F
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	X
1	1	0	1	X
1	1	1	0	X
1	1	1	1	X

2.1 Show your process of deriving the prime implicants
2.2 Select a minimum set of prime implicants and find minimum-literal Sum-of-Product form

ANS) 2.1.

col1		col2		col3	
0V	0000	(0,1)	000-	(0,2,8,10)	-0-0
1V	0001	(0,2)V	00-0	(8,10,12,14)	1--0
2V	0010	(0,8)V	-000	(5,7,13,15)	-1-1
8V	1000	(1,5)V	0-01	(10,11,14,15)	1-1-
5V	0101	(2,10)V	-010	(12,13,14,15)	11--
10V	1010	(8,10)V	10-0		
12V	1100	(8,12)V	1-00		
7V	0111	(5,7)V	01-1		
11V	1011	(5,13)V	-101		
13V	1101	(10,11)V	101-		
14V	1110	(10,14)V	1-10		
15V	1111	(12,13)V	110-		
		(12,14)V	11-0		
		(7,15)V	-111		
		(11,15)V	1-11		
		(13,15)V	11-1		
		(14,15)V	111-		

PI: B'D', AD', BD, AC, AB, A'C'D

2.2.

Select a minimum set of PIs by using a PI chart ignoring don't cares.

		0	1	2	5	7	8	10	11
-0-0	B'D'	X		X			X	X	
1--0	AD'						X	X	
-1-1	BD				X	X			
1-1-	AC							X	X
11--	AB								
0-01	A'C'D		X		X				

Minimum-literal SOP form: A'C'D + AC + BD + B'D'

Problem 3 (Logic gates)

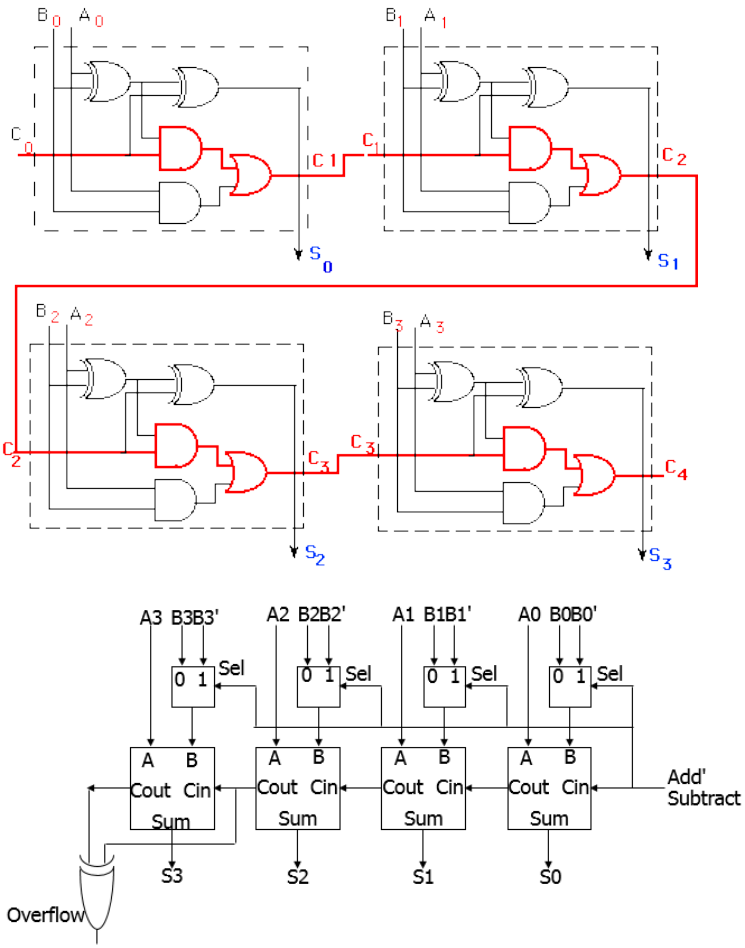
Draw the schematic diagram of the following with basic logic gates. You can use any of n-input logic gates, but do NOT use pre-built packages (Full Adder, Mux, etc.)

3.1. 4-bit ripple carry adder (addition, subtraction)

3.2. 4-bit carry lookahead adder

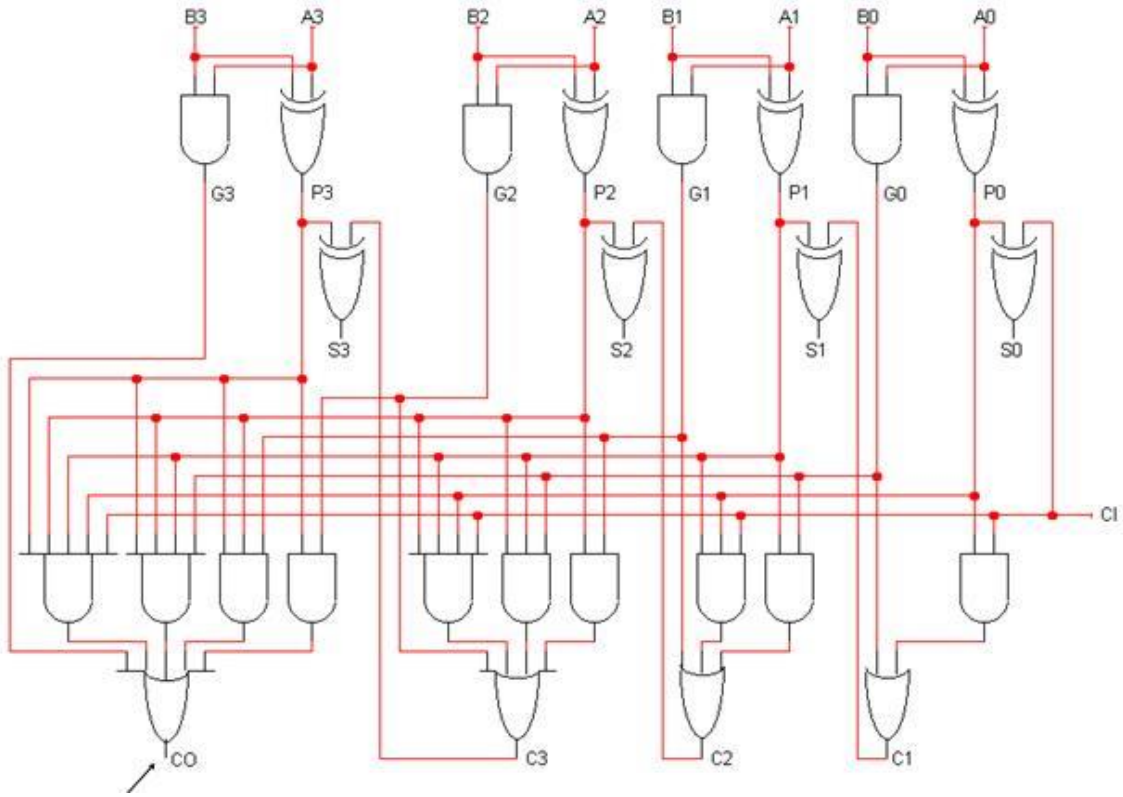
ANS)

3.1.



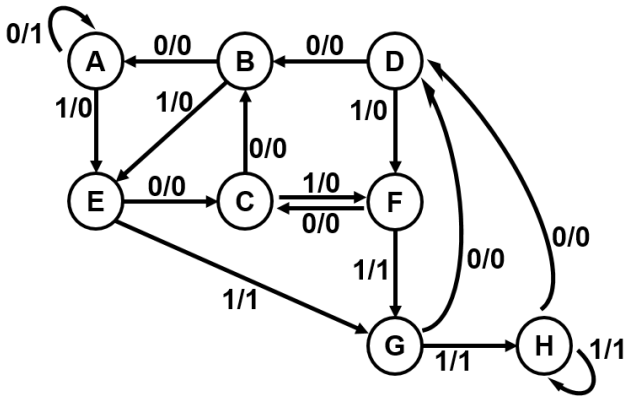
(with 1b FA and 2-1 Mux)

3.2.


B3
A3
B2
A2
B1
A1
B0
A0
G3
P3
G2
P2
G1
P1
G0
P0
S3
S2
S1
S0
CI
CO
C3
C2
C1

Problem 4. (State minimization)

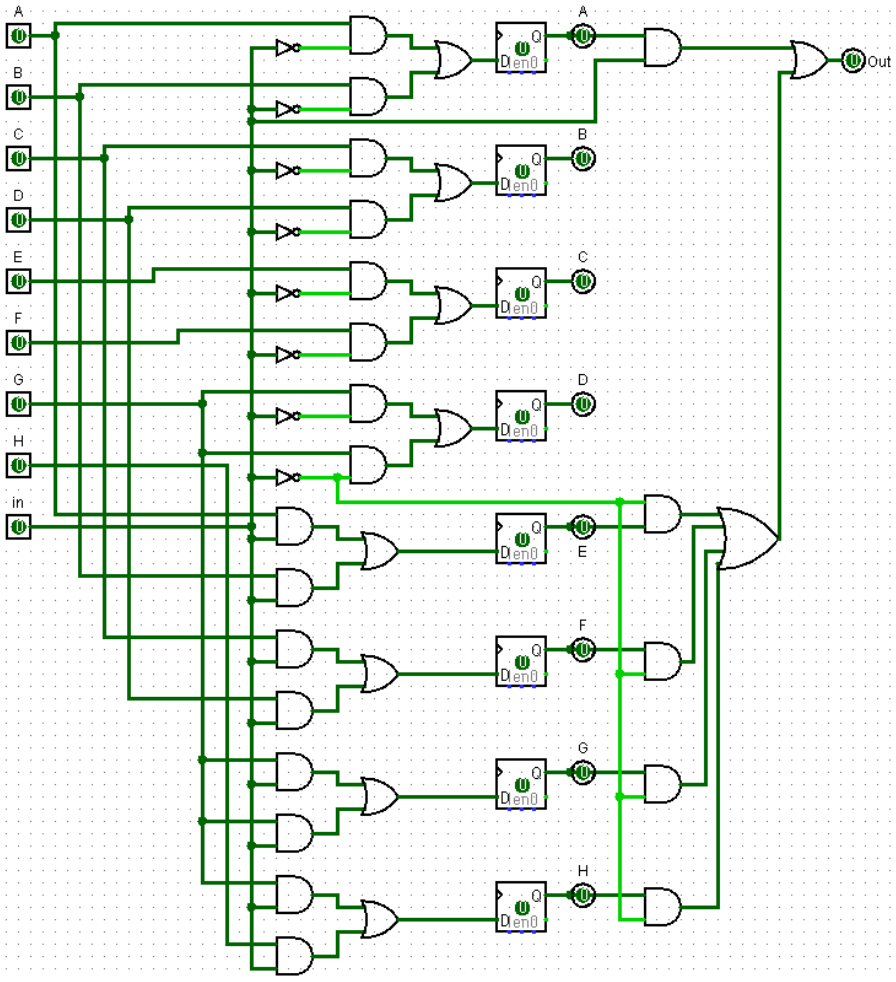
We can minimize the following finite state machine in terms of the number of states



4.1. Draw schematic diagram of the state diagram before the state minimization
4.2. Explain how to minimize the state and draw the state transition diagram after the state minimization
4.3. Draw schematic diagram of the state diagram after the state minimization
4.4. Compare the both schematic diagram in terms of the number of logic gates

Solution)

4.1.



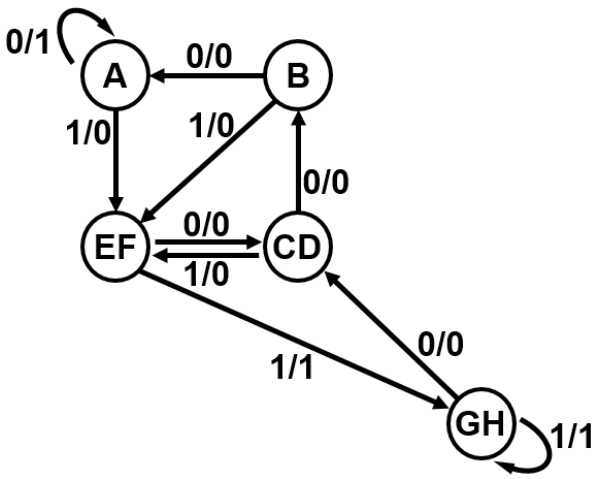
4.2. By Implication chart,

B	AA EE						
C	AB EF	AB EF					
D	AB EF	AB EF	BB FF				
E	AC EG	AC EG	BC FG	BC FG			
F	AC EG	AC EG	BC FG	BC FG	CC GG		
G	AD EH	AD EH	BD FH	BD FH	CD GH	CD GH	
H	AD EH	AD EH	BD FH	BD FH	CD GH	CD GH	DD HH
	A	B	C	D	E	F	G

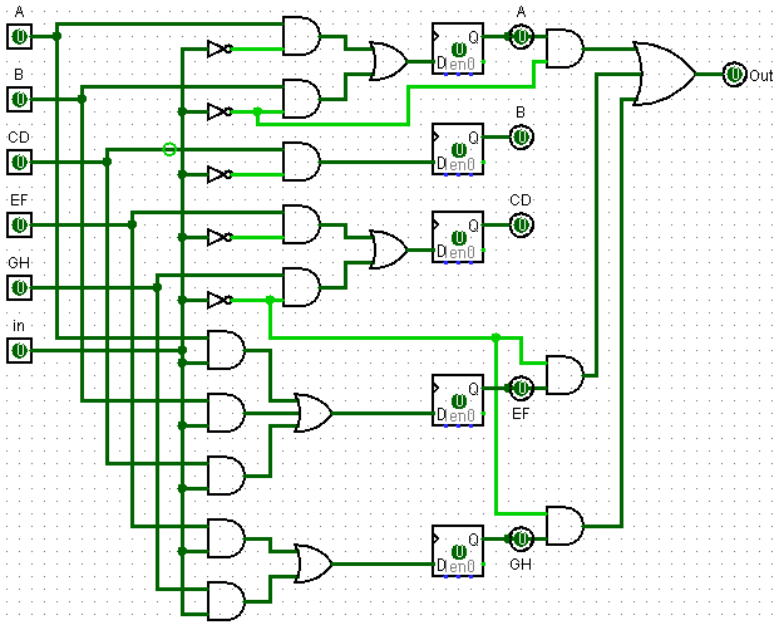
B	~~AA~~ ~~EE~~						
C	~~AB~~ ~~EF~~	~~AB~~ ~~EF~~					
D	~~AB~~ ~~EF~~	~~AB~~ ~~EF~~	BB FF				
E	~~AC~~ ~~EG~~	~~AC~~ ~~EG~~	BC FG	BC FG			
F	~~AC~~ ~~EG~~	~~AC~~ ~~EG~~	BC FG	BC FG	CC GG		
G	~~AD~~ ~~EH~~	~~AD~~ ~~EH~~	BD FH	BD FH	CD GH	CD GH	
H	~~AD~~ ~~EH~~	~~AD~~ ~~EH~~	BD FH	BD FH	CD GH	CD GH	DD HH
	A	B	C	D	E	F	G

B	~~AA~~ ~~EE~~						
C	~~AB~~ ~~EF~~	~~AB~~ ~~EF~~					
D	~~AB~~ ~~EF~~	~~AB~~ ~~EF~~	BB FF				
E	~~AC~~ ~~EG~~	~~AC~~ ~~EG~~	~~BC~~ ~~FG~~	~~BC~~ ~~FG~~			
F	~~AC~~ ~~EG~~	~~AC~~ ~~EG~~	~~BC~~ ~~FG~~	~~BC~~ ~~FG~~	CC GG		
G	~~AD~~ ~~EH~~	~~AD~~ ~~EH~~	~~BD~~ ~~FH~~	~~BD~~ ~~FH~~	CD GH	CD GH	
H	~~AD~~ ~~EH~~	~~AD~~ ~~EH~~	~~BD~~ ~~FH~~	~~BD~~ ~~FH~~	CD GH	CD GH	DD HH
	A	B	C	D	E	F	G

State C and D, E and F, G and H are can be combined



4.3.



4.4.

We can reduce 3 F/F by state minimization (8 → 5)

Problem 5 (Design Problem)

There is a simple queue(FIFO) structure. It has two operations, which are enqueue and dequeue. When enqueue signal is 1(enqueue), the queue stores a value. When dequeue signal is 1(dequeue), the queue removes and returns the oldest value among stored values.

Assume that you already have a queue whose capacity is 3. It can enqueue only if the queue is empty, or it has # of values less than 3. It can dequeue only if the queue is full, or it has equal or more than 1 value. In the other cases, enqueue or dequeue signals have no effect. Make a detector (not the queue itself) which defines the queue is full or not.

The details of the detector:

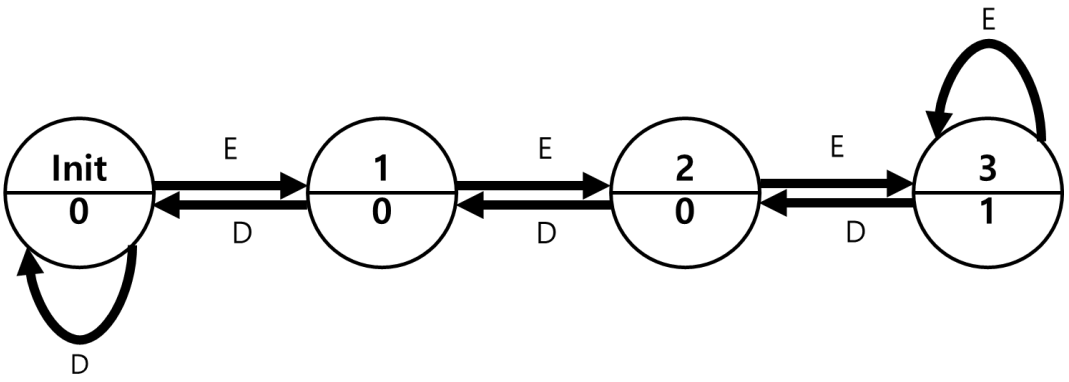
- **Inputs**: Enqueue signal of the queue(E), Dequeue signal of the queue(D)
- **Output**: Whether the queue is full or not(F)

5.1. Draw the state diagram of the detector

5.2. Make a gate-level logic of the detector.

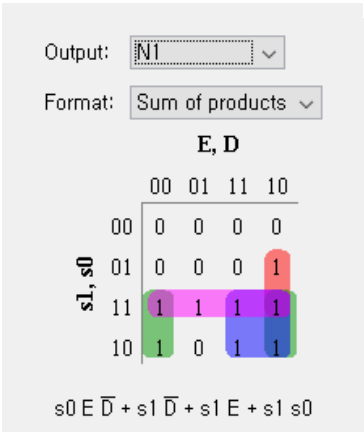
Answer)

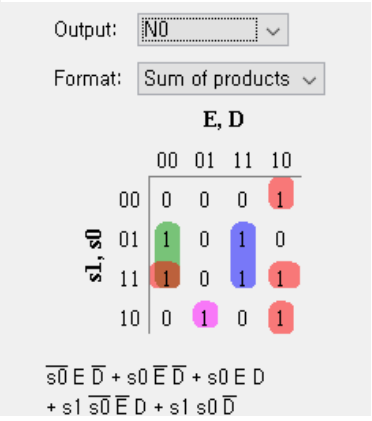
5.1. Moore machine

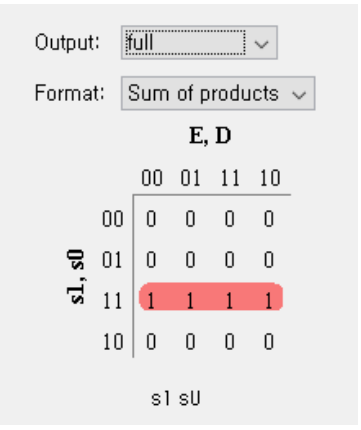


5.2.

s1	s0	E	D	N1	N0	full
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	0	1	0	0	1	0
0	0	1	1	0	0	0
0	1	0	0	0	1	0
0	1	0	1	0	0	0
0	1	1	0	1	0	0
0	1	1	1	0	1	0
1	0	0	0	1	0	0
1	0	0	1	0	1	0
1	0	1	0	1	1	0
1	0	1	1	1	0	0
1	1	0	0	1	1	1
1	1	0	1	1	0	1
1	1	1	0	1	1	1
1	1	1	1	1	1	1

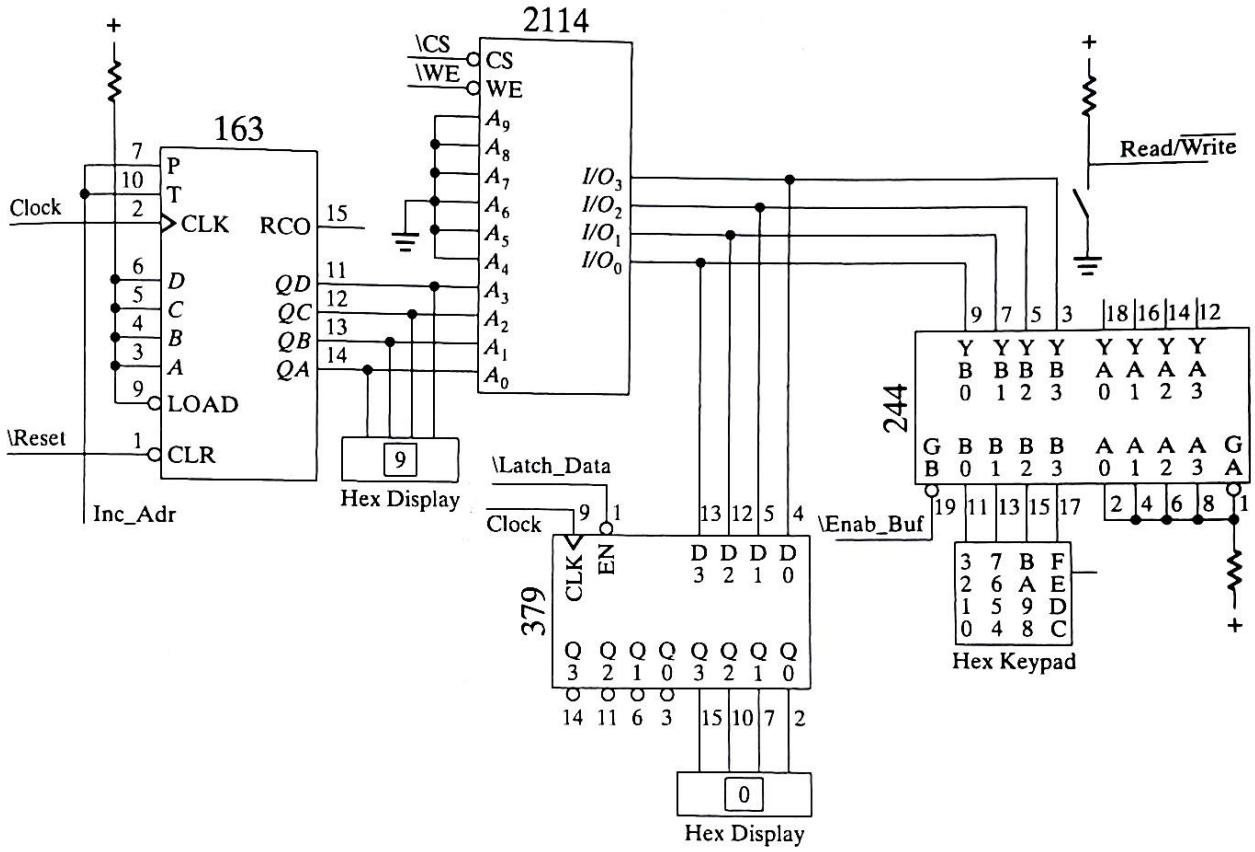




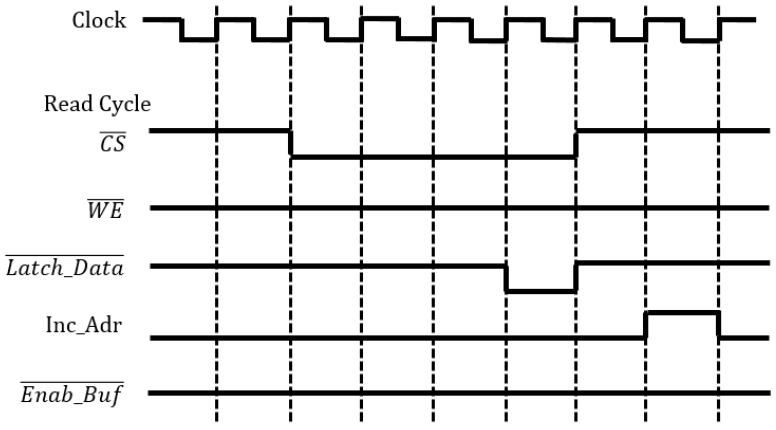


Problem 6 (Control signal)

The following figure shows a system consisting of a memory (2114), a counter (163), two buffers (379 and 244) and two Hex display modules.



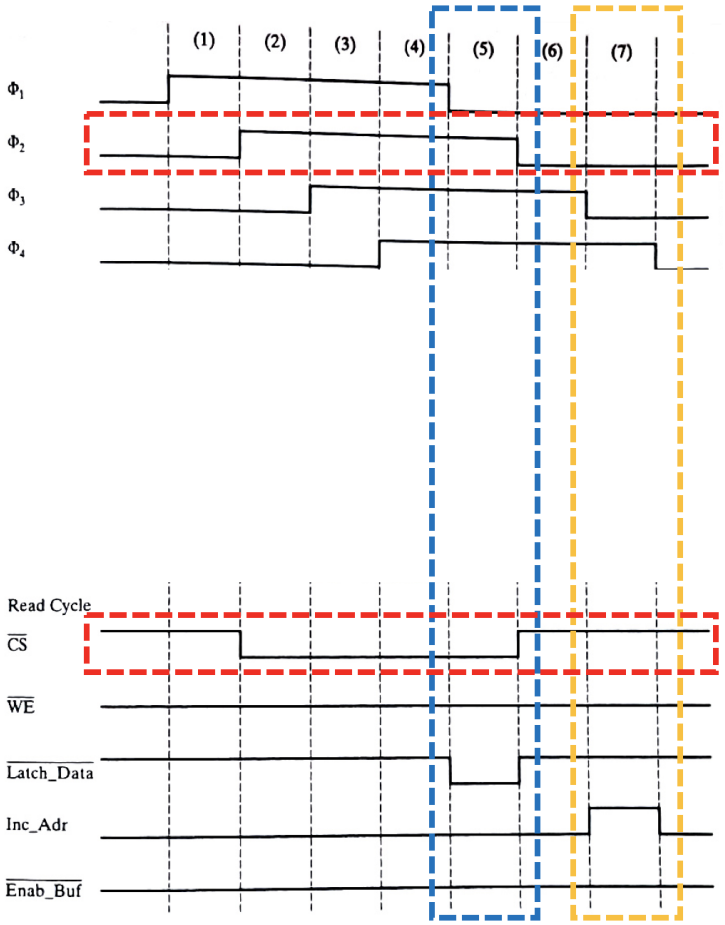
6.1. Explain how the system works by reading data from the memory. Use the following control signals to explain the working.

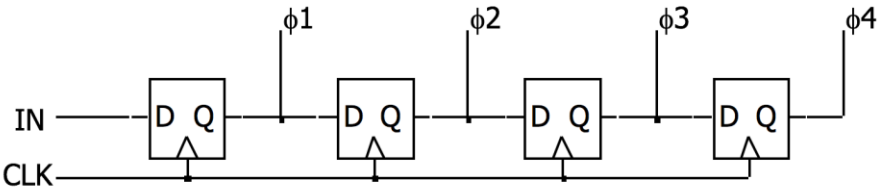


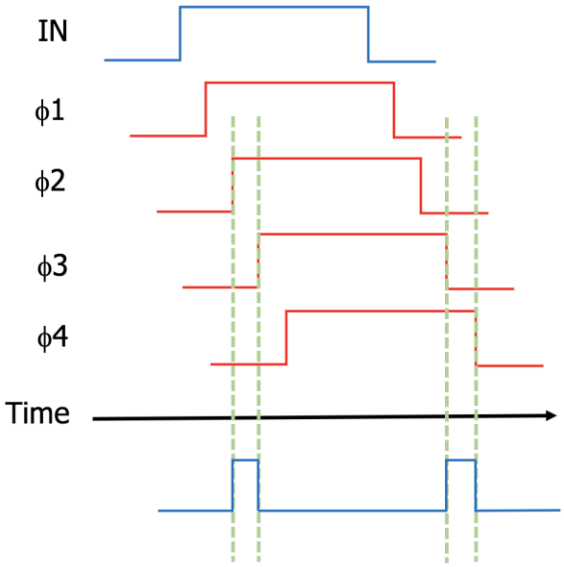
6.2. Explain how to generate the above signals, CS', Latch_data' and Inc_Adr.

Answer)

At time (or clock period) 2, CS' goes low to start read operation. WE' stays high to inform the memory of read operation. At time 5, Latch_Data' goes low to store the data read from the memory. The stored data is shown by the Hex display. At time 7, Inc_Adr goes high to increment the counter, which allows us to read the next data from the memory.



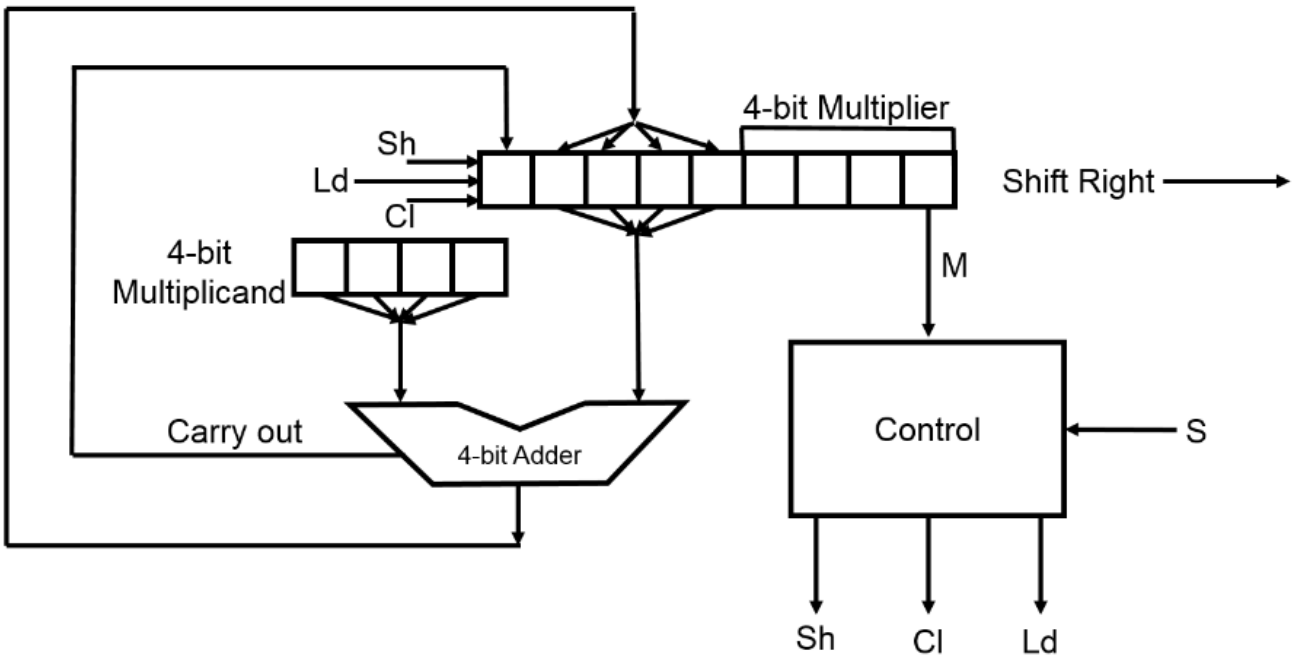




C = ϕ1ϕ2ϕ3′ϕ4′+ϕ1′ϕ2′ϕ3′ϕ4

Problem 7 (Design Problem)

The following figure is data path for sequential 4-bit multiplier. It consist of a 4-bit adder, a 4-bit register and a 9-bit shift register. The 9-bit shift register shifts right when its *Sh* input is asserted (assume that zeros are entered at the left for this operation). A new value is loaded into the high-order 5 bits of the shift register when *Ld* is asserted. The same 5 bits are zeroed when *Cl* is asserted. These signals are synchronous.



7.1. Draw a Mealy machine state diagram for a 4-bit multiplier. The inputs are S (a multiply start signal) and M (the LSB of the multiplier). The outputs are the Sh, Ld and Sl signals.

7.2. Implement the control part for a 4-bit multiplier with basic logic gates.

ANS)

7.1.