CSE1003-DIGITAL LOGIC DESIGN

Digital Assignment-2

(Due date 18th July 2021)

SNo	REGISTER NO	NAME	DA1 set No
1	20BCE2762	SANGAM RAI	A
2	20BCE2765	VIBHRAT VIBHU BASNET	B
3	20BCE2767	ANJAN KUMAR SAH	C
4	20BCE2773	YASH MISHRA	D
5	20BCE2775	PRINCE PANJIYAR	E
6	20BCE2781	KARMA SANGPO GURUNG	F
7	20BCE2782	RIYESH DUWAL SHRESTHA	G
8	20BCE2881	SAURAV SAH	H
9	20BCE2882	JEEWAN GHIMIRE	I
10	20BCE2883	ELINA PARAJULI	J
11	20BCE2884	SNEHA CHAUDHARY	K
12	20BCE2885	SUMIT KUMAR MANDAL	L
13	20BCE2886	NAMAN AGRAHARI	M
14	20BCE2887	SUNIL KUSHWAHA	N
15	20BCE2888	MUDIT BHATTA	O
16	20BCE2891	ADITYA KUNWAR	P
17	20BCE2892	MD.ANWAR MANSURI	Q
18	20BCE2893	ANISH SHRESTHA	R
19	20BCE2894	ASHUTOSH BASHYAL	S
20	20BCE2895	ROHIT KUMAR GUPTA	T
21	20BCE2896	PRAMIT KARKI	A
22	20BCE2897	PRATIK LUITEL	B
23	20BCE2898	SANDESH KHATIWADA	C
24	20BCE2899	SHREYA KARKI	D
25	20BCE2900	PAWAN THAKUR	E
26	20BCE2901	PRAJWAL LAMSAL	F
27	20BCE2902	BINIT BHATTARAI	G
28	20BCE2903	AAYUSH MANDAL	H
29	20BCE2904	BIJAN SHRESTHA	I
30	20BCE2905	ABHAY RATHI	J
31	20BCE2906	ASHWIN POUDEL	K
32	20BCE2907	ANURAG KARKI	L
33	20BCE2908	MUSKAN SAH	M
34	20BCE2909	KRISHNA KUMAR RAUT	N
35	20BCE2911	PRARTHANA SHIWAKOTI	O
36	20BCE2913	SARTHAK GIRI	P
37	20BCE2914	BISHNU ROUNIYAR	Q
38	20BCE2915	GAURAB KUMAR RAUNIYAR	R

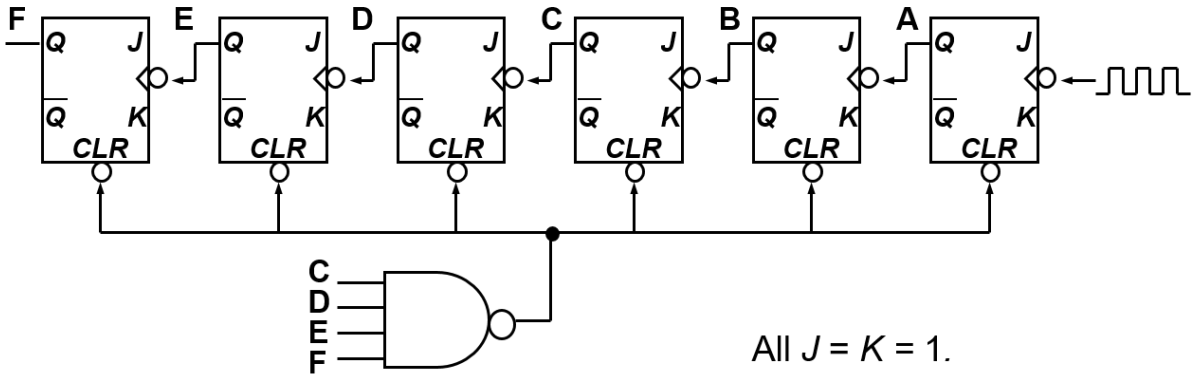
39	20BCE2916	AMAN SINGH	S
40	20BCE2917	YASHWANT BHARDWAJ	T
41	20BCE2918	AVIRAL SHARMA	A
42	20BCE2919	SUDARSHAN BHATTA	B
43	20BCE2920	DIWESH CHAURASIA	C
44	20BCE2921	PRITHAK GAJUREL	D
45	20BCE2922	SARJAK DEVKOTA	E
46	20BCE2923	BISURAJ SHARMA	F
47	20BCE2924	JAYESH PERIWAL	G
48	20BCE2925	LIJAH BABU GONGAL	H
49	20BCE2926	GAUTAM KUMAR MAHATO	I
50	20BCE2927	ADITYA KUMAR SINGH	J
51	20BCE2928	VIVAN SHRESTHA	K
52	20BCE2929	PRIYANSHU KOIRALA	L
53	20BCE2930	AANAND RIMAL	M
54	20BCE2931	PRASHAMAN POKHAREL	N
55	20BCE2932	SAKSHI AGRAWAL	O
56	20BCE2933	KARMA GURUNG	P
57	20BCE2934	ANUJ MISHRA	Q
58	20BCE2935	SHAILAJ GAUTAM	R
59	20BCE2936	SIDDHANT KARKI	S
60	20BCI0328	ANMOL GURAGAIN	T
61	20BCI0329	ABNISHA PAUDEL	A
62	20BCI0330	SHREEMA GAUTAM	B
63	20BCI0331	MD TANUWAR ANJUM	C
64	20BDS0405	BIMAL PARAJULI	D
65	20BDS0406	AADITYA BHETUWAL	E

SET-A

1. Draw a state diagram for a Moore type state machine specified as follows:

- Denote "a" the initial state of the machine set the initial output to the initial input (the input is presented to the machine bit by bit).The output changes value only when three successive inputs have the same value and that value is opposite to the current output. For example, if the current output is 1 and the machine detects three consecutive 0 it changes the output to 0. design the circuit using RS – FF and write the verilog HDL.

2. Design the 4 bit 7311 code to 53-1-1 code and write the Verilog HDL using case statement.
3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? Counter?

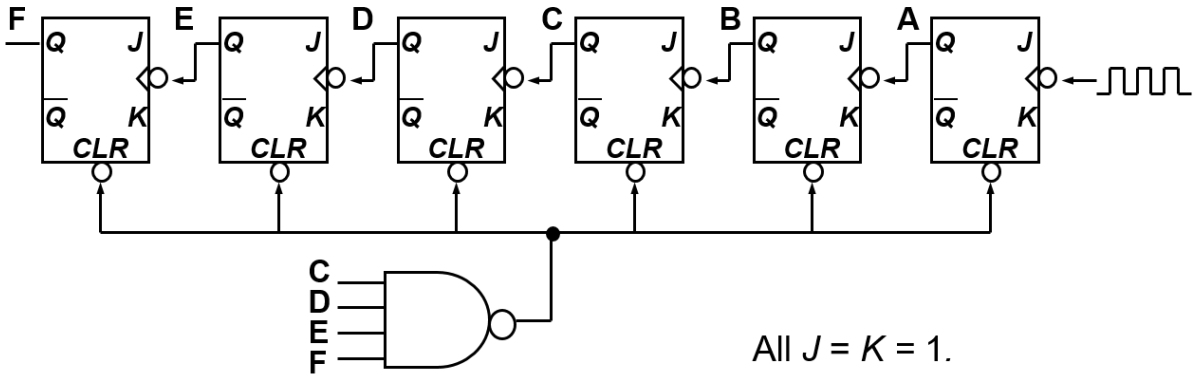


SET-B

1. Design a finite state machine (FSM) for a counter that counts through the 3-bit prime numbers downwards. Assume the counter starts with initial prime value set to 010 as its first 3 bit prime number.

You need to provide the state transition table and the state transition diagram. Assume that the state is stored in three D-FFs. Write the HDL code and simulate Hint: The set of all 3-bit prime numbers includes 2, 3, 5 and7.

2. Gray codes have the useful property that consecutive numbers differ in only a single bit osition. Design a 3-bit Gray code counter FSM with no inputs and three outputs. When reset, the output should be 000. On each clock edge, the output should advance to the next Gray code. After reaching 100, it should repeat with 000. Draw a schematic for this counter using T flip-flops.
3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? Counter?

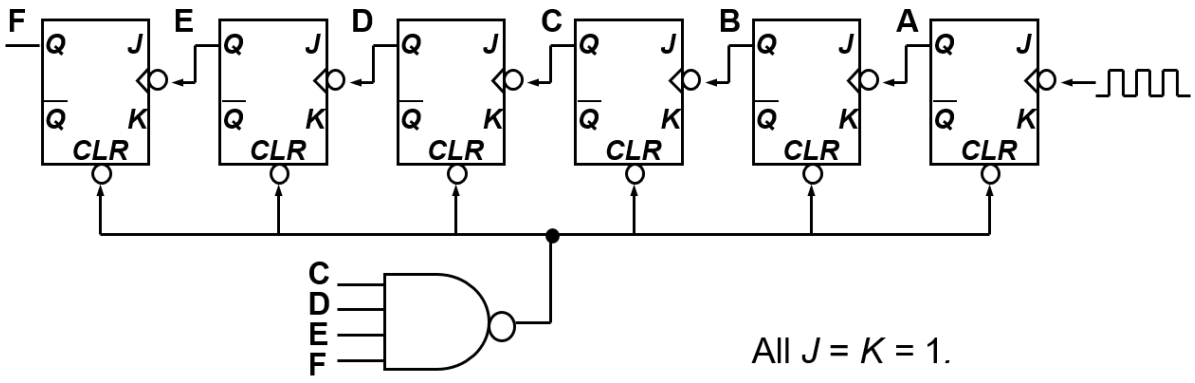


SET-C

1. Design a home security system as follows. There are 3 input sensors called, *e* (enable), *w* (window) and *d* (door). There is one output called *s* (alarm sound). After reset the system is disarmed. The system stays in the disarmed state until *e* is pressed, when it will move to the armed state. In this state, if *e* is pressed again the system will move back to the disarmed state. If however, *w* or *d* sensors are sensed while the system is in the armed state, the system will activate the alarm sound as follows. Signal *s* will be turned on for 500 clock cycles then it will shut down for 300 clock cycles. This sequence will repeat itself until *e* becomes 1, in which case the system returns to the disarmed state. Derive the data-path elements and the FSM in the control-path. Write/simulate the Verilog code to verify the entire design.

2. An up-down mod-4 binary counter has a single input x, such that the it counts up if x=0 and counts down if x=1:
   * Draw the Mealy state diagram of the circuit.
   * Draw the state Moore diagram of the circuit.
   *Implement both circuits using *negative-edge triggered* D-FFs.

3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? Counter?

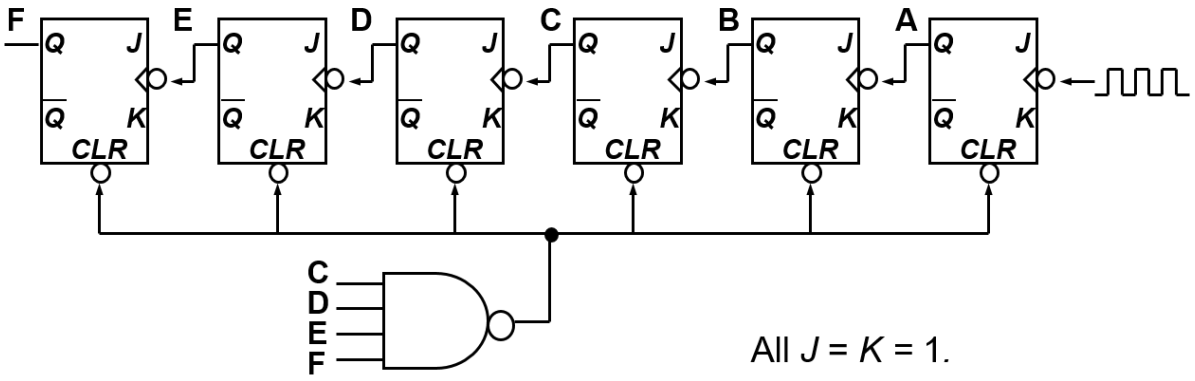


SET-D

1. Design a sequential circuit using RS – FF that has 1 data input (*w*) and 1 data output (*z*). The output *z* will become 1 if in the last 3 clock cycles the number of 1s on the input *w* has been greater than 1. Draw the FSM diagram and write/simulate the Verilog code to verify it.

2. Design the 4 bit 2421 code to 53-1-1 code and write the Verilog HDL using case statement.

3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? Counter?

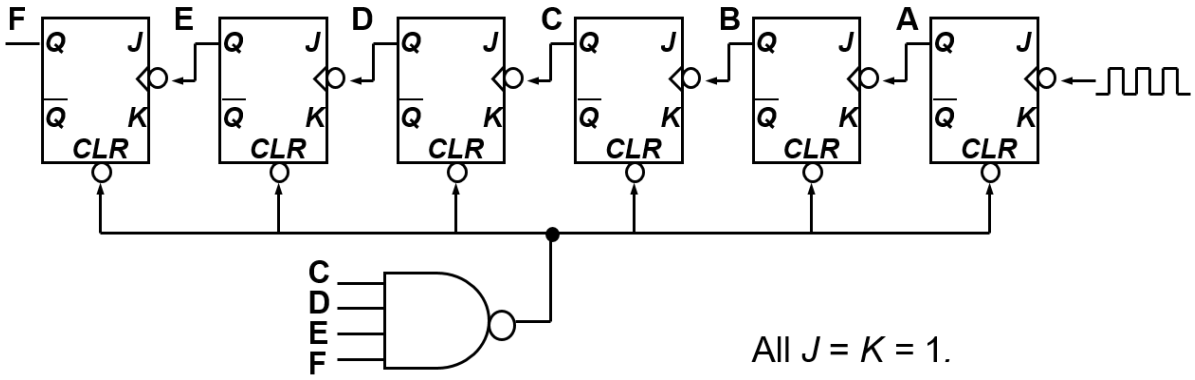


SET-E

1. Given the enclosed source code, draw the equivalent hardware circuit.

```
module setr(input logic clock,
output logic[7:0] f);
logic[7:0] a, b, c;
always_ff @(posedge clock)
begin
a <= b + c;
b = c + a;
c = a + b;
end
assign f = c;
endmodule
```

2. Draw the Moore state diagram and state table for a serial even parity checker. The circuit receives a word of 4-bits serially on its single input X and produces the even parity bit after the fourth bit is received. The single output Z remains 0 except when the final (fourth) bit is received and the total number of 1's in the word is odd. The machine returns to the reset initial state after the 4th input bit. Design the circuit using RS – FF and write the code using gate level and dataflow.
3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? Counter?



SET-F

1. Given the enclosed source code, draw the equivalent hardware circuit.

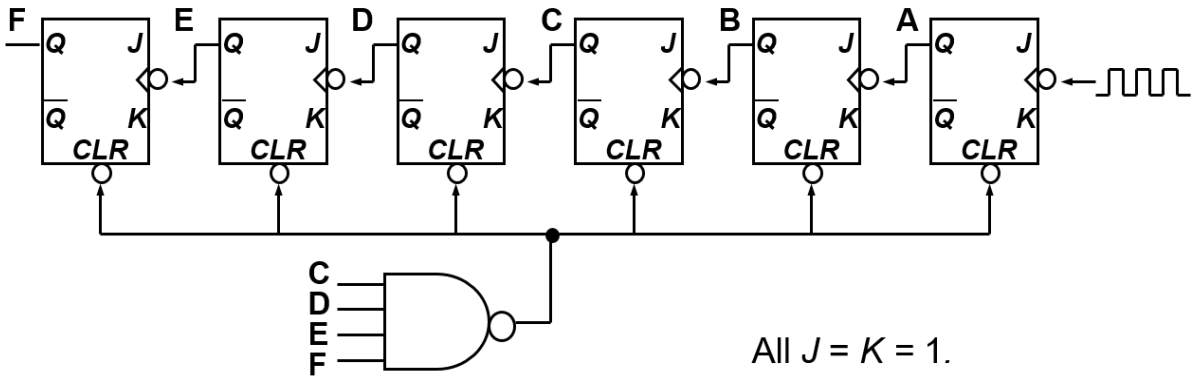
```
module setq(input logic clock,
output logic[7:0] f);
logic[7:0] a, b, c;
always_ff @(posedge clock)
begin
a = b + c;
b <= c + a;
c = a + b;
end
assign f = c;
endmodule
```

2. Draw a state diagram for a mealy type state machine specified as follows:

- Denote "a" the initial state of the machine
- set the initial output to the initial input (the input is presented to the machine bit by bit)
- The output changes value only when three successive inputs have the same value and that value is opposite to the current output. For example, if the current output is 1 and the machine detects three consecutive 0 it changes the output to 0.

design the circuit using T – FF and write the verilog HDL.

3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? Counter?



SET-G

1. Given the enclosed source code, draw the equivalent hardware circuit.

```
module setp(input logic c1, c2, c3, a, b,
output logic f, g);
always_latch
begin
if (c1) begin
f = a;
g = b;
end else if (c2) begin
f = b;
end else if (c3) begin
g = a;
end
```

```
end
endmodule
```

2. Design a 1-bit decrementer (a circuit which subtracts 1). Give
   (a) the arithmetic relationship between input and output signals

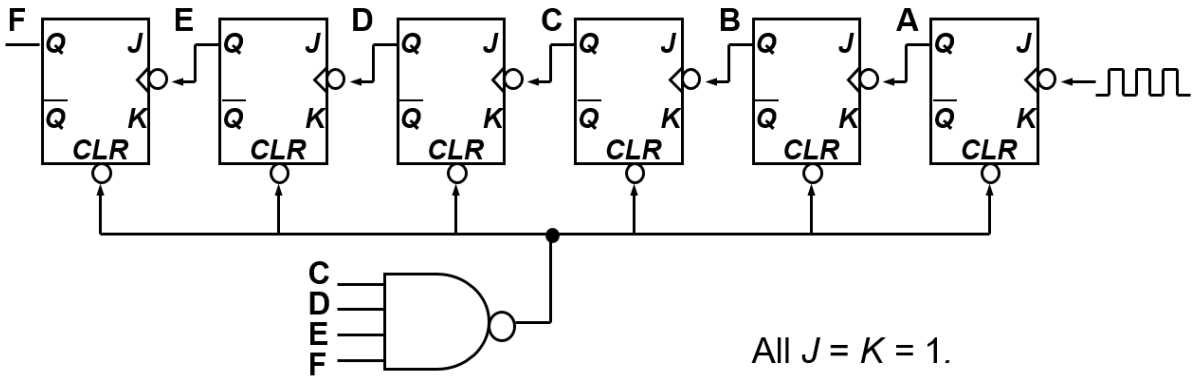
(b) the truth table,
(c) logic diagram.

Design a logic diagram of a 1-bit increment/decrement circuit controlled by an id signal (increment when id = 1, decrement otherwise). Write the HDL.

And Design a block diagram of a 4-bit increment/decrement circuit. Use the above HDL and concatenate.

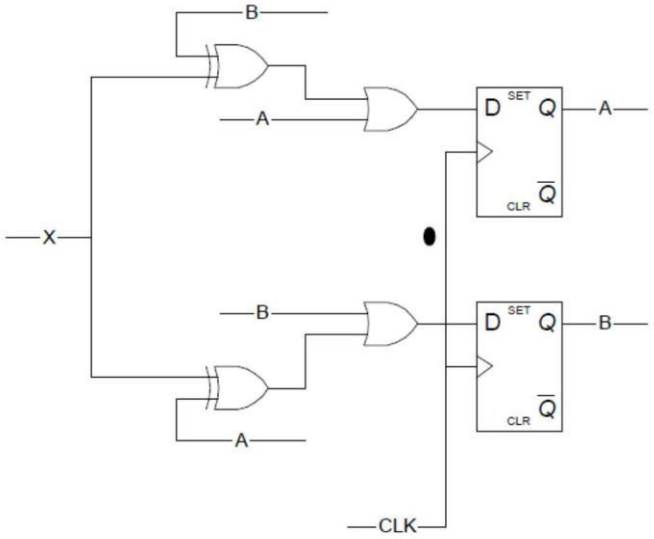
3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?

4. The following is a MOD-? Counter?

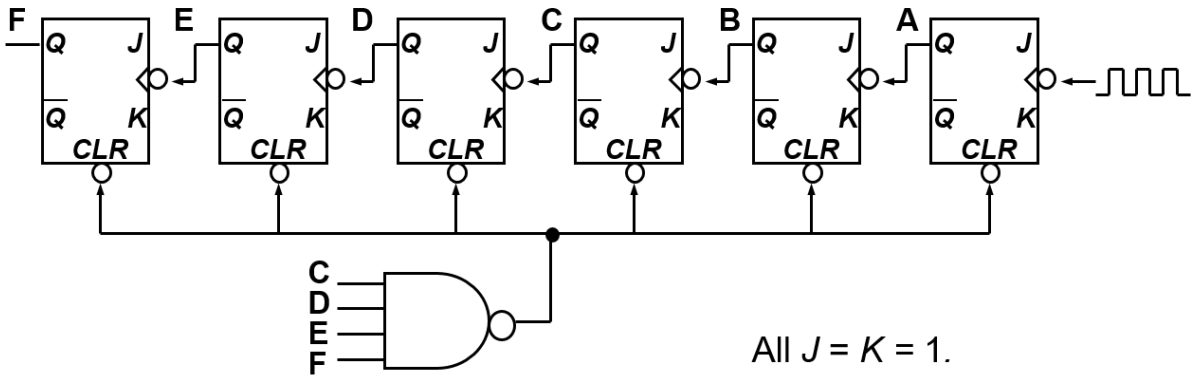


SET-H

1. Design a serial subtractor with accumulator for 5-bit binary numbers. Assume that negative numbers are represented by 2's complement. Use a circuit of the form of Figure 18-1, except implement a serial subtractor using a D-CE flip-flop and any kind of gates. Give the state graph for the control circuit. Assume that *St* will remain 1 until the subtraction is complete, and the circuit will not reset until *St* returns to 0.write the HDL using behavioural level.
2. Derive the state table and state diagram of the synchronous sequential circuit shown (X is an input to the circuit). Explain the circuit function.



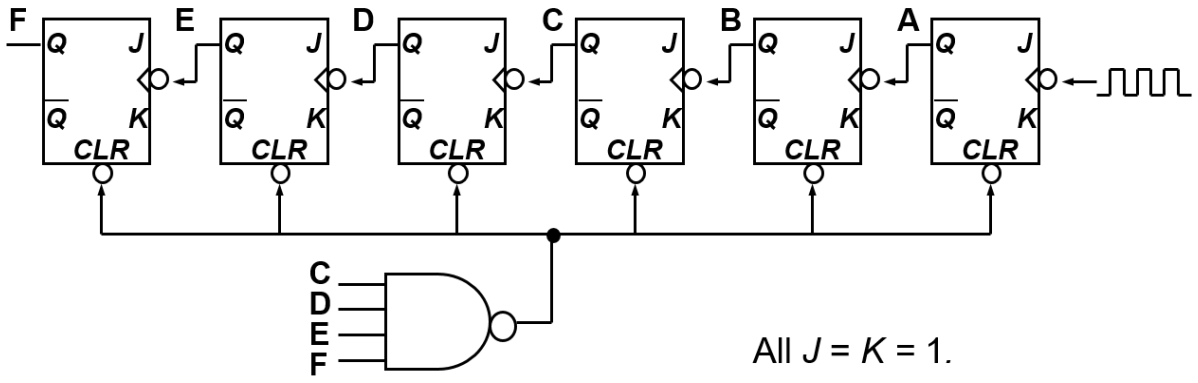
3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? Counter?



SET-I

1. Design and write the Verilog HDL for 4 bit parallel multiplier.
2. Draw the Mealy state diagram and state table for a serial even parity checker. The circuit receives a word of 4-bits serially on its single input X and produces the even parity bit after the fourth bit is received. The single output Z remains 0 except when the final (fourth) bit is received and the total number of 1's in the word is odd. The machine returns to the reset initial state after the 4th input bit. Design the circuit using D – FF and write the code using gate level and dataflow.

3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? Counter?

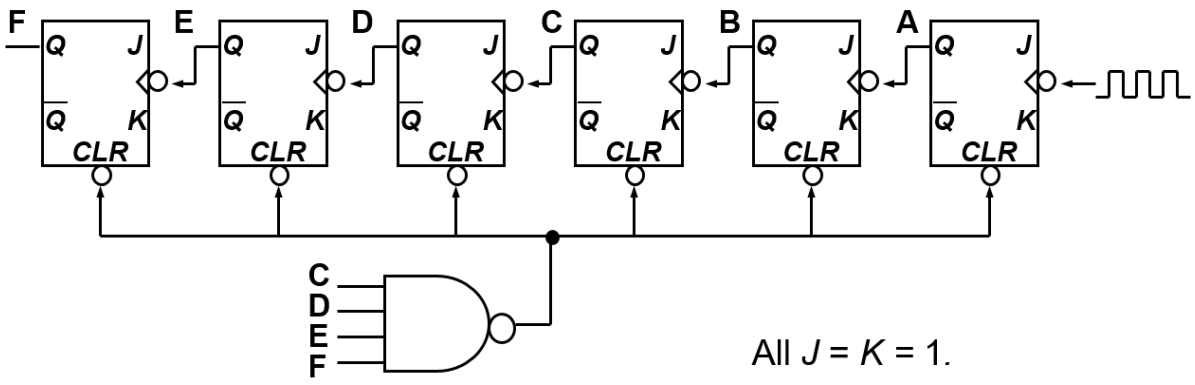


SET-J

1. Given the enclosed source code, draw the equivalent hardware circuit. Draw also the periodic waveforms on the output signals *f* and *g*. If the reference clock is 200 MHz, then what are the frequencies of signals *f* and *g* ?

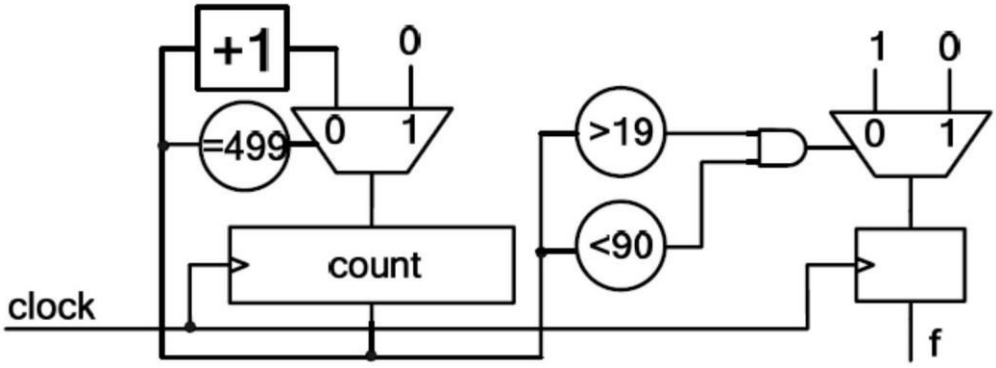
```
module setm(input logic resetn, clock,
output logic f, g);
logic[9:0] counter;
always_ff @(posedge clock or negedge
resetn) if (!resetn) begin
counter <=
10'h000; f <=
1'b0;
g <= 1'b0;
end else begin
f <= 1'b1;
if (counter > 10'd50 && counter < 10'd250) f <=
1'b0; g <= 1'b0;
if (counter > 10'd150 && counter < 10'd400) g <=
1'b1; if (counter < 10'd900) counter <= counter +
10'd1;
else counter <= 10'd0;
end
endmodule
```

2. Design a sequential circuit using D - FF to convert excess-3 code to BCD. This circuit adds three to a binary-coded-decimal digit in the range 0 to 9.The input and output will be serial with the least significant bit first.
3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? Counter?

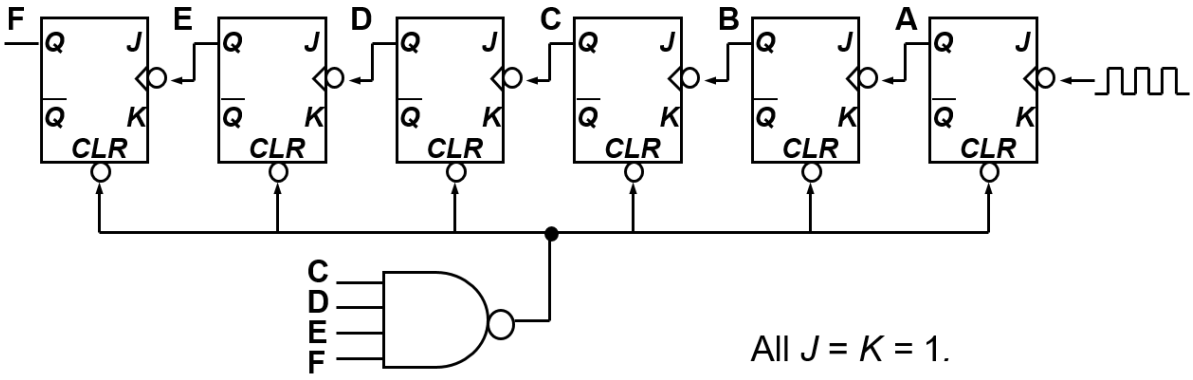


SET-K

1. Given the enclosed hardware circuit, write/simulate the equivalent Verilog code.

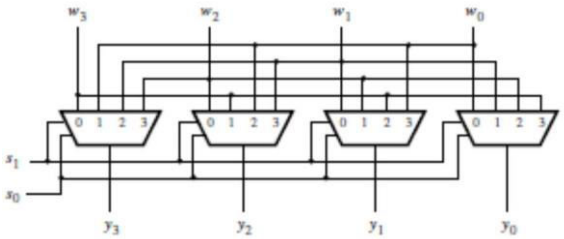


2. Design a sequential circuit using T - FF to convert BCD to excess-3 code. This circuit adds three to a binary-coded-decimal digit in the range 0 to 9.The input and output will be serial with the least significant bit first.
3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? Counter?

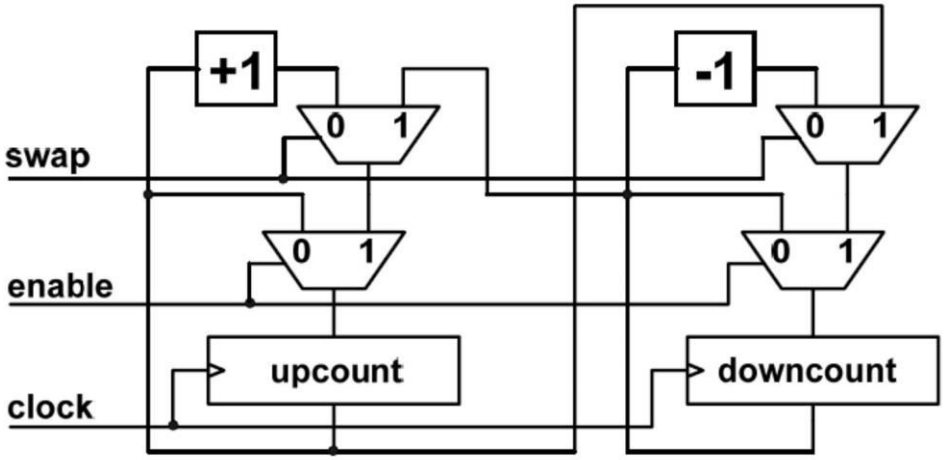


SET-L

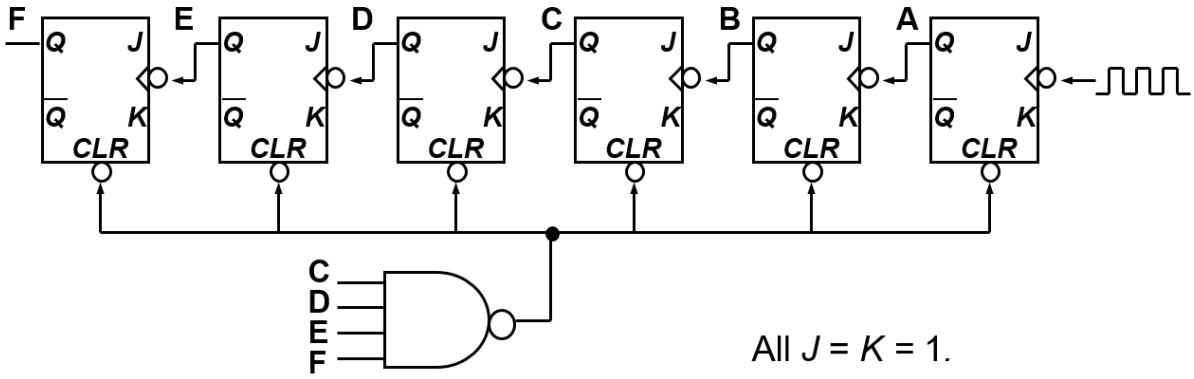
1. Write the Verilog code using if else condition and truth table for the circuit given below.



2. Given the enclosed hardware circuit, write/simulate the equivalent Verilog code.



3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? Counter?



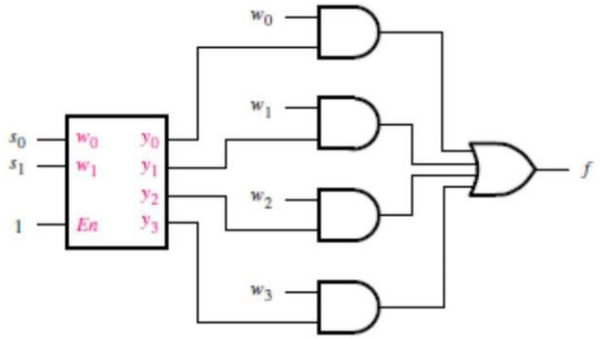
SET-M

1. For the given code give the hardware equivalent circuit and convert the code using case statement.

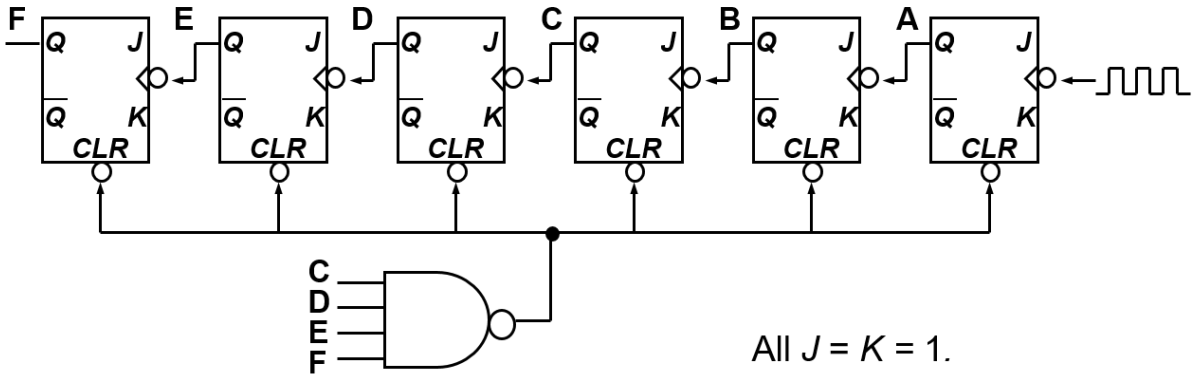
```
module setj(input logic resetn, clock,
input logic start, stop,
input logic load, serial_in,
input logic [7:0] parallel_in,
output logic parity);
logic shift_left;
logic[7:0] shift_register;
```

```
always_ff @(posedge clock or negedge resetn)
begin
if (!resetn) begin
shift_left <= 1'b0;
shift_register <= 8'd0;
    end else begin
    if (start) shift_left <= 1'b1; if
    (stop) shift_left <= 1'b0;
    if (shift_left) shift_register <= {shift_register[6:0],serial_in}; else
    if (load) shift_register <= parallel_in;
    end end
    always_comb begin parity
    = ^shift_register; end
    endmodule
```

2. Write Verilog code that represents the circuit in Figure below. Use the *dec2to4* module as a subcircuit in your code using case statement.



3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? Counter?



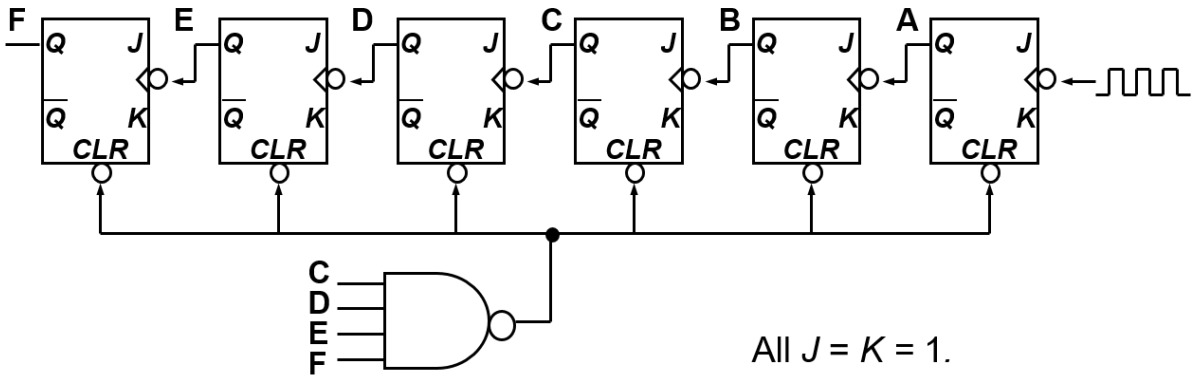
SET-N

1. For the given code give the hardware equivalent circuit and convert the code using case statement

```
module problem(input logic resetn, clock,
input logic x, y, z, c1, c2,
output logic f, g);
always_ff @(posedge clock or negedge resetn)
begin
```

```
if (!resetn) begin f <=
1'b0;
end else begin f <= x;
if (c1) f <= y; if (c2)
f <= z; end
end
always_comb begin if (c2)
g = z;
else if (c1) g = y; else g =
x;
end endmodule
```

2. Design a 4 bit serial adder with accumulator and write the Verilog HDL code.
3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? Counter?

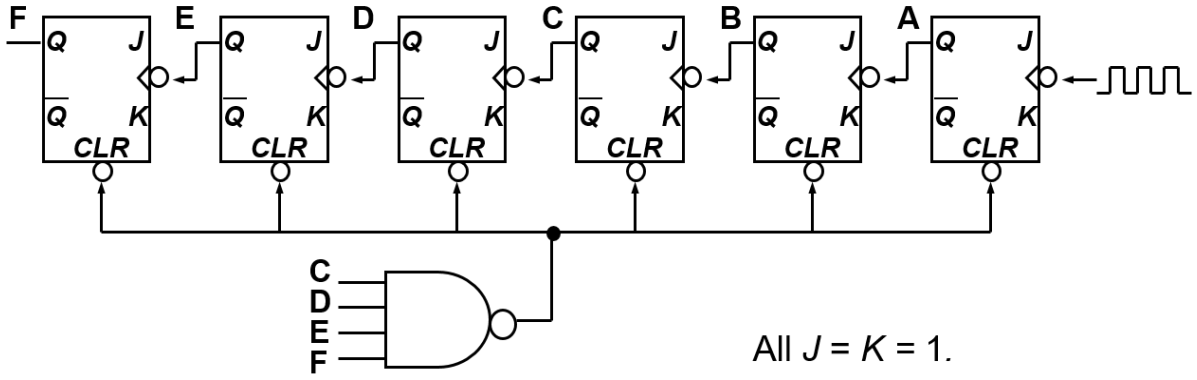


SET-O

1. Design a 1-bit decrementer (a circuit which subtracts 1). Give
(a) the arithmetic relationship between input and output signals
(b) the truth table,
(c) logic diagram.
Design a logic diagram of a 1-bit increment/decrement circuit controled by an id signal (increment when id = 1, decrement otherwise). Write the HDL.
And Design a block diagram of a 4-bit increment/decrement circuit. Use the above HDL and concatenate.
2. Design a sequential circuit that works as a traffic light controller. The circuit does not have any data inputs (the clock and asynchronous reset are obviously available) and it has three data outputs: *Red*, *Green* and *Yellow*. After reset the circuit will activate *Red* for 200 clock

cycles, then it will activate *Green* for 200 clock cycles and then it will activate *Yellow* for 20 clock cycles (this sequence will repeat itself). While one output signal is turned on the other two output signals are deactivated. Derive the data-path elements and the FSM in the control-path. Write/simulate the Verilog code to verify the entire design.

3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? Counter?



SET-P

1 You need to design a synchronous sequential circuit in the form of a positive edge-triggered Moore machine using RS - FF. The input signal *w* is synchronized with the clock pulses *C*. The output signal *z* should become 1 each time the value of the input signal *w* had not changed for two clock pulses. This change in the output value will appear at the clock pulse following the two pulses with the identical *w* values. See the example below for clarification.

w: 0011000110111001111111000010110 ...
z: 0010101001001001010000001000001 ...
Set up the circuit's state table and draw the state diagram

2. As an example of an unstructured combinational circuit design a circuit that divides a 3-bit positive binary number a = $(a_2a_1a_0)_2$ by a 2-bit positive binary number b = $(b_1b_0)_2$ calculating a 3-bit quotient q = $(q_2q_1q_0)_2$ and a 2-bit remainder r = $(r_1r_0)2$ so that

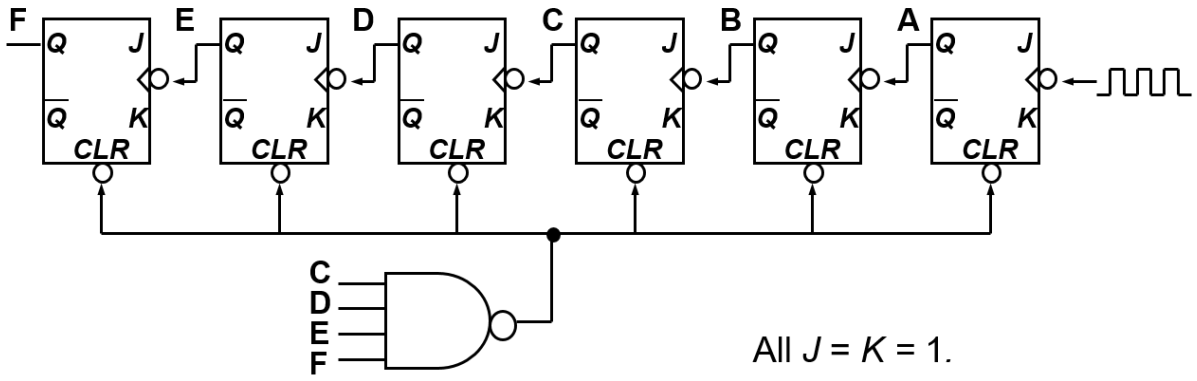
$$\frac{a}{b} = q + \frac{r}{b}$$

To make the task easier, take into account the following:

b	q_2	q_1	q_0	r_1	r_0
0	—	—	—	—	—
1	a_2	a_1	a_0	0	0
2	0	a_2	a_1	0	a_0

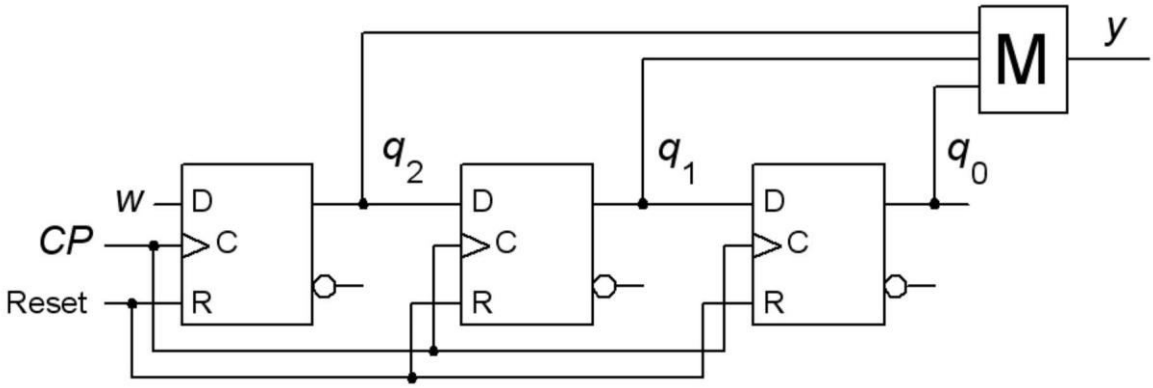
a. Complete the above table.
b. Derive logic equations for q and r. Aim at minimal implementation. Use Karnaugh maps when it makes the job easier.
c. Draw logic diagrams and write the HDL code using gate level and dataflow.

3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? Counter?



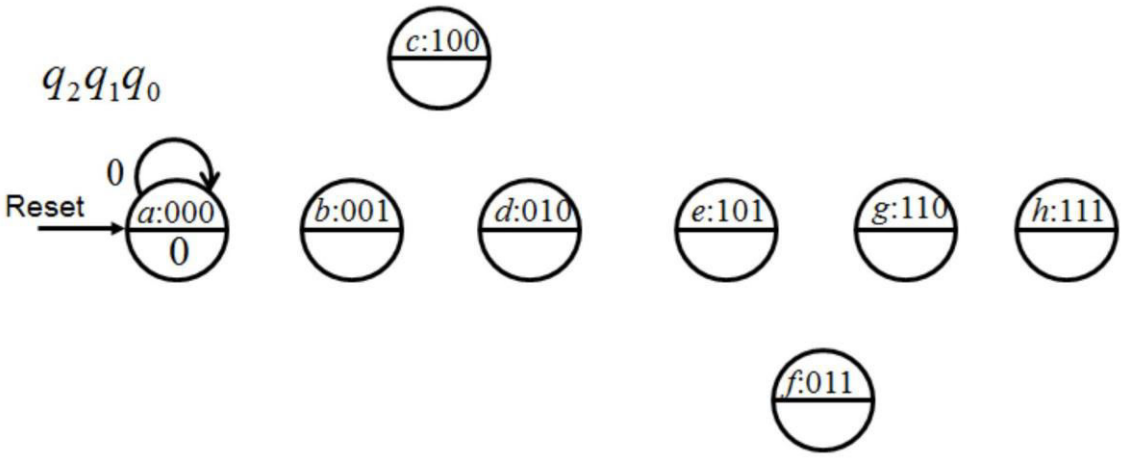
SET-Q

1. A synchronous sequential circuit based on a shift register is used as a "Majority voter". The value of the input signal *w* that occurred most of the times in the past three clock pulses is displayed at the output *y*. The gate denoted by the "M" is a so-called majority gate, it's output takes the same value as the majority of its inputs.

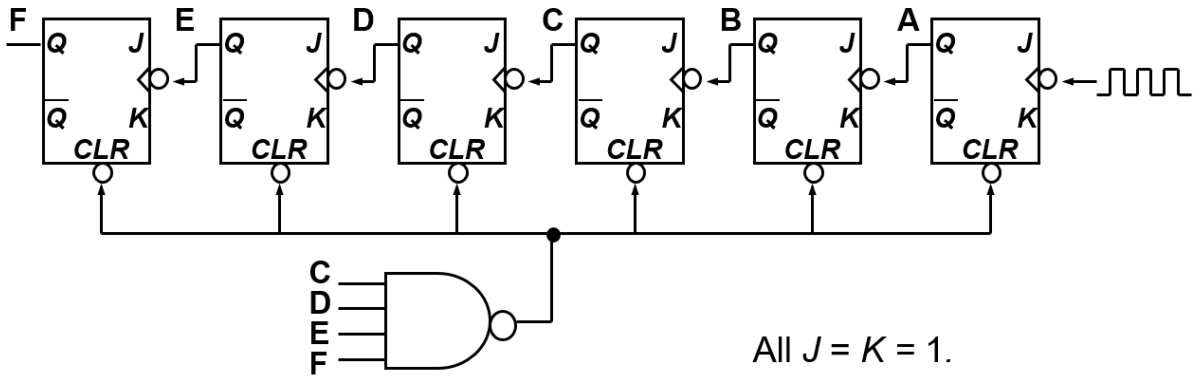


2. Analyze the shift register and draw state diagram and state table for the que 1. (Please take the

help of the initiated state diagram with eight states shown below, but draw your own figure to answer).



3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? Counter?



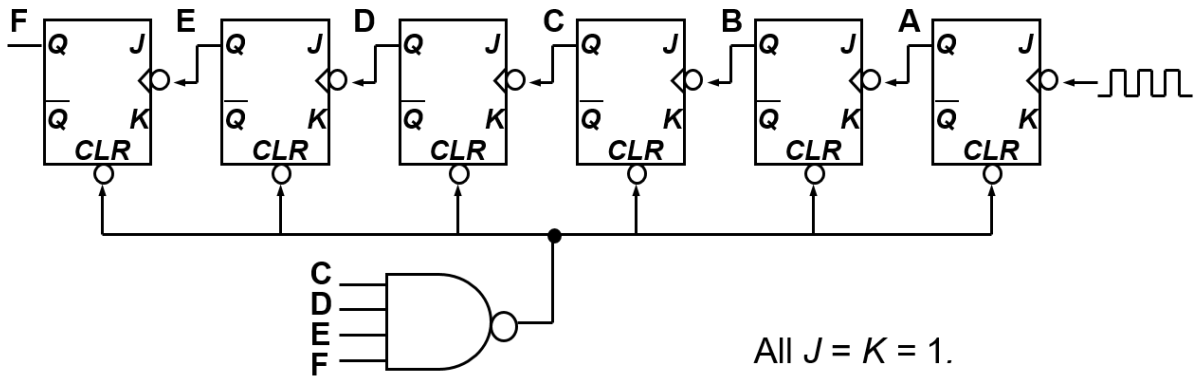
SET-R

1. Draw a state diagram for a mealy type state machine specified as follows:

- Denote "a" the initial state of the machine
- set the initial output to the initial input (the input is presented to the machine bit by bit)
- The output changes value only when three successive inputs have the same value and that value is opposite to the current output. For example, if the current output is 1 and the machine detects three consecutive 0 it changes the output to 0.

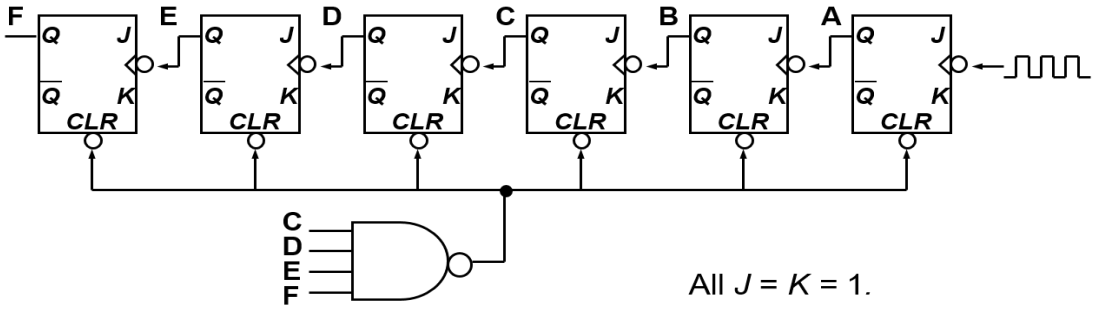
Design the circuit using D – FF and write the Verilog HDL.

2. Design the 4 bit 7311 code to 53-1-1 code and write the Verilog HDL using if else statement.
3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? counter?



SET-S

1. Write an HDL dataflow description of a 4-bit adder/subtractor of unsigned numbers. Use the FOR loop.
2. Design a sequential circuit using T - FF that has 1 data input (w) and 1 data output (z). The output z will become 1 only if the input w has been 1010 over the last 4 clock cycles. Draw the FSM diagram and write/simulate the Verilog code to verify it.
3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? counter?



SET – T

1. Write an HDL dataflow description of a 4-bit adder/subtractor of unsigned numbers. Use the conditional operator.
2. Design a sequential circuit using JK - FF that has 1 data input (w) and 1 data output (z). The output z will become 1 only if the input w has been 1 for the last 5 clock cycles. Draw the FSM diagram and write/simulate the Verilog code to verify it.
3. How to construct an asynchronous MOD-5 counter? MOD-7 counter? MOD-12 counter?
4. The following is a MOD-? counter?