ELEC 2141

Final Exam, Session 1, 2012

Question 1

i)

1 a) $F = \sum m(0, 2, 5, 6, 7, 12, 14)$

2 b) $F = \prod M(1, 3, 4, 8, 9, 10, 11, 13, 15)$

$YZ \backslash WX$	00	01	11	10
00	1	0	1	0
01	0	1	0	0
11	0	1	0	0
10	1	1	1	0

c) Prime implicants:

2 $\bar{W}\bar{X}\bar{Z},\ \bar{W}Y\bar{Z},\ XY\bar{Z},\ WX\bar{Z},\ \bar{W}XY,\ \bar{W}XZ$

d) Essential prime implicants:

2 $\bar{W}\bar{X}\bar{Z},\ WX\bar{Z},\ \bar{W}XZ$

e)

2 $$F = \bar{W}\bar{X}\bar{Z} + WX\bar{Z} + \bar{W}XZ + \begin{cases} \bar{W}XY \\ \text{or} \\ \bar{W}Y\bar{Z} \\ \text{or} \\ XY\bar{Z} \end{cases}$$

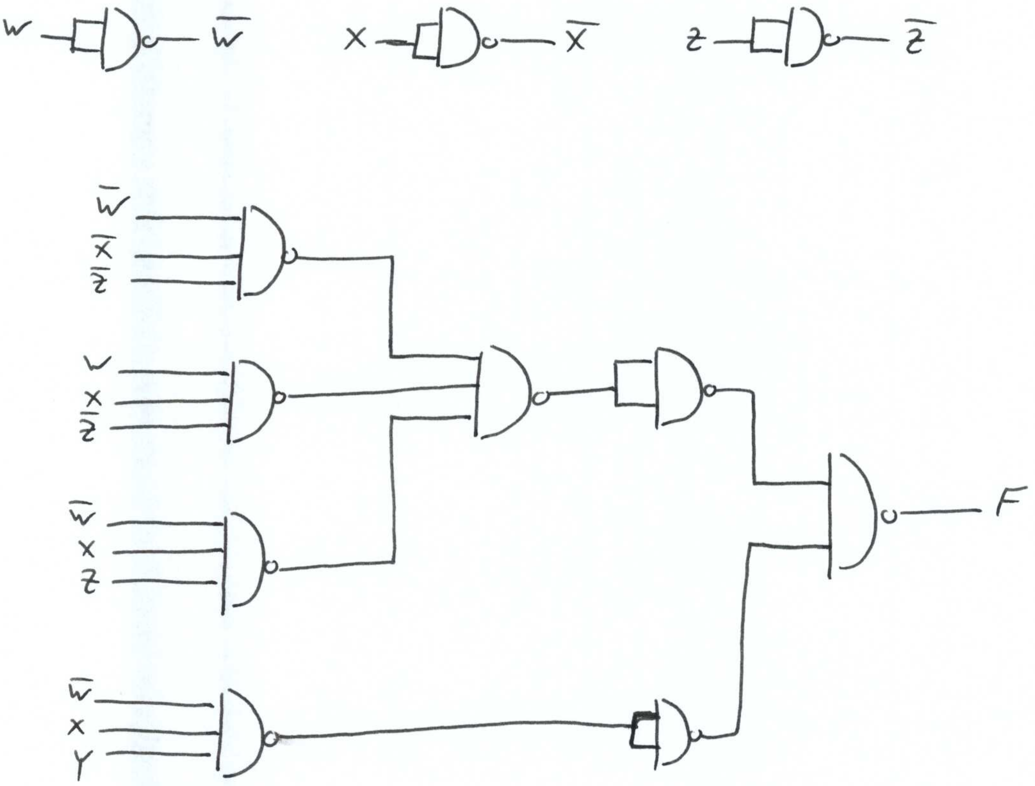
f) $\bar{F} = \bar{X}Z + WZ + W\bar{X} + \bar{W}X\bar{Y}\bar{Z}$

$$\Downarrow$$

2 $$F = (X + \bar{Z})(\bar{W} + \bar{Z})(\bar{W} + X)(W + \bar{X} + Y + Z)$$

g) (other options possible)

3



h) $GIC = 27$

2

ii)

4

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

$$\underbrace{A\bar{A}}_{0}B + \underbrace{A\bar{A}}_{0}C + ACB + ACC + B\bar{A}B + B\bar{A}C + BCB + BCC =$$

$$ABC + AC + \bar{A}B + \bar{A}BC + BC + BC =$$

$$AC(1+B) + \bar{A}B(1+C) + BC =$$

$$AC + \bar{A}B + BC = (A+B)(\bar{A}+C) \checkmark$$

[can also be proved by proving the dual of the identity]

Question 2

i) Convert to decimal first:

5

$$3A57.17_{(12)} = 3\times12^3 + 10\times12^2 + 5\times12 + 7 + 1\times12^{-1} + 7\times12^{-2}$$

$$= 5184 + 1440 + 60 + 7 + \frac{1}{12} + \frac{7}{144}$$

$$= 6691\,\frac{19}{144}\,_{(10)}$$

Convert integer part to senary:

$$\left.\begin{array}{ll} 6691/6 = 1115 & R=1 \\ 1115/6 = 185 & R=5 \\ 185/6 = 30 & R=5 \\ 30/6 = 5 & R=0 \\ 5/6 = 0 & R=5 \end{array}\right\} \quad 50551_{(6)}$$

Convert fractional part to senary:

$$\left.\begin{array}{ll} \frac{19}{144}\times 6 = \frac{19}{24} & I=0 \\ \frac{19}{24}\times 6 = 4\frac{3}{4} & I=4 \\ \frac{3}{4}\times 6 = 4\frac{1}{2} & I=4 \\ \frac{1}{2}\times 6 = 3 & I=3 \end{array}\right\} \quad 0.0443_{(6)}$$

So:

$$\boxed{3A57.17_{(12)} = 50551.0443_{(6)}}$$

ii)

a) (2) In a Moore machine, the output is a function of the current state only and may change only when the state changes.

output = f(state)

In a Mealy machine, the output is a function of both the current input and the current state and may change when either the input or the state change.

output = f(input, state)

b) (2)

current state	next state x=0	next state x=1	output
A	B	A	0
B	D	C	1
C	D	C	0
D	D	A	1

c) (3)

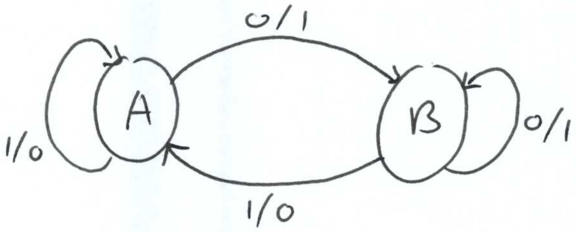
	A	B	C
B	X		
C	B~D A~C	X	
D	X	D~D A~C	X

=> A ~ C
B ~ D =>

current state	next state x=0	next state x=1	output
A	B	A	0
B	B	A	1

d)

3



current state	next state, output $x=0$	next state, output $x=1$
A	B, 1	A, 0
B	B, 1	A, 0

e)

5

```
module fsm(clk, x, z);
    input clk, x;
    output z;
    parameter A = 1'b0, B = 1'b1;
    reg state = A;
    assign z = state;
    always @(posedge clk) begin
        state = x ? B : A;
    end
endmodule
```

Question 3

i) use JK table:

7

q	Q	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

current state	next state		flip-flop inputs			
	$x=0$	$x=1$	$x=0$		$x=1$	
$q_1 q_0$	$Q_1 Q_0$	$Q_1 Q_0$	$J_1 K_1$	$J_0 K_0$	$J_1 K_1$	$J_0 K_0$
0 0	0 0	0 1	0 X	0 X	0 X	1 X
0 1	0 0	1 0	0 X	X 1	1 X	X 1
1 0	0 0	0 1	X 1	0 X	X 1	1 X
1 1	1 1	0 0	X 0	X 0	X 1	X 1

J_1:

$x \backslash q_1 q_0$	00	01	11	10
0	0	0	X	X
1	0	1	X	X

K_1:

$x \backslash q_1 q_0$	00	01	11	10
0	X	X	0	1
1	X	X	1	1

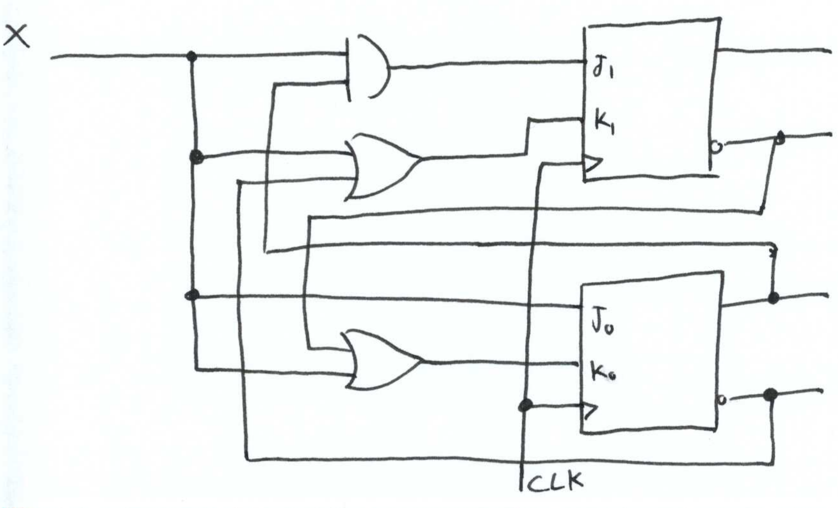
J_0:

$x \backslash q_1 q_0$	00	01	11	10
0	0	X	X	0
1	1	X	X	1

K_0:

$x \backslash q_1 q_0$	00	01	11	10
0	X	1	0	X
1	X	1	1	X

$J_1 = q_0 x$ $\quad$ $K_1 = \bar{q}_0 + x$ $\quad$ $J_0 = x$ $\quad$ $K_0 = \bar{q}_1 + x$



ii)

a) This circuit implements an active-low SR-Latch

1

(or: $\bar{S}\bar{R}$-Latch)

b)

2

$w \sim \bar{S}$ -> set the latch to output 1

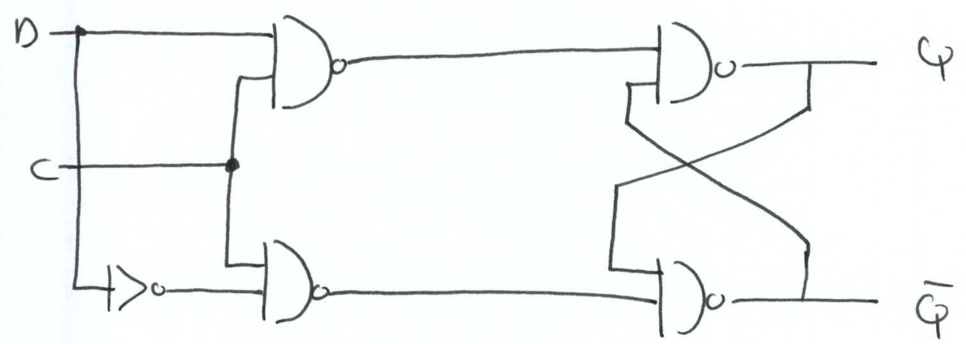
$x \sim \bar{R}$ -> Reset the latch to output 0

$Y \sim Q$ -> Latch output

$z \sim \bar{Q}$ -> Latch output complemented.

c)

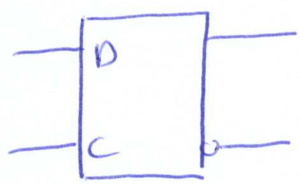
3



function table:

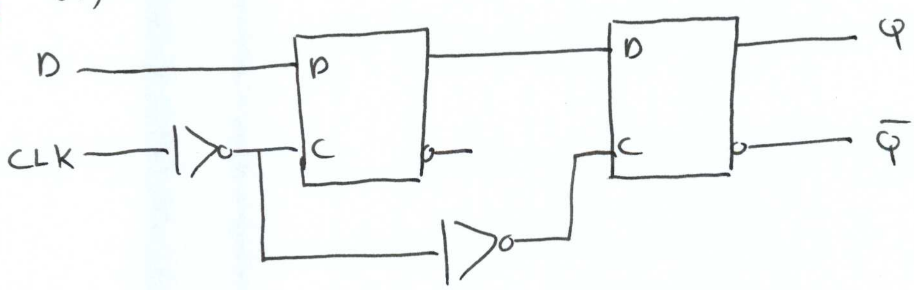
C	D	Q	$\bar{Q}$
0	0	q	$\bar{q}$
0	1	q	$\bar{q}$
1	0	0	1
1	1	1	0

symbol:

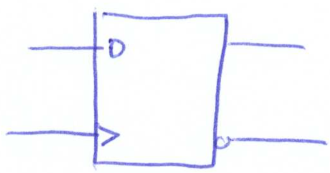


3

d)

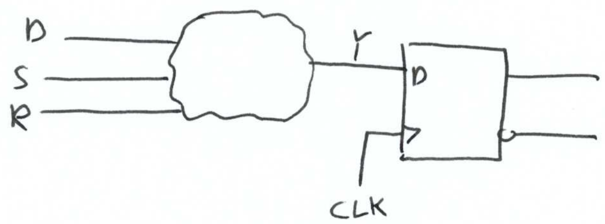


symbol:



e) Add logic to the input of the existing D flip-flop:

4



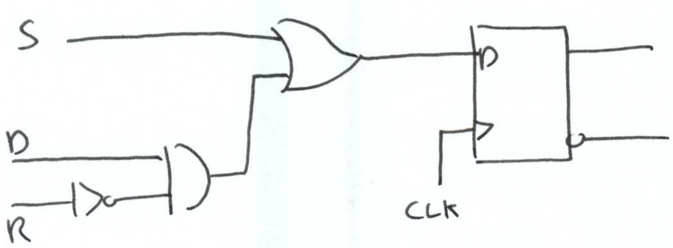
2 possible options:

If S dominates:

S	R	D	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

D \ SR	00	01	11	10
0	0	0	1	1
1	1	0	1	1

$$Y = S + D\bar{R}$$

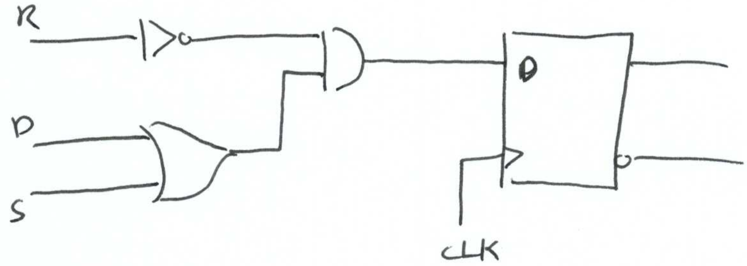


If R dominates:

S	R	D	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

D \ SR	00	01	11	10
0	0	0	0	1
1	1	0	0	1

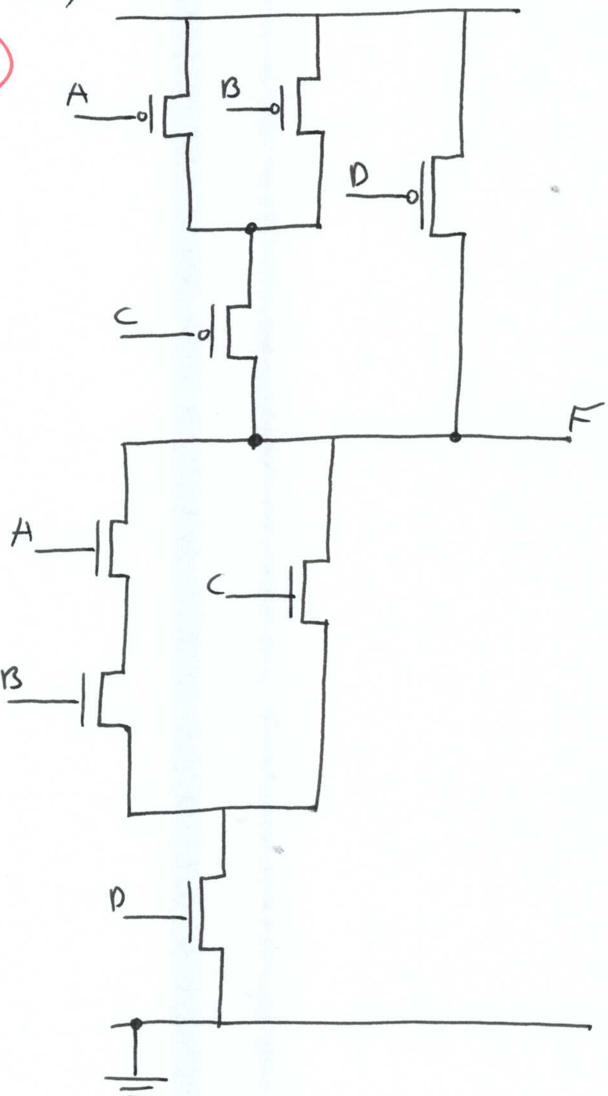
$$Y = S\bar{R} + D\bar{R}$$
$$= \bar{R}(S + D)$$



Question 4

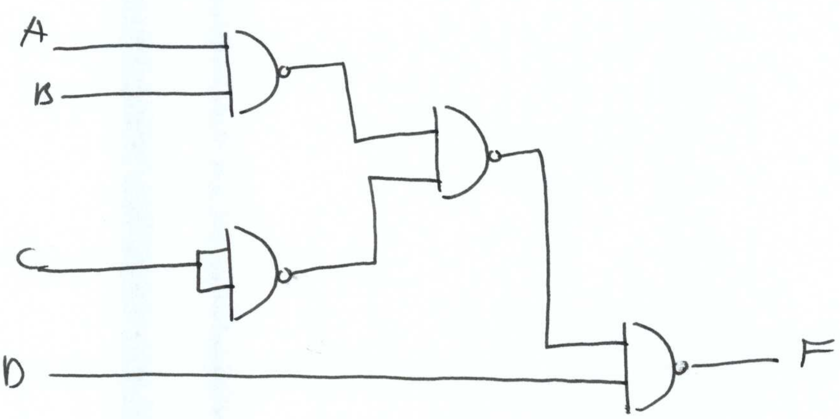
a)

5



b) $F = \overline{(AB+C)D} = \overline{(\overline{\overline{AB+C}})D} = \overline{(\overline{\overline{AB}\cdot\overline{C}})D}$

4

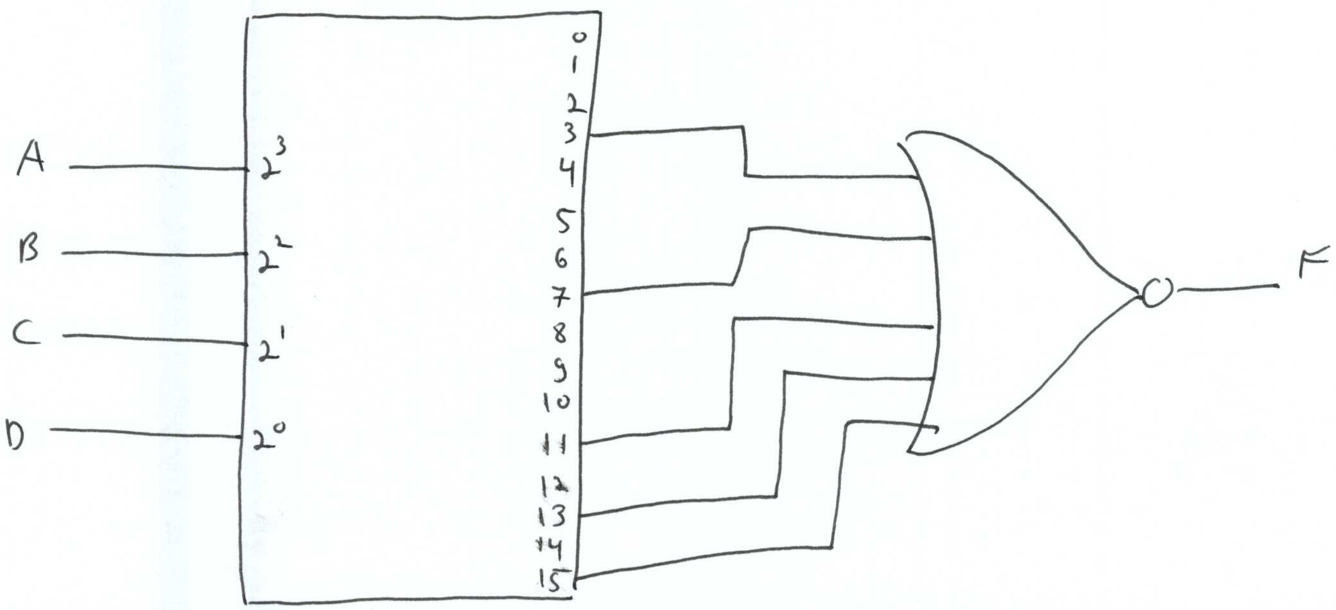


c)

3

A	B	C	D	F	$\bar{F}$
0	0	0	0	1	0
0	0	0	1	1	0
0	0	1	0	1	0
0	0	1	1	0	1
0	1	0	0	1	0
0	1	0	1	1	0
0	1	1	0	1	0
0	1	1	1	0	1
1	0	0	0	1	0
1	0	0	1	1	0
1	0	1	0	1	0
1	0	1	1	0	1
1	1	0	0	1	0
1	1	0	1	0	1
1	1	1	0	1	0
1	1	1	1	0	1

use $\bar{F}$ to find implementation using NOR gate:



(i)

8

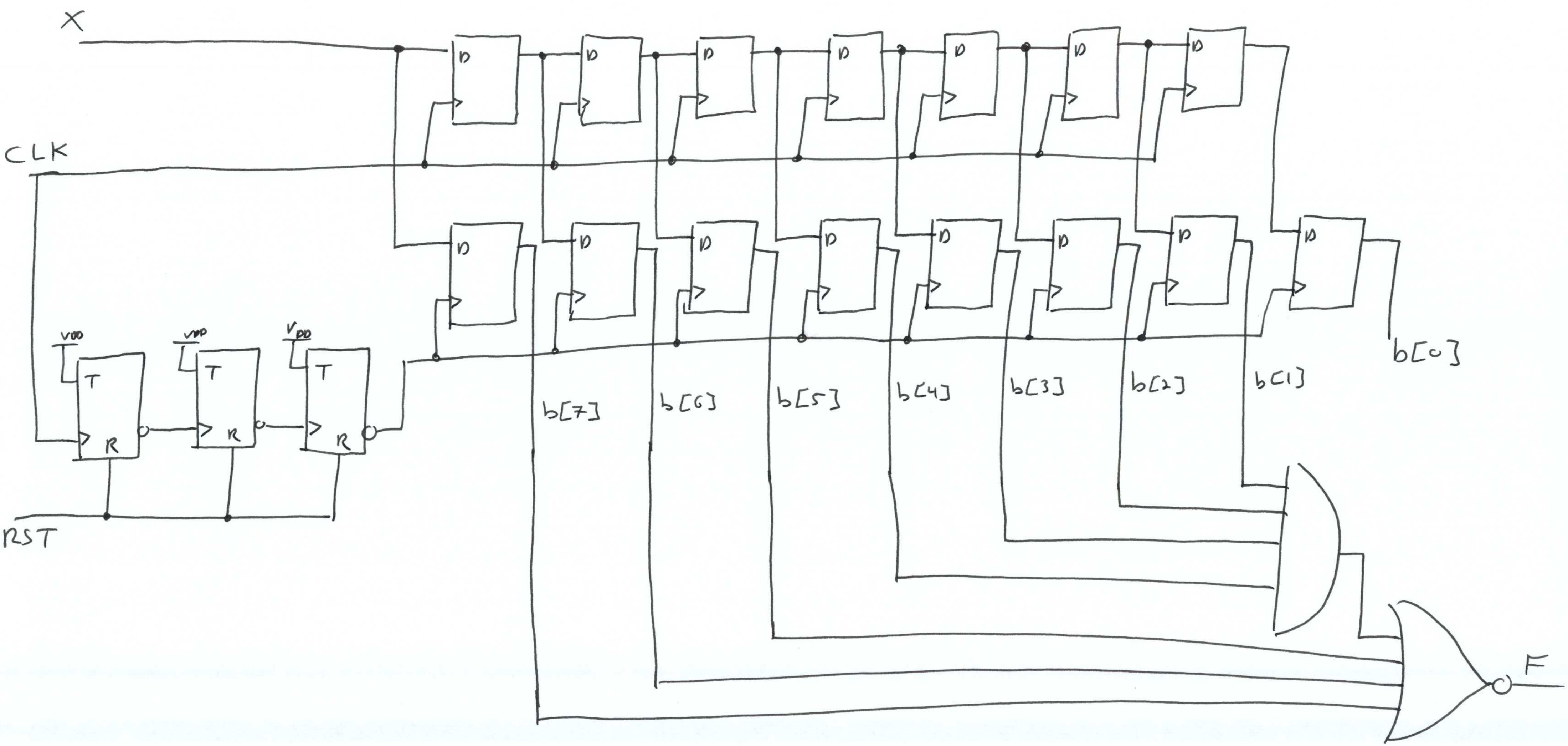
4-bits prime numbers: 2, 3, 5, 7, 11, 13

```
module Prime(N, F);
    input [3:0] N;
    output F;

    reg F;

    always @(N) begin
        if (N==2 || N==3 || N==5 ||
            N==7 || N==11 || N==13)
            F <= 1'b1;
        else
            F <= 1'b0;
    end
endmodule
```

(other implementations are possible).

Question 5

20



Operation:

The top row of D flip-flops forms a shift-register such that the input X is sampled on every rising edge of the clock and the previous bits are shifted to the right. Every 8 clock cycles, the right-most flip-flop contains the LSB, while the MSB is available at the input.

The 3 T flip-flops with asynchronous Reset are connected to form a 3-bit counter that will generate a rising edge at its output every 8 clock cycles. This output is used to clock the bottom row of D flip-flops that will therefore sample the top row every 8 clock cycles and hold its value for the next 8 clock cycles (while the next serial number is read in).

The comparator works as follows:

$$30_{(10)} = 00011110_{(2)}$$

and: $$b \geq 30 \text{ if } b_4 = b_3 = b_2 = b_1 = 1$$

or

$$\text{any of } b_7, b_6, b_5 = 1$$

Therefore, for $b < 30$:

$$F = \overline{(b_1 \cdot b_2 \cdot b_3 \cdot b_4) + b_5 + b_6 + b_7}$$