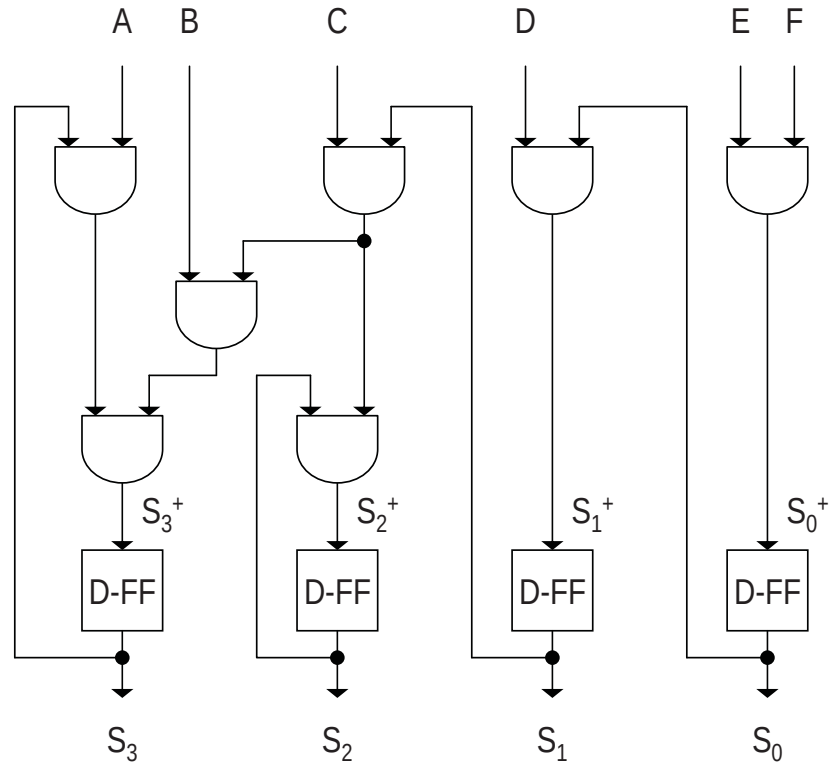


1) Delay Analysis

Assume **2 ns** gate delay for 2-AND. Assume the positive edge-triggered flip-flop delay is $T_{delay_{FF}} = 1 \text{ ns}$ and **setup time** is also $T_{setup_{FF}} = 1 \text{ ns}$.



What are **final delays** for the following:

$$S_0^+ = 2 \text{ ns}$$

$$S_0 = 1 \text{ ns}$$

$$S_1^+ = 3 \text{ ns}$$

$$S_1 = 1 \text{ ns}$$

$$S_2^+ = 5 \text{ ns}$$

$$S_2 = 1 \text{ ns}$$

$$S_3^+ = 7 \text{ ns}$$

$$S_3 = 1 \text{ ns}$$

$$\text{Clock period} = 8 \text{ ns}$$

2) Number Representation

(a) For each binary vector below, what does it represent if it is interpreted as a Decimal number in conventional binary form (positive only), a Decimal number in 2's complement form (positive and negative), an Octal number, and a Hexadecimal number?

Binary	Decimal		Octal	Hexadecimal
	conventional form	2's complement form		
000111	7	7	07 (just 7 OK)	07 (just 7 OK)
001111	15	15	17	0F (just F OK)
011111	31	31	37	1F
111110	62	-2	76	3E
111100	60	-4	74	3C
111000	56	-8	70	38

(b) What is the **minimum** number of bits required to represent the following 2's complement numbers?

-27 num bits = 6

64 num bits = 8

3) Sequential Logic Implementation and Two-Level Minimization

Consider the following incompletely-specified Moore machine (“*” indicates don’t cares), and the following state encoding.

State table

S_1S_0	$S_1^+S_0^+$				F
	$XY = 00$	01	10	11	
A	*	B	B	*	1
B	A	C	C	A	1
C	*	A	A	*	0

State encoding

State	S_1S_0
A	00
B	01
C	10

For each function, fill in the K-map, identify the corresponding primes and essential primes, and find the minimum 2-level logic implementation.

		F	
S_1	S_0	0	1
0		1	1
1		0	*

		S_1^+			
S_1S_0	XY	00	01	11	10
00		*	0	*	0
01		0	1	0	1
11		*	*	*	*
10		*	0	*	0

		S_0^+			
S_1S_0	XY	00	01	11	10
00		*	1	*	1
01		0	0	0	0
11		*	*	*	*
10		*	0	*	0

F	
Prime	Essential
$\bar{S}_1$	$\bar{S}_1$
S_0	

S_1^+	
Prime	Essential
$S_0\bar{X}Y$ $S_0X\bar{Y}$	$S_0\bar{X}Y$ $S_0X\bar{Y}$
S_1S_0 $S_1\bar{X}\bar{Y}$ S_1XY $\bar{S}_0\bar{X}\bar{Y}$ $\bar{S}_0XY$	

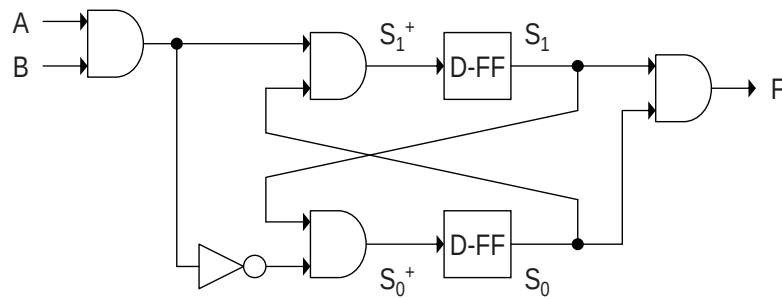
S_0^+	
Prime	Essential
$\bar{S}_1\bar{S}_0$	$\bar{S}_1\bar{S}_0$
S_1S_0 $S_1\bar{X}\bar{Y}$ S_1XY $\bar{S}_0\bar{X}\bar{Y}$ $\bar{S}_0XY$	

$$F = \bar{S}_1$$

$$S_1^+ = S_0\bar{X}Y + S_0X\bar{Y}$$

$$S_0^+ = \bar{S}_1\bar{S}_0$$

4) Reverse engineer sequential circuit to finite state machine



a) Fill out the following functions.

$$\begin{aligned}
 F &= S_1 S_0 \\
 S_1^+ &= A B S_0 \\
 S_0^+ &= (\bar{A} \bar{B}) S_1 = (\bar{A} + \bar{B}) S_1 = \bar{A} S_1 + \bar{B} S_1
 \end{aligned}$$

b) With initial state = **00**, what states, if any, are **not** reachable?

01, 10, 11

c) Assume initial state is **00**, fill in the following state table. If a state is not reachable, indicate the appropriate **don't cares** (use "*" or "X" to indicate don't cares).

Correct answer should be the following b/c only **00** is reachable state.

$S_1 S_0$	$S_1^+ S_0^+$				F
	$AB = 00$	01	10	11	
00	00	00	00	00	0
01	*	*	*	*	*
10	*	*	*	*	*
11	*	*	*	*	*

If they got the the unreachable states wrong, then they might produce the following incorrect logic table (**wrong** because of not considering unreachable states as don't cares).

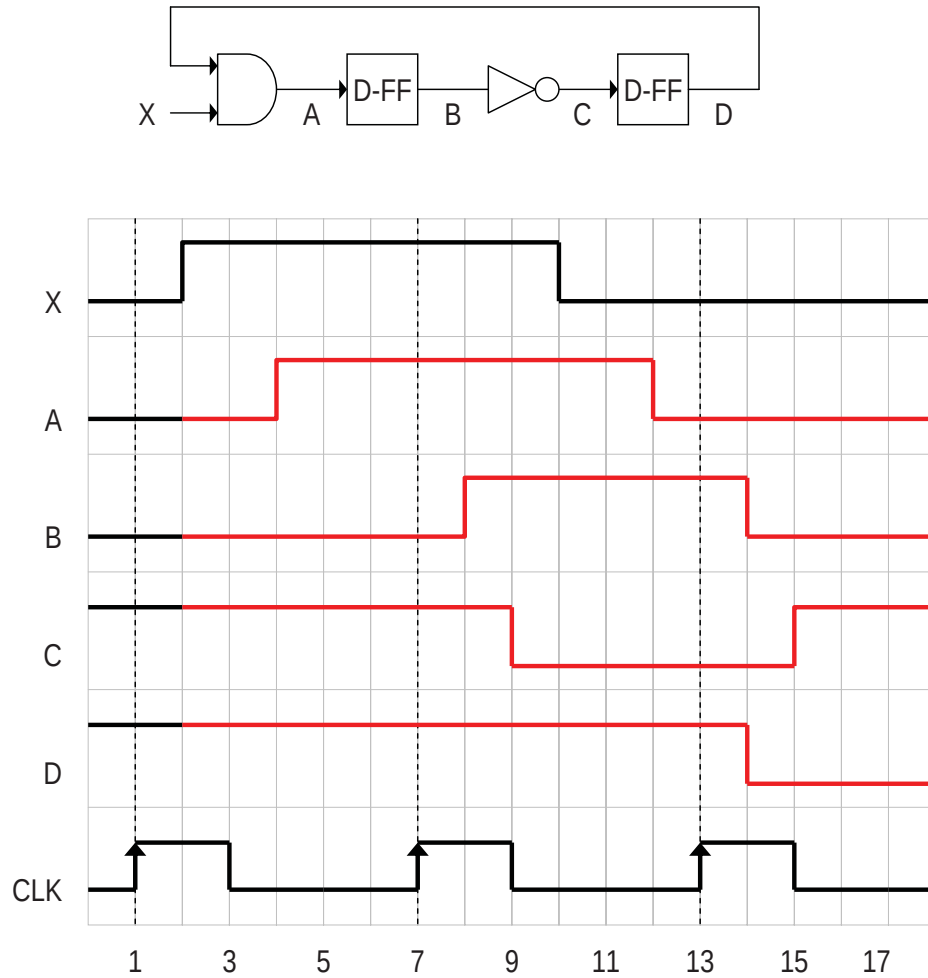
$S_1 S_0$	$S_1^+ S_0^+$				F
	$AB = 00$	01	10	11	
00	00	00	00	00	0
01	00	00	00	10	0
10	01	01	01	00	0
11	01	01	01	10	1

5) Timing analysis

Consider the following sequential logic diagram.

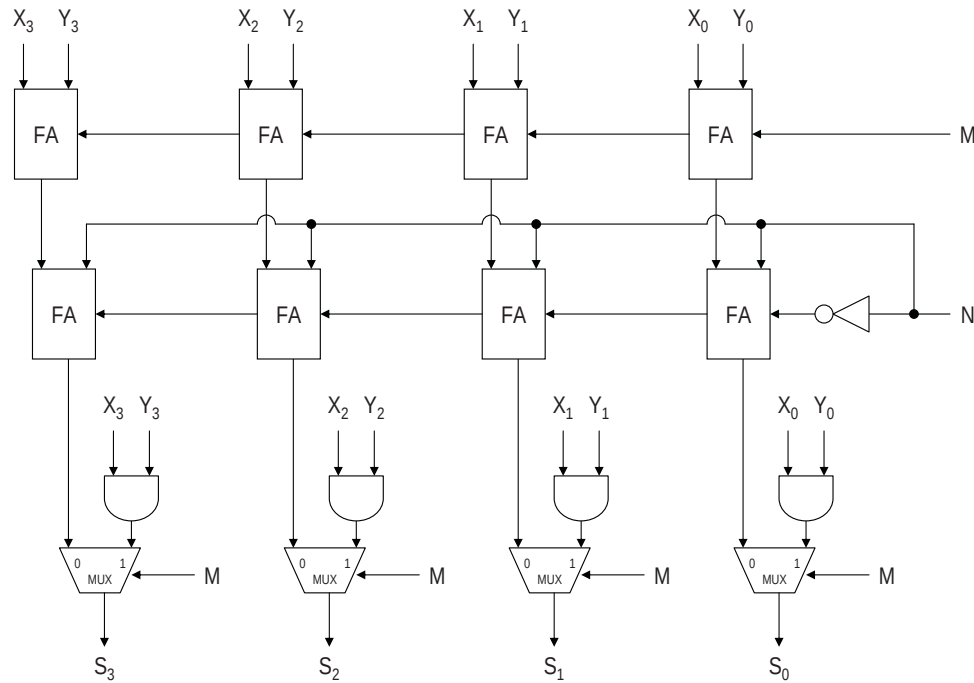
Assume the gate delay for 2-AND is **2 ns** and for INV is **1 ns**. Assume the positive edge-triggered flip-flop delay is $T_{delayFF} = 1 \text{ ns}$, and assume the positive edge-triggered flip-flop setup time is also $T_{setupFF} = 1 \text{ ns}$.

Fill out the rest of the timing diagram for A , B , C , and D .



6) Arithmetic/logic

Consider the following diagram. To avoid drawing many messy wires, the **name** of a wire is used in many cases instead.



Given the following inputs ($M, N, X_3X_2X_1X_0, Y_3Y_2Y_1Y_0$), complete the following logic table.

MN	$X_3X_2X_1X_0$	$Y_3Y_2Y_1Y_0$	$S_3S_2S_1S_0$	Comments
00	0000	0000	0001	$S = X + Y + 1$
00	0010	0100	0111	$S = X + Y + 1$
01	0000	0001	0000	$S = X + Y - 1$
01	0100	0000	0011	$S = X + Y - 1$
10	1111	1111	1111	$S = X Y$
10	0011	1101	0001	$S = X Y$
11	0101	1010	0000	$S = X Y$
11	0000	1111	0000	$S = X Y$

7) Logic minimization

Minimize the number of literals for each question below. (You can introduce new intermediate logic equations if doing so helps to reduce the number of literals in total.)

a) $F = \bar{A}BC + AC + \bar{C}$

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

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

b) $F = \bar{A}\bar{B} + AC + BC + AD + BD + \bar{C}\bar{D}$

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

$$= 1$$

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

c) $F = AB + \bar{A} + \bar{B}$

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

$$= B + \bar{A} + \bar{B}$$

$$= 1$$

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

$$= AB + AC + \bar{B} + CD$$

$$= A + AC + \bar{B} + CD$$

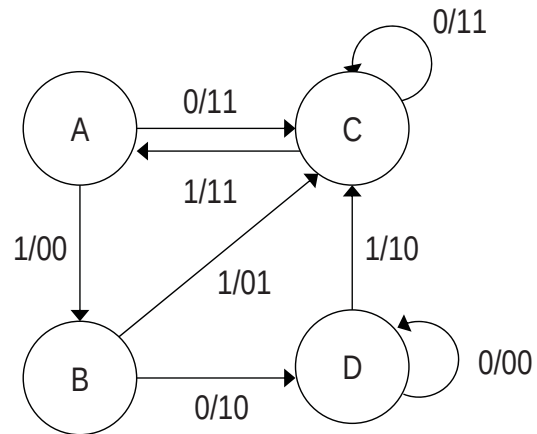
$$= A + \bar{B} + CD$$

e) $F = ABCD + ACD + BCD$

$$= ACD + BCD$$

$$= CD(A + B)$$

8) State machine



For the Mealy machine above, suppose the initial state is **A**. Starting from initial state **A**, what **outputs** are **NOT possible** (e.g. 1/00 means input is 1 and output is 00) when

- a) The input is always **0**: **00**, **01**, **10** are **NOT** possible
- b) The input is always **1**: **10** is **NOT** possible