Name :- Chandrawanshi Mangesh Shivaji

Roll No :- 1801CS16.

Sign :-

Date :- 16/04/2020.

CS226 - Switching Theory Lab - 9.

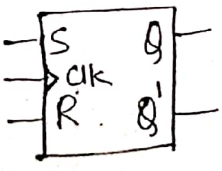
Q.1) 4-bit synchronous counter

(a) Using S-R flipflop :

4 flip-flops required to hold states

clock signal controls when flip-flop memory can change.

Present state				Next state				Flip-flop Input							
C_4	C_3	C_2	C_1	N_4	N_3	N_2	N_1	S_4	R_4	S_3	R_3	S_2	R_2	S_1	R_1
0	0	0	0	0	0	0	1	0	X	0	X	0	X	1	0
0	0	0	1	0	0	1	0	0	X	0	X	1	X	0	1
0	0	1	0	0	0	1	1	0	X	0	X	0	X	1	0
0	0	1	1	0	1	0	0	0	X	1	0	0	1	0	1
0	1	0	0	0	1	0	1	0	X	X	0	0	X	1	0
0	1	0	1	0	1	1	0	0	X	X	0	1	0	0	1
0	1	1	0	0	1	1	1	0	X	X	0	X	0	1	0
0	1	1	1	1	0	0	0	1	0	0	1	0	1	0	1
1	0	0	0	1	0	0	1	X	0	0	X	0	X	1	0
1	0	0	1	1	0	1	0	X	0	0	X	1	0	0	1
1	0	1	0	1	0	1	1	X	0	0	X	X	0	1	0
1	0	1	1	1	1	0	0	X	0	1	0	0	1	0	1
1	1	0	0	1	1	0	1	X	0	X	0	0	X	1	0
1	1	0	1	1	1	1	0	X	0	X	0	1	0	0	1
1	1	1	0	1	1	1	1	X	0	X	0	X	0	1	0
1	1	1	1	0	0	0	0	0	1	0	1	0	1	0	1



Excitation Table

Q	Q(next)	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

C_4C_3 \ C_2C_1	00	01	11	10
00	1 (0)	0 (1)	0 (3)	1 (2)
01	1 (4)	0 (5)	0 (7)	1 (6)
11	1 (12)	0 (13)	0 (15)	1 (14)
10	1 (8)	0 (9)	0 (11)	1 (10)

$$S1 = C_1'$$

C_4C_3 \ C_2C_1	00	01	11	10
00	0	1	1	0
01	0	1	1	0
11	0	1	1	0
10	0	1	1	0

$$R1 = C_1$$

C_4C_3 \ C_2C_1	00	01	11	10
00	0	1	0	X
01	0	1	0	X
11	0	1	0	X
10	0	1	0	X

$$S2 = C_1C_2'$$

C_4C_3 \ C_2C_1	00	01	11	10
00	X	0	1	0
01	X	0	1	0
11	X	0	1	0
10	X	0	1	0

$$R2 = C_1C_2$$

C_4C_3 \ C_2C_1	00	01	11	10
00	0	0	1	0
01	X	X	0	X
11	X	X	0	X
10	0	0	1	0

$$S3 = C_1C_2C_3'$$

C_4C_3 \ C_2C_1	00	01	11	10
00	X	X	0	X
01	0	0	1	0
11	0	0	1	0
10	X	X	0	X

$$R3 = C_1C_2C_3$$

C_4C_3 \ C_2C_1	00	01	11	10
00	0	0	0	0
01	0	0	1	0
11	X	X	0	X
10	X	X	X	X

$$S4 = C_1C_2C_3C_4'$$

C_4C_3 \ C_2C_1	00	01	11	10
00	X	X	X	X
01	X	X	0	X
11	0	0	1	0
10	0	0	0	0

$$R4 = C_1C_2C_3C_4$$

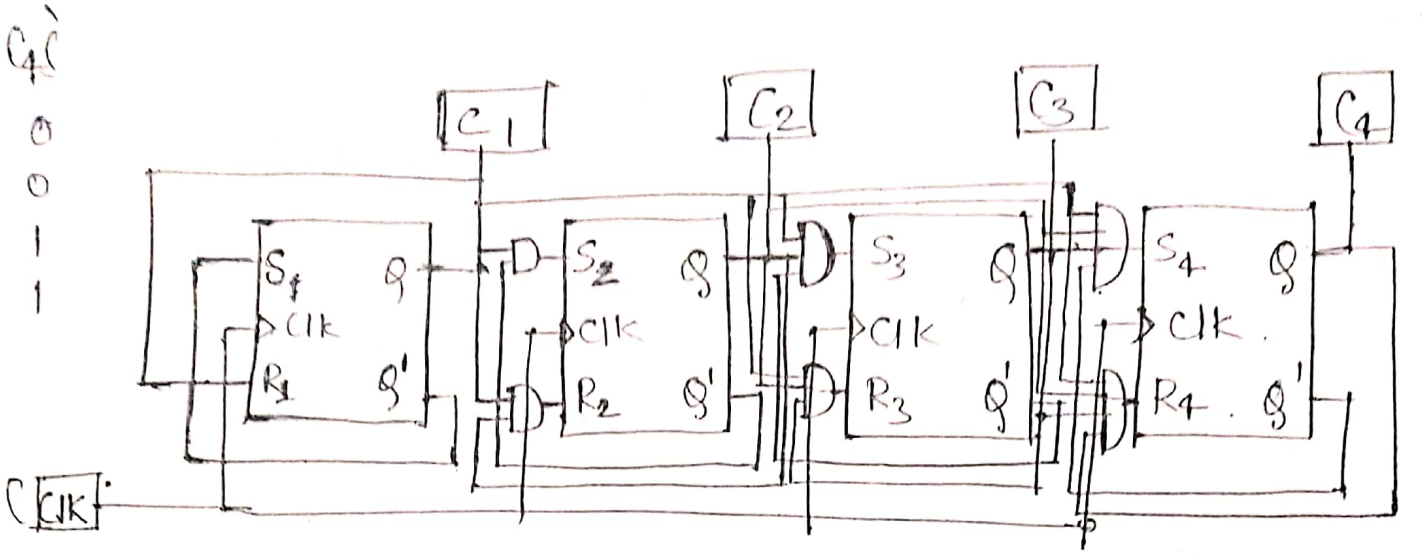


fig. 4-bit RS counter (synchronous).

$S_1 = C_1'$ $R_1 = C_1$ $S_2 = C_1C_2'$ $R_2 = C_1C_2$ $S_3 = C_1C_2C_3'$ $R_3 = C_1C_2C_3$

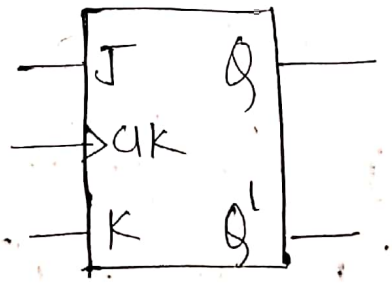
$S_4 = C_1C_2C_3C_4'$ $R_4 = C_1C_2C_3C_4$

(b) Using J-K flipflop.

4-flipflops req. to hold states.

clock signal changes when flip-flop memory can change.

Q	Qnext	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0



Excitation Table.

Present state				Next State				Flip-flop Inputs							
C_4	C_3	C_2	C_1	N_4	N_3	N_2	N_1	J_4	K_4	J_3	K_3	J_2	K_2	J_1	K_1
0	0	0	0	0	0	0	1	0	X	0	X	0	X	1	X
0	0	0	1	0	0	1	0	0	X	0	X	1	X	X	1
0	0	1	0	0	0	1	1	0	X	0	X	X	0	1	X
0	0	1	1	0	1	0	0	0	X	1	X	X	1	X	1
0	1	0	0	0	1	0	1	0	X	X	0	0	X	1	X
0	1	0	1	0	1	1	0	0	X	X	0	1	X	X	1
0	1	1	0	0	1	1	1	0	X	X	0	X	0	1	X
0	1	1	1	1	0	0	0	1	X	X	1	X	1	X	1
1	0	0	0	1	0	0	1	X	0	0	X	0	X	1	X
1	0	0	1	1	0	1	0	X	0	0	X	1	X	X	1
1	0	1	0	1	0	1	1	X	0	0	X	X	0	1	X
1	0	1	1	1	1	0	0	X	0	X	1	X	1	X	1
1	1	0	0	1	1	0	1	X	0	X	0	0	X	1	X
1	1	0	1	1	1	1	0	X	0	X	0	1	X	X	1
1	1	1	0	1	1	1	1	X	0	X	0	X	0	1	X
1	1	1	1	0	0	0	0	X	1	X	1	X	1	X	1

C_4C_3 \ C_2C_1	00	01	11	10
00	1	X	X	1
01	1	X	X	1
11	1	X	X	1
10	1	X	X	1

$J_1 = 1$

C_4C_3 \ C_2C_1	00	01	11	10
00	X	1	1	X
01	X	1	1	X
11	X	1	1	X
10	X	1	1	X

$K_1 = 1$

C_4C_3 \ C_2C_1	00	01	11	10
00	0	1	X	X
01	0	1	X	X
11	0	1	X	X
10	0	1	X	X

$J_2 = C_1$

C_4C_3 \ C_2C_1	00	01	11	10
00	X	X	1	0
01	X	X	1	0
11	X	X	1	0
10	X	X	1	0

$K_2 = C_1$

C_4C_3 \ C_2C_1	00	01	11	10
00	0	0	1	0
01	X	X	X	X
11	X	X	X	X
10	0	0	1	0

$J_3 = C_1C_2$

C_4C_3 \ C_2C_1	00	01	11	10
00	X	X	X	X
01	0	0	1	0
11	0	0	1	0
10	X	X	X	X

$K_3 = C_1C_2$

C_4C_3 \ C_2C_1	00	01	11	10
00	0	0	0	0
01	0	0	1	0
11	X	X	X	X
10	X	X	X	X

$J_4 = C_1C_2C_3$

C_4C_3 \ C_2C_1	00	01	11	10
00	X	X	X	X
01	X	X	X	X
11	0	0	1	0
10	0	0	0	0

$K_4 = C_1C_2C_3$

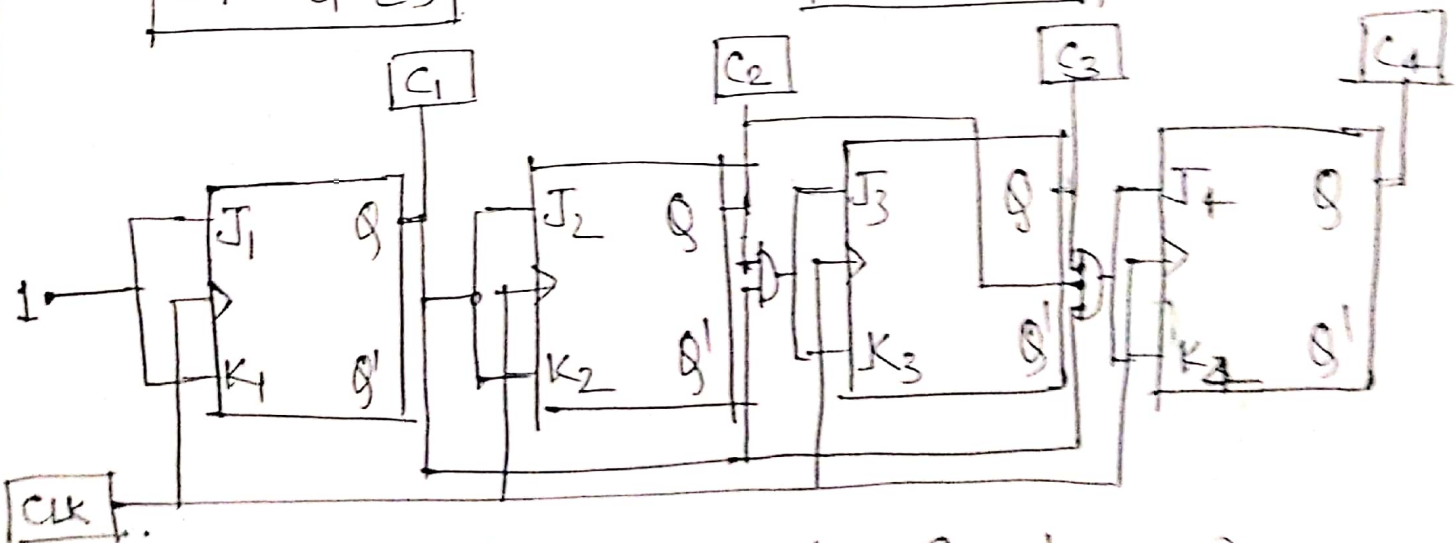


Fig. 4-bit J-K Counter (Synchronous)

(c) Using T-flipflop: 4 flipflop, 1 rising edge of memory state changes clock pulse

Fig. T-flipflop

Q	Q(next)	T
0	0	0
0	1	1
1	0	1
1	1	0

Excitation Table

Present State				Next State				Flip-flop Inputs			
C_4	C_3	C_2	C_1	N_4	N_3	N_2	N_1	T_4	T_3	T_2	T_1
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	1	0	0	1	0	0	0	1	1
0	0	1	0	0	0	1	1	0	0	0	1
0	0	1	1	0	1	0	0	0	1	1	1
0	1	0	0	0	1	0	1	0	0	0	1
0	1	0	1	0	1	1	0	0	0	1	1
0	1	1	0	0	1	1	1	0	0	0	1
0	1	1	1	1	0	0	0	1	1	1	1
1	0	0	0	1	0	0	1	0	0	0	1
1	0	0	1	1	0	1	0	0	0	1	1
1	0	1	0	1	0	1	1	0	0	0	1
1	0	1	1	1	1	0	0	0	1	1	1
1	1	0	0	1	1	0	1	0	0	0	1
1	1	0	1	1	1	1	0	0	0	1	1
1	1	1	0	1	1	1	1	0	0	0	1
1	1	1	1	0	0	0	0	1	1	1	1

C_4C_3 \ C_2C_1	00	01	11	10
00	1	1	1	1
01	1	1	1	1
11	1	1	1	1
10	1	1	1	1

$T_1 = 1$

C_4C_3 \ C_2C_1	00	01	11	10
00	0	1	1	0
01	0	1	1	0
11	0	1	1	0
10	0	1	1	0

$T_2 = C_1$

C_4C_3 \ C_2C_1	00	01	11	10
00	0	0	1	0
01	0	0	1	0
11	0	0	1	0
10	0	0	1	0

$T_3 = C_1C_2$

C_4C_3 \ C_2C_1	00	01	11	10
00	0	0	0	0
01	0	0	1	0
11	0	0	1	0
10	0	0	0	0

$T_4 = C_1C_2C_3$

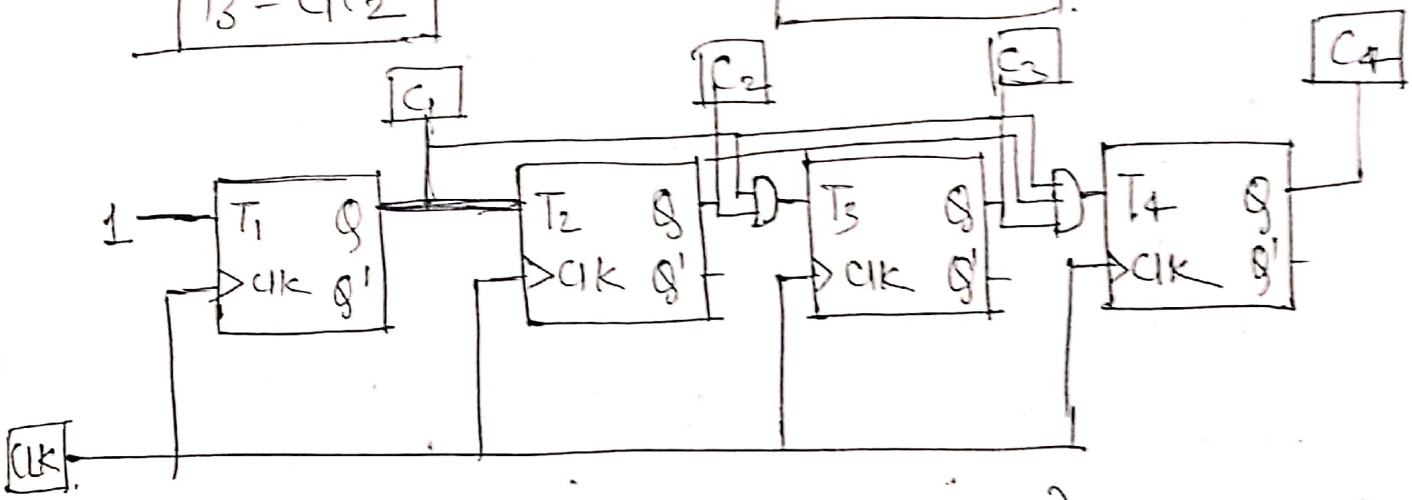
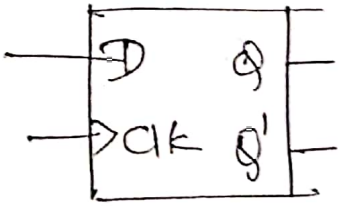


fig. 4-bit T counter (synchronous).

(d) Using D-flipflop. 4 flipflops req.



Excitation Table.

Q	Q(next)	D
0	0	0
0	1	1
1	0	0
1	1	1

Present State				Next State				flip-flop inputs			
Q_4	Q_3	Q_2	Q_1	N_4	N_3	N_2	N_1	D_4	D_3	D_2	D_1
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	1	0	0	1	0	0	0	1	0
0	0	1	0	0	0	1	1	0	0	1	1
0	0	1	1	0	1	0	0	0	1	0	0
0	1	0	0	0	1	0	1	0	1	0	1
0	1	0	1	0	1	1	0	0	1	1	0
0	1	1	0	0	1	1	1	0	1	1	1
0	1	1	1	1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	1	1	0	0	1
1	0	0	1	1	0	1	0	1	0	1	0
1	0	1	0	1	0	1	1	1	0	1	1
1	0	1	1	1	1	0	0	1	1	0	0
1	1	0	0	1	1	0	1	1	1	0	1
1	1	0	1	1	1	1	0	1	1	1	0
1	1	1	0	1	1	1	1	1	1	1	1
1	1	1	1	0	0	0	0	0	0	0	0

Q_4Q_3 \ Q_2Q_1	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	1	0	0	1
10	1	0	0	1

$$D_1 = Q_1'$$

Q_4Q_3 \ Q_2Q_1	00	01	11	10
00	0	1	0	1
01	0	1	0	1
11	0	1	0	1
10	0	1	0	1

$$D_2 = Q_1Q_2' + Q_2Q_1' = Q_1 \oplus Q_2$$

C_4C_3 \ C_2C_1	00	01	11	10
00	0	0	1	0
01	1	1	0	1
11	1	1	0	1
10	0	0	1	0

C_4C_3 \ C_2C_1	00	01	11	10
00	0	0	0	0
01	0	0	1	0
11	1	1	0	1
10	1	1	1	1

$D_3 = C_3C_2' + C_3C_1' + C_1C_2C_3'$

$\boxed{D_3 = (C_1C_2) \oplus C_3}$

$D_4 = C_4C_3' + C_4C_2' + C_4C_1' + C_4'C_1C_2C_3$

$= C_4(C_1' + C_2' + C_3') + C_4'C_1C_2C_3$

$= C_4(C_1C_2C_3)' + (C_1C_2C_3)C_4'$

$\boxed{D_4 = (C_1C_2C_3) \oplus C_4}$

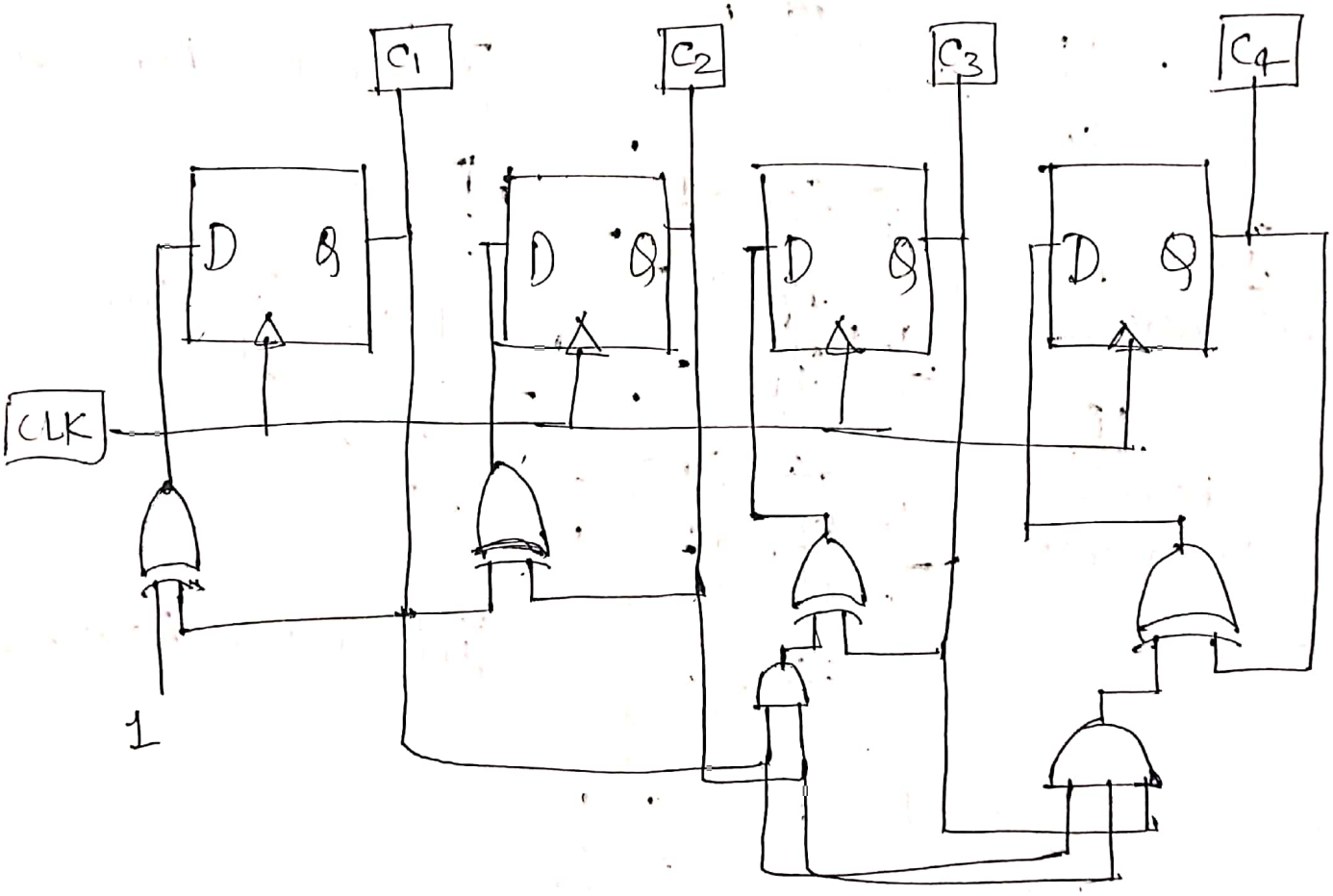


Fig. 4 bit D counter (synchronous)

Q.2) Design a register file (16 X 16) with two read ports and one write port. Perform read and write operation and understand the operation of register file.

D ⇒ IP of register

en ⇒ Enabler.

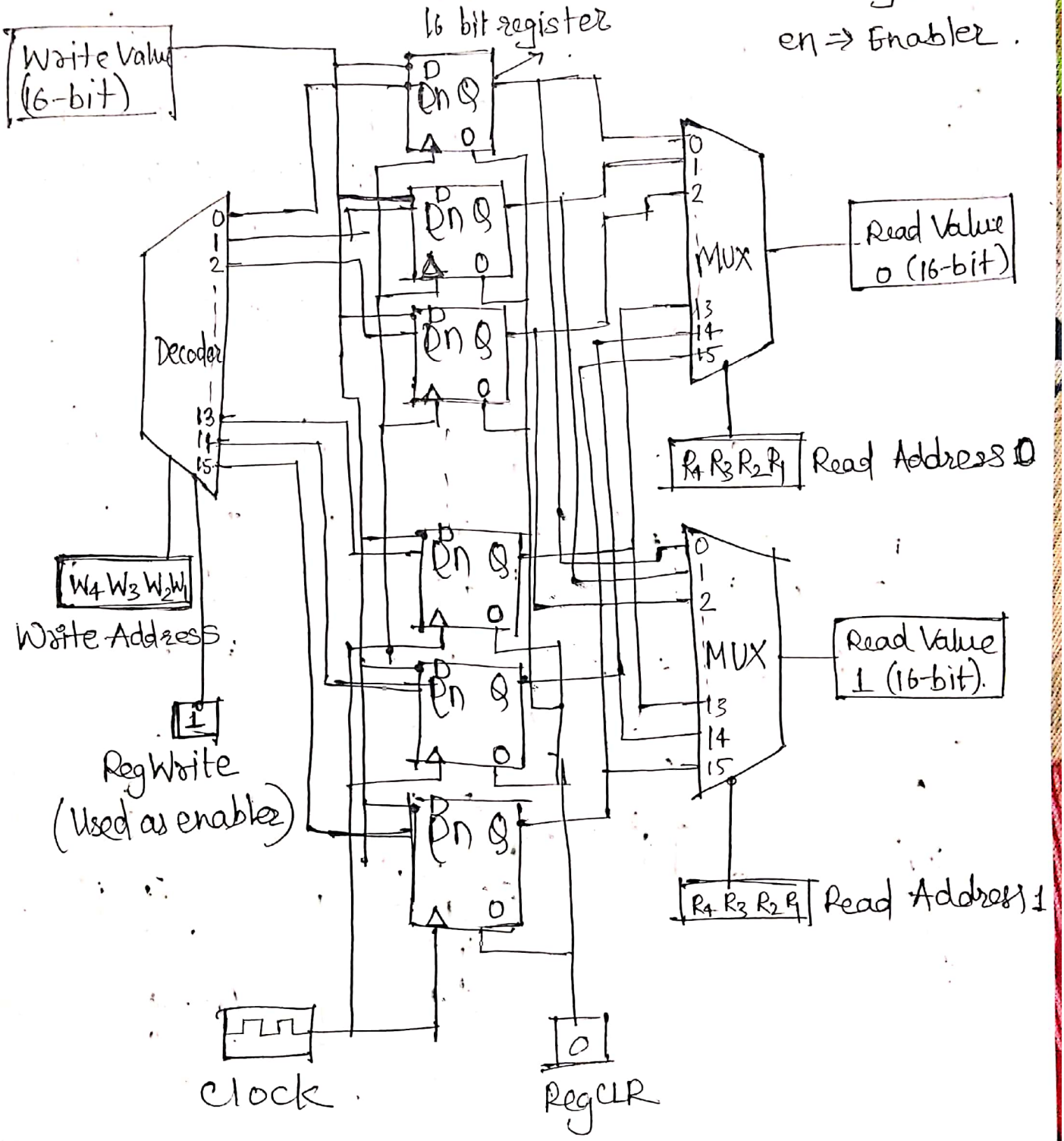


Fig. 16 X 16. Register file with two read ports and one write port.

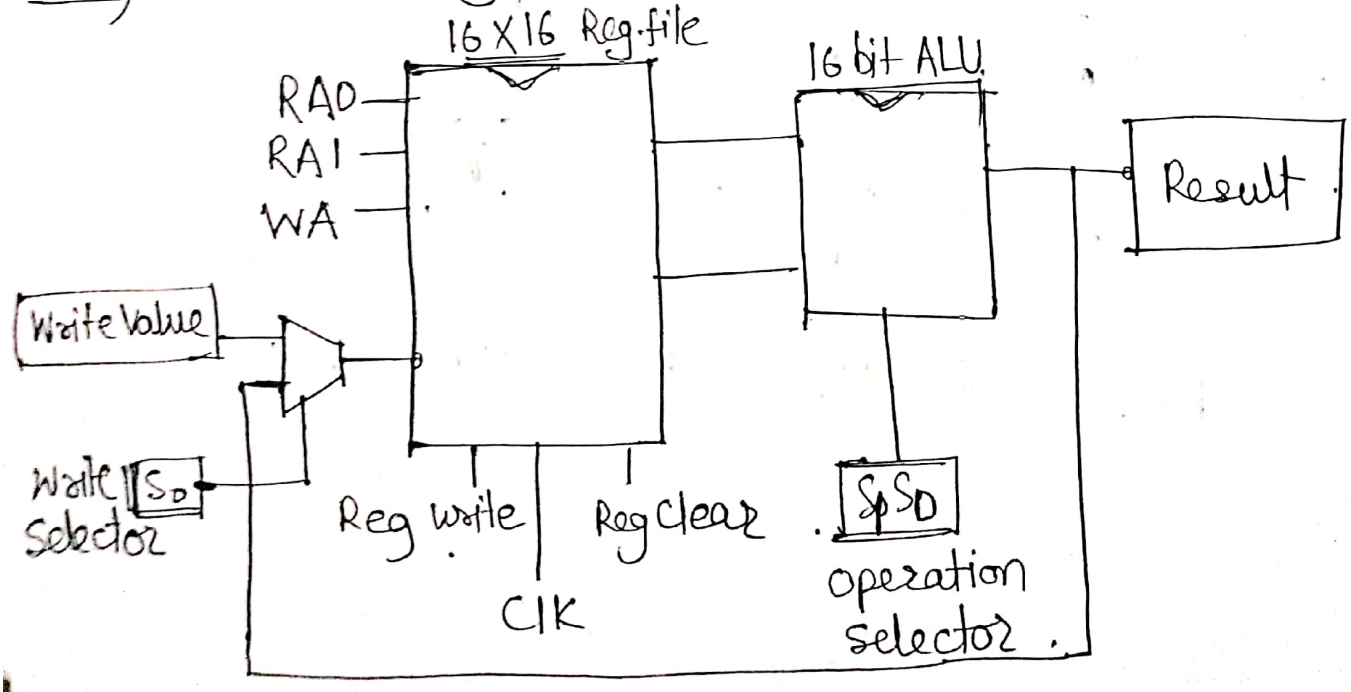
Write Operation :-

① Put the value to be written in Write Value.
② Mention Write Address corresponding to the register where you want to store the value.
③ Set RegWrite (enabler) for Decoder to 1.
④ Generate a clock pulse. At the rising edge, value gets stored (written).

Read Operation :-

① Mention Read Address corresponding to the register, where you want to read value from.
② Generate a clock pulse, at the rising edge of which value is read and displayed.

Q.3) Combine register file and ALU



Write Selector

S_0	Operation
0	→ Writes Input Value to Write Addr
1	→ Writes Result to Write Addr

Operation Selector.

S_0	S_1	Operation
0	0	→ Addition
0	1	→ Subtraction
1	0	→ Bitwise AND
1	1	→ Bitwise OR .

16-bit ALU.

16-bit Reg. file ⇒ same as question 2.

16-bit ALU.

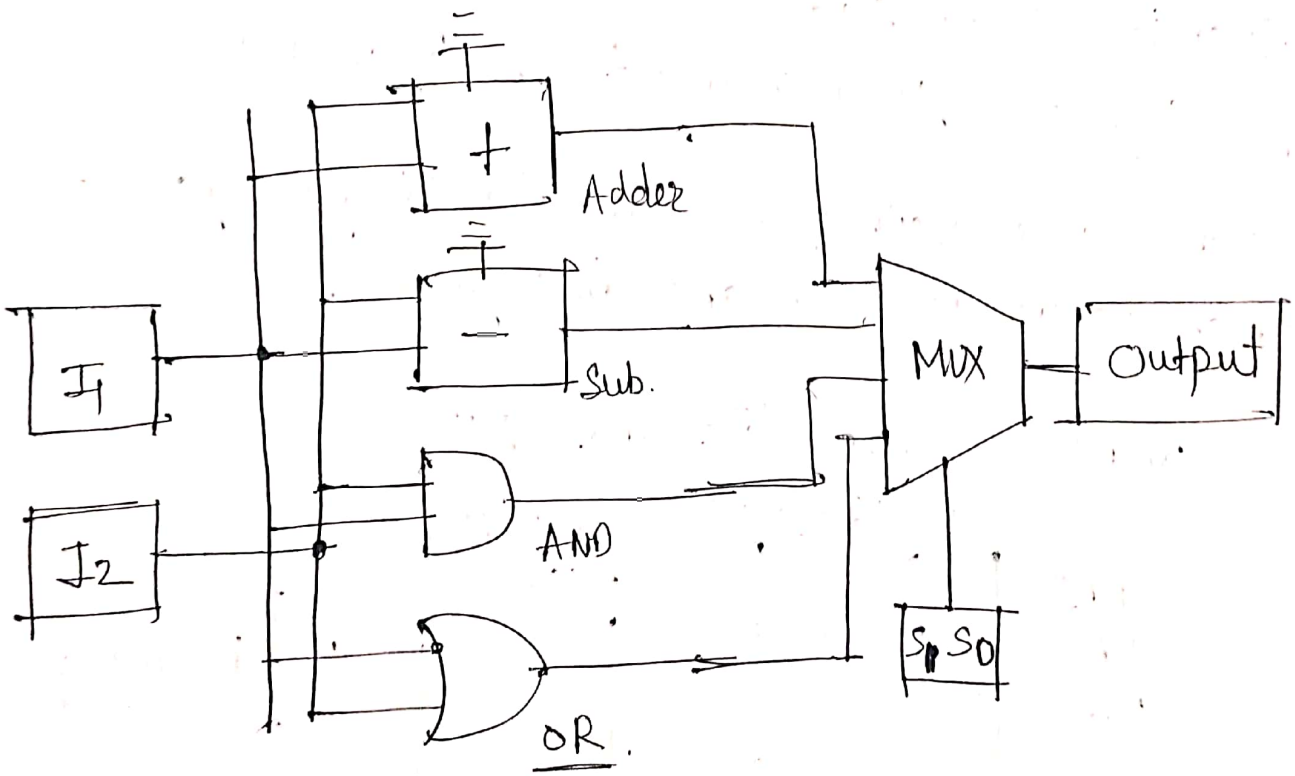


Fig. 16-bit ALU. (4 operations)

Write 1, 2, ——, 10 data to registers (1 to 10) and compute sum.

→ Steps

Write Selector = 0 ; RegWrite = 1. for every read and write operation we have to give a clock pulse.

(I)

Write Address	Write Value
0001	0001
0010	0010
0011	0011
0100	0100
0101	0101
0110	0110
0111	0111
1000	1000
1001	1001
1010	1001.

Write Selector = 1.

(II).

Read Ad 0	Read Ad 1	Result (in decimal).
0000.	0001.	1 → write it
1011	0010.	3 to 1011
1011	0011	6 —//—
1011	0100	10 —//—
1011	0101	15 —//—
1011	0110	21 —//—
1011	0111	28 —//—
1011	1000	36 —//—
1011	1001	45 —//—
1011	1010	55 —//—.

final answer