

Mobile Number: 9815137777

GSTIN: 07ADEPG27191L2P

Bill No.-PI-11

Date of Issue: 22-09-22

IUT NO- AD070322026014R

## Details of Receiver (Shipped to)

Country of Origin : INDIA

Port of Loading- JOGBANI

CUSTOM OFFICE-BIRATNAGAR

Name: JAI DANTAKA MOTOR CYCLE PARTS

Address: VADA NO.7, BIRATNAGAR, NAGARPALIKA, DHARAN

Mobile Number: 9801959416

email ID:jkmabajspur2314@gmail.com

Country: NEPAL

EXIM CODE: 600151460012-13

PAN NO.: 600151649

INCOTERM- FOB DELHI INDIA

S No.	Part No.	Description of goods/ services.	HSN / SAC	MRP IN RS	Quantity	Net Rate IN RS	Amount IN RS
1	30151069	SPRING	73201011	6.20	300.000	4.154	1246.20
2	30151185	SHOE BRAKE CAL 115 / PULS	87149400	223.00	50.000	138.26	6913.00
3	36DE0046	KIT CHAIN SPROCKET	87141090	1923.00	10.000	1288.41	12884.10
4	36DV0011	ST. CONE (37MM) WITH GREASE	84821011	414.00	50.000		
5	36H10007	CYL BLOCK ASSY PULSR200	84099191	3279.00	3.000	2032.98	6098.94
6	36H10007	KIT FRIC PLATES KTM 200	84099199	1146.00	5.000	710.52	3552.60
7	39097315	NUT HEXAGON M5 X 0.5	73181600	4.80	200.000	3.216	643.20
8	39251105	STUD M10	73181900	67.00	40.000	44.89	1795.60
9	56H10114	IL HOLDER STEP RH SLATE GREY	87141090	718.00	7.000	485.16	3116.12
10	DF101055	SCREW TAPPET	73181900	14.50	200.000	9.715	1943.00
11	DG101498	GASKET	84841090	2.60	50.000	1.742	87.10
12	DG151009	CASE METER GEAR ELIMINATO	87141090	197.00	30.000	122.14	3664.20
13	DG171015	SLEEVE	87149290	94.00	40.000	58.28	2331.10
14	DG171034	GUARD	40169990	146.00	50.000	97.82	4891.00
15	DH101059	CHAIN CAM	73151100	189.00	100.000	126.63	12663.00
16	DH101131	GASKET COVER RH	40169340	62.40	200.000	41.808	8361.00
17	DH101391	LEVER CHANGE	87141090	370.00	20.000	229.4	4588.00
18	DH152101	ASSLY BODY BALANCER	87141090	816.00	6.000	505.92	3035.52
19	DH141306	GROMMET SLEEVE SPARK PLU	87149950	26.00	50.000	16.12	806.00
20	DH131204	CDI NON TPS ASSEMBLY K1	85118000	1178.00	10.000	730.36	7303.60
21	DH161405	HOLDER STEP RH	87141090	1219.00	6.000	755.78	4534.68
22	DH201090	SENSOR SPEED-CABLE	85361090	203.00	80.000	136.01	10880.80
23	DJ161064	GASKET EXHAUST	74152100	49.00	40.000	32.83	1313.20
24	DJ161371	BLOCK PISTON (DI FO	87141090	3387.00	2.000	2099.94	4199.88
25	DJ111093	DC CDI UNIT	85112090	1311.00	5.000	812.82	4064.10
26	DH151077	WHEEL 1.6 X 17 DC BLACK	87141090	5628.00	1.000	3489.36	3489.36
27	DJ151071	DISK PAD	87149400	171.00	100.000	106.02	10602.00
28	DJ151089	COUPLING RUBBER	40169990	155.00	50.000	103.85	5192.50
29	DJ181087	SEAL OIL FORK	87149100	67.00	50.000	41.54	2077.00
30	DJ181104	SPRING MAIN WITH SPACER T	73209090	79.00	50.000	52.93	2646.50
31	DJ191082	LEVER & SUPPORT HOLDER ASSY	87149990	474.00	30.000	293.88	8816.40
32	DJ201302	REGULATOR ASSY	85044040	643.00	20.000	430.81	8616.20
33	DK101064	CAM CHAIN LOG LINKS	73151100	223.00	30.000	152.76	4582.80
34	DK101330	GUIDE CHAIN KIT	40169990	106.00	20.000	71.02	1420.40
35	DK101357	RING O COLLAR SPROCK	40169990	2.60	229.000	1.742	398.92
36	DK101614	CAMSHAFT (BLACK)	84831099	623.00	15.000	417.41	6261.15
37	DK101748	SHAFT ROCKER ARM INTAKE	87141090	26.00	19.000	16.12	306.28
38	DK101776	ASSY BODY BALANCER	87141090	951.00	10.000	589.62	5896.20
39	DK121097	CARBURETTOR ASSLY K3 EVAP	84099192	3015.00	2.000	1869.3	3738.60
40	DK151036	CONNECTING LINK O	73159000	25.00	100.000	16.75	1675.00
41	DK181024	BRACKET UNDER ASSY	87149100	1175.00	6.000	728.5	4371.00
42	DK181034	LOCK SET 4 IN 1	83012000	924.00	6.000	619.08	3714.48
43	DK181070	FORK PIPE SPINNING	87149100	1028.00	20.000	637.36	12747.20
44	DK201021	FUSE ASSLY BLADE TY	85443000	9.00	60.000	6.03	361.80
45	DK201078	RALAY STARTER	85364100	358.00	20.000	239.86	4797.20
46	DS101096	CAP CAMSHAFT	40169990	35.00	50.000	23.45	1172.50
47	DT131826	CALIPER ASSLY	87149400	2067.00	4.000	1244.34	4977.36
48	DT1232601	LOCK SET 3 IN 1	83012000	853.00	5.000	571.51	2857.55
49	DT551407	PLATE (HIGH PRESSURE	87141090	185.00	10.000	114.7	1147.00
50	DU101221	PLUG DRAIN MAGNETIC	73181500	26.00	100.000	17.42	1742.00
51	DV171071	SHOCK ABSORBER AVENGER	87141090	1103.00	12.000	683.86	8206.32



## Performa Invoice

Name:AJAY ENTERPRISES Address:2126/59, LAXMI BHAWAN, NAIWALA, Karol Bagh ,New Delhi-110005 Email ID- ajaygupta0126@gmail.com Mobile Number :9811552277 GSTIN: 07ADEPG2719F1ZX Bill No.-PI-11 Date of Issue: 22-09-22	I E Code : 500022798  LUT NO- AD070322026014R						
Details of Receiver (Shipped to) Name: JAI DANTAKALI MOTORCYCLE PARTS Address: VADA NO.7, BHOTEPUL, NAGARPALIKA, DHARAN Mobile Number: 9807059405 email ID-jkmbajajspear2214@gmail.com Country: NEPAL EXIM CODE: 6001514690125NP PAN NO.: 600151649	Country of Origin : INDIA Port of Loading- JOGBANI CUSTOM OFFICE-BIRATNAGAR  INCOTERM- FOB DELHI INDIA						
S No	Part No.	Description of goods/ services	HSN/ SAC	MRP IN RS	Quantity	Net Rate IN RS	Amount IN RS
1	30151069	SPRING	73201011	6.20	300.000	4.154	1246.20
2	30151105	SHOE BRAKE CAL 115 / PULS	87149400	223.00	50.000	138.26	6913.00
3	36DK0040	KIT CHAIN SPROCKET	87141090	1923.00	10.000	1288.41	12884.10
4	36DV0011	KIT ST. CONE (37MM) WITH GREASE	84821011	414.00	50.000	277.38	13869.00
5	36JL0007	CYL BLOCK ASSY PULSR200	84099191	3279.00	3.000	2032.98	6098.94
6	36JU0002	KIT FRIC PLATES KTM 200	84099199	1146.00	5.000	710.52	3552.60
7	39097315	NUT HEXAGON M5 X 0.5	73181600	4.80	200.000	3.216	643.20
8	39253105	STUD M10	73181900	67.00	40.000	44.89	1795.60
9	56JL3Y1Y	JL HOLDER STEP RH SLATE GREY	87141090	718.00	7.000	445.16	3116.12
10	DF101055	SCREW TAPPET	73181900	14.50	200.000	9.715	1943.00
11	DG101498	GASKET	84841090	2.60	50.000	1.742	87.10
12	DG151009	CASE METER GEAR ELIMINATO	87141090	197.00	30.000	122.14	3664.20
13	DG171016	SLEEVE	87149290	94.00	40.000	58.28	2331.20
14	DG171034	GUARD	40169990	146.00	50.000	97.82	4891.00
15	DH101050	CHAIN CAM	73151100	189.00	100.000	126.63	12663.00
16	DH101154	GASKET COVER RH	40169340	62.40	200.000	41.808	8361.60
17	DH101397	LEVER CHANGE	87141090	370.00	20.000	229.4	4588.00
18	DH102104	ASSLY BODY BALANCER	87141090	816.00	6.000	505.92	3035.52
19	DH111016	GROMMET SLEEVE SPARK PLU	87149990	26.00	50.000	16.12	806.00
20	DH11104	CDI NON TPS ASSEMBLY K1	85118000	1178.00	10.000	730.36	7303.60
21	DH161405	HOLDER STEP RH	87141090	1219.00	6.000	755.78	4534.68
22	DH201080	SENSOR SPEED-CABLE	85361090	203.00	80.000	136.01	10880.80
23	DJ101064	GASKET EXHAUST	74152100	49.00	40.000	32.83	1313.20
24	DJ101371	BLOCK PISTON (DJ FO	87141090	3387.00	2.000	2099.94	4199.88
25	DJ111092	DC CDI UNIT	85112090	1311.00	5.000	812.82	4064.10
26	DH151077	WHEEL 1.6 X 17 DC BLACK	87141090	5628.00	1.000	3489.36	3489.36
27	DJ151071	DISK PAD	87149400	171.00	100.000	106.02	10602.00
28	DJ151088	COUPLING RUBBER	40169990	155.00	50.000	103.85	5192.50
29	DJ181087	SEAL OIL FORK	87149100	67.00	50.000	41.54	2077.00
30	DJ181104	SPRING MAIN WITH SPACER T	73209090	79.00	50.000	52.93	2646.50
31	DJ191082	LEVER & SUPPORT HOLDER ASSY	87149990	474.00	30.000	293.88	8816.40
32	DJ201202	REGULATOR ASSY	85044040	643.00	20.000	430.81	8616.20
33	DK101064	CAM CHAIN 100 LINKS	73151100	228.00	30.000	152.76	4582.80
34	DK101330	GUIDE CHAIN KIT	40169990	106.00	20.000	71.02	1420.40
35	DK101352	RING O COLLAR SPROCK	40169990	2.60	229.000	1.742	398.92
36	DK101614	CAMSHAFT (BLACK)	84831099	623.00	15.000	417.41	6261.15
37	DK101748	SHAFT ROCKER ARM INTAKE	87141090	26.00	19.000	16.12	306.28
38	DK101776	ASSY BODY BALANCER	87141090	951.00	10.000	589.62	5896.20
39	DK121092	CARBURETTOR ASSLY K3 EVAP	84099192	3015.00	2.000	1869.3	3738.60
40	DK151036	CONNECTING LINK O	73159000	25.00	100.000	16.75	1675.00
41	DK181024	BRACKET UNDER ASSY	87149100	1175.00	6.000	728.5	4371.00
42	DK181034	LOCK SET 4 IN 1	83012000	924.00	6.000	619.08	3714.48
43	DK181070	FORK PIPE SPINNING	87149100	1028.00	20.000	637.36	12747.20
44	DK201023	FUSE ASSLY BLADE TY	85443000	9.00	60.000	6.03	361.80
45	DK201078	RALAY STARTER	85364100	358.00	20.000	239.86	4797.20
46	DS101096	CAP CAMSHAFT	40169990	35.00	50.000	23.45	1172.50
47	DT131826	CALIPER ASSLY	87149400	2007.00	4.000	1244.34	4977.36
48	DT232601	LOCK SET 3 IN 1	83012000	853.00	5.000	571.51	2857.55
49	DT551402	PLATE CLUTCH PRESSURE	87141090	185.00	10.000	114.7	1147.00
50	DU101223	PLUG DRAIN MAGNETIC	73181500	26.00	100.000	17.42	1742.00
51	DV171001	SHOCK ABSORBER RR AVENGER	87141090	1103.00	12.000	683.86	8206.32



52	DV201010	LAMP HEAD AVENGER	85122010	1816.00	5.000	1216.72	6083.60
53	JC101034	TENSIONER ASSEMBLY	84099191	207.00	60.000	128.34	7700.40
54	JC161017	STAND SIDE	87141090	216.00	25.000	133.92	3348.00
55	JE511233	CAM CHAIN K11(SILENT)	73151100	297.00	10.000	198.99	1989.90
56	JE511251	CAMSHAFT ASSLY	84831099	670.00	6.000	448.9	2693.40
57	JF402004	REGULATOR ASSLY	85044040	2000.00	4.000	1340	5360.00
58	JG351209	PLUG SPARK M12 (BOSCH-VR5NE)	85111000	288.00	30.000	178.56	5356.80
59	JL121001	MAIN SPRING	87149100	283.00	20.000	175.46	3509.20
60	JL131804	CALIPER ASSY BRAKE FRONT	87149400	1781.00	6.000	1104.22	6625.32
61	JL161200	CABLE CLUTCH	87149990	238.00	40.000	147.56	5902.40
62	JL403000	INTERLOCK RELAY	90261020	124.00	10.000	83.08	830.80
63	JL511222	CHAIN SILENT	73151100	381.00	20.000	255.27	5105.40
64	JZ401013	HEAD LAMP CONTROL MODULE	85122020	300.00	30.000	201	6030.00
65	JZ402403	INSTRUMENT CLUSTER	87141090	1721.00	1.000	1067.02	1067.02
66	PD122005	SHOCKABSORBER - A102	87141090	920.00	12.000	570.4	6844.80
67	PD232602	LOCK SET ASSEMBLY - AVENGER	83012000	1007.00	6.000	674.69	4048.14
68	YPD01007	HEADLAMP ASSY WO BULB	85122010	1142.00	5.000	765.14	3825.70
69	DG111008	SPARK PLUG	85111000	99.00	100.000	61.38	6138.00
70	36JU0036	CRANK ASSLY	84831099	3678.00	2.000	2464.26	4928.52
		TOTAL					323986.76

Amount in Words : Three Lakhs, Twenty Three Thousand , Nine Hundred, Eighty Six Rupees and Seventy Six Paise.

Payment : 100 % Advance

ALL PRICES IN INR

Declaration:

The Supply is meant for Export on LUT

100 % ADVANCE PAYMENT

SWIFT DETAILS

NAME	AJAY ENTERPRISES
BANK NAME	KARUR VYSYA BANK
BRANCH NAME	KAROL BAGH BRANCH
ACCOUNT NO	4101115000008063
IFSCODE	KVBL0004101



FOR AJAY ENTERPRISES

PROP

Proprietor

KVBLINBBIND
THE KARUR VYSYA BANK LTD.
INTERNATIONAL DIVISION, CHENNAI

## Contact

srkamatya007@gmail.com

[www.linkedin.com/in/suyan-man-amatya-bb138a204](https://www.linkedin.com/in/suyan-man-amatya-bb138a204) (LinkedIn)  
[srkamatya007.wixsite.com/website](http://srkamatya007.wixsite.com/website)  
(Personal)

## Top Skills

C (Programming Language)  
Python (Programming Language)  
Communication

# Suyan Man Amatya

Programmer—> C || Cpp

Kathmandu, Bāgmatī, Nepal

## Experience

Acme Engineering College  
Admin  
April 2021 - Present (1 year 7 months)

---

## Education

Purbanchal University  
1st year , Bachelor of Engineering -BE, Computer · (2021)

Kathmandu University (KU)  
High school, Physical Science · (June 2018 - October 2020)



# ACME Club Of IT

## ACME Engineering College

Sitapaila, Kathmandu

---

DATE:

.....  

---

✉ acit@acme.edu.np

☎ 984687-3239

📍 Sitapaila, Kathmandu



# Registers

A **Register** is a collection of flip flops. A flip flop is used to store single bit digital data. For storing a large number of bits, the storage capacity is increased by grouping more than one flip flops. If we want to store an n-bit word, we have to use an n-bit register containing n number of flip flops.

The register is used to perform different types of operations. For performing the operations, the CPU use these registers. The faded inputs to the system will store into the registers. The result returned by the system will store in the register.

## Fetch:

- ▶ It is used
  - To take the instructions given by the users.
  - To fetch the instruction stored into the main memory.

## Decode:

- ▶ The decode operation is used to interpret the instructions. In decode, the operation performed on the instructions is identified by the [CPU](#). In simple words, the decode operation is used to decode the instructions.

## Execute:

- ▶ The execution operation is used to store the result produced by the [CPU](#) into the memory. After storing this result, it is displayed on the user screen.

# Shift register

A group of flip flops which is used to store multiple bits of data and the data is moved from one flip flop to another is known as **Shift Register**.

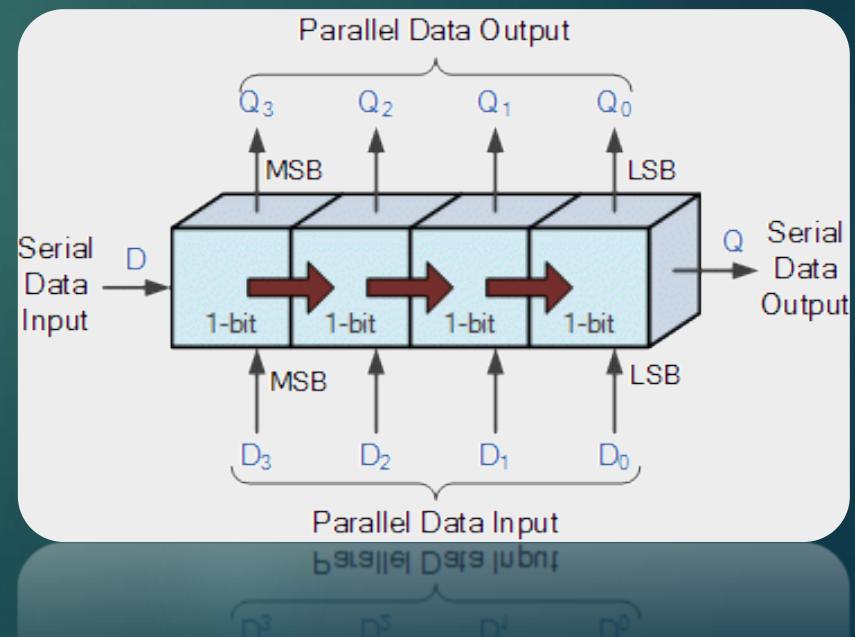
The bits stored in registers shifted when the clock pulse is applied within and inside or outside the registers. To form an n-bit shift register, we have to connect n number of flip flops. So, the number of bits of the binary number is directly proportional to the number of flip flops.

The flip flops are connected in such a way that the first flip flop's output becomes the input of the other flip flop.

A register capable of shifting its binary contents either to the left or right is called shift register.

The shift register is classified into the following types:

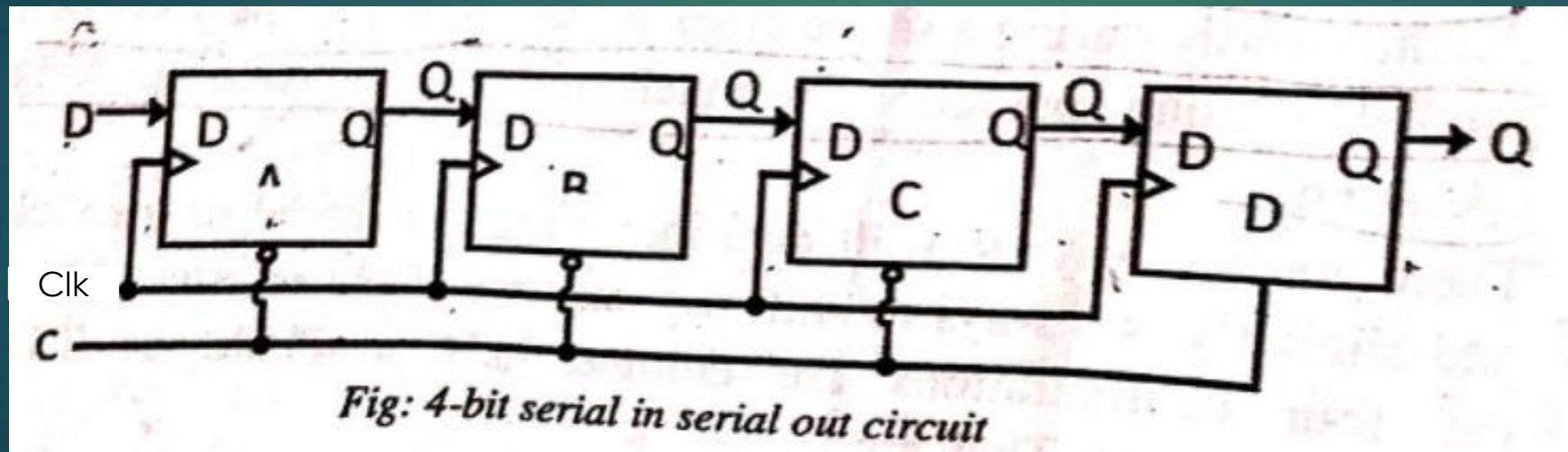
- ▶ Serial In Serial Out (SISO)
- ▶ Serial In Parallel Out(SIPO)
- ▶ Parallel In Serial Out(PISO)
- ▶ Parallel In Parallel Out(PIPO)



# Types

- ▶ SISO shift register

In "Serial Input Serial Output", the data is shifted "IN" or "OUT" serially. In SISO, a single bit is shifted at a time in either right or left direction under clock control



# Truth table

clock	Register content				
	QA	QB	QC	QD	
Initially	0	0	0	0	
CLK1	1	0	0	0	
CLK2	1	1	0	0	
CLK3	0	1	1	0	
CLK4	1	0	1	1	All data stored after 4 <sup>th</sup> clock
CLK5	0	1	0	1	
CLK6	0	0	1	0	
CLK7	0	0	0	1	
CLK8	0	0	0	0	Register clear after 8 <sup>th</sup> clock

Timing pulse	$Q_A$	$Q_B$	$Q_C$	$Q_D$	Serial output at $Q_D$
Initial value	0	0	0	0	0
After 1 <sup>st</sup> clock pulse	1	0	0	0	0
After 2 <sup>nd</sup> clock pulse	1	1	0	0	0
After 3 <sup>rd</sup> clock pulse	0	1	1	0	0
After 4 <sup>th</sup> clock pulse	1	0	1	1	1

## Timing Diagram

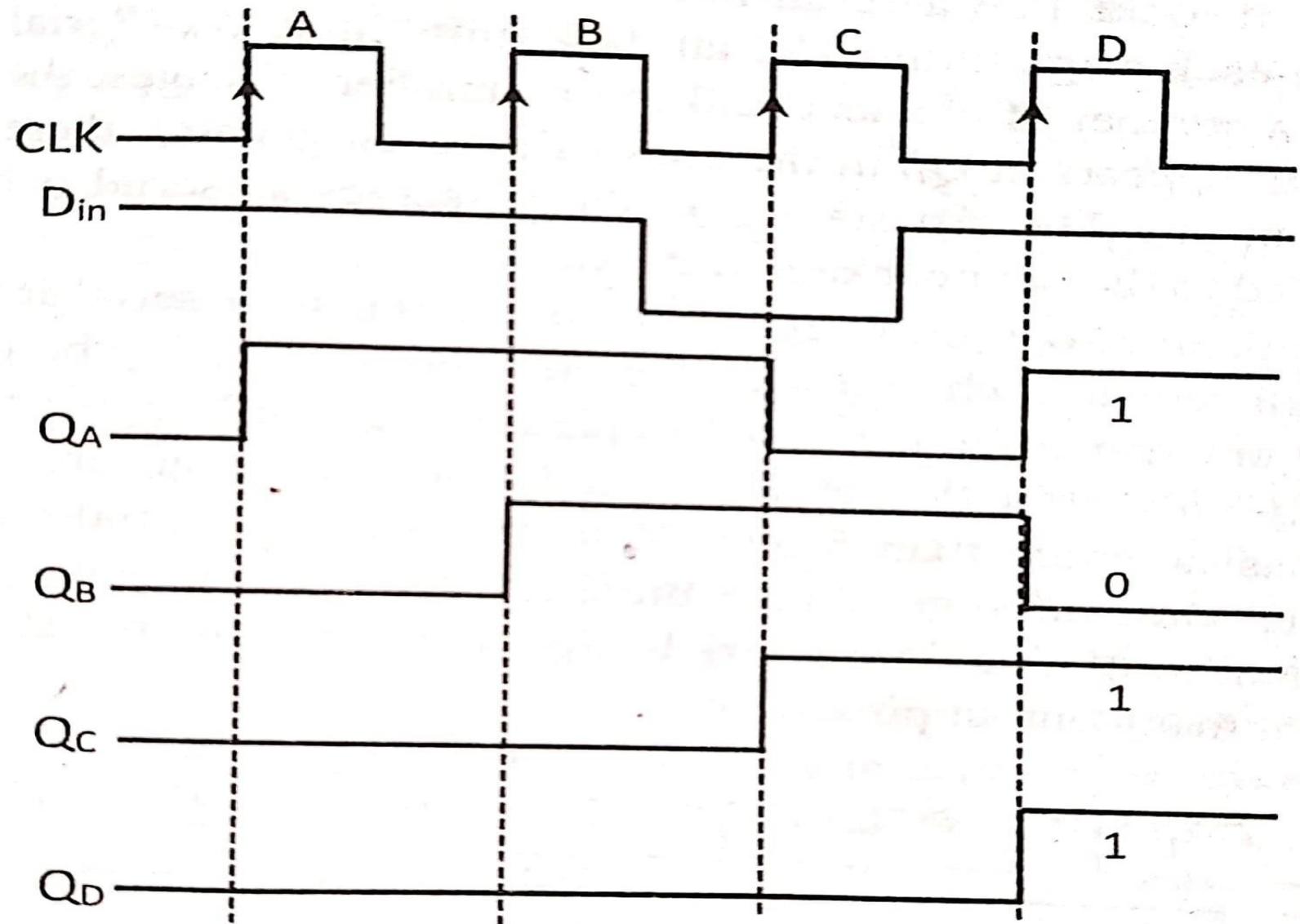
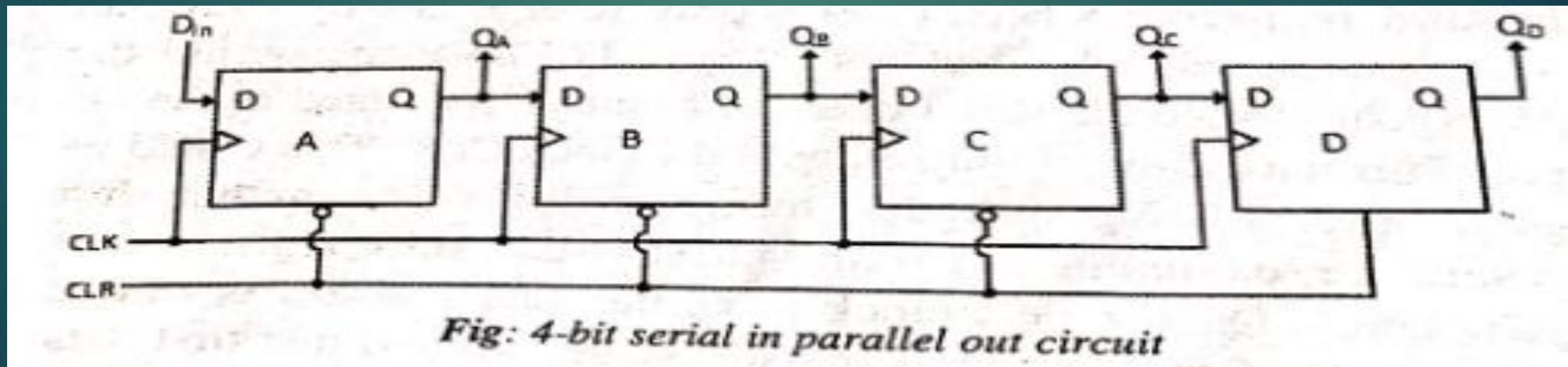


Fig: waveforms of 4-bit serial input shift register

# SIPO shift register

In the "**Serial IN Parallel OUT**" shift register, the data is passed serially to the flip flop, and outputs are fetched in a parallel way. The data is passed bit by bit in the register, and the output remains disabled until the data is not passed to the data input. When the data is passed to the register, the outputs are enabled, and the flip flops contain their return value.

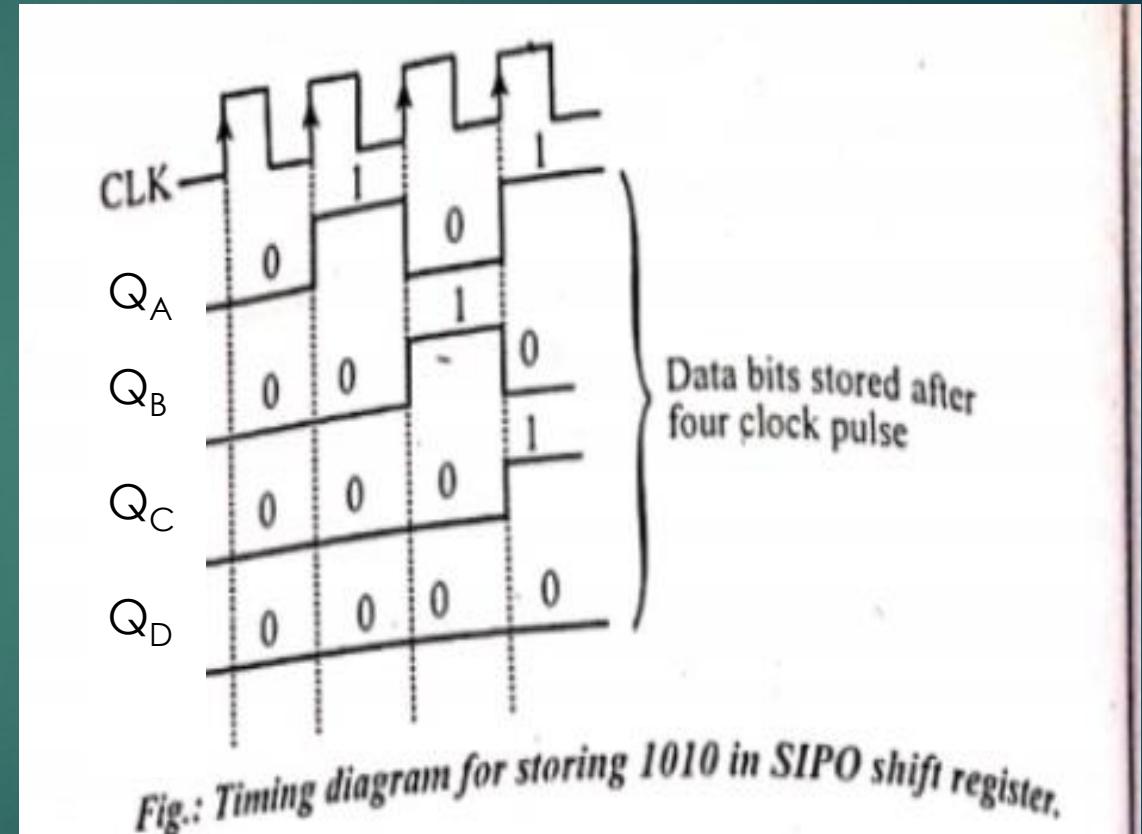


## Truth table

CLK	Q <sub>A</sub>	Q <sub>B</sub>	Q <sub>C</sub>	Q <sub>D</sub>
initially	0	0	0	0
CLK 1	0	0	0	0
CLK 2	1	0	0	0
CLK 3	0	1	0	1
CLK 4	1	0	1	0

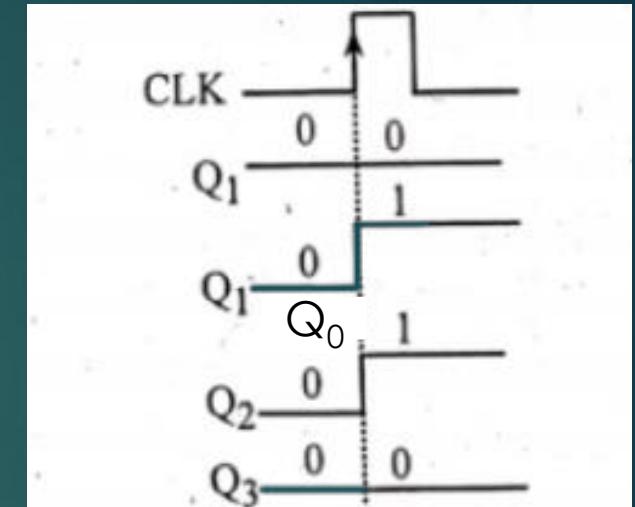
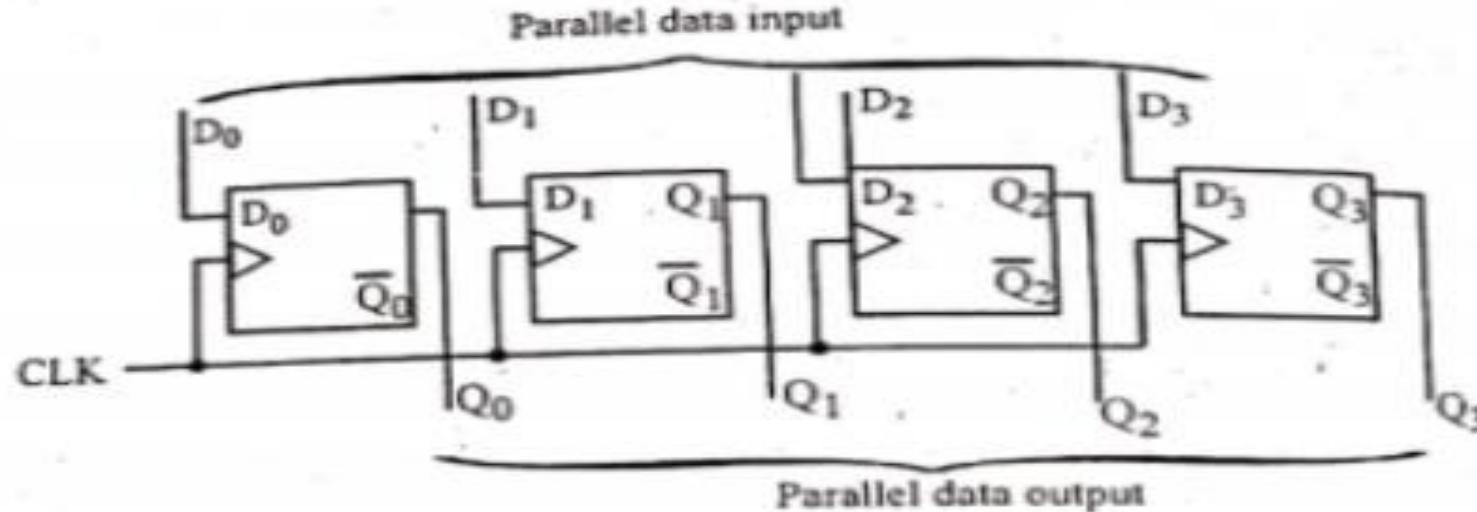
Data is completely stored after 4<sup>th</sup> clk.  
Output is available all at a time parallel.

## Timing diagram



# PIPO shift register

Timing diagram



Truth table

clock				
	Q0	Q1	Q2	Q3
Initially	0	0	0	0
clk1	0	1	1	0

# PISO shift register

4)

## Parallel in Serial Out (PISO) Shift Register

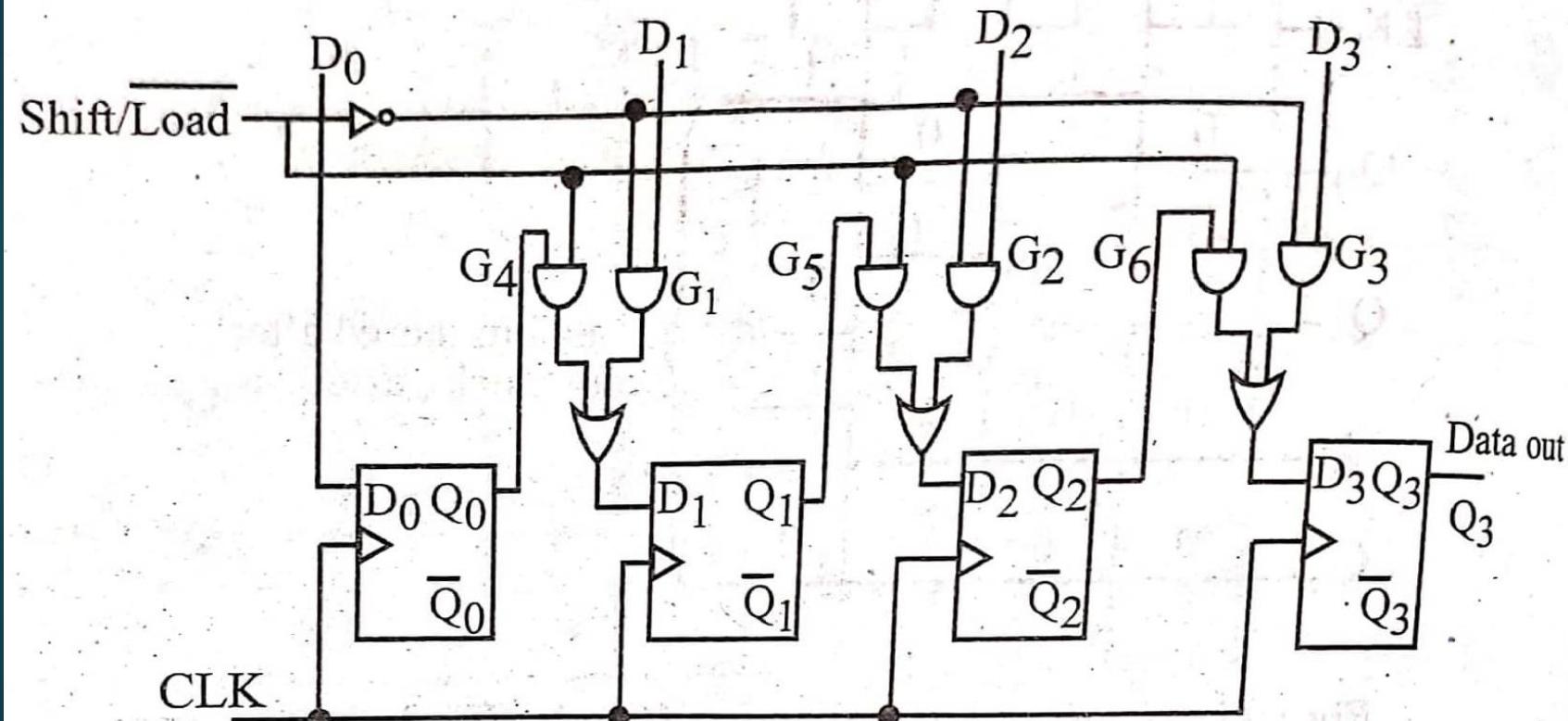
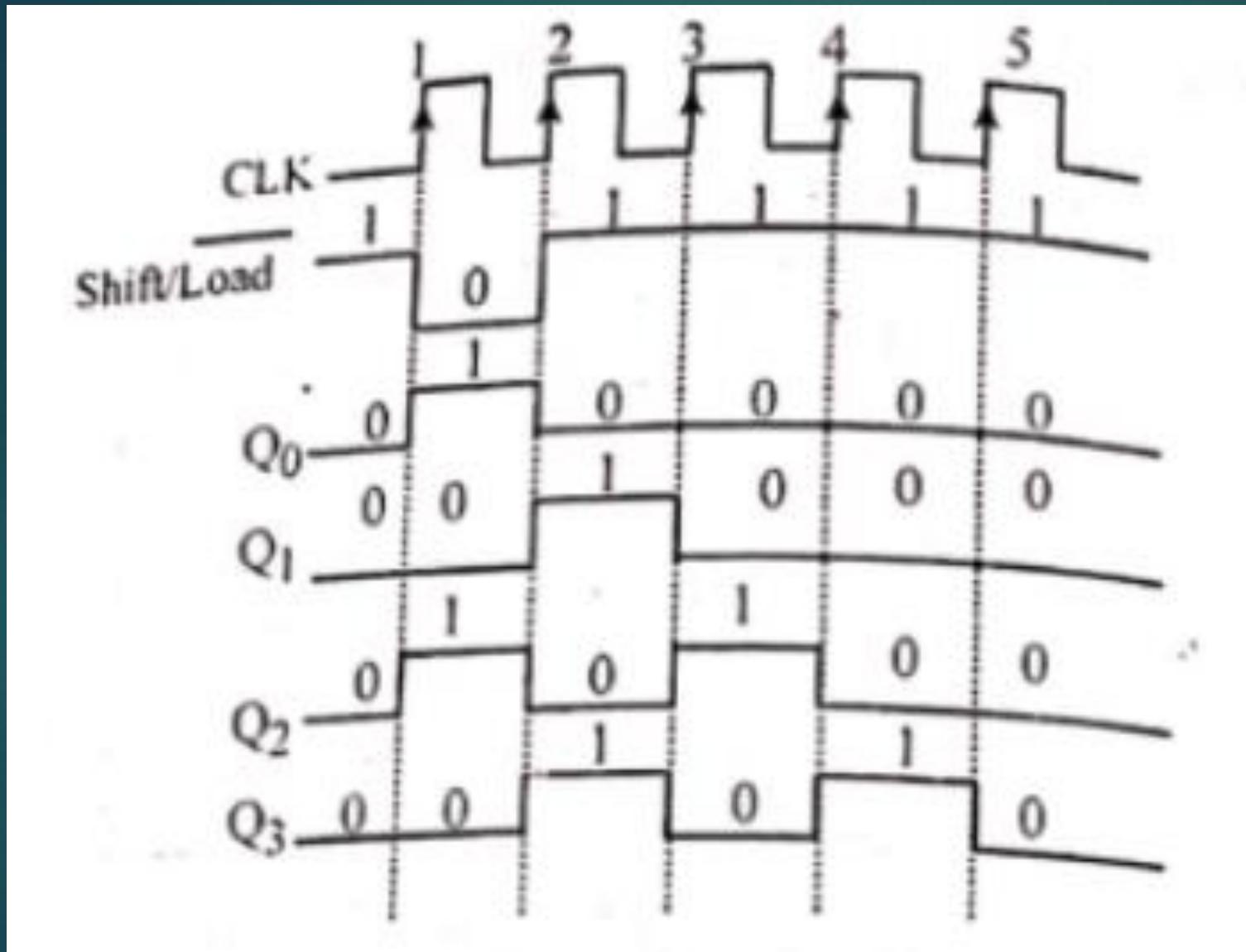


Fig.: Logic diagram of parallel in serial out (PISO) shift register.

Truth table

	$D_0$	$D_1$	$D_2$	$D_3$
Initially	0	0	0	0
Clk1	1	0	1	0
Clk2	0	1	0	1
Clk3	0	0	1	0
Clk4	0	0	0	1
Clk5	0	0	0	0



Timing diagram

# Do yourself

- ▶ Bi-directional shift register
- ▶ Universal shift register

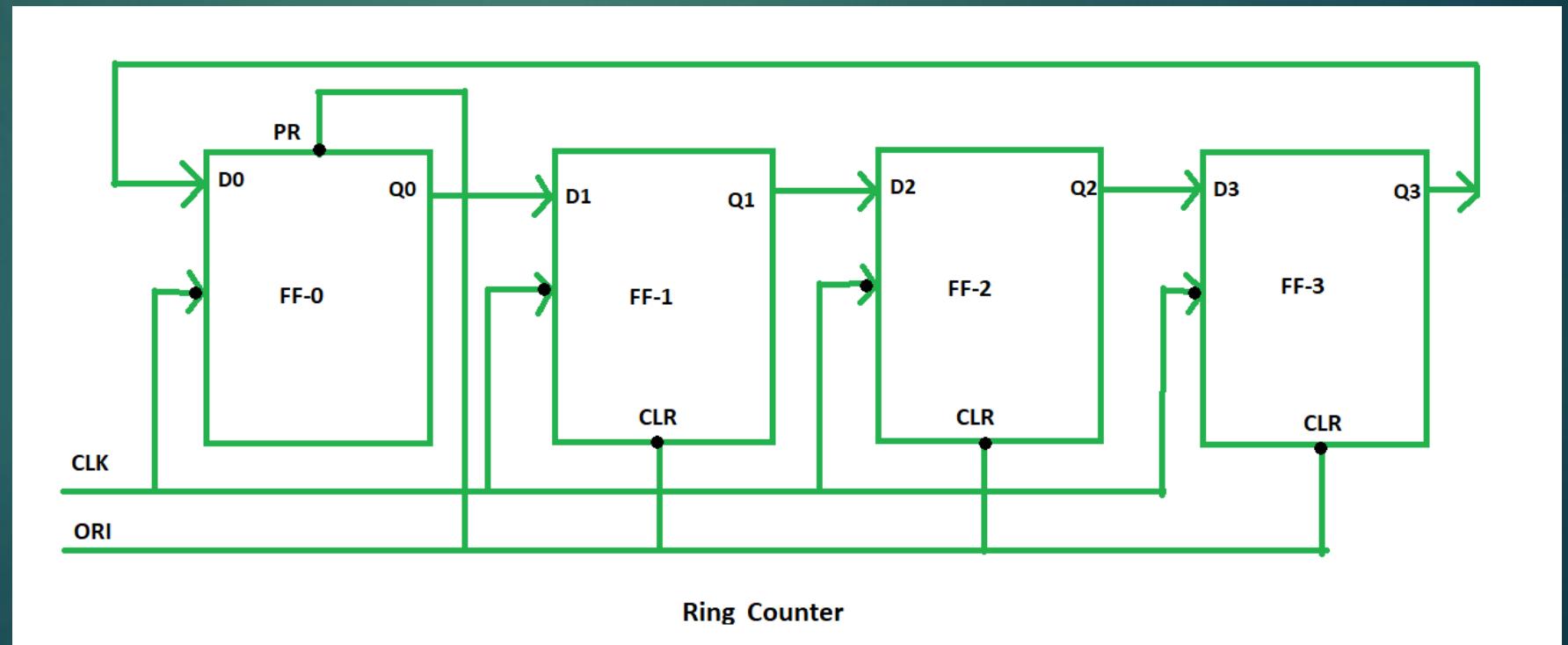
# Applications of Shift register

- ▶ To produce time delay
- ▶ It can be used to count the number of pulses entering into a system as ring counter or Johnson counter( switched tail counter)
- ▶ The serial in serial out shift register can be used as a time delay device. The amount of delay can be controlled by
  1. the no. of stages in the register
  2. The clock frequency

## a) Ring counter

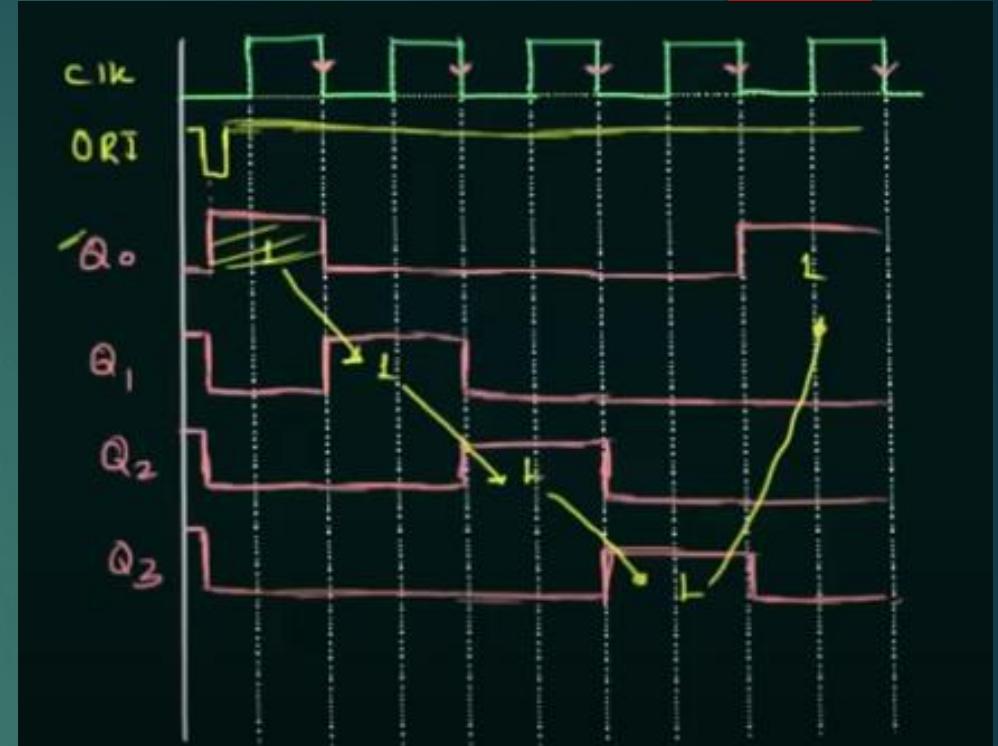
- ▶ It is a type of counter composed of a circular shift register used to count sequence of operation in sequence control of stepper motor etc.
- ▶ In ring counter the output of last flip-flop of shift register is connected to input of first flip-flop and circulates a single 1(or 0)bit around the ring.

(PR=0, Q=1)  
(clr=1, Q=0)



Truth table

ORI	CLK	Q0	Q1	Q2	Q3
Low Pulse U		1	0	0	0
1	1	0	1	0	0
1	2	0	0	1	0
1	3	0	0	0	1
1	4	1	0	0	0



## b) Johnson // twisted/ switched// tail counter

The complement of the output of the last flip-flop is connected to the input of first flip-flop

It is also called as twisted ring counter

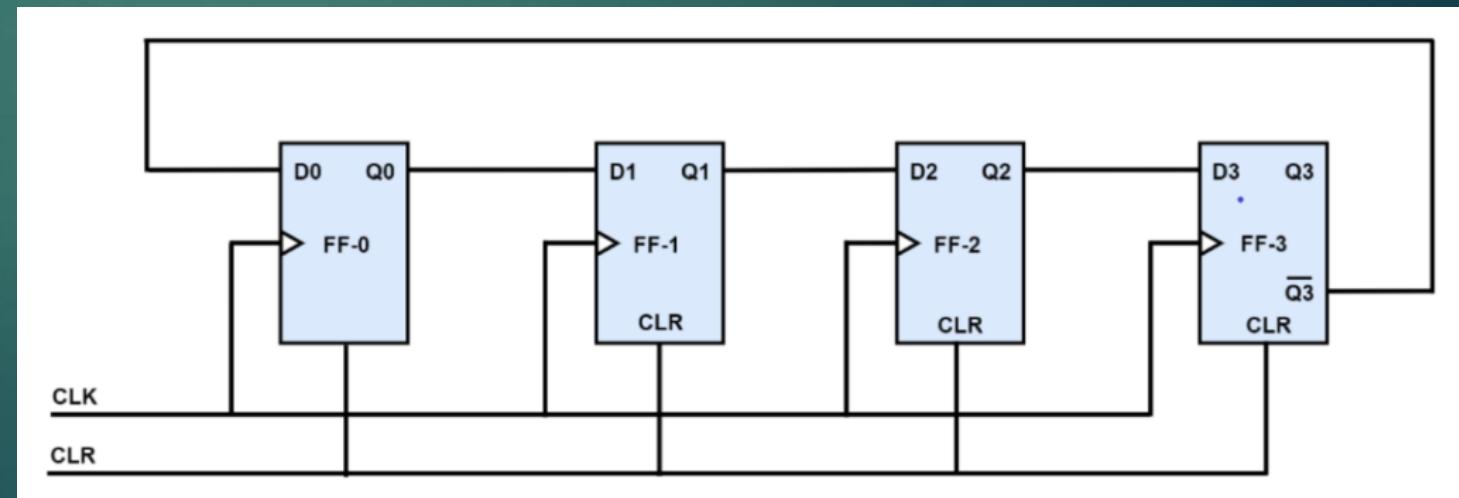
Required fewer flip-flop than ring counter but more than binary flip-flop

## Truth table

Clk	Q0	Q1	Q2	Q3
Initially	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	1	1	1	0
4	1	1	1	1
5	0	1	1	1
6	0	0	1	1
7	0	0	0	1
8	0	0	0	0

Important application of shift register are

- o For temporary data storage
- o For serial adder
- o To produce time delay
- o As ring counter
- o For Johnson counter
- o To convert serial to parallel data and vice-versa

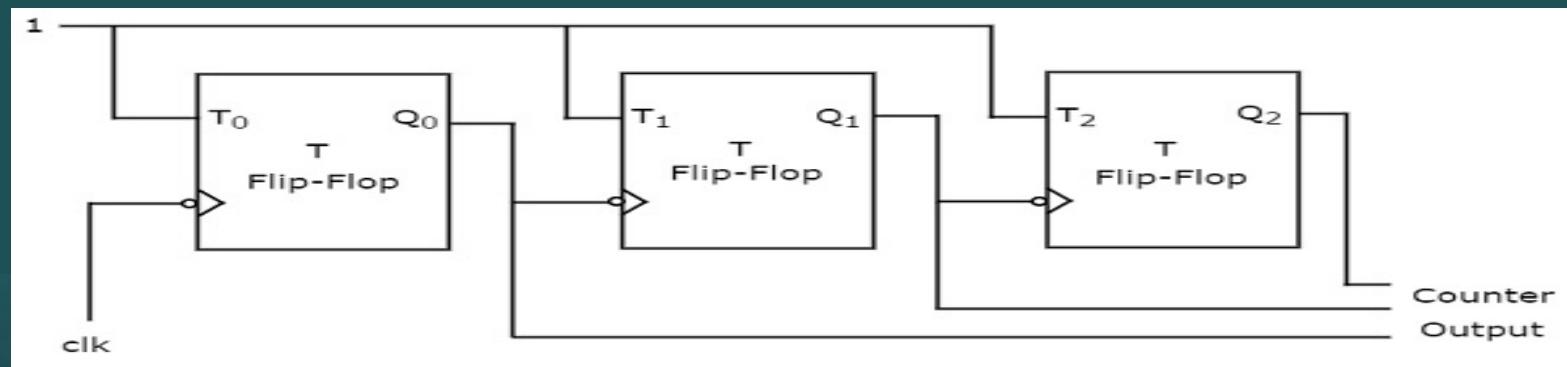


# Counter

- The combination of flip-flops that performs the counting operation are known as counters
- It is most useful and versatile sub-system in a digital system
- it is a sequential circuit that passes through pre-defined no. of states  
counters are classified into two board categories according to the way they are clocked

## i) Asynchronous counter(ripple/serial)

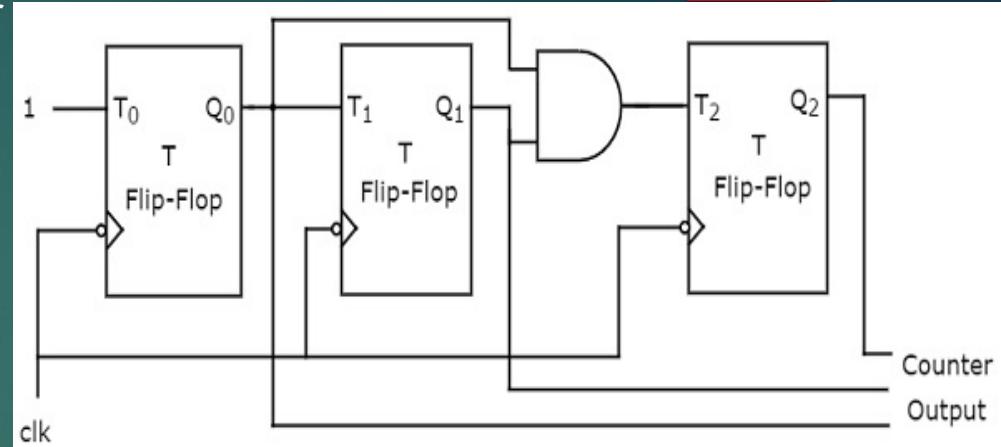
If the flip-flops do not receive the same clock signal, then that counter is called as **Asynchronous counter**. The output of system clock is applied as clock signal only to first flip-flop. The remaining flip-flops receive the clock signal from output of its previous stage flip-flop. Hence, the outputs of all flip-flops do not change affect at the same time.



## ii) Synchronous counter(parallel)

- In this counter the clock i/p is connected simultaneously. An increase in speed of operation can be achieved by use of this counter

### Difference



#### Synchronous counter

There is no connection between o/p of 1<sup>st</sup> flip-flop and clk of next flip-flop

Clked simultaneously

Circuit becomes complicated as no. of states increases

Speed is high as a clk is given at a same time.

#### Asynchronous counter

Flip-flops are connected in such a way that the output of 1<sup>st</sup> flip-flop drives the clk of next flip-flop

Flip-flops are not clked simultaneously

Circuit is simple for more no. of states

Speed is slow as clk is propagated through no. of stages

## Types of counter

- i) up counter(0,1,2,...)
- ii) down counter(10,9,...)
- iii) up-down counter

3 bit asynchronous up counter

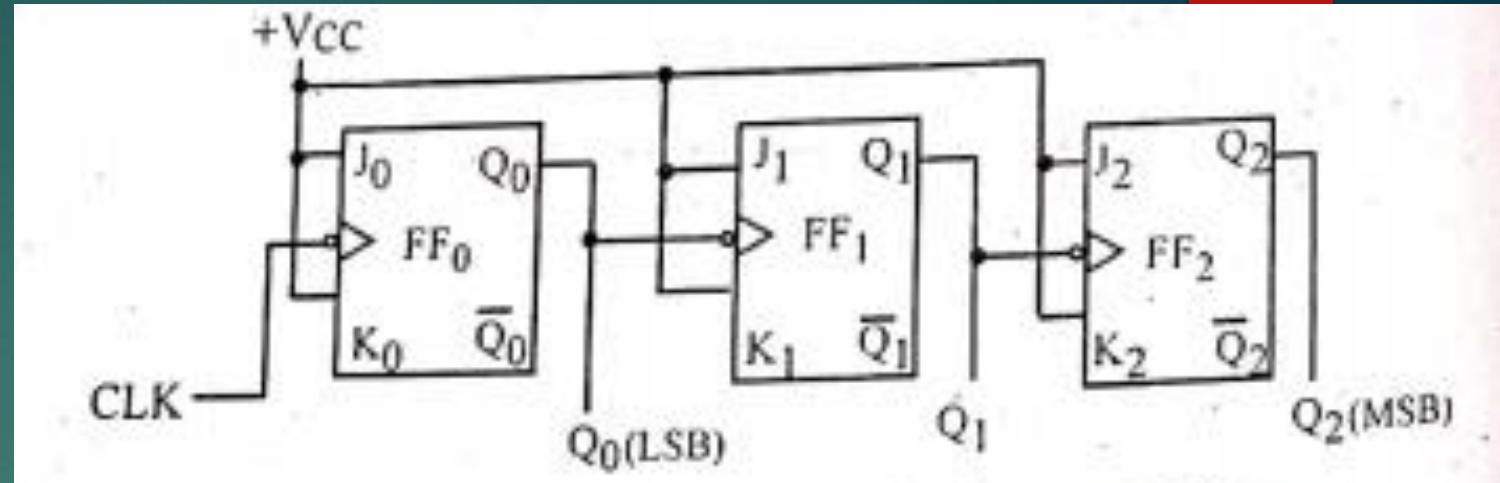


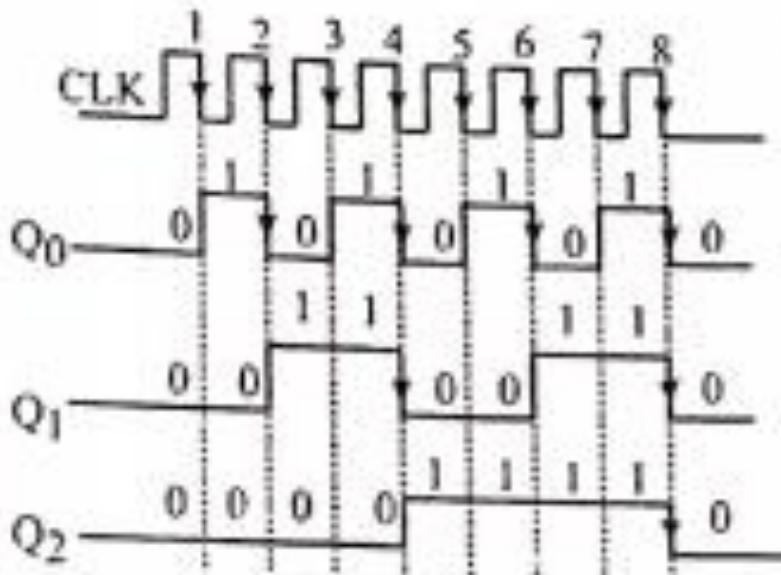
Fig.: A 3-bit binary ripple up counter

- The 3 bit ripple up counter constructed from JK flip-flop is shown above
- The JK input are tied together to +5v such that of each –ve transaction of its clock input the flip-flop toggles, its state

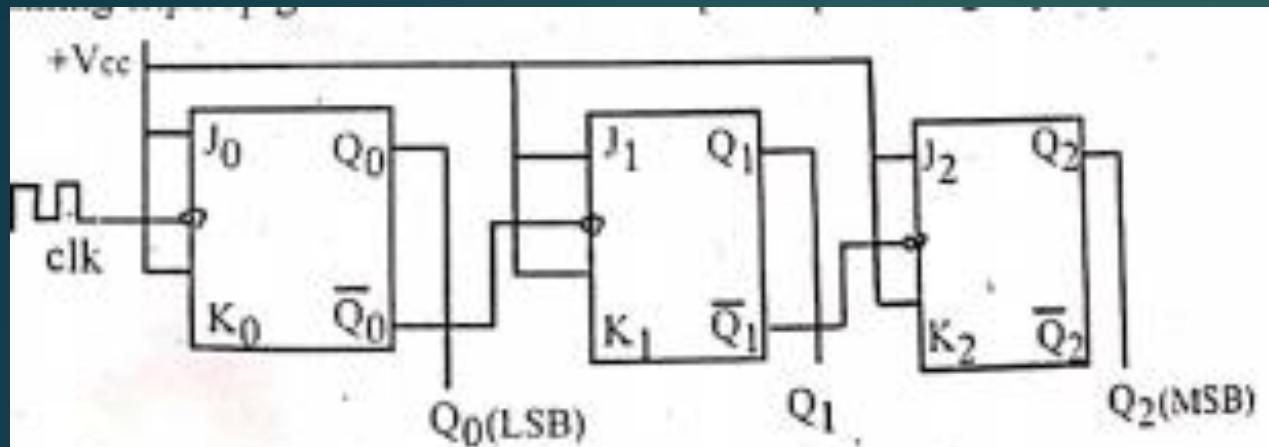
the largest binary number counted by n cascaded flip flops has a decimal equivalent of  $2^n - 1$ .  
MOD-8 counter has count of the largest binary number 111 which has decimal equivalent of  $2^3 - 1 = 7$

*Truth table*

Clock (NT)	$Q_2$	$Q_1$	$Q_0$
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
6	1	0	1
7	1	1	0
8	1	1	1



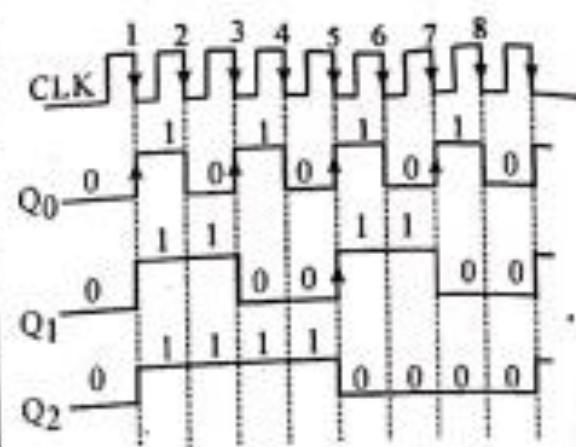
*Fig.: Timing diagram*



*Fig.: 3-bit ripple down counter*

3 bit asynchronous down counter

Truth table				
Clock (NT)	Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>0</sub>	Count
0	1	1	1	7
1	1	1	0	6
2	1	0	1	5
3	1	0	0	4
4	0	1	1	3
5	0	1	0	2
6	0	0	1	1
7	0	0	0	0



*Fig.: Waveforms*

# Classwork

- ▶ Do it for 4 bit Asynchronous up counter
- ▶ Do it for 4 bit Asynchronous down counter

### 3-bit Ripple Up/Down Counter

It is the combination of the two counter discussed previously. As from above two counter, we observe that for

(a) up-counter, output Q is connected to CLK of successive FF

(b) down-counter, output  $\overline{Q}$  is connected to CLK of successive FF

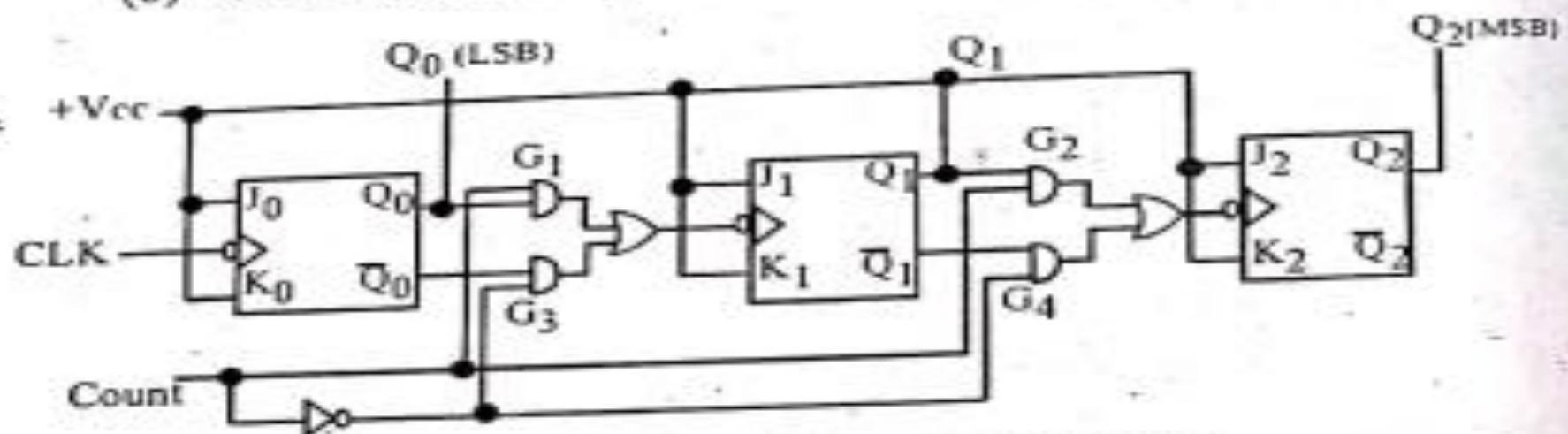


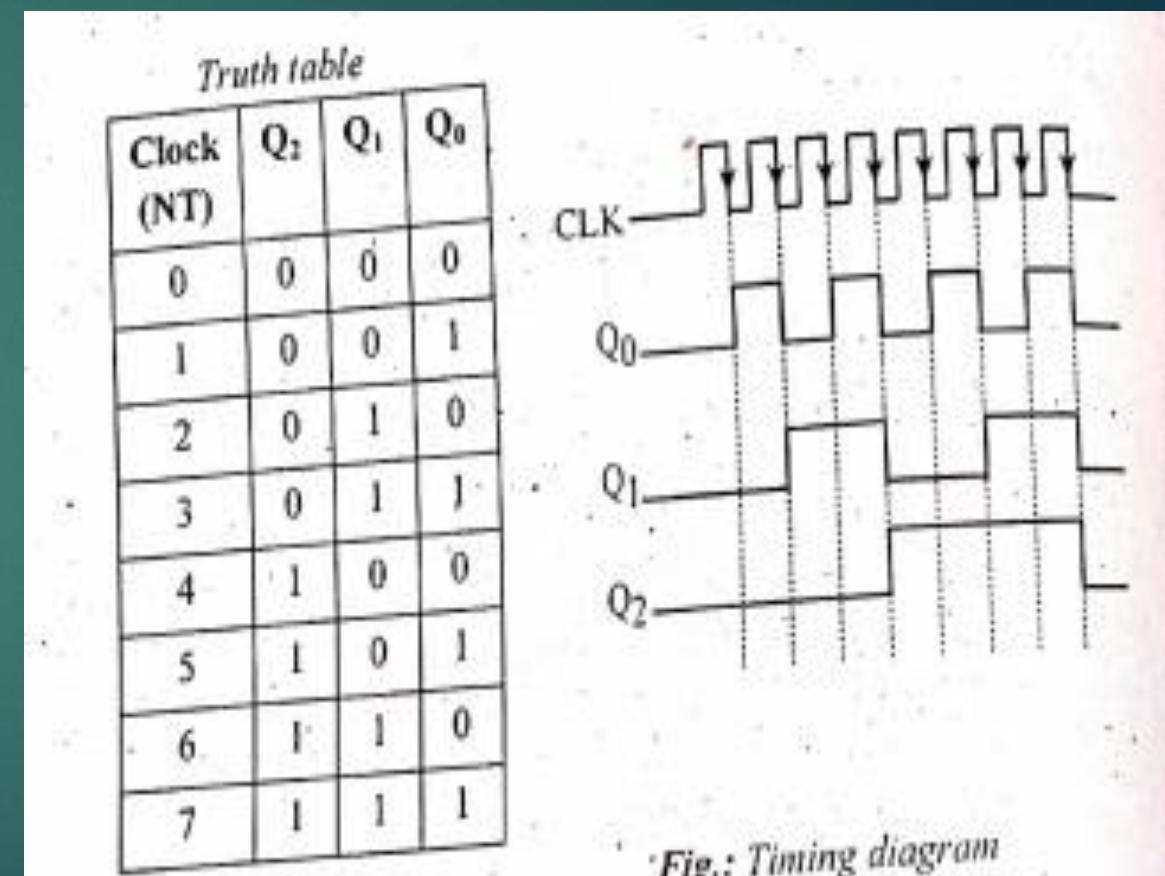
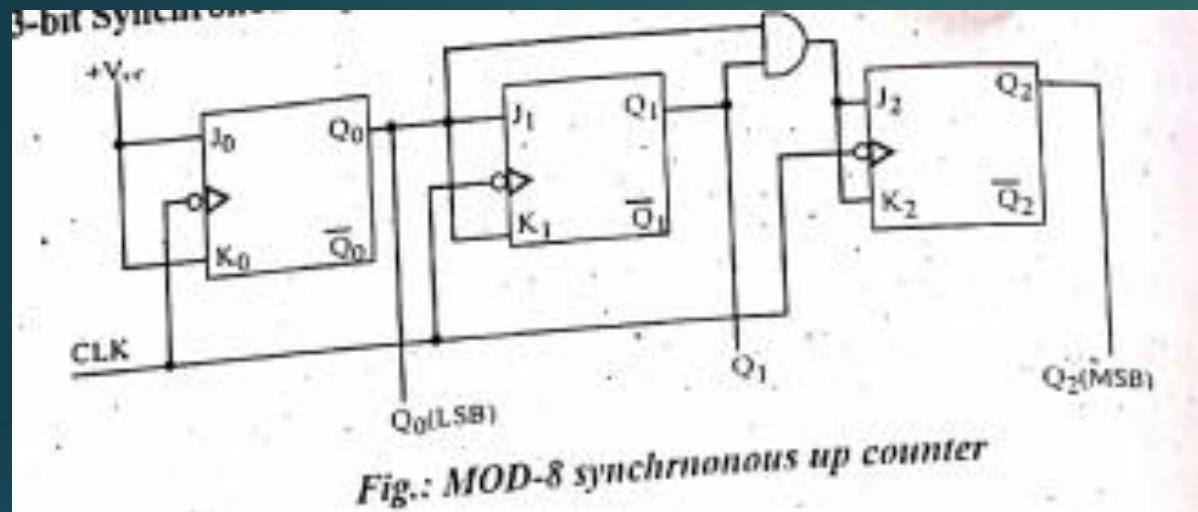
Fig.: 3-bit up/down ripple counter

Here 'Count' controls whether to count up or down.

When Count = 1, gates G<sub>1</sub> & G<sub>2</sub> are enabled, so it performs up count.

When Count = 0, gates G<sub>3</sub> & G<sub>4</sub> are enabled, so it performs down count.

# 3 bit synchronous up counter



## 3 bit synchronous down counter

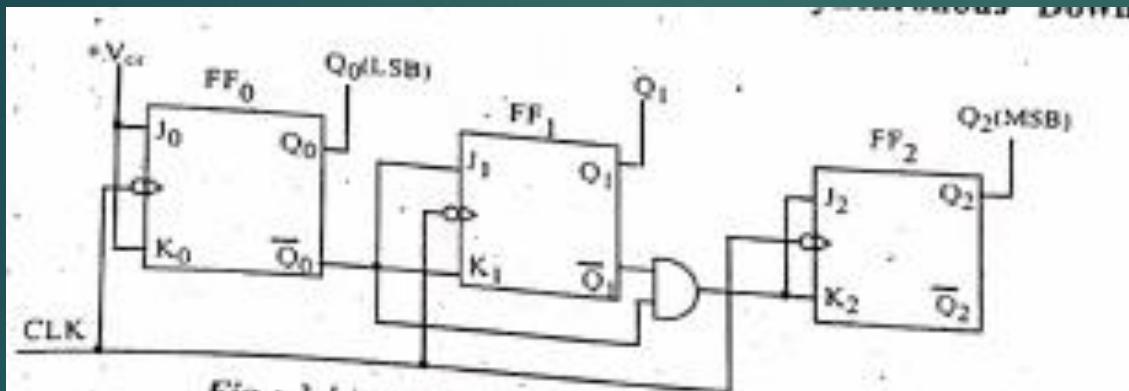


Fig.: 3-bit synchronous down counter

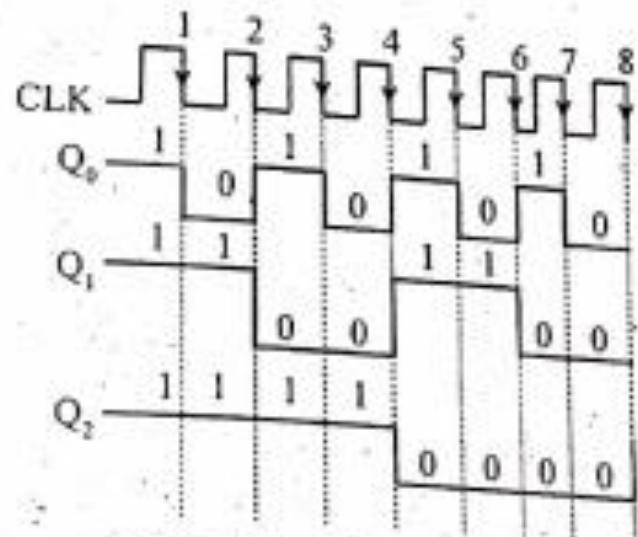


Fig.: Waveforms

# Do it yourself

- ▶ 4 bit synchronous up counter
- ▶ 4 bit synchronous down counter
- ▶ 4 bit synchronous up and down counter
- ▶ 4 bit Asynchronous up/down counter

# Changing the counter modulus

- ▶ Modulus is defined as no. of states that which a counter can progress. The counter which progress one count at a time in a strict binary progression and they all have a modulus given by  $2^n$  where n is equal to no. of flip-flops. Such counter has a (natural count) of  $2^n$  for e.g.
- ▶ Mod-2 counter consists of one flip-flop and it counts two discrete steps(0&1)
- ▶ Mod-4 counter consists of 2 flip-flop and counts 4 discreet steps (00 , 01, 10 & 11)
- ▶ Mod-8→3 flip-flop and counts 8 discreet steps  
(000, 001, 010, 011, 100, 101, 110,111)
- ▶ It is often desirable to construct counter having a modulus other than 2, 4, 8 & so on. E.g.: counter having modulus 3,5,7 would be useful
- ▶ A small modulus counter can also be construct from larger

# Counter designed as a synthesis problem

- ▶ Following steps are to be followed to make synchronous counter
- ▶ Draw the state diagram from given word problem
- ▶ Draw the next state table and find no. of flip-flops required.
- ▶ No of flip-flop= no of bits used in counter
- ▶ Decide the type of flip-flop to be used for the designed then determine the flip-flop extinction table based on transition of present state( $Q_n$ ) to next state( $Q_{n+1}$ )
- ▶ Prepare k-map for each flip-flop input and simplify
- ▶ Connect the circuit using flip-flop and other gate according to minimized expression

# Design a Mod 6 up counter

Step 1: Draw the state diagram

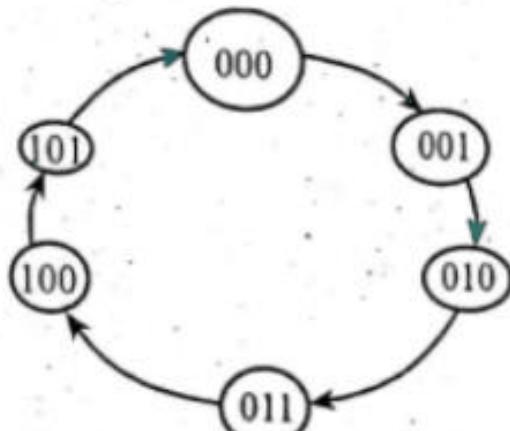


Fig.: State diagram of mod-6 up counter

Step 2 and step 3: Next state table and excitation table

Present State (Q <sub>n</sub> )			Next State (Q <sub>n+1</sub> )			FF Excitation					
Q <sub>2n</sub>	Q <sub>1n</sub>	Q <sub>0n</sub>	Q <sub>2n+1</sub>	Q <sub>1n+1</sub>	Q <sub>0n+1</sub>	J <sub>2</sub>	K <sub>2</sub>	J <sub>1</sub>	K <sub>1</sub>	J <sub>0</sub>	K <sub>0</sub>
0	0	0	0	0	1	0	x	0	x	1	x
0	0	1	0	1	0	0	x	1	x	x	1
0	1	0	0	1	1	0	x	x	0	1	x
0	1	1	1	0	0	1	x	x	1	x	1
1	0	0	1	0	1	0	0	0	x	1	x
1	0	1	0	0	0	x	1	0	x	x	1

Step 4:

Prepare the K-map for J and K input from transition table. 110 and 111 are unused condition, so are don't care conditions.

For J<sub>2</sub>,

		Q <sub>1n</sub> Q <sub>0n</sub>				
		Q <sub>2n</sub>	00	01	11	10
Q <sub>2n</sub>	Q <sub>1n</sub>	0	0	0	(1)	0
		1	x	x	x	x

$J_2 = Q_{1n}Q_{0n}$

For K<sub>2</sub>,

		Q <sub>1n</sub> Q <sub>0n</sub>				
		Q <sub>2n</sub>	00	01	11	10
Q <sub>2n</sub>	Q <sub>1n</sub>	0	x	x	x	x
		1	0	1	x	x

$K_2 = Q_{0n}$

For J<sub>1</sub>,

		Q <sub>1n</sub> Q <sub>0n</sub>				
		Q <sub>2n</sub>	00	01	11	10
Q <sub>2n</sub>	Q <sub>1n</sub>	0	0	(1)	x	x
		1	0	0	x	x

$J_1 = \overline{Q}_{2n}Q_{0n}$

For K<sub>1</sub>,

		Q <sub>1n</sub> Q <sub>0n</sub>				
		Q <sub>2n</sub>	00	01	11	10
Q <sub>2n</sub>	Q <sub>1n</sub>	0	x	x	1	0
		1	x	1	x	x

$K_1 = Q_{0n}$

For J<sub>0</sub>,

		Q <sub>1n</sub> Q <sub>0n</sub>				
		Q <sub>2n</sub>	00	01	11	10
Q <sub>2n</sub>	Q <sub>1n</sub>	0	1	x	x	1
		1	1	x	x	x

$J_0 = 1$

For K<sub>0</sub>,

		Q <sub>1n</sub> Q <sub>0n</sub>				
		Q <sub>2n</sub>	00	01	11	10
Q <sub>2n</sub>	Q <sub>1n</sub>	0	x	1	1	x
		1	x	1	x	x

$K_0 = 1$

Step 5:

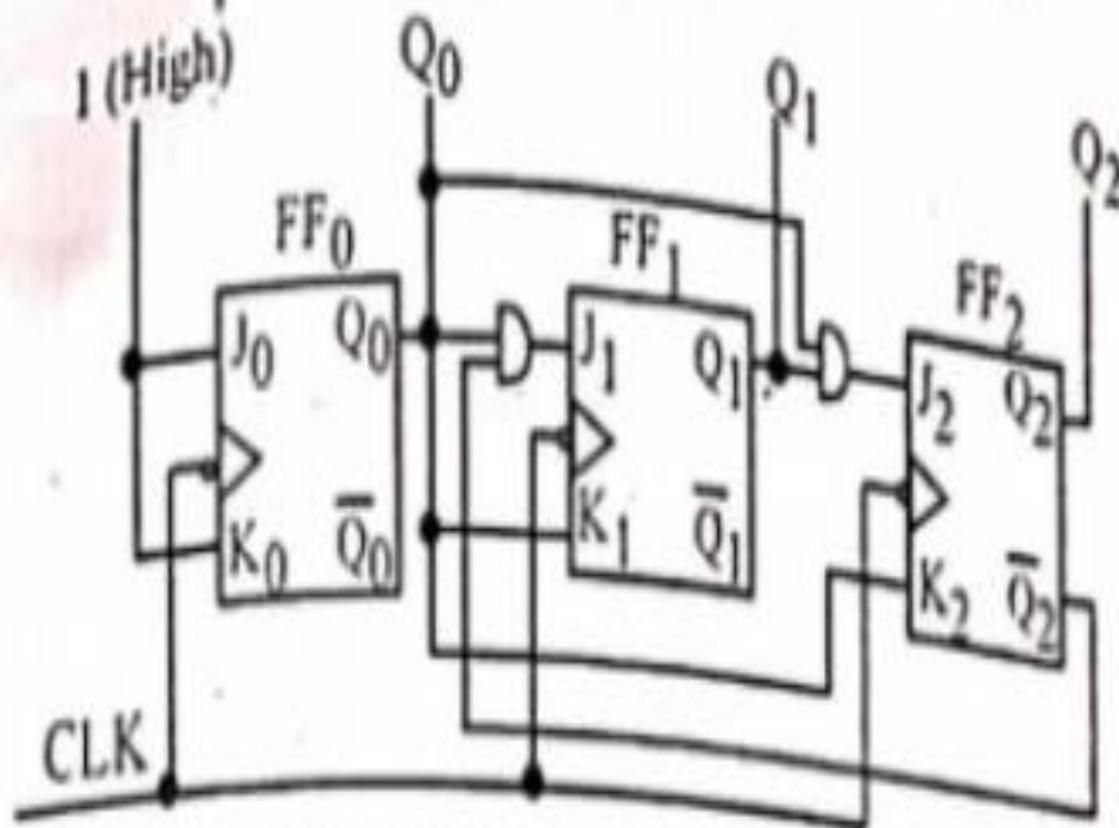


Fig.: Mod-6 synchronous up counter

We can use preset and clear here  
preset = 0 then  $Q=1$   
Clear= 0 then  $Q =0$

1. Design a 2-bit up/down counter using T flipflops.

⇒ Step 1: Let  $Y = 1$  (up counter),  $Y = 0$  (down counter)

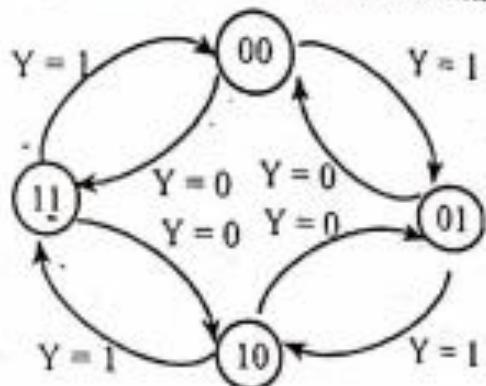


Fig.: State diagram of 2-bit up/down counter

Step 2 and Step 3:

The no. of bits are 2, so no. of flipflops = 2

Present State ( $Q_n$ )		Next State ( $Q_{n+1}$ )			
		$Y = 0$ (down)		$Y = 1$ (up)	
$Q_{1n}$	$Q_{0n}$	$Q_{1n+1}$	$Q_{0n+1}$	$Q_{1n+1}$	$Q_{0n+1}$
0	0	1	1	0	1
0	1	0	0	1	0
1	0	0	1	1	1
1	1	1	0	0	0

Present State $Q_n$		Control Input	Next State ( $Q_{n+1}$ )		Flipflop $T_1$ and $T_0$	
$Q_{1n}$	$Q_{0n}$	$Y$	$Q_{1n+1}$	$Q_{0n+1}$	$T_1$	$T_0$
0	0	0	1	1	1	1
0	0	1	0	1	0	1
0	1	0	0	0	0	1
0	1	1	1	0	1	1
1	0	0	0	1	1	1
1	0	1	1	1	0	1
1	1	0	1	0	0	1
1	1	1	0	0	1	1

Step 4: K-mapping for  $T_1$  and  $T_0$  (inputs are  $Q_{1n}$ ,  $Q_{0n}$ ,  $Y$ )

For  $T_1$ ,

$Q_{0n}Y$	00	01	11	10
$Q_{1n}$	0	1	0	1
0	1	0	1	0
1	1	0	1	0

$$T_1 = \overline{Q_{01}}\bar{Y} + Q_{01}Y \\ = (Q_{01} \oplus Y)$$

For  $T_0$ ,

$Q_{0n}Y$	00	01	11	10
$Q_{1n}$	0	1	1	1
0	1	1	1	1
1	1	1	1	1

$$T_0 = 1$$

Step 5:

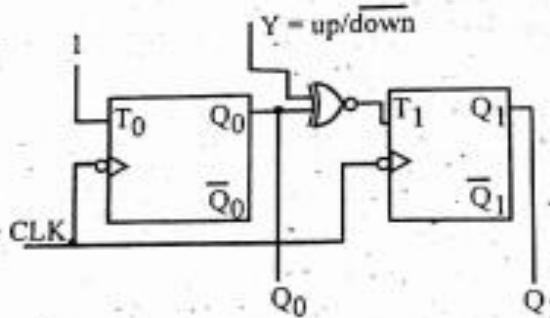
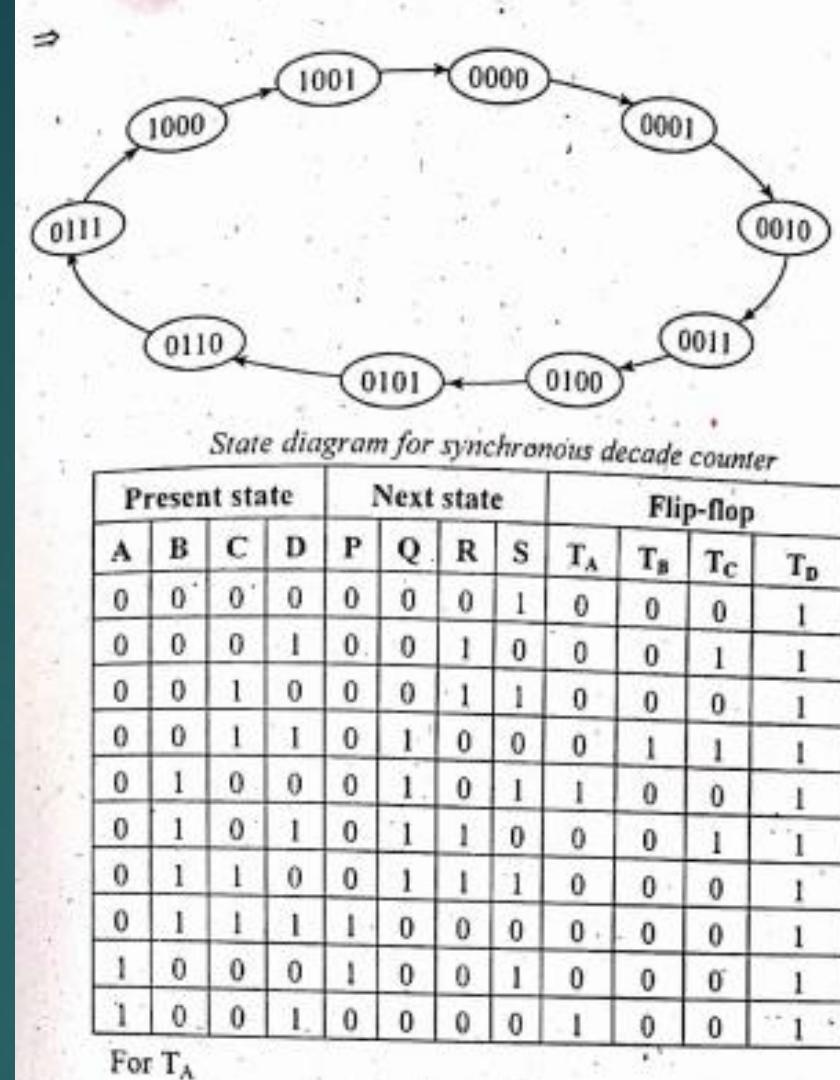


Fig.: 2-bit synchronous up/down counter.

## Design a synchronous Decade counter



		CD	00	01	11	10
		AB	0	0	0	0
		00	0	0	0	0
		01	1	0	1	0
		11	x	x	x	x
		10	0	1	x	x

$$T_A = AD + BC'D' + BCD$$

For T<sub>B</sub>

		CD	00	01	11	10
		AB	0	0	1	0
		00	0	0	1	0
		01	0	0	1	0
		11	x	x	x	x
		10	0	0	x	x

T<sub>B</sub> = RD

For T<sub>C</sub>

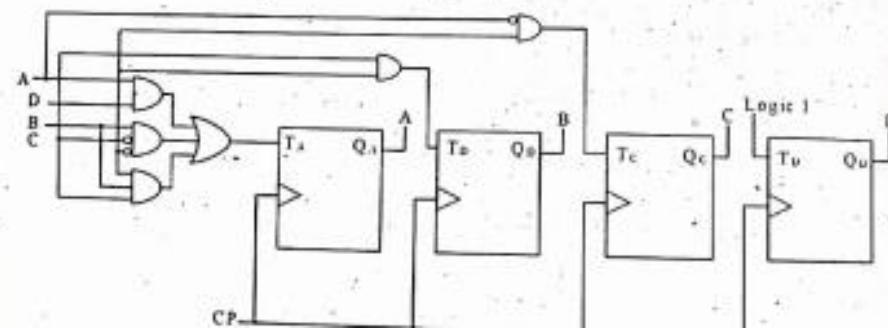
		CD	00	01	11	10
		AB	0	1	1	0
		00	0	1	1	0
		01	0	1	1	0
		11	x	x	x	x
		10	0	0	x	x

T<sub>C</sub> = A'D

For T<sub>D</sub>

		CD	00	01	11	10
		AB	1	1	1	0
		00	1	1	1	0
		01	1	1	1	0
		11	x	x	x	x
		10	1	1	x	x

$$T_D = 1$$



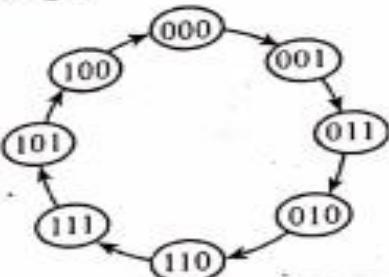
7. List the advantages and disadvantages of a synchronous counter over asynchronous counter. Design a 3-bit synchronous counter which follow gray code sequence. [2074 Ashwin]

⇒ Advantage of synchronous counter over asynchronous counter:

Fast operation

Disadvantages:

Complex circuit design



Present State			Next State			Flip flop		
$Q_{A_n}$	$Q_{B_n}$	$Q_{C_n}$	$Q_{A_{n+1}}$	$Q_{B_{n+1}}$	$Q_{C_{n+1}}$	$D_A$	$D_B$	$D_C$
0	0	0	0	0	1	0	0	1
0	0	1	0	1	1	0	1	1
0	1	1	0	1	0	0	1	0
0	1	1	1	0	1	1	0	1
1	1	0	1	1	1	1	1	1
1	1	1	1	0	1	1	0	1
1	0	1	1	0	0	1	0	0
1	0	0	0	0	0	0	0	0

$Q_{A_n} \setminus Q_{B_n} Q_{C_n}$	00	01	11	10
0	0	0	0	1
1	0	1	1	1

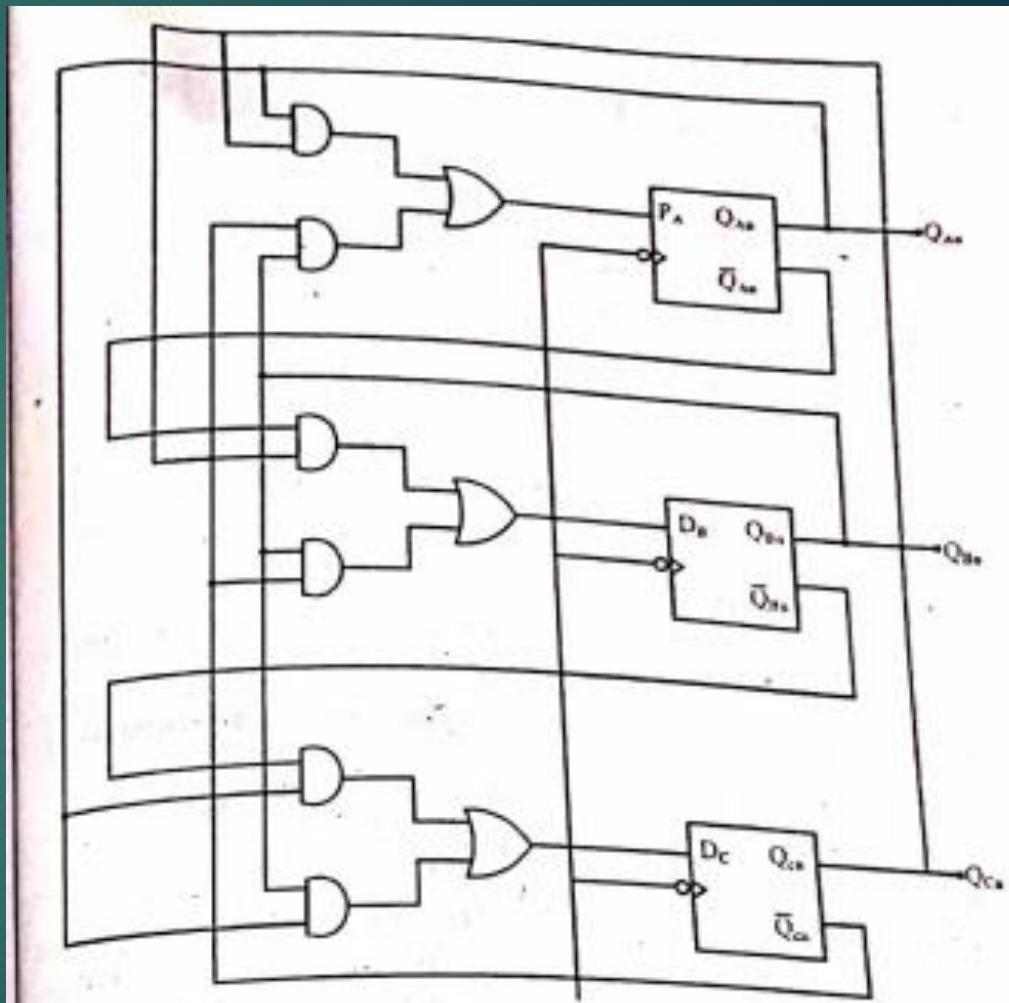
$$D_A = Q_{A_n} Q_{C_n} + Q_{B_n} \bar{Q}_{C_n}$$

$Q_{A_n} \setminus Q_{B_n} Q_{C_n}$	00	01	11	10
0	0	1	1	0
1	0	0	0	1

$$D_B = \bar{Q}_{A_n} Q_{C_n} + Q_{B_n} \bar{Q}_{C_n}$$

$Q_{A_n} \setminus Q_{B_n} Q_{C_n}$	00	01	11	10
0	1	1	0	0
1	0	0	1	1

$$D_C = Q_{A_n} \bar{Q}_{B_n} + Q_{A_n} Q_{B_n}$$



## Memory Unit

- A memory unit is a device to which binary information is transferred for storage and from which information is retrieved when needed for processing.
- A memory unit stores binary information in groups of bits called words. The internal structure of memory unit is specified by the number of words it contains and the number of bits in each word.
- The memory unit is an essential component in any digital computer since it is needed for storing programs and data. Not all accumulated information is needed by the CPU at the same time.
- Therefore, it is more economical to use low-cost storage devices to serve as a backup for storing the information that is not currently used by CPU

S.No.	Unit & Description
1	<b>Bit (Binary Digit)</b> A binary digit is logical 0 and 1 representing a passive or an active state of a component in an electric circuit.
2	<b>Nibble</b> A group of 4 bits is called nibble.
3	<b>Byte</b> A group of 8 bits is called byte. A byte is the smallest unit, which can represent a data item or a character.
4	<b>Word</b> A computer word, like a byte, is a group of fixed number of bits processed as a unit, which varies from computer to computer but is fixed for each computer.  The length of a computer word is called word-size or word length. It may be as small as 8 bits or may be as long as 96 bits. A computer stores the information in the form of computer words.

S.No.	Unit & Description
1	<b>Kilobyte (KB)</b> 1 KB = 1024 Bytes
2	<b>Megabyte (MB)</b> 1 MB = 1024 KB
3	<b>GigaByte (GB)</b> 1 GB = 1024 MB
4	<b>TeraByte (TB)</b> 1 TB = 1024 GB
5	<b>PetaByte (PB)</b> 1 PB = 1024 TB

# Internal Structure of a Memory Unit

The internal structure of a memory unit is specified by the **number of words** it contains and the **number of bits** in each word. Special input lines called **address lines** select one particular word. Each word in memory is assigned an identification number, called an **address**, starting from 0 and continuing with 1, 2, 3, up to  $2^k - 1$  where **k** is the number of address lines. The selection of a specific word inside the memory is done by applying the k-bit binary address to the address lines.

A **decoder** inside the memory accepts this address and opens the paths needed to select the bits of the specified word.

Computer memories may range from 1024 words, requiring an address of 10 bits, to  $2^{32}$  words, requiring 32 address bits. It is customary to refer to the number of words(or bytes) in a memory with one of the letters:

- **K(Kilo)** is equal to  $2^{10}$
- **M(Mega)** is equal to  $2^{20}$
- **G(Giga)** is equal to  $2^{30}$

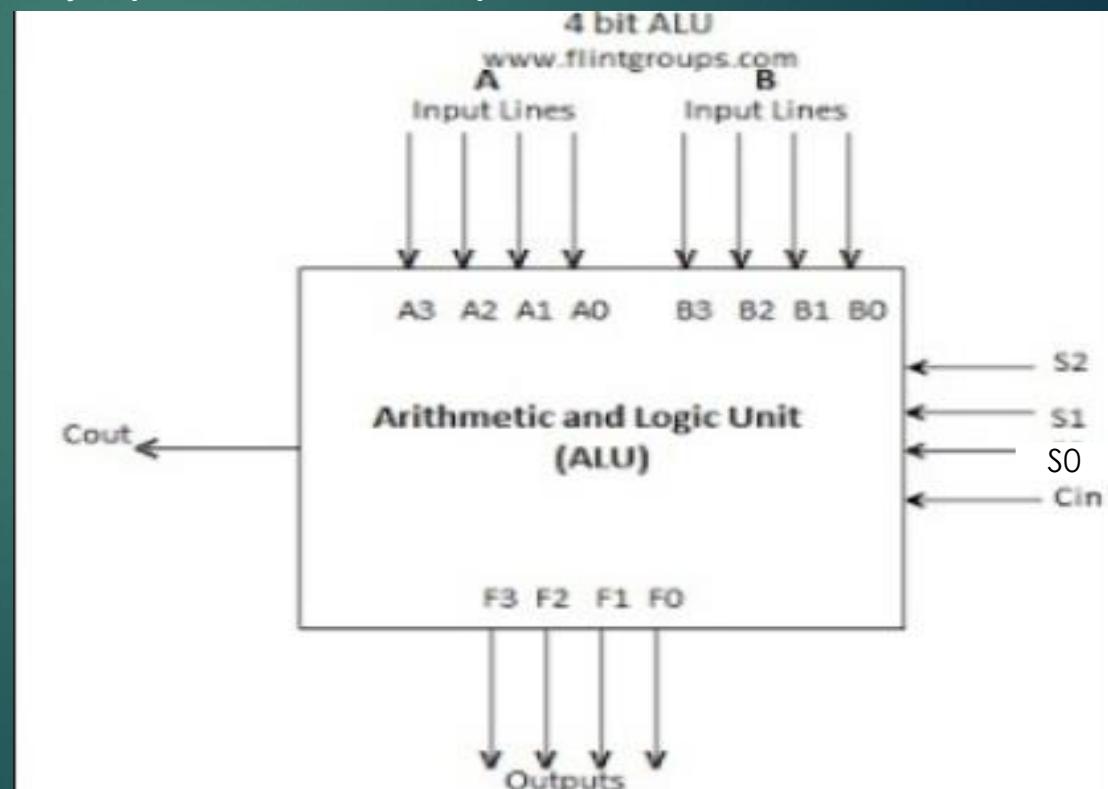
# Arithmetic Logic Unit (ALU)

Prepared by: ER. Simanta Kasaju

# ALU

It is a multi-operation combinational logic digital function it can perform a set of basic arithmetic operation and a set of logic operation

ALU has a number of selection lines to select a particular cooperation in the unit. The selection lines are decoded within the ALU so that K-selection variables can specify up to  $2^k$  distinct operation

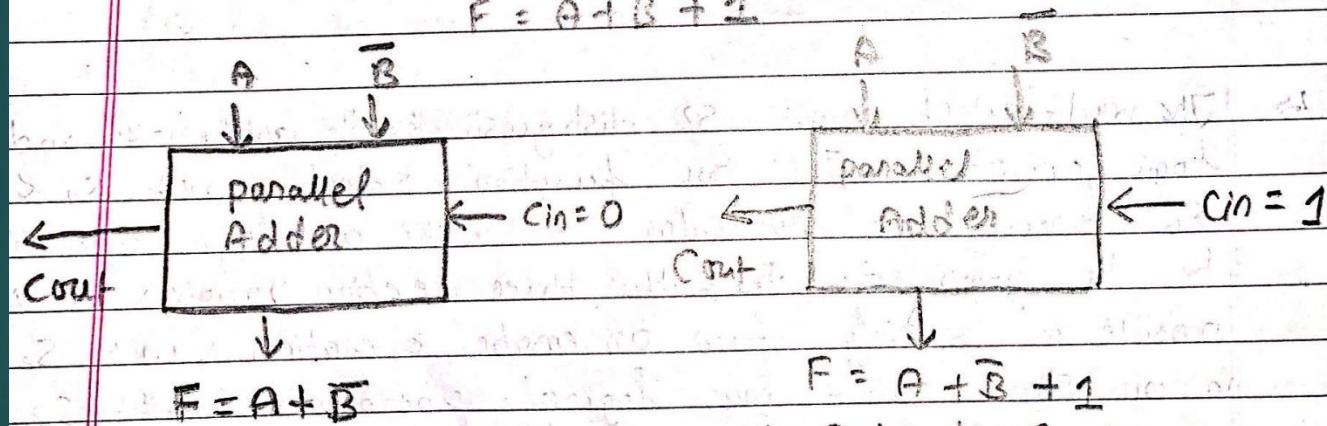
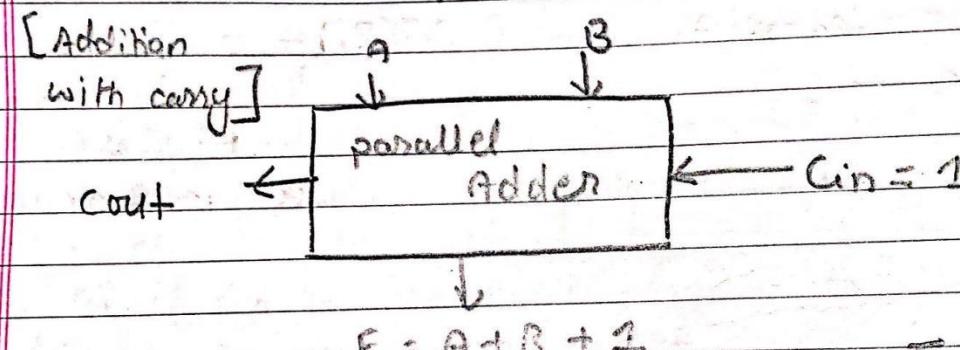
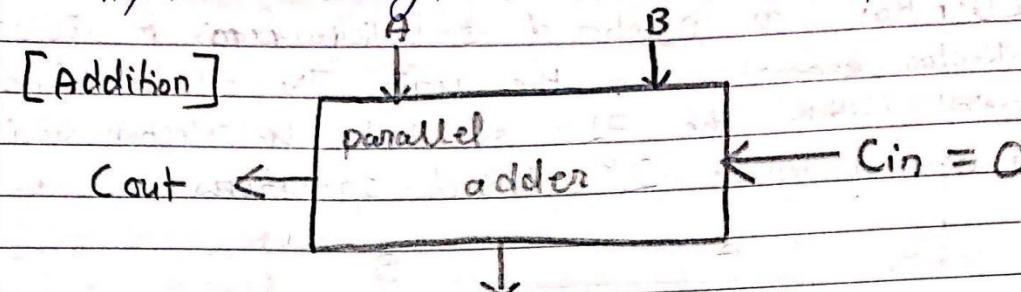


# Contd.

- ▶ The mod-select input S2 distinguish between arithmetic and logic operation. The two function select input S1 and S0 specify the particular arithmetic or logical operation to be generated with three selection variables it is possible to specify four arithmetic operation (with S2 in one step) and four logical operation (with S2 in other step); the input and output carries have meaning only during an arithmetic operation.

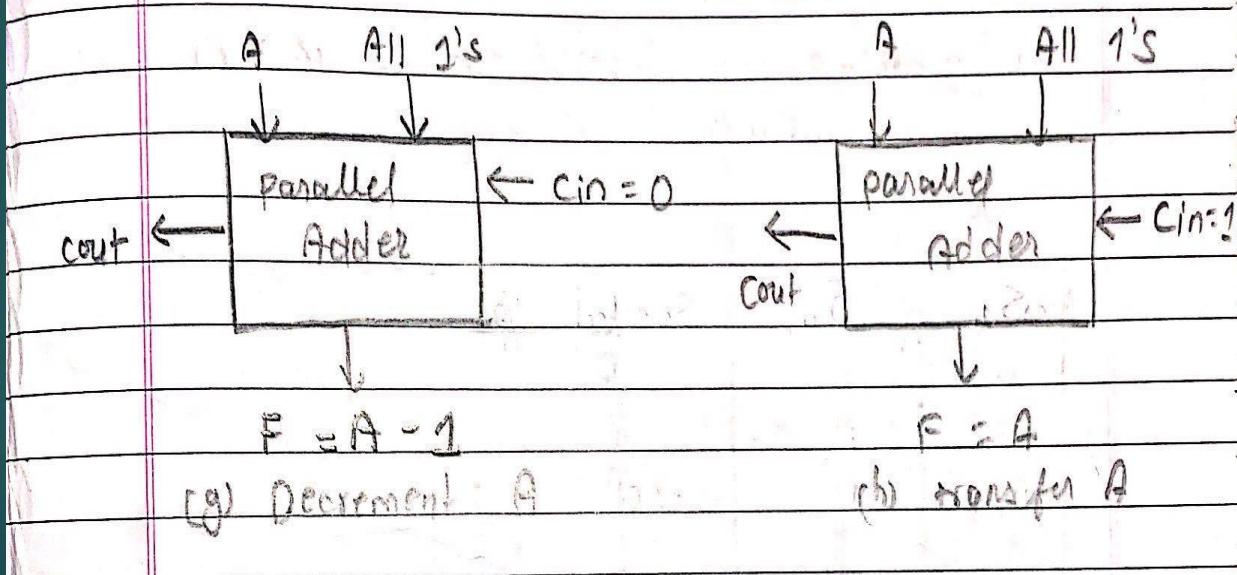
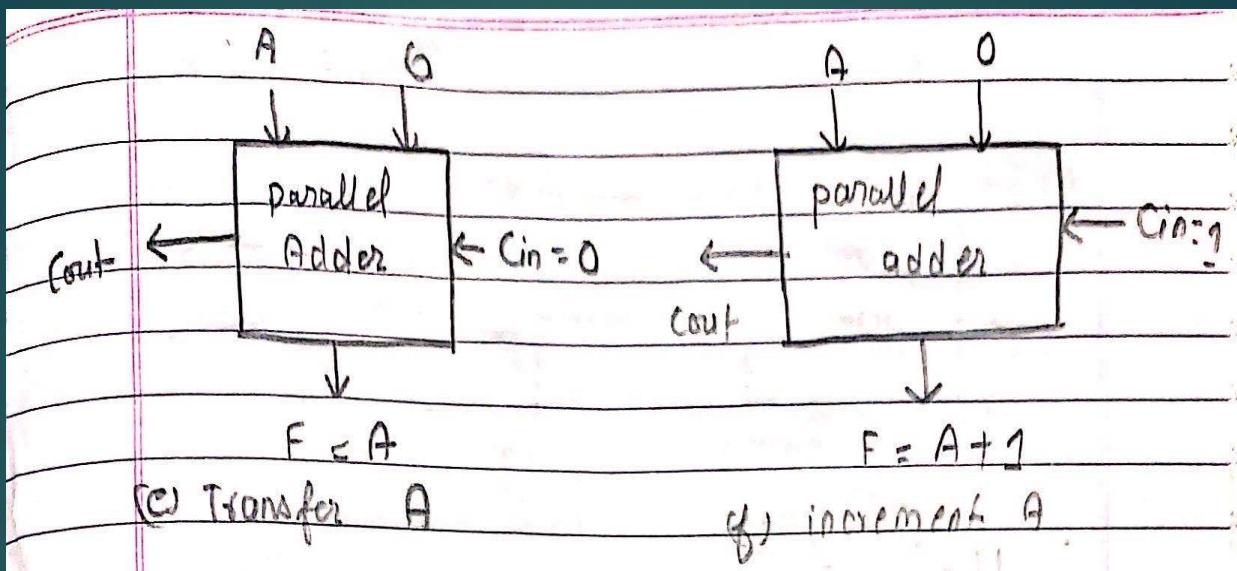
## #Design of arithmetic circuit

The design of arithmetic circuit deals with the design of combination circuit that can be used to do arithmetic operations of the inputs. Usually nibble adder are used in the arithmetic circuit. In the nibble adder by changing the input we can get various arithmetic operation as



(c)  $\Rightarrow$  plus is complement of  $B$

Subtraction

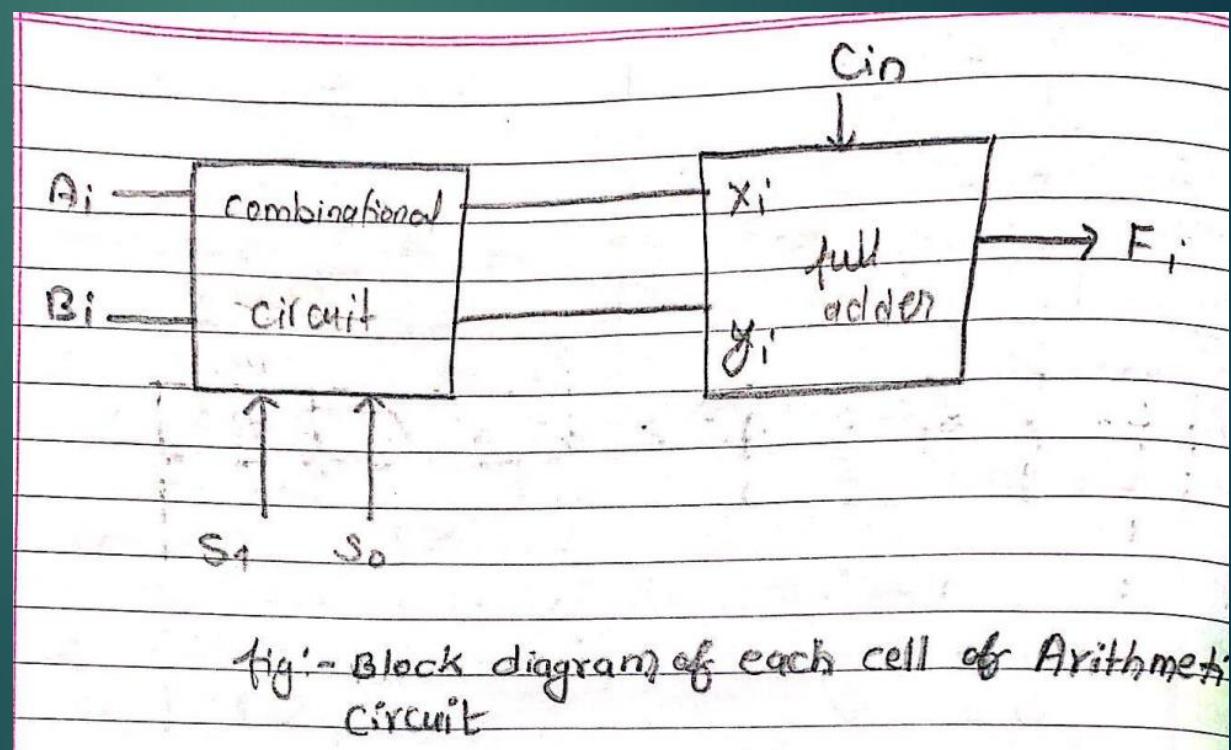


# Contd.

The objective of arithmetic circuit is to design a combination circuit before the parallel adder so that it can produce a required input according to the selection line ( $S_1$  and  $S_0$ )

We know that design of any combinational starts with the specification of the circuit so consider the following specification for the arithmetic circuit where these are two control line. The function is

output



<b>S1</b>	<b>S0</b>	<b>Selected i/p</b>
0	0	0
0	1	B
1	0	$\bar{B}$
1	1	1

→ NOTE

## Function of block diagram

<b>function</b>	<b>select</b>	<b>Y(i/p)</b>			<b>F(o/p)</b>	<b>function</b>
		S1	S0	Cin		
0	0	0	0	0	$F=A$	Transfer of A
0	0	1	0	0	$F=A+1$	Increment of A
0	1	0	B		$F=A+B$	Addition of A and B
0	1	1	B		$F=A+B+1$	Addition with carry
1	0	0	$\bar{B}$		$F=A+ \bar{B}$	Subtraction with borrow
1	0	1	$\bar{B}$		$F=A+ \bar{B}+1$	Subtraction
1	1	0	1		$F=A-1$	Decrement
1	1	1	1		$F=A$	Transfer of A

Lets derive truth table for  $X_i$  and  $Y_i$  i.e. input of full adder  
 note  $\rightarrow X_i = a_i$  and  $y_i$  depends upon  $S_0$  and  $S_1$

$S_1$	$S_0$	$A_i$	$B_i$	$X_i$	$Y_i$
0	0	0	0	0	0
0	0	0	1	0	0
0	0	1	0	1	0
0	0	1	1	1	0
0	1	0	0	0	(B) 0
0	1	0	1	0	(B) 1
0	1	1	0	1	(B) 0
0	1	1	1	1	(B) 1
1	0	0	0	0	(B) 1
1	0	0	1	0	(B) 0
1	0	1	0	1	(B) 1
1	0	1	1	1	(B) 0
1	1	0	0	0	1
1	1	0	1	0	1
1	1	1	0	1	1
1	1	1	1	1	1

### o/p full adder

$A_i + 0$

"

"

"

$A_i + B_i$

"

"

"

"

$A_i + B_i'$

"

"

"

$A_i + 1$

"

"

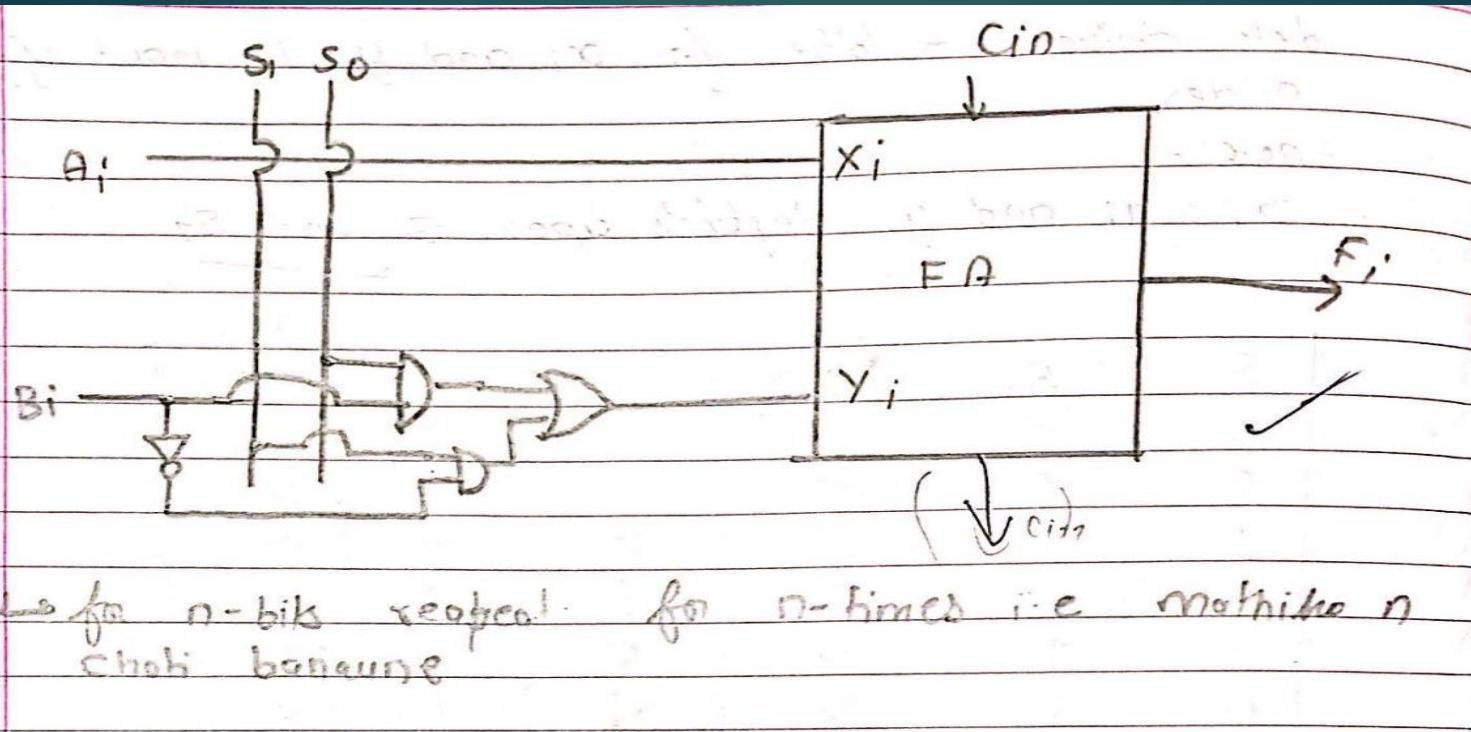
K-map for  $Y_i$ :

$S_1 S_0$	$A_i B_i$	00	01	11	10
00	00	0	0	0	0
01	01	1	1	0	
11	11	1	1	1	1
10	10	1	0	0	1

$$Y_i = S_0 B_i + S_1 \bar{B}_i$$

$$X_i = A_i \quad \because \text{same}$$

For  $y$  look for the  $S_0$  and  $S_1$  of previous table



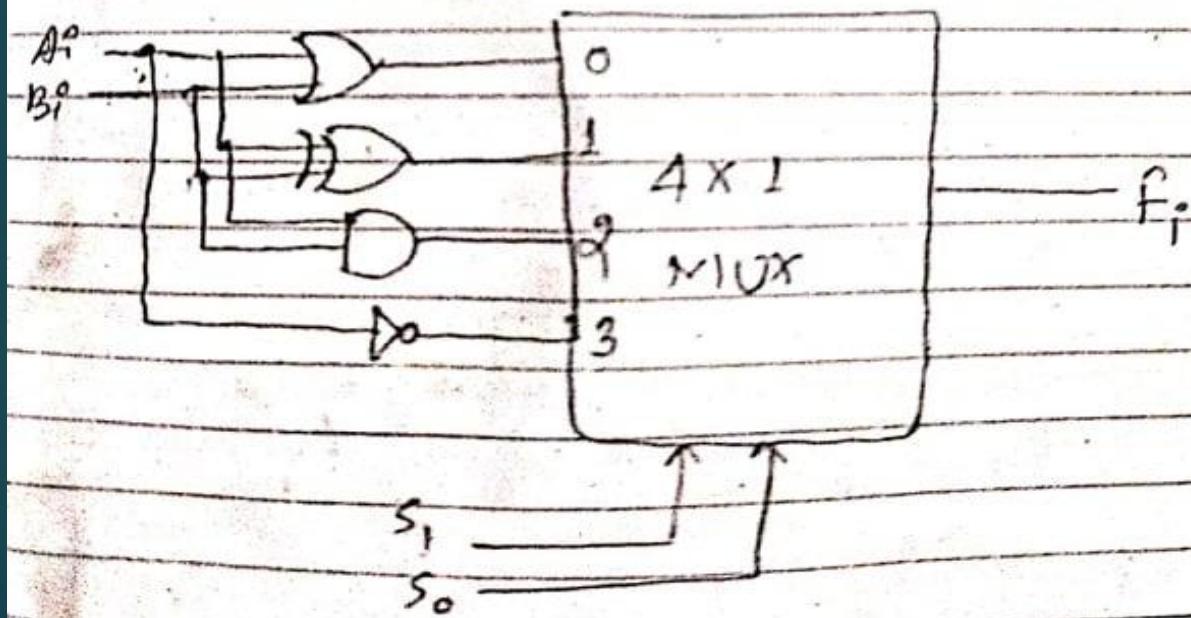
Design of logic circuit

Selection mode  $S_2, S_1, S_0$

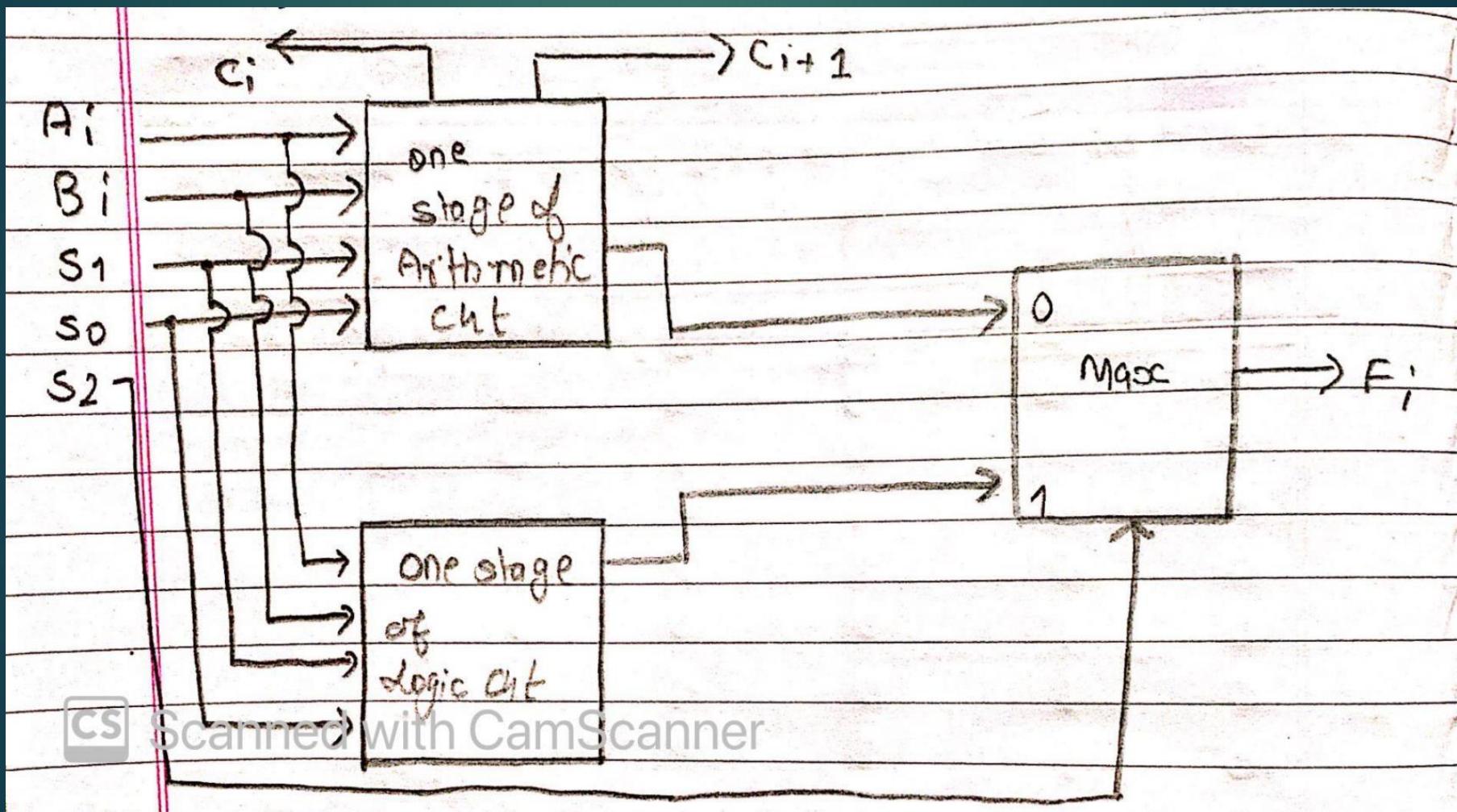
AND, OR, NOT and X-OR

Let's take  $4 \times 1$  MUX. Take  $S_0$  and  $S_1$  as  
 ~~$S$~~  as i/p of  $MUX$

$S_1$ , $S_0$	G/I/P ( $F$ )	Operation
0 0	$A_i P + B_i$	OR
0 1	$A_i P \oplus B_i$	X-OR
1 0	$A_i P \cdot B_i$	AND
1 1	$A_i'$	NOT



## Combining logic and arithmetic circuit



# Design of ALU

We design a ALU with 8 arithmetic operation and four logic operations. Three selection variables S2, S1 and S0. select 8 different operations and the input carry Cin used to select force additional arithmetic operation. With S2=0 selection variable S1 and S0 together with Cin select the 8 arithmetic operation listed in table below. With A S2=1 variable S1 & S0 will select the four logic operations i.e.(OR, XOR, AND, NOT)

The design of an ALU is a combination logic problem because the unit has a regular pattern. It can be broken into identical stages connected in cascade through the carry. We can design one stage of the ALU and duplicate it for number of stages required.

There are 6 inputs to each stage (Ai, Bi, Cin, S2, S1 & S0)

## Function table

S2	S1	S0	Cin	o/p	remark
0	0	0	0	F=A	Transfer
0	0	0	1	F=A+1	Increment of A
0	0	1	0	F=A+B	Addition of A&B
0	0	1	1	F=A+B+1	Addition with carry
0	1	0	0	F=A+ $\bar{B}$	Addition with 1's complement
0	1	0	1	F=A+ $\bar{B}$ +1	Subtraction
0	1	1	0	F=A-1	Decrement
0	1	1	1	F=A	Transfer of A
1	0	0	X	A+B	OR
1	0	1	X	A $\oplus$ B	X-OR
1	1	0	X	A.B	AND
1	1	1	X	$\bar{A}$	NOT

Note : if  $S_2=0 \rightarrow$  Arithmetic  
 $S_2=1 \rightarrow$  Logic

The final ALU is shown below. Only the 1<sup>st</sup> two stages are drawn, but the diagram can be extended to more stages. The i/p to each full adder circuit are specified by Boolean function.

$$X_i = A_i + S_2 S_1' S_0' B_i + S_2 S_1 S_0' B_i'$$

$$Y_i = S_0 B_i + S_1 B_i'$$

$$Z_i = S_2' C_i$$

When  $S_2=0$  the 3 function reduce to

$$X_i = A_i$$

$$Y_i = S_0 B_i + S_1 B_i'$$

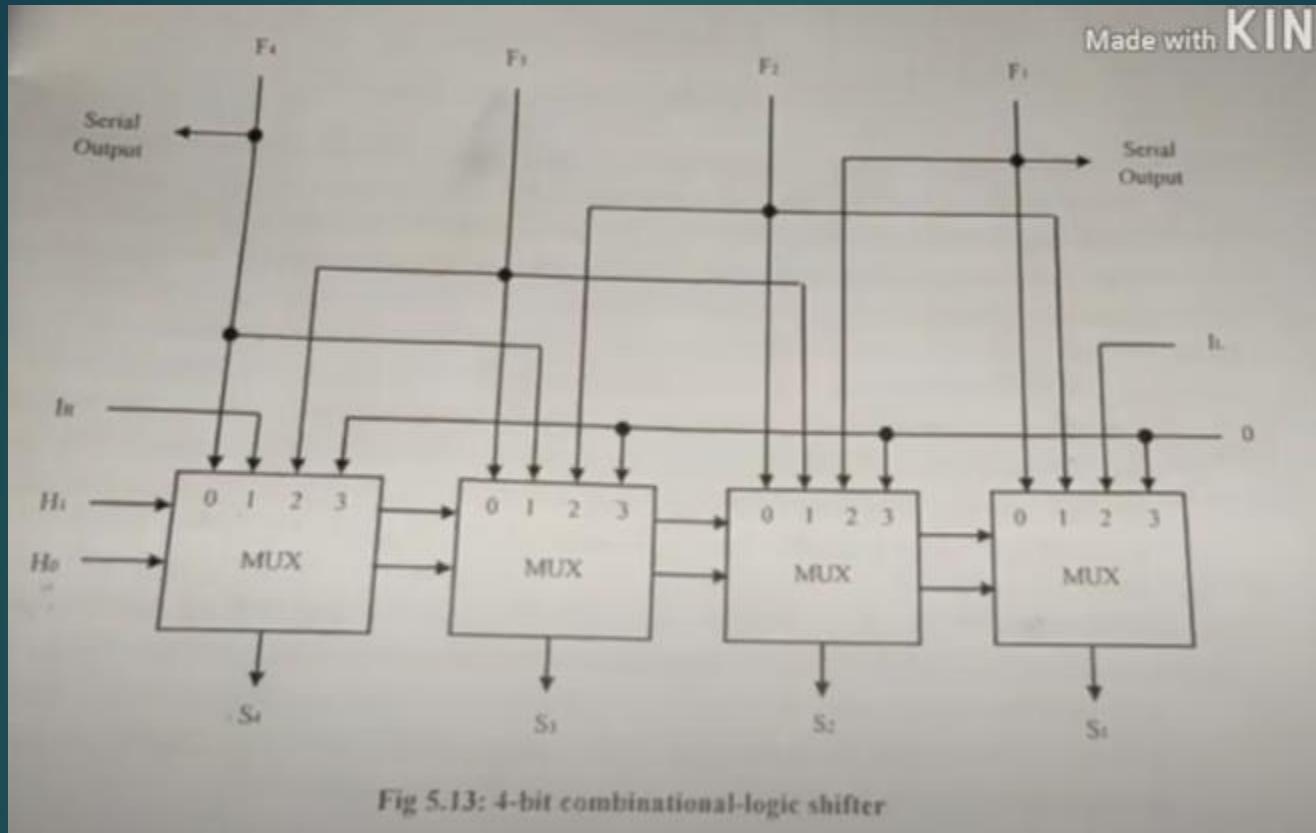
$$Z_i = C_i$$

# Shifter(Design of shifter)

The shifter unit attached to a processor transfer the output of the ALU onto the o/p bus. The shifter may transfer the information directly without a shift or it may shift the information to the right or left. It is bi-directional with parallel load. The information of the ALU can be transferred to the register in parallel than shifted to right or left, for this a clock pulse is needed for the transfer to the shift register and another is needed for the shift.

These two pulses are in addition to the pulse required to transfer the information from the shift register to the destination register. The diagram below shows the four stages of the shifter. The shifter also consist of 'n' stages in a system with 'n' parallel lines. Input IP and IL serves as serial input for the last and first stages during the shift right and shift left respectively. Another selection variable may be specified in what does into IR and IL during the shift

IR

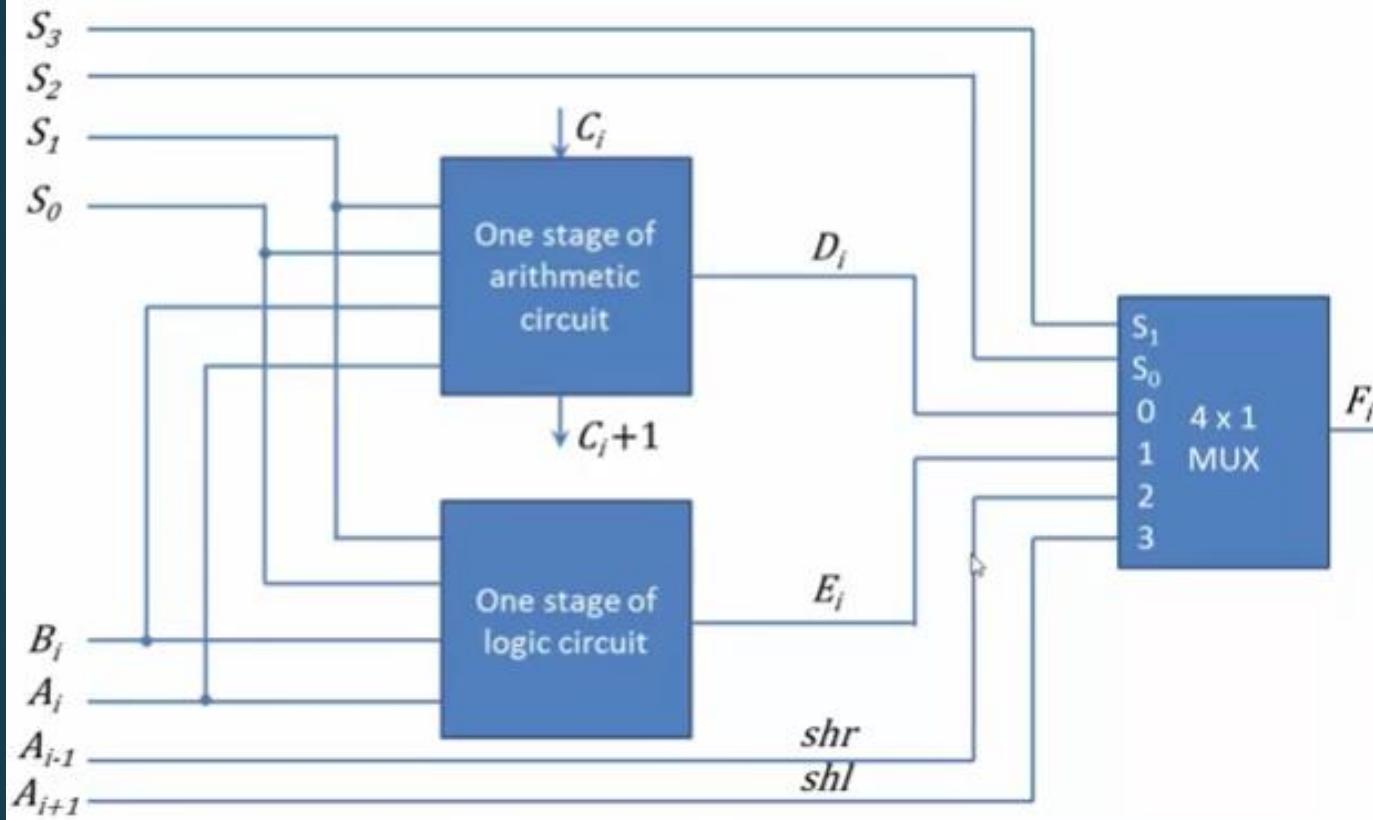


Function table for the shifter

H1	H0	Operation	Function
0	0	S ← F	Transfer F to S (no shift)
0	1	S ← shr F	Shift right F into S
1	0	S ← shl F	Shift left F into S
1	1	S ← 0	Transfer 0's into S

Only single stage is shown if 4 bit then repeat the same circuit 4 times.

## 4 – bit Arithmetic Logic Shift Unit



S3	S2	Operation
0	0	Arithmetic
0	1	Logical
1	0	Shr
1	1	Shl

Point to be remember here 1 extra selection line is used i.e  $S_3$ . So we have to look at the  $S_3$  and  $S_2$  bit for the selection of Arithmetic operation, logical Operation and shift operation.

■ ALU Function

$S_3$	$S_2$	$S_1$	$S_0$	$C_{in}$	Operation	Function
0	0	0	0	0	$F = A$	Transfer A
0	0	0	0	1	$F = A + 1$	Increment A
0	0	0	1	0	$F = A + B$	Addition
0	0	0	1	1	$F = A + B + 1$	Add with carry
0	0	1	0	0	$F = A + B'$	Subtract with borrow
0	0	1	0	1	$F = A + B' + 1$	Subtraction
0	0	1	1	0	$F = A - 1$	Decrement
0	0	1	1	1	$F = A$	Transfer A
0	1	0	0	x	$F = A \vee B$	OR
0	1	0	1	x	$F = A \oplus B$	XOR
0	1	1	0	x	$F = A \wedge B$	AND
0	1	1	1	x	$F = A'$	Complement A
1	0	x	x	x	$F = \text{shr } A$	Shift right A into F
1	1	x	x	x	$F = \text{shl } A$	Shift left A into F

# Contd.

## Status register

The relative magnitudes to two numbers may be determined by subtracting one from another and then checking certain bit conditions in the resultant difference. If the two numbers are unsigned , the bit conditions of interest are the output carry and a possible zero result. If the two number include a sign bit in the highest order position the bit conditions of interest are the sign of the result, a zero indication and a overflow condition.

Figure below shows the block diagram of an 8-bit ALU with four bit status register. The four status bits are symbolized by ‘c’, ‘s’, ‘z’ & ‘v’. The bits are set a clear as a result of a operation performed in the ALU.

1. Bit C is set ‘1’ if the output carry of the ALU is one.

It is cleared if the output carry is zero.

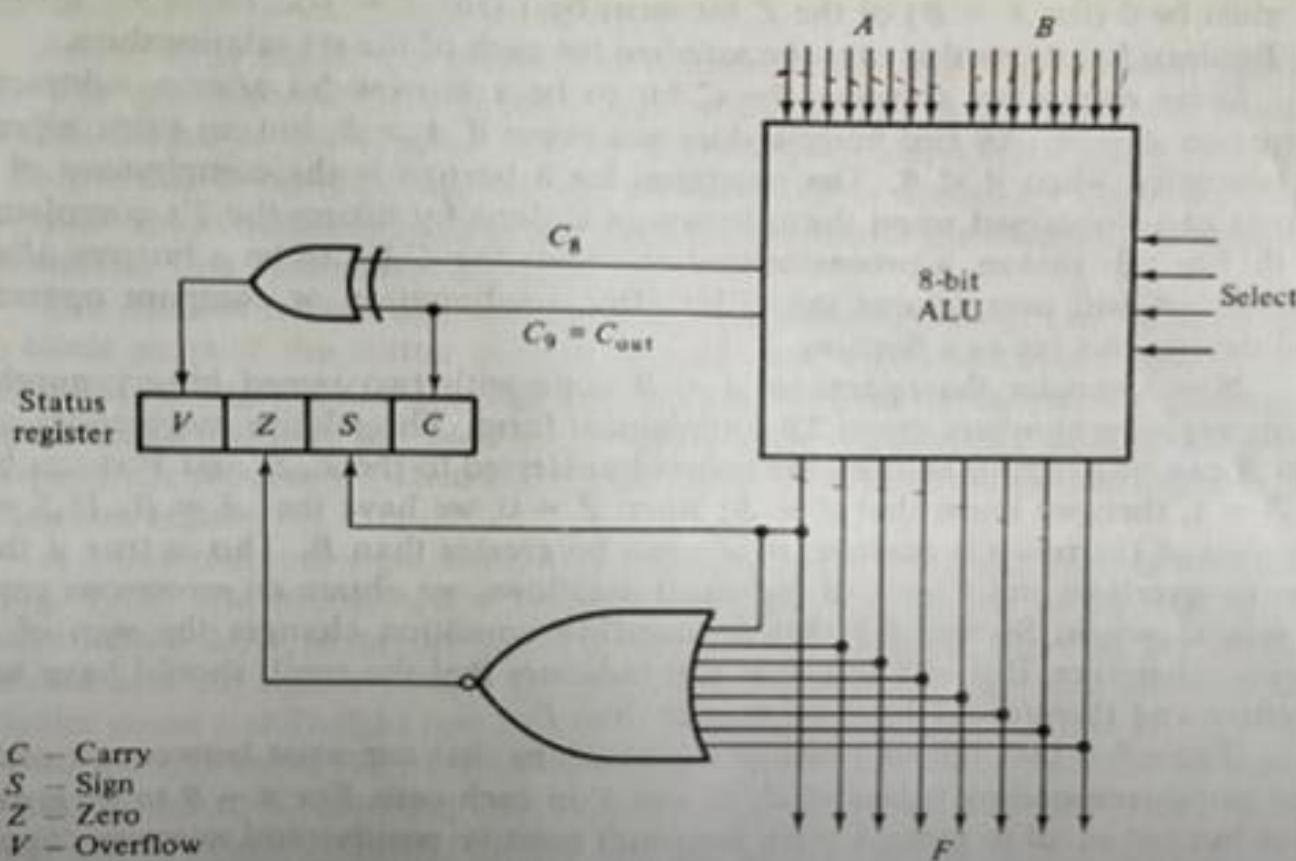
2. Bit S is set ‘1’ if the highest order bit of the result in the output of the ALU (the signed bit) is 1

It is cleared if the highest bit is zero

3. The bit ‘Z’ set if the output of the ALU contains all zeros and cleared otherwise. Z=1 if the result is zero and Z=0 if the result is not zero.

# Contd.

4. Bit 'V' is set if the exclusive-OR of carries C6 and C7 is one and cleared otherwise. This is the condition for overflow when the numbers are in since 2 complement of representation.



relationship	Condition of status bits	Boolean function	
$A > B$	$C=1 \text{ & } Z=0$	$CZ'$	
$A \geq B$	$C=1$	$C$	
$A < B$	$C=0$	$C'$	
$A \leq B$	$C=0 \text{ or } Z=1$	$C'+Z$	Status bits after the subtraction
$A=B$	$Z=1$	$Z$	Of unsigned number (A-B)
$A \neq B$	$Z=0$	$Z'$	

TABLE 9-6 Status bits after the subtraction of sign-2's complement numbers ( $A - B$ )

Relation	Condition of status bits	Boolean function
$A > B$	$Z = 0 \text{ and } (S = 0, V = 0 \text{ or } S = 1, V = 1)$	$Z'(S \odot V)$
$A > B$	$S = 0, V = 0 \text{ or } S = 1, V = 1$	$S \odot V$
$A < B$	$S = 1, V = 0 \text{ or } S = 0, V = 1$	$S \oplus V$
$A < B$	$S = 1, V = 0 \text{ or } S = 0, V = 1 \text{ or } Z = 1$	$(S \oplus V) + Z$
$A = B$	$Z = 1$	$Z$
$A \neq B$	$Z = 0$	$Z'$

Describe an arithmetic circuit with one selection variable 'S' and two n-bit data inputs a & b. the circuit generates the following four arithmetic operation in construction with input carry C-in draw the logic diagram for first two stages

S	C in=0	C in=1
0	$D = A + B$	$D = A + 1$
1	$D = A - 1$	$D = A + \bar{B} + 1$

S	C in	D	Remarks
0	0	$D = A + B$	Addition of A & B
0	1	$D = A + 1$	Increment of A
1	0	$D = A - 1$	Decrement of A
1	1	$D = A + \bar{B} + 1$	subtraction

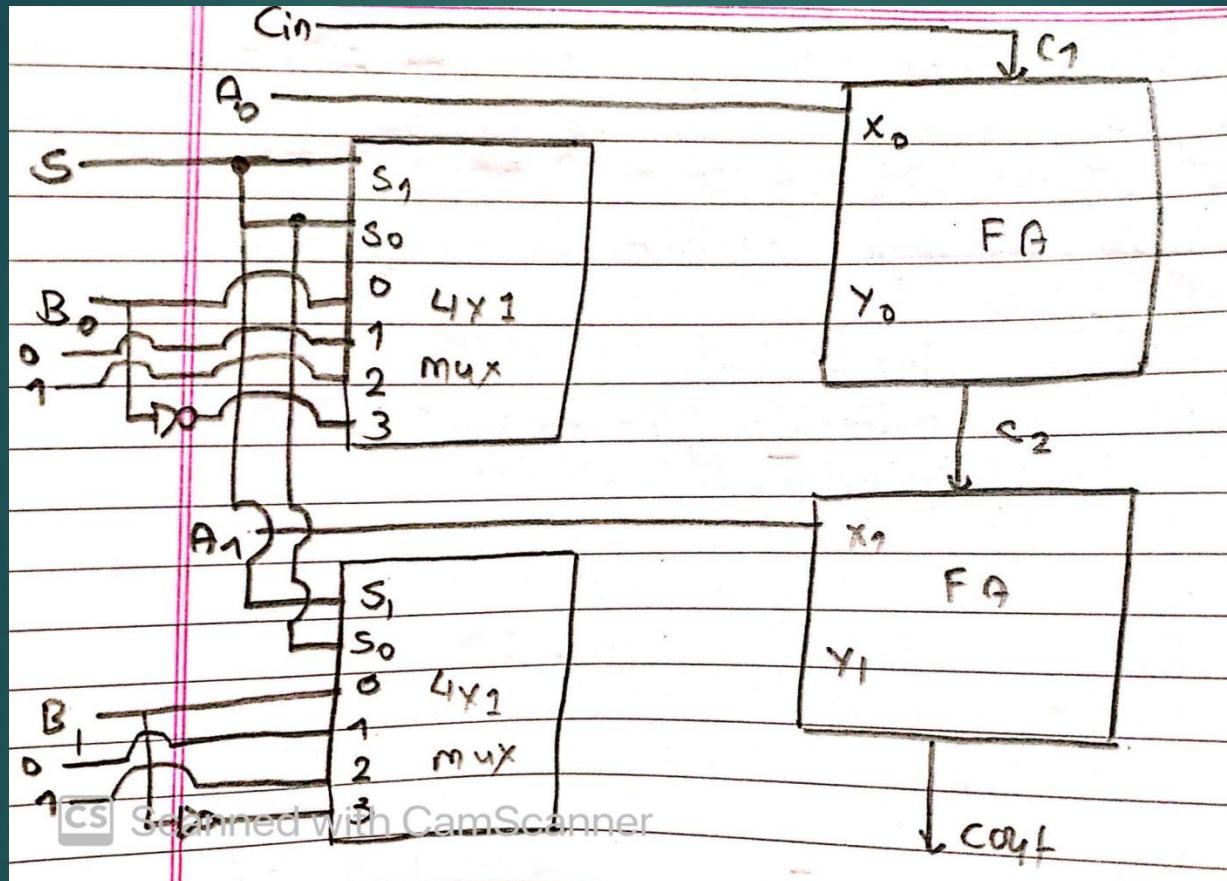
S	C in	X	$Y(A + B + C \text{ in})$
0	0	A	B
0	1	A	0
1	0	A	1
1	1	A	$\bar{B}$

0	0	B
0	1	0
1	0	1

1	1	$\bar{B}$
---	---	-----------

Note:



S1	S0	C in=0	C in=1
0	0	$F=A+B$	$F=A+B+1$
0	1	$F=A$	$F=A+1$
1	0	$F=\bar{B}$	$F=\bar{B}+1$
1	1	$F=A+\bar{B}$	$F=A+\bar{B}+1$

$$\begin{array}{r}
 0 \quad 0 \quad B \\
 \hline
 0 \quad 1 \quad 0 \\
 1 \quad 0 \\
 1 \quad 1 \quad \bar{B}
 \end{array}$$

S1	S0	C in	function F	remark
0	0	0	$F=A+B$	Addition of A & b
0	0	1	$F=A+B+1$	Addition with carry
0	1	0	$F=A$	Transfer of D
0	1	1	$F=A+1$	Increment
1	0	0	$F=\bar{B}$	Complement of B
1	0	1	$F=\bar{B}+1$	
1	1	0	$F=A+\bar{B}$	
1	1	1	$F=A+\bar{B}+1$	

# Do yourself

- ▶ Processor Organization