DIGITAL SYSTEM DESIGN
EC-503

BY

DR. GAURAV KUMAR BHARTI

ASSISTANT PROFESSOR

DEPT. OF ELECTRONICS & COMMUNICATION ENGINEERING

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY BHOPAL

REVIEW OF SEQUENTIAL CIRCUITS

COMBINATIONAL VS SEQUENTIAL

Combinational

- Combinational circuits generate output solely based on the current inputs.
- These circuits have no internal memory elements to store past inputs or outputs.
- Output depends only on the present combination of inputs.
- Output changes instantly with any change in input.
- All outputs are available simultaneously.

➢ **Examples :**

- Logic Gates: Basic building blocks like AND, OR, NOT.
- **Multiplexers, Demultiplexers:** Devices that perform operations based on input combinations.

Sequential

- Sequential circuits include memory elements (usually flip-flops) to store past inputs and outputs.
- Output not only depends on current inputs but also on the circuit's past states.
- Utilizes feedback to retain information.
- Synchronized operation with a clock signal.
- Outputs are generated sequentially, one after another.

➢ **Examples :**

- **Registers:** Used for data storage in CPUs.
- **Counters:** Sequential circuits that count or generate sequences.
- **Memory Units:** Store information for later retrieval.

Combinational and sequential circuits

- The Fig1 and Fig2 shows the pictorial representation of circuits

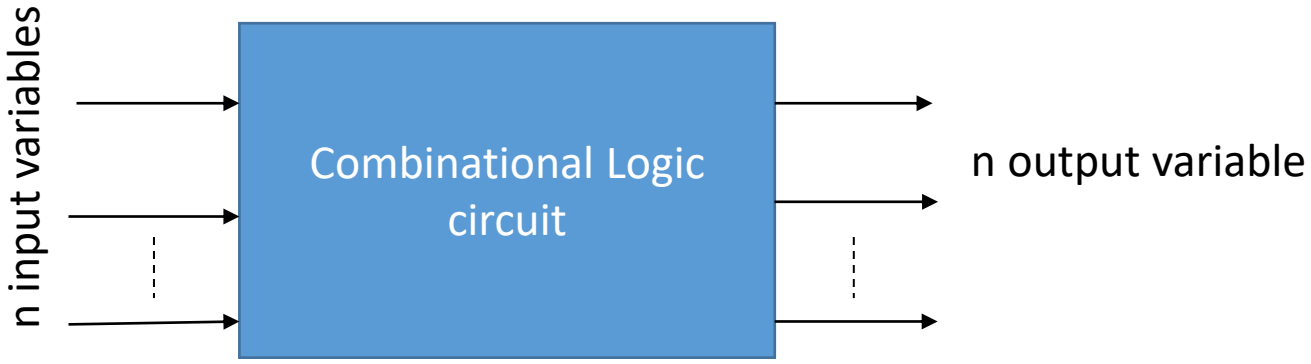


Fig 1. Combinational Logic Circuit

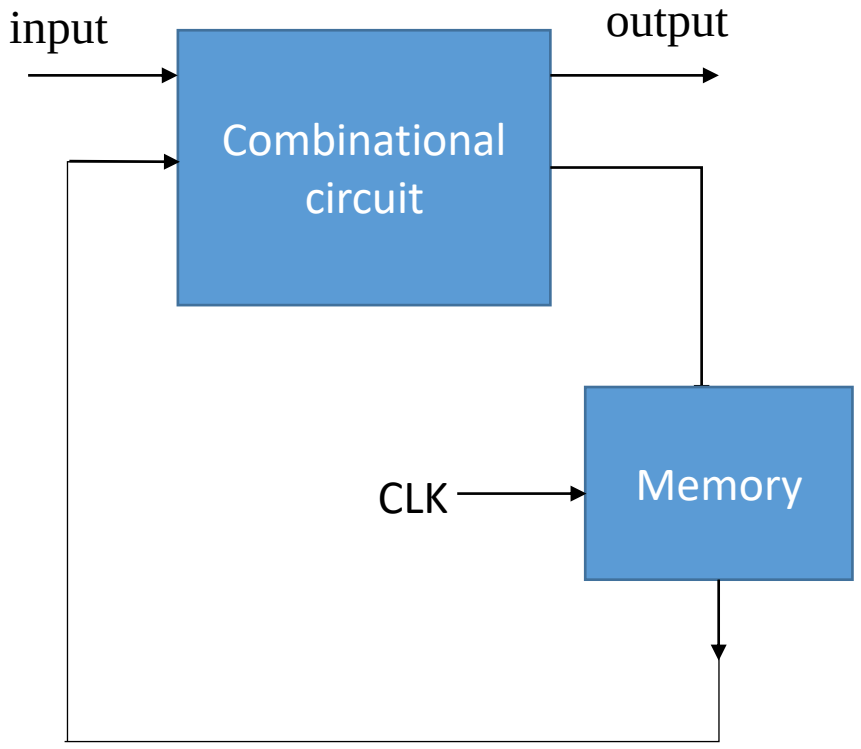


Fig 2. Sequential circuit

Review of Sequential circuits

- Sequential circuits are commonly used in digital systems to implement state machines, timers, counters, and memory elements.
- The memory elements in sequential circuits can be implemented using flip-flops, which are circuits that store binary values and maintain their state even when the inputs change.
- ***There are two types of sequential circuits:***

    1.Asynchronous sequential circuit

    2.Synchronous sequential circuits.

- *Asynchronous sequential circuit* **do not use a clock signal** but uses the pulses of the inputs.
- *Synchronous sequential circuits* These circuits **uses clock signal** and level inputs (or pulsed) (with restrictions on pulse width and circuit propagation).

Flip-Flops vs Latch

- A FF is an electronic device that has two stable states. One state is assigned the logic 1 value and the other is the logic 0.
- These circuits are binary cells capable of storing one bit of information.
- A latch is a bistable circuit that is the fundamental building block of a flip-flop. It exists in one of the two states (e.g. 1 and 0), and in the absence of the input, it remains in that state.
- The following Fig. 3 illustrate a simple FF or 1 bit memory (i.e. it can store one bit of information y = 0 or y = 1) and since this information is locked or latched so, this FF is known as. a latch

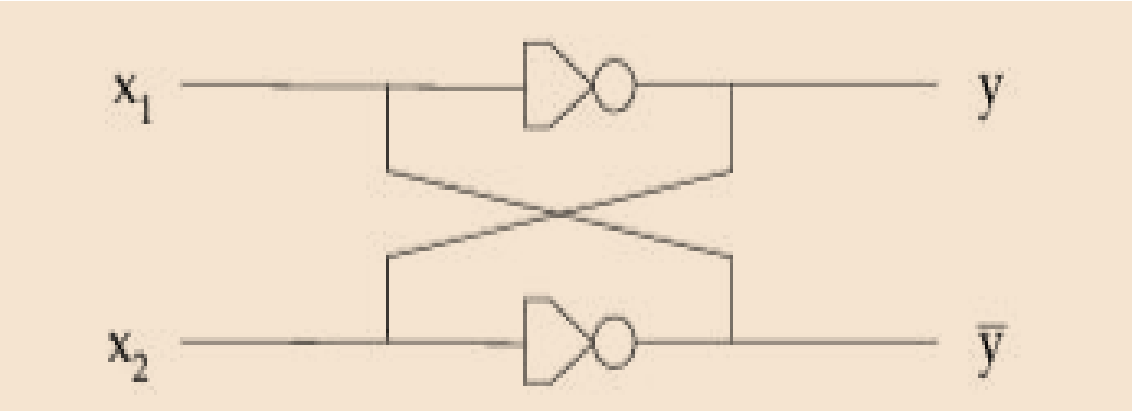


Fig 3 : Simple Flip Flop or latch

Flip-Flops

- *SR (Set-Reset) Flip Flop* 2ANDand2NOR

Case 1: S=0, R=0
Gate1 = 0, Gate2 = 0, Gate3/Q(n+1) = Q, Gate4/Q(n+1)' = Q'
Note: Since one input of both gate3 and gate4 is 0 and both the gates are NOR gates the output of both the gates will complement the second input as per the property of NOR gates.
Case 2: S=0, R=1
Gate1 = 1, Gate2 = 0, Gate3/Q(n+1) = 0, Gate4/Q(n+1)' = 1
Note: Since one input of gate3 is 1 and gate3 is a NOR gate, output gate3 will be 0 irrespective of other input as per the property of NOR gate.
Case 3: S=1, R=0
Gate1 = 0, Gate2 = 1, Gate4/Q(n+1)' = 0, Gate3/Q(n+1) = 1
Note: Since one input of gate4 is 1 and gate4 is a NOR gate, output gate4 will be 0 irrespective of other input as per the property of NOR gate.
Case 4: S=1, R=1
Gate1 = 1, Gate2 = 1, Gate4/Q(n+1)' = 0, Gate3/Q(n+1) = 0

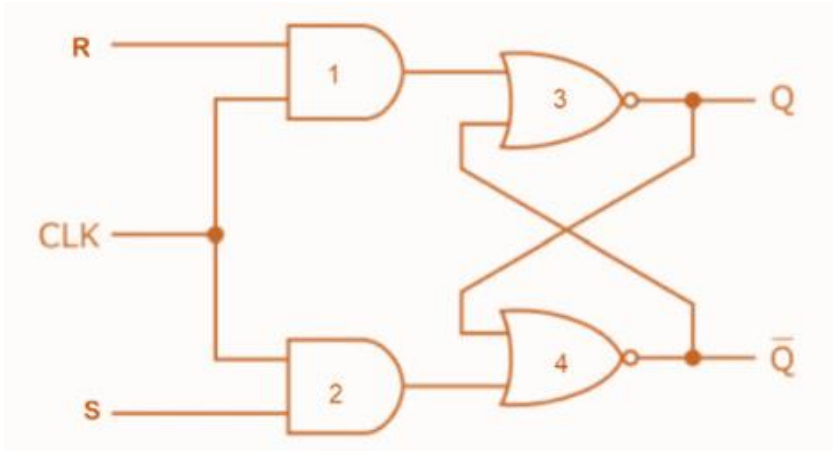


Fig 4 SR Flip-Flop

S	R	Qn+1	State
0	0	Qn	Hold
0	1	0	Reset
1	0	1	Set
1	1	X	Invalid

Fig 5 : Truth table of SR

Characteristic Table and Boolean expression

Characteristic Table

S	R	Qn	Q(n+1)
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	X
1	1	1	X

Fig 6 : Characteristic Table

K-Map

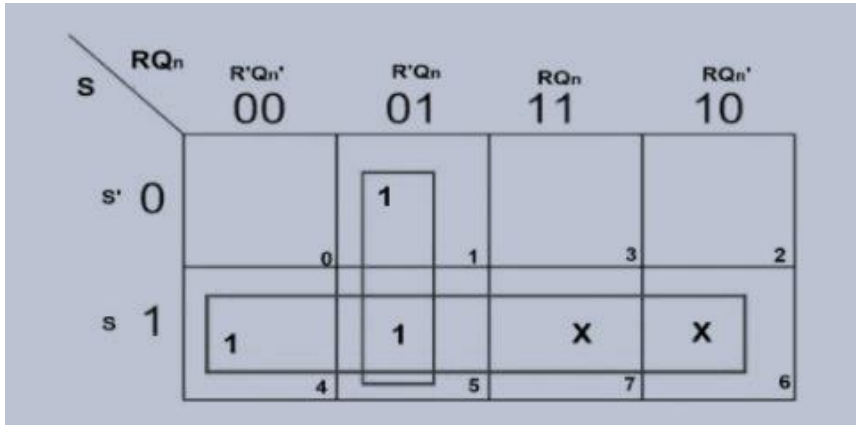


Fig 7 : K-Map

From the K-map we get 2 pairs. On solving both we get the following characteristic equation:

$Q(n+1) = S+R'Qn$

D FF(Works on output follows input principle)

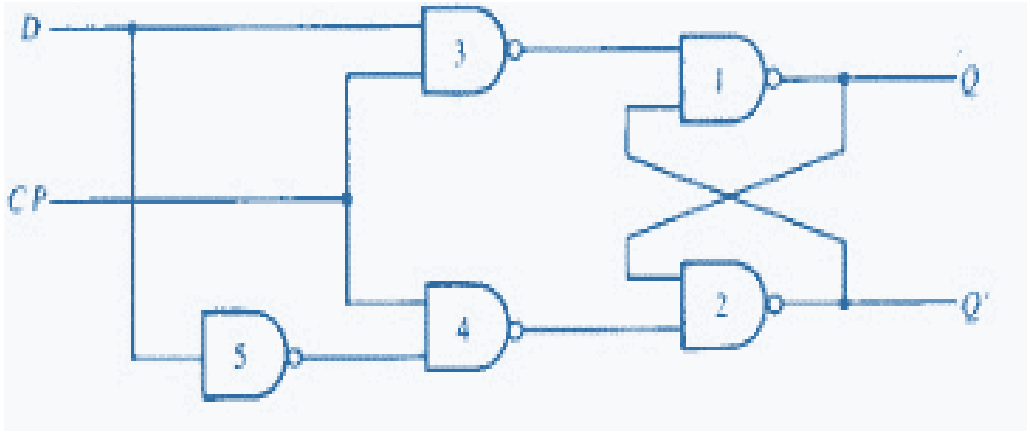


Fig 8: D Flip-Flop

Q	D	Q(t+1)
0	0	0
0	1	1
1	0	0
1	1	1

Fig 9: Truth Table of D Flip-Flop

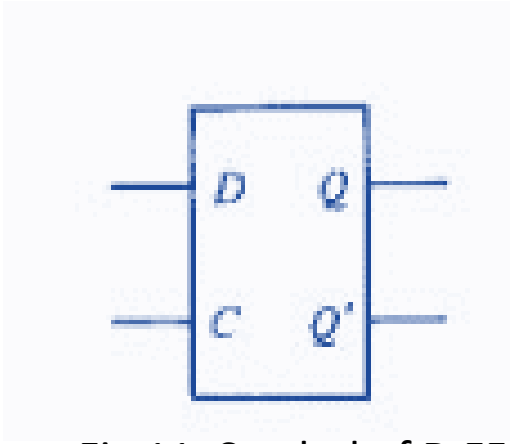


Fig 11 :Symbol of D FF

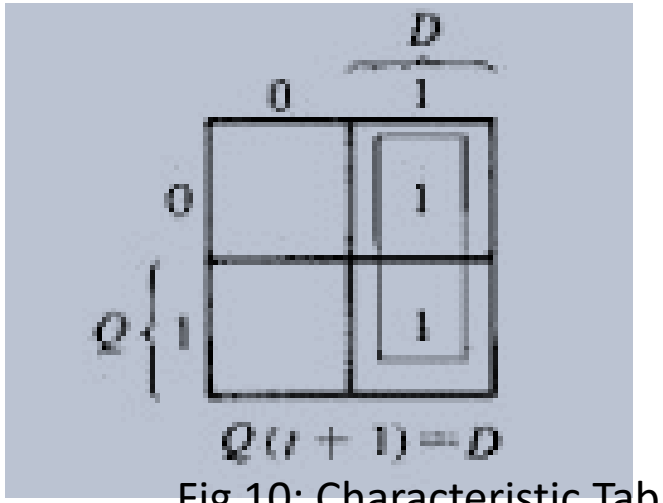


Fig 10: Characteristic Table

JK-FF(It operates with only positive clock transitions or negative clock transitions)

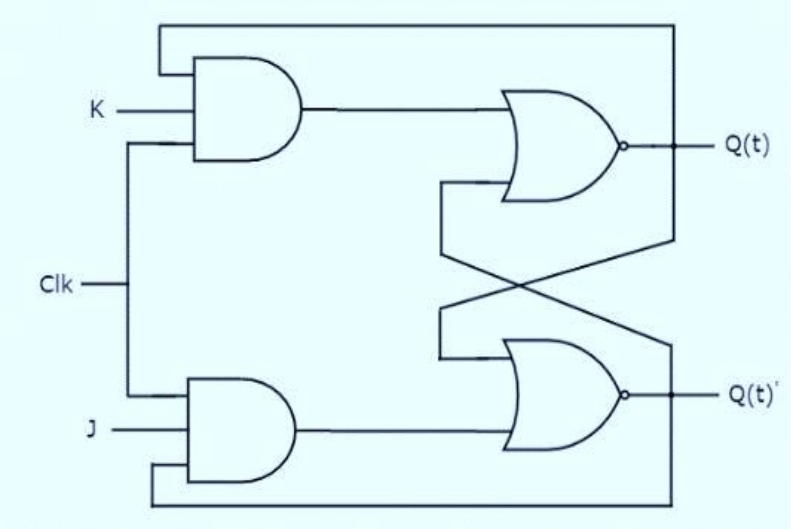


Fig 12: JK FF

J	K	$Qt+1$
0	0	Qt
0	1	0
1	0	1
1	1	Qt'

Fig 13: State Table of JK FF

Present Inputs		Present State	Next State
J	K	Qt	$Qt+1$
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	0

Fig 14: Characteristic Table

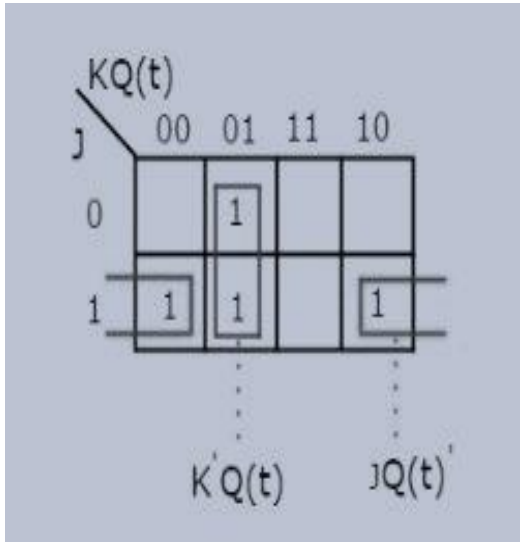


Fig 15: K Map of JK FF

Simplified expression of K map is Q(t+1)=JQ(t)'+K'Q(t)

T-FF (It is obtained by connecting the same input 'T' to both inputs of JK flip-flop. It operates with only positive clock transitions or negative clock transitions.)

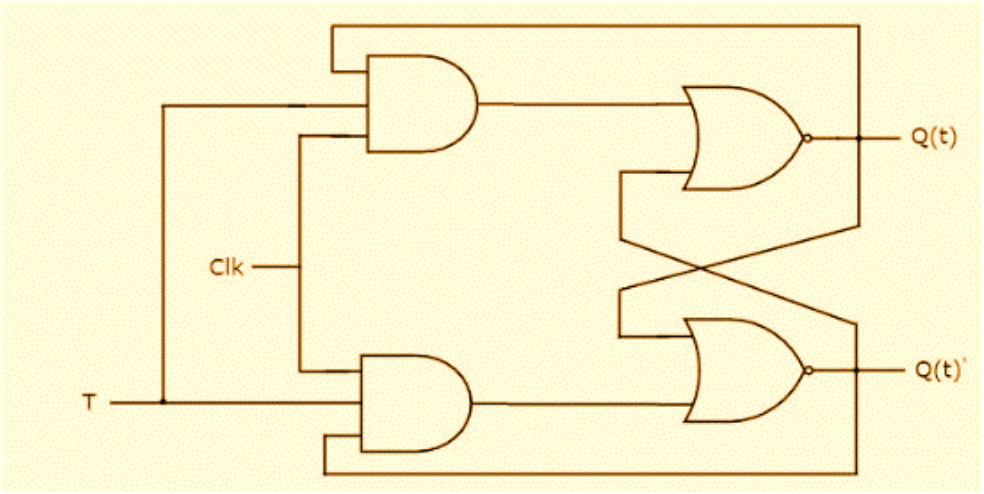


Fig 16: T Flip-Flop

Inputs	Present State	Next State
T	Qt	$Qt+1$
0	0	0
0	1	1
1	0	1
1	1	0

Fig 18: Characteristic Table of T-FF

D	$Qt+1$
0	Qt
1	Qt'

Fig 17 : State Table of T-FF

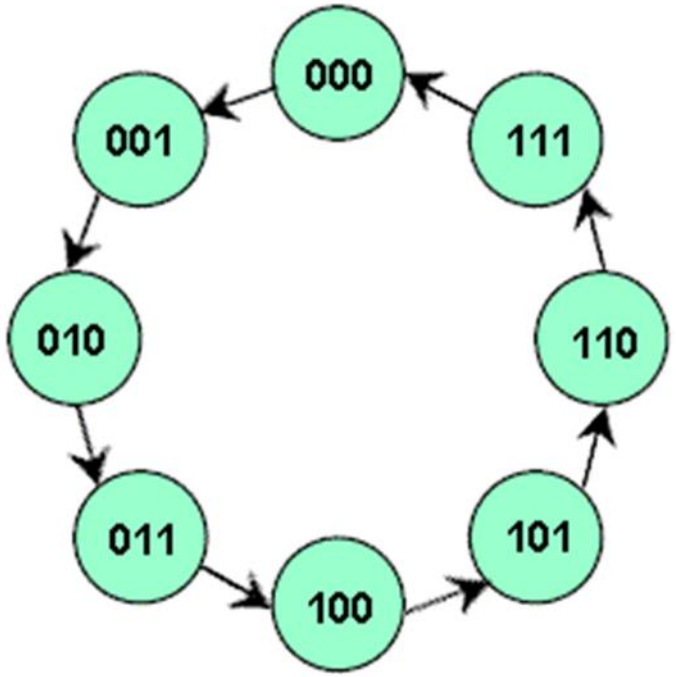
From the above characteristic table, By solving we can write the **next state equation** as

$$Q(t+1)=T'Q(t)+TQ(t)'$$

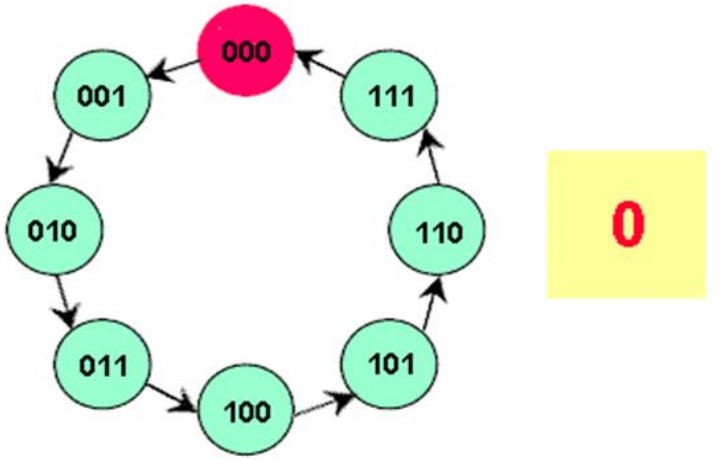
$$\Rightarrow Q(t+1)=T\oplus Q(t)$$

Counters

- A counter is first described by a state diagram, which is shows the sequence of states through which the counter advances when it is clocked.



- The circuit has no inputs other than the clock pulse.
- No outputs other than its internal state (outputs are taken off each flip-flop in the counter).
- The next state of the counter depends entirely on its present state, and the state transition occurs every time the clock pulse occurs.



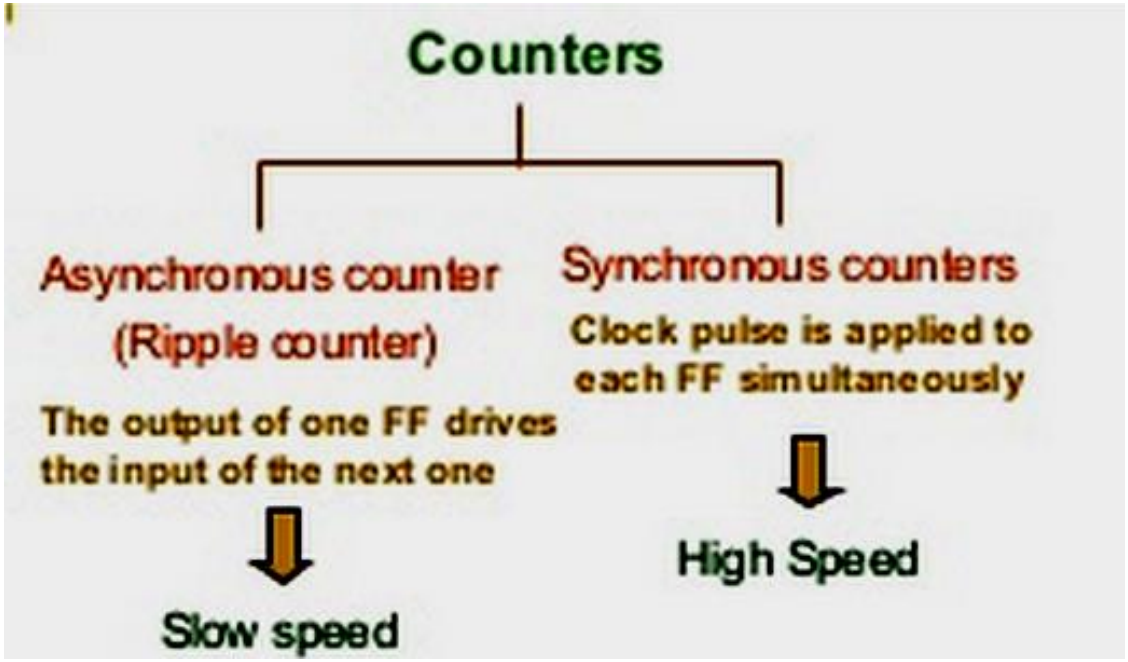
- Once the sequential circuit is defined by the state diagram, the next step is to obtain the next-state table, which is derived from the state diagram

Present State Q2 Q1 Q0	Next State Q2 Q1 Q0
0 0 0	0 0 1
0 0 1	0 1 0
0 1 0	0 1 1
0 1 1	1 0 0
1 0 0	1 0 1
1 0 1	1 1 0
1 1 0	1 1 1
1 1 1	0 0 0

- The number of distinct states the counter can take is 2^N. It depends on the number of flip-flops that the counter has.

- There are two types of counting namely Up counting & Down counting
- In Up counting, If start the count with 0, the number of states will go to 2^{N-1}
- In Down counting, the count starts from 2^{N-1} to 0.
- Let us assume that there are three flip-flops have the outputs A B C as QA QB and QC and the count starts from 0 0 0 then 0 0 1 .... 0 1 1.
- The complementary outputs (Q bar) Q'A Q'B and Q'C and this would start from 1 1 1, 1 1 0 .... 000.
- When take the output from Q, it becomes up counting as well as the output of Q' or Q bar becomes down counting.
- So it is a very major use of Flip-flops, (i.e.) use flip-flops in a chain and for counting events.
- If counter start from 000 to 111 and then again from 000, it is referred to full cycle.
- The total number of states the counter has is called MODULO or MODULUS.
- Sometimes, not want to count till the end of the last state or want to terminate the count at the count which is less than maximum and go back to 0.

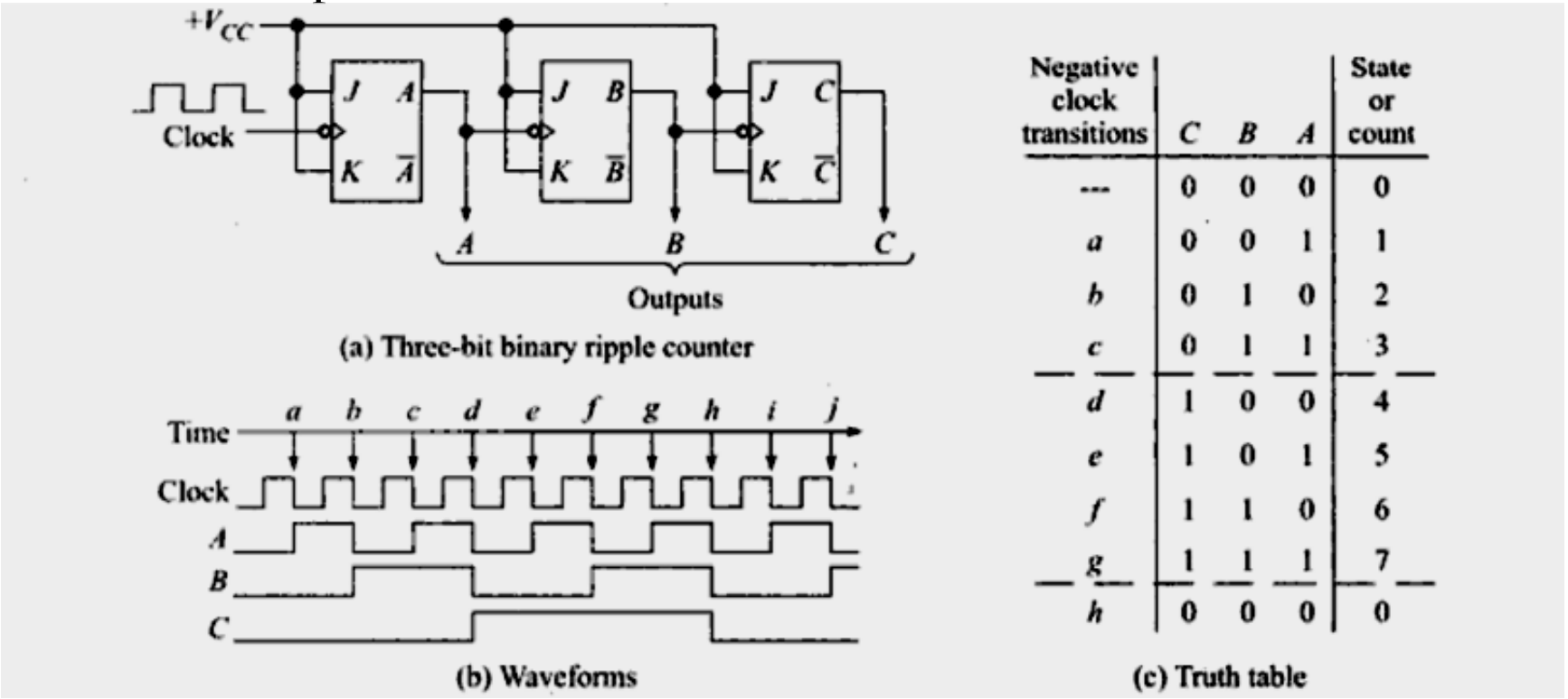
- For example, the counter has four bits and it has 0 to 15 states and may want to terminate with 9 that is 0 to 9 and start again with 0, there will be a decimal counter.
- By using this decimal counter, able to start at any count or end at any count within the possible range.



Asynchronous (Ripple) Counters

- When the output of a flip-flop is used as the clock input for the next flip-flop, the counter is referred to as ripple counter or asynchronous counter.
- Asynchronous counters: the flip-flops do not change states at exactly the same time as they do not have a common clock pulse.
- In ripple counters, the input clock pulse "ripples" through the counter – the drawback is cumulative delay.
- A counter has a natural count of 2^n where n is the number of FFs in the counter.
- Counter can operate in either a count-up or count-down mode.
- Digital clock is an typical application for counters.

- A binary ripple counter can be constructed using clocked JK FFs.
- The following figure shown Negative Edge Triggered, JK FFs connected in cascade.
- The system clock drives FF A. The output of A drives B, the output of B drives FF C.
- All the J & K inputs are tied to $+V_{cc}$. This means that each FF will change state with a NT at its clock input.

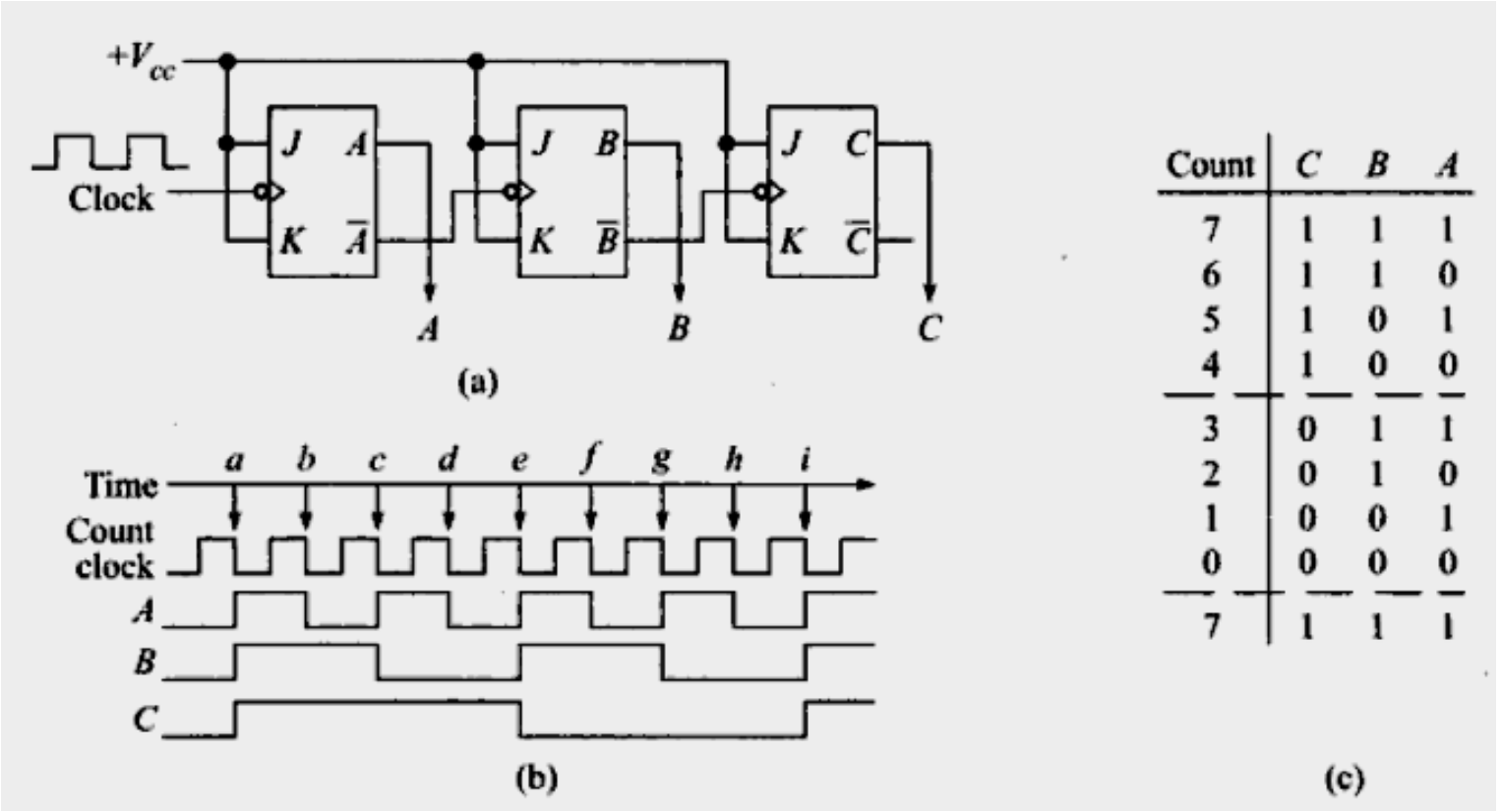


Negative clock transitions	C	B	A	State or count
---	0	0	0	0
a	0	0	1	1
b	0	1	0	2
c	0	1	1	3
d	1	0	0	4
e	1	0	1	5
f	1	1	0	6
g	1	1	1	7
h	0	0	0	0

(c) Truth table

Ripple Down Counter [3-bit ripple counter]

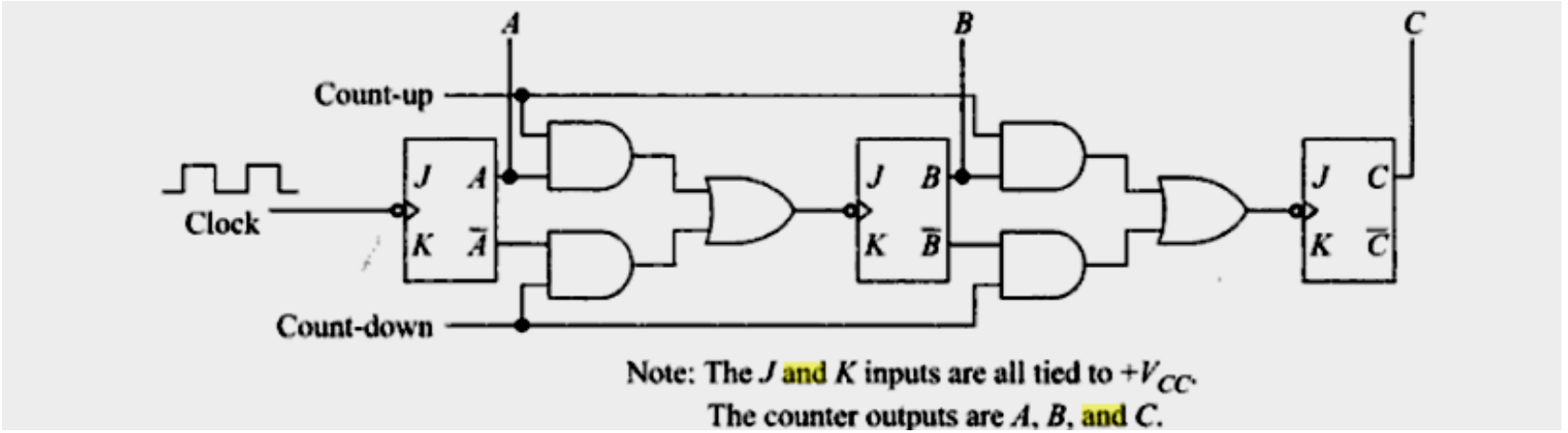
- The system clock is still used at the clock input to FF a, but the complement of A, A' is used to drive FF B, likewise, B' is used to drive FF C.



Count	C	B	A
7	1	1	1
6	1	1	0
5	1	0	1
4	1	0	0
3	0	1	1
2	0	1	0
1	0	0	1
0	0	0	0
7	1	1	1

(c)

- A 3-bit asynchronous up-down counter that counts in a straight binary sequence is shown below.
- It is simply a combination of the two counters discussed previously.
- For this, counter to progress through a count-up sequence, it is necessary to trigger each FF with the true side of the previous FF (as opposed to the complement side).
- If the count-down control line is low and count-up control line high, and the counter will have count-up wave forms.
- On the other hand, if count-down is high and count-up is low, each FF will be triggered from the complement side of the previous FF.
- The counter will then be in a count-down mode and will progress through the waveforms.

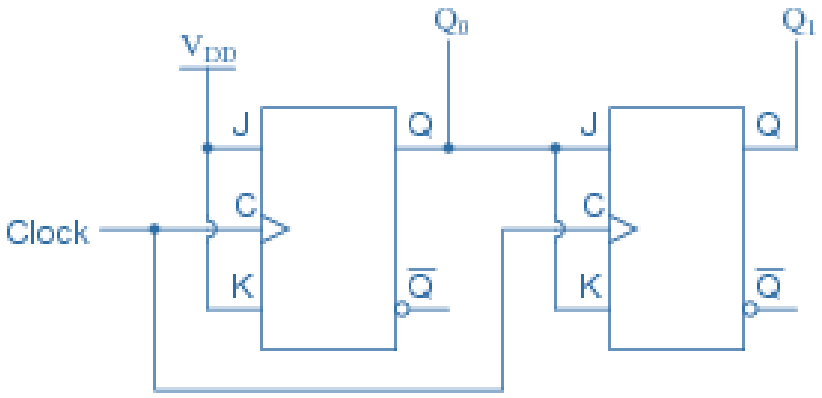


Note: The J and K inputs are all tied to $+V_{CC}$.
The counter outputs are A, B, and C.

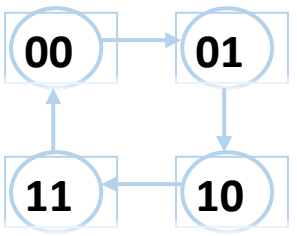
Synchronous (Parallel) Counters

- Asynchronous counters are not useful at very high frequencies, especially for counters with large number of bits.
- Another problem caused by propagation delays in asynchronous counts occurs when try to electronically detect (decode) the counter outputs states.
- Both of these problems can be overcome by the use of a synchronous or parallel counter.

- Synchronous (parallel) counters: the clock inputs of all the flip-flops are connected together and are triggered by the input pulses.
- Clock pulses are applied to the input of all FFs.
- All the flip-flops change state simultaneously (in parallel).
- T=0 or J=K=0 (FF does not change state)
- T=1 or J=K=1 (FF complements)
- Design these counters using the sequential logic design process.
- Example: 2-bit synchronous binary counter (using T flip-flops, or JK flip-flops with identical J,K inputs).



Present state		Next state	
A_1	A_0	A_1^+	A_0^+
0	0	0	1
0	1	1	0
1	0	1	1
1	1	0	0



References

- chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://ia600607.us.archive.org/3/items/DigitalLogicAndComputerDesignByM.MorrisMano2ndEdition/Digital%20Logic%20And%20Computer%20Design%20By%20M.%20Morris%20Mano%20%282nd%20Edition%29.pdf
- https://www.geeksforgeeks.org/introduction-of-sequential-circuits/
- https://www.electronicsforu.com/technology-trends/learn-electronics/sr-flip-flop-circuit-truth-table-limitations-applications