**UNIT III**

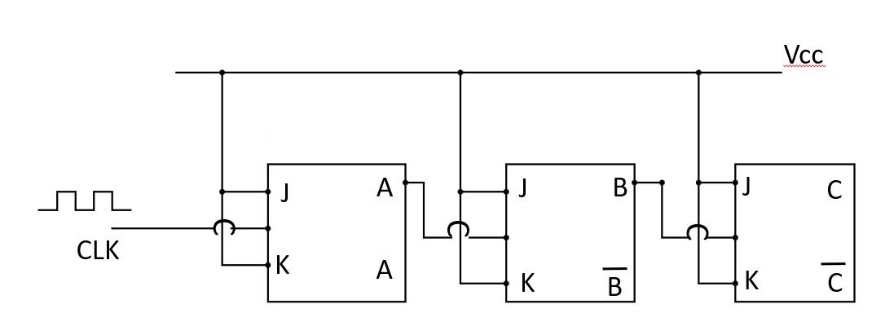
**COUNTERS:**

A counter is probably one of the most useful and versatile subsystems in a digital system. A counter driven by a clock can be used to count number of clock pulses. Since the clock pulses occur at known intervals, the counter can be used as an instrument for measuring time and therefore period or frequency.

**3-Bit Binary Serial Counter, 3- Bit Asynchronous Counter, 3 Bit Ripple Counter**

A binary counter is called serial counter because the flipflops are connected to one another in serial way. A binary serial counter is also called ripple counter because the clock pulse moves like a ripple in the water. Further, a binary serial counter is also called Asynchronous counter because the output of the previous flipflop drives the clock input of successive flipflop. So, the clock pulses move in Asynchronous way hence the counter is known as Asynchronous counter. The output of the counter is expressed in binary form hence it is called Binary counter.

A 3-bit (Mod 8) serial counter with negative edge triggering can be constructed using three flipflops as shown in the figure 1.



**FIG 1: 3 -BIT SERIAL COUNTER/BINARY COUNTER/ASYNCHRONOUS COUNTER**

The three flipflops are connected in cascade. A clock pulse drives first flipflop. The output of first flip flop (A) drives clock input of B Flip flop and output of B Flip flop in turn drives clock input of C Flip flop. All J and K inputs are tied to Vcc. This means each Flip flop will toggle on negative transition of the clock pulse.

Lets assume all Flipflops are initially in RESET condition to give 0 output. Therefore, the output condition is C B A = 0 0 0.

When a first clock pulse arrives and when it goes low, the flipflop A toggles and changes from low state to high state. So, at the end of the first clock pulse the output becomes C B A = 0 0 1.

Note that A output has gone from 0 to 1 which is a positive change. When fed to clock input of B flipflop. Since, this is a positive change and has no effect on output of B flipflop because Flipflop only responds to negative going change of the clock pulse.

When a second clock pulse arrives, A flipflop toggles again on the negative going edge of the clock pulse. As it changes, A now goes from 1 to 0, which is negative going change. This negative going change triggers B Flipflop. Therefore, B goes from 0 to 1. This is a positive going change and it has no effect on C. So at the end of the second clock pulse, the output condition becomes C B A = 0 1 0.

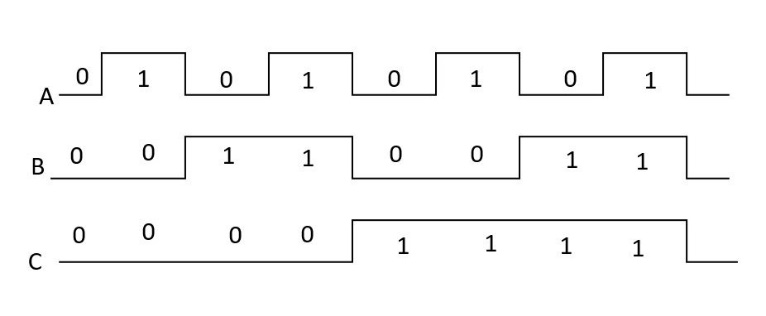
After the third clock pulse, A again toggles. It goes from 0 to 1, which is a positive going change. This will not change state of B Flip flop and hence C will also not change the state. Therefore, at the end of the third clock pulse, the output becomes C B A = 0 1 1.

If we notice the output condition of the truth table, then it is binary equivalent of number of clock pulses that have hit A Flipflop. The truth table of 3-bit Binary counter is as shown below.

|  |  |  |  |
| --- | --- | --- | --- |
| C | B | A | Clock |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 2 |
| 0 | 1 | 1 | 3 |
| 0 | 0 | 0 | 4 |
| 0 | 0 | 1 | 5 |
| 0 | 1 | 0 | 6 |
| 0 | 1 | 1 | 7 |

**TABLE 1: TRUTH TABLE OF 3 -BIT SERIAL COUNTER/BINARY COUNTER/ASYNCHRONOUS COUNTER**

The maximum number of natural counts a counter can do is given by 2n-1, where n is number of Flipflops used to construct the counter. The waveform diagram of 3-bit serial counter is as shown in the figure 2 below.



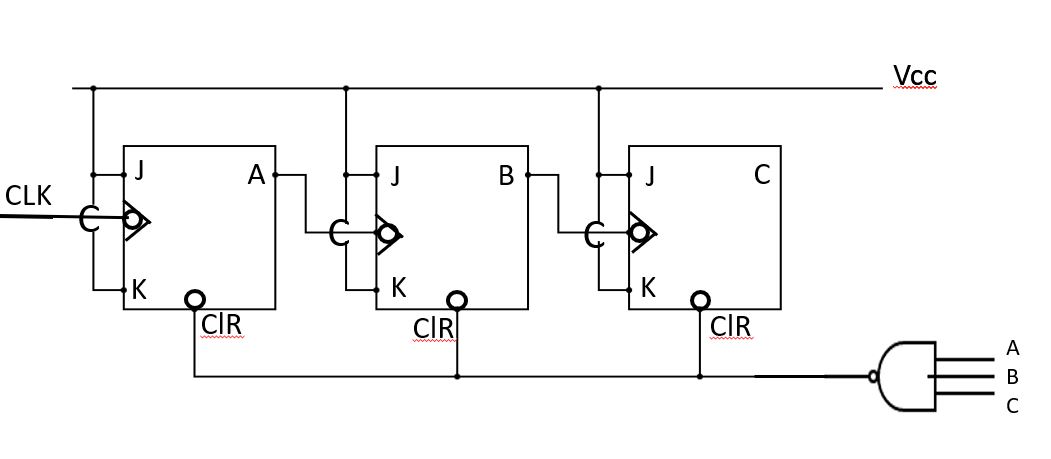
**FIG 2: WAVE FORM DIAGRAM OF 3 -BIT SERIAL COUNTER**

As the clock pulse proceeds, the output proceeds through binary number equivalent to number of clock pulses that has hit counter, hence it is a binary counter.

**Asynchronous Counter using feedback (Mod 7, 6, 5)**

Total number of count or states through which the counter can progress is given by 2n, where n is total number of flipflops used to make counter. For e.g. a counter made using three flipflops, can count maximum 23i.e. 8 states. A 3 Flipflop counter is often referred as Mod-8 counter. One method used to cause a counter to skip counts is to feedback a signal from output of some flipflop to previous flipflop.

Consider Mod-8 serial counter shown in figure 2 along with the truth table shown in Table 1. We wish to construct Mod-7 counter from it and hence we are required to skip one state. Close examination of truth table reveals that count 7 i.e. A B C = 1 1 1 occurs only once during the natural eight counts. If A B C are connected to NAND gate as shown in figure 3, we can skip one state.



**FIG 3: Mod -7 Serial Counter**

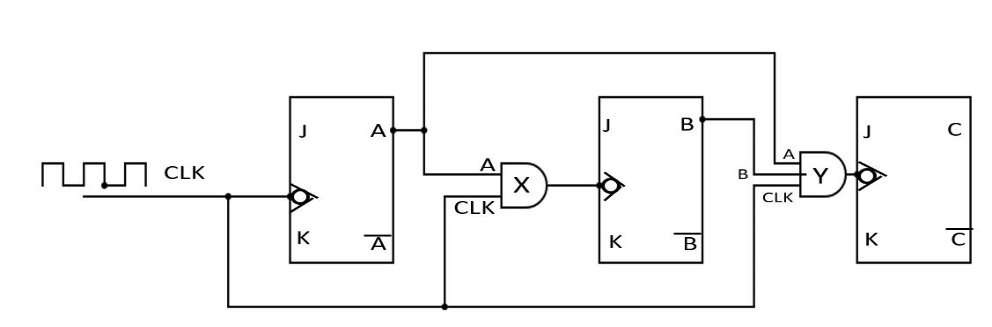
The output of NAND gate is connected to clear input of all flipflops. The output of the NAND gate will be high for all input conditions from 0 to 6 and hence will not affect the working of counter. When C B A =1 1 1, the output of NAND gate will be low and as the output of NAND gate is connected to clear of each flipflop, all the flipflops will be cleared to 0 immediately. Hence, it essentially skips this state and therefore it is Mod-7 counter. This type of counter is called Permuting counter since its natural count has been permuted.

For constructing Mod-6 counter we have to skip two states. So suppose that we need to skip states 6 and 7, then after 5th clock pulse, the counter should reset on 6th pulse. For this again we use the feedback technique. For Mod-6 counter the inputs to the NAND gate should be C=1, B=1 and Ā =1. The output of NAND gate is applied to clear input of all the flipflops then at C B Ā = 1 1 1input, the output of NAND gate will be low which will skip counts 6 and 7 and thus it is Mod-6 counter.

**Parallel Counter/ Synchronous Counter (Mod-8)**

The ripple counter is simplest to build but there is a limit to its highest operating frequency. Each flipflop has a delay time. In ripple counter this delay time is additive and total settling time for counters is approximately delay time times the total number of flipflops. This speed limitation can be overcome by the use of synchronous or parallel counter. The difference here is that every flipflop is triggered by the clock. Thus, they all make their transition simultaneously.

The construction of parallel binary counter is as shown in the figure 4 along with truth table 1.



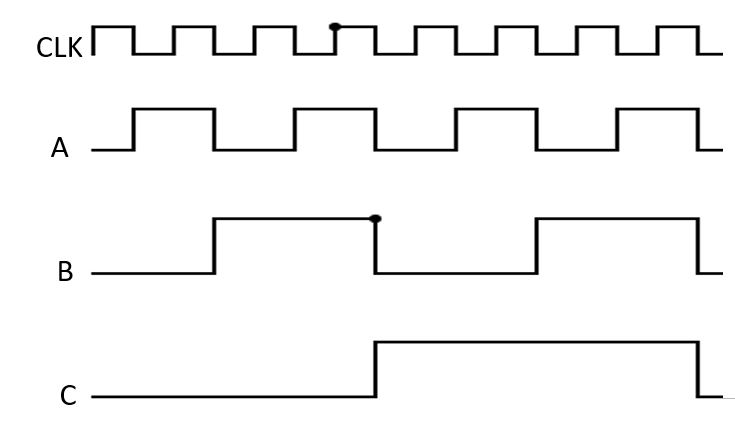
**FIG 4: Mod 8 Parallel Counter**

The clocked JK flipflop responds to negative transition of input clock pulse (1 → 0) and toggles as both J and K are high for all the three flipflops. Having no connection to J and K inputs is equivalent to having positive voltage or true level of these inputs.

Flipflop A changes state with each negative transition of the clock pulse. Initially we assume that all flipflops are in RESET condition. So, on the first clock pulse, when it goes negative, flip flop A will SET.

The output of A flipflop is input for AND gate X. X gate goes high whenever clock is high and A is high. Thus, on the next negative transition of the clock pulse, Flipflop B will toggle. Thus, flipflop B will change state after every two pulses.

Now, the output of B flipflop is fed to AND gate Y. The other inputs of AND gate Y are clock pulse and A. The output of AND gate Y will be high when A is high, B is high and clock is high. Thus, flipflop C changes state after every fourth clock pulse. Thus, if we totalize all outputs of all flipflops and see the truth table then it follows the binary sequence from 0 to 7 and hence it is Mod-8 parallel counter. Figure 5 shows waveform diagram for Mod 8 parallel counter.



**FIG 5: Waveform diagram of Mod-8 Parallel/Synchronous Counter**

**Mod-7 and Mod-6 Synchronous Counter:**

The parallel Mod 8 counter is shown in Fig. 4. It can be used as basis for building counters of other moduli. If we wish to construct Mod-7 parallel counter, it is necessary to find some means of eliminating one state from of eliminating one state from natural count sequence. The examination of the truth table shows that all flipflops are high during count 7. In changing from count 7 to count 0, all flipflops RESETS.

We need to find some means to prevent flip flop A from being RESET during the transition from count 7 to count 0 without affecting the operations of flipflop B and C. Thus, the counter will progress from count 7 directly to count 1 and count 0 will be skipped and this will result into Mod 7 counter. Circuit diagram of Mod-7 Parallel counter is shown in figure 6.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  |  |  |  | | --- | --- | --- | --- | | C | B | A | Clk | | 0 | 0 | 1 | 1 | | 0 | 1 | 0 | 2 | | 0 | 1 | 1 | 3 | | 1 | 0 | 0 | 4 | | 1 | 0 | 1 | 5 | | 1 | 1 | 0 | 6 | | 1 | 1 | 1 | 7 | |

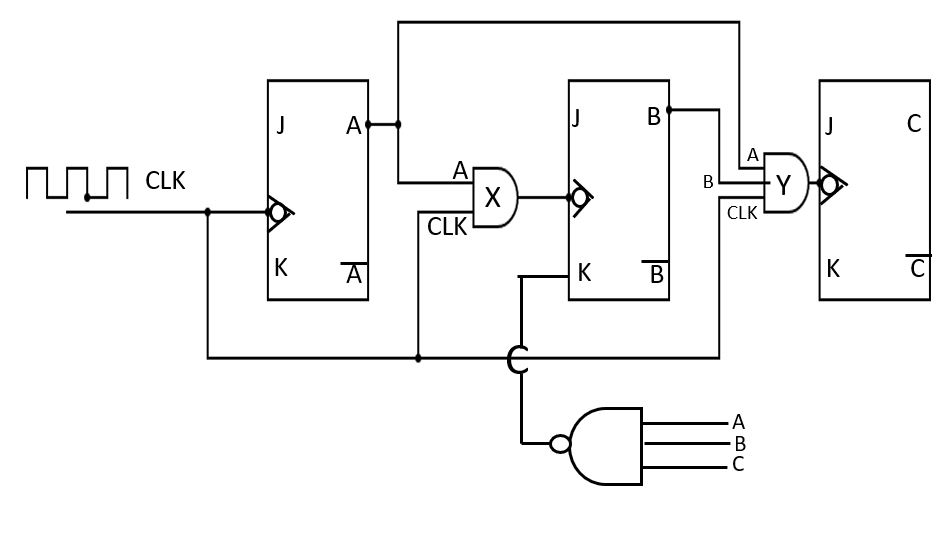
**FIG 6: Mod-7 Parallel/Synchronous Counter Table 2: Truth Table of Mod-7 Counter**

To construct Mod 7 counter from the basic Mod 8 counter, it is necessary to ensure that K input of flipflop A is held low during count 7. This is easily accomplished by using true outputs of all flipflops as input to NAND gate and output of NAND gate is connected to K input of flipflop A. Thus, K input of flipflop A will be held low only during count 7 and will be high for all other times (1 to 6 counts). During count 7, J is high and K is low and so on negative transition of clock pulse, flipflop A will SET and thus state 0 is skipped. This results into Mod 7 counter.

**Mod 6 Synchronous Counter:**

The configuration of Mod 7 counter can easily be modified to form Mod 6 counter. If the output of NAND gate is removed from K input of flipflop A and connected to K input of flipflop B, then Mod 6 counter will be formed. Here, two states will be skipped, 0 and 1. The required connections for Mod 6 counter are shown in figure 7. The truth table of Mod 6 counter along with waveform diagram are shown in Table and figure 8 respectively.

**FIG 5: Waveform diagram of Mod-8 Parallel/Synchronous Counter**



**FIG 7: Circuit diagram of Mod-6 Parallel/Synchronous Counter**

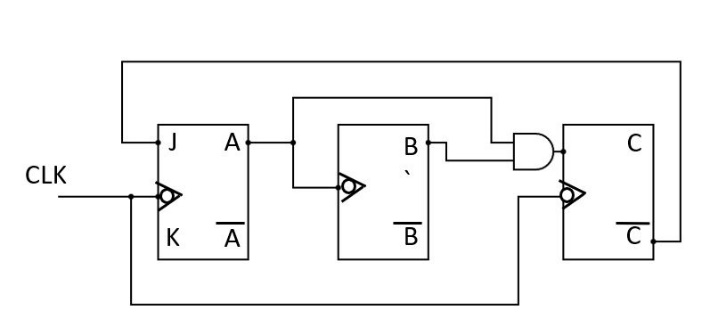
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | C | B | A | Clk | | 0 | 1 | 0 | 2 | | 0 | 1 | 1 | 3 | | 1 | 0 | 0 | 4 | | 1 | 0 | 1 | 5 | | 1 | 1 | 0 | 6 | | 1 | 1 | 1 | 7 | |  |

**Table 3: Truth Table of Mod-6 Fig 8: Wave form diagram of Mod-6 Parallel/Synchronous Counter Parallel/ Synchronous Counter**

**Mod 5 Combinational Counter:**

Asynchronous counter has main drawback of delay time (as it is additive) and in Synchronous counter the main draw back is too much wiring and complexity of circuit and many components. The best compromise between these two limitations is the Combinational counter/Series Parallel counter.

Consider Mod 5 counter as shown in the figure 9 below.



**Fig. 9: Circuit diagram of Mod 5 Combinational Counter**

Flipflop A changes state each time the clock goes negative except during the transition from count 4 to count 0. Note that is high during all counts except count 4. Thus if is connected to J input of Flipflop A, we will have desired waveform. This is true, since J and K inputs to Flipflop A are both true for all counts except count 4. Thus, flipflop toggles each time the clock goes negative. During count 4, J is low and when clock goes negative the flipflop will be prevented from being SET.

Flipflop B must change state each time A goes negative. The clock input of flipflop B is driven by A.

Flipflop C is driven by the clock pulse. While J input is low and K input is high and whenever clock goes negative, the flipflop will RESET. From the truth table and waveform diagram, C will be high on count 4 and for rest of all counts it is low. The necessary levels of J input is obtained by ANDing the true outputs of Flipflops A and B. Since A and B are both high only during count 3, when the clock goes negative during the transition from count 3 to count 4. Flipflop C will be SET. At all other times J input to flipflop C is low and it is held in the RESET state. Fig. 10 shows waveform diagram

and Table 4 shows truth table of Mod 5 combinational counter.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  |  |  |  | | --- | --- | --- | --- | | C | B | A | CLK | | 0 | 0 | 0 | 0 | | 0 | 0 | 1 | 1 | | 0 | 1 | 0 | 2 | | 0 | 1 | 1 | 3 | | 1 | 0 | 0 | 4 | |

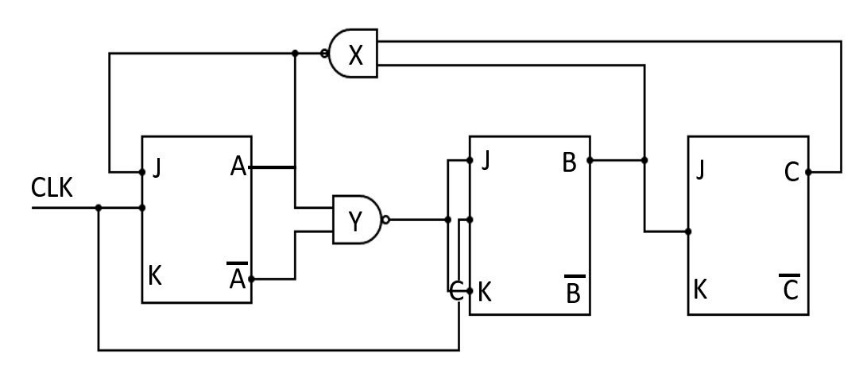
**Fig 9: Wave form diagram of Mod-5 Table 4: Truth Table of Mod-5   
 Combinational Counter Combinational Counter**

**Mod 7 Combinational Counter:**

A straight binary Mod 7 combinational counter can be easily constructed as shown   
 in figure 10. Flipflop C must change state each time B goes negative. Therefore B   
 can be used as trigger input to flipflop C.

Flipflop B must change state each time the clock goes negative and A is high. It must   
 change state during the transition from count 6 to count 0. Count 6 can be   
 recognized when both B and C are high. When both J and K are high, flipflop B will   
 toggle on negative transition of the clock pulse.

Careful examination shows that if output of NAND gate X is low, then output of   
 NAND gate Y will be high i.e. J and K both are high and so flipflop B will change   
 state. So anytime condition A+BC is true, flipflop B will change state.



**Fig. 10: Circuit diagram of Mod 7 Combinational Counter**

Finally notice that flipflop A must change state every time the clock goes negative   
 except during the transition from count 6 to count 0. Thus, the clock should be   
 used as the trigger input to flipflop A. Flipflop A can be prevented from going   
 positive during the transition from count 6 to count 0 by holding J input to flipflop   
 A low during count 6. The output of NAND gate X is low only during count 6, if it is   
 connected to J input of flipflop A.

Thus, the counter progresses automatically from the illegal state 7 to legal state 0.   
 The truth table and waveform diagram of Mod 7 combinational counter is shown   
 in Table 5 and figure 11 respectively.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  |  |  |  | | --- | --- | --- | --- | | C | B | A | Count | | 0 | 0 | 0 | 0 | | 0 | 0 | 1 | 1 | | 0 | 1 | 0 | 2 | | 0 | 1 | 1 | 3 | | 1 | 0 | 0 | 4 | | 1 | 0 | 1 | 5 | | 1 | 1 | 0 | 6 | |

**Fig 11: Wave form diagram of Mod-7 Table 5: Truth Table of Mod-7   
 Combinational Counter Combinational Counter**