Counters and Registers

Counters and registers belong to the category of MSI sequential logic circuits. They have similar architecture, as both counters and registers comprise a cascaded arrangement of more than one flipflop with or without combinational logic devices. Both constitute very important building blocks of sequential logic, and different types of counter and register available in integrated circuit (IC) form are used in a wide range of digital systems. While counters are mainly used in ounting applications, where they either measure the time interval between two unknown time instants or measure the frequency of a given signal, registers are primarily used for the temporary storage of data present at the output of a digital circuit before they are fed to another digital circuit. We are all familiar with the role of different types of register used inside a microprocessor, and also their use in microprocessor-based applications.

Because of the very nature of operation of registers, they form the basis of a very important class of counters called shift counters. In this chapter, we will discuss different types of counter and register as regards their operational basics, design methodology and application-relevant aspects. Design aspects have been adequately illustrated with the help of a large number of solved examples. A

comprehensive functional index of a large number of integrated circuit counters and registers is given towards the end of the chapter.

Ripple (Asynchronous) Counter

A *ripple counter* is a cascaded arrangement of flip-flops where the output of one flip-flop drives the clock input of the following flip-flop. The number of flip-flops in the cascaded arrangement depends upon the number of different logic states that it goes through before it repeats the sequence, a parameter known as the modulus of the counter.

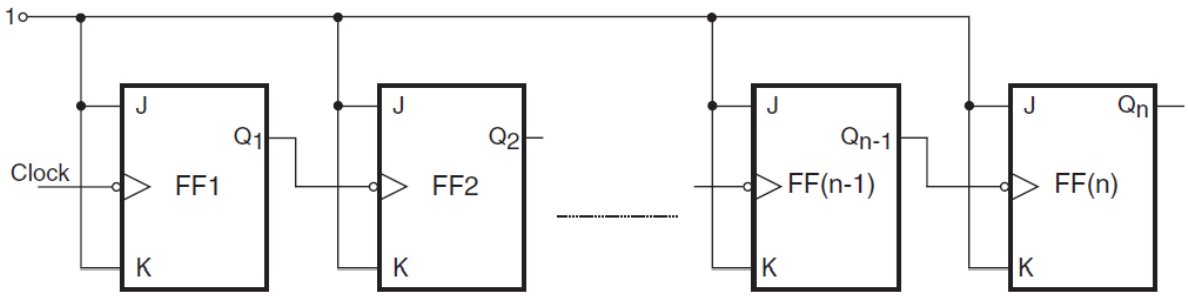


Figure 1 Generalized block schematic of n-bit binary ripple counter.

Propagation Delay in Ripple Counters

A major problem with ripple counters arises from the propagation delay of the flip-flops constituting the counter. As mentioned in the preceding paragraphs, the effective propagation delay in a ripple counter is equal to the sum of propagation delays due to different flip-flops. The situation becomes worse with increase in the number of flip-flops used to construct the counter, which is the case in larger bit counters. Coming back to the ripple counter, an increased propagation delay puts a limit on the maximum frequency used as clock input to the counter. We can appreciate that the clock signal time period must be equal to or greater than the total propagation delay. The maximum clock frequency therefore corresponds to a time period that equals the total propagation delay. If t_{pd} is the propagation delay in each flip-flop, then, in a counter with N flip-flops having a modulus of less than or equal to 2^N, the maximum usable clock frequency is given by $f_{max} = 1/(N \times t_{pd})$. Often, two propagation delay times are specified in the case of flip-flops, one for LOW-to-HIGH transition (t_{pLH}) and the other for HIGH-to-LOW transition (t_{pHL}) at the output. In such a case, the larger of the two should be considered for computing the maximum clock frequency.

As an example, in the case of a ripple counter IC belonging to the low-power Schottky TTL (LSTTL) family, the propagation delay per flip-flop typically is of the order of 25 ns. This implies that a four-bit ripple counter from this logic family can not be clocked faster than 10 MHz. The upper limit on the clock frequency further decreases with increase in the number of bits to be handled by the counter.

Synchronous Counter

In a synchronous counter, also known as a parallel counter, all the flip-flops in the counter change state at the same time in synchronism with the input clock signal. The clock signal in this case is simultaneously applied to the clock inputs of all the flip-flops. The delay involved in this case is equal to the propagation delay of one flip-flop only, irrespective of the number of flip-flops used to construct the counter. In other words, the delay is independent of the size of the counter.

Modulus of a Counter

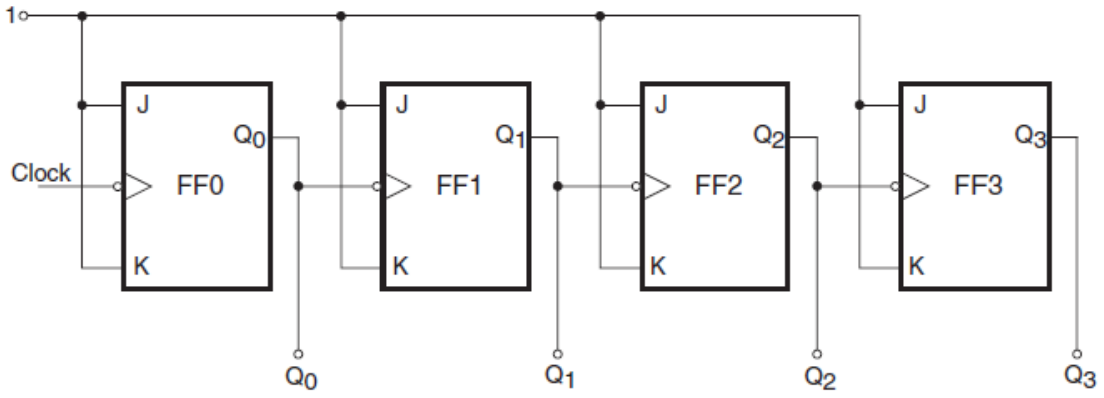
The *modulus* (MOD number) of a counter is the number of different logic states it goes through before it comes back to the initial state to repeat the count sequence. An n-bit counter that counts through all its natural states and does not skip any of the states has a modulus of 2^n. We can see that such counters have a modulus that is an integral power of 2, that is, 2, 4, 8, 16 and so on. These can be modified with the help of additional combinational logic to get a modulus of less than 2^n.

To determine the number of flip-flops required to build a counter having a given modulus, identify the smallest integer m that is either equal to or greater than the desired modulus and is also equal to an integral power of 2. For instance, if the desired modulus is 10, which is the case in a decade counter, the smallest integer greater than or equal to 10 and which is also an integral power of 2 is 16. The number of flip-flops in this case would be 4, as $16 = 2^4$. On the same lines, the number of flip-flops required to construct counters with MOD numbers of 3, 6, 14, 28 and 63 would be 2, 3, 4, 5 and 6 respectively. In general, the arrangement of a minimum number of N flip-flops can be used to construct any counter with a modulus given by the equation

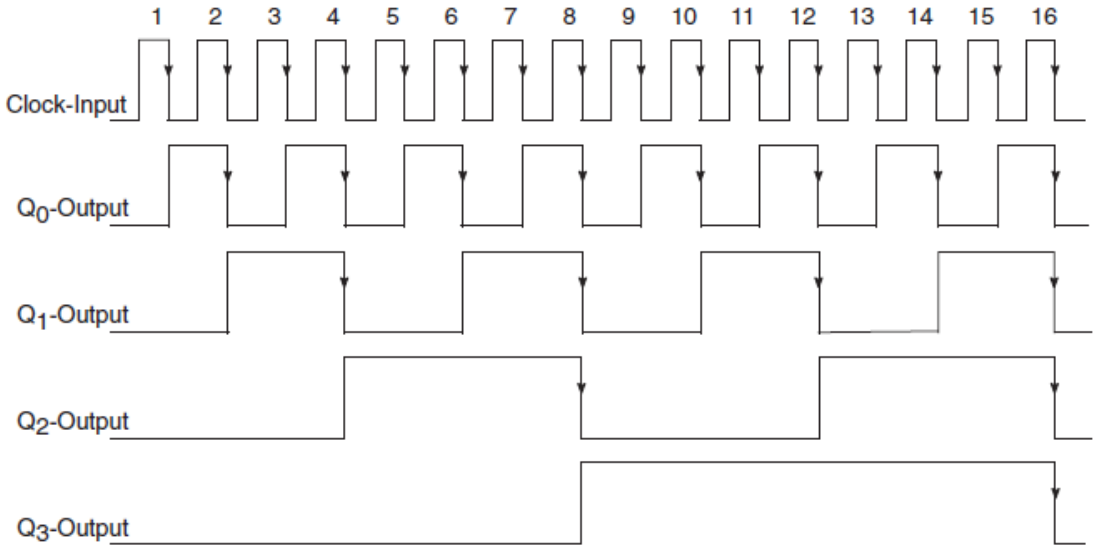
$$(2^{N-1} + 1) \leq modulus \leq 2^N$$

Binary Ripple Counter – Operational Basics

The operation of a binary ripple counter can be best explained with the help of a typical counter of this type. Figure 2 shows a four-bit ripple counter implemented with negative edge-triggered J-K flip-flops wired as toggle flip-flops. The output of the first flip-flop feeds the clock input of the second, and the output of the second flip-flop feeds the clock input of the third, the output of which in turn feeds the clock input of the fourth flip-flop. The outputs of the four flip-flops are designated as Q0 (LSB flip-flop), Q1, Q2 and Q3 (MSB flip-flop). Figure 2 shows the waveforms appearing at Q0, Q1, Q2 and Q3 outputs as the clock signal goes through successive cycles of trigger pulses. The counter functions as follows.



(a)



(b)

Figure 2 Four-bit binary ripple counter.

Ripple Counters in IC Form

In this section, we will look at the internal logic diagram of a typical binary ripple counter and see how close its architecture is to the ripple counter described in the previous section. Let us consider binary ripple counter type number 74293. It is a four-bit binary ripple counter containing four master–slavetype *J-K* flip-flops with additional gating to provide a divide-by-2 counter and a three-stage MOD-8 counter. To get the full binary sequence of 16 states, the Q output of the LSB flip-flop is connected to the B input, which is the clock input of the next higher flip-flop. The counter can be cleared to the 0000 logic state by driving both RESET inputs to the logic HIGH state.

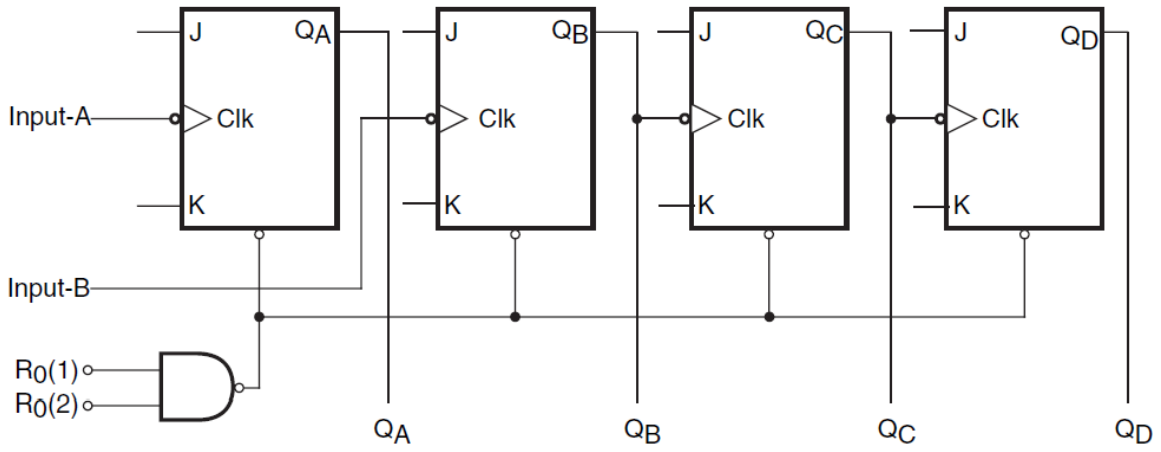


Figure 3 Logic diagram of IC 74293.