Learning Objectives:

At the end of this topic you will be able to;

☑ describe the effect of propagation delay on count rate.

From the ET5 Specification:

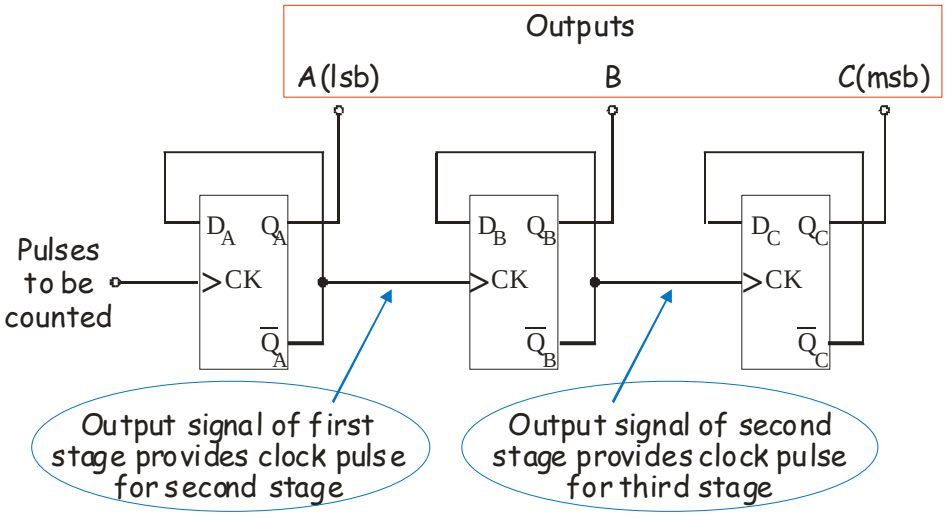
5.1 Counting Systems

This section contrasts the performance of ripple counters, covered in ET1, with synchronous counters, based on D-type flip-flops, another component of ET1. It also includes the use of synchronous counters as sequence generators.

Ripple counters

In module ET1, section 1.3.3, you connected a series of D-type flip-flops together to create 4-bit counters. In these, the output signal of the first D-type became the clock signal for the second D-type, and so on.

The diagram of a 3-bit up-counter illustrates this idea:



These counters are known as ripple counters because of the way that these signals move step-by-step through the system. This behaviour is fine, but it takes a finite time for the signals to reach the last stage of the counter.

For example:
A 12 bit binary counter has a propagation delay of 5ns per stage.
In other words, it takes 5ns for the output of the first stage of the counter to react when a new pulse arrives at its clock input.
The second stage receives its clock pulse from the output of the first stage, and then its output reacts to that, a further 5ns later.
In this way it will take 60ns (=12 x 5ns) for the last of the 12 stages to react to the new pulse.

The problem:
If another new pulse arrives within that 60ns, the first stage reacts to that new pulse, and its output changes. However, the counter had not finished adjusting to the previous pulse, and so an incorrect binary number is seen at the output. The table shows the 12 bits produced by the counter.
Remember - the first stage contains the least significant bit (lsb) of the binary number.

	msb				**Counter output**							**lsb**
bit	11	10	9	8	7	6	5	4	3	2	1	0
Initial count	0	0	1	1	1	1	1	1	1	1	1	0
1st pulse arrives at time t=0												
At t = 60ns	0	0	1	1	1	1	1	1	1	1	1	1
2nd pulse arrives at t = 200ns												
At t = 205ns	0	0	1	1	1	1	1	1	1	1	1	1→0
At t = 210ns	0	0	1	1	1	1	1	1	1	1	1→0	0
At t = 215ns	0	0	1	1	1	1	1	1	1	1→0	0	0
- - - - - - - - - - - -												
At t = 240ns	0	0	1	1	1→0	0	0	0	0	0	0	0
3rd pulse arrives at t = 240ns												
At t = 245ns	0	0	1	1→0	0	0	0	0	0	0	0	0→1
At t = 250ns	0	0	1→0	0	0	0	0	0	0	0	0→1	1

The first pulse produces a correct output of 001111111111 after 60ns.
The second pulse should produce an output of 010000000000 at time 260ns.
This never appears, because the third pulse arrives before that time, and another series of changes works it way through the stages of the counter.

The link between propagation delay and maximum reliable counting frequency:

An example:

Propagation delay = 5ns per stage
Number of stages = 12
Time for a new count to ripple through all stages = 12 x 5 = 60ns.

The output will show a false count if a second pulse arrives within that time.

Hence **minimum** period for pulses arriving at the counter = 60ns.
Hence **maximum** pulse frequency = 1 / 60ns
= 16.7MHz

In practice it would be wise to limit the maximum frequency to well below this theoretical maximum.

Examination Style Questions.

1. Each stage of an 8-bit ripple counter has a propagation delay of 80ns.
Show how this limits the maximum frequency of counting to around 1 MHz. [2]

……

……

……