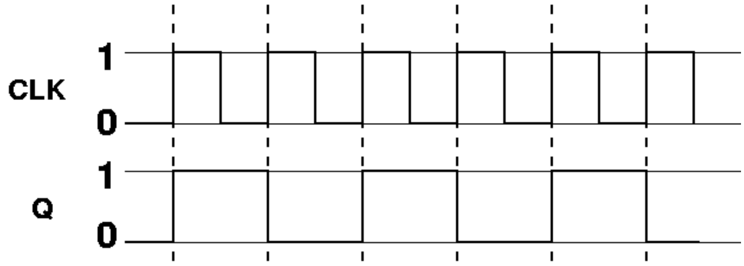
Asynchronous Counters

Introduction

A counter usually counts from 0 to N. Some counters allow you to hold the value or to increment one at a time. We're going to build a very simple counter using T flip flops. This counter is called asynchronous because not all flip flops are hooked to the same clock.

T flip flop whose control input is 1

What happens to the output of a T flip flop when control input is 1?
We can observe the behavior of the output of a T flip flop by looking at a timing diagram.



The output, Q, resembles a clock as well. If the period of the clock is T, then the period of Q (the output of the flip) is 2T.

Thus, we have a very easy way to create a clock that runs twice as slow. We feed the clock into a T flip flop, where T is hardwired to 1. The output will be a clock whose period is twice as long.

So, imagine we feed the output of this T flip flop, whose period is 2T, as the clock of another T flip flop, which also has its T input hardwired to 1. What would the period of that clock be?

It creates a clock that has twice the period. Since we fed in a clock with period 2T (the output of the first T flip flop), the output of the second flip flop has period 4T.

If we keep feeding the output of one T flip flop, into the clock input of another T flip flop hardwired to 1, and we do this for N flip flops, then what is the period of the output of the last flip flop? It's 2NT. Each flip flop doubles the period, so N flip flops is 2 raised to the Nth power.

Counting in Binary

Row	x2	x1	x0
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

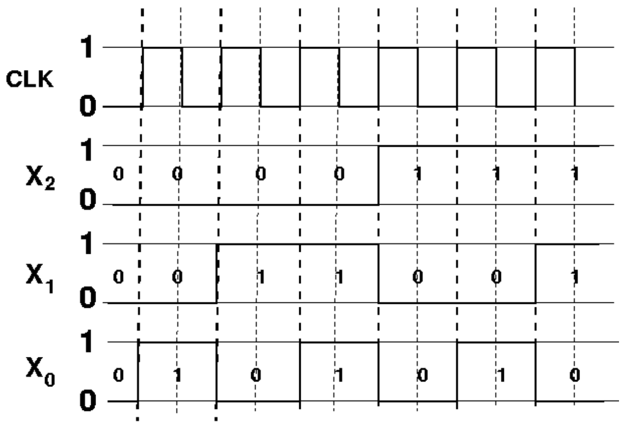
Look at the column labelled x0. It reads 0, 1, 0, 1, 0, 1, etc. It looks like a clock. Assume that this clock has period T.

Now look at the column labelled x1. It reads 0, 0, 1, 1, 0, 0, 1, 1. It looks like a clock too! However, it stays 0 twice as long, then 1 twice as long. In fact, it looks like a clock that has a period of 2T.

Now look at the column labelled x2. It reads 0, 0, 0, 0, 1, 1, 1, 1. Again, this looks like a clock, except it's going twice as slow. It has a period of 4T!

They look like the chained T flip flops we had above.

This is the timing diagram that shows how the counter behaves.



What's going on in this timing diagram? First look at the row that says CLK. That's the clock.

Then look at row X0. This toggles between 0 and 1 on the positive edge of the clock. That's because the clock is fed into the bottommost T flip flop.

X1 toggles according to X0. That's because we feed X0' into the clock of the middle T flip flop.

X2 toggles according to X1. That's because we feed X1' into the clock of the top T flip flop.

Start reading the timing diagram from the left most column, you will see 000, then 001, then 010, then 011, and so forth. As you can see the output of the flip flops is incrementing as it should. The counter increments at a period of 2T, assuming the clock has a period of T.

Summary

Creating an asynchronous counter from T flip flops relies on two observations. First, if you hardwire a 1 into a T flip flop, the output of the T flip flop (i.e., Q) toggles at twice the period.
Second, if x0 acts like a clock of period T, then xi acts like a clock of period 2iT. That is, each success column to the left is a clock that doubles the period. Thus, we can combine these two facts together to generate a counter. Notice that the counter must increment based on negative edges. Thus, Xi+1 toggles on Xi. This is accomplished by feeding the negative output of a T flip flop (i.e. Q') to the clock of the next T flip flop.

This counter is considered asynchronous, since each flip flop runs on its own