

Lab 5 Lecture - Introduction to Counters and CPLDs

Outline of Topics:

1. Review -- The Basics of CPLDs.
2. 1-2-3 -- The Concept of Counters.
3. One For You, Two For Me -- Ways to Generate A Sequence.

Part 1: Review -- The Basics of CPLDs.

In the previous lab, you were introduced to CPLDs, (Complex Programmable Logic Device). These belong to a class of ICs called programmable logic devices (PLDs). This class of devices includes small-scale programmable logic such as PALs (Programmable Array Logic) which takes the place of around 20 logic gates, larger scale devices, such as the CPLD, which can represent designs consisting of a few hundred to a couple thousand gates, to PLDs of truly enormous scale, such as the newest Virtex II devices put out by Xilinx, which can represent system designs containing eight to ten million logic gates.

Recall that CPLDs are non-volatile devices. That means that whatever was programmed onto the device persists when the device loses power. This is why we asked you to reprogram the XCR boards with the original test program after you finished your work on Lab 3.

Part 2: 1-2-3 -- The Concept of Counters.

A counter is one of the more useful digital circuits you will encounter. Counters are used in many applications. Examples include frequency dividers, and frequency counters. A quick example of a digital counter would be the circuit that drives your stopwatch. This circuit takes at least two inputs, a clock signal from the quartz oscillator, and an input from a button you press. It starts counting when you press the button once, and stops when you press the button again.

Before we go on, note that a digital counter has the following characteristics:

1. A maximum number of counts before it rolls over (returns to zero.) This is referred to as the counter's modulus.
2. It can count in either direction (ascending, from low to high, or descending, from high to low.)
3. It is either synchronous or asynchronous. That is, it counts with the system clock, or it counts independently of the system clock.
4. It can function either as a monostable or an astable circuit. The former means that it runs once and stops. The latter means that it will run forever, or until interrupted.

Traditionally, a counter circuit was constructed with flip-flops. This was because a counter is a circuit which requires a memory, that is, it has to have some way of tracking where it is in the sequence. As a result, a four-bit counter would be constructed with four flip-flops, and some control logic. One could conceivably replace the four flip-flops with a single four-bit latch, which is a type of register.

Part 3: One for You, Two for Me -- Ways to Generate a Sequence.

In this section, we will briefly discuss ways of generating a count. The first one involves an adder, a register, and some control logic. The output of the counter is simply some fixed number added to the previous input. This is known as a ripple counter.

Example -- A four-bit ripple counter.

The following is an example of a four-bit ripple counter. This is a circuit which will count from 0 to 15, then stop. It is constructed with a four-bit register, and an adder. The output of the register is delayed and fed into one of the inputs of the adder. This makes it a Mealy machine rather than a Moore machine. This is because its output is a function of its current inputs, and its history.

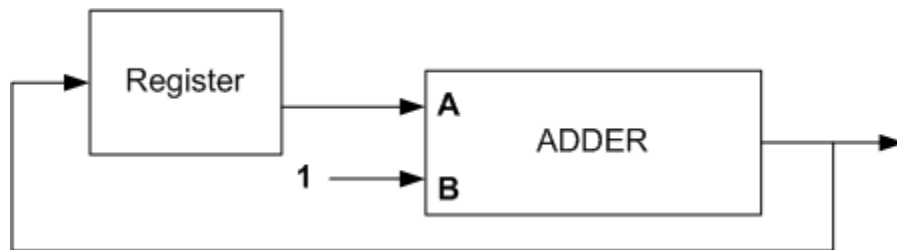


Figure 1: Block diagram of a 4-bit Ripple Counter.

In the previous figure, we have omitted the control logic in order to save space. This counter is called a modulus-16 counter, because its four bits allow us to represent sixteen different things (2^4).

Another type of counter is a grey-coded counter. You may recall grey-coding from when you were first introduced to K-Maps. Remember that a grey-code is one where only one bit changes at a time. The following tables show the difference between a three-bit modulus-8 ripple counter, and a three-bit grey-coded counter.

Ripple	Grey-Coded
000	000
001	001
010	011

011	010
100	110
101	100
110	101
111	111

Note that in the grey-coded sequence, only one bit changes at a time. Yet, we have still represented all the possible bit combinations. Developing a grey-coded counter requires a little more work than a simple ripple-counter. For the grey-coded counter, use a finite state-machine (FSM) to do the work. You may remember the FSM from the sequence detector you developed in Lab 4.