



State Machines

ENEE 245: Digital Circuits and Systems Laboratory Lab 3

Objectives

The objectives of this laboratory are the following:

- To design a simple state machine
- To learn how to test the state machine on various inputs

State machines are the fundamental underpinnings of computers and computer systems. In this lab, you will build a simple 2-bit saturating counter and test it manually.

Saturating Counters

A saturating counter is one that does not “wrap around” when it reaches the maximum or minimum value. In normal 2’s complement arithmetic, when a binary number reaches the maximum possible value and you add 1 to it, you get the largest negative number as a result. Similarly, when you subtract 1 from the minimum number you get the largest positive number as a result. This can be seen in a simple 3-bit example:

011	3
010	2
001	1
000	0
111	-1
110	-2
101	-3
100	-4

If you add 1 to the binary value “011” you get “100” which, in a 4-bit representation or larger, would equal the decimal number 4. However, in a 3-bit number, this is the value -4. Similarly, if you subtract 1 from the binary value “100” you get “011.” This is normal behavior in 2’s complement.

Saturating counters are those that do not wrap around. If you add 1 to the largest positive number in a saturating counter, the counter stays pegged at that number. If you subtract 1 from the largest negative number, the counter stays pegged there as well.

Digital Logic Analyzer

The Intronix Logic Port has considerable digital signal measurement capabilities and we will learn a few of its features in this course. A picture of the logic port hardware is shown in Fig. 1. We will use the name Digital Logic Analyzer (DLA) to refer to the Logic Port.

The main difference between the acquisitions of digital signals as compared to analog signals is that for the digital work there are only two possible output states: logical zero and logical one. Thus, a digital logic analyzer only needs to use 1 bit per time step per channel whereas analog signals need multiple bits/time/channel. Each channel samples the signal on an input channel and sets the result to zero if the voltage is less than some threshold and sets it to one otherwise. This threshold can be adjusted. This approach means that you will never learn anything about rise and fall times or about any absolute voltage in a digital circuit.

One advantage of a DLA is that it can look at many signals in a circuit simultaneously. While an oscilloscope can typically look at 2-4 channels, our DLA can look at 34 channels. For the same

amount of memory, DLAs can store at least 8 times as many points, so they usually have good “glitch-capture” capabilities, i.e. the ability to see transitions that happen much more rapidly than the nominal “frequency” of a clocked circuit. Mixed-Signal Oscilloscopes (MSOs) have both digital and analog channels, and are nice for digital circuits, since the analog channels can be used to obtain rise and fall time data as well as absolute voltage levels on digital signals of interest.

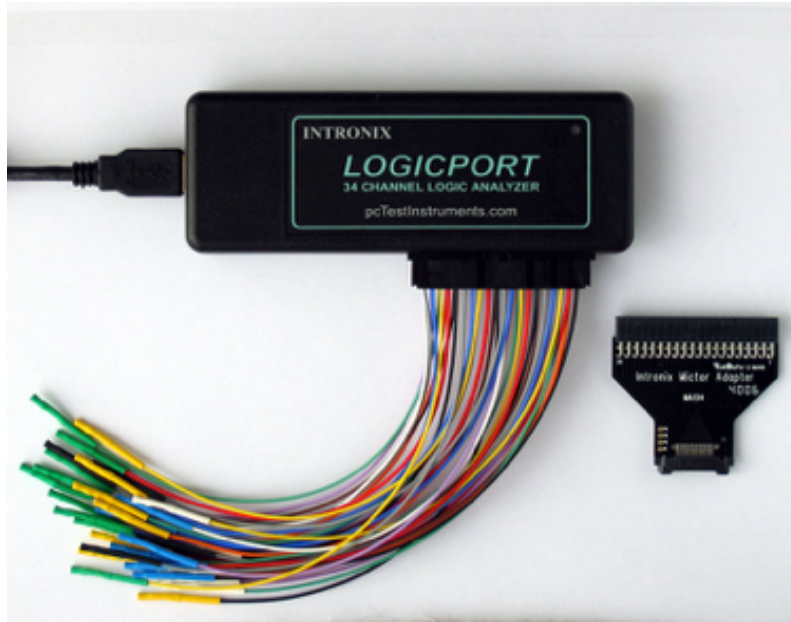


Figure 1 Intronix Logic Port

The trigger for the DLA is set in the Logic Port software. In DLA, the trigger method can be much more sophisticated than that of an oscilloscope. Like the oscilloscope, the DLA is continuously sampling data at some specified rate. When the trigger signal is identified, that time is placed at the center of the trace and the outputs before and after the trigger time are displayed on the screen. There are many ways to set the triggers. For example, you can look for a certain transition to occur or you can look for a particular combination of states to occur in certain channels. For example, let's say you had a circuit that contained a 4-bit binary counter along with some other logic elements. Let's say you sampled 9 signals altogether, and the counter signals were on the lowest channels 0-3. If you wanted to trigger the circuit when the counter output was nine, you would specify the trigger as XXXXX1001. The “X's” represent don't cares, (and there would be up to 12 of them, depending on how many digital channels were active). This technique is known as “pattern triggering.”

You can have two “levels” of triggers (this is called sequence triggering). For example, maybe you want to trigger when the output is 10XX10101, but only if sometime prior to that the output was 00000XXXX. The former sequence would be entered as the “find event” (P1) and the latter sequence would be entered as the trigger event (P2). When the first level combination is sensed, the second sequence is searched for. When the complete trigger sequence is found, the selected signals are displayed on the CRT. The middle of the display corresponds to the time that the trigger occurred, so the left side of the screen displays the signals that existed before the trigger and the right side shows the evolution of the signals after the trigger. A reset event can also be specified, and if this event occurs after the pre-trigger event was detected but before the trigger event happened, the trigger is reset to its initial state and the DLA waits for the pre-trigger signal again.

Standard edge-triggering can also be used, and for pattern triggering a single transition (up or down) for one of the signals can be specified. Edge triggering can be specified as the type of events searched for during sequence triggering.

Pre-Lab Preparation

Design a 2-bit saturating counter using any of the following components:

- a 4-bit clocked register
- a 2-bit adder
- multiplexers (2-to-1 and/or 4-to-1)
- individual NAND gates, NOR gates, and/or inverters

You do not have to design the internals of the components, as you have already done that in the previous two labs. Your counter should increment when it receives a “1” as input, decrement when it receives a “0” as input, and hold at maximum & minimum values of “01” (decimal 1) and “10” (decimal -2).

Draw the logic diagram as well as the wiring diagram using the available CMOS chips.

Use SPICE to simulate your design, using a clock input (alternating 0/1 values) at all four starting points of the counter, with both possible starting points of the clock. In other words, run eight different SPICE simulations. Most importantly are the “01” counter starting point with a “1” input value from the clock to begin with and the “10” counter starting point with a “0” input value from the clock to begin with. Hopefully the reason why these are the most important test cases is clear.

Think about how you are going to use the DLA to do the same sort of testing as SPICE simulation.

In-Lab Procedure

Bring flash drives to store your traces.

Ask the TA questions regarding any procedures about which you are uncertain.

Turn off all power supplies any time that you make any change to the circuit.

Do NOT apply more than 5 V to the circuit at any time.

Arrange your circuit components neatly and in a logical order.

Compare your breadboard carefully with your circuit diagram before applying power to the circuit.

Complete the following tasks:

- Construct your 2-bit saturating counter.
- Drive your counter with a 1-second clock, a 7-segment display, and a manual “1/0” add/sub control input.
- Manually test your counter to reach all possible combinations of counter values and input values.
- Record the full test through the logic analyzer to show that all possible combinations of state and input were tested.

Post-Lab Report

Write up your circuit, schematic, and lab procedure. Demonstrate the correctness of your design using the test results that you collected in the lab.