# Lab 14: Sequential Logic

U.C. Davis Physics 116B (Rev. 2/7/03)

## INTRODUCTION

In this lab, you will build some examples of sequential logic circuits. Recall that sequential logic contains some form of memory. These circuits change state when they are given a clock pulse.

## 1. DATA REGISTER

A data register is a rather trivial application of the D flip flop. Build the 4 bit data register shown in figure 1 using two 74LS74 D flip flops.

*Note: be sure to connect the "active low" preset and clear inputs to +5 V. You could get incorrect transitions otherwise. If problems persist, try connecting a 0.01 μF capacitor between $V_{CC}$ and ground at the chip to reduce glitches on the supply line during transitions.*

For your lab report, briefly discuss what it would be used for. Save this circuit for the next section.

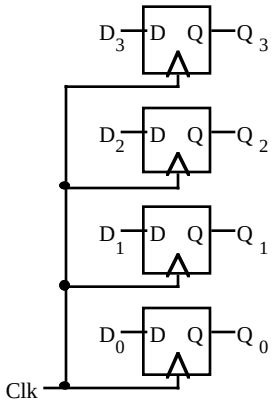


Figure 1: Data register.

## 2. SHIFT REGISTER

Use the same 4 flip flops to construct the 4 bit shift register shown in figure 2. For your lab report, draw a timing diagram demonstrating how you would serially load a 4 bit data word into this 4 bit register. Save this circuit for the next section.

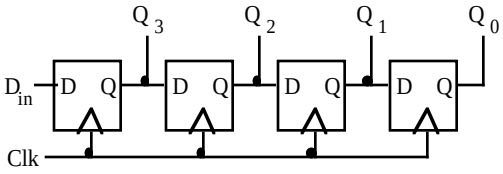


Figure 2: Shift register.

*Optional but fun*: Use extra LEDs (with series resistors) to monitor the outputs. Try using 8, 12, or 16 LEDs by having each output go to 2, 3, or 4 of the LEDs. In this way, you can make a "walking lights" display like the ones in casinos or game shows.

## 3. TWISTED RING COUNTER

If you connected $Q_0$ to $D_{in}$ in the shift register above, the 4 data bits in the shift register would continually be shifted right from $Q_3$ to $Q_0$ and then around to $Q_3$ again. This is called a "ring counter." Note that the counter does not count in the conventional "weignted binary" sequence. If $Q_0$ is inverted before it is fed into $D_{in}$, then it is a "twisted ring counter". This circuit is also called a Johnson counter. Make a twisted ring counter by adding an XOR gate to your circuit above:

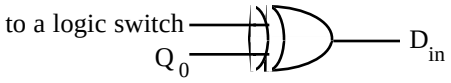


Figure 3: Gating for twisted ring counter..

For your lab report, describe how you can get any pattern of bits into the 4 flip flops and draw a timing diagram illustrating what happens as the clock is pulsed. (Hint on doing the timing diagram: After how many clock cycles will the pattern repeat?)

## 4. PSEUDO RANDOM BIT SEQUENCE GENERATOR

A simple change to the circuit above will give it vastly different behavior. Instead of the above connections to the XOR gate, use these:

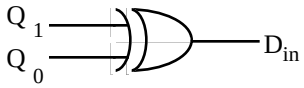


Figure 4: Gating for pseudo random bit sequencer.

Make sure that at least one output bit is non zero. Then, look at the resulting pattern of lights as you clock the circuit. Does it look random? It probably does, but in fact the pattern repeats after 15 clock cycles. Number patterns that look random but actually aren't are called "pseudo random" (false random). For your lab report, draw the timing diagram for this circuit for at least 16 clock cycles. Show how the $Q_0$ values are the XOR of the $Q_2$ and $Q_3$ values from the previous clock cycle. Indicate where the pattern starts to repeat.

## 5. ASYNCHRONOUS (RIPPLE) COUNTER

Now construct the circuit shown in figure 5. This is the first true binary counter in this lab, even though a previous circuit had the name "counter" in it. For your lab report, do a timing diagram for this circuit and on this timing diagram, interpret the outputs as a 4 bit binary number ($Q_0$ is the least significant bit). Does this circuit count? What is its range?

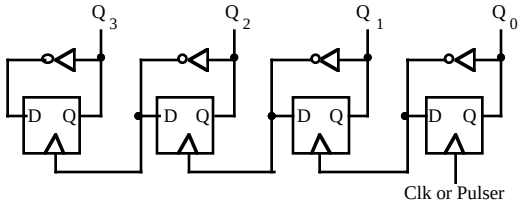


Figure 5: Asynchronous binary counter.

Drive this circuit with a fast clock and look at the output with the oscilloscope. Measure the propagation delay from the time the clock pulse rises to the time $Q_3$ changes state. Also measure the delay to $Q_2$ which should be shorter, and to $Q_1$ which should be shorter still and to $Q_0$ which should be the shortest. Can you see why these delay times are different? On your lab report, give these times and explain why they are different.

## 6. SYNCHRONOUS COUNTER

### *NOTE: OPTIONAL PART (NOT TO BE DONE UNLESS EXPLICITLY ASSIGNED)*

The propagation delay you measured above for an asynchronous (not simultaneously clocked) counter can severely limit the speed at which a counter can count. To speed things up, all the flip flops can be clocked at the same time. Combinatorial logic can determine whether a given flip flop should flip or not.

*This part of the lab can be done ahead of time:* Draw a state diagram for a 3 bit binary counter. (Optionally, do it for a 4 bit counter.) From this, determine the logic needed at the $D_2$, $D_1$, and $D_0$ inputs. That is, determine the parts of the circuit of figure 6 in the boxes with question marks.

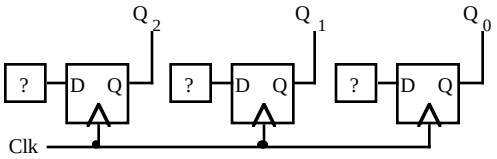


Figure 6: Synchronous counter (logic TBD).

In the lab, construct this circuit and see if your input logic is correct. Does your counter count? Compare the propagation delay of this circuit (Clk to $Q_2$) to the corresponding propagation delay for the asynchronous counter. Does your synchronous counter have a shorter propagation delay? (It should if everything worked.) That means the synchronous counter could count at a higher frequency than the asynchronous one above, and by most peoples' standards, that means the synchronous one is better. For your report, include your state diagram, the logic circuits you built, a timing diagram for the counter, and the measured propagation delay compared to the asynchronous counter's.

## 7. *BIG* PSEUDO RANDOM BIT SEQUENCE GENERATOR

### (DEMO IF TIME PERMITS)

The 74LS164 is an 8 bit shift register on a single chip. Here, 2 of these chips are cascaded to get a 16 bit shift register. From a data table in a book, we see that hooking an XOR gate to outputs $Q_{13}$ and $Q_{14}$ will give a pseudo random sequence 32,768 clock cycles long. By hooking one of the output bits to a speaker (using an RTL inverter, essentially), we can hear this random sequence, or "noise". For your lab report, describe what was most interesting to you about this circuit.