4. Digital Circuits

Learning Objectives

Build and test ripple counters using flip-flops and pseudorandom number generator using shift registers and XOR gate.

Assignments Before the Laboratory Session

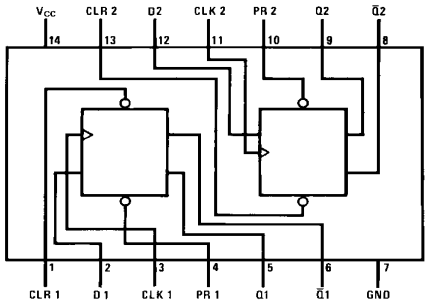
Search the web for "flip-flop" and "ripple counter."

Search the web for "shift register" and "pseudorandom number generator."

There is a large selection of circuits to perform digital tasks. Simple circuits use few ICs while complicated circuits may require a field programmable gate array or a custom VLSI to implement. A flip-flop is a bistable device. The output responds to the input by switching between two stable states, high and low. Since it stays in the same state until new input arrives, it serves as a storage device, i.e., a single-bit memory. There are a few variations, e.g., RS, D, JK, and T flip-flops.

The following ICs are needed: one 74LS00, two 74LS74, and one 74LS86. Put together the divide-by-16 counter on the prototype circuit board by cascading two divide-by-4 counters shown in Fig. 3 before attending the laboratory session.

Connection Diagram



Function Table

Inputs				Outputs	
PR	CLR	CLK	D	Q	$\overline{Q}$
L	H	X	X	H	L
H	L	X	X	L	H
L	L	X	X	H (Note 1)	H (Note 1)
H	H	↑	H	H	L
H	H	↑	L	L	H
H	H	L	X	Q_0	$\overline{Q}_0$

H = HIGH Logic Level
X = Either LOW or HIGH Logic Level
L = LOW Logic Level
↑ = Positive-going Transition
Q_0 = The output logic level of Q before the indicated input conditions were established.

Note 1: This configuration is nonstable; that is, it will not persist when either the preset and/or clear inputs return to their inactive (HIGH) level.

Fig. 1. 74LS74

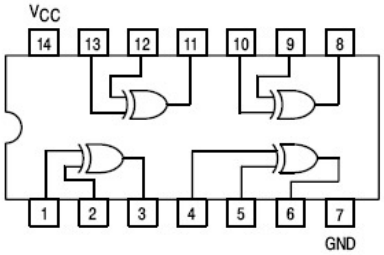


Fig. 2. 74LS86

Activities During the Laboratory Session

Verify the integrity of the oscilloscope probe by connecting it to the internal square wave source. Verify the integrity of the multimeter test lead by measuring its resistance.

D-type flip-flops can be used to form binary dividers or counters as shown in Fig. 3. There are two stages. Each stage performs division by two. By cascading N stages, one can divide by 2^N. Each stage has one clock input and one data input. It has two complementary output nodes. In addition, there are preset node and clear node to control the operation of the IC. The first stage is driven by a clock signal generated by the function generator by setting the amplitude to 2.5 V and offset to 1.25V. The actual voltages are twice the displayed values. The second stage is driven by the output of the first stage. Each 74LS74 chip has two flip-flops. By cascading two chips, one can get a divide-by-16 counter. Since there is a propagation delay from stage to stage, there is a delay between the time when the state of the first stage changes and the time when the last stage changes. Such a counter is asynchronous and called ripple counter. By using logic gates, you can prematurely terminate the counting cycle and form counters for division by a number less than 16, e.g., divide-by-10, divide-by-12, etc.

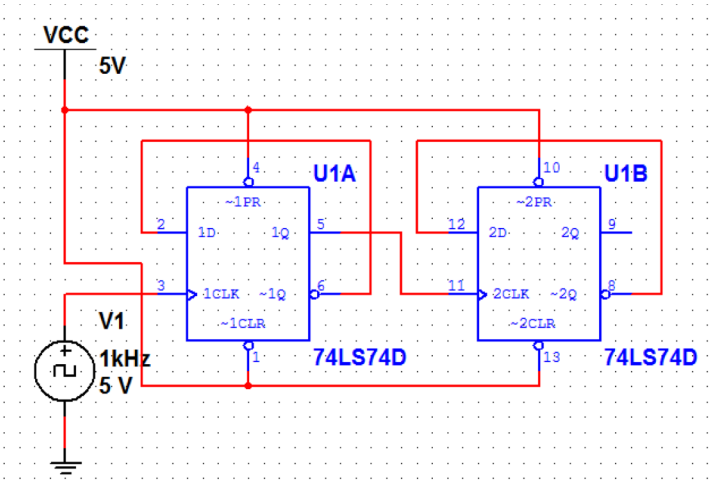


Fig. 3. A divide-by-four counter based on D flip-flops. The input is a digital clock signal generated by a function generator. The input and output can be monitored simultaneously by a dual trace oscilloscope. Or, one can connect an LED with a series resistor of 330 Ω to the output of each flip-flop as an indicator when the clock is at very low frequency.

Cascade two 74LS74 chips to form a divide-by-16 counter. Connect the power supply, 5 V, to pin 14 and ground to pin 7 first. Use the adjustable power supply so that you can set the current limit to 0.18 A. Tie preset nodes, i.e., pins 4 and 10, of both chips together as one node. Then connect this common node to logic H, i.e., 5 V. Don't connect pin 4 and pin 10

individually to logic H. You will need this common node later when you build the decade counter. Likewise do the same to clear nodes, i.e., pin 1 and pin 13. Monitor the input clock signal and the output signal of each binary divider; from 2 to 16. Select the output signal of the divider as the trigger source. Record waveforms using the clock signal as a reference, i.e., as channel 1 of the oscilloscope. Can you determine the delay between the clock signal and the output of the divide-by-16 counter? Change the time base of the oscilloscope to finer scale. Increase the clock frequency if needed. The delay exists in ripple counters but not in synchronous counters in which all switching events take place simultaneously according to the timing of one master clock signal.

Starting from the least significant bit, which is closest to the clock input, each stage carries a weight of 2^0, 2^1, 2^2, 2^3. By monitoring states of 2^1 and 2^3 stages, one can easily form a divide-by-10, i.e., decade counter, as shown in Fig. 4. The common preset node is redirected to the output of the NAND gate. Record the waveform at the output. Increase the frequency of the clock signal to see whether the counter functions properly at the highest frequency of the function generator.

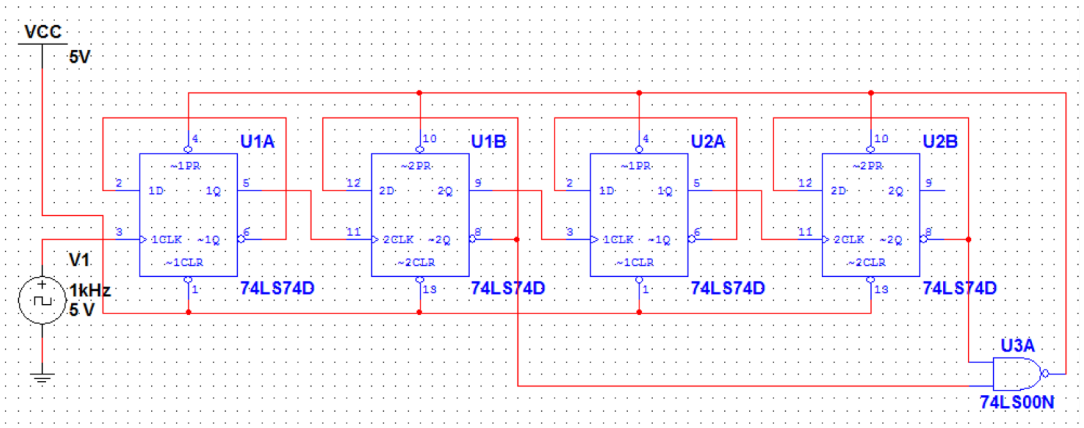


Fig. 4. A divide-by-ten, decade counter.

The circuit shown in Fig. 4 is a binary coded decimal (BCD) counter. The state of the counter is determined by $S_1*2^0+S_2*2^1+S_3*2^2+S_4*2^3$, where S_i is the state of the i-th stage. The state of each stage can be either zero or one. By monitoring states of S_2 and S_4, one can catch the arrival of 2^1+2^3, i.e., the 10^{th} clock pulse. At that moment, the NAND gate resets counters to the state of 0000. The signal at the last stage Q-node consists of 80% high and 20% low. Such a waveform has a duty cycle, i.e., percentage of time in the high state, of 80%. Of course, if you monitor the output at the conjugate node, it is 20% duty cycle. Instead of using the circuit diagram shown in Fig. 4, one can build another decade counter by cascading a divide-by-5 counter followed by a binary, i.e., divide-by-2 counter. With a binary counter as the last stage, the duty cycle will be 50%.

In addition to counters, one can also form shift registers with cascaded D flip-flops. Furthermore, by adding XOR gate, one can form a pseudorandom number generator. One example is shown in Fig. 5. The sequence generated depends on how the XOR gate is configured. In the configuration shown in Fig. 5, output nodes of the 3rd and the 4th D flip-flops are used to drive the XOR gate. You can change the configuration by using another pair of output nodes to drive the XOR gate. To start the pseudorandom number generator, connect the preset node of the 1st flip-flop to logic L, i.e., ground first. Then, connect it to logic H, i.e., 5-V power bus line. Record the sequence generated by displaying the clock signal and the output waveform on the oscilloscope. Change how XOR gate is connected. Record another pseudorandom sequence. Change how XOR gate is connected again. Record the waveform of the third pseudorandom sequence. Altogether, record three pseudorandom sequences along with circuit diagrams showing how XOR gate is connected. Determine the length of the pseudorandom sequence in term of the number of input clock cycles. Where to place XOR gates to get the maximum length can be found by web search, e.g., Wikipedia.

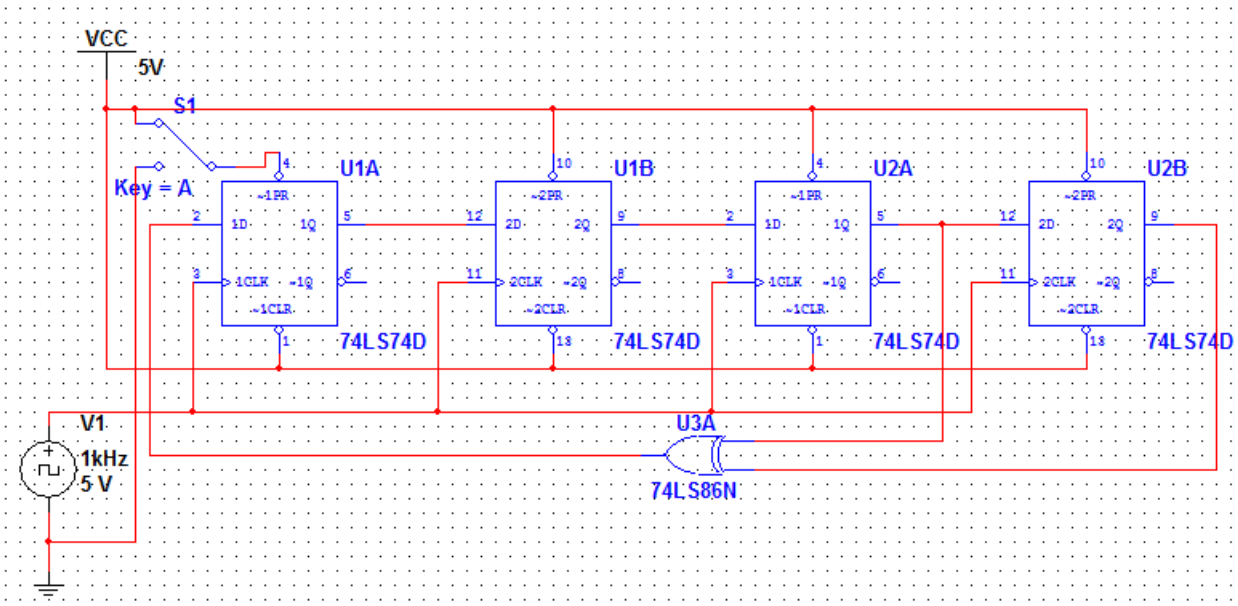


Fig. 5. A pseudorandom number generator circuit.

Activities After the Laboratory Session

Search the web and study the advantage of synchronous counters. Which chip is used in forming the synchronous counter? What does it mean by down counter or up counter? What are applications of counters? Discuss findings in your own words. Don't copy and paste information on the web. Provide references.

Prepare and submit a report using the IEEE manuscript template. Include the scoring sheet. Check the scoring sheet so that you won't leave any data out.

Self Study

1. What does flip-flop do? Can you name few circuits based on flip-flops?
2. Can you determine the waveform of a divider circuit from the truth table of chip used?
3. Can you design a divider which divides by 12? How about divide by 60?
4. What are the applications of pseudorandom number generators?

Cover and Score Sheet

Experiment 4 - Digital Circuits

Author:_____ Partner:_____

Score

Item	Credit	Score
Data	8	
Binary Counter		
Waveforms Showing Divide by 2		
Waveforms Showing Divide by 4		
Waveforms Showing Divide by 8		
Waveforms Showing Divide by 16		
Decade Counters		
Waveforms Showing Divide by 10		
Frequency Limit		
Pseudorandom Number Generators		
Sequence 1 With Circuit Diagram and Waveforms		
Sequence 2 With Circuit Diagram and Waveforms		
Sequence 3 With Circuit Diagram and Waveforms		
Discussion of Synchronous Counter		
Format and Contents	4	
Total	**12**	

TA Signature:_____ Date:_____