CDA 3201L Thursday Section
Lab number 05 -- Sequential Logic Circuits I
06/21/14 , John Gangemi & Kyle Peck

PURPOSE AND OBJECTIVES

Lab number 5 is a basic introduction to sequential logic and the operation of essential circuits referred to as “Flip-flops”, where the next state for the circuit depends on the current state of inputs and outputs.

Part A of the lab demonstrates the functionality of a simple JK flip flop using the TTL IC 74LS76 and a function generator as the clock source. The state of the outputs Q and Q’ are to be observed based on the current inputs and clock signal form.

Part B of the lab demonstrates the functionality of a simple D flip-flop using the TTL IC 74LS74 and a function generator as the clock source. The state of the outputs Q and Q’ are to be observed based on the current input and clock signal form.

Part C of the lab requires that a JK flip-flop be configured to function properly as a D flip-flop provided that any other logic components may be used including a function generator as the clock source.

Part D of the lab requires that a JK flip-flop be configured to function properly as a Toggle flip-flop provided that any other logic components may be used including a function generator as the clock source.

Part E of the lab requires that a SR flip-flop is implemented using logic gates and the state of the outputs Q and Q’ are observed based on the current inputs and clock signal form, verifying the correctness with a truth table for the SR flip-flop. Next, modify the design for the SR flip-flop by adding additional circuit elements to the input lines. The functionality of the modified SR flip-flop is shown in the following truth table:

Inputs to Circuit		**SR FF Outputs**	
A	**B**	**Q**	**Q'**
0	0	Previous Q	Previous Q'
0	1	1	0
1	0	0	1
1	1	1	0

COMPONENTS USED

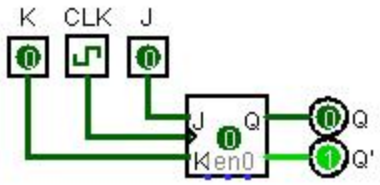
- Integrated Circuits For Demonstration
    - . 74LS04 HEX Inverter
    - . 74LS02 Quad 2-Input NOR
    - . 74LS76 Dual JK Flip-Flop with Preset and Clear
    - . 74LS74 Dual D Flip-Flop with Preset and Clear
- 10 x 5mm Red LEDs
- 10 x 470 ohm Resistors
- 5 Volt Regulated DC Power Supply
- Assortment of 22 AWG Jumper Wires
- 2250 point Breadboard
- Logisim Program version 2.7.1

DESIGN DESCRIPTION

Part A

To implement a basic JK flip-flop using the TTL IC 74LS76 the inputs to the chip's pins must be configured properly. Referencing the datasheet for the 74LS76 shows that the inputs for J, K, and CLK are not considered when either or both of the inputs for the clear and preset pins are set low (GND). Then for the purposes of the verification of a JK flip-flop the clear and preset pins are set high (+5Vcc) and the clock source is provided by a function generator set to a low oscillation of 10 cycles per second (10 hz).

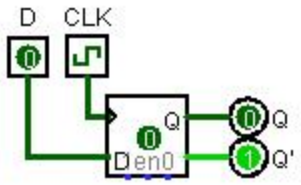
The logic circuit for a JK flip-flop:



Part B

To implement a basic D flip-flop using the TTL IC 74LS74 the inputs to the chip's pins must be configured properly. Referencing the datasheet for the 74LS74 shows that the inputs for D and CLK are not considered when either or both of the inputs for clear and preset pins are set to low (GND). Then for the purposes of the verification of a D flip-flop the clear and preset pins are set high (+5Vcc) and the clock source is provided by a function generator set to a low oscillation of 10 cycles per second (10 hz).

The logic circuit for a D flip-flop:



Part C

For a JK flip-flop to function properly as a D flip-flop the JK flip-flop should switch only between "set" and "reset" modes and ignore the behavior of the "latch" and "toggle" modes. Additionally the JK flip-flop should only accept a single input to both the J and K input lines of the JK flip-flop.

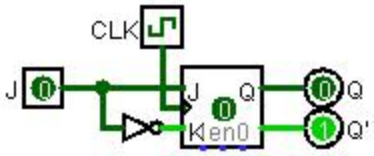
A D flip-flop truth table:

D	CLK	Q	Q'
x	low	Previous Q	Previous Q'
0	hi	0	1
1	hi	1	0

In order to switch between "set" and "reset" modes of the JK flip-flop the inputs for J and K must always be opposite of each other. Input J can be modelled after the D input of the truth table shown above and justifiably input K will be the negation of input J.

The modified JK flip-flop truth table and logic diagram:

J	K	CLK	Q	Q'
x	x	low	Previous Q	Previous Q'
0	1	hi	0	1
1	0	hi	1	0



Part D

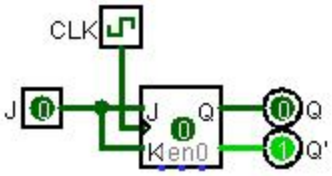
For a JK flip flop to function properly as a Toggle flip-flop the JK flip-flop should change output state when in “toggle” mode and ignore the “set” and “reset”modes. Additionally the JK flip-flop should accept single input for both the J and K input lines.

A Toggle flip-flop truth table:

T	CLK	Q	Q'
0	low/hi	Previous Q	Previous Q'
1	low	Previous Q	Previous Q'
1	hi	Next Q	Next Q'

To get the function of the “toggle” mode for the JK flip-flop both J and K inputs must be high(+5Vcc) and the clock signal needs to be a leading-edge to trigger the state change between the outputs. Input J and K can be modelled after the T input for the Toggle flip-flop truth table shown above. Accordingly the single input for the JK flip-flop will drive both inputs J and K.

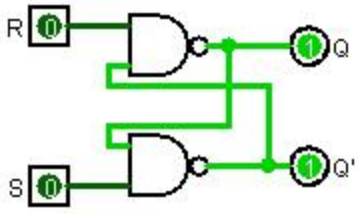
A logic diagram of the modified JK flip-flop:



Part E

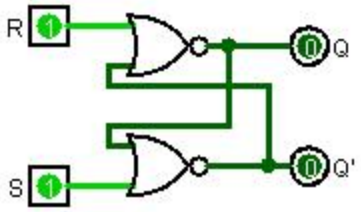
The SR flip-flop is an asynchronous circuit in that it does not require a clock source to trigger state changes in the outputs. Inherit to the design of the SR flip-flop state changes are triggered by the previous state of Q and Q’ dependent upon the current state of the inputs S and R. An SR flip-flop can be designed using NAND or NOR gates however there exists an invalid state for the outputs of either implementation.

An SR flip-flop using NAND gates and its truth table: (invalid state is shown)



S	R	Q	Q'
0	0	Invalid	Invalid
0	1	0	1
1	0	1	0
1	1	Previous Q	Previous Q'

An SR flip-flop using NOR gates and its truth table: (invalid state is shown)



S	R	Q	Q'
0	0	Previous Q	Previous Q'
0	1	0	1
1	0	1	0
1	1	Invalid	Invalid

To achieve the functionality shown in the truth table for part E of the lab assignment it is first crucial to understand which implementation of gates to use for the SR flip-flop. Reviewing the output states in correlation with their inputs it can be concluded that an SR flip-flop designed with NOR gates will give the correct state for S=0 and R=0 (this is an invalid state for the NAND SR flip-flop).

Analyzing the truth table further provides the following two Boolean expressions:

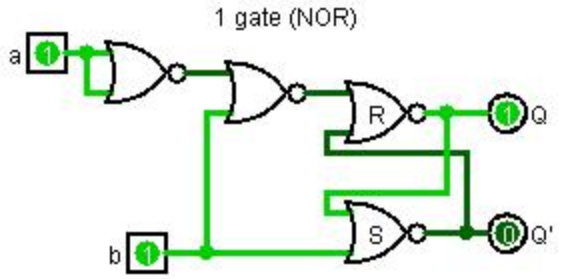
$$Q = A'B + AB = B(A'+A) = B$$
$$Q' = AB'$$

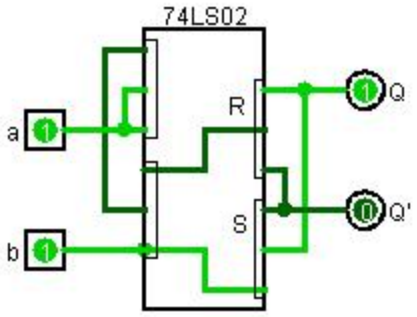
While this design implementation is correct, it uses a total of 3 logic gates, by applying Boolean algebra it is possible to realize the Boolean expression for Q' as a NOR only network:

$Q' = AB' = (AB')'' = (A'+B)' = ((A+A)'+B)'$ by De Morgan's and Involution

Therefore the design of part E now consists of a single network of NOR gates (4 in total) and can be implemented on a single TTL IC 74LS02.

Logic diagrams for the modified SR flip-flop:





OBSERVATIONS AND DATA ANALYSIS

In this lab our group observed the "memory" features of sequential logic circuits in particular the correlation between the JK, D, Toggle, and SR flip-flops. All except the SR flip-flop utilize a clock source to trigger state changes in the outputs (gated SR flip-flop can use clock signals).

DISCUSSION AND CONCLUSION

In the lab there was great difficulty when working with the function generators and the oscilloscope. After much frustration the decision to forgo “manual” clocking was made rather quickly. In this procedure the simulation of clock pulses was done so by manually changing the clock input to the individual gates from low (GND) to high (+5Vcc) repetitiously. While not ideal it proved to be reliable and allowed our group to show the in-lab demonstration to the lab instructor in a timely fashion.