ECE 85L Digital Logic Design Laboratory

Fresno State, Lyles College of Engineering
Electrical and Computer Engineering Department
Spring 2015

Laboratory 11 – Bistable Multivibrators, Characteristic Tables, and State Diagrams

1. OBJECTIVES

- Be Exposed to Several Representative Bistable Multivibrators
- Understand the use of State Diagrams and Characteristic Tables in the Description of Bistable Multivibrators
- Develop State Diagrams and Characteristic Tables for Several Representative Latches and Flip-Flops

2. DISCUSSION

2.1 Introduction to State Machines

Up to this point, the laboratory experiments have dealt primarily with Combinatorial Logic Circuits; that is, circuits whose output is only a function of (a linear combination of) inputs and whose output is independent of time (even though the inputs may in fact change with time).

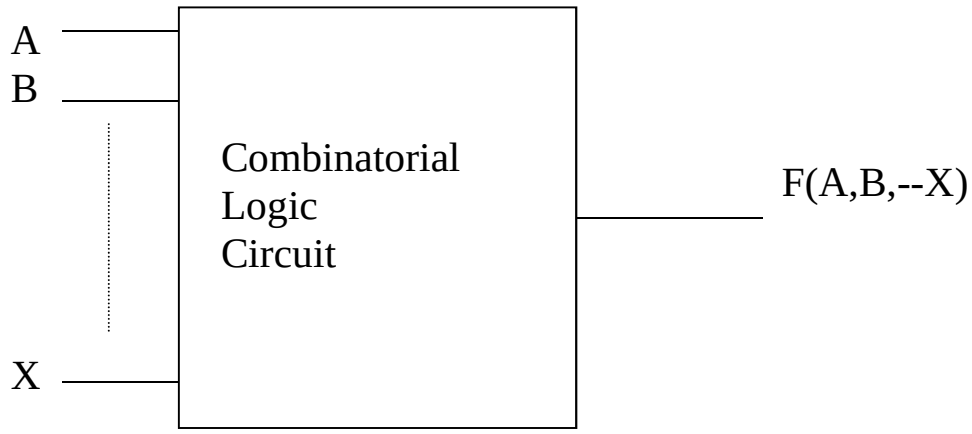

Figure 2.1: Combinatorial Logic Circuit Block Diagram

The operation of such (Combinatorial) circuits can be described either by a Boolean expression which is a function only of the input variables (and not of time) or, equivalently, by a Truth Table where the entries for a particular input combination are either (Boolean) 0 or 1.

There is another class of circuits called Sequential Circuits in which time is a dependent variable – that is the next output (at time T_{n+1}) depends not only on the inputs (***A***, ***B***, ***C*** … at time T_n) but also on the past history (T_{n-1},T_{n-2}, ...). It is convenient to characterize such circuits as State Machines, where the State of the circuit at T_n represents the (independent) Boolean status of all circuit elements in response to the previous set of inputs (***A***, ***B***, ***C***, ... at T_{n-1}). Such circuits (as opposed to simple Combinatorial Circuits) possess memory, and are fabricated from Bistable Multivibrators (and also potentially combinatorial circuits).

2.2 Bistable Multivibrators

A Bistable Multivibrator possesses one or more inputs and a set of complementary outputs usually labeled ***Q*** and ***Q****. As its name suggests, a Bistable Multivibrator has two stable states, either of which the device remains in until switched by the appropriate input(s). There are two classes of Bistable Multivibrators:

- Unclocked (Level Triggered, Asynchronous) – in which the circuit responds to each change in one or more inputs. These multivibrators often are referred to as Latches.

- Clocked (Edge-Triggered, Synchronous) – in which the circuit responds to the input(s) only on the leading or trailing edge of a clock input. These multivibrators often are referred to as Flip-Flops.

The symbol for a Bistable Multivibrator is a rectangle with input(s) on the left and output(s) on the right. A special triangular symbol (associated only with flip-flops) indicates that the flip-flop is affected by the leading edge or the trailing edge of clock.

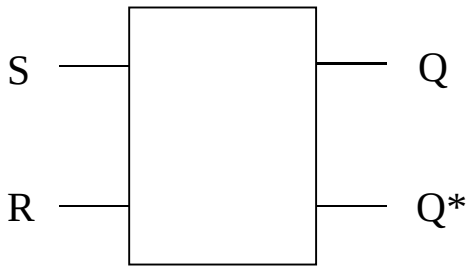

Figure 2.2: S-R Latch

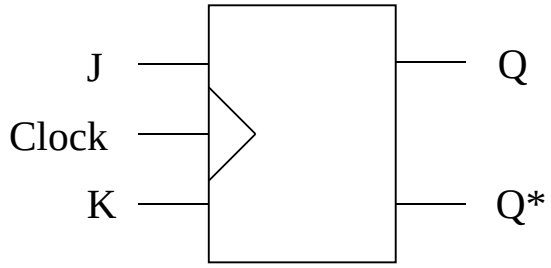

Figure 2.3: Positive Edge-Triggered J-K Flip Flop

Bistable Multivibrators are described using a Characteristic Table or a State Diagram.

2.3 Characteristic Tables

The Characteristic Table is derived from the so-called Next State Table. This (Next State) table is organized much like a Truth Table-in tabular form with inputs on the left and the output on the right.

Table 2.1: Next State Table

Inputs				**Current State**	**Next State**
A	B	C	...	Qn	Qn+1
.	.	.		.	.
.	.	.		.	.
.	.	.		.	.

Since a Bistable Multivibrator has one bit of memory (i.e. it can remember its "last state"), the current state $\boldsymbol{Q_n}$ also is an input. In this instance the next state of the output $\boldsymbol{Q_{n+1}}$ depends on the current state $\boldsymbol{Q_n}$ and the current inputs (***A***, ***B***, **C**, ... at time $\boldsymbol{T_n}$).

This table can be reduced by recognizing that for a particular combination of inputs (***A***, ***B***, ***C***, ...) the output $\boldsymbol{Q_{n+1}}$ can be either:

- 0 if for that combination of inputs the output is 0 irrespective of $\boldsymbol{Q_n}$
- 1 if for that combination of inputs the output is 1 irrespective of $\boldsymbol{Q_n}$
- Q_n if for that combination of inputs the output is the same as $\boldsymbol{Q_n}$
- Q_n* if for that combination of inputs the output is the opposite of $\boldsymbol{Q_n}$

This (reduced) table is called the Characteristic Table.

Table 2.2: J-K Multivibrator

Next State Table

Inputs		Current	Next
J	***K***	**State (Q_n)**	**State (Q_{n+1})**
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Characteristic Table

Inputs		Next
J	***K***	**State (Q_{n+1})**
0	0	Q_n
0	1	0
1	0	1
1	1	Q_n*

2.4 State Diagrams

The operation of a Bistable Multivibrator also may be explained with the use of a State Diagram. A State is represented as a circle. Within the circle is the designation for the State, which may either be some arbitrary designation (e.g. State 1, State A) or a designation which relates to the circuit itself (e.g. ***Q*** = 0, ***Q*** = 1). Transitions from one State to another are represented as directed arcs, with the direction represented by an arrow on the arc. For example:

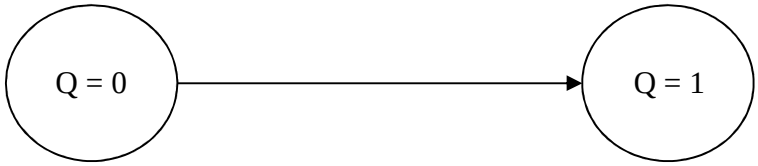

Figure 2.4: State Diagram representing a change from State ***Q*** = 0 to State ***Q*** = 1.

Each arc is labeled with the input (or combination of inputs) that produces the State change. For example:

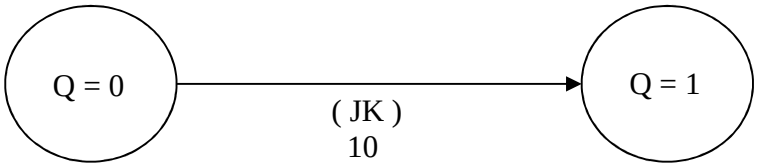

Figure 2.5: State Diagram that represents a change from State ***Q*** = 0 to State ***Q*** = 1 as a result of the inputs 10 (***J*** = 1; ***K*** = 0).

Some input combinations do not result in a State change. These are represented as an arc whose terminus is at the same State. For example:

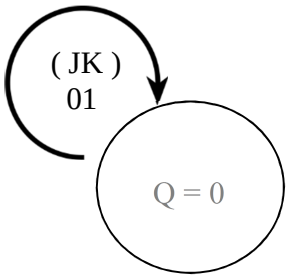

Figure 2.6: State Diagram that represents that while in State ***Q*** = 0, the input combination 01 (***J*** = 0; ***K*** = 1) does not result in any State change.

The J-K flip-flop shown in the previous example may be represented by the following State Diagram (which is equivalent to the Characteristic Table).

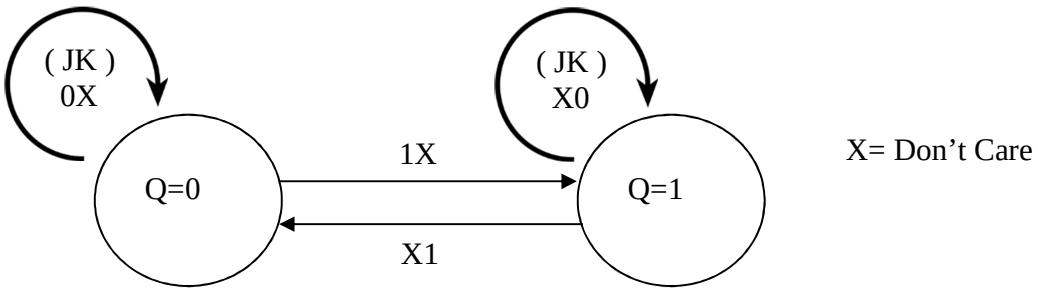

Figure 2.7: State Diagram using Don't Care States

3. PRELAB

1. Develop the State Diagram which corresponds to the following Characteristic Table:

Table 3.1: Truth Table

T	Q_{n+1}
0	Q_n
1	Q_n*

2. Develop the Characteristic Table that corresponds to the following State Diagram:

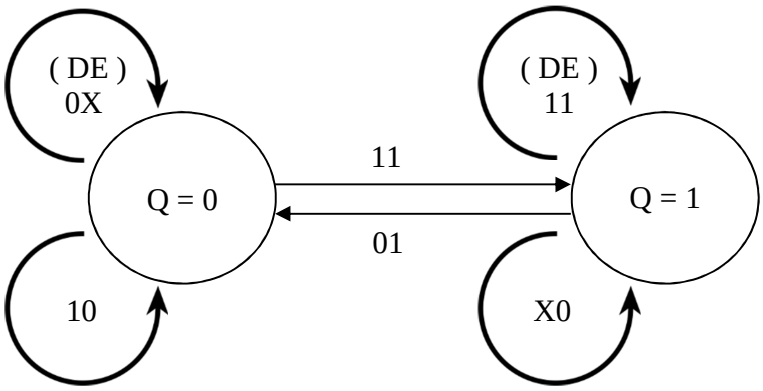

Figure 3.1: State Diagram

3. Virtually all Bistable Multivibrators are based on a form of the R*-S* Latch.

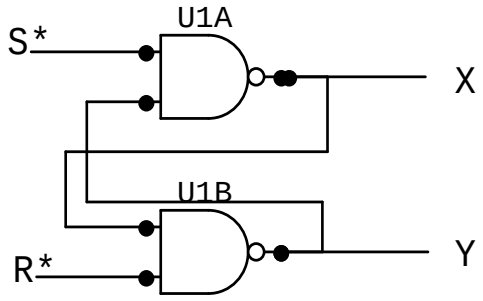

Figure 3.2: S-R Latch

Prove that the circuit has two stable states (***X*** = 0 ***Y*** = 1; ***X*** = 1 ***Y*** = 0) in the absence of input (*S** = *R** = 1).

4. By analyzing the operation of the circuit, produce a theoretical state diagram for the R*-S* latch described above. You may assume that each gate has identical propagation delay $\mathbf{T_{PD}}$ and zero rise and fall times. The theoretical State Diagram contains a circle for each ***XY*** State and an arrow for each ***R*S**** transition. Thus if ***XY*** = 01 and ***R*S**** = 10, then after one time delay, $\mathbf{T_{PD}}$, ***XY*** will be 11:

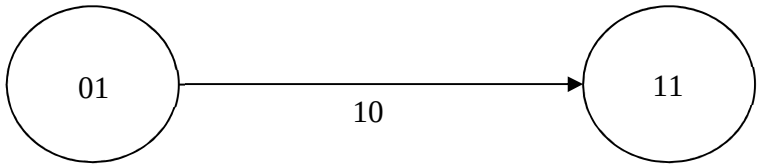

Figure 3.3: State Transition when ***XY*** = 01 and ***R*S**** = 10.

Again, if ***XY*** = 11 and ***R*S**** = 10, then an additional TPD later, ***XY*** will be 10:

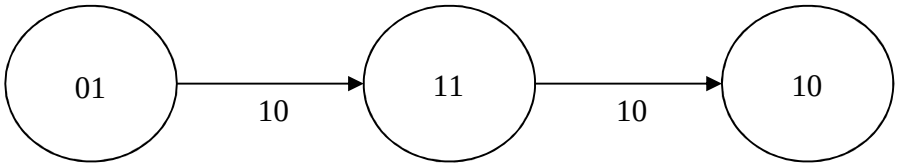

Figure 3.4: State Transitions when ***XY*** = 01 (initially) and ***R*S**** = 10.

Normally this "transient state" ***XY*** = 11 is omitted from state diagrams although it is essential to an understanding of the circuit operation. Thus the state diagram for this condition (drawn omitting the transient state) would be:

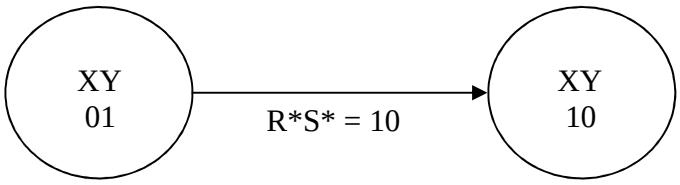

Figure 3.5: State Transition without Transient ***XY*** = 11 state.

By continuing in this way, develop the State Diagram for the R*-S* Latch. For convenience it is useful to start the diagram in one of the stable states (say ***XY*** = 01 for the input combination ***R*S**** = 11), so that the initial part of the diagram is:

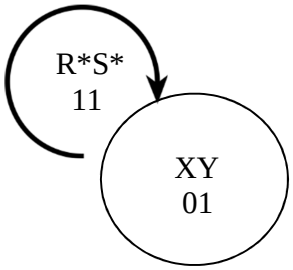

Figure 3.6: Partial State Diagram for an R*-S* Latch.

Your diagram should (theoretically) have exactly 4 States (corresponding to the four possible combinations of ***X*** and ***Y*** – although the circuit may not be able to attain all four states), and 4 arrows exiting each State (corresponding to the four possible combinations of inputs ***R**** and ***S**** for that State).

4. LAB ASSIGNMENT

4.1 R*-S* Latch

1. Construct the circuit of Figure 4.1. Connect the ***R**** and ***S**** inputs to two Logic Bit Input Switches and its ***X*** and ***Y*** outputs to two Logic Bit Output LEDs.

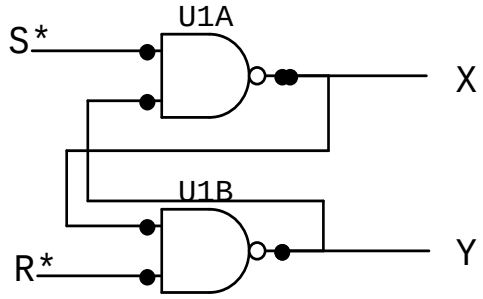

Figure 4.1: R*-S* Latch

By experiment, draw a State Diagram for this circuit: This diagram contains a circle for each ***XY*** State (e.g. 10 for State ***XY*** = 10) and an arrow for each R*-S* transition. For example,

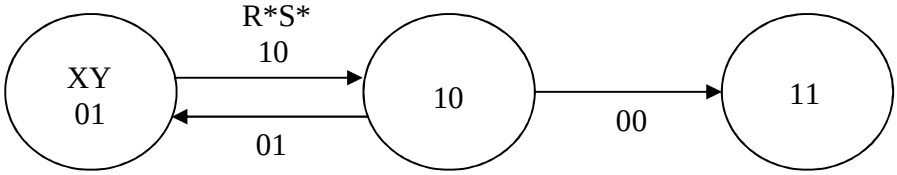

Figure 4.2: Development of State Diagram for an R*-S* Latch

Figure 4.2 is a part of the desired State Diagram for the R*-S* Latch. Note that while in State 11 under input 11 a "race" condition occurs, meaning that the next State depends on which input (***R****** or ***S****) became a logic 1 first. Explain why this is so. Represent the race in your State Diagram with a splitting dashed arrow. Check over your State Diagram to be sure it has four arrows coming out of each State. Which are the stable States? Is there a State which does not appear in your State Diagram? Compare your observed State Diagram with the theoretical one which you developed in Prelab and explain any differences.

Note that the outputs ***X*** and ***Y*** of the Latch are not always complementary even though they are frequently denoted by ***Q*** and ***Q****. The Latch is Set (***Q*** = 1) or Reset (***Q*** = 0) when only one input is asserted. When both inputs are the same the latch either "remembers" (remains in) the previous State or is forced to a State in which both outputs are the same. Check experimentally whether this happens for "1"s or "0"s at the inputs. Comment on how this circuit could be used as a "simultaneity detector" to determine which of two events occurs first.

2. One of the plagues of mechanical switches is that the contacts often "bounce" several times before settling:

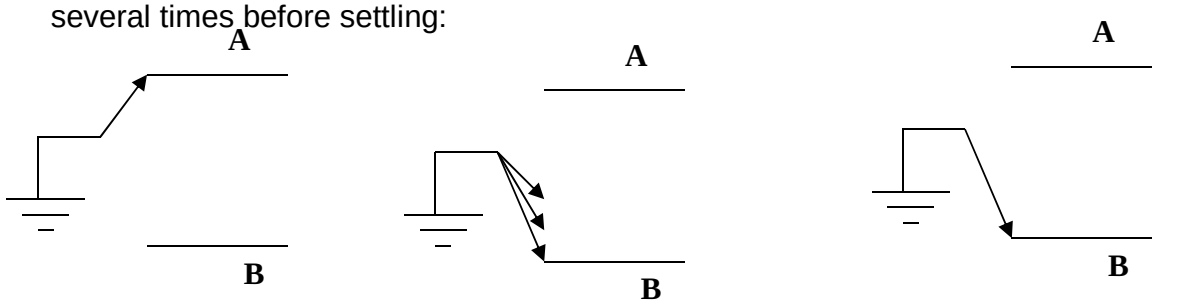

Figure 4.3: Switch Bouncing in a Circuit

As a result of the contact bounce the logic level at the switch input vacillates until the contact bounce has terminated:

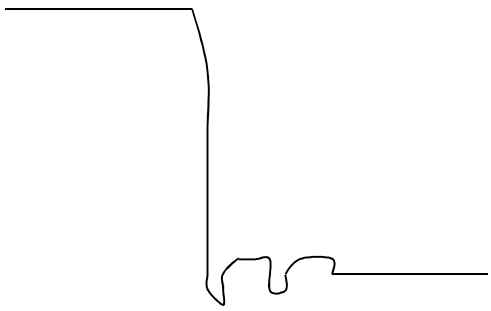

Figure 4.4: Voltage Level Uncertainty with Switch Bouncing

The R*-S* latch can be utilized to "debounce" a mechanical switch such that the output represents a "clean" signal which represents the position of the switch (**UP** = Boolean 1 and **DOWN** = Boolean 0).

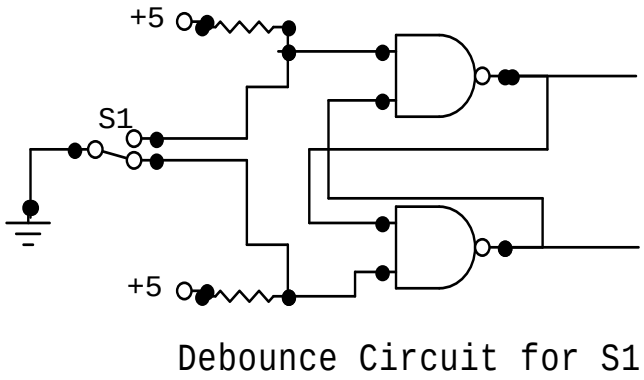

Figure 4.5: Debounce Circuit using an R*-S* Latch

Without wiring this "debounced switch", explain how it works.

4.2 Gated R-S Latch and Edge-Triggered R-S Flip-Flops

1. Figure 4.6 shows a gated latch designed to respond to the ***SE*** and ***RE*** control signals when the latch is enabled (by ***E***). Note that when the latch is disabled (***E*** = 0) the outputs (***X*** and ***Y***) remain in their current state. This condition corresponds to the ***S**** = ***R**** = 1 state of the R*-S* latch of Part 4.1. If the latch is enabled (***E*** = 1) this circuit operates identically to that of the R*-S* latch of Part 4.1 except that the inputs (***SE*** and ***RE***) are inverted. You need not fabricate this circuit, merely understand its operation, and explain how it works.

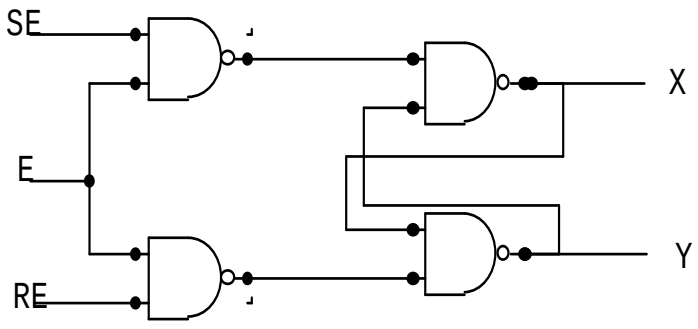

Figure 4.6: Gated R-S Latch with Enable

2. Edge-triggered R-S Flip-Flops are designed to change state only on the leading (or trailing) edge of a clock signal. Construct the following (leading) edge-detector circuit. Use all type 00 NAND gates in your implementation. Draw in your notebook the circuit you built and the theoretical response of this circuit to a square wave clock input.

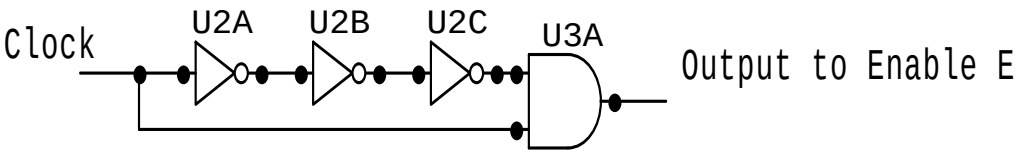

Figure 4.7: Leading Edge Detector

This circuit provides an output only because of non-zero propagation delay. Verify the operation of this circuit by observing its operation on the Scope when driven from the Function Generator at a frequency of 100 kHz.

Convert the Gated R-S Latch of Figure 4.6 into an Edge-Triggered Flip-Flop by connecting the output of the edge detector to the enable input (E) of the latch. In anticipation of the construction of the circuit of Part 4.3 (Figure 4.9), you should use three input NAND gates. Instead of using the Clock or the Function Generator as a source of clock pulses, connect the output of a Logic Bit Input Momentary Contact Pushbutton Switch to the Clock input of the Edge-Detector. Depressing and releasing this pushbutton switch generates one "clock" pulse. With the switches and LEDs connected to the circuit, study how the inputs affect the outputs. Do the outputs change on the rising edge (positive-edge triggering) or on the falling edge (negative-edge triggering) of the clock? Note that this circuit produces a very narrow pulse which, if the gates are fast enough, will be insufficient to switch the flip-flop. If this is the case you may consider widening the pulse by inserting an additional two inverters in series with the existing three.

Draw the observed State Diagram for this circuit in which transitions are labeled with values of ***SE*** and ***RE*** but not with values of the clock. Thus:

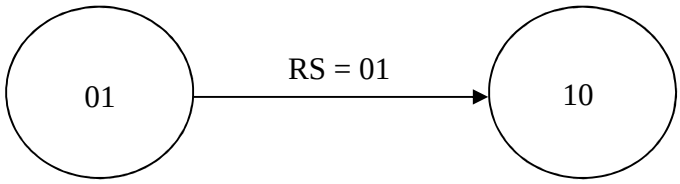

Figure 4.8: Partial State Diagram of the Edge-Trigered Flip Flop

means that at the time of a clock edge, input ***RS*** = 01 (***R*** = ***RE***; ***S*** = ***SE***) causes the circuit to switch from State 01 to State 10. How many stable states are there?

4.3 Edge-triggered J-K Flip-Flop

1. Construct the J-K circuit of Figure 4.9 which is obtained from the edge-triggered R-S flip-flop by adding feedback. Use type 00 NAND gates for the two inverters in the feedback loop. Make sure that the two gates used to form the two inverters are on the same chip so the propagation delays are the same. Note that the timing of this circuit is critical for the input combination ***J*** = ***K*** = 1. For this input combination, only one of the (three-input) input gates is enabled; the inverters in the feedback loop from ***Q*** and ***Q**** insure that the (single) input gate is enabled which causes the output to toggle (change states). For example, if the flip-flop is in the ***Q*** = 1 state only the lower input gate is enabled, so that on the next clock pulse (for ***J*** = ***K*** = 1) the flip-flop is reset to ***Q*** = 0. The critical timing arises from a "race" between the width of the clock pulse (three propagation delays) and the (four propagation delay) path which enables the opposite input gate. For the input combination ***J*** = ***K*** = 1, the leading edge of the clock will initiate the state change of the flip-flop (which will take three propagation delays to complete). This change is reflected back to the input after four propagation delays. By this time the clock pulse would have gone away lest a second flip-flop transition be gated through the input gates. Normally this circuit is self-contained within a single integrated circuit so that the gate delays can be controlled and the stability guaranteed.

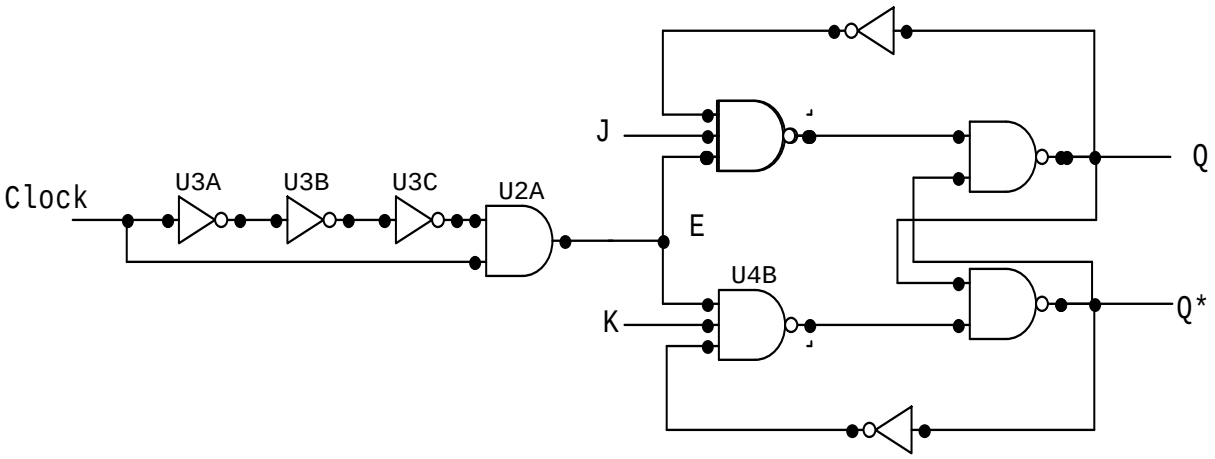

Figure 4.9: Edge-Triggered J-K Flip-Flop

2. Use Logic Bit Input Switches and Logic Bit Output LED's to produce the observed State Diagram for this circuit which has just two States, ***Q*** = 1 and ***Q*** = 0, with transitions labeled with the values of J and K but not with clock. Explain how this circuit differs from that of the Edge-Triggered R-S Flip-Flop of Part 4.3.

3. Convert this State Diagram to the Characteristic Table for the leading edge-triggered J-K flip-flop. The (leading edge) clock input is represented by either the following two symbols.

Hence:

Table 4.1: Characteristic Table for a J-K Flip Flop

Characteristic Table

J	K	CL	Q_{n+1}
0	0		
0	1		
1	0		
1	1		

Complete the Characteristic Table for the J-K Flip Flop.

4. Connect the Clock output from the Logic Development System as the Clock input to the Flip-Flop (Edge-Detector Circuit) and set the frequency to (approximately) 1Hz. With the Logic Bit Input Switches set so that ***J*** = ***K*** = 1, observe the output of the flip-flop in response to each Clock pulse.

5. REVIEW QUESTIONS

1. Why is the input combination S=R=1 not allowed for an S-R flip-flop?
2. How many potential States does a single flip-flop possess?
3. What is the difference between an R-S flip-flop and a J-K flip-flop?
4. How might the S-R latch function as a simultaneity detector?
5. Why is it necessary to debounce switches?