# Digital Logic Design
# (EL-227)
## LABORATORY MANUAL
## Spring-2021


# LAB 10
# Latches

# Lab Session 10: Latches

**OBJECTIVES:**

- Design SR Latch, D Latch and verify their functional characteristics
- Explain the difference between Sequential and Combinational logic Define 'Set', 'Reset', 'Preset' and 'Clear' concepts.

**APPARATUS:** Logic trainer, Logic probe

**COMPONENTS:** ICs 74LS08, 74LS32, 74LS04, 74LS86, 74LS02

## Combinational Logic Circuit

1. In combinational logic circuit the output of the logic circuit depends only on the present input combinations
2. No matter the inputs are changed in any order, the outputs will remain the same for a particular input combination.

## Sequential Logic Circuit

1. In sequential logic the output depends not only on the present input combinations but also on the past state of the output.
2. The output depends on the sequence in which the inputs are changed.
3. This implies that a sequential logic circuit has some kind of memory.

## Latch & Flip-Flop

In digital circuits, a flip-flop is a term referring to an electronic circuit (a bistable multivibrator) that has two stable states 0 and 1. A flip-flop is a memory device and thus will store a value while power remains in the circuit. Storage elements that operate with signal levels (rather than signal transitions) are referred to as latches; those controlled by a clock transition are flip-flops. Latches are said to be level sensitive devices; flip-flops are edge-sensitive devices. Both are the simplest examples if sequential systems.

### SR-Latch Using NOR Gate:

A bi-stable multivibrator has two stable states, as indicated by the prefix 'bi' in its name. Typically, one state is referred to as *set* and the other as *reset.* The simplest bi-stable device, therefore, is known as a *set-rest*, or *S-R Latch*.

To create an SR Latch, we can wire two NOR Gates in such a way that the output of one feedbacks to the input of the another, and vice versa. Consider the figures below for SR Latch circuit diagram as well as Pin Configuration of IC 74LS02.

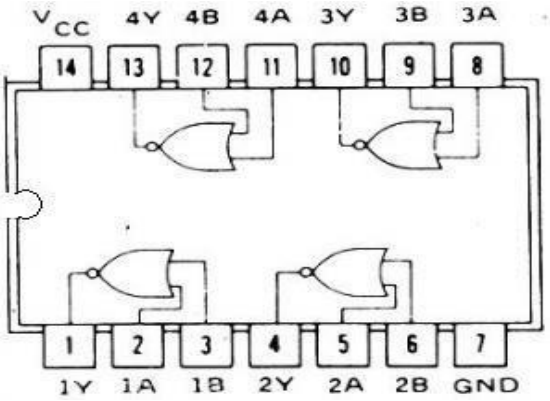


Fig1-a. IC 74LS02PinConfiguration

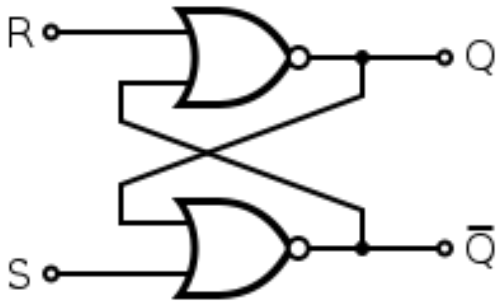


Figure1-b. SR Latch Logic Diagram

**Function Table of SR Latch**

| Inputs | | Outputs | | Comments |
|---|---|---|---|---|
| **S** | **R** | **Q** | **Q'** | |
| 0 | 0 | 0 | 0 | Initial state |
| 0 | 1 | 0 | 1 | Reset |
| 1 | 0 | 1 | 0 | Set |
| 1 | 1 | - | - | Not allowed |

At any time, only of those two inputs should be '1'. If both inputs are '1', then the next state Qt+1 value is undefined.

D-Latch

This latch is known as data latch. It has a data input (D) and enable input (E). It will not respond to a signal input if the enable input is 0. When the enable input is 1, however, Q output follows the D input. This latch has no *"invalid"* or *"illegal"* state. Q and Q' are always opposite of one another.

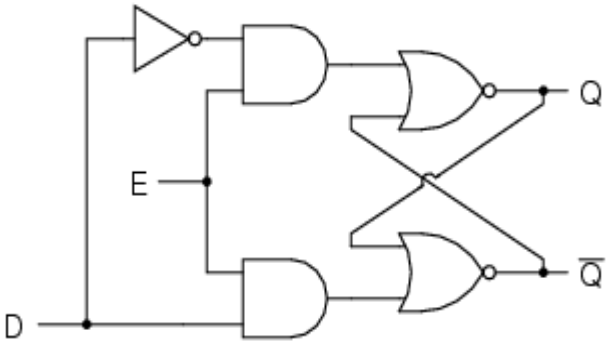


Figure2. D Latch Logic Diagram


Figure3. D and S-R logic Diagram

**Function Table of D Latch**

| Inputs | | Outputs | | Comments |
|---|---|---|---|---|
| **E** | **D** | **Q** | **Q'** | |
| 0 | 0 | Latch | | No change |
| 0 | 1 | Latch | | No change |
| 1 | 0 | 0 | 1 | Set |
| 1 | 1 | 1 | 0 | Reset |

**Lab Task #1**

**Implement SR Latch by using logic gates.**

**Lab Task #2**

**Implement D Latch by using logic gates.**