Digital Logic Design
(EL-1005)
LABORATORY MANUAL
Spring-2025

LAB 09
Multiplexers and Latches

_____ _____ ___

STUDENT NAME ROLL NO SEC

_____

INSTRUCTOR SIGNATURE& DATE

MARKS AWARDED: /10

Lab Session 09: Multiplexers and Latches

OBJECTIVES:

- Understand the basic concept of Multiplexing
- Perform the 1-of-8-line multiplexing by using 74LS151 eight input multiplexer IC
- Familiarize with inputs, outputs, data select and enable pins of 74LS151
- Discuss practical applications of Multiplexer

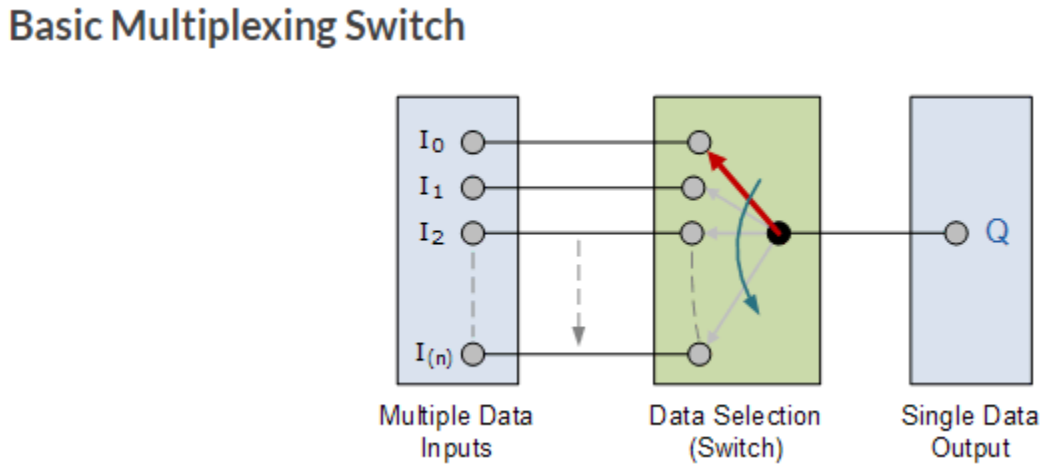
APPARATUS: Logic trainer, Logic probe,

COMPONENTS: **74LS151 (8x1 Multiplexer)**

THEORY:

Multiplexing is the generic term used to describe the operation of sending one or more analogue or digital signals over a common transmission line at different times or speeds and as such, the device we use to do just that is called a **Multiplexer**.

The *multiplexer*, shortened to "MUX" or "MPX", is a combinational logic circuit designed to switch one of several input lines through to a single common output line by the application of a control signal. Multiplexers operate like very fast acting multiple position rotary switches connecting or controlling multiple input lines called "channels" one at a time to the output.



A **74LS151** has eight inputs that can be individually selected by three select lines. The output is connected to the input line selected by the binary value on the three select lines. If the three select lines are all zeros, then input line "0" is selected and connected to the output line.

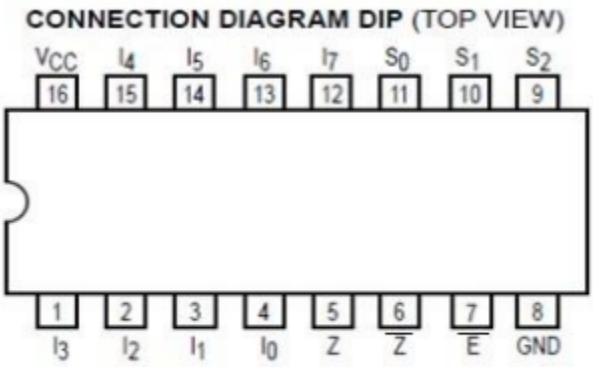


Fig 1. Pin Configuration

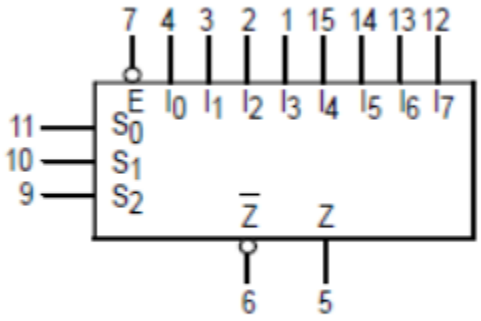


Fig 2. Logic Symbol

PIN Description:

S0–S2	**Select Inputs**
E'	**Enable (Active LOW)**
Input I0–I7	**Multiplexer Inputs**
Z	**Multiplexer Output**
Z'	**Complementary Multiplexer Output**

Applications of Multiplexer:

Multiplexer are used in various fields where multiple data need to be transmitted using a single line. Following are some of the applications of multiplexers –

1. **Communication system –**

Communication system is a set of system that enable communication like transmission system, relay and tributary station, and communication network. The efficiency of communication system can be increased considerably using multiplexer. Multiplexer allow the process of transmitting different type of data such as audio, video at the same time using a single transmission line.

2. **Telephone network –**

In telephone network, multiple audio signals are integrated on a single line for transmission with the help of multiplexers. In this way, multiple audio signals can be isolated and eventually, the desire audio signals reach the intended recipients.

3. **Computer memory –**

Multiplexers are used to implement huge amount of memory into the computer, at the same time reduces the number of copper lines required to connect the memory to other parts of the computer circuit.

LATCHES

Latches are digital circuits that store a single bit of information and hold its value until it is updated by new input signals. They are used in digital systems as temporary storage elements to store binary information. Latches can be implemented using various digital logic gates, such as AND, OR, NOT, NAND, and NOR gates.

There are two types of latches:

1. S-R (Set-Reset) Latches: S-R latches are the simplest form of latches and are implemented using two inputs: S (Set) and R (Reset). The S input sets the output to 1, while the R input resets the output to 0. When both S and R are at 1, the latch is said to be in an "undefined" state.
2. D (Data) Latches: D latches are also known as transparent latches and are implemented using two inputs: D (Data) and a clock signal. The output of the latch follows the input at the D terminal as long as the clock signal is high. When the clock signal goes low, the output of the latch is stored and held until the next rising edge of the clock.
3. Latches are widely used in digital systems for various applications, including data storage, control circuits, and flip-flop circuits. They are often used in combination with other digital circuits to implement sequential circuits, such as state machines and memory elements.

SR (Set-Reset) Latch – They are also known as preset and clear states. The SR latch forms the basic building blocks of all other types of flip-flops.

SR Latch is a circuit with:

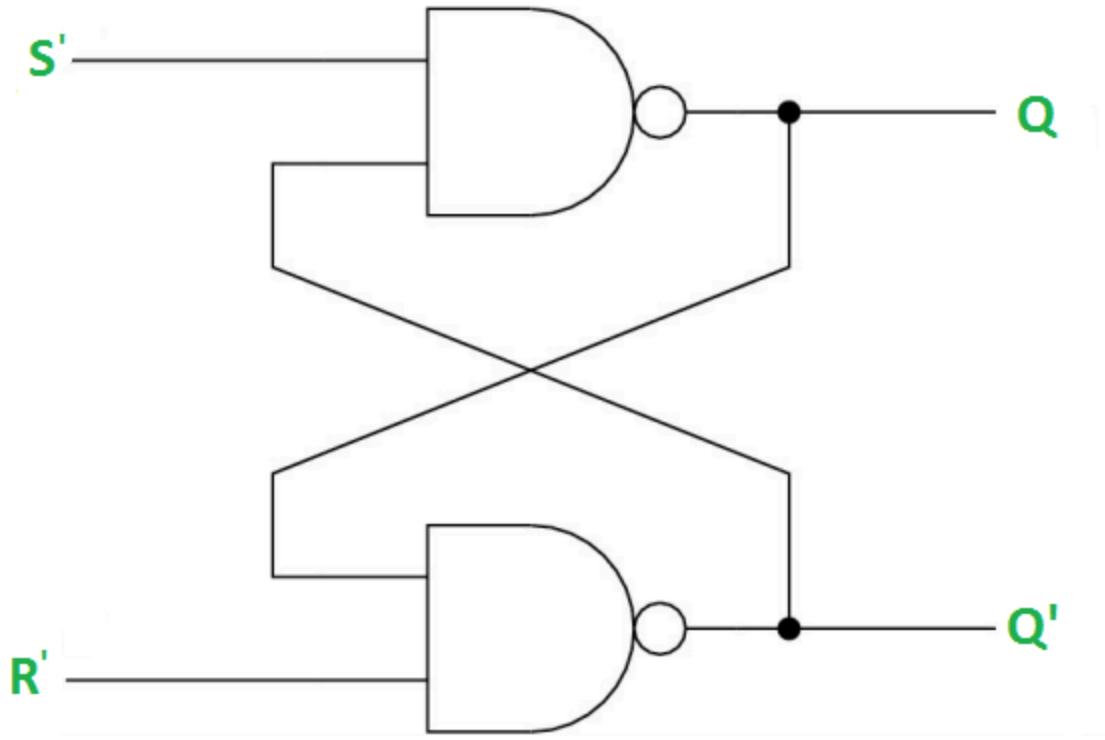
(i) 2 cross-coupled NOR gate or 2 cross-coupled NAND gate.

(ii) 2 input S for SET and R for RESET.

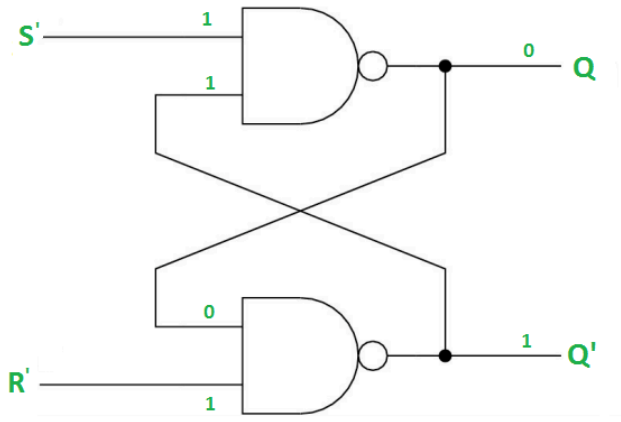
(iii) 2 output Q, Q'.

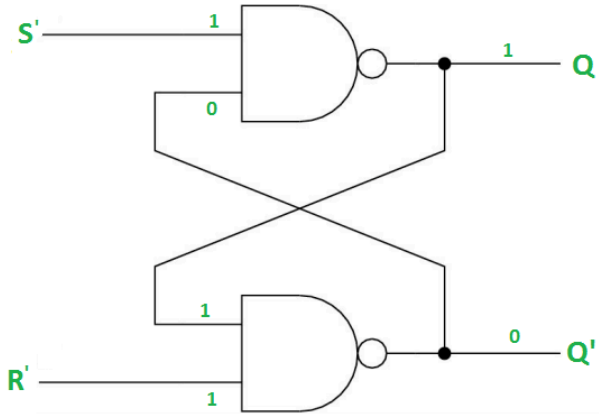
Q	**Q'**	**STATE**
1	0	Set
0	1	Reset

Under normal conditions, both the input remains 0. The following is the RS Latch with NAND gates:

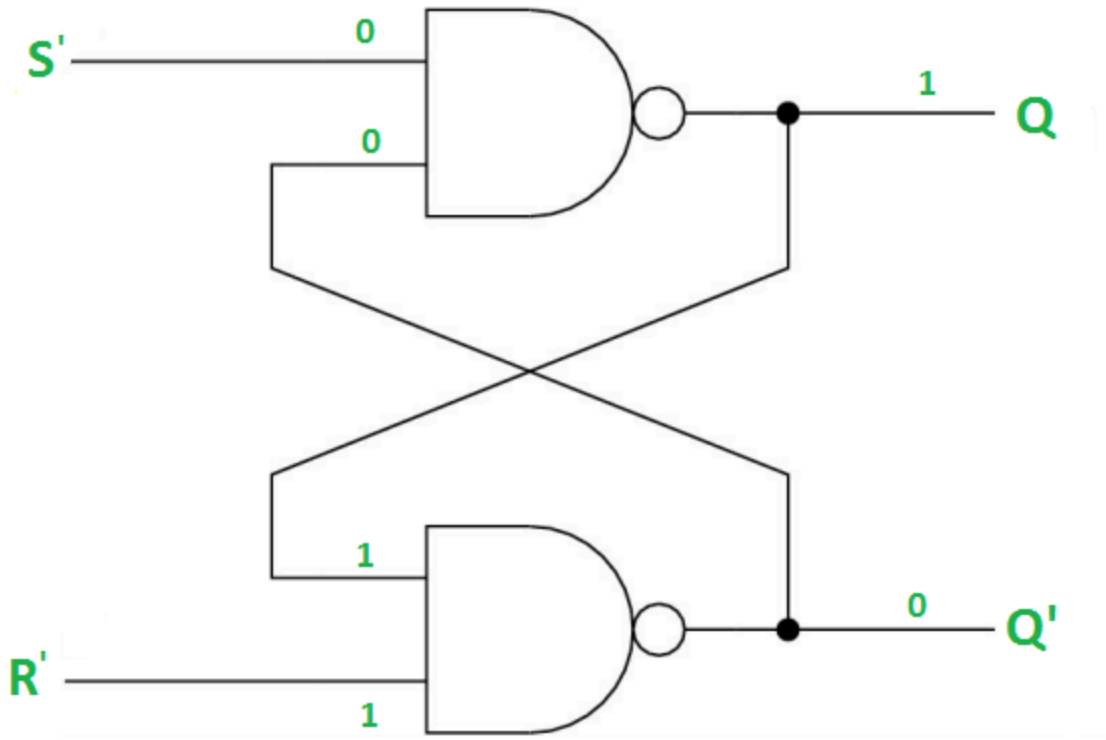


Case-1: S'=R'=1 (S=R=0) –
If Q = 1, Q and R' inputs for 2nd NAND gate are both 1.
If Q = 0, Q and R' inputs for 2nd NAND gate are 0 and 1 respectively.

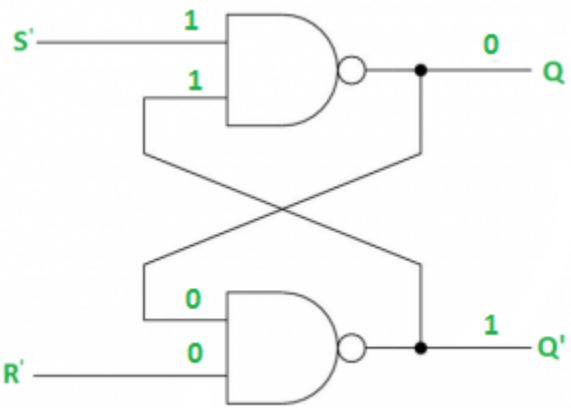




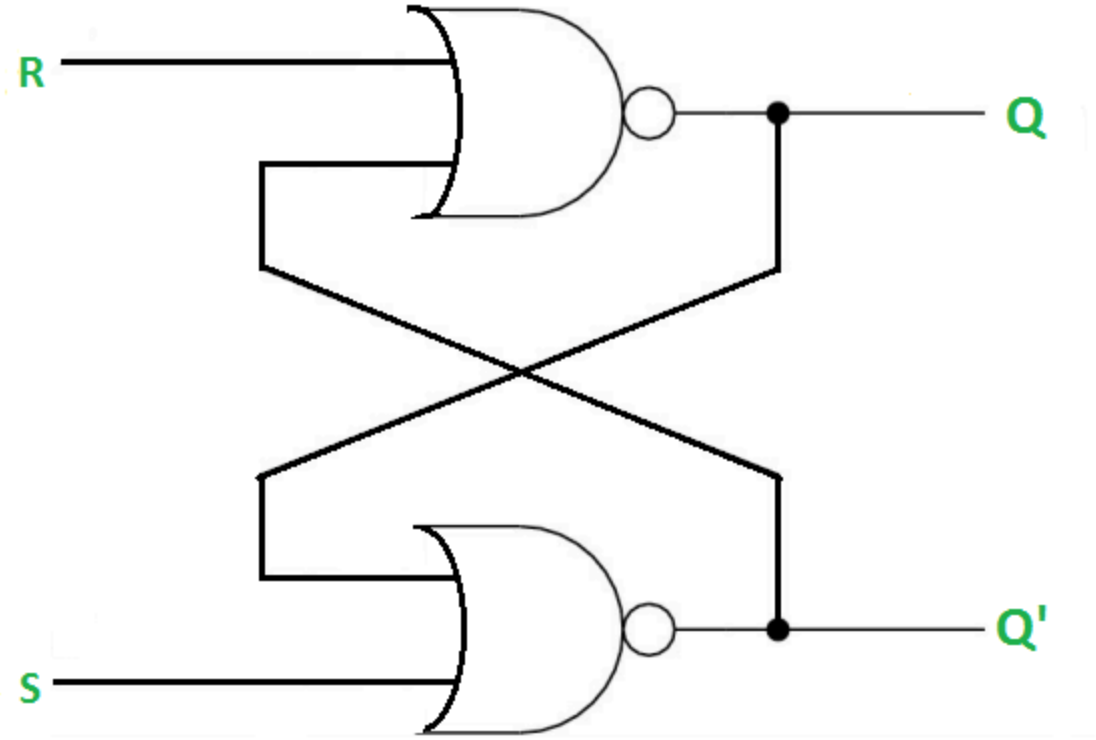
Case-2: S'=0, R'=1 (S=1, R=0) –
As S'=0, the output of 1st NAND gate, Q = 1(**SET state**). In 2nd NAND gate, as Q and R' inputs are 1, Q'=0.



Case-3: S'= 1, R'= 0 (S=0, R=1) –
As R'=0, the output of 2nd NAND gate, Q' = 1. In 1st NAND gate, as Q and S' inputs are 1, Q=0(**RESET state**).



Case-4: S'= R'= 0 (S=R=1) –
When S=R=1, both Q and Q' becomes 1 which is not allowed. So, the input condition is prohibited.
The SR Latch using NOR gate is shown below:

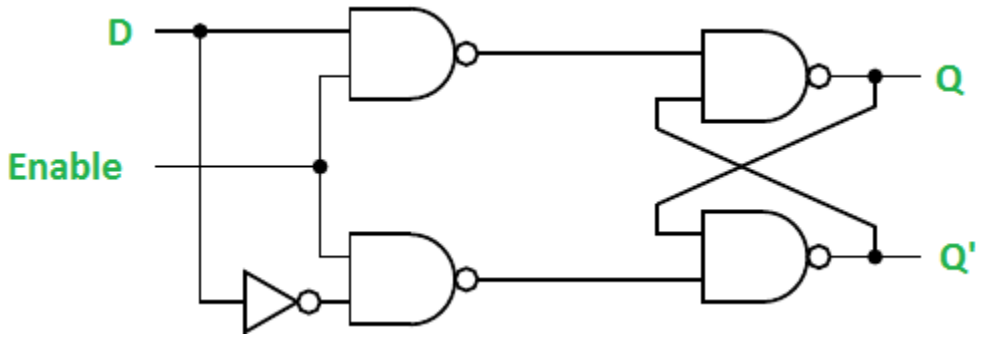


Gated SR Latch –

A Gated SR latch is a SR latch with enable input which works when enable is 1 and retain the previous state when enable is 0.

Gated D Latch –

D latch is similar to SR latch with some modifications made. Here, the inputs are complements of each other. The letter in the D latch stands for “data” as this latch stores single bit temporarily. The design of D latch with Enable signal is given below:



The truth table for the D-Latch is shown below:

Enable	**D**	**Q(n)**	**Q(n+1)**	**STATE**
1	0	x	0	RESET

Enable	D	Q(n)	Q(n+1)	STATE
1	1	x	1	SET
0	x	x	Q(n)	No Change

As the output is same as the input D, D latch is also called as *Transparent Latch*. Considering the truth table, the characteristic equation for D latch with enable input can be given as:

Q(n+1) = EN.D + EN'.Q(n)

LAB TASK

1. Design and simulate a 4:1 multiplexer using logic gates.
2. Design and simulate a 4:1 multiplexer using logic gates.
3. Create a 16-1 mux using 2 8-1 mux but one multiplexer should have a 2 bit input window while the other one should have a single bit input window. (Hint use splitters from logism)
4. Create a Demux from the same Q3 circuit with 16 outputs.
5. Design a D Latch Using NAND Gates Only
6. Design a Gated SR Latch
7. Design a SR Latch using NOR gate
8. Create a 3-bit memory using gated S-R Latches the inputs should be the output of a 2-bit adder
9. Create the same circuit with D-Latch but the output should be from a multiplier circuit