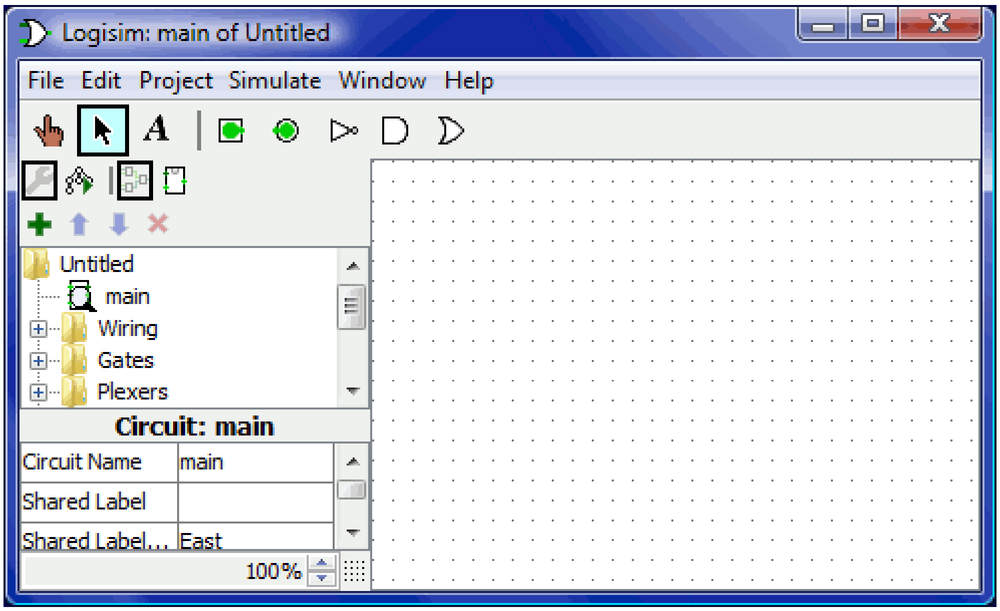
PRACTICAL - 1

AIM: Introduction of Tool **Logisim** and revision on basic Logic gates

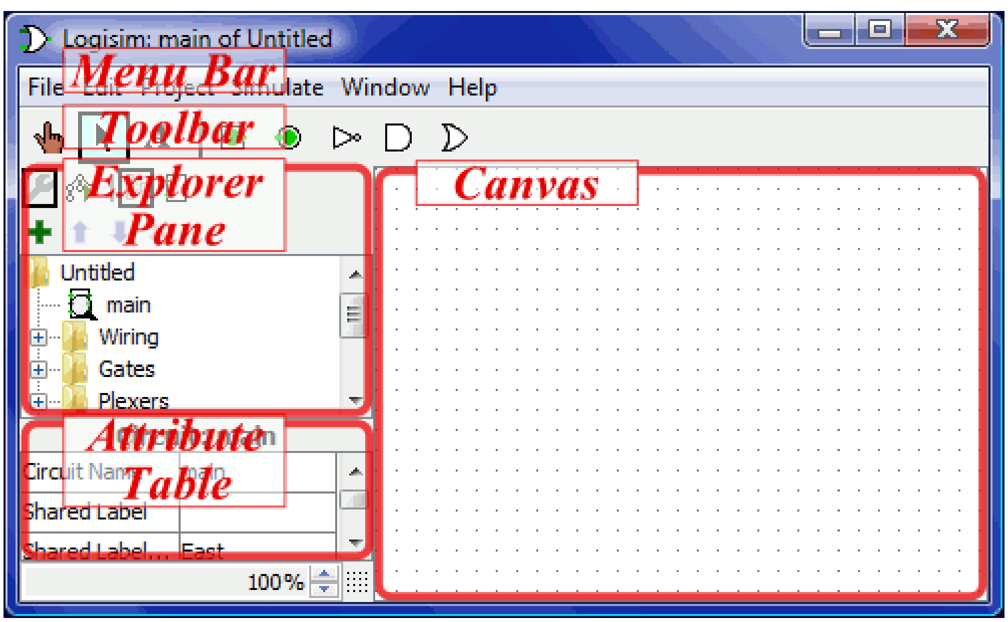
Logisim allows you to design and simulate digital circuits. It is intended as an educational tool, to help you learn how circuits work.

Getting Started:

When you start Logisim, you'll see a window similar to the following.



All Logisim is divided into three parts, called the *explorer pane*, the *attribute table*, and the *canvas*. Above these parts are the *menu bar* and the *toolbar*.



The canvas is

where you'll draw your circuit; and the toolbar contains the tools that you'll use to accomplish this.

Toolbar:

- The toolbar contains short cuts to several commonly used items.
- The poke tool (shaped like a hand) is used to alter input pins.
- The input pin (green circle surrounded by a box) is used to send a signal through a wire. When placing the input on the canvas it initializes to 1-bit. This number of bits can be increased in the Attribute Table.
- The output pin (green circle in a circle) is used to observe output from a gate. The output pin toggles in real time as long as the simulation is enabled from the menu bar Simulate > Simulate enabled.

Explorer Pane:

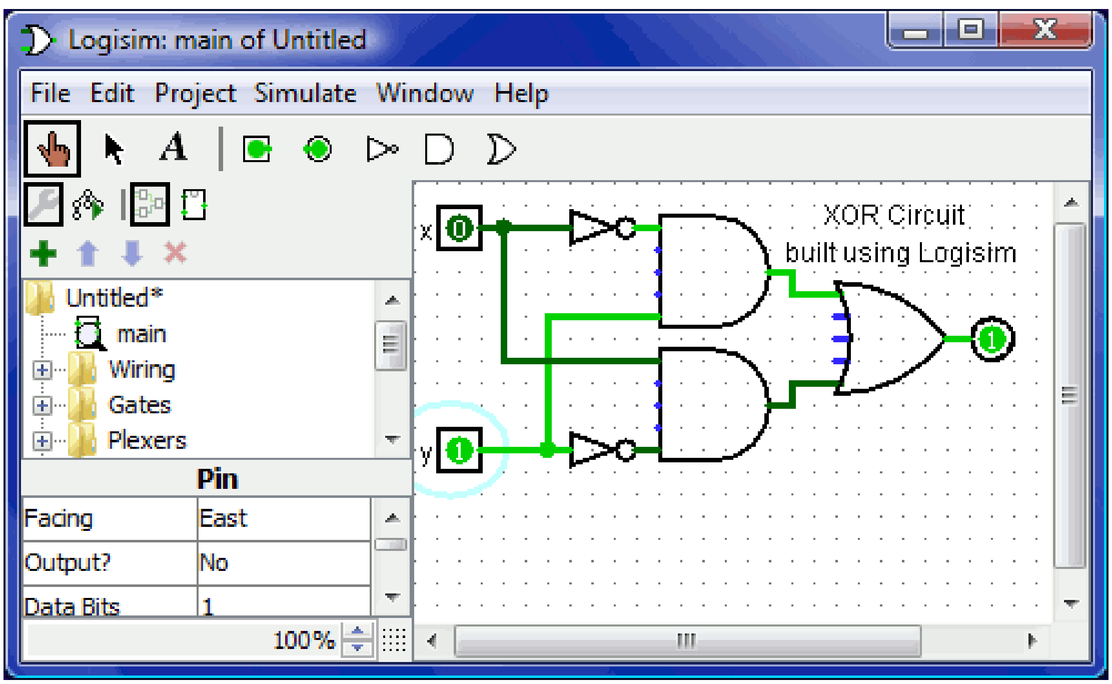
- The list of wiring, gates, multiplexers, etc... That are available for digital design in Logisim. Please note not all items are allowed to be used in every project.

Attribute Table:

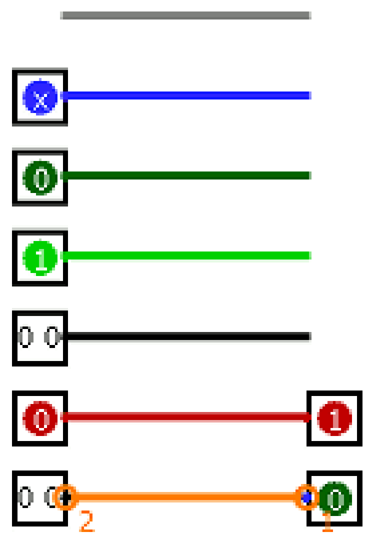
- Gives detailed attributes of digital design components (e.g., AND, OR, XOR gates). The attribute table allows you to alter the number of inputs/outputs that a digital design component.

Canvas:

- The canvas is the area for you to create your digital circuits. In the canvas area you may simulate your circuits while designing in real time.



Wire colors:



- **Gray:** The wire's bit width is unknown. This occurs because the wire is not attached to any components' inputs and outputs. (All inputs and outputs have a defined bit width.)
- **Blue:** The wire carries a one-bit value, but nothing is driving a specific value onto the wire. We call this a *floating* bit; some people call it a *high-impedance* value. In this example, the component placing a value onto the wire is a three-state pin, so it can emit this floating value.
- **Dark green:** The wire is carrying a one-bit 0 value.
- **Bright green:** The wire is carrying a one-bit 1 value.
- **Black:** The wire is carrying a multi-bit value. Some or all of the bits may not be specified.
- **Red:** The wire is carrying an error value. This often arises because a gate cannot determine the proper output, perhaps because it has no inputs. It could also arise because two components are trying to send different values onto the wire; this is what happens in the above example, where one input pin places 0 onto the wire while another places 1 onto the same wire, causing a conflict. Multi-bit wires will turn red when any of the bits carried are error values.
- **Orange:** The components attached to the wire do not agree in bit width. An orange wire is effectively "broken": It does not carry values between components. Here, we've attached a two-bit component to a one-bit component, so they are incompatible.

Simulation:

Simulation is not tricky, we can auto simulate as if it were a live circuit and quickly see the result. However there are several other types of simulation which are important in different situations. The most common case will be when using a clock to enable a stateful circuit (i.e., a circuit with some type of memory). The types of simulation supported by Logisim are:

- **Simulate Enabled**:

Let the circuit run based on it’s inputs. Must be enabled for all tick simulations to work as well.

- **Step Simulation**:

Allows the user to simulate a single step at a time. If an input changes in step simulation you must advance the signal through each gate by stepping (ctrl-i).

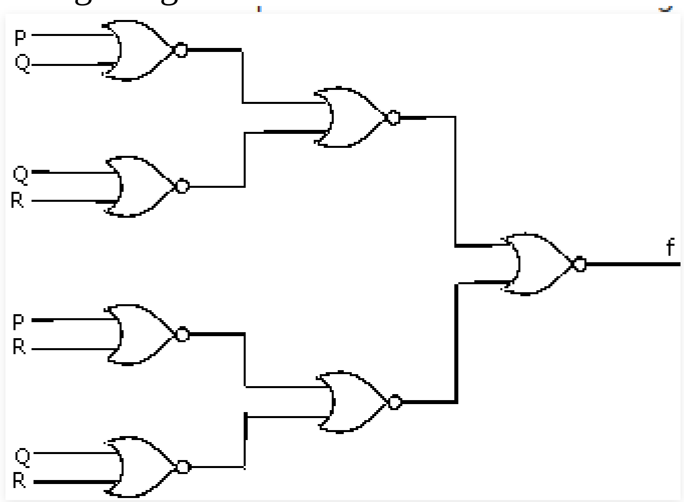
- **Tick Simulation**

Used to tick a clock (found in Explorer Plane Wiring > Clock). This is vital for stateful circuits (e.g., RAM, flip-flops, etc...)

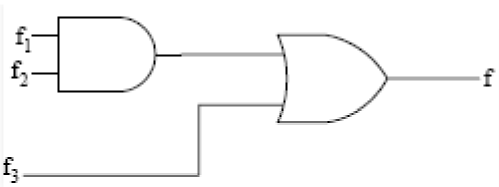
Tick Once:
Tick the clock once (go from high to low or vice versa)
Ticks Enabled:
Tick automatically at the rate of tick frequency
Tick Frequency:
How often to tick the clock (measured in Hz).

Exercise:

1. Build Logical Circuit for Full Adder/ Subtractor using Logisim tool.
2. What is the Boolean expression for the output f of the combinational logic circuit of NOR gates given below?



3. Given f_1, f_3 and f in canonical sum of products form (in decimal) for the circuit given circuit:



$f_1 = \Sigma\, m\,(4, 5, 6, 7, 8)$
$f_3 = \Sigma\, m\,(1, 6, 15)$
$f = \Sigma\, m\,(1, 6, 8, 15)$
then f_2 is

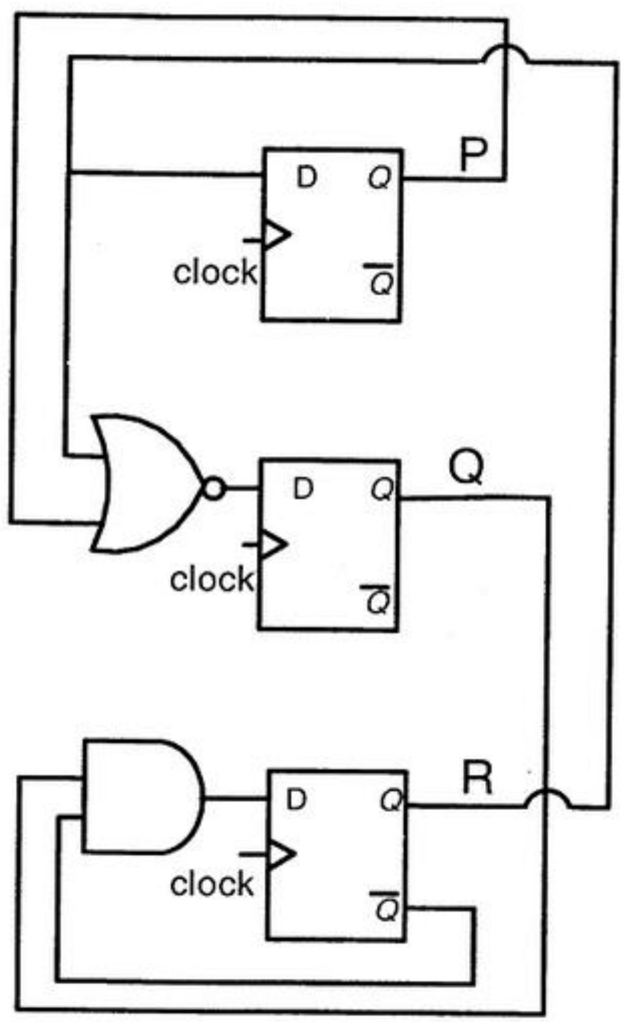
A) $\Sigma m(4, 6)$

B) $\Sigma m(4, 8)$

C) $\Sigma m(6, 8)$

D) $\Sigma m(4, 6, 8)$

4. Consider the following circuit involving three D-type flip-flops used in a certain type of counter configuration. If at some instance prior to the occurrence of the clock edge, P, Q and R have a value 0, 1 and 0 respectively, what shall be the value of PQR after the clock edge?



5. In the sequential circuit shown below, if the initial value of the output Q1Q0 is 00, what are the next four values of Q1Q0?