



CSCI 150

Introduction to Digital and Computer System Design

Lecture 2: Combinational Logical Circuits I



Jetic Gū
2020 Summer Semester (S2)

Overview

- Focus: Boolean Algebra
- Architecture: Combinatory Logical Circuits
- Textbook v4: Ch2 2.1 2.2; v5: Ch2 2.1 2.2
- Core Ideas:
 1. Logical Gates
 2. Introduction to LogicWorks

Logic Gates

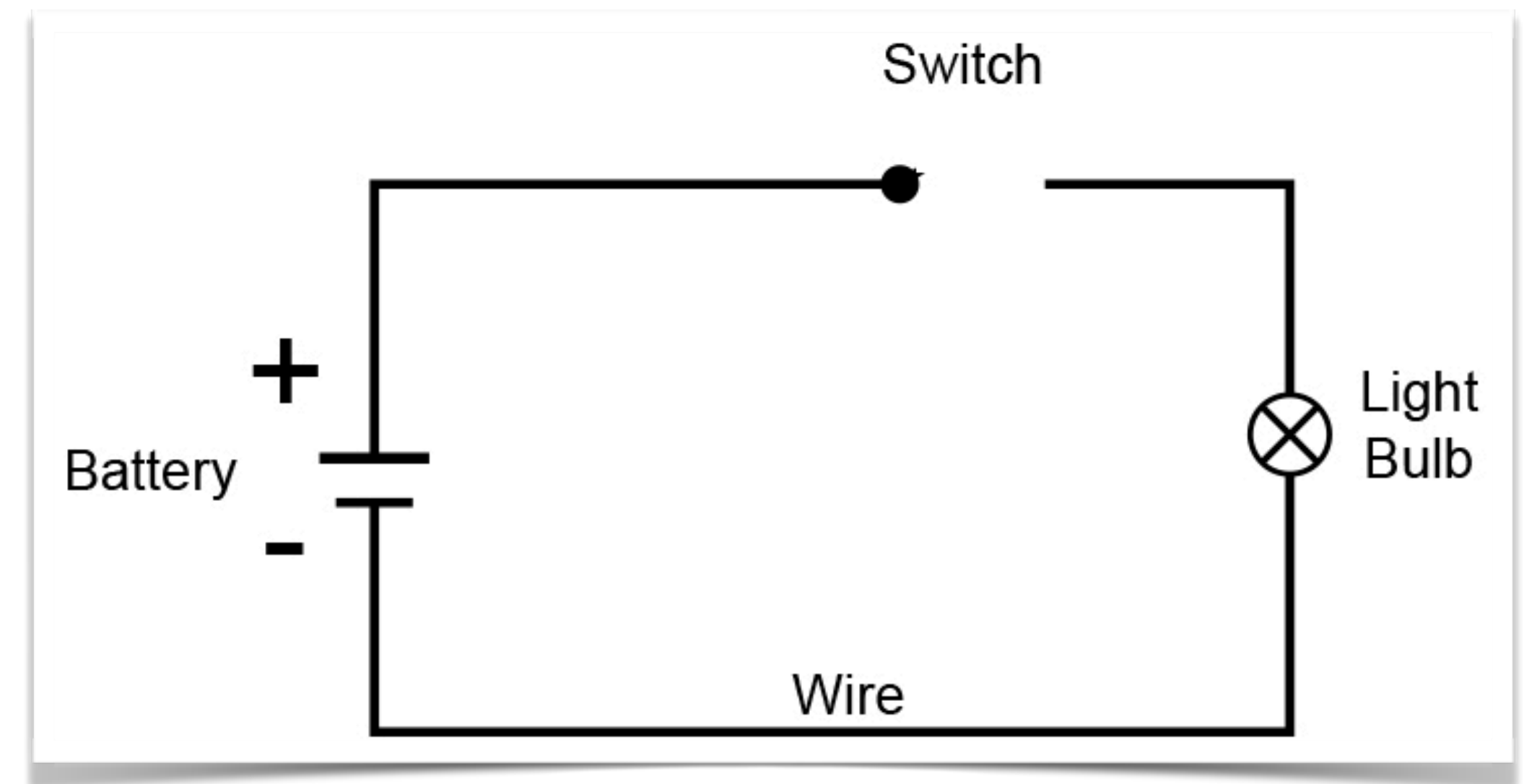
And, Or, Not Gates, LogicWorks

What is a Logic Gate?

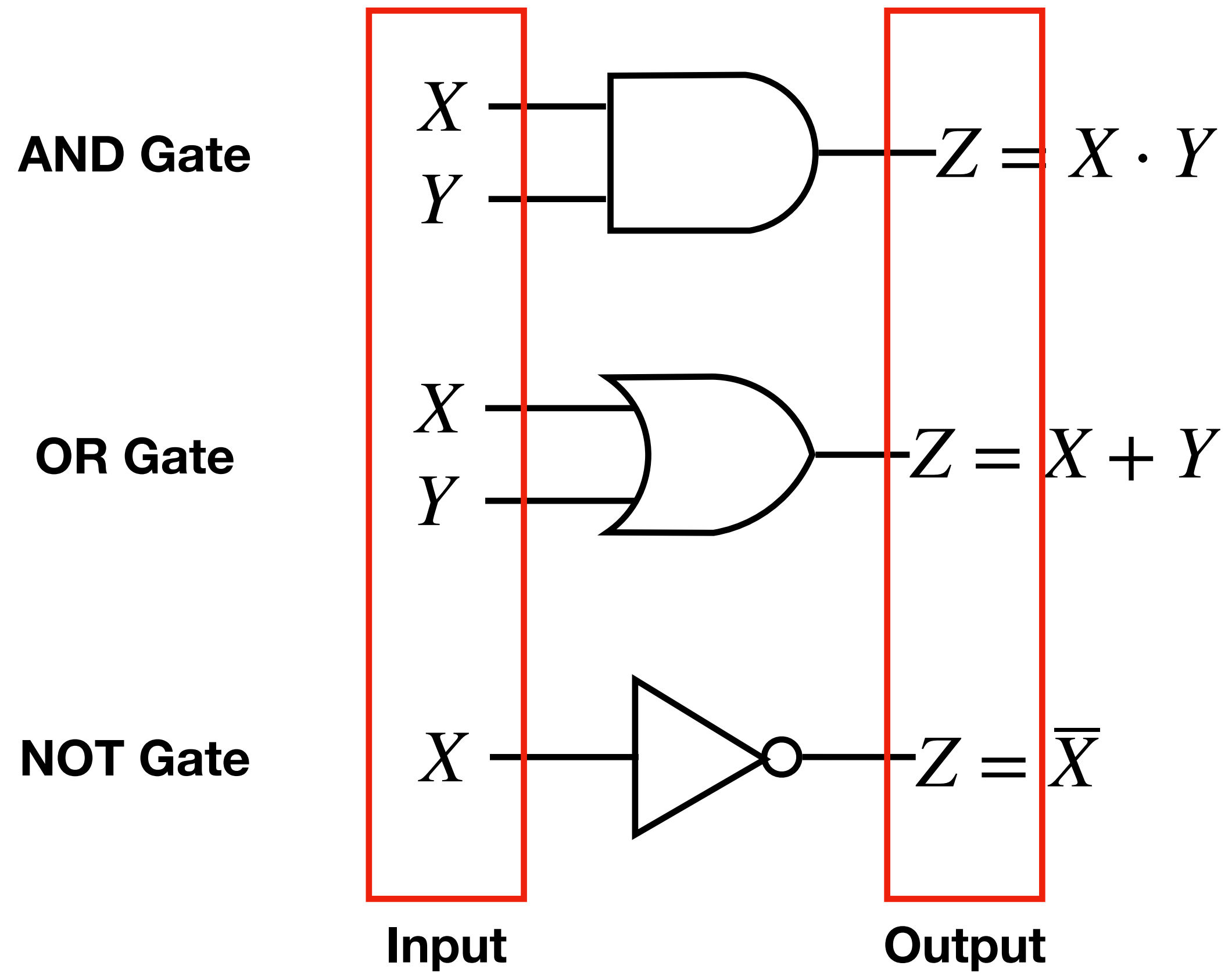
- A basic circuit unit implemented using transistors and interconnections
- We when analysing a digital circuit, are not concerned with the internals of a gate, but only it's external properties
- Performs a single logical operation
operate on one or more binary input signals to produce an output signal

What is a Logic Gate?

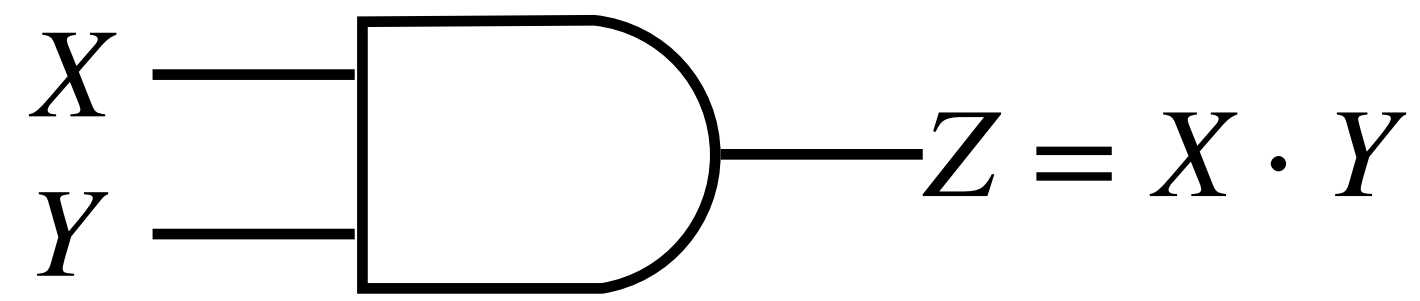
- Similar to in electric circuit design, we are not concerned with the design of the lightbulb or battery, but we know what it does.
- A logic gate is like that, we know it's external logic properties, that's enough.



First 3 Gates

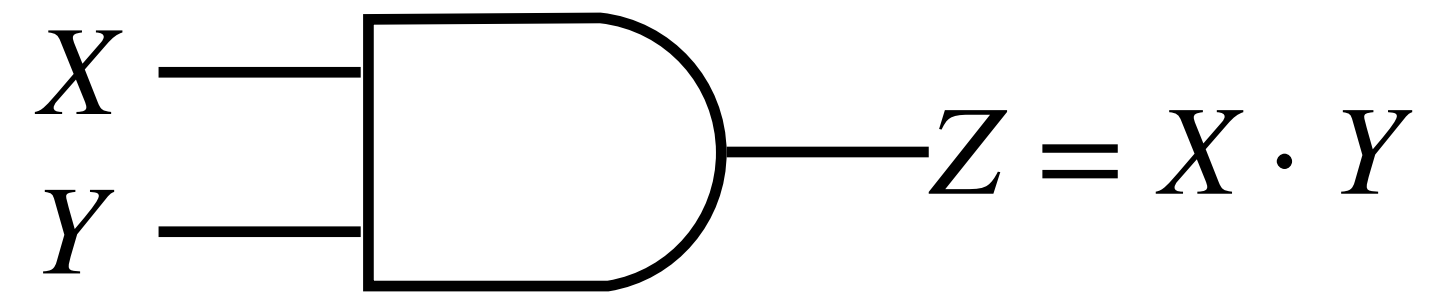


AND Operator and Boolean Algebra / Binary Logic



- Boolean Algebra
 - Each variable can only have one of two values:
 - TRUE/ON/1
 - False/OFF/0
 - AND: Z is equal to X AND Y
 - Operator: \cdot (`\cdot`)

AND Operator and Boolean Algebra

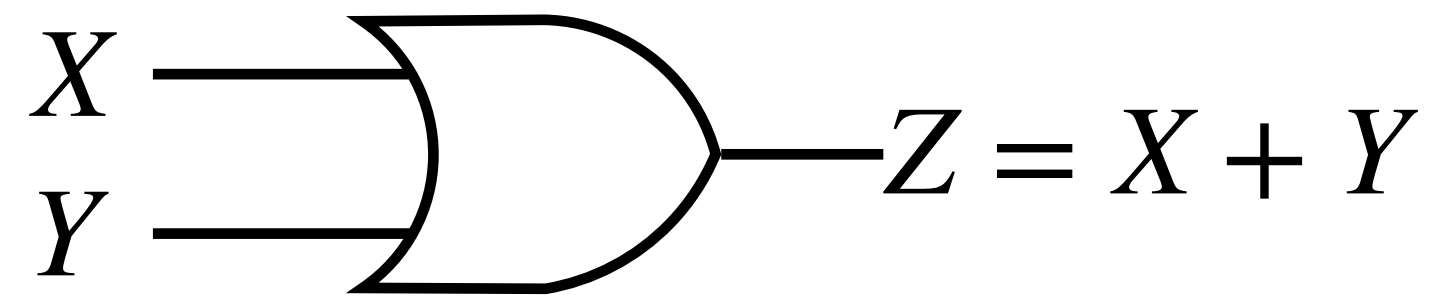


AND Truth Table

| X | Y | $Z = X \cdot Y$ |
|-----|-----|-----------------|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

- AND: Z is equal to X AND Y
- Operator: \cdot (`\cdot`)
- Truth Table
 - Left: all combinations of input values
 - Right: corresponding output values

OR Operator

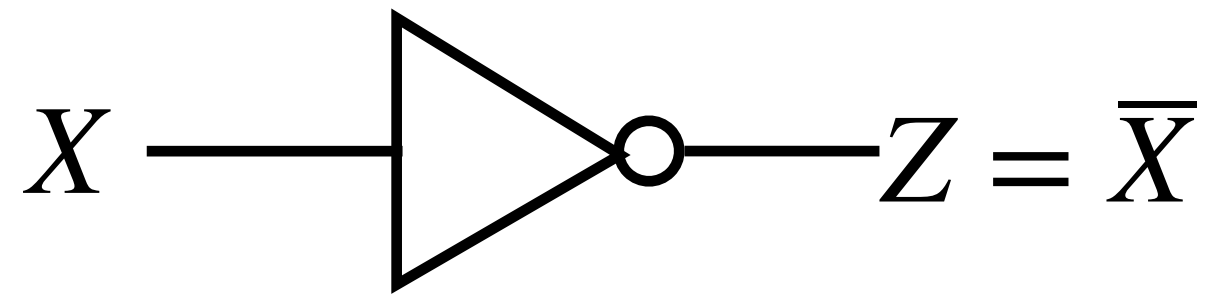


OR Truth Table

| X | Y | $Z = X + Y$ |
|-----|-----|-------------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

- OR: Z is equal to X or Y
- Operator: $+$

NOT Operator



OR Truth Table

| X | $Z = \bar{X}$ |
|-----|---------------|
| 0 | 1 |
| 1 | 0 |

- NOT: Z is equal to NOT X
- Operator: \bar{X} (\overline{X})
- Also called: *Complement* operation;
Inverter gate

Logic Gate and Boolean Algebra

- Logic Gates
 - AND Gate, OR Gate, NOT Gate
 - Actual physical components
- Boolean Algebra Operators
 - AND (\cdot), OR ($+$), NOT (\bar{X})
 - Mathematical Representations

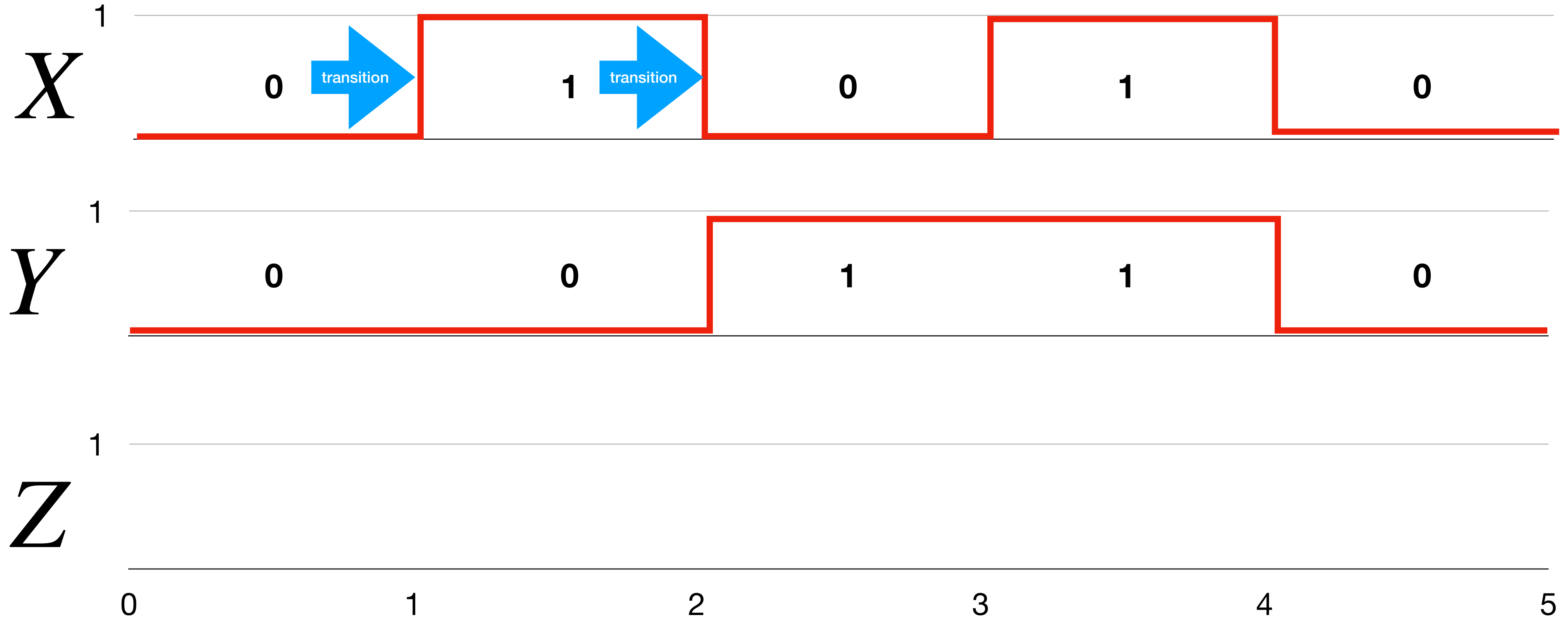
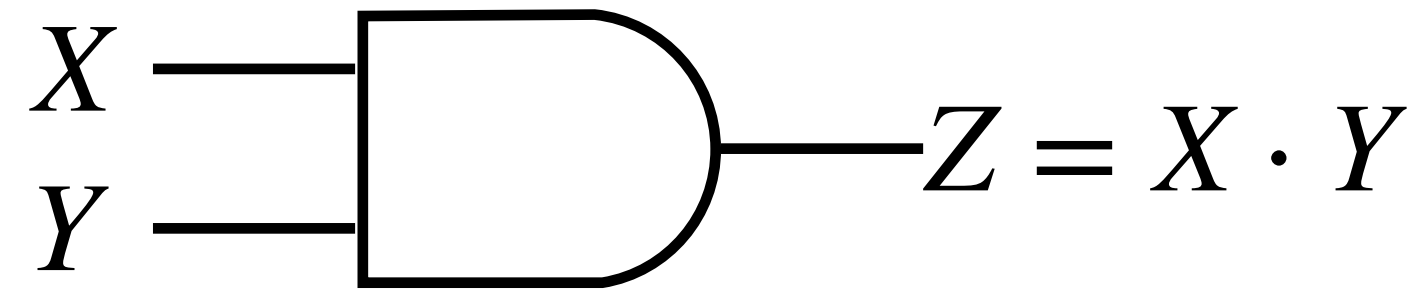
**THEY HAVE
DIFFERENCES!**

Digital Logical Gates

- In math, everything happens simultaneously
- An equation like $250 + 760 = 1010$ doesn't change with time/location
- In digital circuits, we have **electrons as 'messengers'**. They travel at about 2,200 kilometres per second
- Logic gates are tiny circuits, which means they still have internal components: even slower
- This means: there will be tiny delays called **Gate Delay**

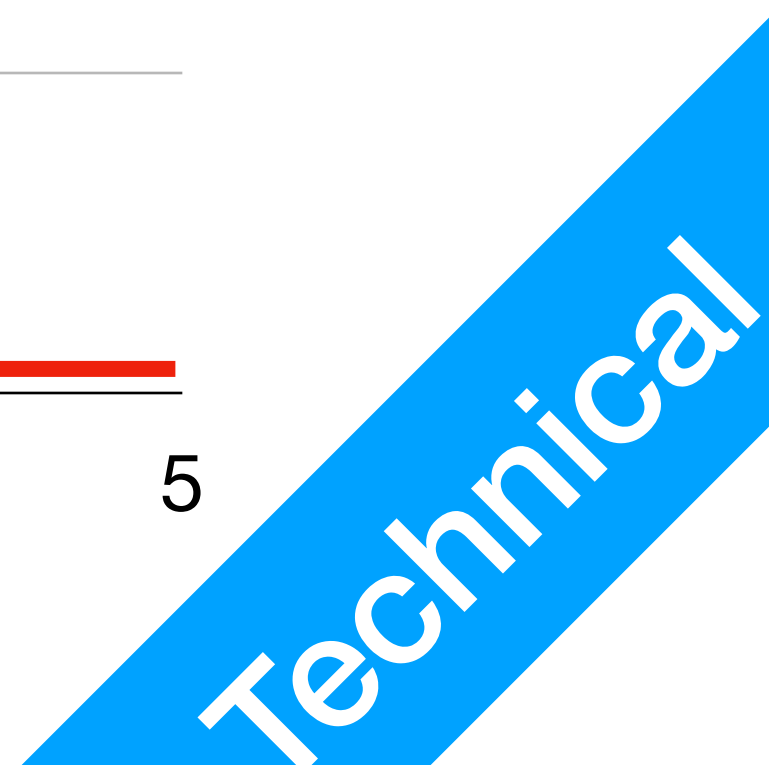
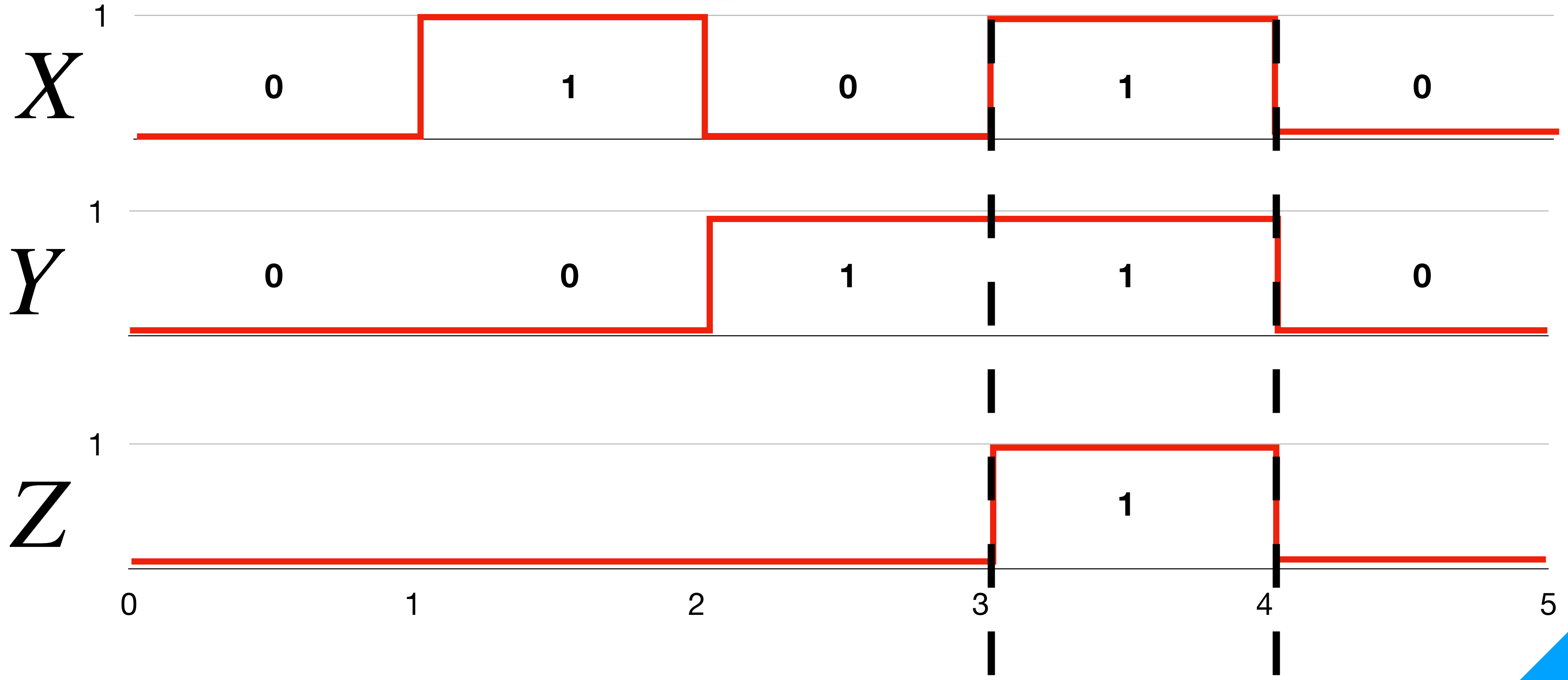
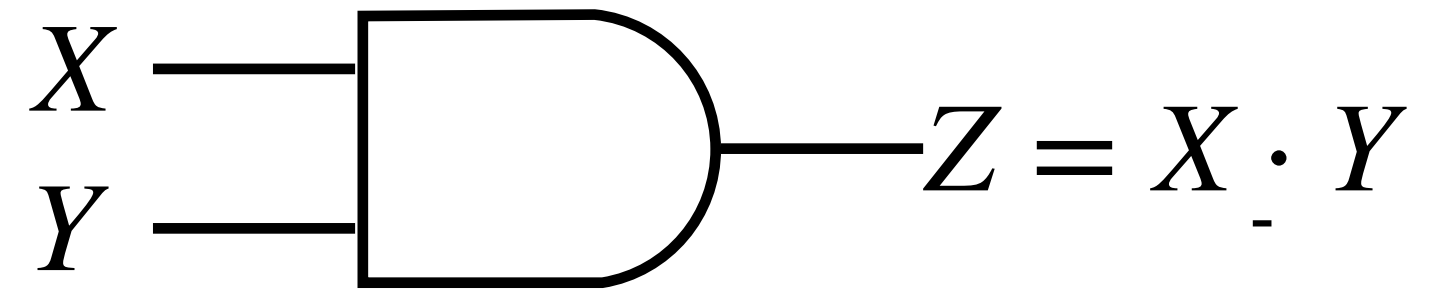
Digital Logical Gates

Timing Diagram



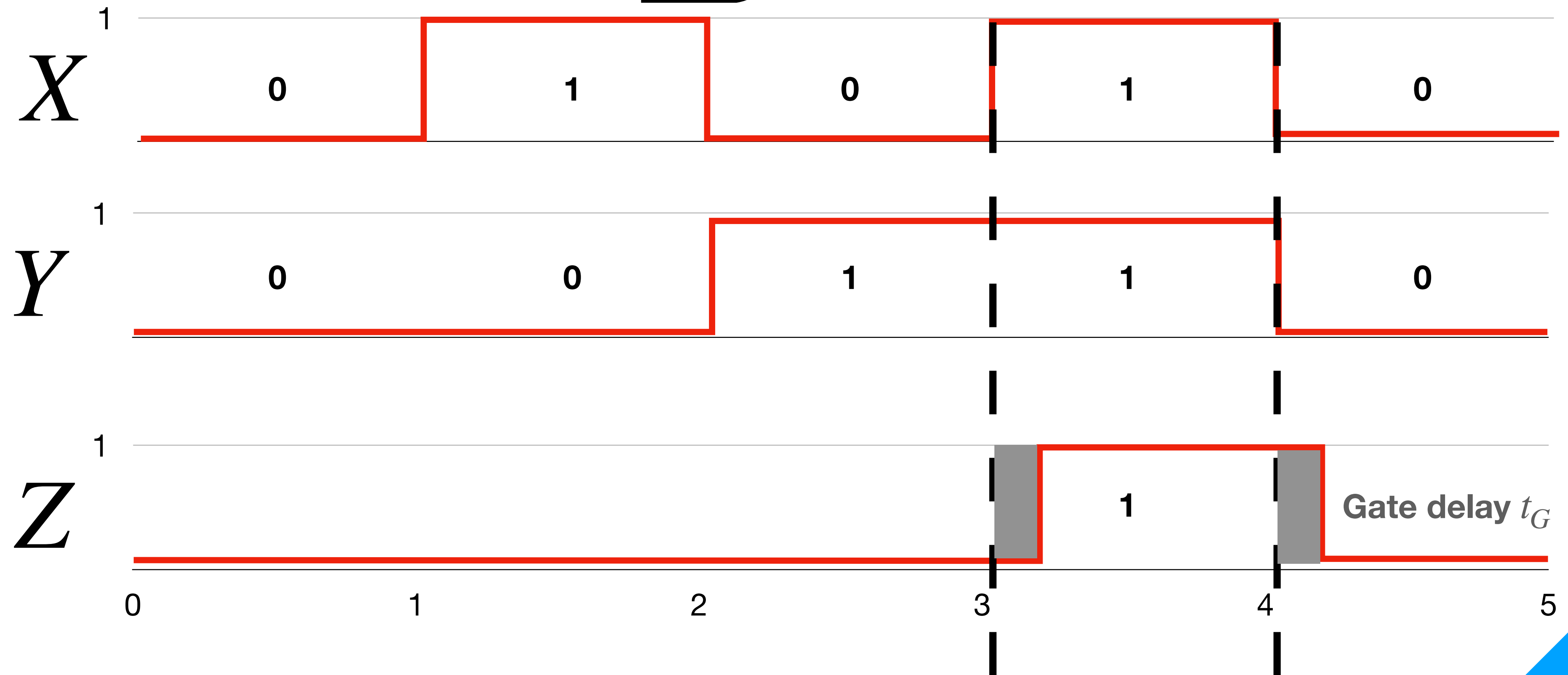
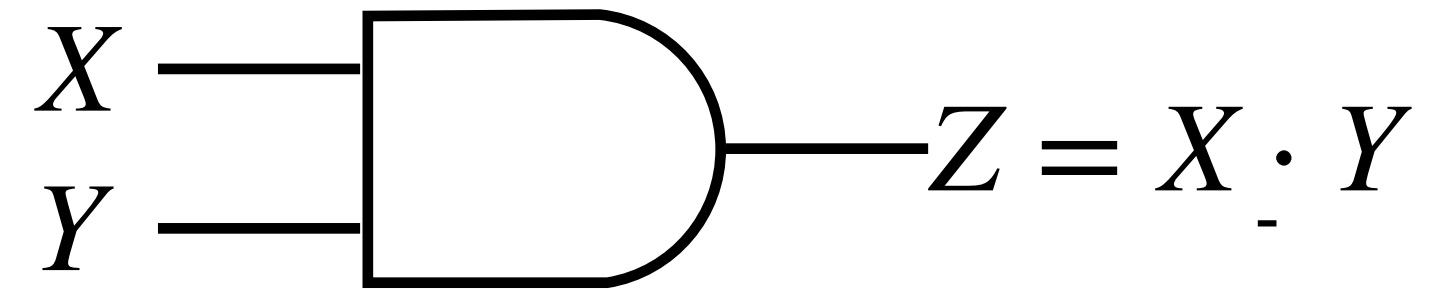
Digital Logical Gates

Timing Diagram (**IDEAL**)



Digital Logical Gates

Timing Diagram (**REALITY**)



Technical

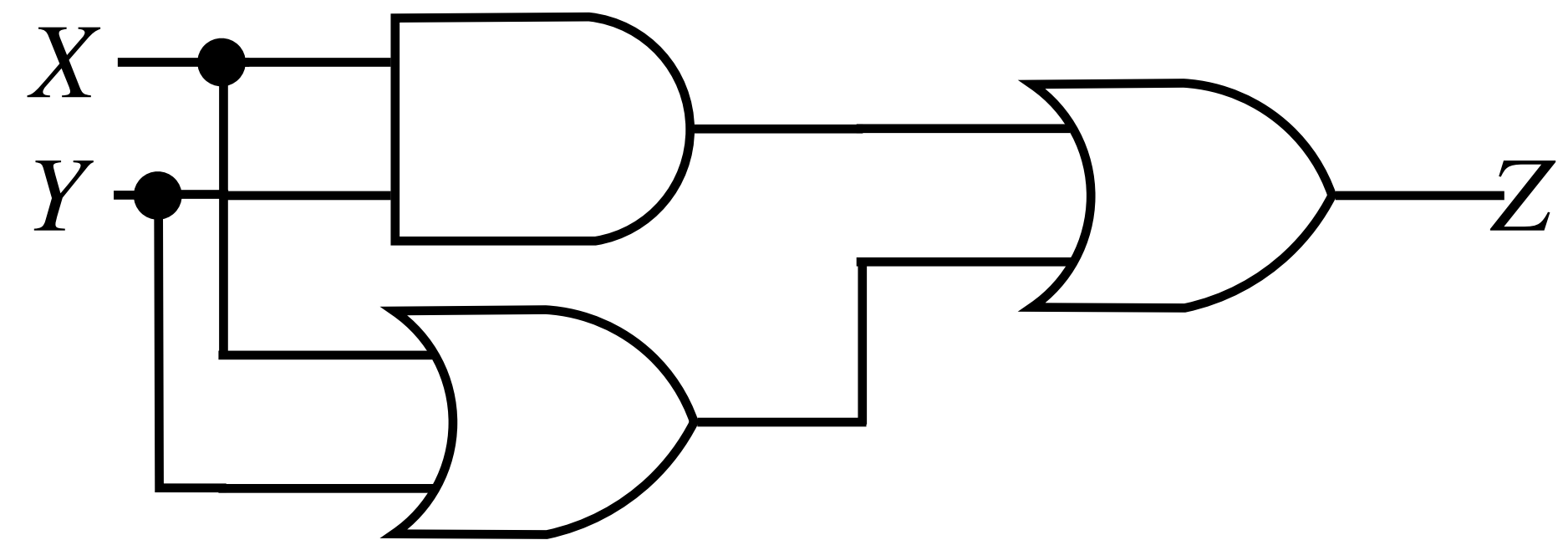
Gate Delay

- Gate delay are small, but not ignorable in practice
 - for simulation, you can ignore it for now
- Gate delay differs for different types and implementations of Gates

Simulation 1

Truth Table

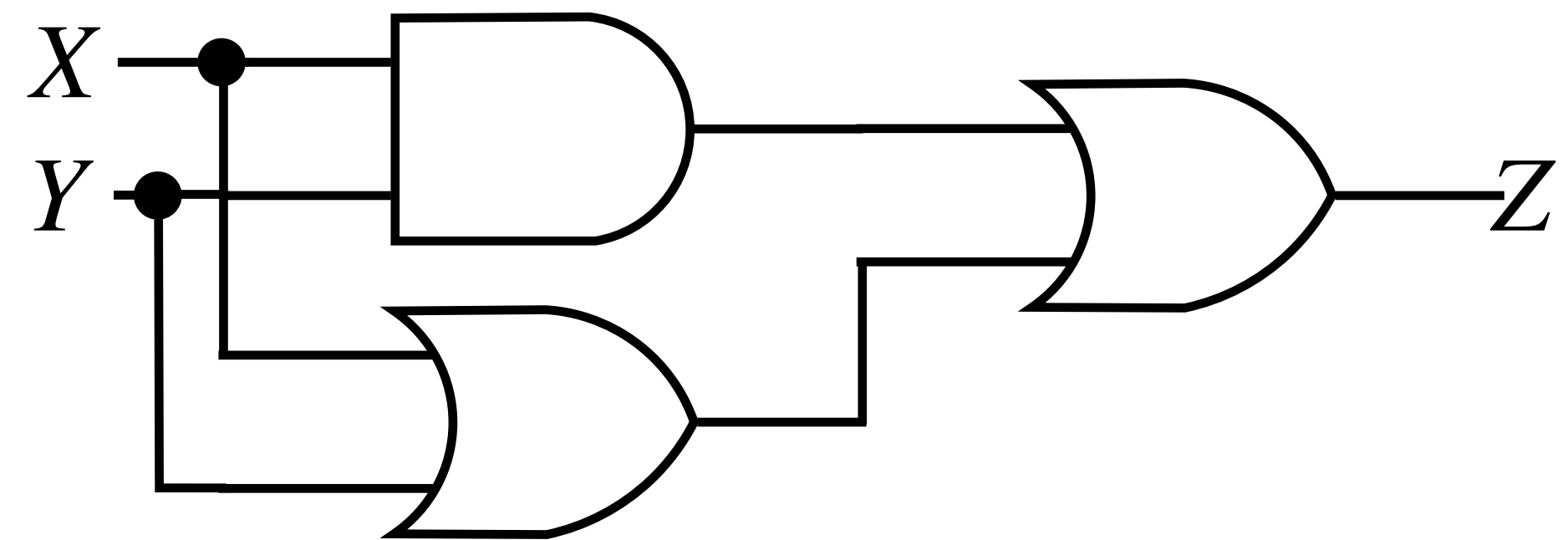
| X | Y | $Z = (X \cdot Y) + (X + Y)$ |
|-----|-----|-----------------------------|
| 0 | 0 | |
| 0 | 1 | |
| 1 | 0 | |
| 1 | 1 | |



Simulation 1

Truth Table

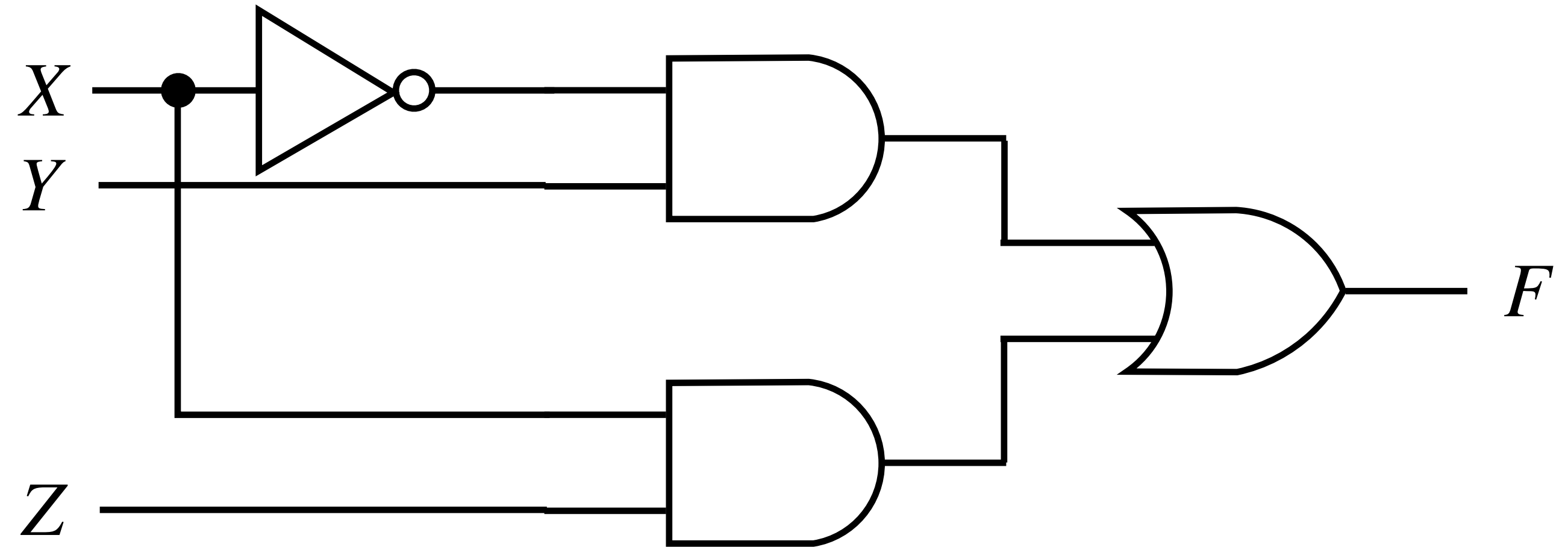
| X | Y | $Z = (X \cdot Y) + (X + Y)$ |
|-----|-----|-----------------------------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |



Simulation 2

Truth Table

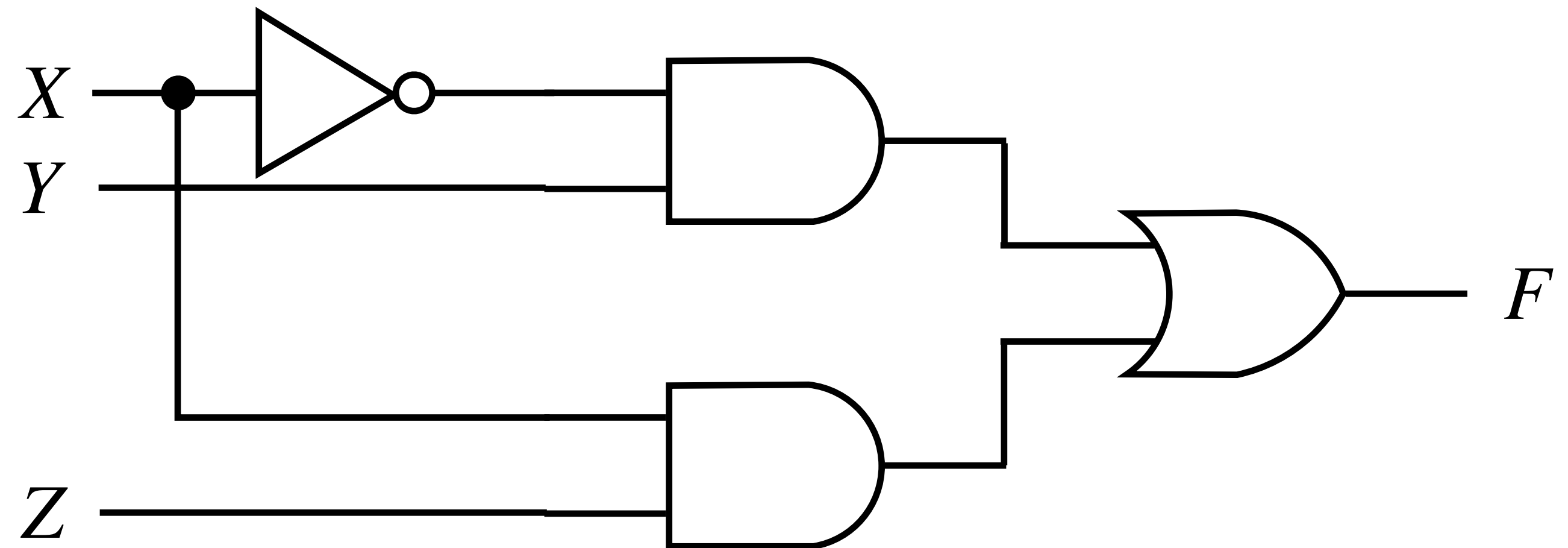
| X | Y | Z | $F = (\bar{X}Y) + (XZ)$ |
|-----|-----|-----|-------------------------|
| 0 | 0 | 0 | |
| 0 | 1 | 0 | |
| 1 | 0 | 0 | |
| 1 | 1 | 0 | |
| 0 | 0 | 1 | |
| 0 | 1 | 1 | |
| 1 | 0 | 1 | |
| 1 | 1 | 1 | |



Simulation 2

Truth Table

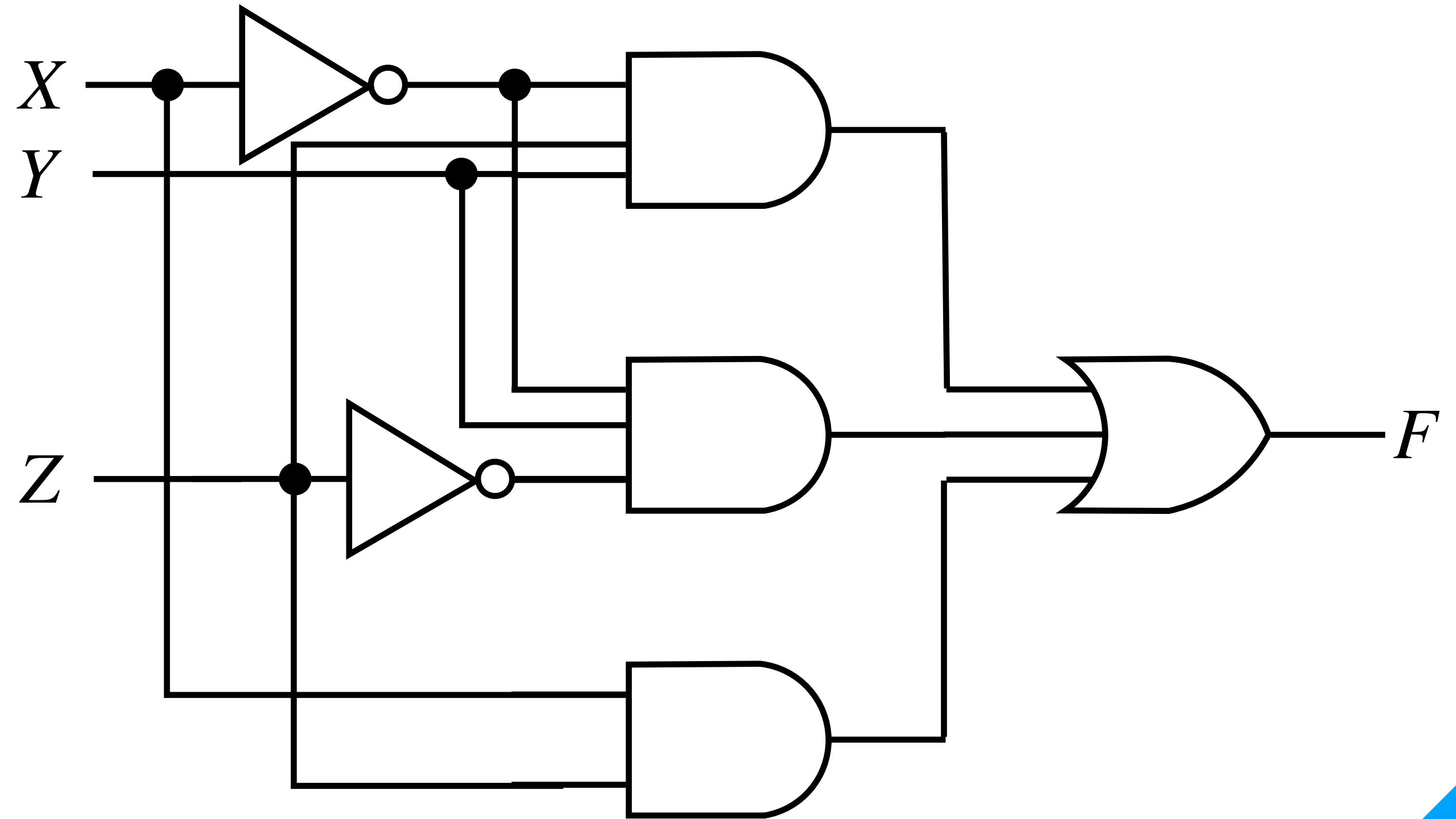
| X | Y | Z | $F = (\bar{X}Y) + (XZ)$ |
|-----|-----|-----|-------------------------|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 |



Simulation 3

Truth Table

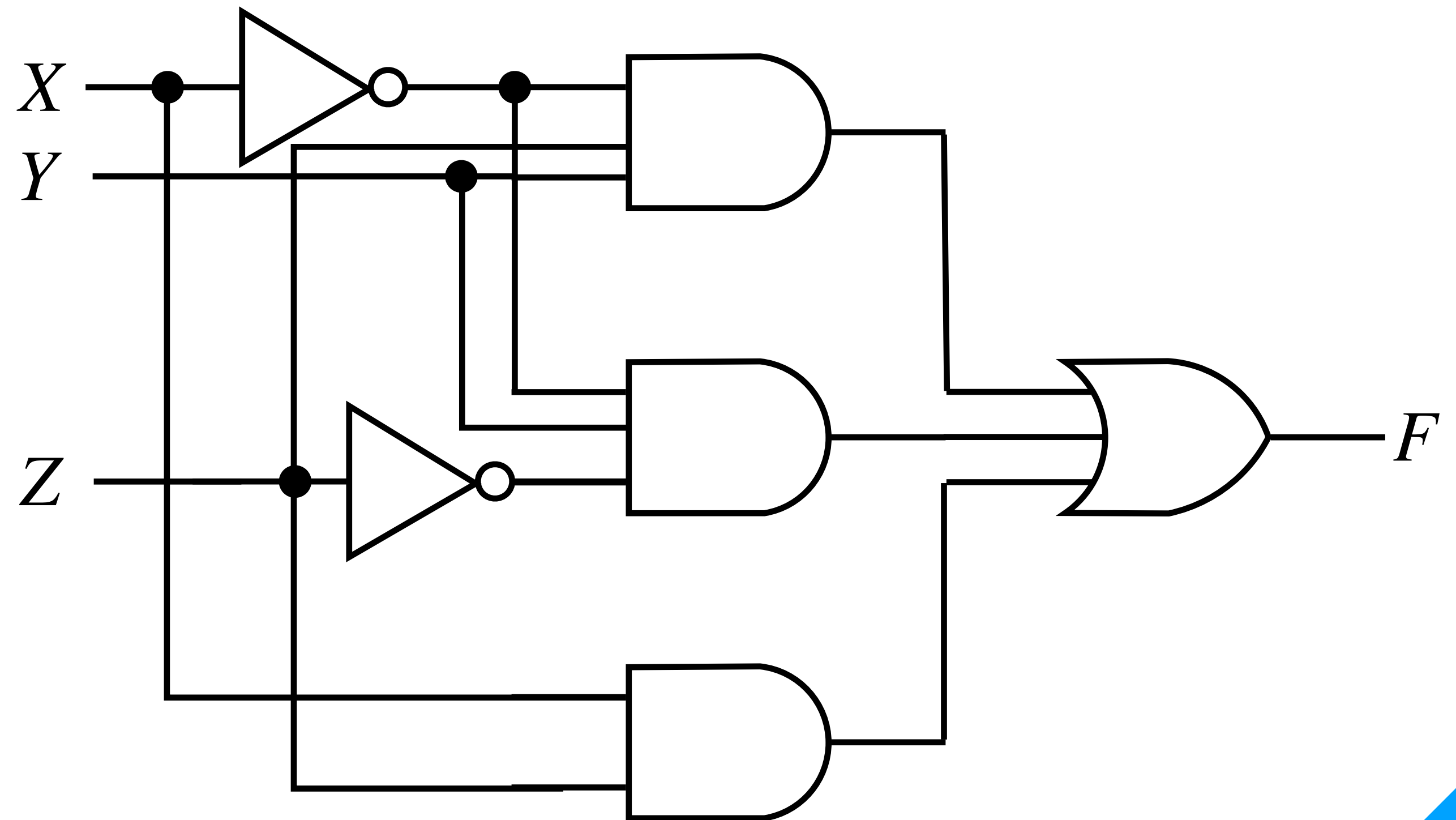
| X | Y | Z | $F = \bar{X}YZ + \bar{X}Y\bar{Z} + XZ$ |
|-----|-----|-----|--|
| 0 | 0 | 0 | |
| 0 | 1 | 0 | |
| 1 | 0 | 0 | |
| 1 | 1 | 0 | |
| 0 | 0 | 1 | |
| 0 | 1 | 1 | |
| 1 | 0 | 1 | |
| 1 | 1 | 1 | |



Simulation 3

Truth Table

| X | Y | Z | $F = \bar{X}YZ + \bar{X}Y\bar{Z} + XZ$ |
|-----|-----|-----|--|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 |

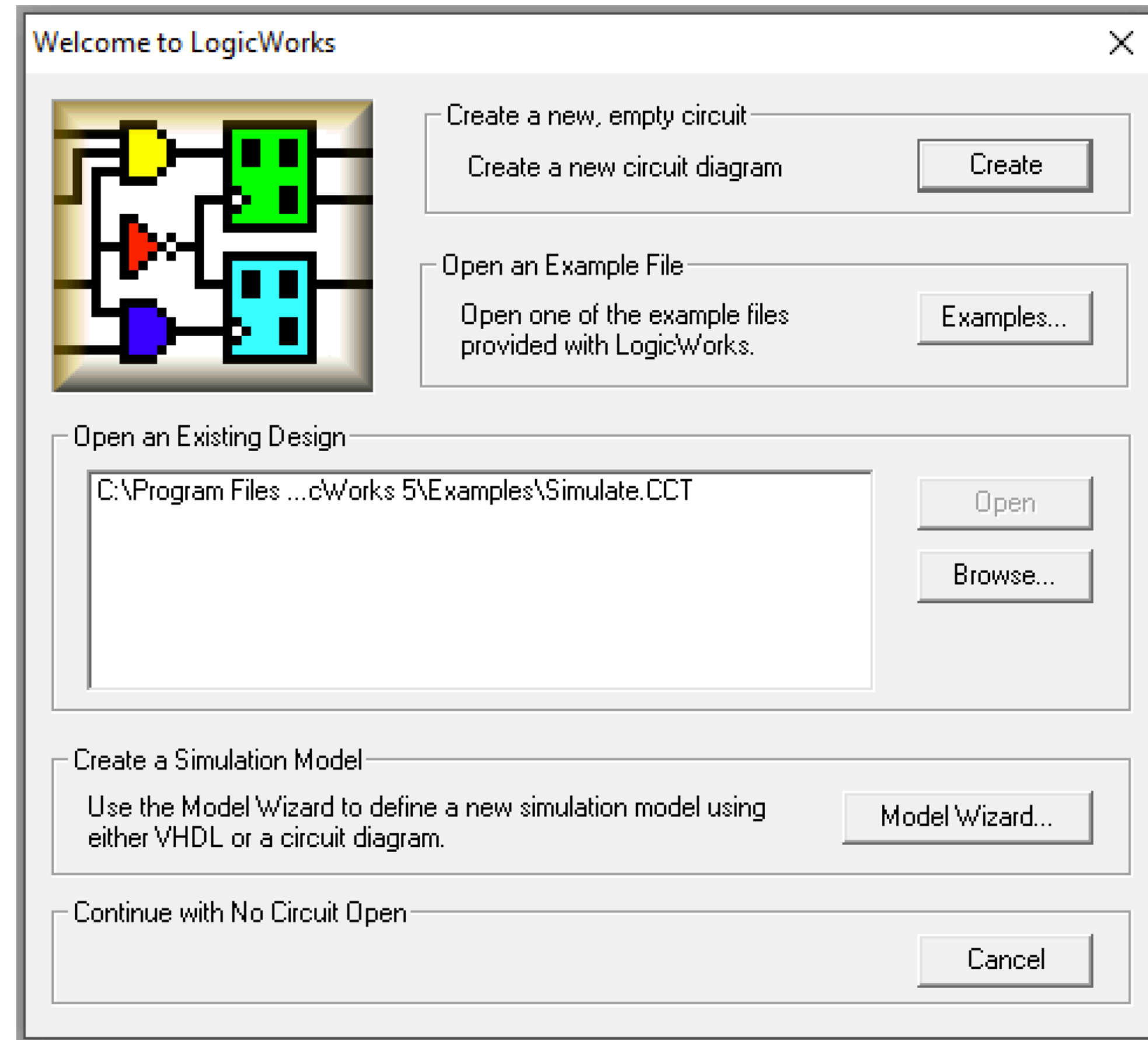


Summary

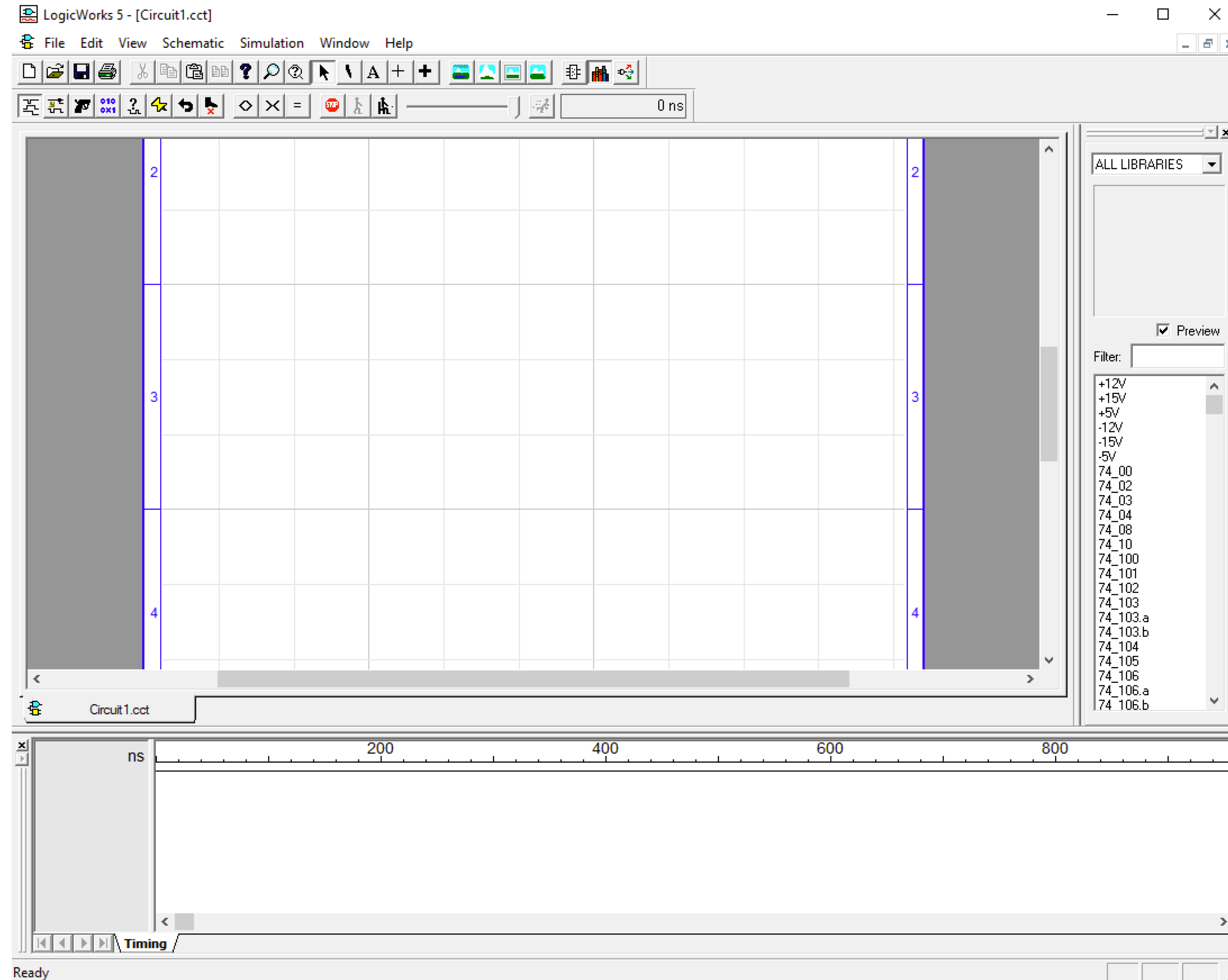
- AND, OR, NOT Operators
- Logic Gates
- Timing diagram
- Truth Table
- Gate Delay

LogicWorks

Fire up your computer please!



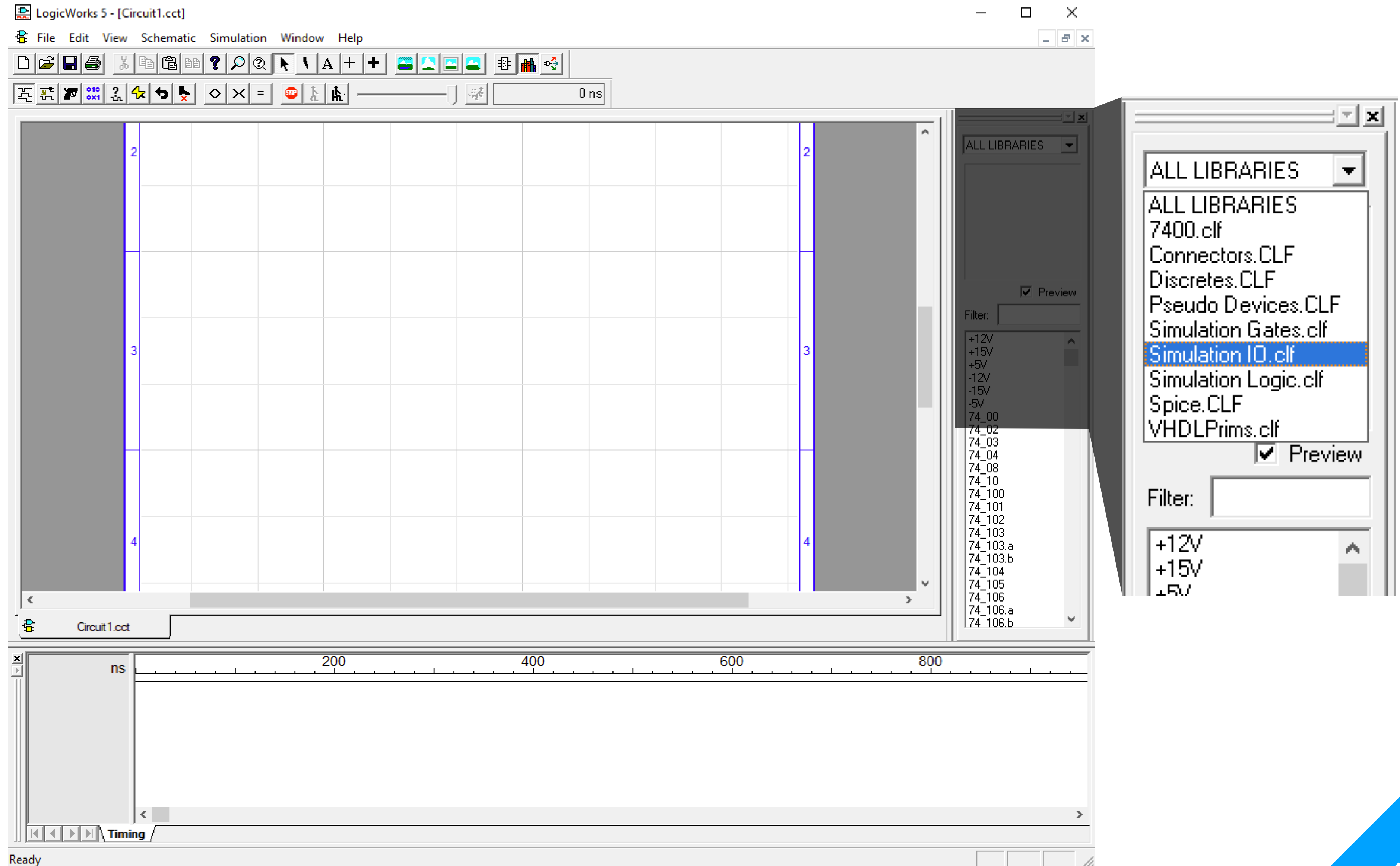
P2 Tutorial



1. This is the main interface

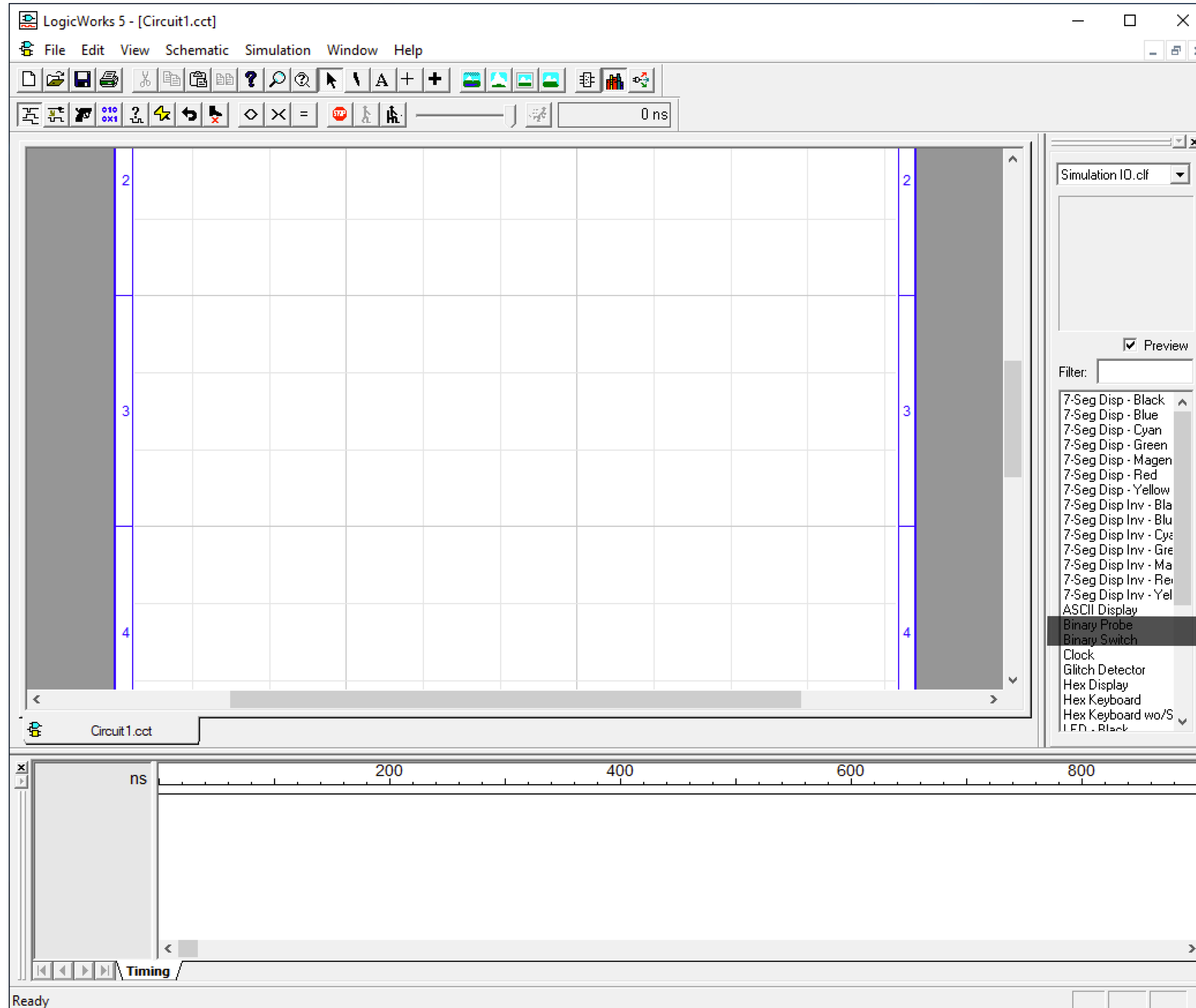
Technical

P2
Tutorial



1. Select Simulation IO from the Parts Palette

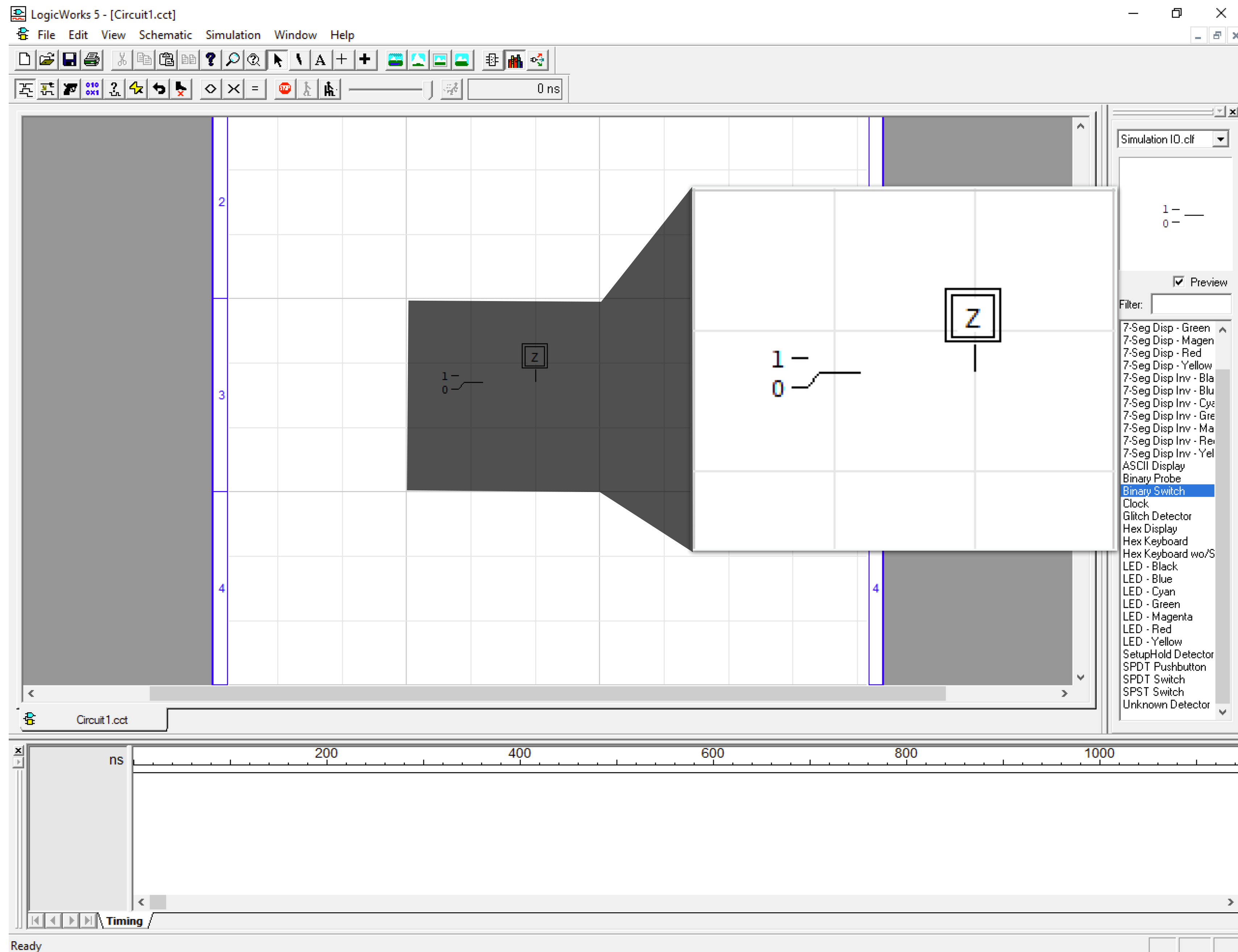
Technical



Binary Probe
Binary Switch

Technical

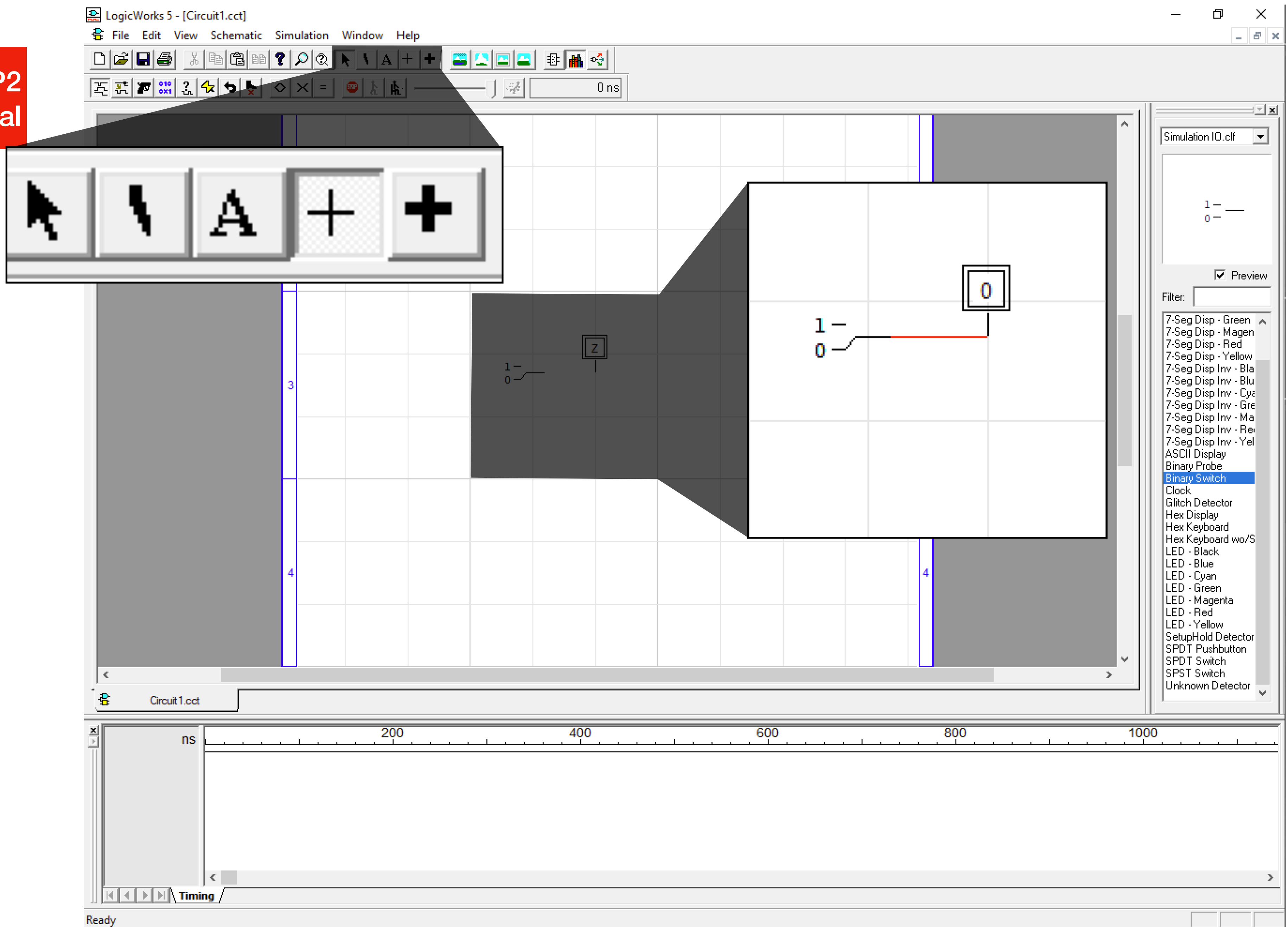
P2
Tutorial



Technical

1. Double click and place one of each on the main board

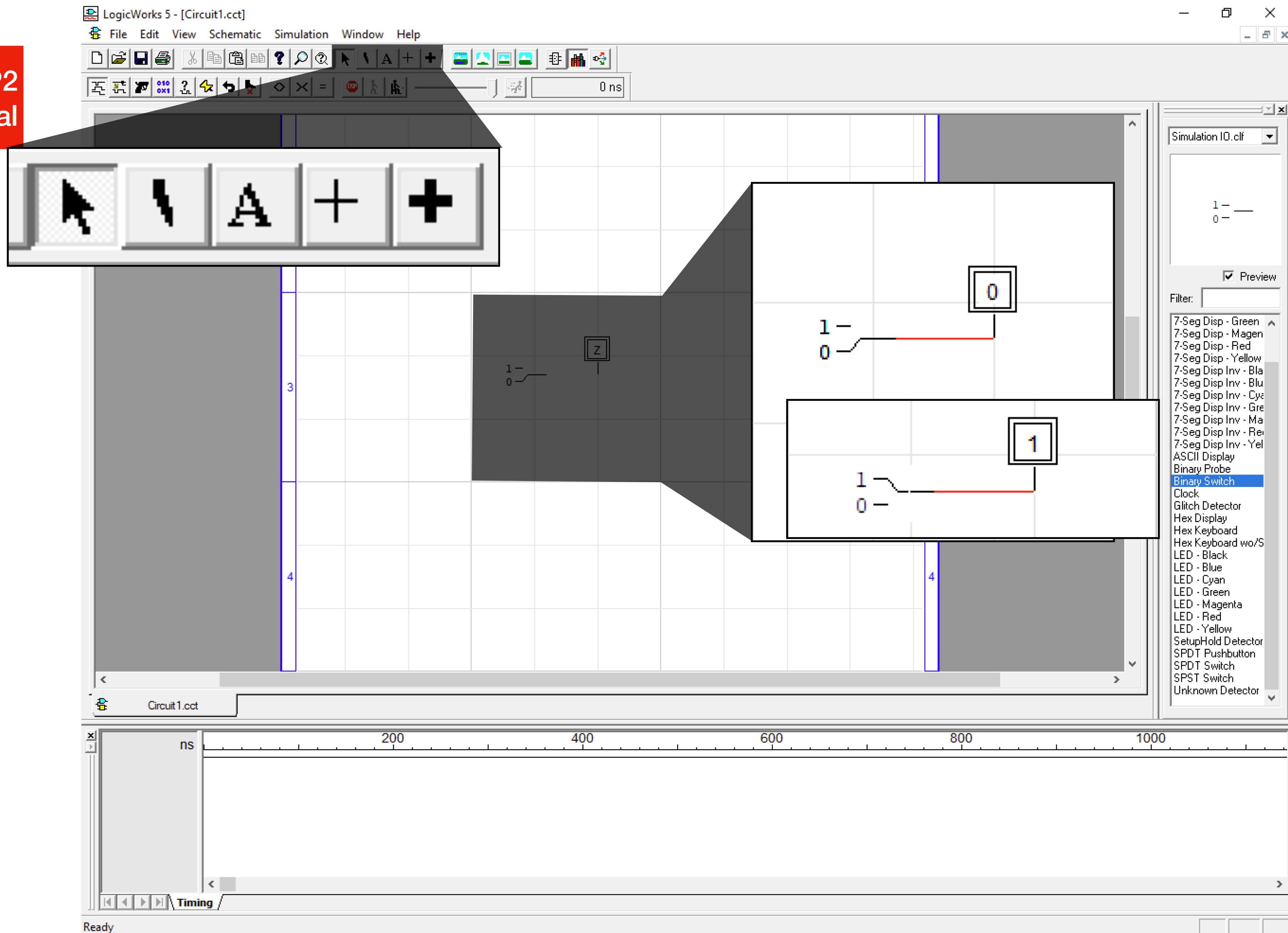
P2
Tutorial



1. Use the 'Draw Signal' tool to connect the two parts

Technical

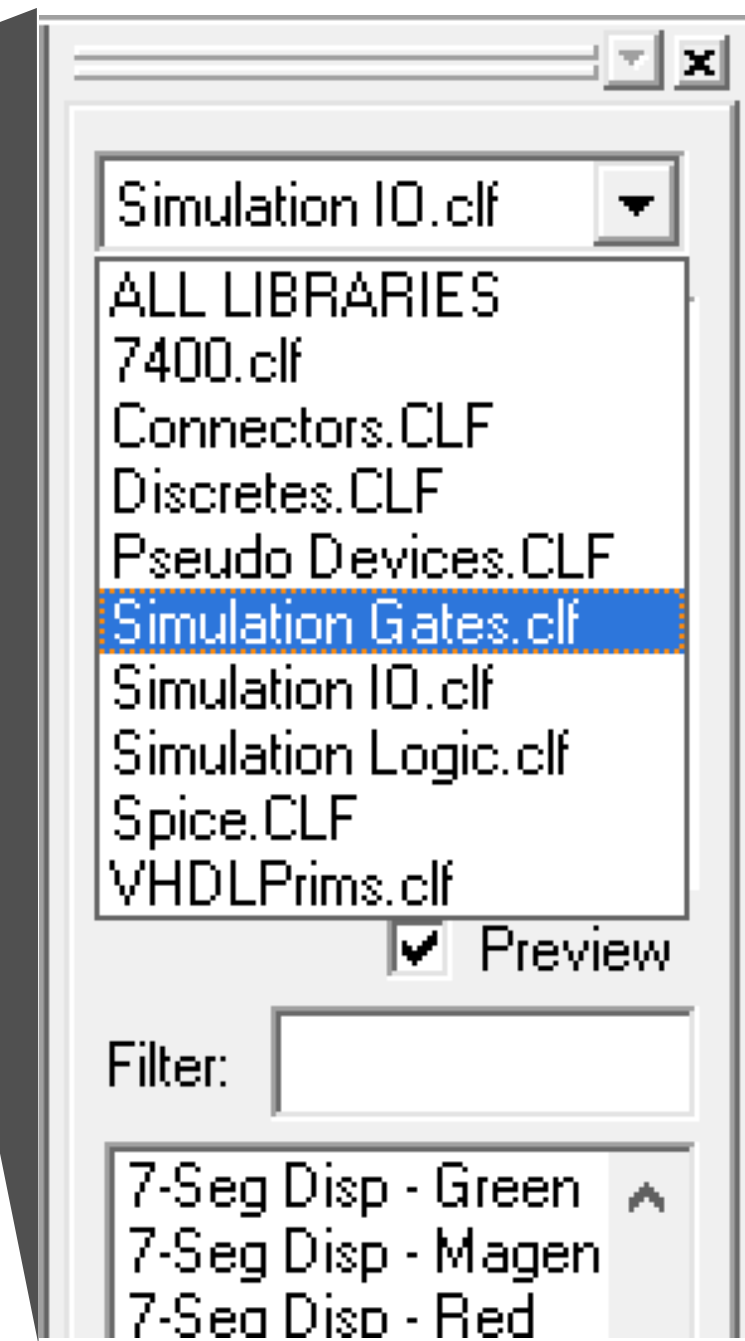
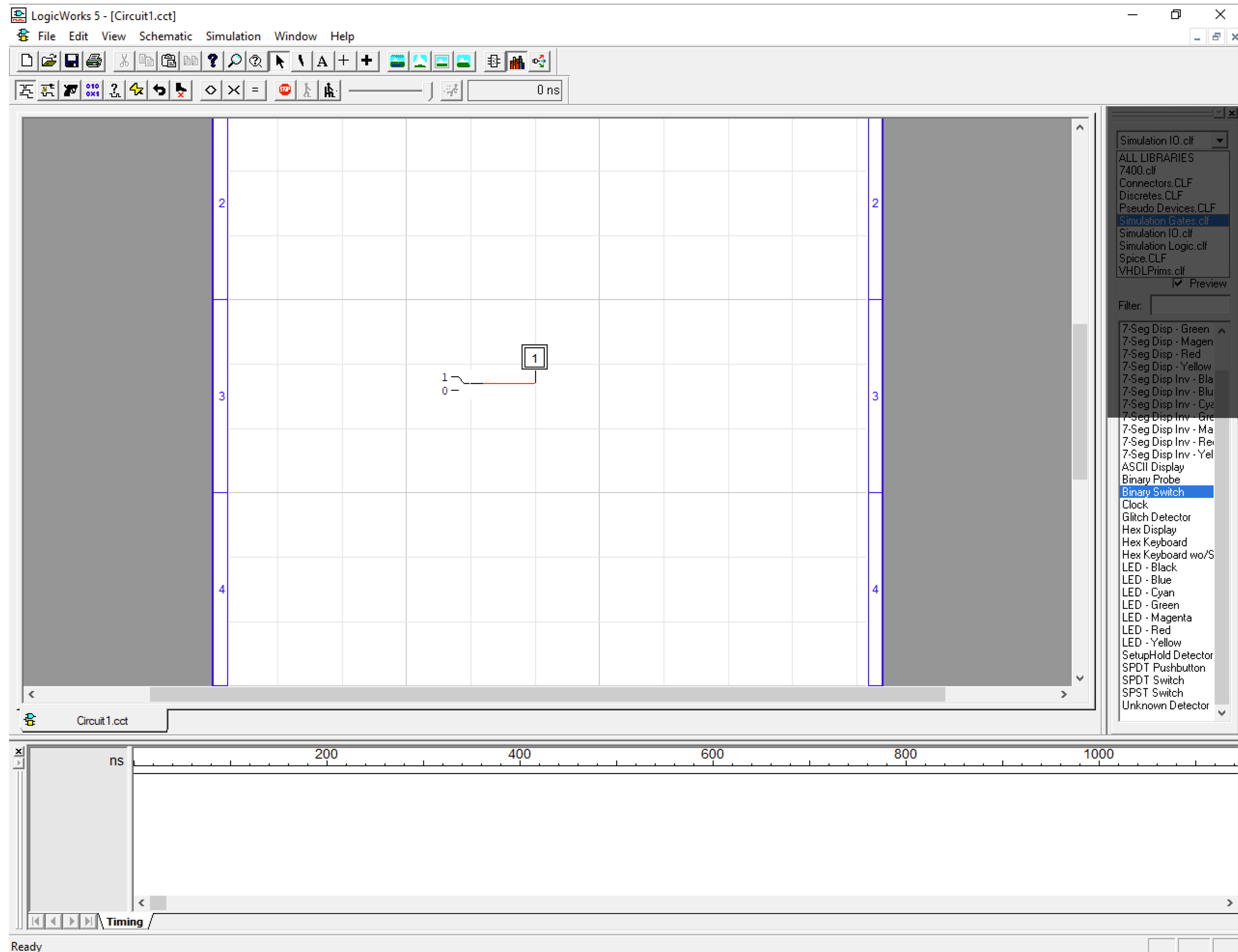
P2
Tutorial



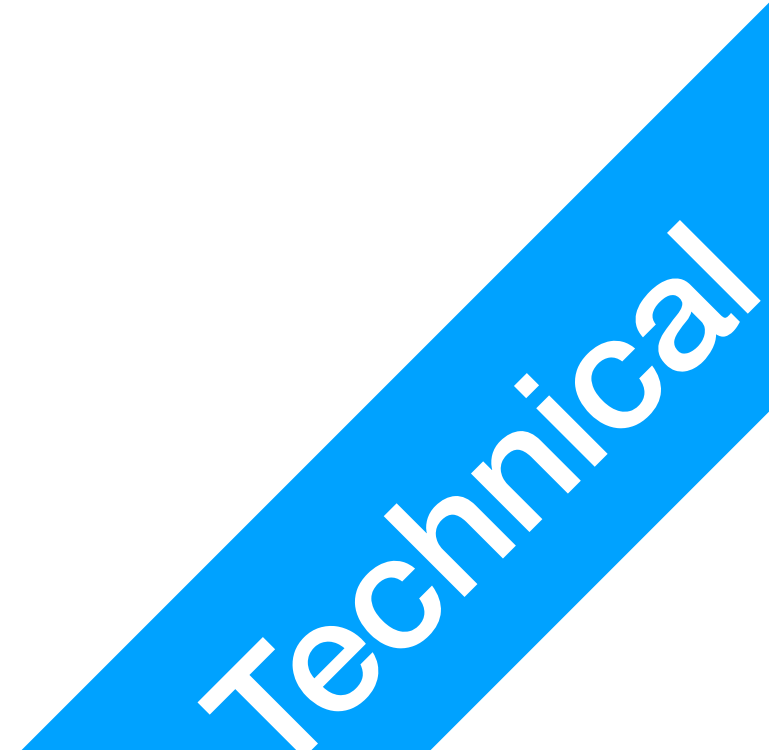
1. Switch back to the **Cursor** mode, now click on the switch and you can change the signal value

Technical

P2
Tutorial



1. Select Simulation Gates



P2
Tutorial

The screenshot displays the LogicWorks 5 interface. The main workspace shows a circuit diagram with a 2-bit bus on the left, a 2-bit bus on the right, and a 4-bit bus at the bottom. The circuit includes a 2-to-1 multiplexer, an AND gate, and two output indicators labeled '1' and '0'. The AND gate's inputs are connected to the outputs of the multiplexer. The output of the AND gate is connected to the indicator labeled '0'. The indicator labeled '1' is currently showing a '1'. The simulation window on the right shows a list of components, with 'AND-2' selected. The timing diagram at the bottom shows a signal that is high from 0 to 200 ns and low from 200 to 1000 ns.

1. Select **AND-2**, then complete the above diagram

Technical

P2
Tutorial

The screenshot displays the LogicWorks 5 interface. The main workspace shows a circuit with a clock source (square wave), an AND gate, and two binary switches. The top window shows the simulation IO component list, with 'Clock' selected. The bottom window shows a timing diagram with a time axis from 0 to 1000 ns.

1. Select **Clock** from **Simulation IO**, then replace the lower **Binary Switch** with it. A **Clock** is a device that generates a 1 at a certain frequency

Technical

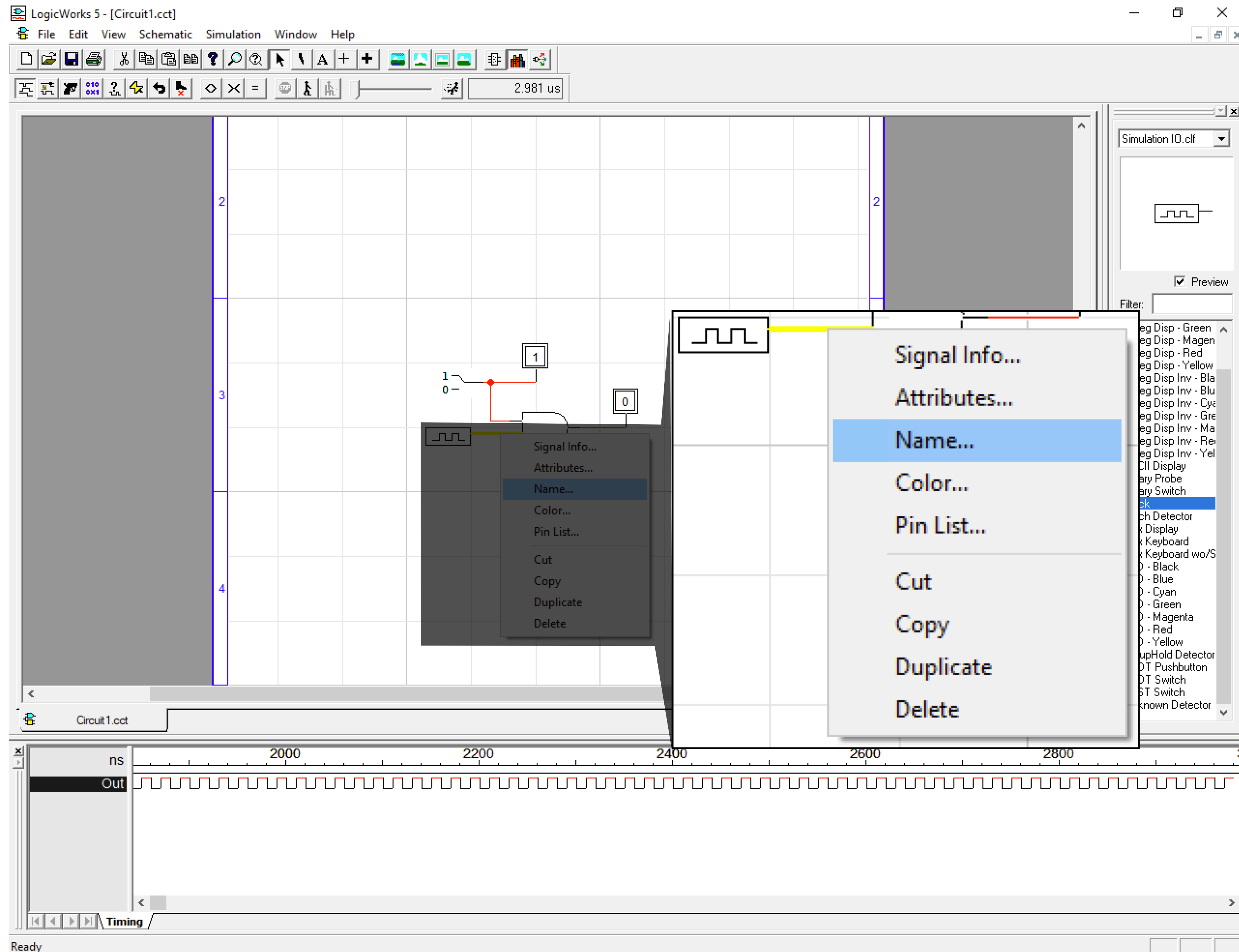
P2
Tutorial

The screenshot shows the LogicWorks 5 interface with a circuit simulation in progress. The circuit consists of a clock source (square wave), a binary switch (set to '1'), an AND gate, and two LEDs. The simulation IO window is open, showing the 'Clock' component selected in the Simulation IO.clf window. The Simulation Gates.clf window shows a list of components, with 'Clock' highlighted. The timing diagram at the bottom shows the clock signal and the output of the AND gate.

1. Select **Clock** from **Simulation IO**, then replace the lower **Binary Switch** with it

Technical

P2
Tutorial



1. Right click the **Red Wire**, select **Name**

Technical

P2
Tutorial

The screenshot shows the LogicWorks 5 interface. The main workspace contains a circuit diagram with two AND gates. The top gate has inputs '1' and '0' and an output '1'. The bottom gate has inputs 'In1' and 'In2' and an output 'Out'. A timing diagram at the bottom shows waveforms for In1, In2, and Out. The Out signal is high only when both In1 and In2 are high. A toolbar with various icons is visible at the top left, and a component list is on the right side.

1. Complete the diagram like above, then click **Reset Simulation**

Technical

P2
Tutorial

The screenshot shows the LogicWorks 5 interface. At the top, the menu bar includes File, Edit, View, Schematic, Simulation, Window, and Help. Below the menu is a toolbar with various icons. The main workspace displays a schematic of an AND gate with two inputs labeled 'In1' and 'In2', and an output labeled 'Out'. A simulation panel is overlaid on the schematic, showing a stop button, a play button, a slider, and a digital display showing '190 ns'. Below the schematic is a large timing diagram with a time axis from 0 to 400 ns. The diagram shows three signals: 'In1' (a constant high signal), 'In2' (a square wave), and 'Out' (a square wave that is high only when both In1 and In2 are high). A smaller timing diagram is visible in the bottom-left corner. The right sidebar contains a component library with 'Clock' selected.

Technical

1. Use the Simulation Panel to control the speed of simulation, then you will see the Timing Diagram!

Exe 1

- Curtain Motor Control
 - Button1: 1 when user wants to open the curtain
 - Button2: 1 when user wants to close the curtain
 - Output1: 1 to make the motor open the curtain
 - Output2: 1 to make the motor close the curtain
 - Light: motor is active



- When both buttons are pressed, motors do nothing

Exe 2

- Curtain Motor Control
 - **Sensor1: 1 when curtain is fully closed**
 - **Sensor2: 1 when curtain is fully open**
 - Button1: 1 when user wants to open the curtain
 - Button2: 1 when user wants to close the curtain
 - Output1: 1 to make the motor open the curtain
 - Output2: 1 to make the motor close the curtain
 - Light: motor is active



- Stop the motor when the curtain is already fully opened/
closed

Summary

- Simulation in LogicWorks
- Binary Probe / Binary Switch in LogicWorks
- Gates in LogicWorks
- Clocks in LogicWorks
- Timing Diagram in LogicWorks