|  |  |  |
| --- | --- | --- |
| **American University of Sharjah**  **College of Engineering**  Dept of Computer Science & Engg  P. O. Box 26666  Sharjah, UAE | A picture containing logo  Description automatically generated | **Lab Instructor:** Eng. Donthi Sankalpa  **Office:** ESB – 1036C  **Phone**: 971-6-5154826  **e-mail**: [dsankalpa@aus.edu](mailto:dsankalpa@aus.edu)  **Semester**: Fall 2023 |

Lab #1 – Introduction to Logic Gates

**Objectives**

* Familiarization with the LogiSim environment.
* Use of switches as inputs and LEDs as outputs.
* Understand and test operations of basic logic gates.
* Determine experimentally the truth tables for the AND, OR, and inverter gates.
* Interpret the Data Sheet of the used chips

**Lab Equipment and Circuit Components**

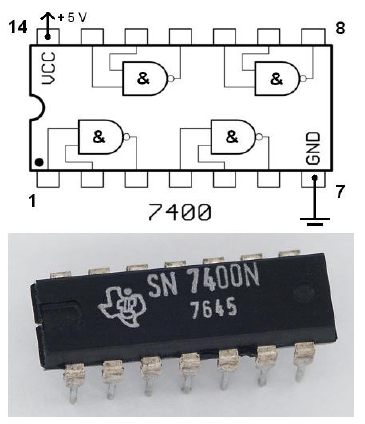
* 1. **Equipment** 
     + LogiSim

Figure 1. 7400 NAND DIP

* 1. **Circuit Components**
     + 7404 inverter Chip
     + 7408 AND
     + 7432 OR

**Introduction:**

***Logic gates and Integrated Circuits:*** The 7400 family of Integrated Circuits (ICs) will be used throughout this course. The chips come in various packages, but the package used in these labs is a dual in-line package, otherwise known as a DIP as shown in Figure 1. In order to determine the polarity of the chip, a notch is put on one side of the chip. From a top view, pin one is on the left of the notch with other pins numbered sequentially in a counterclockwise manner.

The 7400 series is implemented using Transistor–transistor logic (TTL), which uses bipolar transistors (BJT) to form its integrated circuits. The 7400 family has many subfamilies made for special considerations, referred to by letters written after the family name; such as 74LS08, these subfamilies are summarized in Table 1.

|  |  |
| --- | --- |
| **Family** | **Meaning** |
| 74 – | Standard TTL |
| 74L | Low-power |
| 74H | High speed |
| 74S | High-speed Schottky |
| 74LS | Low-power Schottky |
| **Table 1. 7400 Families** | |

Figure 2 shows the internal logic of the 7404, 7408, 7432 chips.

***Breadboard:*** In order to build the circuit, a digital design kit that contains a power supply, switches for input, light emitting diodes (LEDs), and a breadboard will be used. Figure 3.a shows a common breadboard, while Figure 3.b

shows how each set of pins are tied together electronically. Watch this [video](https://youtu.be/6WReFkfrUIk) to understand how a breadboard works.

|  |  |  |
| --- | --- | --- |
| Image result |  |  |
| 7404 hex inverters | 7408 Quad 2-input AND Gates | 7432 Quad 2-input OR Gates |
| **Figure 2. Connection diagram of 7404, 7408m and 7432** | | |

|  |  |
| --- | --- |
|  |  |
| **Figure 3.a: Breadboard** | **Figure 3.b Common connections** |

**LogiSim:**

Logisim is an educational tool for designing and simulating digital logic circuits. With its simple toolbar interface and simulation of circuits as you build them, it is simple enough to facilitate learning the most basic concepts related to logic circuits. With the capacity to build larger circuits from smaller subcircuits, and to draw bundles of wires with a single mouse drag, Logisim can be used (and is used) to design and simulate entire CPUs for educational purposes.

Logisim is used by students at colleges and universities around the world in many types of classes, ranging from a brief unit on logic in general-education computer science surveys, to computer organization courses, to full-semester courses on computer architecture.

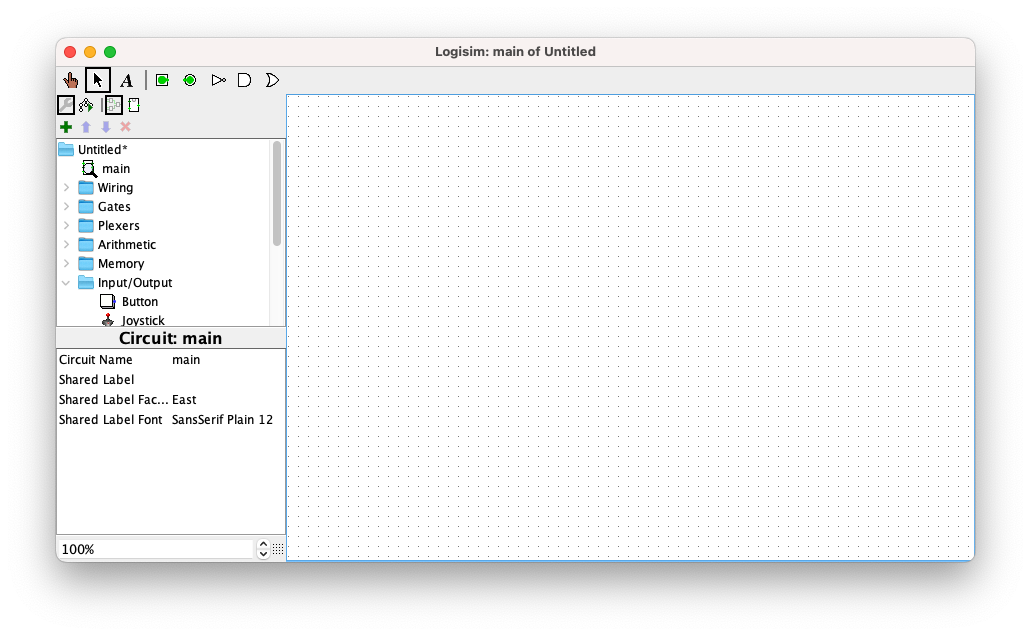
**LogiSim Environment:**

1. If you don’t have it already, you can download it from the following link:

[https://drive.google.com/file/d/1Qeo63cYMRT95W1s6hUoM3bvHM0V1BAnL/view?usp=shari](https://drive.google.com/file/d/1Qeo63cYMRT95W1s6hUoM3bvHM0V1BAnL/view?usp=sharing  )

[ng](https://drive.google.com/file/d/1Qeo63cYMRT95W1s6hUoM3bvHM0V1BAnL/view?usp=sharing  )

1. Extract the zipped file, you will see many files, runLogiSim will run the classic LogiSim while runLogiSimEvolution will run the modern version of LogiSim.
2. Figure 4 shows the opening window of the LogiSum 2.7 program (classic version). Some of the components of the tool bars that are important for the present course are highlighted by being labeled.



Use Arrow to place components and move them, Hand to press the switches

Circuit Design Section

Components Section

Figure 4

1. To design a simple digital circuit, choose the required component as shown in figure 5.

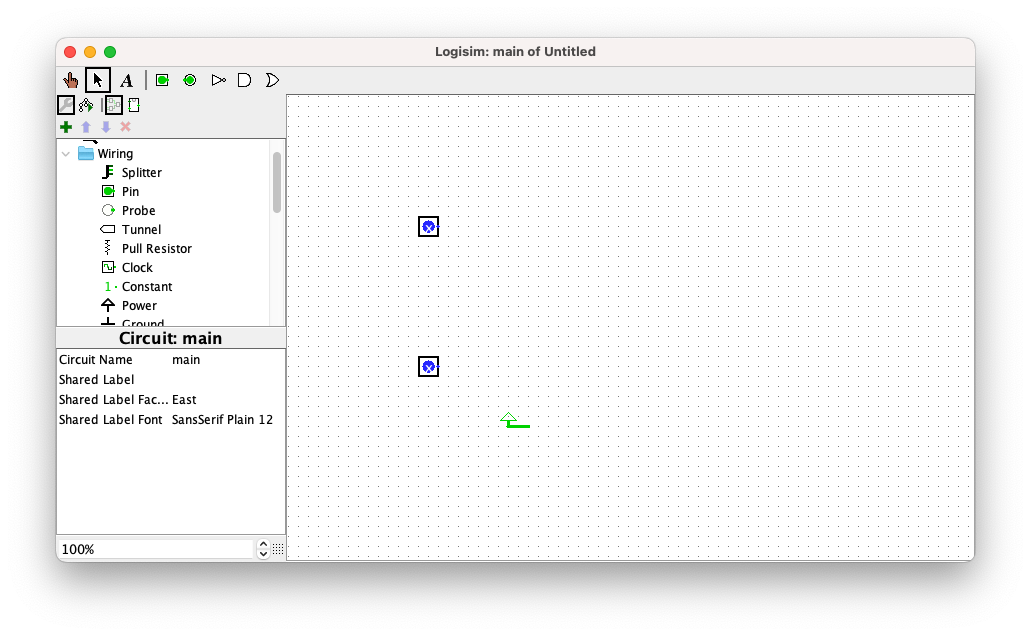


Figure 5

1. Add the LED indicator as the following:

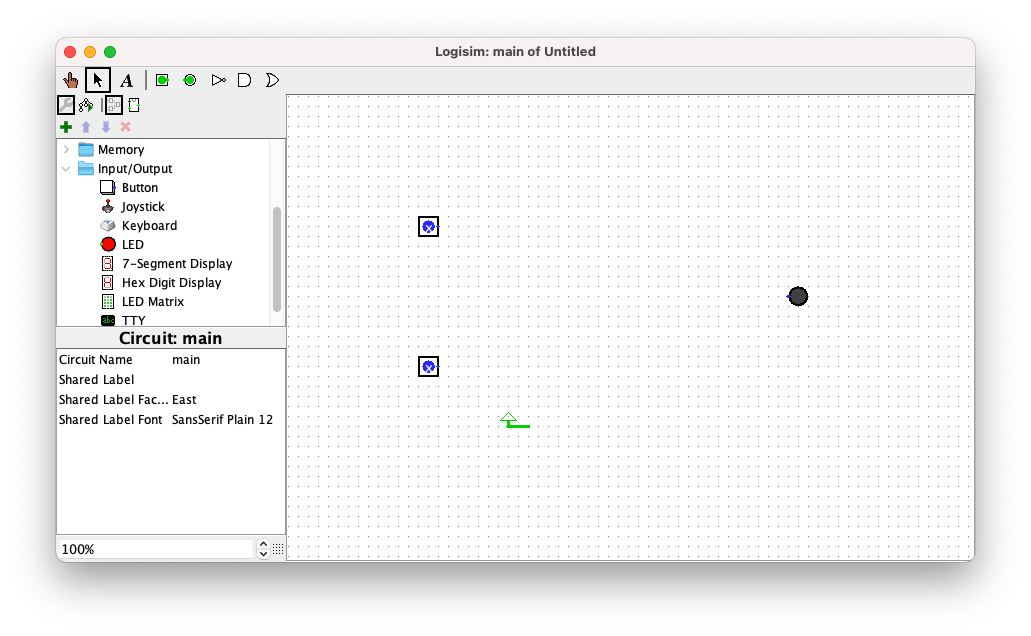


Figure 6

1. Add the other components, in this example we will add only 1 AND gate

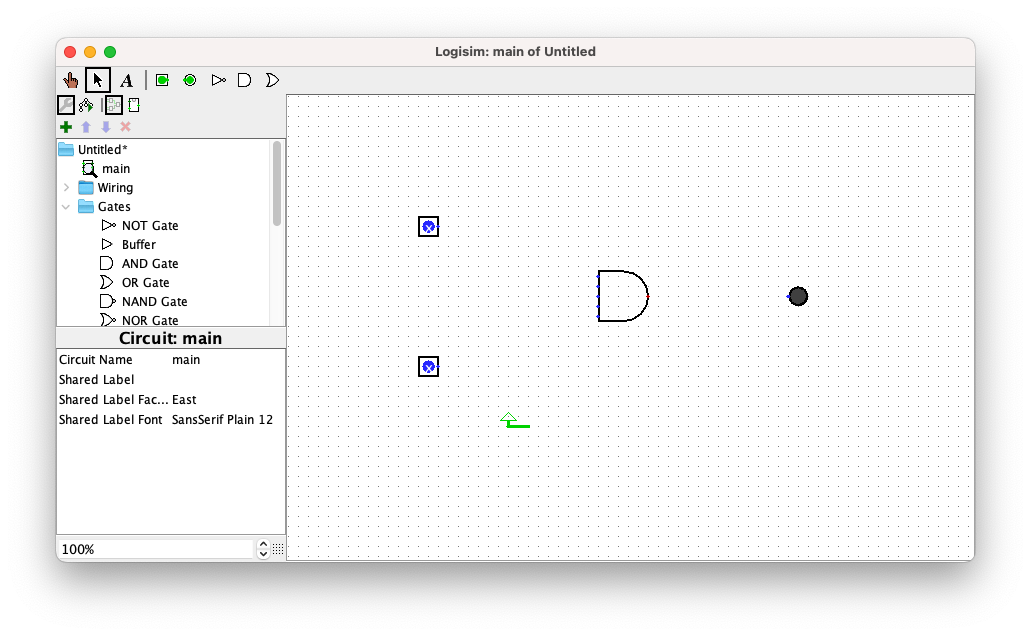


Figure 7

1. Do the wiring as the following:

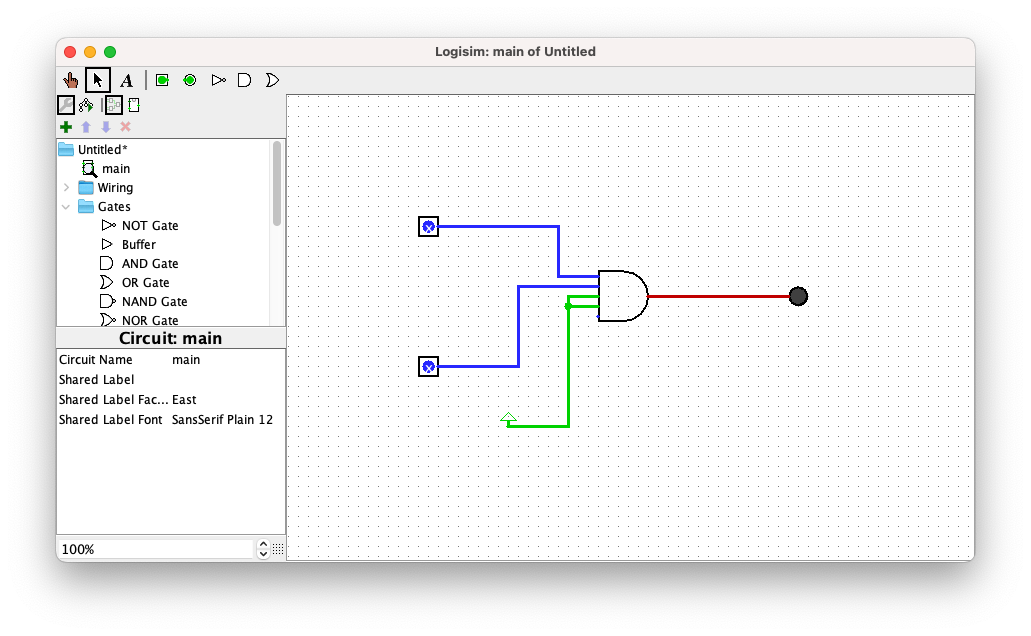
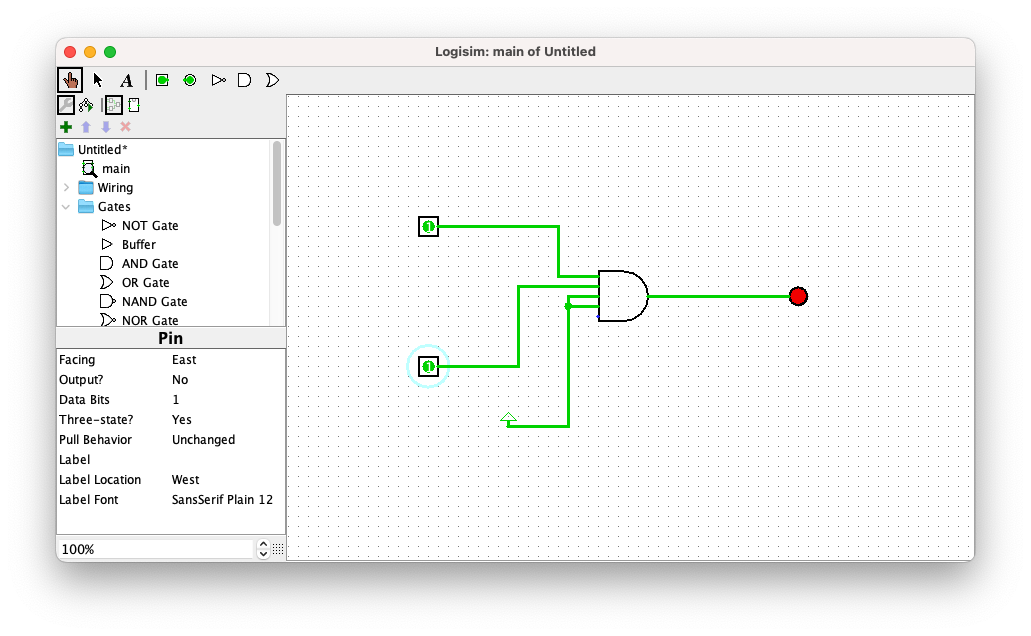


Figure 8

1. Test your circuit using the hand tool:



Initially, the button Pin has the value of X, if you click it will be 0, click it again it will be 1, and if you click it again it will be X and so on

True And True is True (The LED is on)

Hand Tool is selected

Figure 9

**Lab Activities:**

1. Complete the following timing diagram and build the truth table for inputs and outputs:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **+3.3V** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Input A** | **0v** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **+3.3V** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Input B** | **0v** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **+3.3V** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Input C** | **0v** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **+3.3V** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **A ⚫ B ⚫ C** | **0v** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **+3.3V** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **A + B + C** | **0v** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

a. When is the output of an AND gate HIGH? When all 3 inputs are high (1) .



b. When is the output of an OR gate LOW? When all 3 inputs are low (0).

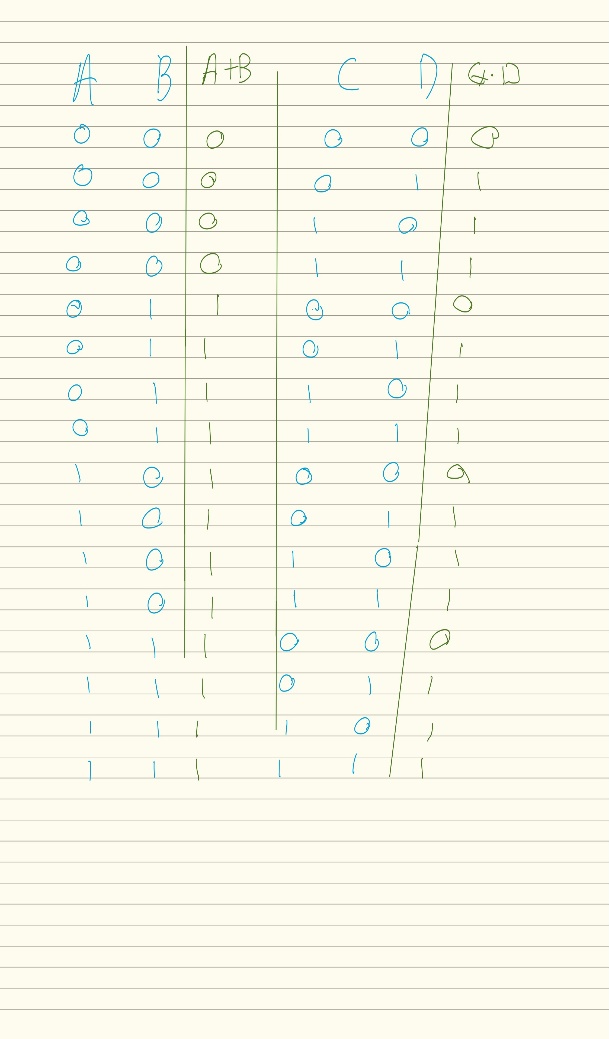
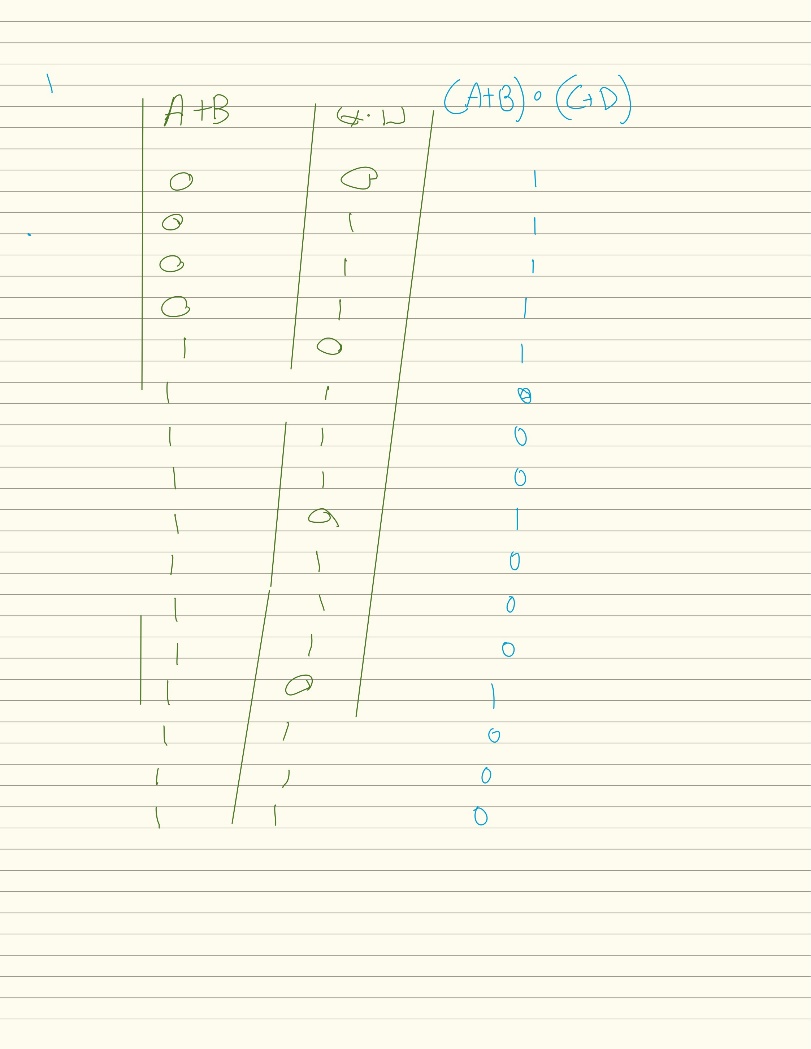


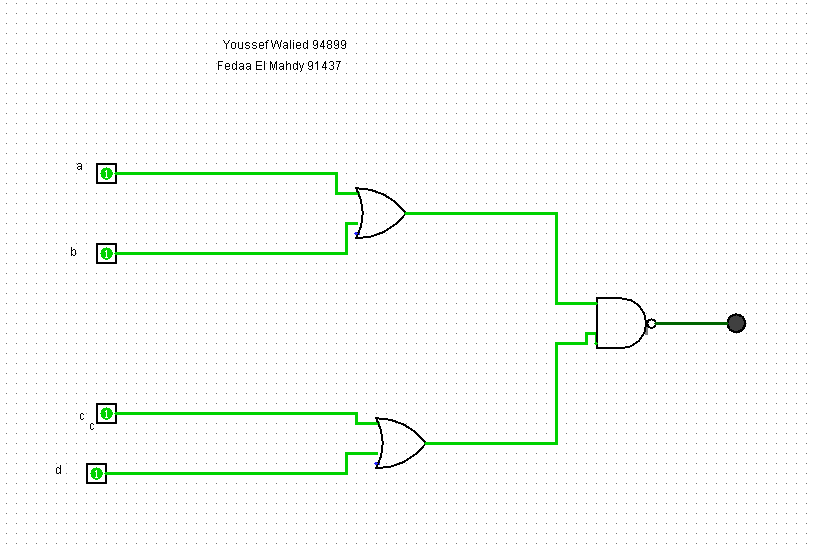
1. Given the following digital circuit: Write the Boolean expression F:

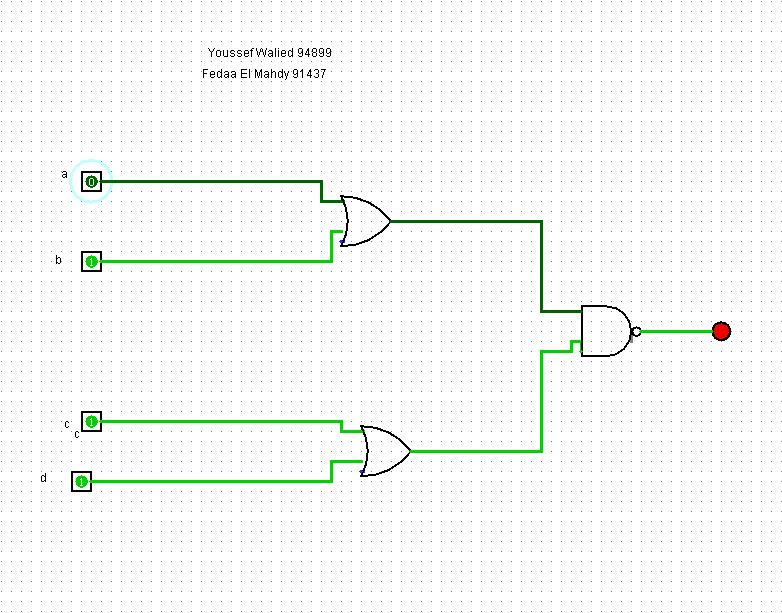
Diagram

Description automatically generated

1. Write the Boolean expression F. ((A+B) (C D))’
2. Implement the circuit and verify your answer for (2.a)







1. Draw and implement the logic circuit represented by the below expression. Write the truth table to verify your circuit.

