DIGITAL LOGIC FAMILIARIZATION LAB REPORT

NAME: Lakshmi Sai Bhargav Vemparala

ROLL NO.: 2025102061

GROUP: G9

EXPERIMENT 1 : Digital Test Kit and NOT Gate Implementation

Aim:

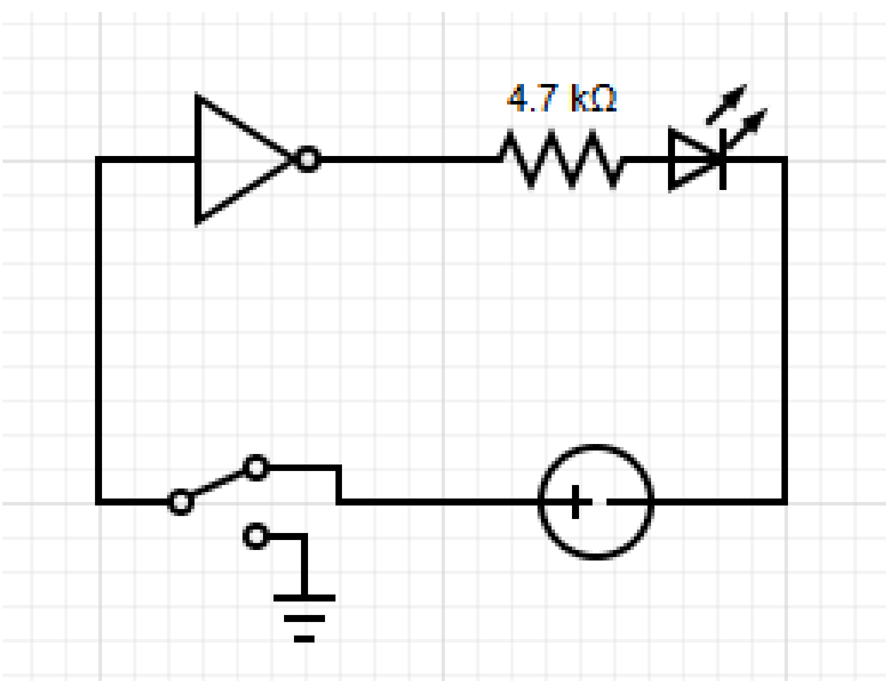
To learn the usage of the Digital Test Kit and study binary logic levels through the implementation of a NOT gate.

Components Required:

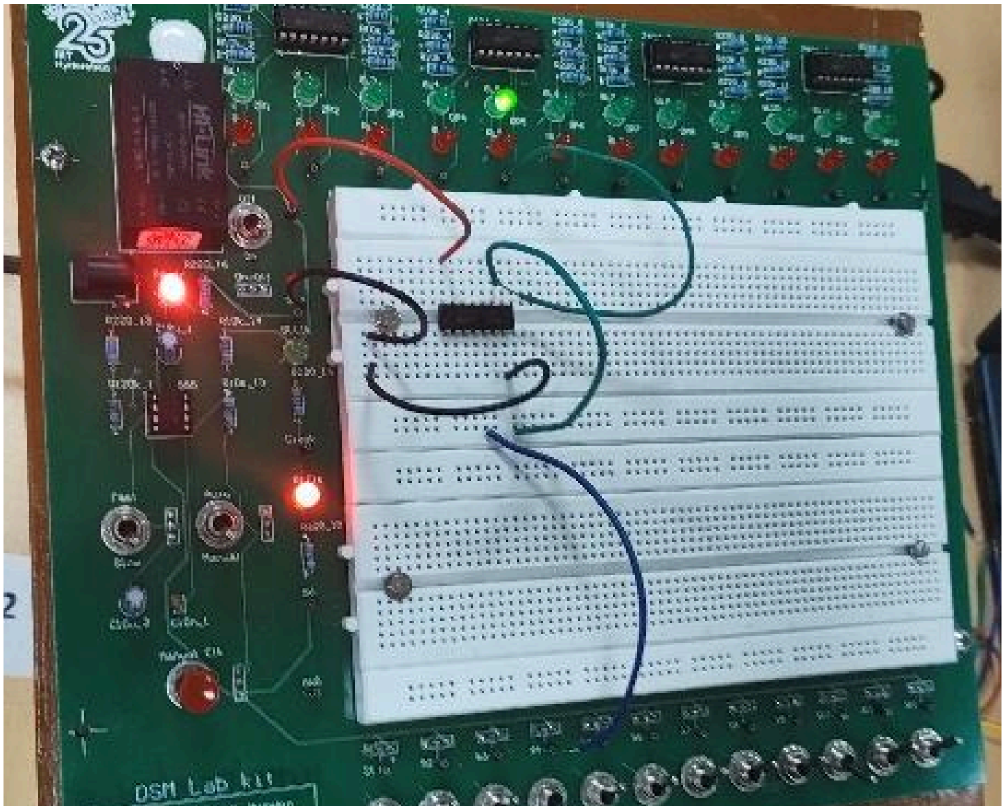
1. Digital Test Kit (DSM Lab Kit)
2. Connecting wires
3. Breadboard
4. IC 7404 (Hex Inverter/NOT gate)

Circuit Implementation and Reference Circuit:

In this experiment, the NOT gate IC was connected to the digital test kit to check its logic behavior. The circuit was put together using standard digital logic methods.



Visual Circuit:



Procedure:

1. Verify the functionality of Digital Test Kit by testing the LEDs and input switches.
2. Connect the NOT gate IC carefully to the breadboard.
3. Connect power supply from source using red wire for VCC and black wire for GND at the designated places for the IC.
4. Connect one of the inputs from the inputs below to the NOT gate input.
5. Connect output to one of the LEDs of the Digital Test Kit.
6. Now carefully turn on the power supply and record the observations

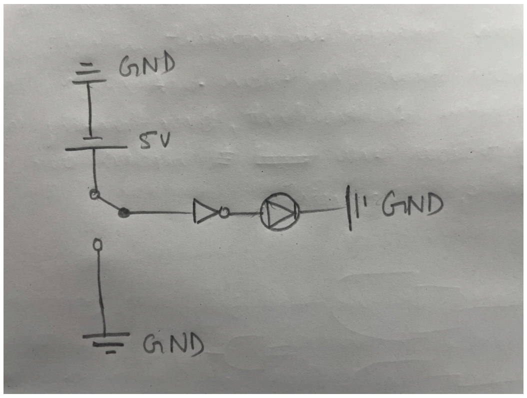
Observation:

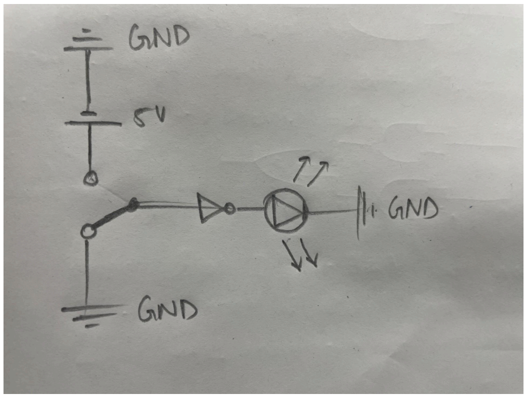
The experimental observations confirmed the inverting behavior of the NOT gate

Input state	LED state	Inference
HIGH (1)	LED connected doesn't glow	Output is LOW (0)
LOW (0)	LED connected glows	Output is HIGH (1)

Result and Analysis:

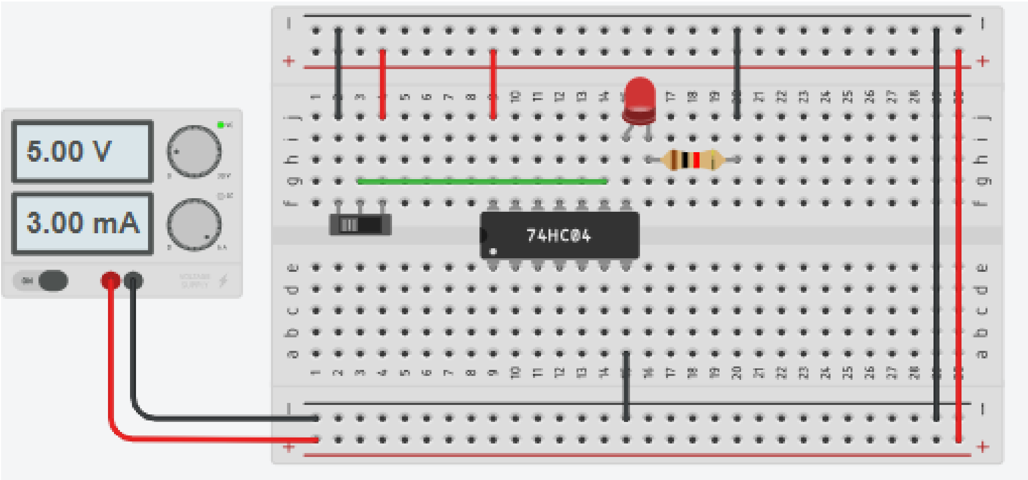
The NOT gate successfully demonstrated its fundamental property of logic inversion, where the output is always the complement of the input signal.





Circuit Simulation:

https://www.tinkercad.com/things/8uGIWwM08pS-not-gate?sharecode=d9cndLah_dPiOi1pgK0iG9NnFioLaAY6Jxn7ZLZu-fA

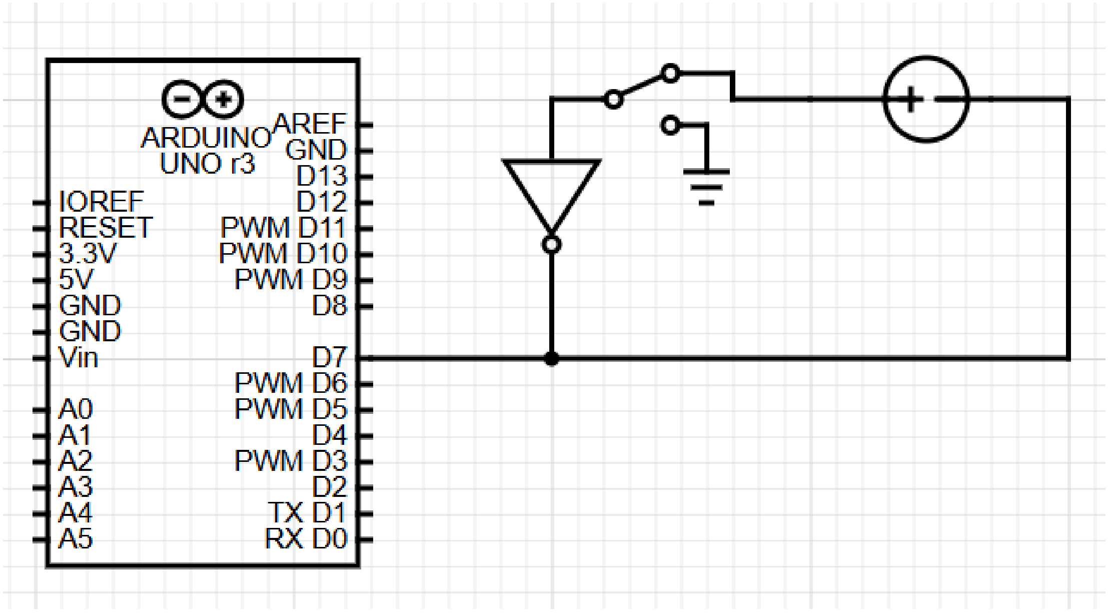


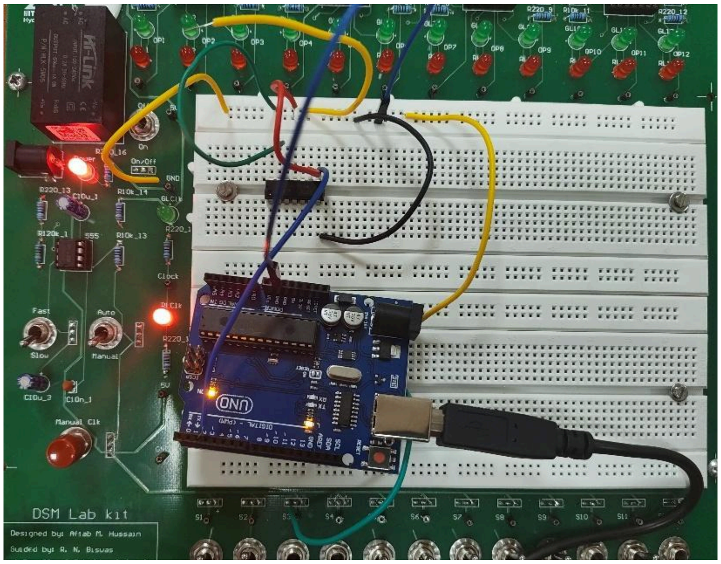
EXPERIMENT 2 : Digital Test Kit and NOT Gate Implementation

Aim:
To understand the interface of Arduino IDE with the help of NOT gate output by displaying a conditional message based on the state of the digital signal.

Components Required:
1. Digital Test Kit with NOT gate circuit (from Experiment 1)
2. Additional connecting wires
3. Arduino UNO microcontroller
4. USB cable for programming

Circuit Implementation and Reference Circuit:
In this experiment, the Arduino was connected to the NOT gate circuit by linking the gate's output to digital pin 5. A common ground was also shared between the test kit and the Arduino to maintain proper signal levels.





Procedure:

1. Setup Arduino using the connecting cable and Arduino IDE on the laptop.
2. Carefully connect the digital input pin 5 to the output of NOT gate IC.
3. Connect the ground (GND) on the Arduino to the ground on the breadboard.
4. Setup the serial monitor for output display.
5. Write a code to print “Hello world” when input is HIGH and “Bye” when input is LOW, in the serial monitor.
6. Upload the code to Arduino using the connection port.
7. Observe the serial monitor by changing the sliding switch position.

Software Implementation:

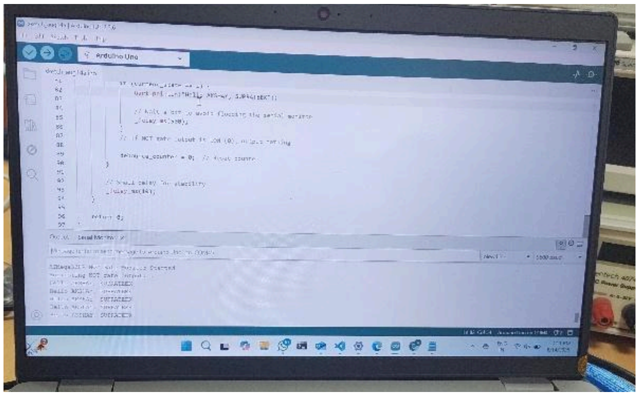
By writing the following code in Arduino IDE we can configure the serial monitor to display the desired message.

```
void setup() {
  Serial.begin(9600);
  pinMode(7, INPUT);
}
void loop() {
  int gate_output = digitalRead(7);
  if (gate_output == HIGH)
   {
     Serial.println("Hello world");
   }
  else
   {
     Serial.println("Bye");
   }
  delay(1000);
}
```

Observation:

The system’s behavior was observed using the serial monitor:

- When the NOT gate output was HIGH (input LOW), the display showed 'Hello World'.
- When the NOT gate output was LOW (input HIGH), the display showed 'Bye'.

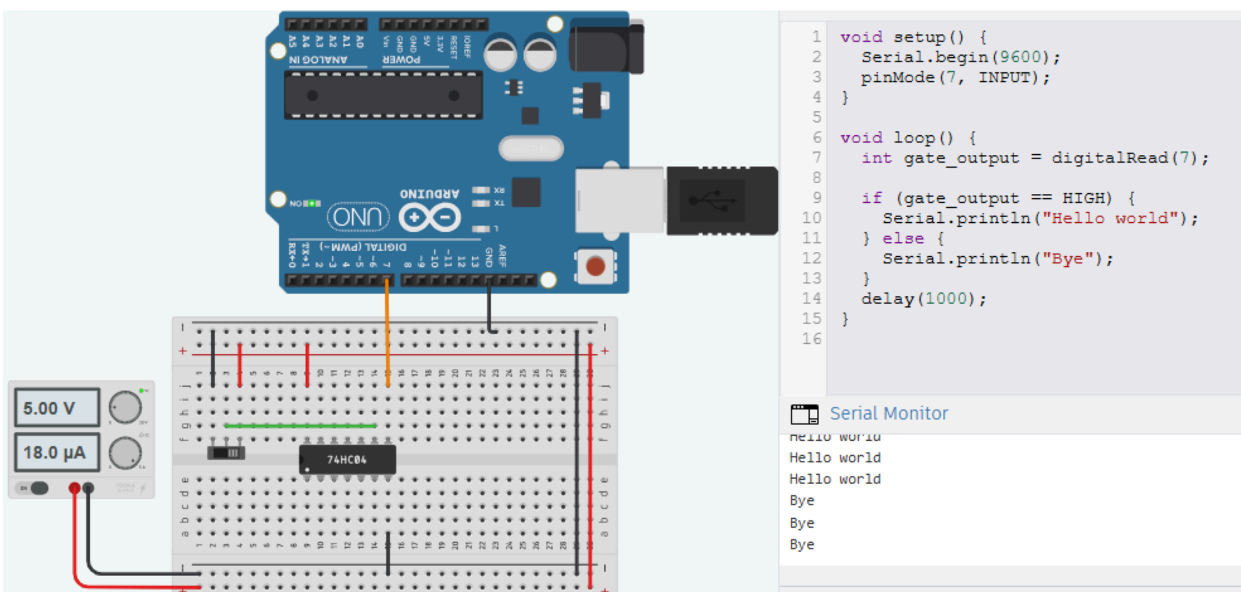


Input state (switch)	Serial monitor display	Inference
HIGH (1)	Hello world	Reading at output gate is LOW (0)
LOW (0)	Bye	Reading at output gate is HIGH (1)

This verifies the proper interfacing of the digital logic circuit with the microcontroller, confirming successful integration of hardware and software.

Circuit Simulation:

https://www.tinkercad.com/things/gGtUDTzpiWX-arduino-not-gate?sharecode=V4-2GV6tCvoyBiYEgYnLSwVwUyvOCrQr4ApWtc_C9Pk



Conclusion:

The experiments were successfully carried out, resulting in the following outcomes:

1. **Logic Fundamentals:** Gained hands-on experience in using a NOT gate and validating its truth table.
2. **Circuit–Microcontroller Link:** Established successful interfacing between discrete logic circuits and a microcontroller.
3. **Coding Application:** Developed Arduino programs for handling digital inputs and enabling serial communication.
4. **Integrated System Learning:** Demonstrated how basic digital logic principles can be extended into programmable systems.

In both standalone and microcontroller-integrated circuits, the NOT gate reliably showed its inverting function, validating fundamental digital logic principles.