DSM LAB REPORT-1

Lab1: using the digital test kit and introduction to Arduino

Group 14: Aaryan Nakul Shah (2024113014)

Experiment – 1

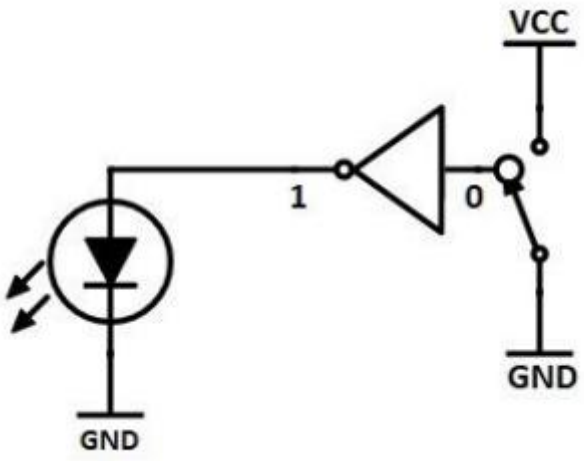
Objective:

Familiarization with Digital Test Kit and Binary Logic Levels.

Components Used:

Digital Test Kit, IC 7404 Hex Converter, and wires

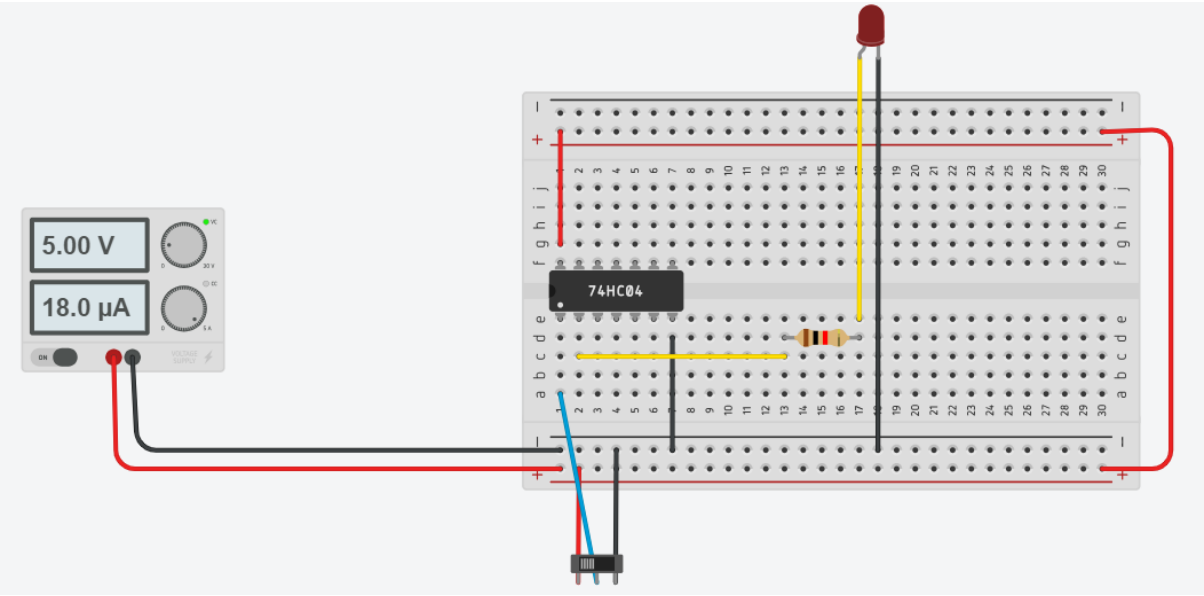
Reference Circuit:

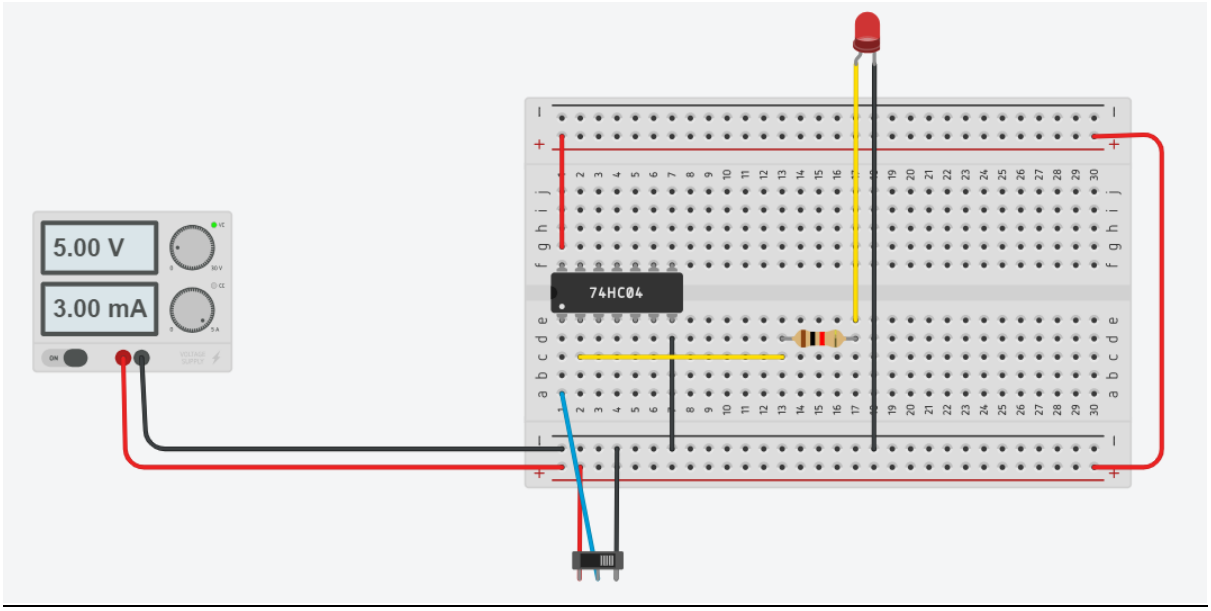


Output:

Tinkercad Circuits:

https://www.tinkercad.com/things/erWsnsSVf2s-lab1-dsm?sharecode=vRdRixgRYpdfPGpi7cxHo8SEKS22-SVFmR5TRmmlM90





Procedure:

1. Connect the IC's VCC and GND pins to the VCC and GND of the Digital Test Kit. Make sure the clock control switch is set to 'FAST'.

2. Connect one of the input switches of the Digital Test Kit to a pin from one of the gates of the IC. Then connect the output pin of the same gate to one of the display points of the Digital Test Kit.
3. Now, on turning the Digital Test Kit on, the functionality of a NOT gate can be observed.

Observation:

When the input switches were set to ‘low’ (0 according to Boolean representation), the LED turned red, and when set to ‘high’ (1 according to Boolean representation), it turned green.

Conclusion:

From our observations, we can infer that the IC 7404 Hex Converter functions as a NOT gate, providing us with the following truth table:

Input	Output
1	0
0	1

Experiment – 2

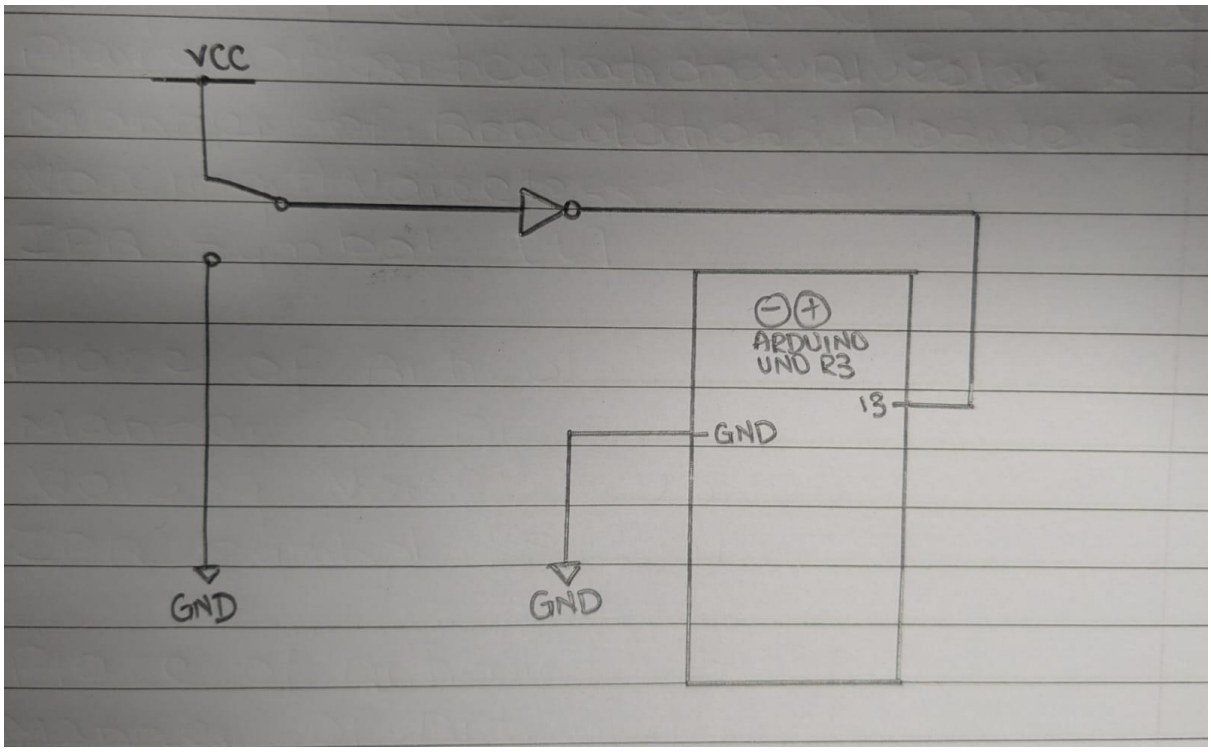
Objective:

Introduction to Arduino through a simple Hello World program.

Components Used:

Digital Test Kit, Arduino UNO board, IC 7404 Hex Converter and wires

Reference Circuit:



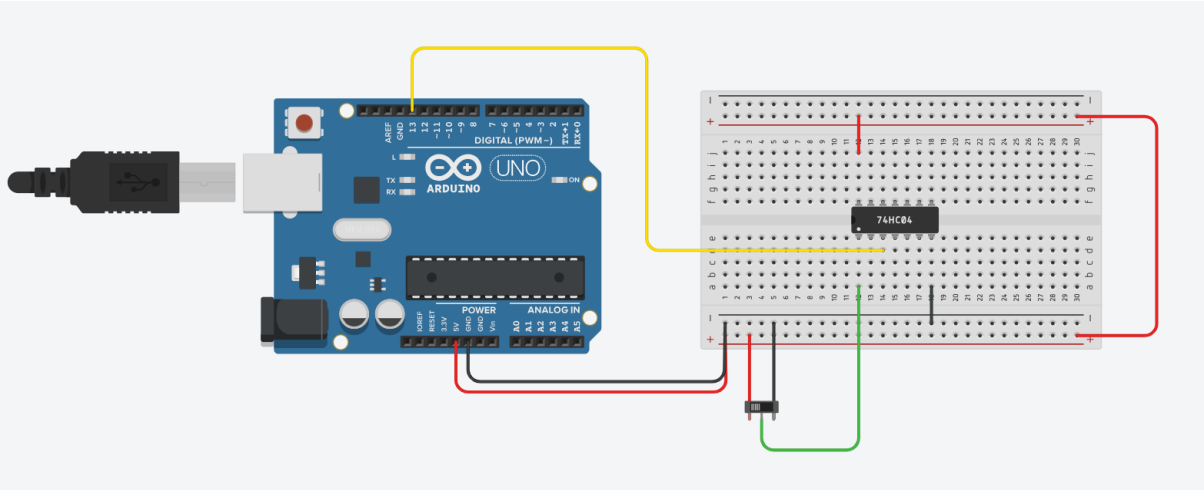
Output:

DSM_Lab-1(Hello World).TS.mp4

DSM_Lab-1(Hello World).TS.mp4

Tinkercad Circuit:

https://www.tinkercad.com/things/gQydfoFNym4-dsmlab1exp2?sharecode=Pl3jt8b1SG277R-94yZnt8qPFC3buylo-wHSSn4zwB4



Procedure:

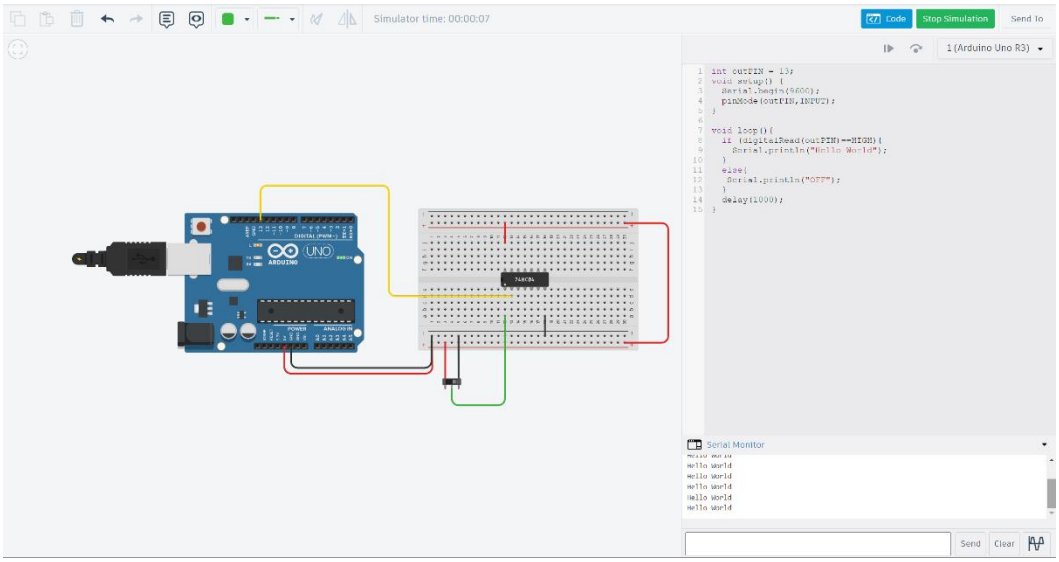
1. Using the setup from Experiment 1, connect the output pin of the gate to a digital pin of the Arduino Uno instead of connecting it to a display point of the Digital Test Kit.
2. Connect the '5V' pin on the Arduino to the breadboard. Ensure that the ground of the Arduino is connected to the ground on the breadboard.
3. Write the necessary code, mine is as follows:

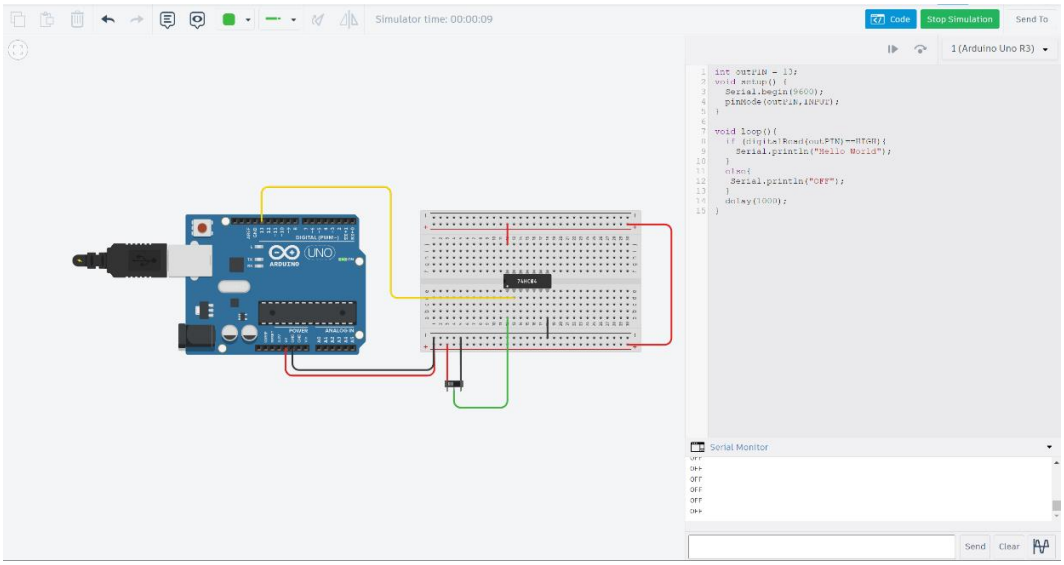
```
int outPIN = 13;
void setup() {
  Serial.begin(9600);
  pinMode(outPIN,INPUT);
}

void loop(){
  if (digitalRead(outPIN)==HIGH){
    Serial.println("Hello World");
  }
  else{
   Serial.println("OFF");
  }
  delay(1000);
}
```

Observation:

The serial monitor displayed “Hello World” whenever the output was ‘high’ i.e. the input was ‘low’ and “OFF” whenever the output was ‘low’ i.e. the input was ‘high’.





Conclusion:

Understood how Arduino detects signals, and how to translate these signals into the output required desired by the user through the means of a code.