Faculty of Engineering and Technology

Department of Electrical and Computer Engineering

ENCS4380

INTERFACING TECHNIQUES

Topic:

Interface Task II

Team members:

Samuel Tannous – 1201123

Jubran Khoury - 1201264

Jihad Anfous - 1200704

Ramiz Shahin - 1201661

Instructor: Dr. Wasel Ghanem

Date: 28/5/2024

Contents

List of tables

Introduction

This task aimed to teach us how to simulate a circuit using Tinkercad software, and subsequently apply this simulation to real hardware, addressing and overcoming any hardware difficulties that arise.

In this task, we used an ultrasonic sensor, an IR sensor, an LCD, and a push button. The ultrasonic sensor monitored the right side and displayed the distance from an object on the LCD, invoking a delay function to simulate a time-consuming process (3 seconds). The IR sensor fired an interrupt whenever an object approached the system from the left side within a distance of less than 5 cm and displayed an alert on the LCD. Additionally, a push button was implemented to fire an interrupt when pressed, lighting an LED. We used Tinkercad to simulate the circuit before implementation.

Theory

Ultrasonic sensor: the most popular sensor for measuring distance, the sensor consists of two ultrasonic transducers. One is a transmitter that produces ultrasonic sound pulses, while the other is a receiver that listens for reflected waves.

"The sensor has 4 pins. VCC and GND connect to the Arduino's 5V and GND pins, respectively, while Trig and Echo connect to any digital Arduino pin. The Trig pin sends the ultrasonic wave from the transmitter, while the Echo pin detects the reflected signal". [1]

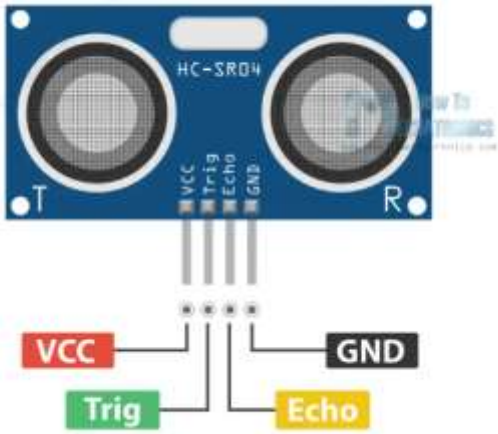

Figure 1(Ultrasonic sensor) [1]

Liquid crystal display(LCD)

"It is used for displaying different messages on a miniature liquid crystal display". [2]

"It is based on the HD44780 microcontroller (Hitachi) and can display messages in two lines with 16 characters each". [1] "It displays all the letters of alphabet, Greek letters, punctuation marks, mathematical symbols etc. In addition, it is possible to display symbols made up by the user. Other useful features include automatic message shift (left and right), cursor appearance and LED backlight ". [2]

Figure 2(LCD) [2]

IR sensor

"An **infrared proximity sensor or IR Sensor** is an electronic device that emits infrared lights to sense some aspect of the surroundings and can be employed to detect the motion of an object. As this is a passive sensor, it can only measure infrared radiation". [3]

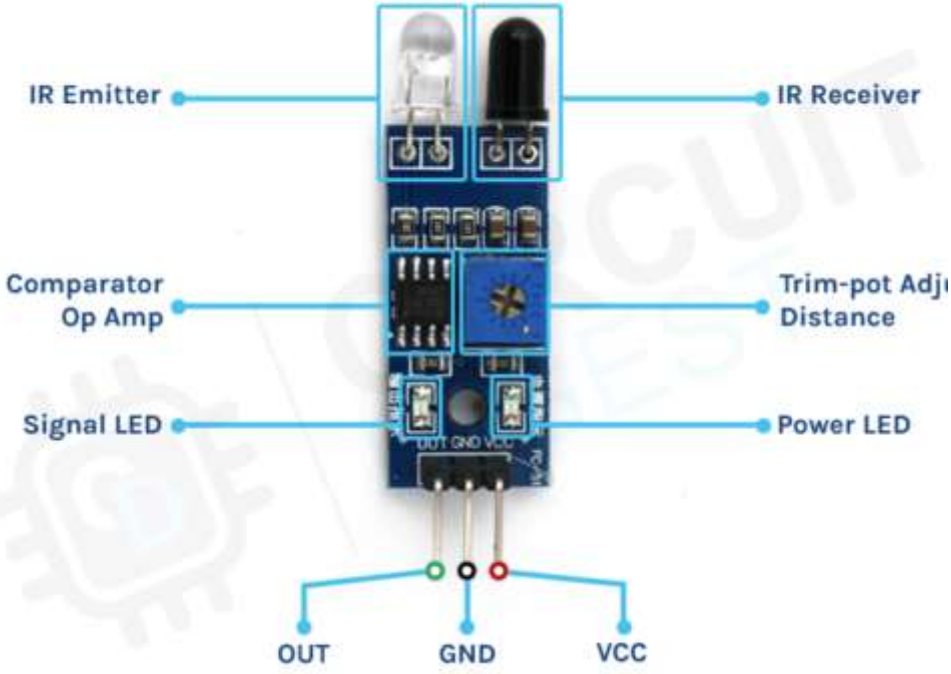

Figure 3(IR sensor) [3]

Procedure

The circuit was simulated on the Tinkercad first, as shown in the fig below

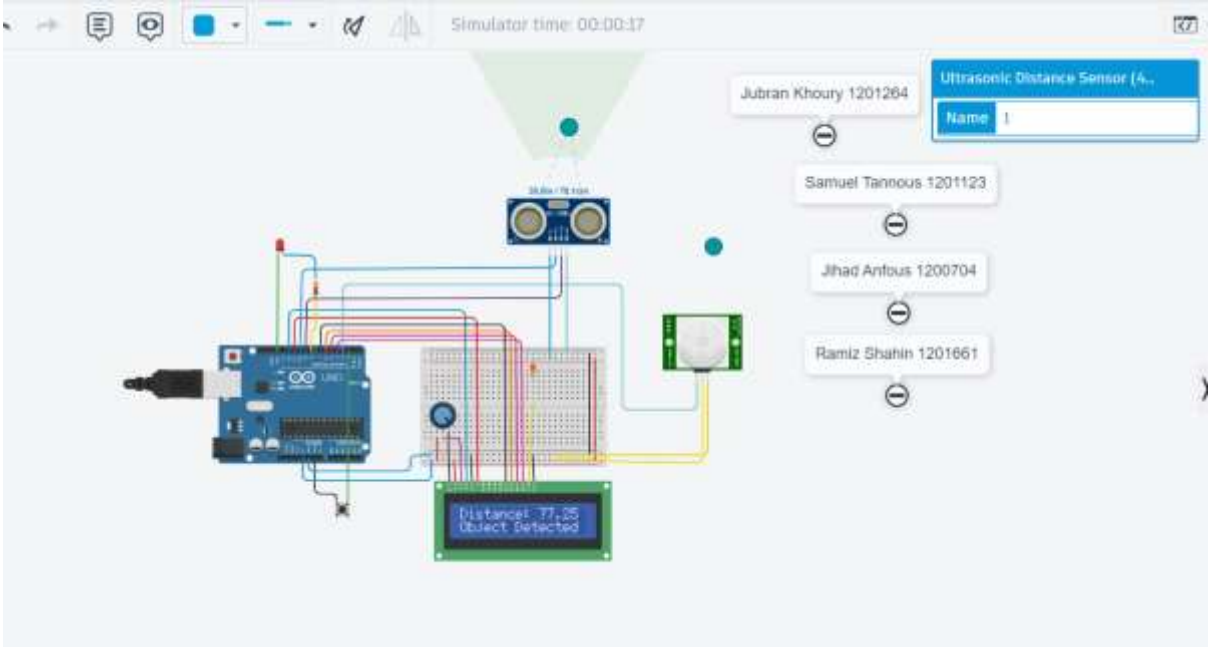

Figure 4(Tinkercad simulation)

Initially, the ultrasonic sensor reads distances by emitting an ultrasonic wave upon receiving a trigger signal. This wave travels through the air at the speed of sound and, upon encountering an object, bounces back to the sensor. The Echo pin of the sensor goes high when the wave is emitted and goes low when the reflected wave returns, marking the duration of the echo pulse. Using **pulseIn()** function, we measure the time the Echo pin remains high. The distance to the object is then calculated by halving this duration (Because go forward and go back) and multiplying by the speed of sound in air (~343 m/s). After the ultrasonic has read the distance, a delay of three seconds is applied. The LCD is used to display this distance in cm.

Additionally, we connected a push button so that each time the button is pressed, it toggles the state of the LED. This step was done using the interrupt ensuring that the LED changes its state (on or off) every time the button is pressed.

Finally, we also connected an IR sensor, also using an interrupt. When the sensor detects an object approaching within a distance of less than 5 cm, it triggers the interrupt, displaying a "Object Detected" on the LCD.

Hands on hardware circuit connection and testing

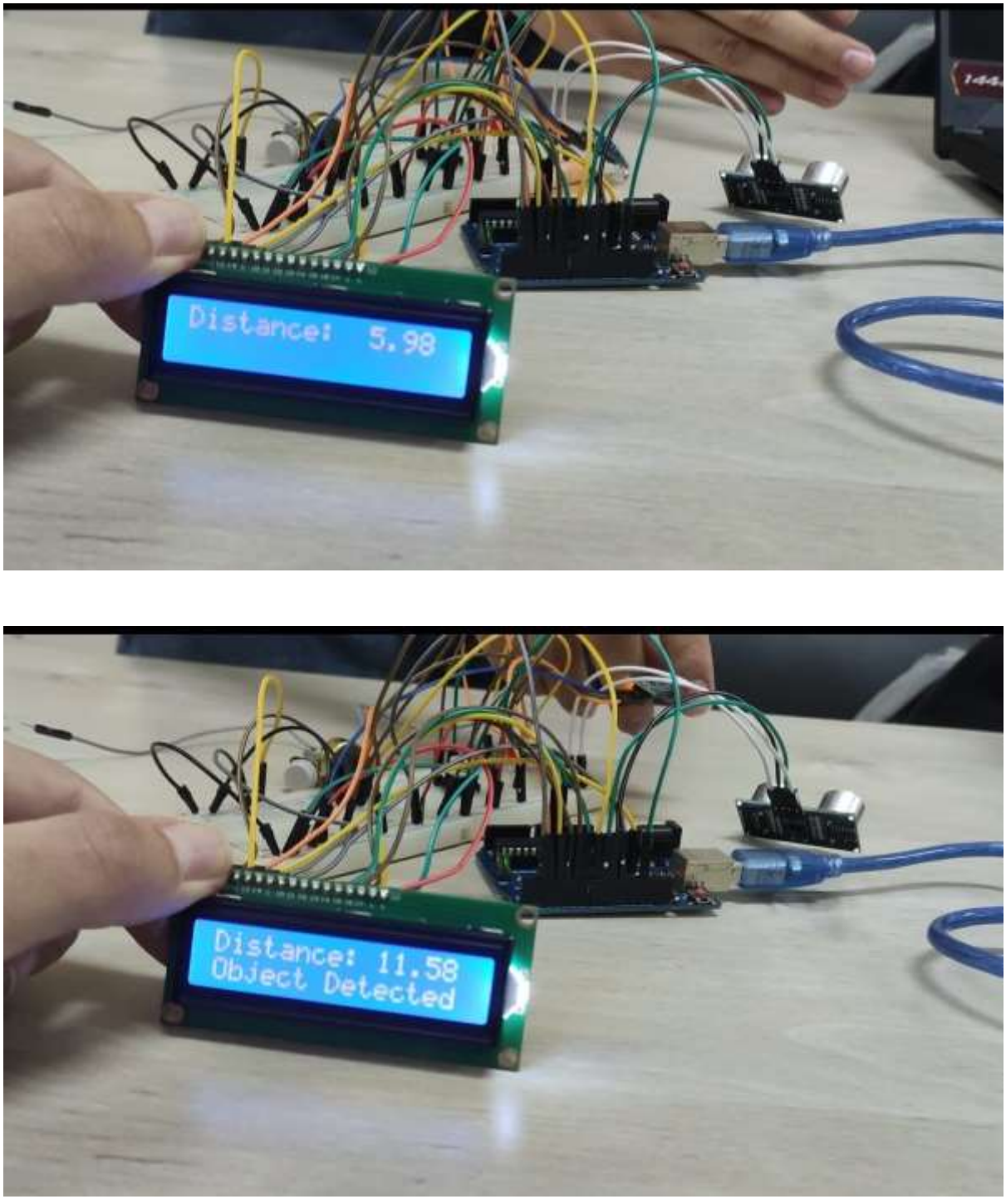

Figure 5(Hardware connection and testing)

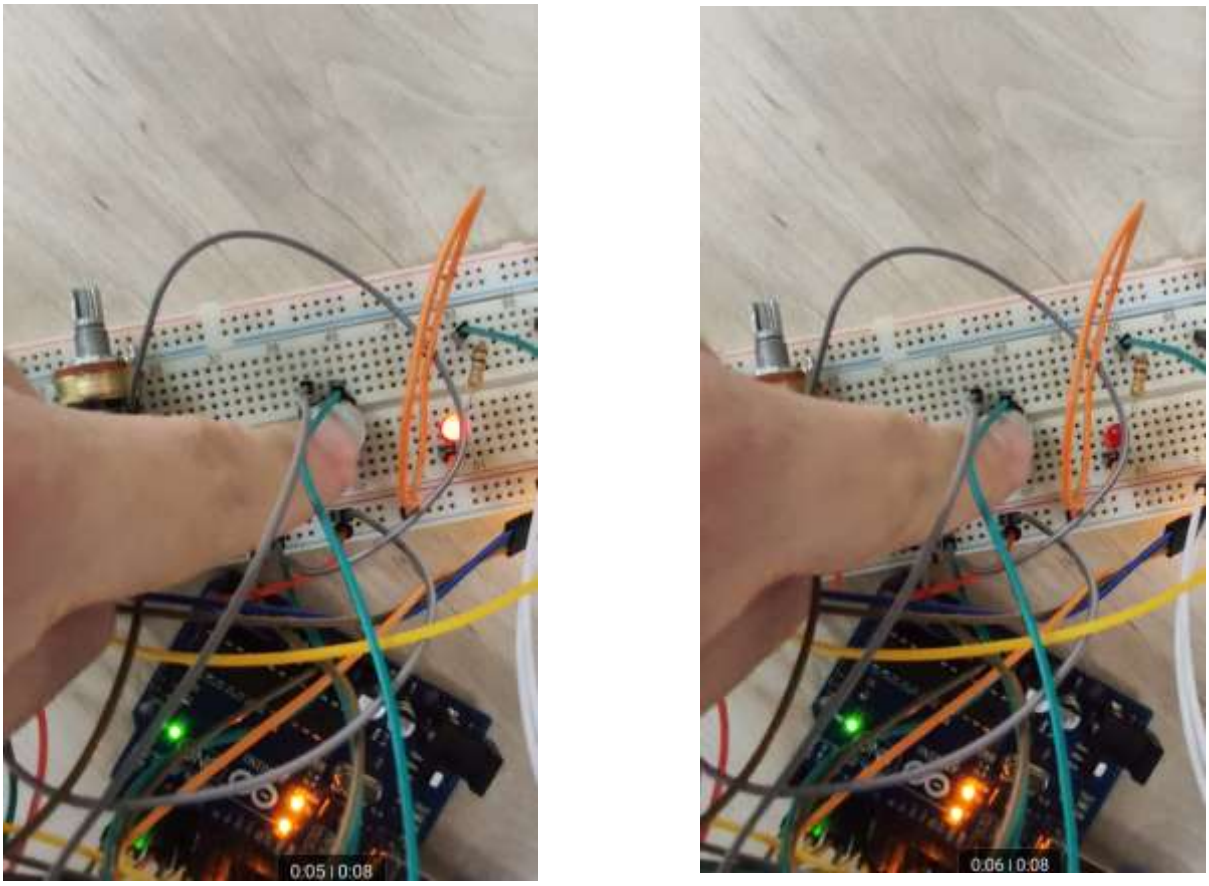

Figure 6(Hardware connection and testing)

Conclusion

In conclusion, this task gave us a better understanding and hands on practical with Arduino hardware circuits and how that the hardware connection differ from the simulation is that during the physical connection and real world is that the environment is not ideal and affected by a noise and also a damaged physical components like wires or any devices, which lead us to check all components before using them, unlike the simulation which has always the ideal conditions with no defected components. During this task we don’t face any problem and the result was as expected.

References

[1]: https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/

[Accessed on 28 May 24, 20:30]

[2]: Manual for Computer Design Lab, 2023, Birzeit University.

[3]:shttps://circuitdigest.com/microcontroller-projects/interfacing-ir-sensor-module-with-arduino

[Accessed on 28 May 24, 20:30]