An-Najah National University
Faculty of Engineering and IT

جامعة النجاح الوطنية
كلية الهندسة وتكنولوجيا

Computer Engineering Department
Microcontroller Lab (10636496)
Report Grading Sheet

Instructor Name: Hikmat Darawsheh	**Experiment: Arduino 1**
Academic Year: 2024	**Performed on:**
Semester:	**Submitted on:**
Student Names:	
1- Wala' Essam Ashqar	2- Sadeen Hawash
3-	4-
5-	6-

Evaluation Criterion	CLO	Grade	Points
Abstract and Aims Aims and idea of the experiment are clearly stated in simple words		10	
Introduction, Apparatus and Procedures Introduction is complete and well-written, all grammar/spelling correct, Appropriate background information related to the principles of the experiment is provided. The list of apparatus and procedures are also provided		15	
Experimental Results, Calculations and Discussion Results analyzed correctly. Experimental findings adequately and specifically summarized, in graphical, tabular, and/or written form. Comparison of theoretical predictions to experimental results, including discussion of accuracy and error analysis as needed.		50	
Conclusions Conclusions summarize the major findings from the experimental results with adequate specificity. Highlighting the most important results		15	
Appearance Title page is complete, page numbers applied, content is well organized, correct spelling, fonts are consistent, good visual appeal. You have also to use reference for the information you provide		10	
Total		100	

Contents

Abstract

In this experiment we got to use Arduino mega 2560 and use an Arduino IDE to write code to do different task and get familiar with it.

Introduction

Through this experiment we got to work with different component and connect them to Arduino mega 2560 and write some code for each component to do a different task and to do so we used Arduino IDE and then we implemented our code and tested it on the components.

Materials

- Arduino Mega 2560.
- Arduino IDE.
- Wires.
- PIR Sensor.
- Servo motor.
- IR Sensor.
- Ultrasonic Sensor.
- Keypad.

Methods

To do each task we needed first to connect each component with the correct pins to ensure that is compatible with our code and also make sure that the same pins are mapped in the code to the ones we used for the component and then we run the code for each task and observe the outcome.

Experimental Results

Task 1: we wanted to flash a led every 1 seconds so we used pin 13 and we edit it value to 1 (on) and delay for 1 second and then value to 0 (off) for 1 seconds and the operation keep going for ever (inside loop).

```
void setup() {
pinMode(13, OUTPUT);
} void loop() {
  digitalWrite(13, HIGH);
  delay(1000);
  digitalWrite(13, LOW);
  delay(1000);     }
```

Task 2: Serial Monitor: we used serial monitor to print hello world every 1 second and we set the baud rate to 9600.

```
void setup() {
  Serial.begin(9600);
}
void loop() {
    delay(1000);
    Serial.println("Hello World");
 }
```

Task 3: PIR Sensor: we used pin 9 as output pin from the PIR sensor and whenever a motion is detected it will set the led on and then print hello world in the serial monitor evert 1 second.

```
int out =9;
void setup() {
    Serial.begin(9600);
  pinMode(out, INPUT);
}void loop() {
  int val = digitalRead(out);
  digitalWrite(13, val);
  if(val ==1 )
    Serial.println("Hello World");
  delay(1000);
 }
```

Task 4: Basic Servo Control: we set pin 9 to the servo object and then we wrote a for loop to change the servo in 180 degree and put some delay to give the command some time to execute

```
#include <Servo.h>

Servo servo1;

int posn = 0;

void setup()

{

  servo1.attach (9);

}

void loop()

{

  for (posn = 0; posn < 180; posn += 1)

{

    servo1.write (posn);

    delay (10);

  }

}
```

Task 5: Obstacle Detection: IR Sensor vs. Ultrasonic Sensor:

- Testing the Ultrasonic sensor: we set trig as pin 8 and echo as pin 9 and for trig pin we set it high (1) for 10 microseconds and then we set it low for 2 microseconds and we do that to enable the component and then we read the value from echo pin and we calculate the distance by multiplying it by the speed of sound and then divided by 2 because we only want the distance to the object and then print the value to the serial monitor.

```
int trig =8; int echo =9;
void setup() {
   Serial.begin(9600);
   pinMode(trig, OUTPUT);
   pinMode(echo, INPUT);
}
void loop() {
  digitalWrite(trig, HIGH);
  delayMicroseconds(10);
  digitalWrite(trig, LOW);
  delayMicroseconds(2);
  int duration = pulseIn(echo, HIGH);
  int distance = (duration * 0.034 /2 );
  Serial.print("distance =" );
  Serial.println(distance );
  delay(1000);
}
```

- Testing the IR sensor:

we set IR as pin 9 and as input to read from and when there is object detected by the sensor the value of IR will be 1 (high) and then we print Object detected in the serial monitor or if there is no object its value will be 0(low) and then we print No Object detected in the serial monitor and we check every half a second.

```
int IR =9;
void setup() {
   Serial.begin(9600);
   pinMode(IR, INPUT);
}
void loop() {
  int val =  digitalRead(IR);
  if( val == HIGH){
   Serial.println("Object detected" );
  }
  else {
   Serial.println(" No Object detected" );
  }
  delay(500);
}
```

Task 6: Keypad: here we used a library called Keypad.h and then identified for each button its value as the keypad and then connected the keypad to the corresponding pins in the code and then whenever a key is pressed we print its value in the serial monitor.

```
# include <Keypad.h>

const byte ROWS = 4;

const byte COLS = 4;

// Define the Keymap

char keys[ROWS][COLS] = {

  {'1', '2', '3', 'A'},

  {'4', '5', '6', 'B'},

  {'7', '8', '9', 'C'},

  {'*', '0', '#', 'D'} };

byte rowPins[ROWS] = {0, 1, 2, 3};

byte colPins[COLS] = {4, 5, 6, 7};

 Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);

void setup() {

  Serial.begin(9600); }

void loop() {

  char key = keypad.getKey();

  if (key) {

    Serial.println(key);   }

  delay(10);

}
```

Discussion

To work with different components, we need to map them to the correct pins that corresponds to the code and also make sure that the VCC and GND for each component is connected and then we test the code from the IDE to the component.

Conclusion

This lab we learnt how to work with Arduino Mega 2560 and Arduino IDE and did different type of tasks with different components and for example we calculated the distance between a component and a sensor and we worked with a motion detector.