



מטלה 3 - Arduino

קבוצה 6 - מעבדה 11			
206769754	208687996	206314874	319024105

מטרת המערכת ותיאור משימת הבקרה

במטלה זו בנינו מערכת אשר מדמה מערכת שמירה על מקרר גלידות גומיגם בבית מלון "GROUP 6".

- **מסך ה-LCD** יקרין הודעת "Gumigam Refrigerator" בעת תחילת הסימולציה.
- **חיישן האולטרא סוניק (ultrasonic distance sensor)** מחשב את המרחק בקרבת מקרר הגלידות.
- כל עוד האדם יהיה במרחק הגדול מ-30 ס"מ, **נורת לד צהובה (Yellow LED)** תדלק בעוצמה מקסימלית.
- במצב של זיהוי דמות מתקרבת ותהיה במרחק הקטן מ-30 ס"מ, הנורה הירוקה תכבה ובמקומה תדלק **נורת לד אדומה (Red LED)** תדלק בעוצמה מקסימלית ומסך ה-LCD יקרין הודעת "Warning!!!" , **חיישן הקול (Piezo)** יפעיל אזעקה קבועה ו**מנוע ה-Servo** יתחיל לסרוק את השטח בתנועה סיבובית וידמה תנועת פרופלור של רחפן צילום אוויר שיתעד את שטח סביב המקרר והגנב שהתקרב.

תיאור התכנ

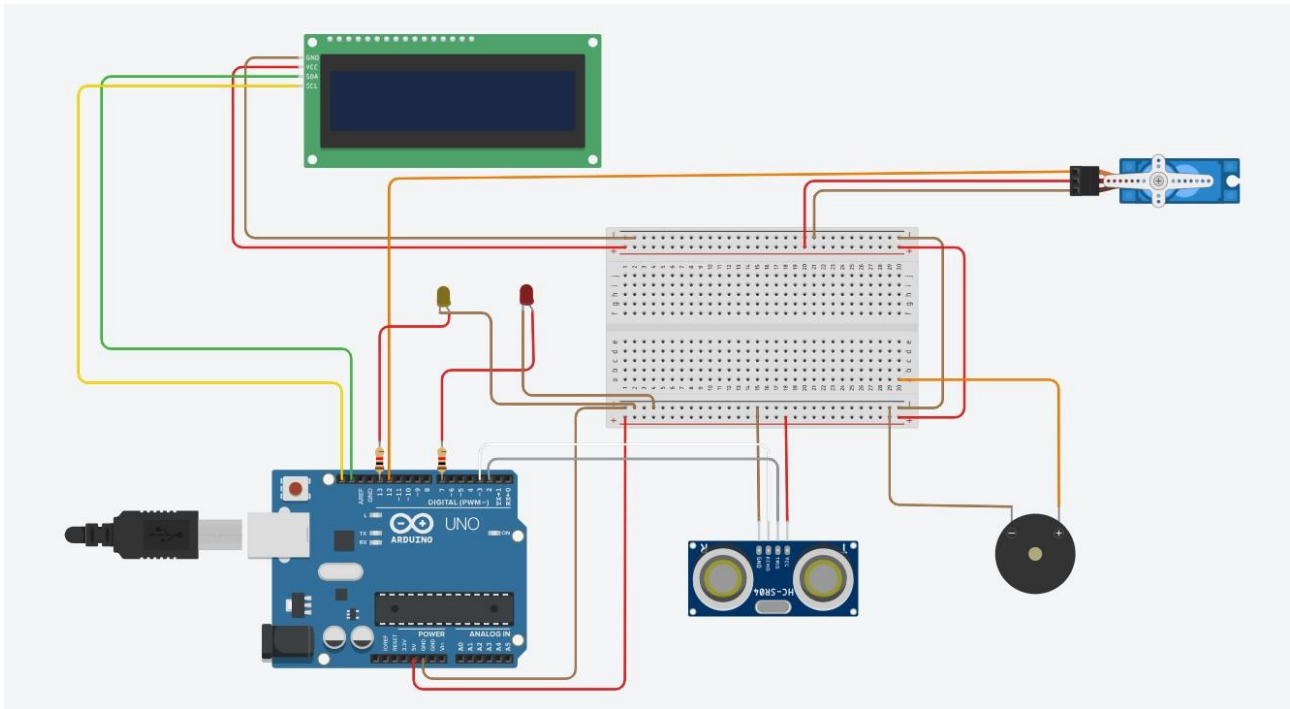
הפרויקט מומש ע"י בקר ארדואינו אנו בנוסף לרכיבים נוספים :

טבלת הפינים של הבקר			
שם הרכיב	מספר הפין	תצורה (OUTPUT/INPUT)	הפעלה
LCD 16X2 (I2C)	-	OUTPUT	משמש להצגת הודעות והנחיות לאדם המתקרב למקרר הגלידות של המלון
Ultrasonic Distance Sensor	2,3	INPUT	משמש למדידת המרחק בין האדם למקרר הגלידות
Micro Servo	12	OUTPUT	משמש לסריקת האזור
Yellow LRD	13	OUTPUT	דלוקה במצב שגרה
Red LRD	7	OUTPUT	דלוקה כאשר המערכת זיהתה התקרבות למקרר
Piezo	8	OUTPUT	משמש להתרעה קולית כאשר המערכת מזהה התקרבות למקרר



התכן נעשה בעזרת תכנת הסימולטור Tinkercad (גרסה 1.4, מבית חברת Autodesk).

מצ"ב שרטוט מתוך תוכנת הסימולציה Tinkercad :



תיאור הבדיקות שבוצעו

מטרת הבדיקות הינה לוודא את אמינות תוצאות חיישן האולטרא סוניק כאשר הוא מודד את המרחק למקרר הגלילות. ביצענו את הבדיקה הנוספת באמצעות סרגל מדידה והשווינו לתוצאת החיישן. המבחן הסטטיסטי שביצענו הינו T מזווג אשר משווה בין בדיקות שונות שמבוצעות על אותו הפרט ברמת בטחון של 95%, כאשר השערת האפס הייתה שהפרש התוחלות בין תוצאות החיישן לסרגל הינו 0, כלומר החיישן אמין.

$$\begin{cases} H1, & \mu_1 = \mu_2 \\ H0, & \text{else} \end{cases}$$



מספר בדיקה	מרחק הנמדד ע"י החיישן	מרחק הנמדד ע"י הסרגל	השגיאה
1	30	29	1
2	20	18	2
3	30	30	0
4	15	15	0
5	30	30	0
6	20	20	0
7	30	29	1
8	15	15	0
9	20	20	0
10	30	30	0
11	30	30	0
12	30	30	0
13	30	30	0
14	20	20	0
15	30	30	0
16	15	14	1

תוצאות הבדיקות

Paired t-test

```
data: sensor_data and ruler_data
t = 2.0761, df = 15, p-value = 0.05549
alternative hypothesis: true mean difference is not equal to 0
95 percent confidence interval:
 -0.008325641  0.633325641
sample estimates:
mean difference
      0.3125
```

ניתן לראות כי לא נדחה את השערת האפס, כלומר החיישן זהה בזיהוי המרחק מהמקרר לתוצאות המדידה של הסרגל בפועל ברמת ביטחון של 95% ולכן החיישן מבצע את עבודתו במערכת ברמת אמינות גבוהה.



סיכום ומסקנות

במהלך הפרויקט למדנו והתמקצענו בעבודה עם בקר ארדואינו אונו ותוכנת Arduino IDE ובאמצעותם פיתחנו מערכת התראה עבור התקרבות למקרר גלידות בבית מלון..
להמשך הפיתוח נמליץ על הטמעת מערכת הודעות מתוחכמת יותר כך שבמקום להפעיל רק אזעקה המערכת תשלח התראה למייל ולטלפון של מנהל התפעול של המלון.

נספחים

README - הוראות למפעיל

Components Used

Arduino board (UNO or similar)

LEDs (Yellow and Red)

Servo motor

Piezo buzzer

Ultrasonic distance sensor

LCD display with I2C interface

Installation and Setup

Connect the components as per the provided pin configurations in the code:

Yellow LED to pin 13

Red LED to pin 7

Piezo buzzer to pin 8

Ultrasonic distance sensor to pin 2 and pin 3

Servo motor to pin 12

LCD display using I2C communication (Address: 0x27, Rows: 16, Columns: 2)



Connect Arduino uno to the computer using USB.

Choose in "Tools", Board and from the list choose "Arduino uno".

Again choose "Tools", Port and choose the port on the "Arduino uno".

Upload the provided Arduino sketch to your Arduino board using the Arduino IDE.

Once uploaded, power up the Arduino board and ensure that all components are functioning correctly.

Operation

Upon startup, the system initializes and displays "Gumigam Refrigerator" on the LCD Screen.

In normal operation (no motion detected):

Yellow LED is turned on.

Servo motor do not rotates.

When motion is detected by the Ultrasonic distance sensor:

Servo motor starts moving.

Yellow LED turns off, and the Red LED turns on to indicate an emergency situation.

LCD screen displays "Warning!!!" to alert the user.

Piezo buzzer emits a sound for 5 seconds to draw attention to the detected motion

After 5 seconds of the emergency state, the system returns to normal operation, waiting for the next motion detection.

קוד מבחן T - מזווג בתוכנת R-Studio

```
sensor_data<-c(30,20,30,15,30,20,30,15,20,30,30,30,30,20,30,15)
ruler_data<-c(29,18,30,15,30,20,29,15,20,30,30,30,30,20,30,14)
result<-t.test(sensor_data,ruler_data,paired=TRUE)
print(result)
```



```
#include <LiquidCrystal_I2C.h>
#include <Servo.h>
#include <Wire.h>
LiquidCrystal_I2C lcd(0x27,16,2);
#define led13 13 // port number of the yellow led
#define led7 7 // port number of the red led
int Piezo = 8; // port number of the piezo
Servo servo_12;
int servo = 12; // port number of the servo
const int pingPin = 2;
const int echoPin = 3;
long duration, cm = 0;

void setup()
{
  pinMode(servo, INPUT);
  servo_12.attach(12);
  pinMode(Piezo, OUTPUT);
  pinMode(led13, OUTPUT);
  pinMode(led7, OUTPUT);
  Serial.begin(9600);
  lcd.begin();
  lcd.setBacklight(1);
  Serial.println("Start Operation") ;
  pinMode(pingPin, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin, INPUT); // Sets the echoPin as an Input
}

void loop()
{
  // duration = pulseIn(echoPin, HIGH);
  ultraSonic();
  printCmDistance(cm);
  if (cm < 30)
  { // which means that motion is detected
    Serial.println("Motion detected!");
    servo_12.write(0); // makes the servo stop moving
    digitalWrite(led13, LOW); // makes the yellow led turn off
    digitalWrite(led7, HIGH); // makes the red led turn on
    lcd_emergency(); // function that makes the lcd write "Warning!!!"
    tone(Piezo,1000); // start making noise
    delay(1000);
    noTone(Piezo);
  }
  else { // which means no motion is detected
    Serial.println("Area Clean") ;
    lcd_normal(); // function that makes the lcd write "Gumigam Refrigerator"
```



```
servo_move();  
digitalWrite(led13, HIGH);  
digitalWrite(led7, LOW);  
noTone(Piezo); // stop making noise  
}  
}  
  
void lcd_normal(){ // make the lcd screen write "Gumigam Refrigerator" when area clean  
  lcd.setCursor(0,0);  
  lcd.print("Gumigam ");  
  lcd.setCursor(4,1);  
  lcd.print("Refrigerator");  
}  
  
void lcd_emergency(){ // make the lcd screen write "Warning!!!" when motion detected  
  lcd.setCursor(0,0);  
  lcd.print("Warning!!!");  
  lcd.setCursor(4,1);  
  lcd.print ("          ");  
}  
void servo_move(){ // make the servo start rotating  
  Serial.println("Servo start moving");  
  servo_12.write(90);  
  delay(15);  
}  
  
long microsecondsToCm(long microseconds){  
  return (microseconds)/29/2;  
}  
  
void printCmDistance(long cm){  
  Serial.print(cm);  
  Serial.print("cm");  
  Serial.println();  
}  
  
long ultraSonic(){  
  digitalWrite(pingPin, LOW);  
  delayMicroseconds(2);  
  // Sets the trigPin on HIGH state for 10 micro seconds  
  digitalWrite(pingPin, HIGH);  
  delayMicroseconds(10);  
  digitalWrite(pingPin, LOW);  
  // Reads the echoPin, returns the sound wave travel time in microseconds  
  duration = pulseIn(echoPin, HIGH);  
  cm = microsecondsToCm(duration);  
  return cm;  
}
```