מערכת זיהוי ומזיגת מים אוטומטית - תמי 4

חברתי הצוות: יונתן לרקין 204792337 יותם קומש 208721746 עומר עיני 316112028 דניאל קחטן 208803023

קבוצת תרגול: 33

תאריך הגשה: 21.05.2023

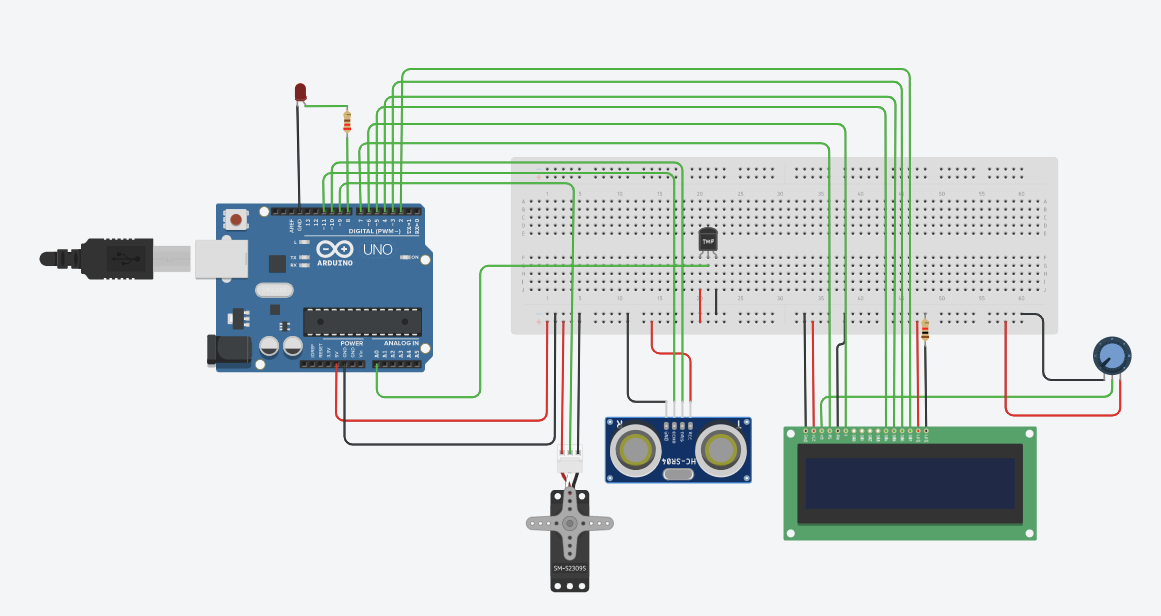
1. **מטרת הפרויקט ואופן פעולת המערכת**

מטרת הפרויקט: ליצור מערכת מזיגה אוטומטית לפי חיישן מרחק שמזהה כוס וחיישן טמפרטורה שמודד שהמים מתחת ל10 מעלות - המערכת מיועדת לבתים פרטיים בהם יש צורך במזיגת מים קרים בצורה אוטומטית.

הלוגיקה של פעולת המערכת תוכננה כך שכאשר המערכת מזהה כוס במרחק של עד 20 ס"מ ובמקביל המים נמצאים בטמפרטורה נמוכה מ10 מעלות, המערכת תתחיל את תהליך המזיגה באמצעות מנוע הסרבו במשך 5 שניות, מסך הlcd ידפיס שהמערכת במזיגה והנורה תחל להבהב. בעת תחילת עבודה יודפס על מסך הLCD תמי 4, המערכת פועלת ובמצבי קיצון תודפס הודעה מתאימה. אל הSerial monitor יודפסו המרחקים של הכוס הנוכחי והמרחק הקודם שהכוס הייתה בה כל שתי שניות.

1. **תיאור התכן**

הפרויקט מומש באמצעות בקר ארדואינו אונו הרכיבים הכלולים בפרויקט הינם: מסך, נורת לד, חיישן מרחק, מנוע סרבו, חיישן טמפרטורה, מטריצה ופצנטיומר. התכן (איור 1, טבלה 1) נעשה בעזרת תכנת הסימולטור thinkercad (שם התכנה, הגרסא, החברה). את התכן ניתן לראות בקישור: [קישור](https://www.tinkercad.com/things/8MWuUP8OVEa-smashing-blad-leelo/editel?tenant=circuits). תיעוד הקוד מפורט בנספח למסמך זה.



איור 1: שרטוט חשמלי של הפרויקט

*טבלה 1 תיאור הפינים של הבקר*

|  |  |  |  |
| --- | --- | --- | --- |
| מספר הפין | שם המשתנה | תצורה | הפעלה |
| 10 | TRIG\_PIN | OUTPUT | חיישן לבדיקת מצב המרחק – שולח אות לביצוע הבדיקה. פועל כל איטרציה. |
| 11 | ECHO\_PIN | INPUT | חיישן לחישוב המרחק– מקבל את האות חזרה מהאובייקט ומחשב את המרחק. |
| 0A | TEMP\_PIN | OUTPUT | חיישן טמפרטורה – מחובר לתצוגה האנולוגית ומספק מתח פלט בפרופרציה לטמפרטורה. |
| 9 | SERVO\_PIN | OUTPUT | מנוע הסרבו – על מנת להזין זוויות שונות למנוע |
| 8 | LED\_PIN | OUTPUT | נורת לד אדומה – מהבהבת כאשר המים נמצאים בתהליך מזיגה |

1. **בדיקות ולידציה של המערכת**

בוצעו שתי בדיקות ולידציה של המערכת בדיקת התאמה של החומרה לדרישות ובדיקה של הלוגיקה של המערכת. הבדיקה הראשונה בחנה האם דיוק חיישן המרחק מתאים לדרישות המערכת. נדרש כי חיישן המרחק יהיה מדויק כמו סרגל סטנדרטי. לכן בוצעו 11 מדידות של החיישן במרחקים שונים. המרחק בכל מדידה נבדקה על ידי קריאת החיישן ובעזרת מדידת סרגל סטנדרטי. (טבלה 2) מדד דיוק החיישן הינו השגיאה האבסולוטית בין קריאות החיישנים. בוצע מבחן t לבדיקה האם ממוצע השגיאות הינו אפס (תוצאות מלאות בנספחים). תוצאת המבחן הייתה שברמת מובהקות של 95% אין הבדל בין המרחקים. בהתאם הסקנו כי החיישן תקין.

עבור בדיקת הלוגיקה של המערכת נעשו 8 חזרות על מעברים בין מרחקים שונים. כלומר א. מעבר ממרחק מחוץ לתחום המזיגה אל מחוץ לתחום המזיגה. ב. מעבר ממרחק מחוץ לתחום המזיגה אל מתוך לתחום המזיגה ג. מעבר ממרחק מתוך לתחום המזיגה אל מחוץ לתחום המזיגה ו-ד' מעבר ממרחק מתוך לתחום המזיגה אל מתוך לתחום המזיגה. בכל הבדיקות המערכת פעלה כמצופה (טבלה 3). בהתאם הסקנו כי המערכת תוכנתה על פי התכנון.

טבלה 2 מדידות שגיאת חיישן מרחק

|  |  |  |  |
| --- | --- | --- | --- |
| מספר בדיקה | מרחק חיישן | מרחק חיישן בדיקה | שגיאה אבסולטית |
| 1 | 5 | 5.4 | 0.4 |
| 2 | 5 | 5.8 | 0.8 |
| 3 | 5 | 5.6 | 0.6 |
| 4 | 5 | 4.8 | 0.2 |
| 5 | 5 | 4.4 | 0.6 |
| 6 | 20 | 21.3 | 1.3 |
| 7 | 20 | 22.1 | 2.1 |
| 8 | 20 | 21.9 | 1.9 |
| 9 | 20 | 19.4 | 0.6 |
| 10 | 40 | 42.3 | 2.3 |
| 11 | 40 | 41.8 | 1.8 |

טבלה 3 בדיקת לוגיקה

|  |  |  |  |
| --- | --- | --- | --- |
| סוג בדיקה (א-ד) | מרחק לפני | מרחק אחרי | תגובת המערכת |
| 1 | 5 | 10 | לא פעלה מחדש |
| 2 | 5 | 15 | לא פעלה מחדש |
| 3 | 10 | 30 | לא פעלה מחדש |
| 4 | 5 | 25 | לא פעלה מחדש |
| 5 | 25 | 10 | החלה לפעול |
| 6 | 30 | 18 | החלה לפעול |
| 7 | 40 | 22 | לא פעלה |
| 8 | 22 | 30 | לא פעלה |

1. **סיכום**

פותחה מערכת זיהוי והפעלה של מערכת שתייה אוטומטית. חיישן המרחק של המערכת נבחן ונמצא מתאימים לדרישות. המערכת פעלה כשורה בכל המעברים של המרחקים בסט הבדיקות. הנחנו שיש מנוע טמפרטורה נוסף שפועל כאשר הטמפרטורה מגיע למעל 27 מעלות. להמשך הפיתוח מומלץ לבחון חיבור של המערכת למכשיר קירור בכדי לאפשר מתן הפעלה אוטומטית גם של מנוע הקירור במערכת.

# נספחים

## תיעוד קוד:

// C++ code

// Automatic drinking system

#include <LiquidCrystal.h>

#include <Servo.h>

const int TRIG\_PIN = 10;

const int ECHO\_PIN = 11;

const int SERVO\_PIN = 9;

const int LED\_PIN = 8;

LiquidCrystal lcd(7, 6, 5, 4, 3, 2);

//LiquidCrystal\_I2C lcd\_1(0x27,7, 6, 5, 4, 3, 2);

int pos = 0;

float previousdistance = 30;

int TEMP\_PIN = A0;

float distance\_us, distance;

int temp\_input, temp;

Servo servo;

void setup()

{

   //initialize the LCD

  //lcd.init();

  //turn on the blacklight and print a message

  //lcd.backlight();

  lcd.begin(16, 2);

  pinMode(TRIG\_PIN, OUTPUT);

  pinMode(ECHO\_PIN, INPUT);

  pinMode(SERVO\_PIN, OUTPUT);

  pinMode(LED\_PIN, OUTPUT);

  servo.attach(SERVO\_PIN);

  servo.write(0);

  Serial.begin(9600);

}

void loop()

{

 // Measure distance

  digitalWrite(TRIG\_PIN, LOW);

  delayMicroseconds(2);

  digitalWrite(TRIG\_PIN, HIGH);

  delayMicroseconds(10);

  digitalWrite(TRIG\_PIN, LOW);

  distance\_us = pulseIn(ECHO\_PIN, HIGH);

  distance = distance\_us\*0.017;

  // Measure temperature

  temp\_input = analogRead(TEMP\_PIN);

  temp = map(((temp\_input/1024.0)\*5-0.5)\*100,0,1023,-50,125);

  // Display temperature on LCD

  //lcd.clear();

  lcd.setCursor(0, 0);

  lcd.print("Tami 4 ");

  //lcd.setBacklight(1);

  //lcd.print("Temp: ");

  //lcd.print(temp);

  //lcd.print("C");

  // Activate servo motor and LED if object is within 20cm and temperature is below 10C

  // And if the distance had not change

  if (distance <= 20 && temp < 27 && abs(distance - previousdistance) > 5.00) {

    previousdistance = distance;

    lcd.setCursor(0, 1);

    lcd.print("Water pouring");

    //lcd.setBacklight(1);

    digitalWrite(LED\_PIN, HIGH);

    servo.write(360);

    delay(1000);

    servo.write(0);

    delay(4000);

    digitalWrite(LED\_PIN, LOW);

  }

  else {

    lcd.setCursor(0, 1);

    lcd.print(" ");

    digitalWrite(LED\_PIN, LOW);

    servo.write(0);

  }

  Serial.println("distance: ");

  Serial.println(distance);

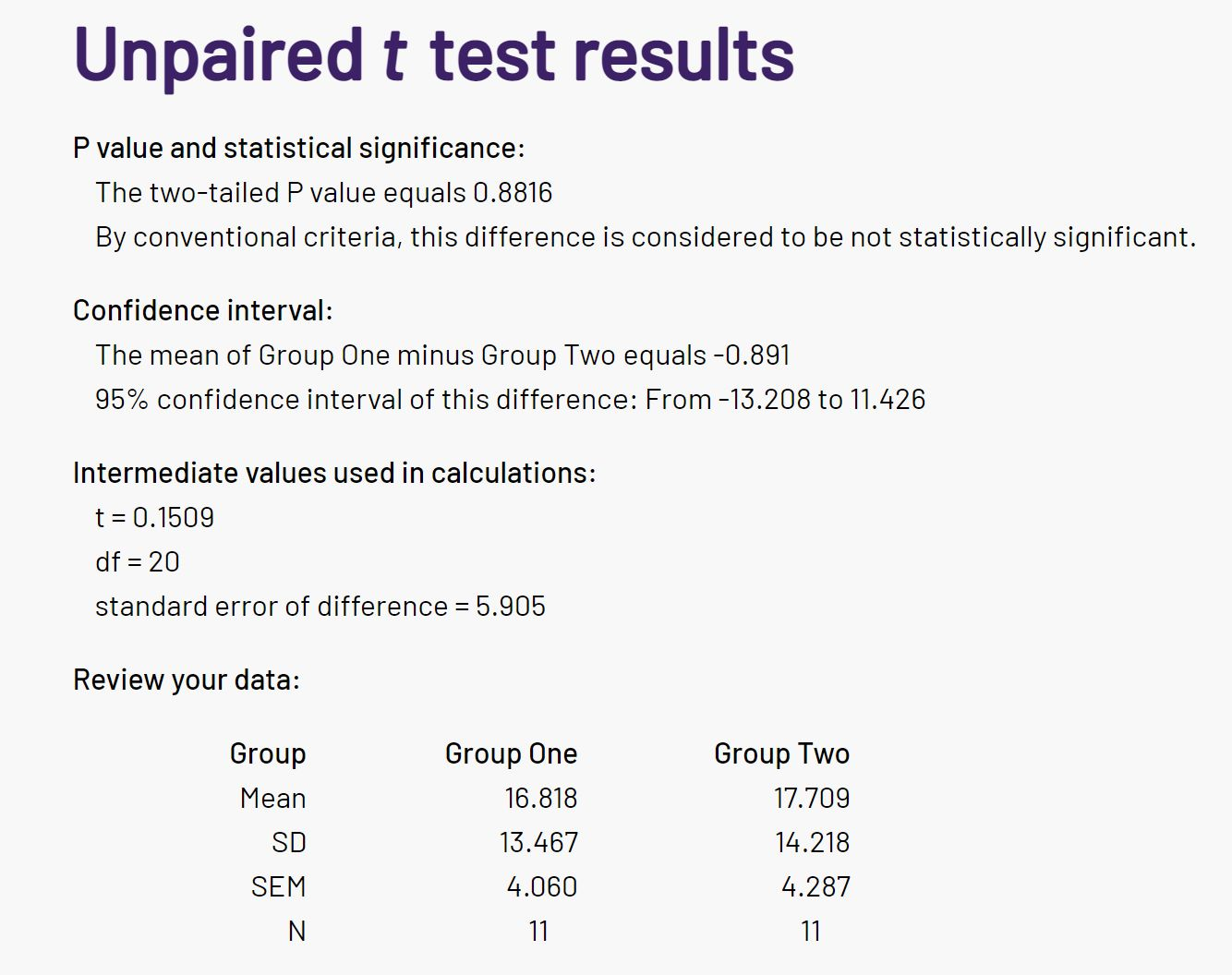
  Serial.println("temp: ");

  Serial.println(temp);

delay(500);

}

**מבחן t שבוצע:**

****