Darbą atliko:

Justas Nasvytis

**Įvadas į robotiką**

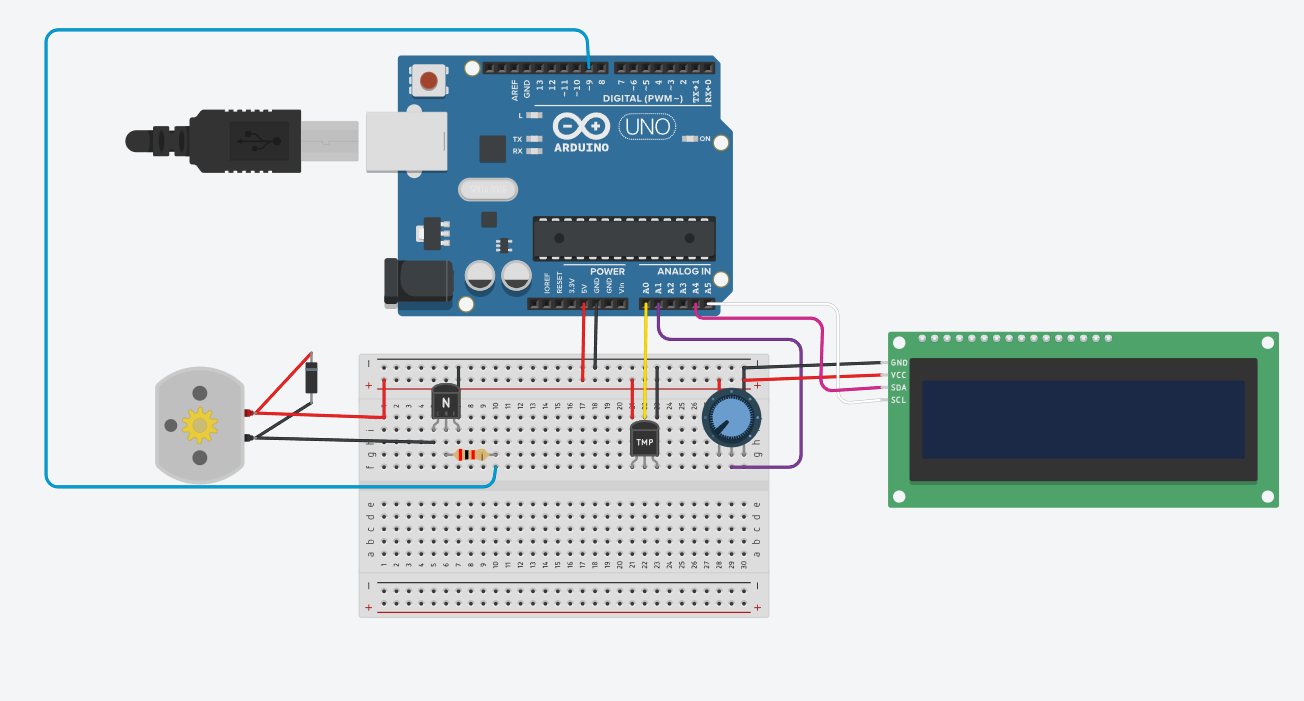
**Pirmasis namų darbas**

**Projektas:**Susimuliuotas ventiliatoriaus veikimas, kuris pradeda suktis, kai temperatūra viršija 25 0C arba oro drėgmė yra daugiau nei 50% . Tačiau jo sukimosi greitis priklauso ir nuo temperatūros ir oro drėgmės, kuo karščiau ir kuo didesnė oro drėgmė tuo greičiau sukasi. Be to prijungtas LCD ekranas, kuris rodo kokia drėgmė ir temperatūra

**Dizainas:**Arduino plokštė: Ji nuskaito jutiklio signalą, apdoroja duomenis ir priima sprendimus.  
Įvesties jutiklis: TMP36 temperatūros jutiklis ir drėgmės sensorius. Jie konvertuoja temperatūra ir oro drėgmę į signalą.  
Išvesties komponentas: DC motoriukas su NPN trazistoriumis ir diodu. Tranzistorius leidžia Arduino kontroliuoti didesnę srovė, o diodas apsaugo tranzistoriu nuo variklio sukuriamos atvirkštinės įtampos. LCD ekranas, kuris atvaizduoja temperatūra ir drėgmę.

**Komponentų sąrašas:**

1. Arduino Uno R3
2. TMP36 temperatūros jutiklis
3. Oro drėgmės matuoklis
4. DC motoriukas
5. Tranzistorius
6. Diodas
7. 220 omų rezistorius
8. Maketavimo plokštė
9. Sujungimo laidai
10. LCD ekranas 16 x 2 (I2C)

**Schema:  
**

**Kas veikia:**

Temperatūros jutiklis tiksliai nuskaito aplinkos temperatūra.

Oro drėgmes matuoklis nuskaito oro drėgmę.

Ventiliatoriaus įjungimas ir išjungimas veikia tinkamai.

Ventiliatoriaus sukimosi greitis proporcingai reaguoja į temperatūros ir oro drėgmės pokyčius.

**Kodas:**  
#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x20, 16, 2);  
  
int TmpSensorPin = A0;   
int motorPin = 9;   
int HumSensorPin = A1;

float voltage, temperatureC;  
int fanSpeed;  
void setup() {  
 lcd.init();  
 lcd.backlight();  
 pinMode(motorPin, OUTPUT);  
 Serial.begin(9600);

}

void loop() {

int TmpSensorValue = analogRead(TmpSensorPin);  
 int HumSensorValue = analogRead(HumSensorPin);  
   
 voltage = TmpSensorValue \* (5.0 / 1023.0);  
 temperatureC = (voltage - 0.5) \* 100.0;   
 int humidityPercentage = map(HumSensorValue, 0, 1023, 10, 70);

int baseFanSpeed = map(temperatureC, 24, 50, 0, 255);  
 if (humidityPercentage > 50) {  
 int humidityBoost = map(humidityPercentage, 50, 79, 0, 100);  
 fanSpeed = baseFanSpeed + humidityBoost;  
 }  
 else {  
 fanSpeed = baseFanSpeed;  
 }   
 lcd.clear();   
 fanSpeed = constrain(fanSpeed, 0, 255);  
 analogWrite(motorPin, fanSpeed);

lcd.setCursor(0, 0);   
 lcd.print("Temp: ");  
 lcd.print(temperatureC);  
 lcd.print(" C");

lcd.setCursor(0, 1);   
 lcd.print("Dregme: ");  
 lcd.print(humidityPercentage);  
 lcd.print(" %");

delay(1000);  
}