

INDUSTRIA ELEKTRONIKAREN ETA AUTOMATIKAREN
INGENIARITZAKO GRADUA

GRADU AMAIERAKO LANA

***TRESNA ELEKTRONIKOENTZAKO KARGAGAILU
FOTOVOLTAIKO BATEN DISEINUA,
SIMULAZIOA ETA PROTOTIPAKETA***

II ERANSKINA - PROGRAMAZIOA

Ikaslea: Jorge, Aristorena, Asier

Zuzendaria: Otaegi, Aizpeolea, Aloña

Ikasturtea: 2020-2021

Data: Bilbon, 2021eko ekainaren 25ean

Aurkibidea

ATMega 328P programazio kodea	1
-------------------------------------	---

ATMega 328P programazio kodea

```

#include <Wire.h> //I2C busa erabili ahal izateko liburutegia
#include <Adafruit_GFX.h> // Display grafikoetarako liburutegia
#include <Adafruit_SSD1306.h> // SSD1306 driverraren liburutegia

#define ZABALERA 128 //OLED pantailaren px zabalera
#define ALTUERA 64 // OLED pantailaren px altuera
#define OLED_RESET 4 //Beharrezko datua liburutegiarentzat
#define Puls01 2
Adafruit_SSD1306 oled(ZABALERA,ALTUERA,&Wire,OLED_RESET); //oled
aldagaia sortzen du liburutegia erabiliz
float vout = 0.0; //Vout
float vin = 0.0; //Vin
float vout2 = 0.0; //Vout2
float vout3 = 0.0; //Vout3
float vin2 = 0.0; //Vin2
float vin3 = 0.0; //Vin3
float ain=0.0;
float R1 = 100000.0; // R1 (100K)
float R2 = 10000.0; // R2 (10K)
float R3 = 20000.0; // R3 (20K)
float R4 = 10000.0; // R4 (10K)
int feedback = A1; //Feedback A1 pin analogikoa
int feedback2 = A2; //Feedback A2 pin analogikoa
int feedback3 = A3; //Feedback A3 pin analogikoa
int PWM = 3; //PWM seinalea 3 pin digitaletik
int pwm = 124; //PWM-aren DC-aren hasierako balioa
int ehunekoa;

void setup() {
  Wire.begin(); //I2C komunikazioa hasi
  oled.begin(SSD1306_SWITCHCAPVCC,0x3C);
  oled.setTextColor(WHITE);
  pinMode(feedback, INPUT);
  pinMode(feedback2, INPUT);
  pinMode(feedback3, INPUT);
  pinMode(Puls01, INPUT);
  pinMode(PWM, OUTPUT);

  TCCR2B = TCCR2B & B11111000 | B00000001; // 3 eta 11 pinen PWM
  maiztasuna 31372.55 Hz
}
void OLED (float vin,float ain){
  ehunekoa=(100/(4.3-3.3))*(abs(vin-3.3));
  oled.clearDisplay(); // pantaila garbitu
  oled.setCursor(27,15); // 0,0 kordenatuetan
  oled.setTextSize(2); // testuaren tamaina 2
  oled.setTextColor(WHITE); //testuaren kolorea txuria
  oled.print(ehunekoa); //ehunekoa inprimatu
  oled.print(" %");
  oled.setCursor(35,48); // 35,48 kordenatuetan
  oled.setTextSize(2);
  oled.print(ain);
  oled.print("mA");
  oled.drawRoundRect(15, 0, 88, 42,10, WHITE); //pilaren irudia
  marraztu
  oled.fillRoundRect(102,13,12,16,3,WHITE);

```

```
oled.drawRect(14, 44, 20, 20,WHITE); // Eguzki plaka irudia marraztu
oled.fillRect(16,46,4,4,WHITE);
oled.fillRect(22,46,4,4,WHITE);
oled.fillRect(28,46,4,4,WHITE);
oled.fillRect(16,52,4,4,WHITE);
oled.fillRect(22,52,4,4,WHITE);
oled.fillRect(28,52,4,4,WHITE);
oled.fillRect(16,58,4,4,WHITE);
oled.fillRect(22,58,4,4,WHITE);
oled.fillRect(28,58,4,4,WHITE);

oled.display(); //aurrekoa ikustarazi
delay(3000);
oled.clearDisplay(); //pantaila garbitu
oled.display();

}
void loop() {

float output = analogRead(feedback); //Feedback-aren balioa neurtu
vout = (output * 3.3) / 1024.0; //Vout kalkulua elikadura
tentsioaren menpe
vin = vout / (R2/(R1+R2)); // Vin tentsio banatzailearen kalkulua
float output2 = analogRead(feedback2); //Feedback-aren balioa neurtu
vout2 = (output2 * 3.3) / 1024.0; // Vout2 kalkulua
vin2 = vout2 / (R4/(R3+R4));
float output3 = analogRead(feedback3); //Feedback-aren balioa neurtu
vout3 = (output3 * 3.3) / 1024.0; // Vout3 kalkulua
vin3 = vout3 / (R4/(R3+R4));
ain=((vin2-vin3)/0.22)*1000; //korrontearen balioa mA-tan

//vin balioa 4.22 baino txikiagoa bada pwm seinalearean duty-cycle
txikitu
if (4.22 > vin){
pwm = pwm-1;
pwm = constrain(pwm, 1, 254);
}

//vin balioa 4.22 baino handiagoa bada pwm seinalearean duty-cycle
handitu
if (4.22 < vin){
pwm = pwm+1;
pwm = constrain(pwm, 1, 254);
}
analogWrite(PWM,pwm); //PWM seinalea sortu

if (digitalRead(Puls01)==HIGH){ //Pultsadorea sakatzerakoan
OLED(vin,ain); //OLED funtzioa
}

}
```