

PAMETNO GRIJANJE – SUSTAV ZA REGULACIJU TEMPERATURE U PROSTORIJI

Workshop on Embedded Systems 2020

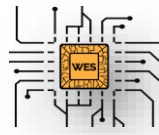
Tim Povuci-Potegni

Natan Turčinović

Marin Luštica

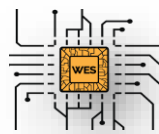
Leon Kranjčević

17.12.2020 u Zagrebu



Sadržaj

1. Projektni zadatak	2
2. Izrada projektnog zadatka.....	3
3. Dokumentacija	Error! Bookmark not defined.



1. Projektni zadatak

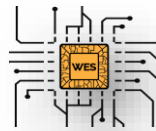
Sudionicima radionice Workshop on Embedded Systems 2020 predložen je zadatak dizajniranja i izrade sustava za reguliranje temperature u prostoriji. Sustav ostvaren pomoću mikrokontrolera koji će pomoću temperaturnog senzora i WIFI modula mjeriti i signalizirati promjenu temperature u prostoriji.

Preko web aplikacije korisnik zadaje temperaturu na koju želi da se prostorija zagrije ili ohladi, a u procesu snižavanja ili povećanja temperature sustav pomoću LED dioda javlja u kojem je stanju (stanje GRIJANJE ili stanje HLAĐENJE). Kad je željena temperatura postignuta, sustav miruje (LED diode se gase, stanje MIROVANJA).

U svrhu točnijeg signaliziranja stanja, uključili smo donju i gornju granicu (mrtva zona) temperaturnog pomaka kako ne bi došlo do naglih promjena stanja pri graničnim vrijednostima.

Sustav je povezan s WiFi modulom čija bi svrha bila povezivanje s ostalim pametnim uređajima u prostoriji, u našem slučaju prvenstveno klime. Zbog nedostatka pametne klime, mobitel služi kao točka razmjene informacija.

Zadatak je imao rok za izradu od 15.12 do 17.12 u kojem su vremenu osigurane prostorije na Fakultetu elektrotehnike i računarstva u Zagrebu.



2. Izrada projektnog zadatka

Za izradu projektnog zadatka korišteni su sljedeći materijali:

- 1) Croduino NOVA 2 mikrokontroler
- 2) 2 LED diode (plava i crvena)
- 3) 2 otpornika otpora 1000 ohm
- 4) WiFi modul ESP8266
- 5) DHT11 temperaturni senzor
- 6) Žice

Preko dva otpornika (1k ohm) spojili smo diode na digitalne ulaze mikrokontrolera kao i DHT11 temperaturni senzor.

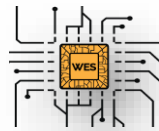
```
int LED_red = 16;  
int LED_blue=14;  
  
#define DHT11PIN 12
```

Slika 1 Arduino kod definiranja pinova mikrokontrolera

Nakon spajanja strujnog kruga i pripadajućih elemenata definirali smo početno stanja LED diode, to jest pri paljenju su u niskoj razini, to jest 0. Nadalje, inicijalizirali smo WiFi modul s danom lozinkom i imenom WiFi hotspota kojeg odašilje modul.

```
pinMode(LED_red, OUTPUT);  
pinMode(LED_blue, OUTPUT);  
boolean conn = WiFi.softAP(AP_Name, AP_Password);  
server.begin();  
  
digitalWrite(LED_red, LOW);  
digitalWrite(LED_blue, LOW);
```

Slika 2 Arduino kod definiranja postavljanja sustava



Čitanjem korisnikovog unosa željene temperature ulazimo u petlju koja pali i gasi LED diode ovisno o mjerenju koje šalje temperaturni senzor.

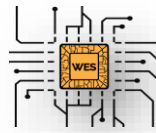
```
int chk = DHT11.read(DHT11PIN);  
int currentTemperature = (float)DHT11.temperature;
```

Slika 3 Arduino kod čitanja temperature u prostoriji

Nadalje, pri očitavanju temperature u prostoriji provjeravamo u koje stanje da sustav prebacimo. Ukoliko je korisnikova temperatura manja od očitane temperature sustav pali plavu LED diodu što signalizira hlađenje prostorije (uključivanje klime), a ukoliko je unesena temperatura veća od očitane, pali se crvena LED dioda i sustav prelazi u stanje grijanja (uključivanje radijatora).

```
if(currentTemperature < setTemperature - lowBoundary) { //heating  
    digitalWrite(LED_red,HIGH); // LED diode is signaling heating  
    heating = true; // flag that indicates system is in heating mode  
} else if (currentTemperature > setTemperature + highBoundary)  
    digitalWrite(LED_blue,HIGH); // LED diode is signaling cooling  
    cooling=true; // flag that indicates system is in cooling down  
}  
if(heating && currentTemperature >= setTemperature) { // heating  
    digitalWrite(LED_red,LOW);  
    heating = false;  
}  
if(cooling && currentTemperature <= setTemperature) { // cooling  
    digitalWrite(LED_blue,LOW);  
    cooling = false;  
}
```

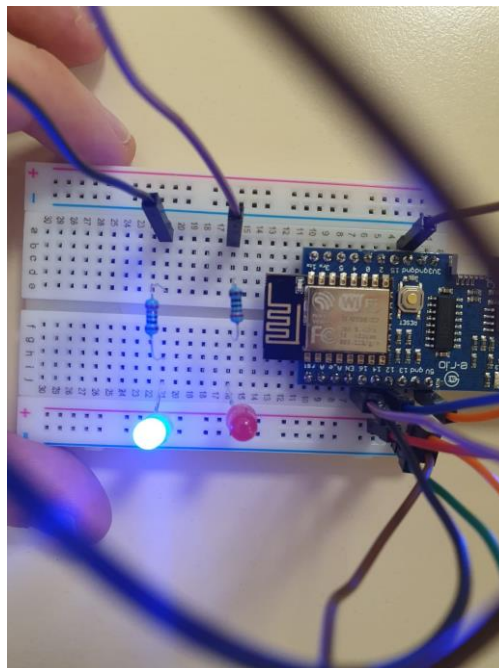
Slika 4 Arduino kod promjene stanja sustava



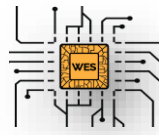
Naposlijetku, određena dioda gasi se kad se temperatura prostorije izjednači s korisnikovom željenom temperaturom u granicama odstupanja.

```
if(heating && currentTemperature >= setTemperature)
    digitalWrite(LED_red, LOW);
    heating = false;
}
if(cooling && currentTemperature <= setTemperature)
    digitalWrite(LED_blue, LOW);
    cooling = false;
}
}
```

Slika 5 Arduino kod za resetiranje stanja sustava



Slika 6 Strujni krug s pripadajućim elementima



3. Literatura

<https://e-radionica.com/hr/blog/2019/09/11/croduino-nova-kao-apaccess-point/>

<https://www.circuitbasics.com/how-to-set-up-the-dht11-humidity-sensor-on-an-arduino/>