

Upravljanje toplinskim procesom putem Web sučelja

Škrobar, Fabijan

Undergraduate thesis / Završni rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University North / Sveučilište Sjever**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:122:379150>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2024-11-27**



Repository / Repozitorij:

[University North Digital Repository](#)





Sveučilište Sjever

Završni rad br. 040/MEH/2024

Upravljanje toplinskim procesom putem Web sučelja

Fabijan Škrobar, 0336053418

Varaždin, rujan 2024. godine



Sveučilište Sjever

Odjel za Mehatroniku

Završni rad br. 040/MEH/2024

Upravljanje toplinskim procesom putem Web sučelja

Student

Fabijan Škrobar, 0336053418

Mentor

Miroslav Horvatić, dipl. ing.

Varaždin, rujan 2024. godine

Predgovor

Glavni cilj ovog projekta "Upravljanje toplinskim procesom putem Web sučelja" je demonstrirati praktičnu primjenu mikrokontrolera u sustavima upravljanja. S razvojem IoT (Internet of Things) tehnologija, mikrokontroleri poput ESP8266 omogućuju jednostavnu i povoljnu integraciju uređaja u mrežne sustave. U projektu se koristi ESP8266 kao mozak operacije jer ima sposobnost primanja i slanja WiFi signala.

Web sučelje nam omogućava pristup fizički udaljenim lokacijama u stvarnom vremenu putem standardnog internetskog preglednika, ukoliko ima WiFi povezanosti. Iz sučelja možemo poslati upravljački signal koji uključuje ili isključuje aktuator procesa te očitati trenutno stanje procesa.

Kroz ovaj rad prikazano je kako se koristeći jednostavne alate i tehnologije mogu kreirati sustavi nadzora i upravljanja, s mogućim primjenama u industriji, automatizaciji ili nadzoru.

Projekt također pruža uvid u temeljne aspekte dizajna Web stranica, upravljanja podacima u stvarnom vremenu, te način na koji se mogu povezati mikrokontroleri i mrežni sustavi za stvaranje fleksibilnih i učinkovitih rješenja.

Sažetak

Ovaj rad opisuje implementaciju sustava za daljinsko upravljanje tehnološkim procesima putem IoT tehnologije. Korištenjem mikroupravljača ESP8266, senzora, aktuatora i ostalih komponenti, omogućeno je praćenje i upravljanje parametrom temperature preko Web sučelja. Sustav omogućuje operaterima da u stvarnom vremenu prate stanje procesa i po potrebi izvršavaju prilagodbe, bez potrebe za fizičkom prisutnošću. Uređaji komuniciraju putem Wi-Fi signala, a Web sučelje pruža jednostavan grafički prikaz podataka i kontrolu, čime se povećava efikasnost i sigurnost procesa.

Nakon planiranja i prikupljana komponenti procesa započelo je testiranje mikroupravljača. Korištenjem jednostavnijeg programa prvo su se testirale programske biblioteke (library) temperaturnog senzora DS18B20, one pretvaraju analogne veličine u prepoznatljivije stupnjeve Celzijeve.

Zatim se povezuje mikroupravljač ESP8266 s Web serverom korištenjem programskih biblioteka i namještanjem „baud rate-a“. Kada se uspješno povezalo uključen je i HTML kojim se ispisao izgled Web stranice. HTML također zahtijeva testiranje za uspješnu implementaciju.

Logika procesa tj. logika upravljanja se programira te povezuje s HTML-om. Logika se provjerava pomoću LED žaruljica koje se kasnije koriste kao signalne.

Na Web sučelju mogu se odabrati sljedeći načini rada:

- Hlađenje/grijanje – uključuju se ventilator ili grijač
- Mirovanje – sve se gasi
- Auto – održava se upisana temperatura s histerezom ($\pm 1^{\circ}\text{C}$)

Kad je utvrđeno da sve funkcionira na logičkoj i softverskoj razini izrađeno je kućište i spojeni aktuatori procesa preko elektromagnetskih releja. Kod završnog spajanja testiraju se nazivne struje i naponi pinova mikroupravljača kao i različite komponente procesa.

Nakon svih spanjanja i testiranja sve je implementirano u kućište, doveden je izvor napajanja odvojen od računala i putem Web stranice može se upravljati procesom.

Summary

This paper describes the implementation of a system for remote management of technological processes using IoT technology. By utilizing the ESP8266 microcontroller, sensors, actuators, and other components, it enables monitoring and control of temperature parameters through a Web interface. The system allows operators to monitor the process in real-time and make adjustments as needed without requiring physical presence. Devices communicate via Wi-Fi signals, and the Web interface provides a simple graphical display of data and controls, increasing the efficiency and safety of the process.

After planning and gathering the necessary components, testing of the microcontroller began. Using a simpler program, the libraries for the DS18B20 temperature sensor were first tested to ensure they correctly converted analog values into recognizable degrees Celsius.

Next, the ESP8266 microcontroller was connected to the Web server using the appropriate libraries and configuring the baud rate. Once the connection was established successfully, HTML was implemented to display the Web page. The HTML setup was also tested to ensure proper implementation.

The control logic was then programmed and integrated with HTML, with verification carried out using LED lights, which would later serve as signals. The control logic operates as follows, based on selections made from the web interface:

- **Cooling/Heating:** The fan or heater is activated.
- **Standby:** All systems are turned off.
- **Auto:** The set temperature is maintained with hysteresis ($\pm 1^{\circ}\text{C}$).

After confirming that everything functioned correctly at the logical and software levels, the casing was built and actuators were connected through electromagnetic relays. During the final assembly, the nominal currents and voltages of the microcontroller pins and various process components were tested.

With all connections and tests completed, the entire assembly was placed in the casing, connected to a power source separate from the computer, and the process was operated via the Web interface.

Popis korištenih kratica

HTML	HyperText Markup Language Sintaksa za obilježavanje hipertekstualnih dokumenata.
IoT	Internet of Things
PMW	Pulse Width Modulation tj. pulsno širinska modulacija
LED	Light Emiting Diode
NC	Normally Closed
NO	Normally Open

Sadržaj

1.	Uvod.....	1
2.	Komponente procesa.....	3
2.1.	ESP8266 mikroupravljač.....	3
2.2.	Elektromagnetski relej za grijač procesa.....	5
2.3.	Releji s dodanim NPN tranzistorom.....	6
2.4.	Temperaturni senzor s izolacijom DS18B20	7
2.5.	Grijač sa zavojnicom	8
2.6.	Ventilator / DC motor	9
2.7.	Izvori napajanja – Akumulator i pinovi mikroupravljača	10
2.8.	Signalne LED žaruljice	11
2.9.	Ostale komponente	12
2.10.	Kućiče.....	12
3.	Električni dio.....	14
3.1.	Spajanje temperaturnog senzora DS18B20 na ESP8266 mikroupravljač	14
3.2.	Spajanje ventilatora i grijača preko releja	14
3.3.	Blokovski prikaz procesa	15
3.4.	Shematski prikaz sklopa procesa.....	16
4.	Programski dio	17
4.1.	Arduino IDE program	17
4.1.1.	Definiranje varijabli i uključivanje programskih biblioteka (library).....	17
4.1.2.	Funkcija void setup()	18
4.1.3.	Funkcija void loop().....	20
4.2.	Serijski monitor	24
4.3.	Web sučelje i primjer korištenja sustava.....	24
5.	Zaključak.....	27
6.	Literatura.....	28
	Popis slika	29
	Popis programskih sekvenci	30

1. Uvod

Svaki tehnološki proces zahtijeva kontinuirano praćenje, održavanje i upravljanje kako bi se osigurao optimalan rad i spriječile potencijalne greške ili zastoji. S obzirom na sve veću potrebu za uštedom vremena, kao i često fizičku nedostupnost lokacija na kojima se procesi odvijaju, razvijaju se nova rješenja poput automatizacije te daljinskog upravljanja. Uvođenje automatizacije smanjuje potrebu za fizičkom prisutnošću operatera, omogućujući nadzor i kontrolu procesa s udaljenih lokacija. Korištenjem Web sučelja, operateri mogu pratiti i upravljati procesima u stvarnom vremenu, s bilo kojeg mjesta na svijetu, pod uvjetom da postoji Wi-Fi povezanost.

Takva tehnologija upravljanja procesima putem interneta pripada širem konceptu poznatom kao Internet of Things (IoT), gdje uređaji komuniciraju putem interneta, razmjenjuju podatke i izvršavaju naredbe bez potrebe za izravnom ljudskom intervencijom. IoT omogućava jednostavno povezivanje fizičkih objekata s digitalnim svijetom, čineći procese dostupnima i upravljivima na daljinu. Web sučelje koje se koristi u IoT sustavima pruža sveobuhvatan uvid u željene parametre tehnološkog procesa, kojima se vidi trenutno stanje. Operaterima omogućuje praćenje parametara poput temperature, tlaka, razine tekućine ili bilo kojeg drugog važnog pokazatelja u stvarnom vremenu, što im daje opciju donošenja pravovremenih odluka, tj. u stvarnom vremenu.

Isto tako Web sučelje nudi i mogućnost aktivne kontrole tehnoloških procesa, omogućujući korisnicima da putem prikazanih naredbi na ekranu izravno upravljaju procesima. Na primjer, operater može daljinski pratiti stanje tekućine u nekoj posudi te prema potrebi promijeniti parametre procesa. Ukoliko je senzorom izmjerena temperatura izvan željenih granica, putem sučelja se može poslati naredba koja uključuje ili isključuje grijače ili hlađenje, čime se regulira stanje sustava. Na taj način operateri imaju kontrolu nad sustavima, bez obzira na njihovu fizičku udaljenost, čime se povećava efikasnost i sigurnost procesa.

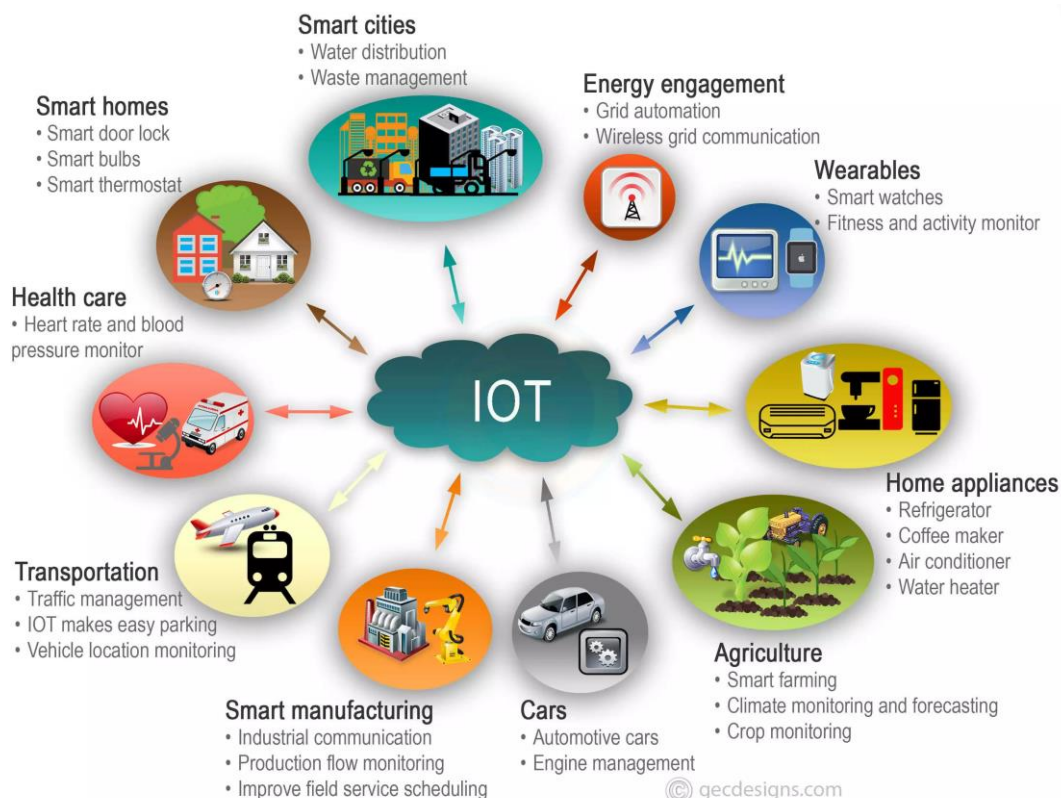
Jedan od elemenata ovakvog sustava je senzorka koja prikuplja podatke. Na primjer, senzor temperature neprekidno prati toplinsko stanje tekućine u posudi te šalje podatke mikroupravljaču. Ovi podaci se potom obrađuju i prikazuju na web sučelju u obliku lako razumljivih brojki ili grafičkog prikaza, kao što su stupnjevi Celzijevi. Operater u svakom trenutku može vidjeti trenutnu temperaturu. Takav prikaz stanja omogućava brz i točan uvid u proces.

Mozak operacije poput mikroupravljača oznake ESP8266 ima sposobnost slanja i primanja podataka putem Wi-Fi signala te omogućuje prijenos podataka na web sučelje. Njegova glavna funkcija je povezivanje fizičkog sustava sa softverskim dijelom – on prima signale sa senzora, obrađuje ih prema unaprijed programiranim algoritmima, a zatim rezultate šalje korisniku putem internetskog preglednika. Osim toga mikrokontroler šalje odabrane naredbe aktuatorima.

Aktuatori, kao izvršni elementi sustava, provode naredbe, poput uključivanja ili isključivanja grijača, ventilatora ili drugih uređaja koji reguliraju proces. Na primjer, ako senzori detektiraju da temperatura prelazi zadanu vrijednost, ESP8266 šalje signal aktuatorima da uključe hlađenje i vrata sustav u zadanu stanje.

U ovom projektu detaljno je opisan proces implementacije jednog takvog sustava, podijeljenog u više glavnih cjelina. Prva cjelina obuhvaća teorijski dio, u kojem se objašnjavaju osnovni pojmovi vezani za IoT tehnologiju, mikroupravljače i senzorske sustave, kao i uvod u zadatak tj. projekt. Druga cjelina opisuje korištene fizičke komponente – ESP8266 mikroupravljač, senzore, aktuatorске sustave i Wi-Fi komunikaciju. Treća cjelina bavi se prikazom shema i dijagrama sustava, objašnjavajući način na koji su različiti dijelovi povezani i kako komuniciraju međusobno. Četvrta cjelina obrađuje programska rješenja, koja uključuju pisanje koda za ESP8266, postavljanje web sučelja i osiguravanje dvosmjerne komunikacije između korisnika i sustava.

Kroz ovaj projekt, cilj je prikazati kako se korištenjem jednostavnih, ali učinkovitih tehnologija može ostvariti napredno rješenje za daljinsko upravljanje procesima. Integraciju senzora, mikroupravljača i web sučelja u jedan sustav koji služi svojoj svrhi. Ovakva rješenja predstavljaju realnost upravljanja procesima u digitalno povezanom svijetu.



Slika 1.1, IoT

2. Komponente procesa

Pošto se radi o sustavu upravljanja koji se samoodržava ili operira sa udaljenosti komponente sustava trebale bi biti pouzdane i izdržljive.

Komponente potrebne za realizaciju sklopa su:

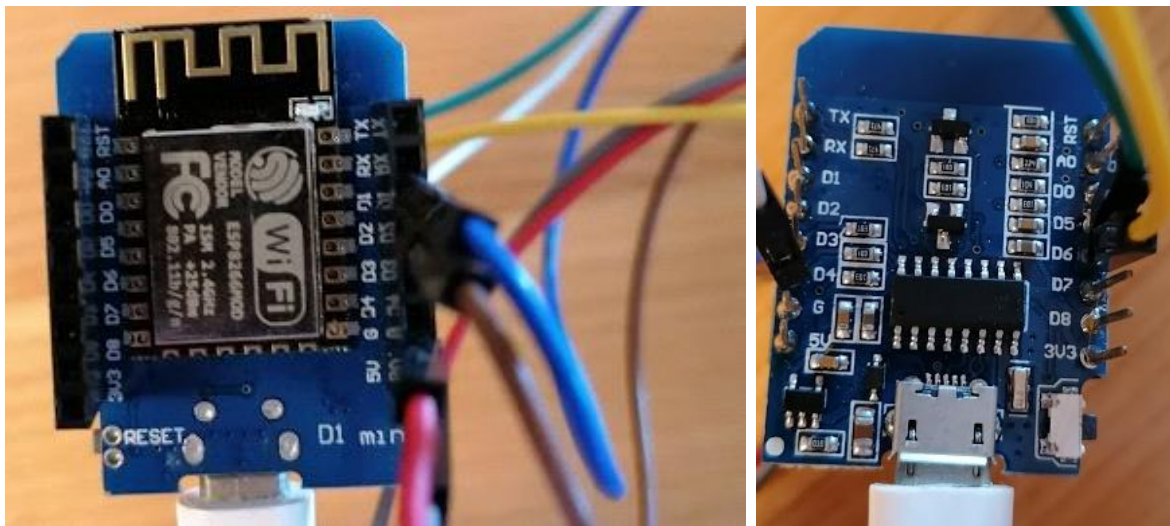
- ESP8266 mikroupravljač,
- Elektromagnetski relej (x2),
- NPN tranzistor,
- Akumulator 12V,
- 5V/3.3V izvor napajanja,
- Temperaturni senzor s izolacijom DS18B20,
- LED (Light Emiting Diode) žaruljice
- Otpornik 4.7 kOhma, otporni grijač, DC ventilator, eksperimentalna pločica i spojne žice.

2.1. ESP8266 mikroupravljač

ESP8266 mikroupravljač radi prema programskim naredbama koje su učitane sa Arduino IDE programa na računalu.

Mikroupravljač se napaja pomoću micro-USB kabla ili kroz 3V3 i G pinove. Sastoji se od 9 ulazno-izlaznih digitalnih pinova (D0-D8) koji imaju sposobnost PWM (Pulse Width Modulation) tj. pulsno širinske modulacije i jednog analognog ulaza A0. Slika 2.1 prikazuje raspored pinova.

Uspostavom mreže s kojom je mikroupravljač povezan stvara se WiFi komunikacija između sučelja i čipa. Čip zatim procesira informacije i prema programu izvršava naredbe.



Slika 2.1, Izgled i pinovi ESP8266

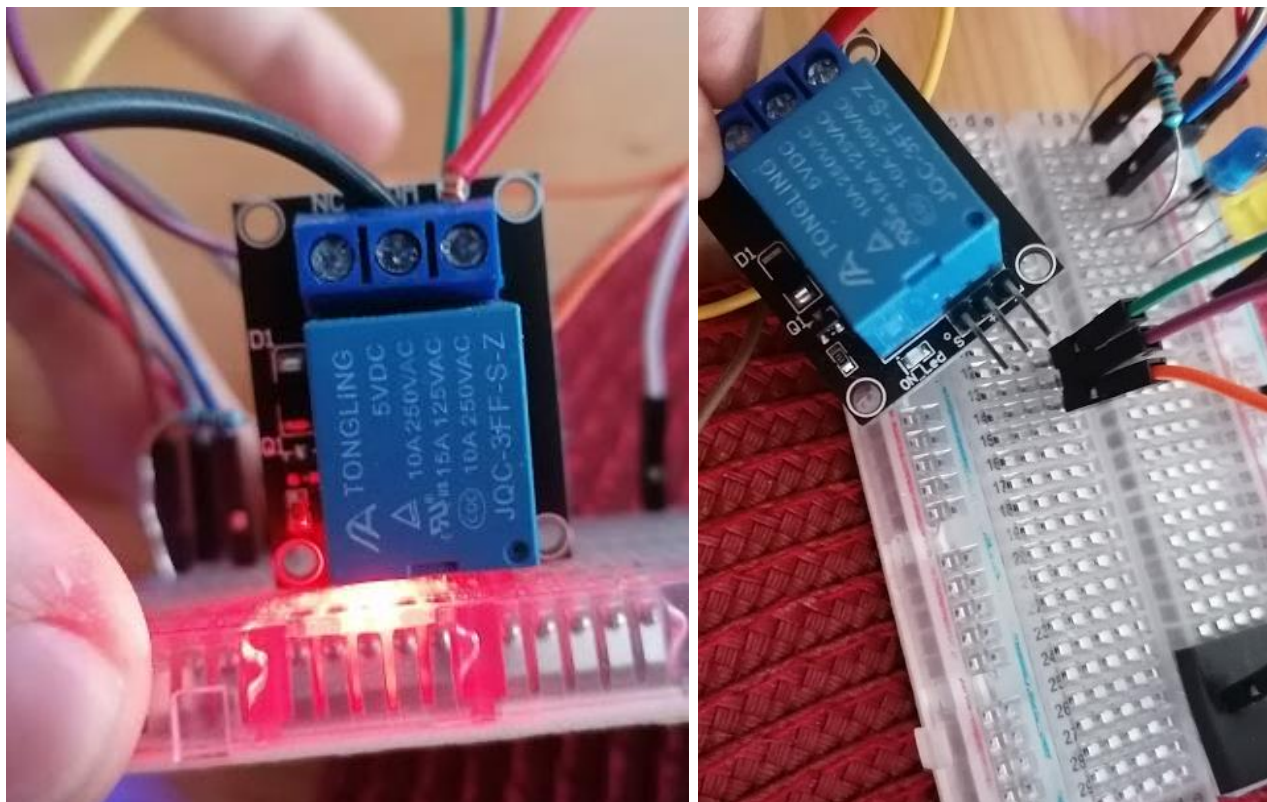
ESP8266 mikroupravljač upravlja uređajem te prima i šalje informacije, te stvara WiFi povezanost sučelja i ulaznih te izvršnih elemenata. Temperaturni senzor mu šalje analogni signal na A0 pin koji se pretvara u prepoznatljive stupnjeve Celzijeve , a zatim se prikazuje na WEB sučelju. Sa Web sučelja mikroupravljač prima upravljačke informacije koje se također prema programu obrađuju. Kroz digitalne pinove prvo elektromagnetski releji, a zatim aktuatori procesa dobivaju te upravljačke informacije.

Karakteristike

- Procesor: L106 32-bit
- Napajanje s micro-USB
- Radni napon: 3.3V
- Raspon analognog ulaza A0: 0 - 3.3V
- Sposobnost WiFi komunikacije

2.2. Elektromagnetski relej za grijač procesa

Dobivanjem signala sa mikroupravljača otvara se ili zatvara odvojeni strujni krug kojim teče jača struja. Elektromagnetski relej radi na principu elektromagneta tj. zavojnice koja s protokom struje dobiva magnetna svojstva. NC (normally closed) pin zatvara vanjski strujni krug dok relej nema signala, a NO (normally open) pin se aktivira signalom sa mikroupravljača.



Slika 2.2, Relej za grijač

Relej omogućava upravljanje grijača pomoću signala sa pinova mikroupravljača koji su slabiji od struje koja se želi koristiti. Na slici 2.2 prikazuje se spajanje grijača (COM, NO) i upravljačkih signala (S, +, -).

Tehničke karakteristike korištenog releja su:

- Radni napon: 5V DC
- Maksimalna upravljiva struja: 10A, 250 V AC

2.3. Relej s dodanim NPN tranzistorom

NPN tranzistor se sastoji od 3 kontakta; baza, emiter i kolektor. Dolaskom slabijeg signala na bazu tranzistora zatvara se jači strujni krug od kolektora prema emiteru. Na kolektor je spojeno napajanje, a na emiter uzemljenje upravljačkog kruga releja.



Slika 2.3, Ispitivanje potrebne struje multimetrom, NPN tranzistor i Relej

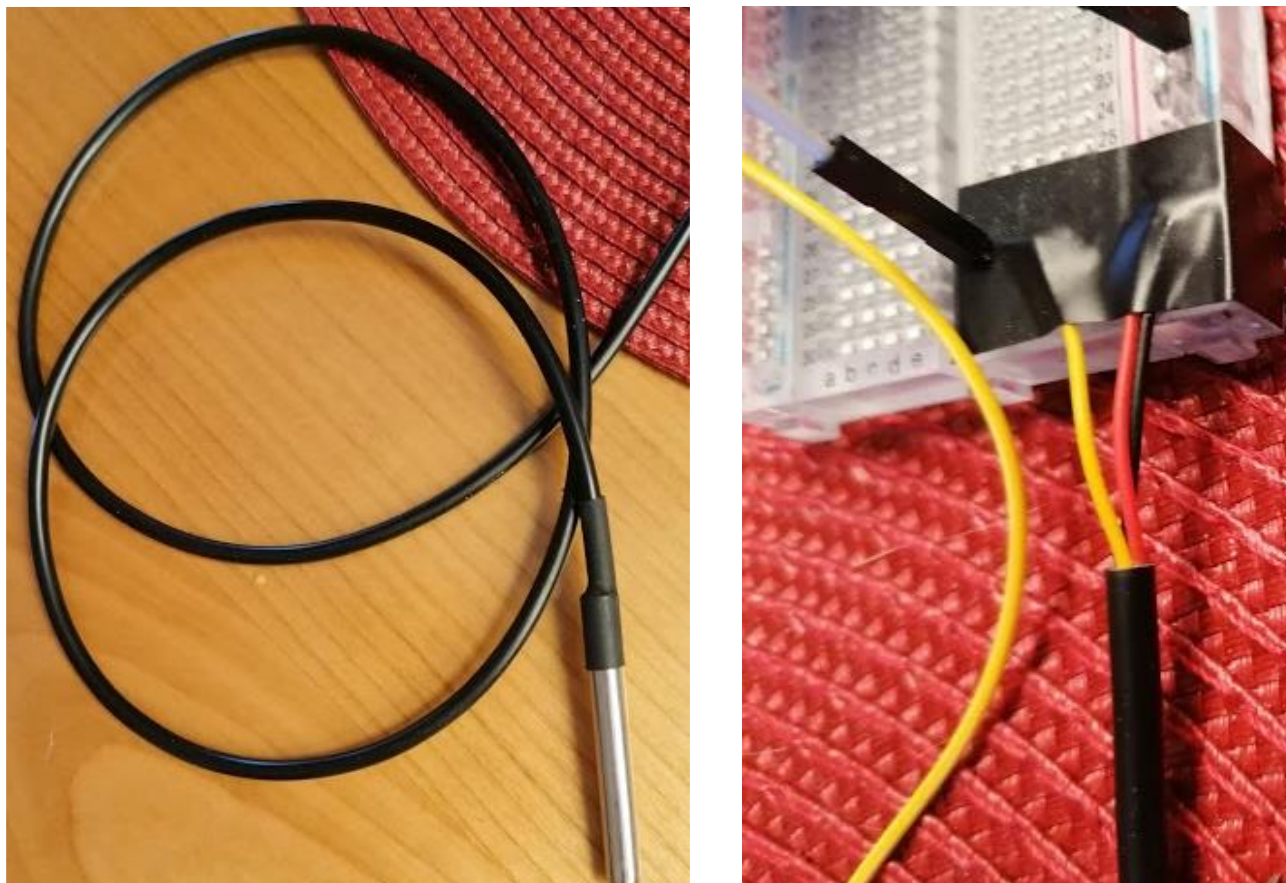
Zavojnica releja zahtijeva 32mA električne struje, a najveća izlazna struja pinova na ESP8266 mikroupravljaču je 12mA. Stoga NPN tranzistor služi kao poveznica izlaznog pina i upravljačkog strujnog kruga releja. Relej je preko tranzistora spojen na pin 5V mikroupravljača čija izlazna struja odgovara zavojnici releja. Baza tranzistora je na D2 izlaznom pinu.

Karakteristike releja:

- Radni napon: 5V DC
- Maksimalna upravljiva struja: 10A, 250 V AC

2.4. Temperaturni senzor s izolacijom DS18B20

Temperaturni senzor s izolacijom sastoji se od napajanja i signalne žice. Napaja se 5 V izvorom, a izlazna veličina se mijenja s obzirom na temperaturu, tj. temperaturom se mijenja otpor koji određuje jačinu signala. Spaja se na A0 pin mikroupravljača.



Slika 2.4, Temperaturni senzor i njegove žice

Temperaturni senzor mjeri tekućine kao analogni ulaz na mikroupravljaču. Taj ulaz temperaturnog senzora se zatim obrađuje u mjernu jedinicu stupnjeva Celzijevih. Na slici 2.4 vide se crvena i crna žica kao napajanje senzora i žuta žica kao signalna.

Karakteristike temperaturnog senzora:

- Raspon izlaznog signala: 0 - 3.3V
- Radni napon: 5V
- Vodootporna izolacija

2.5. Grijač sa zavojnicom

Grijač sa zavojnicom je električni uređaj koji pretvara električnu energiju u toplinsku. Prolaskom struje kroz zavojnicu stvara se električni otpor što rezultira zagrijavanjem materijala i okoline. Dobivanje topline na grijaču ne ovisi o smjeru prolaska električne struje, a ima samo dva stanja; uključeno ili isključeno. Radi na principu Joulovog prvog zakona, tj. snaga grijanja je proporcionalna umnošku kvadrata protoka električne struje i otpora materijala

$$P = I^2 R. \quad (1)$$

Snaga grijača mjeri se u Wattima ili Joul po sekundi ($W = J/s$).



Slika 2.5, Izgled grijača

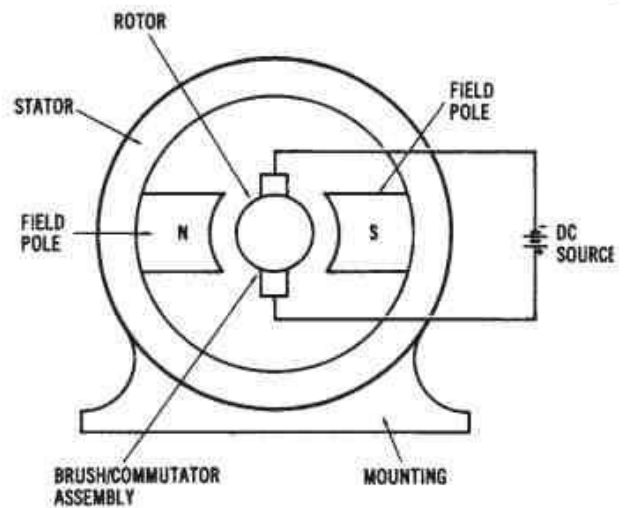
Izvršni element tj. grijač koji služi za zagrijavanje tekućine ili smjese u posudi. Spojen je preko releja, a može se upravljati putem Web sučelja. Na automatskom načinu rada uključuje se kad temperatura padne ispod zadane temperature.

Karakteristike grijačeg elementa:

- Radni napon: 12 V
- Radna struja: 2 A
- Snaga grijača: 24 W

2.6. Ventilator / DC motor

Tri glavne komponente DC motora su rotor, stator i komutator. Rotor se nalazi s unutarnje strane motora, dok je stator smješten s vanjske strane. Rotor sadrži namotaje zavojnice koje pokreće istosmjerna struja, a stator sadrži trajne magnete ili elektromagnetske namotaje. Kad motor pokreće istosmjerna struja, unutar statora se stvara magnetsko polje koje privlači ili odbija magnete na rotoru. To uzrokuje rotaciju rotora. Komutator omogućava daljnje gibanje, tj. zamjenu smjera struje kad rotor dođe do položaja privlačnosti magnetskog polja. Lopatice na motoru stvaraju strujanje zraka kojim se proces hladi.



Slika 2.6, Ventilator i shema DC motora [2]

Ventilator je aktuator procesa koji omogućava hlađenje smjese ili tekućine u posudi. Spaja se preko releja, a upravlja se sa Web sučelja. Na automatskom načinu rada uključuje se kad temperatura naraste iznad zadane temperature.

Karakteristike ventilatora:

- Radni napon: 12 V DC
- Radna struja: 0.14 A

2.7. Izvori napajanja – Akumulator i pinovi mikroupravljača

Akumulator je punjiva vrsta baterija kojoj se električna energija pohranjuje u kemijsku i zatim se prema potrebi, može opet pretvoriti u električnu.

Pinovi mikroupravljača koji je izvan funkcije samo preusmjeravaju napajanje tog mikroupravljača na 5.0V, 3.3V i GND pinove.



Slika 2.7, Akumulator i naponski pinovi mikroupravljača

Akumulator preko releja napaja aktuator procesa, tj. ventilator i grijač.

Pinovi mikroupravljača služe kao napajanje ESP8266 mikroupravljača i releja koji koristi tranzistorsku sklopku.

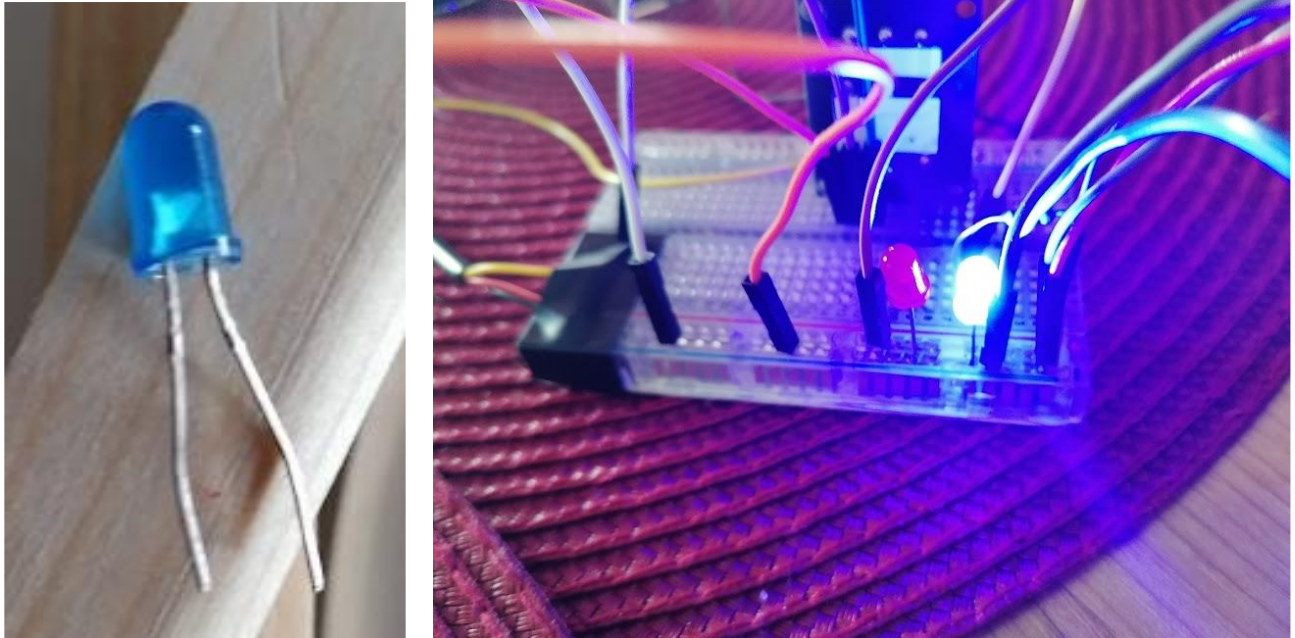
ESP8266 se napaja s 3.3.V, za relej je potrebno 5V.

Karakteristike akumulatora:

- Radni napon: 12V
- Električni naboj: 9Ah

2.8. Signalne LED žaruljice

LED žaruljica je poluvodički element koji prolaskom struje emitira svjetlost. Količina svjetlosti ovisi o naponu i struji.



Slika 2.8, LED žaruljice

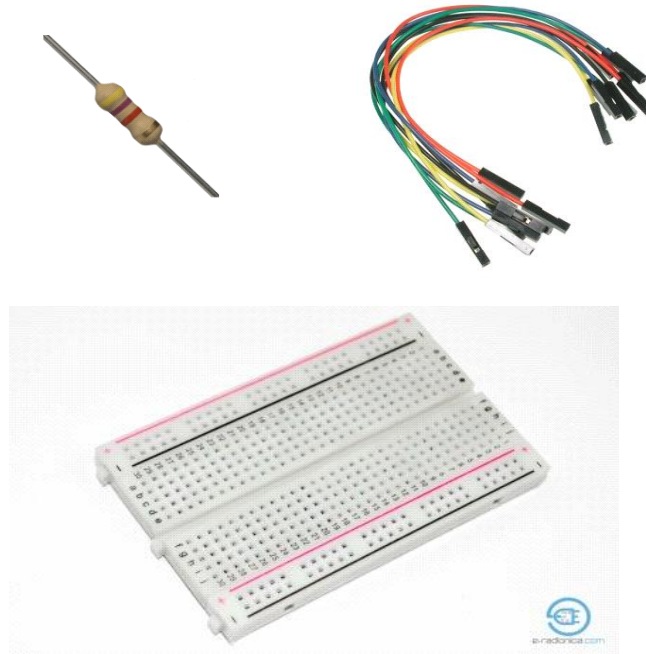
LED žaruljice vizualno signaliziraju trenutno stanje procesa. Crvena se upali kod grijanja, a plava kod hlađenja. Ugašene obje LED žaruljice signaliziraju stanje mirovanja.

Karakteristike LED žaruljice:

- Radni napon: 3.3V
- Radna struja: 12mA

2.9. Ostale komponente

Služe za spajanje i povezivanje glavnih elemenata procesa



Slika 2.9, Otpornik, Spojne žice, Eksperimentalna pločica

2.10. Kućište

Za izradu kućišta korištena je jedna daska izrezana na potrebne dijelove i ploča iverice kao podloga. Daske su pričvršćene vijcima i vrućim ljepilom. Unutarnji dio kućišta ima prostora za posudu u kojoj se nalazi tekućina. Oko metalne posude je mjesto za komponente sustava i utor za akumulator.

Dimenzije

Podloga (mm): 20x377x406mm

Vanjske daske (mm):

- 17x118x406 (x2)
- 17x118x446 (x2)

Unutarnje daske (mm):

- 17x98x215 (x2)
- 17x98x238

Utor za akumulator (mm):

- 17x17x79 (x2)
- 17x17x150



Slika 2.10, Kućište

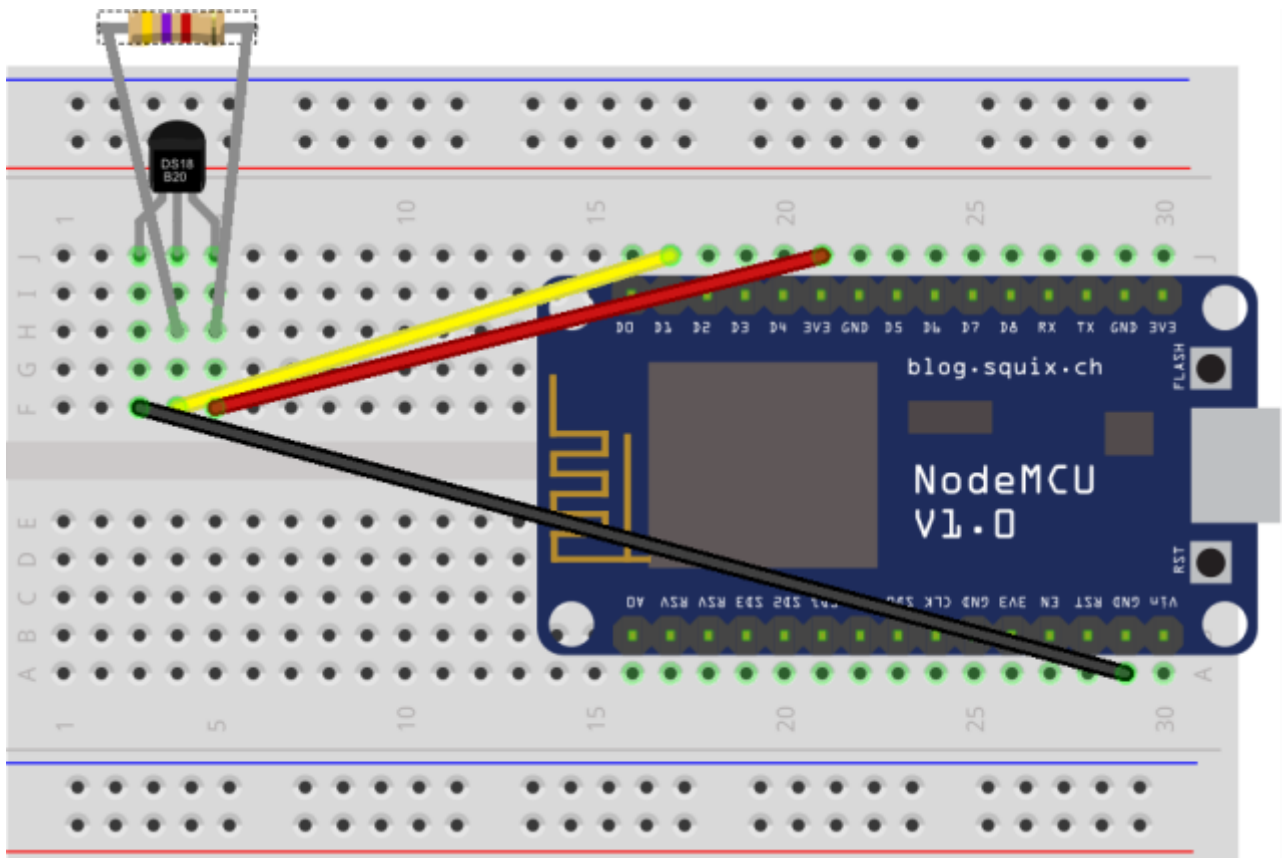


Slika 2.11, Kućište sa svim komponentama

3. Električni dio

3.1. Spajanje temperaturnog senzora DS18B20 na ESP8266 mikroupravljač

Temperaturni senzor DS18B20 sastoji se od 3 žice. Crvena se spaja na napajanje od 5V, crna je uzemljenje, a žuta daje analogni signal mikroupravljaču. Između crvene i žute žice spaja se pull up otpornik od 4700 Ohma radi neometanog prolaska signala. Signalna se žica spaja na pin A0, tj. analogni ulaz mikroupravljača. Shema je izrađena Web editorom „Tinkercad“ [3].



Slika 3.1, Kontakti senzora na pinovima mikroupravljača [3]

3.2. Spajanje ventilatora i grijača preko releja

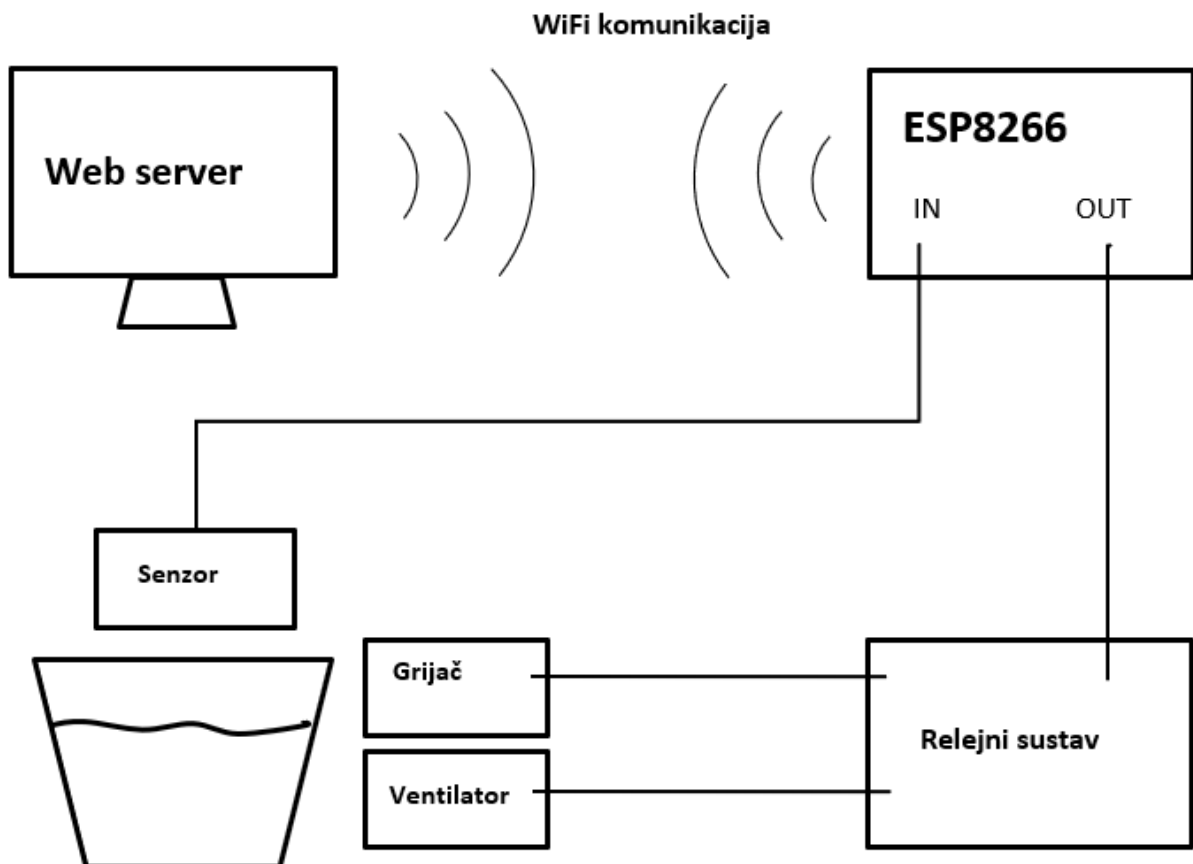
Relej se sastoji od 6 kontakata; napajanja, uzemljenja i signala za upravljački krug te NC, NO i ulaza strujnog kruga kojim se upravlja (COM). Napajanje upravljačkog strujnog kruga spaja se na

5V, a signal na jedan od digitalnih pinova mikroupravljača. COM kontakt se spaja na jednu od žica vanjskog napajanja, a preko NO kontakta je spojen DC motor. Na isti način se spaja i grijač na zasebni relej.



Slika 3.2, Shema spajanja DC motora preko releja [4]

3.3. Blokovski prikaz procesa

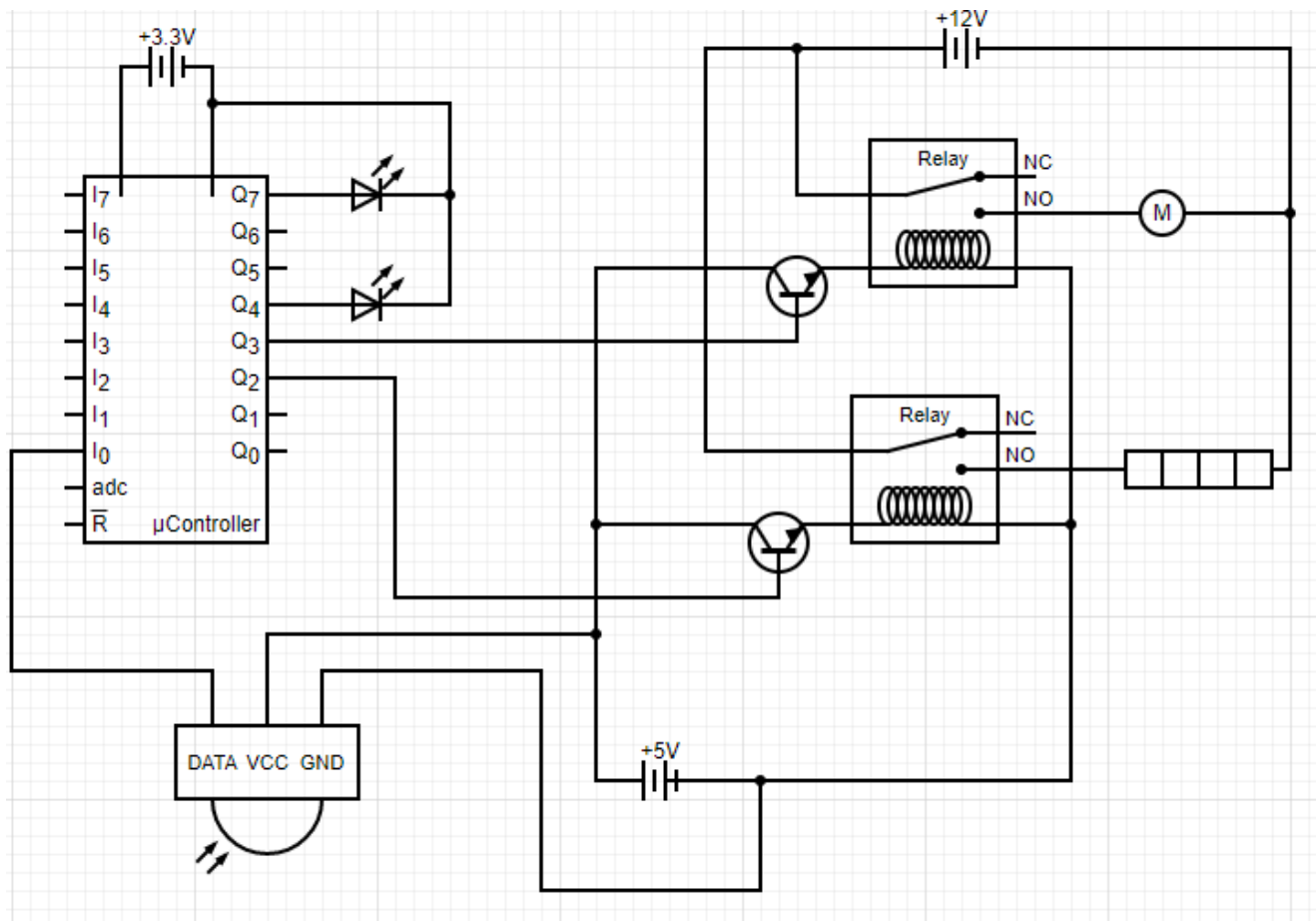


Slika 3.3, Blokovski prikaz procesa

3.4. Shematski prikaz sklopa procesa

Shematski prikaz sklopa procesa sastoji se od relejnog sustava, mikroupravljača, tri izvora napajanja, aktuatora, temperaturnog senzora i signalnih dioda.

Shema je izrađena na web stranici Circuit Diagram. (<https://www.circuit-diagram.org/editor/open>). [5]



Slika 3.4, Shematski spoj sklopa

4. Programski dio

4.1. Arduino IDE program

Arduino IDE (Integrated Development Environment) je softverska platforma koja se koristi za programiranje Arduino mikroupravljača.

Korištenjem Web stranice www.w3schools.com [8] iskopiran je program u originalnoj boji kao što i pronađene HTML naredbe za slova: đ, ć, č i ž.

4.1.1. Definiranje varijabli i uključivanje programskih biblioteka (library)

U ovom dijelu programa definiraju se pinovi mikroupravljača zbog lakšeg razumijevanja, kao i temperatura tekućine, vrijednost histereze, SSID WiFi mreže i lozinka, te pin za senzor. Uključuju se biblioteke senzora i WiFi komunikacije.

```
//int varijable definirane za lakše određivanje pinova na ESP8266
mikroupravljaču
static const uint8_t D0 = 16;
static const uint8_t D1 = 5;
static const uint8_t D2 = 4;
static const uint8_t D3 = 0;
static const uint8_t D4 = 2;
static const uint8_t D5 = 14;
static const uint8_t D6 = 12;
static const uint8_t D7 = 13;
static const uint8_t D8 = 15;
static const uint8_t D9 = 3;
static const uint8_t D10 = 1;

int temp_tekucine=23;
//temperatura tekucine koja se održava u automatskom načinu rada
int histereza=1;
//vrijednost histereze u Celzijevim stupnjevima
int value=0;
//vrijednost varijable prema kojoj se ispisuje trenutno stanje sustava

#define ONE_WIRE_BUS D4 //definira se naziv pina na koji spajamo
temperaturni senzor

const char* ssid = "Optima-F62798";
//char varijabla koja se koristi za upisivanje ssid-a mreže na koju se
želimo spojiti
const char* lozinka = "OPTIMA3229267007123";
//char varijabla koja se koristi za upisivanje šifre mreže na koju se
želimo spojiti
```

```

#include <OneWire.h>
//library kojom se uzimaju izlazne vrijednosti temperaturnog senzora
#include <DS18B20.h>
//library u kojoj se nalaze proračuni izlaznih vrijednosti temperaturnog senzora
#include <ESP8266WiFi.h>
//library za ESP8266 WiFi komunikaciju

OneWire oneWire(ONE_WIRE_BUS);
//definira se ime klase za dobivanje vrijednosti sa temperaturnoga senzora
//i određuje se pin se kojeg dobiva podatke

DS18B20 sensor(&oneWire);
//definira se ime klase za proračun vrijednosti temperaturnoga senzora
// i određuje se pin se kojeg dobiva podatke

WiFiServer server(80);

```

Programska sekvenca 1

4.1.2. Funkcija void setup()

Ovaj dio programa sastoji se od:

- Uspostavljanja ulaznih i izlaznih pinova,
- Postavljanje brzine serijske komunikacije (baud rate)
- Uspostave WiFi komunikacije ESP8266 mikroupravljača i Web servera

```

//u void setup funkciji naredbe se izvršavaju samo jednom na početku
//i nakon resetiranja mikroupravljača

void setup(){
  sensor.begin(); //započinje rad temperaturnog senzora

  //započinje se serijska komunikaciju na brzini 115200 baud rate-a
  Serial.begin(115200);

  //funkcije čiji parametri služe za određivanje stanja pina, D1 i D2 šalju signale,
  //a A0 i D4 primaju
  pinMode(D2, OUTPUT);
  pinMode(D1, OUTPUT);
  pinMode(D5, OUTPUT);
  pinMode(D6, OUTPUT);
  pinMode(D4, INPUT);
  pinMode(A0, INPUT);

```

```

Serial.println();
Serial.println();

Serial.print("Spajanje na ");
// na serijski se monitor ispisuje poruka: "Spajanje na ", a zatim naziv
te mreže

Serial.println(ssid); //na serijski se monitor ispisuje naziv te mreže
WiFi.begin(ssid, lozinka); //započinje se wifi komunikacija
WiFi.mode(WIFI_STA);
while (WiFi.status() != WL_CONNECTED) {
// dok ESP8266 nije spojen s WiFi mrežom na serijski monitor ispisuje
se: "."

delay(500); //prestanak izvršavanja naredbi na 500 milisekundi
Serial.print(".");
}

Serial.println("");
Serial.println("WiFi connected");
//ako se ESP8266 spojio na WiFi mrežu ispisuje se poruka: "WiFi
connected"

// pokrenuti web server
server.begin(); //započinje rad WiFi servera
Serial.println("Server started"); //poruka na serijski monitor: "Server
started"

// Ispisuje se na serijski monitor IP adresa koju kopiramo na web
preglednik
Serial.print("Use this URL to connect: ");
Serial.print("http://");
Serial.print(WiFi.localIP()); //IP adresa
Serial.println("/");
}

```

Programska sekvenca 2

4.1.3. Funkcija void loop()

Glavni dio upravljačke logike koji povezuje Web sučelje, mikroupravljač i temperaturni senzor.

U „Programska sekvenca 3“ očitava se temperatura senzora i uspostavlja WiFi veza kako bih dobili URL Web sučelja.

```
//Funkcija koja se stalno izvršava, a ne vraća nikakve vrijednosti
void loop(){

  sensor.requestTemperatures();
  //naredba kojom se dobivaju izlazne veličine senzora

  float trenutna_temp = sensor.getTempC();

  WiFiClient client = server.available(); // provjerava se spojenost
  klijenta
  if (!client) {
    return;
  }

  // Čekati da klijent pošalje nekakve podatke
  Serial.println("new client");
  while(!client.available()){
    delay(1);
  }

  // Dobivaju se podaci sa web sučelja i ispisuju na serijski monitor
  String request = client.readStringUntil('\r');
  Serial.println(request);
  client.flush();
}
```

Programska sekvenca 3

U „Programska sekvenca 4“ postavlja se varijabla „temp_tekucine“ tj. temperatura koja je upisana na Web sučelju. Također se učitava koja od 4 opcija je pritisnuta na Web sučelju: održavanje, hlađenje, grijanje ili mirovanje.

```
//UNOS ZELJENE TEMPERATURE
  if (request.indexOf("GET /?temp=") >= 0) {
    int pos = request.indexOf("temp="); // Pronađi poziciju
    'temp='
    String tempString = request.substring(pos + 5,
    request.indexOf(' ', pos)); // Izdvoji vrijednost temperature kao
    string
    temp_tekucine = tempString.toInt(); // Konvertiraj string u
    integer
    Serial.println("Nova temperatura: " + String(temp_tekucine) +
    " C");
  }
}
```

```

//Provjeravanje koja je od opcija odabrana na web sučelju
if (request.indexOf("/AUTO=ON") != -1) { //ukoliko je odabran
automatski način rada

//digitalWrite je funkcija čijim se parametrima određuje pin
//i binarno stanje tog pina (5V ili 0V)

    if (trenutna_temp > temp_tekucine + histereza){ // ako je
temperatura veća od zadane

//pinovi na koju su spojeni releji i signalne led žaruljice

    digitalWrite(D2, HIGH); //Hlađenje
    digitalWrite(D1, LOW);
    digitalWrite(D6, HIGH);
    digitalWrite(D5, LOW);
    value = 1; //vrijednost prema kojoj se ispisuje stanje procesa
na WEB sučelju

    }
    if (trenutna_temp < temp_tekucine - histereza){ // ako je
temperatura manja od zadane

    digitalWrite(D1, HIGH); //Grijanje
    digitalWrite(D2, LOW);
    digitalWrite(D5, HIGH);
    digitalWrite(D6, LOW);
    value = 2;

    }
}

//Ukoliko je odabrana opcija hlađenja na web sučelju
if (request.indexOf("/VENTILATOR=ON") != -1){

    digitalWrite(D2, HIGH); //Hlađenje
    digitalWrite(D1, LOW);
    digitalWrite(D6, HIGH);
    digitalWrite(D5, LOW);
    value = 1;
}

//Ukoliko je odabrana opcija grijanja na web sučelju
if (request.indexOf("/GRIJAC=ON") != -1){
    digitalWrite(D1, HIGH); //Grijanje
    digitalWrite(D2, LOW);
    digitalWrite(D5, HIGH);
    digitalWrite(D6, LOW);
    value = 2;
}

```



```

//Ukoliko je odabrana opcija mirovanja na web sučelju
if (request.indexOf("/MIROVANJE=ON") != -1){

    digitalWrite(D1, LOW); //Mirovanje
    digitalWrite(D2, LOW);
    digitalWrite(D5, LOW);
    digitalWrite(D6, LOW);
    value = 0;
}

```

Programska sekvenca 4

U „Programska sekvenca 5“ ispisuje se izgled Web sučelja pomoću HTML-a.

```

// Ispisivanje jednostavne web stranice u web preglednik
client.println("HTTP/1.1 200 OK"); // obaveno na početku
client.println("Content-Type: text/html");// obaveno na početku
client.println(""); // obaveno na početku
client.println("<!DOCTYPE HTML>"); // obaveno na početku
client.println("<html>"); // početak HTML-a za prikaz web stranice

client.print("Trenutno stanje: "); //Na web sučelje se ispisuje:
Trenutno stanje:

//Na temelju varijable "value" ispisuje se trenutno stanje procesa
na WEB sučelje
if (value == 1) {
client.print("HLA&Dstrok;ENJE"); //Ispisuje se "HLADENJE" na WEB
sucelje
}
else if(value==2) {
client.print("GRIJANJE"); //Ispisuje se "GRIJANJE" na WEB sucelje
}
else{
    client.print("MIROVANJE"); //Ispisuje se "MIROVANJE" na WEB
sucelje
}

client.println("<br><br>");

//Sadržaj WEB sučelja
client.print("Trenutna temperatura teku&acute;ine: "); //Poruka
koja se ispisuje na WEB sučelje
    sensor.requestTemperatures(); //naredba kojom se dobivaju izlazne
veličine senzora
    client.println(trenutna_temp); //Ispisuje se dobivena temperatura

```

```

client.println("<br><br>");
client.println("Pritisnite <a href=\"/AUTO=ON\">ovdje</a> za
odr&zcaron;avanje upisane temperature <br>");
// Ispisuje se opcija za uključivanje automatskog načina rada
client.println("Pritisnite <a href=\"/VENTILATOR=ON\">ovdje</a> za
hla&dstrok;enje <br>");
// Ispisuje se opcija za uključivanje hlađenja
client.println("Pritisnite <a href=\"/GRIJAC=ON\">ovdje</a> za grijanje
<br>");
// Ispisuje se opcija za uključivanje grijanja
client.println("Pritisnite <a href=\"/MIROVANJE=ON\">ovdje</a> za
stanje mirovanja <br>");
// Ispisuje se opcija za uključivanje stanja mirovanja

    client.println("<html>");
    client.println("<head><title>Postavi temperaturu</title></head>");
    client.println("<body>");
    client.println("<h1>Unesi &zcaron;eljenu temperaturu</h1>");
    client.println("<form action='/' method='GET'>");
    client.println("<label for='temp'>Temperatura (C): </label>");
    client.println("<input type='number' id='temp' name='temp' min='0'
max='100' required>");
    client.println("<input type='submit' value='Postavi'>");
    client.println("</form>");
    client.println("<p>Trenutno postavljena temperatura: " +
String(temp_tekucine) + " C</p>");
    client.println("</body>");
    client.println("</html>");

    client.println("<meta http-equiv='refresh' content='5'>");

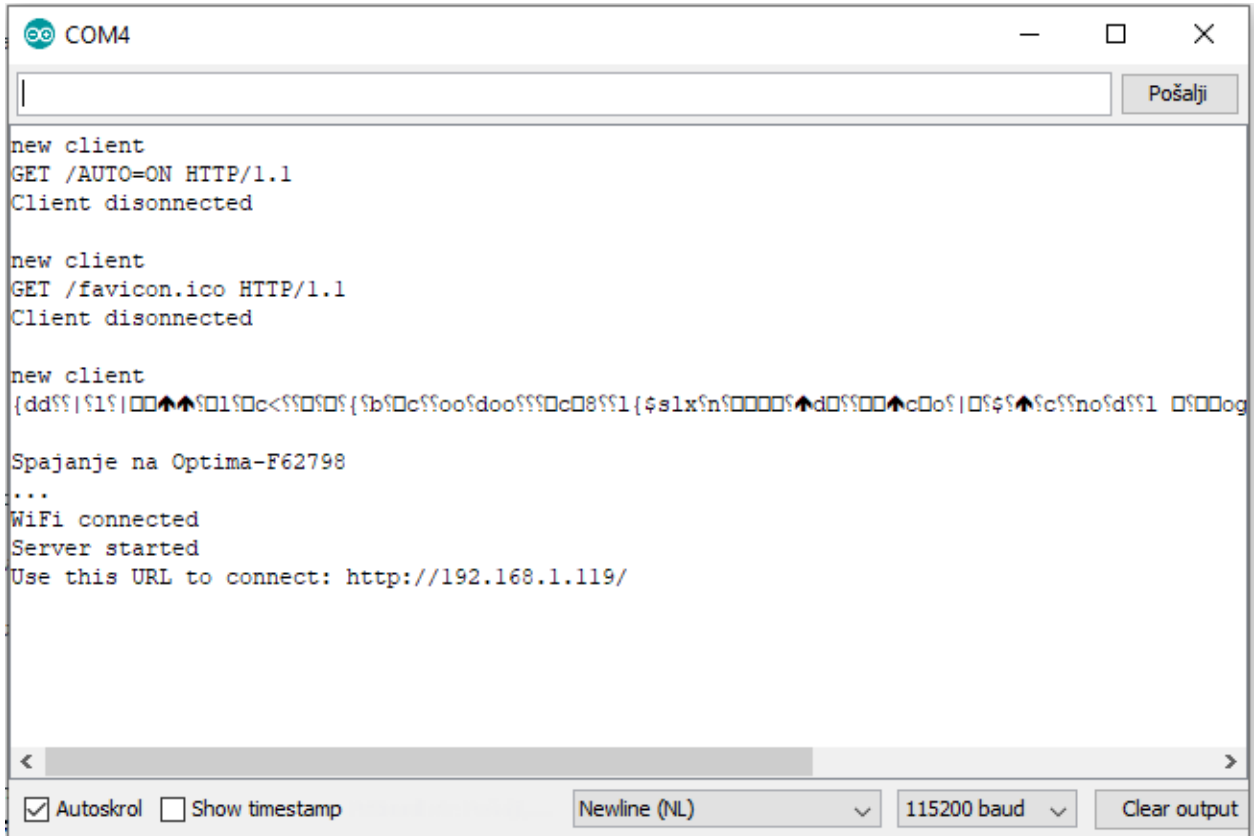
delay(1);
Serial.println("Client disconnected");
Serial.println("");
}

```

Programska sekvenca 5

4.2. Serijski monitor

Ako je WiFi povezanost uspješna na serijski monitor (*Slika 4.1*) se ispisuje URL Web sučelja. To je navedeno u „Programska sekvenca 2“.



```
COM4
new client
GET /AUTO=ON HTTP/1.1
Client disconnected

new client
GET /favicon.ico HTTP/1.1
Client disconnected

new client
[garbled text]

Spajanje na Optima-F62798
...
WiFi connected
Server started
Use this URL to connect: http://192.168.1.119/

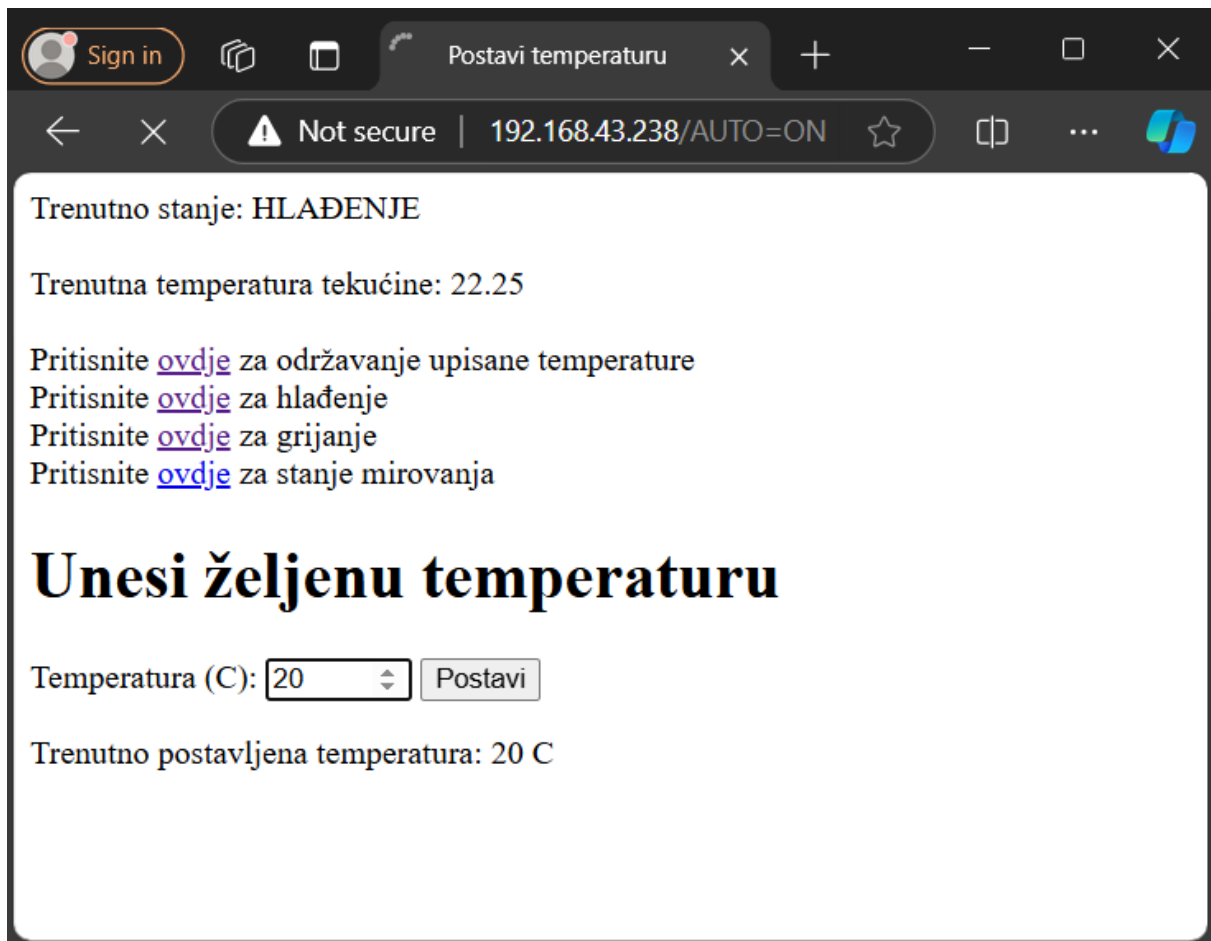
Autoskrol [checked] Show timestamp [unchecked] Newline (NL) [dropdown] 115200 baud [dropdown] Clear output
```

Slika 4.2

4.3. Web sučelje i primjer korištenja sustava

Web sučelje (*Slika 4.3*) je izrađeno prema programu iz „Programska sekvenca 5“. Web sučelje omogućuje korisniku upisivanje zadane temperature održavanja, pregled trenutne temperature kao i upravljanje aktuatorima procesa.

Kod korištenja ukoliko se želi koristiti opcija održavanja zadane temperature prvo se mora upisati u kućicu, a zatim odabrati „Pritisnite [ovdje](#) za održavanje upisane temperature“.



Slika 4.4, Web sučelje

Na Web sučelju trenutno je postavljena temperatura 20°C, a temperaturni senzor DS18B20 očitava 22.5 °C. Pošto je odabrano održavanje temperature (Pritisnite [ovdje](#) za održavanje upisane temperature) sa Web sučelja šalje se signal u obliku teksta „/AUTO=ON“ prema kojem se putem mikroracunala uključuju plava signalna LED žaruljica i ventilator za spuštanje temperature tekućine u posudi (Slika 4.5). Taj slijed programa opisan je u „Programska sekvenca 4“, gdje se vide i ostali mogući signali: „/MIROVANJE=ON“, „/GRIJAC=ON“ i „/VENTILATOR=ON“.



Slika 4.6, Proces u hlađenju

5. Zaključak

Ovaj rad prikazuje mogućnost primjene tehnologija kao što su mikroupravljači, IoT sustavi i Web sučelja u procesima upravljanja. Korištenjem ESP8266 mikroupravljača kao središnjeg dijela sustava, omogućeno je jednostavno povezivanje fizičkih komponenti s digitalnim platformama.

Pokazalo se kao praktično rješenje za izazove u industriji, automatizaciji i drugim sektorima. Daljinsko upravljanje omogućuje operaterima i korisnicima da prate procese u stvarnom vremenu te donose brze odluke, bez potrebe za fizičkom prisutnošću.

U konačnici, ovakvi sustavi pridonose smanjenju troškova i povećanju operativne efikasnosti, dok istovremeno olakšavaju rad korisnicima.

Najviše vremena izrade rada sam posvetio povezivanju Web stranice i mikroupravljača te izradi Web stranice. Malih problema sam i imao kod povezivanja komponenti zbog različitih napona na kojima komponente funkcioniraju.

Zadovoljan sam svojim radom, a mogao bih još poboljšati izgled stranice i/ili napraviti mobilnu aplikaciju na istim principu.

6. Literatura

- [1] <https://work-management.org/lp/top-5-marketing-management-software/>, dostupno 22.09.2024
- [2] <https://www.thetalearningpoint.com/2023/03/construction-of-dc-machines.html>, dostupno 22.09.2024
- [3] www.tinkercad.com
- [4] <https://forum.arduino.cc/t/do-a-12v-dc-motor-achieve-its-full-efficiency-when-connected-to-a-5v-relay/666658>, dostupno 22.09.2024
- [5] <https://www.circuit-diagram.org/editor/open>
- [6] <https://randomnerdtutorials.com/>
- [7] <https://howtomechatronics.com/>
- [8] www.w3schools.com/
- [9] <https://youtu.be/pL3dhGtmcMY?si=0xvfozejXr38-ehV>, dostupno 22.09.2024

Popis slika

Slika 1.1, IoT	2
Slika 2.1, Izgled i pinovi ESP8266.....	4
Slika 2.2, Releji za grijač	5
Slika 2.3, Ispitivanje potrebne struje multimetrom, NPN tranzistor i Releji.....	6
Slika 2.4, Temperaturni senzor i njegove žice.....	7
Slika 2.5, Izgled grijača	8
Slika 2.6, Ventilator i shema DC motora [2]	9
Slika 2.7, Akumulator i naponski pinovi mikroupravljača.....	10
Slika 2.8, LED žaruljice	11
Slika 2.9, Otpornik, Spojne žice, Eksperimentalna pločica.....	12
Slika 2.10, Kućište.....	13
Slika 2.11, Kućište sa svim komponentama	13
Slika 3.1, Kontakti senzora na pinovima mikroupravljača [3]	14
Slika 3.2, Shema spajanja DC motora preko releja [4]	15
Slika 3.3, Blokovski prikaz procesa	15
Slika 3.4, Shematski spoj sklopa	16
Slika 4.1	24
Slika 4.2, Web sučelje	25
Slika 4.3, Proces u hlađenju.....	26

Popis programskih sekvenci

Programska sekvenca 1	18
Programska sekvenca 2	19
Programska sekvenca 3	20
Programska sekvenca 4	22
Programska sekvenca 5	23

Prijava završnog rada

Definiranje teme završnog rada i povjerenstva

ODJEL Odjel za Mehatroniku

STUDIJ Stručni prijediplomski studij Mehatronika

PRISTUPNIK Fabijan Škrobar

MATIČNI BROJ 0336053418

DATUM 10.9.2024.

KOLEGIJ Automatsko upravljanje

NASLOV RADA Upravljanje toplinskim procesom putem Web sučelja

NASLOV RADA NA ENGL. JEZIKU Thermal process control through the Web interface

MENTOR Miroslav Horvatić, dipl. ing.

ZVANJE viši predavač

ČLANOVI POVJERENSTVA

1. mr. sc. Ivan Šumiga, viši predavač - predsjednik povjerenstva
2. izv. prof. dr. sc. Emil Dumić - član
3. Miroslav Horvatić, dipl. ing., viši predavač - član / mentor
4. Josip Srpak, dipl. ing., viši predavač - zamjenski član
- 5.

Zadatak završnog rada

BROJ 040/MEH/2024

OPIS

Izraditi automatizirani sustav za upravljanje toplinskim procesom putem Web sučelja. Za upravljanje temperaturom procesa koristiti mikroupravljač koji ima mogućnost bežične WiFi komunikacije. Odabrati odgovarajući senzor temperature koji će mjeriti temperaturu procesa i tu informaciju prosljediti mikroupravljaču. Za grijanje i hlađenje procesa odabrati odgovarajući grijač, ventilator i sustav za dovođenje energije na te elemente. Stvarna temperatura toplinskog procesa mora se moći očitati putem web sučelja, putem kojeg se mora moći zadati i željena temperatura toplinskog procesa. Dokumentirati projektiranje, realizaciju, ispitivanje i korištenje automatiziranog sustava za upravljanje toplinskim procesom.

ZADATAK URUČEN

11.09.2024.

POTPIS MENTORA

Jen



Sveučilište
Sjever

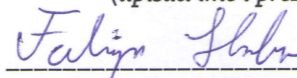
IZJAVA O AUTORSTVU

Završni/diplomski/specijalistički rad isključivo je autorsko djelo studenta koji je isti izradio te student odgovara za istinitost, izvornost i ispravnost teksta rada. U radu se ne smiju koristiti dijelovi tuđih radova (knjiga, članaka, doktorskih disertacija, magistarskih radova, izvora s interneta, i drugih izvora) bez navođenja izvora i autora navedenih radova. Svi dijelovi tuđih radova moraju biti pravilno navedeni i citirani. Dijelovi tuđih radova koji nisu pravilno citirani, smatraju se plagijatom, odnosno nezakonitim prisvajanjem tuđeg znanstvenog ili stručnoga rada. Sukladno navedenom studenti su dužni potpisati izjavu o autorstvu rada.

Ja, Fabijan Škrobar (ime i prezime) pod punom moralnom, materijalnom i kaznenom odgovornošću, izjavljujem da sam isključivi autor/ica završnog rada pod naslovom Upravljanje toplinskim procesom putem Web sučelja (upisati naslov) te da u navedenom radu nisu na nedozvoljeni način (bez pravilnog citiranja) korišteni dijelovi tuđih radova.

Student/ica:

(upisati ime i prezime)



(vlastoručni potpis)

Sukladno članku 58., 59. i 61. Zakona o visokom obrazovanju i znanstvenoj djelatnosti završne/diplomske/specijalističke radove sveučilišta su dužna objaviti u roku od 30 dana od dana obrane na nacionalnom repozitoriju odnosno repozitoriju visokog učilišta.

Sukladno članku 111. Zakona o autorskom pravu i srodnim pravima student se ne može protiviti da se njegov završni rad stvoren na bilo kojem studiju na visokom učilištu učini dostupnim javnosti na odgovarajućoj javnoj mrežnoj bazi sveučilišne knjižnice, knjižnice sastavnice sveučilišta, knjižnice veleučilišta ili visoke škole i/ili na javnoj mrežnoj bazi završnih radova Nacionalne i sveučilišne knjižnice, sukladno zakonu kojim se uređuje umjetnička djelatnost i visoko obrazovanje.