

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

**Sustav za automatsko upravljanje
mikroklimatskim uvjetima zatvorenog
prostora za uzgoj biljaka**

Niko Šikić

Zagreb, srpanj 2024.

Student: Niko Šikić

Naslov teme:

Sustav za automatsko upravljanje mikroklimatskim uvjetima zatvorenog prostora za uzgoj biljaka

Naslov teme (engleski):

A system for automatic control of the microclimatic conditions in rooms for growing plants

Studij:

Elektrotehnika i informacijska tehnologija

Mentor:

Prof. dr. sc. Hrvoje Džapo

Kratki opis:

Upoznati se s ključnim parametrima okolišnih uvjeta u zatvorenim prostorima za uzgoj biljaka (npr. plostenici, urbani vrtovi). Odabrat odgovarajuće senzore za praćenje okolišnih uvjeta (temperatura, vlaga, vlažnost tla, osvjetljenje, koncentracija plinova itd.). Upoznati se s aktuatorima za kontrolu parametara okolišnih uvjeta u zatvorenim prostorima za uzgoj biljaka (grijači, ventilatori, lampe i sl.). Upoznati se s arhitekturom mikrokontrolera ESP32 i programskim razvojnim okvirom ESP-IDF. Izraditi pokazni sustav temeljen na mikrokontroleru ESP32 te odabranim senzorima i aktuatorima koji će omogućiti nadzor i kontrolu parametara mikroklima u zatvorenom prostoru za uzgoj biljaka. Omogućiti povezivost sustava putem WiFi mreže, vezu prema udaljenom serveru, prikupljanje i pohranu podataka na server, prikaz trenutnog stanja mikroklima i kontrolu okolišnih parametara. Razviti korisničko sučelje za upravljanje sustavom i vizualizaciju podataka. Ispitati funkcionalnost rješenja na pokaznom sustavu.

*Zahvaljujem profesoru Džapi i asistentici Salamun na pomoći pri izradi ovog rada te
Poljoprivrednom institutu u Osijeku na obilasku komora rasta*

Sadržaj

1.	Uvod	1
2.	Praćenje i regulacija uvjeta u stakleniku	2
2.1.	Moderni staklenički uzgoj	2
2.2.	Mikroklimatski parametri	4
2.2.1.	Temperatura zraka	4
2.2.2.	Vlažnost zraka	5
2.2.3.	Koncentracija CO ₂ u zraku	5
2.2.4.	Vlažnost tla	5
2.2.5.	Intenzitet svjetlosti.....	6
2.3.	Ciljevi rada i opis rješenja	6
3.	Ugradbeno računalo temeljeno na ESP32 mikrokontroleru	8
3.1.	Razvojni programski okvir ESP-IDF	8
3.2.	Razvojni sustav ESP32-DevkitC V4	9
3.3.	Načini rada smanjene potrošnje.....	10
4.	Senzori za praćenje okolišnih uvjeta	12
4.1.	Senzor temperature i vlažnosti zraka DHT11.....	12
4.2.	Senzori spojeni na A/D pretvornik	14
4.2.1.	Analogno-digitalna pretvorba na ESP mikrokontroleru.....	14
4.2.2.	Senzor vlažnosti tla.....	16
4.2.3.	Senzor plinova MQ-135	19
4.3.	Senzori spojeni putem I2C sučelja	22
4.3.1.	I2C komunikacijski protokol	22
4.3.2.	I2C komunikacija na ESP32 mikrokontroleru.....	23
4.3.3.	Senzor ambijentnog osvjetljenja i udaljenosti LTR-507	24
5.	Aktuatori za automatsku regulaciju uvjeta	26

5.1.	Algoritmi automatske regulacije	26
5.2.	Izvedba rješenja u sustavu za regulaciju uvjeta u stakleniku	27
6.	Platforma ESP RainMaker	29
6.1.	Povezivanje.....	29
6.2.	Programski kod.....	30
7.	Pokazni model sustava za praćenje i regulaciju uvjeta u stakleniku	34
7.1.	Sklo povski dio sustava	34
7.2.	Programska potpora.....	37
7.3.	Rezultati.....	38
8.	Zaključak	41
	Literatura	42
	Sažetak.....	45
	Summary.....	46

1. Uvod

Suvremenih uzgoj biljaka u zatvorenim prostorima podrazumijeva osiguravanje optimalnih mikroklimatskih uvjeta za njihov rast. U tu svrhu nastaju moderni staklenici s razvijenim sustavima za praćenje tih parametara. Ti sustavi nerijetko implementiraju izrazito učinkovite mehanizme automatske regulacije i time smanjuju ljudsku uključenost i poljoprivredne napore. To je izrazito važno za moderno društvo, koje podrazumijeva brzu i učinkovitu proizvodnju za prehrambene i istraživačke svrhe.

Danas dizajn takvih sustava postaje sve jednostavniji i pristupačniji. Staklenički se uzgoj sve češće uključuje u mrežu međusobno povezanih uređaja s ciljem pohrane i dijeljenja (senzorskih) podataka obuhvaćenu pojmom interneta stvari (engl. *Internet of things – IoT*). Zbog toga senzori, aktuatori i digitalni sustavi kojima se povezuju postaju dostupniji i jednostavniji za korištenje.

Glavni je cilj ovog rada ilustrirati primjenjivost rješenja interneta stvari u stakleničkom uzgoju. Ta će se primjenjivost demonstrirati izradom pokaznog modela staklenika, u kojem će se pratiti i regulirati mikroklimatski parametri te pohranjivati mjerni rezultati, koji će biti dostupni korisniku putem bežične mreže. Model će se temeljiti na mikrokontroleru ESP32 i platformi ESP RainMaker.

U drugom poglavlju predstaviti će se moderni staklenički sustavi i objasniti važnost njihova razvoja. Također će se navesti najvažniji mikroklimatski parametri te senzori i aktuatori koji se najčešće koriste za njihovu regulaciju. Na kraju poglavlja predložit će se rješenje za izradu pokaznog modela za regulaciju tih parametara, čiji će dijelovi biti detaljnije opisani u poglavljima koja slijede. U trećem poglavlju opisan je mikrokontroler ESP32 i njegova bitna obilježja. Glavna obilježja korištenih senzora i pripadajući komunikacijski protokoli objašnjeni su u četvrtom poglavlju. Peto poglavlje posvećeno je odabranim aktuatorima i metodama njihova automatskog upravljanja. U šestom poglavlju predstavljena je platforma ESP RainMaker i programska potpora u jeziku C potrebna za rad pokaznog modela. Pokazni model sklopovski je i programski opisan u sedmom poglavlju.

2. Praćenje i regulacija uvjeta u stakleniku

2.1. Moderni staklenički uzgoj

Moderni su staklenici izrazito učinkoviti sustavi za uzgoj biljnih vrsta radi dobivanja poljoprivrednih proizvoda ili u istraživačke svrhe. Staklenički je uzgoj danas raširen u području botanike, vrtlarstva i prilagođen za komercijalnu proizvodnju. Načelo takvog uzgoja temelji se na održavanju stalnih mikroklimatskih uvjeta unutar staklenika neovisno o vanjskim klimatskim čimbenicima. Transparentni materijal staklenika (najčešće staklo ili plastika) omogućava prođor i zadržavanje vidljivog sunčevog zračenja, ali i cirkulaciju toplinske energije koju biljke zrače u infracrvenom spektru [1]. Na taj način moguće je uzgoj jestivih, ukrasnih, ljekovitih i dr. biljaka u bilo kojem klimatskom području. Staklenici se od drugih metoda uzgoja (pokrov poljoprivrednih redova, niski i visoki tuneli, mreže za zasjenjivanje i dr.) ističu po naprednijoj tehnološkoj izvedbi [2] i ubrzanim razvoju. Većina tehnoloških napora usmjerena je upravo prema praćenju mikroklimatskih uvjeta u stakleniku. Kontinuiranim mjeranjem okolišnih parametara dobiva se jasna slika o njihovu utjecaju na biljke, što služi kao smjernica pri optimalnom uzgoju. Poljoprivrednik ili botaničar prilagođava uvjete u stakleniku sukladno mjernim rezultatima. Takvo se reguliranje smatra ručnim (manualnim) i izrazito je izazovno i vremenski zahtjevno. Zato se razvijaju sustavi s autonomnom regulacijom, koji automatski pokreću aktuatori ovisno o senzorskim mjerenjima [3]. Tako se smanjuje ljudska uključenost i povećava produktivnost.

Daljnji napredak u uzgoju biljaka usmjeren je prema povezivanju hortikulturnih pametnih sustava u svrhu analize i vizualizacije mikroklimatskih parametara [1]. Dizajn takvih sustava postaje sve jednostavniji uspostavom bežičnih mreža međusobno povezanih uređaja koji điđe podatke i komuniciraju preko oblaka (engl. *cloud*). Takvi su sustavi obuhvaćeni pojmom interneta stvari (engl. *Internet of things - IoT*), koji predstavlja jednostavno i povoljno rješenje za obradu senzorskih podataka u stvarnom vremenu. Takva rješenja postaju sve dostupnija i cjenovno prihvatljivija; već se s komercijalnim senzorima i aktuatorima upravljanim jeftinim mikrokontrolerima mogu postići zadovoljavajući rezultati praćenja i regulacije mikroklimatskih uvjeta u stakleniku, što će se nastojati prikazati ovim radom.

Pristupačnost novih tehnologija i njihova upotreba u hortikulti izrazito su važni za rješavanje globalnih problema poput gladi i siromaštva u sklopu održivog razvoja. Moderni

hidroponski i aeroponski sustavi, poput onog prikazanog slikom (Slika 2.1), omogućuju visoku iskoristivost vode i zemljišta pri uzgoju hrane [4], a opremljeni su ugradbenim sustavima kojim se osiguravaju najbolji uvjeti za rast biljnih vrsta. Nadalje, posebne komore za uzgoj biljaka u kontroliranim uvjetima poput modularnih uređaja Fitotron prikazanih na slici (Slika 2.2) koriste se u farmaceutskoj industriji i u istraživačke svrhe.



Slika 2.1 Hidroponski vrt njujorškog ICE-ja [5].



Slika 2.2 Komora rasta Fitotron tvrtke Weiss Technik [6].

2.2. Mikroklimatski parametri

Za izradu pokaznog modela razmatrani su sljedeći mikroklimatski parametri:

- temperatura zraka,
- vlažnost zraka,
- koncentracija ugljikova dioksida u zraku,
- vlažnost tla,
- intenzitet svjetlosti.

Oni su odabrani zbog njihove važnosti za uzgoj biljaka i dostupnosti senzora za praćenje mjernih rezultata. U nastavku slijedi kratak opis pojedinih parametara te najčešće upotrebljavani senzori i aktuatori za njihovo upravljanje.

2.2.1. Temperatura zraka

Osnovno načelo stakleničkog uzgoja temelji se na zadržavanju toplinske energije unutar zatvorenog prostora, a temperatura predstavlja fizikalnu veličinu kojom se ta toplina kvantificira. Temperatura unutar staklenika ne mora biti i najčešće nije jednoliko raspoređena. Ovisno o veličini, pojedini dijelovi staklenika mogu biti i do par stupnjeva topliji od drugih, pa je senzore potrebno postaviti na više mjesta [7].

Najčešće korišteni senzori jesu termoparovi, termistori i nešto skuplji otpornički temperaturni detektori (engl. *Resistance Temperature Detector – RTD*). Za većinu jednostavnih primjena, pa tako i za potrebe ovog rada, koriste se jeftini i manje pouzdani integrirani sklopovi za mjerjenje temperature spremni za izravno spajanje na mikrokontroler [8].

Aktuatori za regulaciju temperature najčešće su realizirani toplinskim pumpama, električnim, plinskim i posebno održivim solarnim grijačima zraka. Koriste se i ventilacijski uređaji, kojima se toplina izmjenjuje s okolinom strujanjem zraka.

2.2.2. Vlažnost zraka

Vlažnost zraka predstavlja količinu vodene pare u zraku nastale u procesima isparavanja u tlu, transpiracijom biljaka, izmjenom vlage s okolinom zbog ventilacije i dr. Najčešće se izražava relativno u postotku i fizikalno je vezana uz temperaturu zraka. Zato se na tržištu često pojavljuju senzorski moduli kojima se mijere oba parametra [7].

Senzori se dijele u tri velike skupine: kapacitivni (engl. *Capacitive Humidity Sensor – CHS*), otpornički (engl. *Resistive Humidity Sensor – RHS*) i termički (engl. *Thermal Conductivity Humidity Sensor – TCHS*). Za zahtjeve brzog odziva i visoke preciznosti koriste se kapacitivni, a kod zahtjevnijih klimatskih uvjeta (visoka temperatura, otrovne pare) primjenjiviji su otpornički i termički senzori [8].

Kao i kod temperaturne regulacije, ventilatori su česti aktuatori. Osim njih, koriste se odvlaživači i posebni ovlaživači, koji osim povećavanja vlažnosti snižavaju i temperaturu zraka.

2.2.3. Koncentracija CO₂ u zraku

Ugljikov je dioksid uz atmosferski kisik najvažniji plin za život biljaka. Biljke ga koriste za dobivanje energije u fotokemijskom procesu fotosinteze, pa je praćenje njegove koncentracije (najčešće relativno kao ppm, dio na 10^6 dijelova) izrazito važno u hortikulturi. U zatvorenom prostoru staklenika drastično se povećava koncentracija CO₂ (400 – 1000 ppm) u odnosu na atmosferu (oko 400 ppm) [7], a regulacija se provodi ventilacijom.

Senzori su podijeljeni u tri skupine: elektrokemijski, optički (infracrveni, NDIR) i metal-oksidni [8], kojima će se dati detaljniji opis u narednim poglavljima.

2.2.4. Vlažnost tla

Dostatna količina vode i mineralnih tvari u tlu ključna je za rast i razvoj biljaka, no prevelika količina može voditi do truljenja i smrti. Apsolutnu je vlažnost tla izrazito teško odrediti jer ovisi o brojnim faktorima poput gustoće i dielektričnosti tla, ali i svojstvima dodane vode i njezina utjecaja na tlo. Zato se vlažnost pojednostavljeno iskazuje gravimetrijski (maseni udio vode u tlu) ili volumetrijski (volumni udio vode u tlu). Kako se volumetrijsko mjerenje češće koristi, ono je odabранo za potrebe rada.

Praćenje se najčešće provodi mjerjenjem napetosti vode tenziometrom i granularnim matričnim senzorom (engl. *Granular Matrix Sensor – GMS*) ili izravnog sastava vode u tlu reflektometrom vremenske domene (engl. *Time Domain Reflectometry – TDR*) i kapacitivnim senzorima [8]. Jeftini senzori spremni za spajanje na mikrokontroler najčešće su otpornički i kapacitivni, pa će se oni detaljnije razmatrati u narednim poglavljima.

Sustavi navodnjavanja u staklenicima postaju sve napredniji, a posebno se ističe automatska irigacijska regulacija [7], koja u kombinaciji s preciznim senzorima održava stalnu vlažnost tla štedeći vodu.

2.2.5. Intenzitet svjetlosti

Sunčev zračenje biljkama omogućuje svjetlosnu energiju potrebnu za fotosintezu te toplinsku energiju za održavanje životnih uvjeta. Količina potrebne svjetlosti ovisi o biljnoj vrsti, a karakteristike svjetlosti o izvoru i strukturnim značajkama staklenika. Intenzitet vidljive svjetlosti obično se izražava u lumenima po četvornom metru (mjerna jedinica lux), a u hortikulturi češća je jedinica PPFD (gustoća fotosintetski aktivnog zračenja u $\mu\text{mol}/\text{m}^2/\text{s}$), koja u obzir uzima svjetlosni spektar relevantan za biljni uzgoj (valne duljine 400 – 700 nm) [7].

Senzori za mjerjenje vidljive svjetlosti najčešće su fotometrijski i fotootpornički, a u nešto skupljoj kategoriji ističu se piranometri za mjerjenje ukupnog Sunčevog zračenja te kvantni svjetlosni senzori, kojima se najčešće mjeri PPFD [8].

Najčešći je aktuator LED rasvjeta za rast biljaka zbog niske potrošnje i robusnosti. Razvojem te tehnologije omogućeno je zračenje svjetlosti određene valne duljine prilagođene uzgajanoj biljnoj vrsti [7]. Njihov se rad automatski regulira ovisno o dnevnom ciklusu i životnom stadiju biljke.

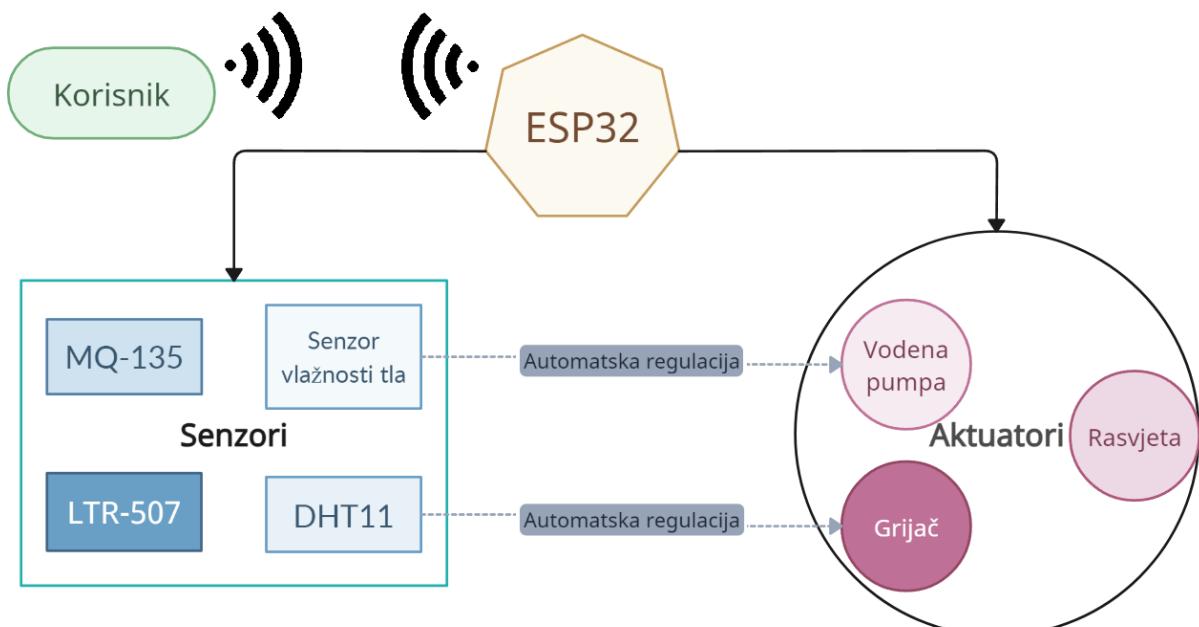
2.3. Ciljevi rada i opis rješenja

Glavni je cilj rada ilustrirati primjenu IoT rješenja u stakleničkom uzgoju izradom pametnog sustava za praćenje i regulaciju mikroklimatskih uvjeta unutar pokaznog modela. Nakon evaluiranja senzora i aktuatora iz prethodnog poglavlja, uzimajući u obzir njihovu cijenu i dostupnost, odabrani su oni prikazani su tablici (**Tablica 2.1**).

Tablica 2.1 Prikaz odabralih senzora i aktuatora

Parametar	Senzor	Aktuator
Temperatura zraka	DHT11	Električni grijač
Vlažnost zraka		-
Koncentracija CO ₂ u zraku	MQ-135	-
Vlažnost tla	Kapacitivni senzor	Vodena pumpa
Intenzitet svjetlosti	LTR-507ALS-01	LED rasvjeta

Pokazni će model biti staklenik manjih dimenzija namijenjen za unutarnji ili vanjski prostor. Temeljit će se na mikrokontroleru ESP32 tvrtke Espressif, koji će obrađivati mjerne podatke senzorskog čvora i pogoniti aktuatorske elemente. Sustav će biti povezan na Wi-Fi mrežu, preko koje će se slati i pohranjivati mjerne podatke. Korisnik će moći tim podatcima pristupiti mobilnim uređajem spojenim na bilo koju mrežu te podešiti parametre regulacije i isertati rezultate. Blokovska shema sustava prikazana je slikom (Slika 2.3), a u narednim poglavljima pojašnjena je uloga svakog dijela sheme.



Slika 2.3 Blokovska shema pokaznog modela.

3. Ugradbeno računalo temeljeno na ESP32 mikrokontroleru

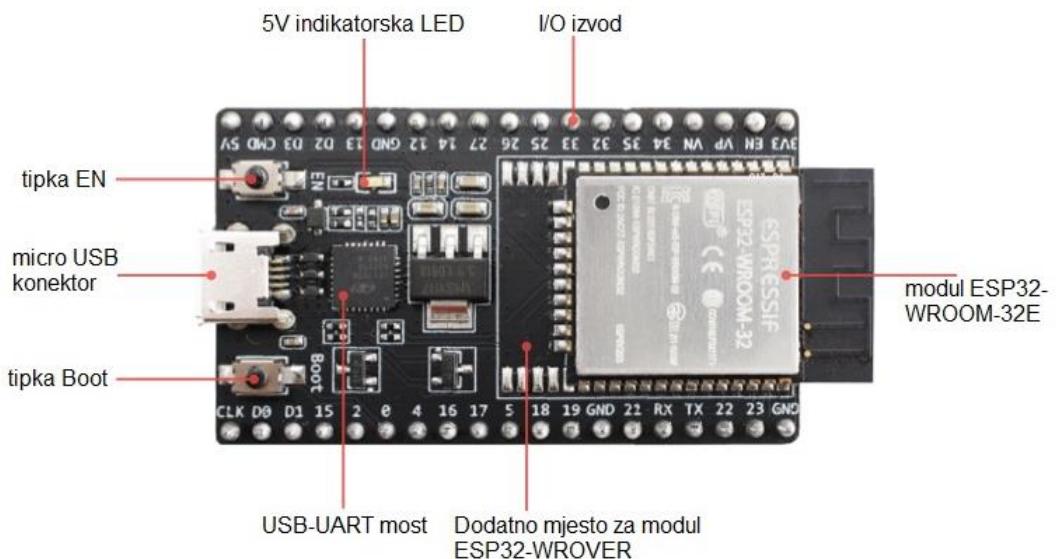
ESP32 jeftin je i moćan mikrokontroler koji proizvodi tvrtka *Espressif Systems Company* (Šangaj, Kina). Glavne su mu karakteristike brzi jednojezgreni ili dvojezgreni procesor, integrirane Wi-Fi i Bluetooth funkcionalnosti, velik broj izvoda GPIO opće namjene te rad prilagođen niskoj potrošnji. Mikrokontroler koristi snažni 32-bitni mikroprocesor Xtensa LX6, koji omogućava višezadaćnost i izvršavanje složenih zadataka. Također podržava brojne bežične protokole poput BLE (*Bluetooth Low Energy*), 802.11 b/g/n/e/I, WLAN MAC i Wi-Fi Direct. Protokolom *Peer-to-Peer* (ESP-NOW) omogućuje povezivanje više mikrokontrolera i neposrednu bežičnu razmjenu podataka. GPIO izvodi koriste se za pristup velikom broju vanjskih uređaja i komunikacijskih protokola. Među njima se ističu pretvornici ADC i DAC te sučelja I2C, SPI, UART i PWM [9]. Espressif također nudi niz razvojnih okvira kojim korisnicima olakšava razvoj IoT rješenja među kojima su: ESP RainMaker (upravljanje uređajima preko oblaka), ESP-WHO (okruženje za raspoznavanje lica), ESP-ADF (za zvučne aplikacije), ESP-DASH (kreiranje nadzornih ploča) te protokoli ESP-MESH i ESP-TLS.

3.1. Razvojni programski okvir ESP-IDF

Uobičajeni način programiranja mikrokontrolera jest pomoću razvojnog okvira koji omogućava Espressif pod nazivom ESP-IDF (engl. *Espressif Systems Internet of Things Development Framework*). Razvojni okvir sadrži potrebne biblioteke i izvorni kod za mikrokontroler, kao i potrebne skripte za alatni lanac (engl. *toolchain*) za prevođenje koda [10]. Također omogućava korisniku grafički izbornik za konfiguraciju naredbom `idf.py menuconfig` u terminalu. Opcijama u izborniku mogu se promijeniti postavke povezivanja (Wi-Fi, Bluetooth i dr.), memorije (NVS, particije memorije *flash* i dr.), periferija (ADC, I2C, SPI i dr.), načina rada niske potrošnje i slično. Najčešći programski jezik za programiranje ESP32 jest C, pa je u njemu pisana većina biblioteka. Mikrokontroler se jednako lako može programirati u jeziku C++ korištenjem Arduino biblioteka, no u radu je razmatrano programiranje u C-u [11].

3.2. Razvojni sustav ESP32-DevkitC V4

Za izvedbu rada odabrana je razvojna pločica ESP32-DevKitC V4 zbog njezine dostupnosti, male veličine i niske cijene. Na njoj se nalazi modul ESP32-WROOM-32E. Na pločici se nalaze dvije tipke: za ponovno postavljanje (engl. *reset*) i prenošenje programskog koda na mikrokontroler (engl. *boot*), ulaz tipa micro USB i USB-UART konverter za komunikaciju mikrokontrolera s računalom te ukupno 38 izvoda, koji uključuju izvod s 5 V i reguliranih 3.3 V. Pločica se može baterijski napajati preko ta dva izvoda ili preko USB konektora. Dijelovi razvojne pločice prikazane su slikom (Slika 3.1), a najbitnija obilježja modula ESP32-WROOM-32E tablicom (Tablica 3.1).



Slika 3.1 Glavni dijelovi ESP32-DevkitC V4 [12].

Tablica 3.1 Glavna obilježja modula ESP32-WROOM-32E [13].

Obilježje	Izvedba
Procesor i memorija	<ul style="list-style-type: none"> • Xtensa dvojezgreni 32-bitni procesor (do 240 Mhz) • 448 kB ROM • 520 kB SRAM • 16 kB SRAM RTC-a
Wi-Fi	<ul style="list-style-type: none"> • 802.11b/g/n • brzina (802.11n) do 150 Mbps • A-MPDU i A-MSDU agregacija • zaštitni interval 0.4 μs • središnja frekvencija 2412 - 2484 MHz
Bluetooth	<ul style="list-style-type: none"> • bluetooth V4.2 BR/EDR i BLE • Odašiljač klase 1/2/3 • AFH • CVSD i SBC
Periferije	<ul style="list-style-type: none"> • SD card, UART, SPI, SDIO, I2C, LED, (Motor) PWM, I2S, IR, brojač pulsa, GPIO, kapacitivni senzor dodira, ADC, DAC, TWAI®
Integrirane komponente	<ul style="list-style-type: none"> • kristalni oscilator 40 MHz • 4/16/8 MB SPI flash memorija
Antena	<ul style="list-style-type: none"> • na pločici
Uvjeti rada	<ul style="list-style-type: none"> • napajanje 3.0 – 3.6 V • radna temperatura -40 – 85 °C

3.3. Načini rada smanjene potrošnje

Iako će pokazni model biti napajan iz gradske mreže, zadatak svakog ugradbenog računalnog sustava jest smanjenje potrošnje. U tu svrhu ESP32 podržava dva načina rada niske potrošnje: *light-sleep* i *deep-sleep*. Ti načini rada niske potrošnje (engl. *sleep*) usporavaju ili u potpunosti zaustavljaju rad procesora, memorije i periferija. Uz to, mogu se podesiti uvjeti prelaska u pojedini način rada te hardverski i softverski pozivi koji vraćaju mikrokontroler u stanje uobičajene potrošnje (engl. *wake-up calls*).

U načinu rada *light-sleep* brzina je rada periferija, radne memorije i procesora ograničena te im je smanjena razina napajanja. S druge strane, u načinu rada *deep-sleep* iste su komponente potpuno isključene. Samo određene periferije ostaju uključene: RTC kontroler sa sporom i brzom memorijom te koprocessor iznimno niske potrošnje (engl. *ultra low power – ULP*). Kako oba načina isključuju bežične periferije, nisu primjenjivi za potrebe rada, u kojem je važno održavati stalnu Wi-Fi vezu.

Ako je potrebno održati stalnu bežičnu vezu uz nižu potrošnju, koristi se načini rada Wi-Fi i Bluetooth *modem-sleep* i automatski *light-sleep*. Definicije funkcija i konstanti uključene su u `esp_wifi.h` i `esp_pm.h` bibliotekama. Način rada modema pokreće se funkcijom `esp_wifi_set_ps()` koja prima jednu od tri konstante:

- `PS_MODE_NONE`; zadani način rada u kojem mikrokontroler radi bez optimizacije potrošnje,
- `PS_MODE_MIN_MODEM`; u tom se načinu rada mikrokontroler postavljen u način rada stanice (STA) budi na svaki DTIM (engl. *Delivery Traffic Indication Message*) zbog primanja malih paketa podataka (engl. *beacon*). Za korisnika važni podaci ne mogu biti izgubljeni jer se šalju nakon DTIM-a, ali trajanje DTIM-a određuje pristupna točka, pa ne dolazi do veće uštede ako je ono kratko.
- `PS_MODE_MAX_MODEM`; način rada modema najveće uštede. Mikrokontroler se budi svaki interval osluškivanja (engl. *listen interval*) kako bi primio *beacon*. Što je taj interval duži, to je veća ušteda energije, ali korisnički se podaci mogu izgubiti ako je mikrokontroler u neaktivnom stanju tijekom DTIM-a [14].

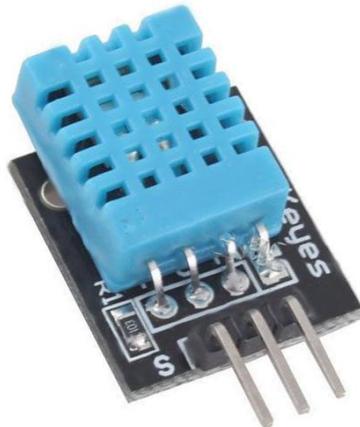
Interval osluškivanja postavlja se pozivom funkcije `esp_wifi_set_config()`, kojoj se predaje struktura `wifi_config_t`, u kojoj je interval definiran. Vrijeme u sekundama tijekom kojeg stanica (mikrokontroler) čeka *beacon* prije prekida veze postavlja se pozivom funkcije `esp_wifi_set_inactive_time()`.

Pozivom funkcije `esp_pm_configure()` mogu se podešiti dodatne postavke uštede energije. Funkciji se predaje struktura `esp_pm_config_t` u kojoj se definira minimalna i maksimalna frekvencija rada, pa se na taj način frekvencija dinamički skalira (engl. *Dynamic Frequency Scaling – DFS*). U istoj se strukturi omogućuje automatski način rada *light-sleep*, u koji mikrokontroler ulazi na određeno vrijeme ako ne obavlja zahtjevne zadatke.

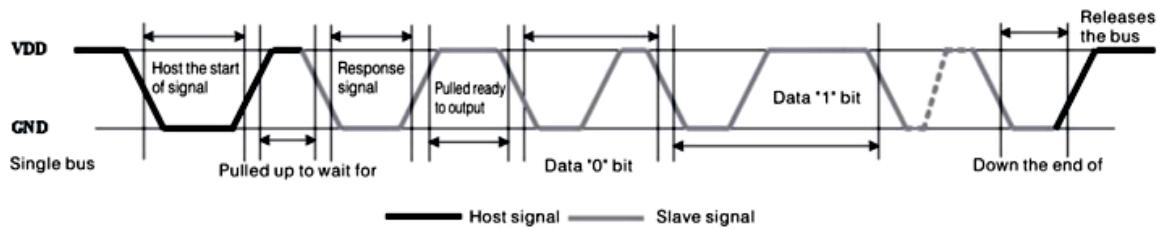
4. Senzori za praćenje okolišnih uvjeta

4.1. Senzor temperature i vlažnosti zraka DHT11

DHT11 digitalni je senzor temperature i vlažnosti zraka. Pokriva mjerno područje od 20 % do 90 % vlažnosti uz $+/- 5\%$ preciznosti te od 0 °C do 50 °C uz $+/- 2\%$ preciznost pri mjerenuju temperatu. Senzor je tvornički kalibriran, a mjerne su vrijednosti obiju veličina izražene 8-bitnom cijelobrojnom vrijednošću. Senzor na pločici prikazan je slikom (Slika 4.1). Komunikacija s mikrokontrolerom odvija se serijski preko jedne podatkovne linije (izvoda GPIO) u obama smjerovima (engl. *I-Wire*). Preko iste podatkovne linije obavlja se sinkronizacija i pravovremeno slanje podataka, a senzor postavlja vrijednosti na liniju samo kad mikrokontroler to zatraži. Prijenos podataka prikazan je slikom (Slika 4.2).



Slika 4.1 Pločica sa senzorom DHT11 [14].



Slika 4.2 Komunikacija mikrokontrolera (uredaj *host*) i senzora DHT11 (uredaj *slave*) [15].

Linija je prije početka komunikacije postavljena u stanje visoke naponske razine priteznim otpornicima, a senzor je u stanju niske potrošnje (izlazna struja oko 0,1 mA). Mikrokontroler pokreće komunikaciju slanjem *start* signala pritezanjem linije na nisku naponsku razinu minimalno 18 ms, nakon čega ju pušta i dopušta da se vrati na visoku razinu. To se postiže postavljanjem izvoda GPIO kao izlaznog, postavljanjem odgovarajuće razine i blokirajućom

odgodom u mikrosekundama pozivom funkcije `ets_delay_us()`. Kad senzor primi početni signal, prelazi u aktivno stanje (izlazna struja od 0,5 do 2,5 mA) te nakon kraće odgode šalje potvrdni signal povlačenjem linije na nisku naponsku razinu 80 µs i otpuštanjem linije 80 µs. Odgoda se osigurava blokirajućim procesom trajanja minimalno 20 µs. Odgovor senzora potrebno je programski provjeriti, što se postiže pozivom funkcije `wait_for_response()`. Funkcija prima maksimalno vrijeme čekanja i logičku razinu odgovora koja ga predstavlja te u slučaju pravovremenog odgovora vraća `ESP_OK`, a u protivnom pogrešku oblika `ESP_ERR_TIMEOUT`. Programski je kod funkcije prikazan u odsječku koda 4.1.

```
esp_err_t wait_for_response (int time_us, int level)
{
    int us_counter=0;
    while (gpio_get_level(GPIO_pin) == level)
    {
        ets_delay_us(1);
        us_counter++;
    }
    if (us_counter > time_us)
        return output.status = ESP_ERR_TIMEOUT;
    else
        return ESP_OK;
}
```

Kôd 4.1 – Funkcija za čekanje određene logičke razine.

Nakon odgovora senzora kojim se osigurava sinkronizacija, senzor počinje slati mjerne podatke predstavljene u 40 bitova (5 bajtova), a prije svakog bita čeka se 50 µs. Binarna je nula predstavljena pritezanjem linije na masu u trajanju kraćem od 28 µs, a jedinica pritezanjem linije između 28 i 70 µs. Dekodiranje znamenki provodi se funkcijom `sent_bit()` koja vraća odgovarajuću znamenku po opisanom pravilu ili pogrešku `ESP_ERR_INVALID_RESPONSE` ako je linija bila u niskoj naponskoj razini više od 70 µs. 40 se dekodiranih bitova pohranjuje u 5 bajtova u polju `DHT11_data[]`. Prvi i treći bajt predstavljaju cijelobrojne rezultate temperature odnosno vlažnosti zraka, drugi i četvrti bit uvijek su 0 (korisni su za precizniji senzor DHT22 koji pohranjuje decimalne rezultate), a peti je bajt kontrolna suma za provjeru valjanosti čitanja [16]. Ako su mjerni podatci

valjani, iznos kontrolne sume jednak je zbroju prethodna 4 bajta, u protivnom, program vraća pogrešku `ESP_ERR_INVALID_SIZE`. U svrhu pohrane mjernih rezultata i detekcije i obrade pogrešaka, inicijalizirana je statička struktura `output` tipa `dht11_reading`, u koju se pohranjuju temperatura, vlažnost zraka i status mjerena. Na početku svakog mjerena prepostavljeni je status `ESP_OK`, a status se ažurira u slučaju bilo koje od spomenutih grešaka. Time se pozivom funkcije `DHT11_read()` dohvaćaju mjerni rezultati, ali i točan izvor pogreške ako je do nje došlo. Valja napomenuti da je opisani jednolinijski protokol vremenski vrlo osjetljiv te da lako može doći do problema kod sinkronizacije. Zbog toga se slanje senzorskih podataka smatra kritičnom sekcijom koja se ograđuje funkcijama `portENTER_CRITICAL()` i `portEXIT_CRITICAL()`. Njima se sprječavaju prekidi i pozivi drugih funkcija u istoj dretvi dok se odvija mjerenje.

4.2. Senzori spojeni na A/D pretvornik

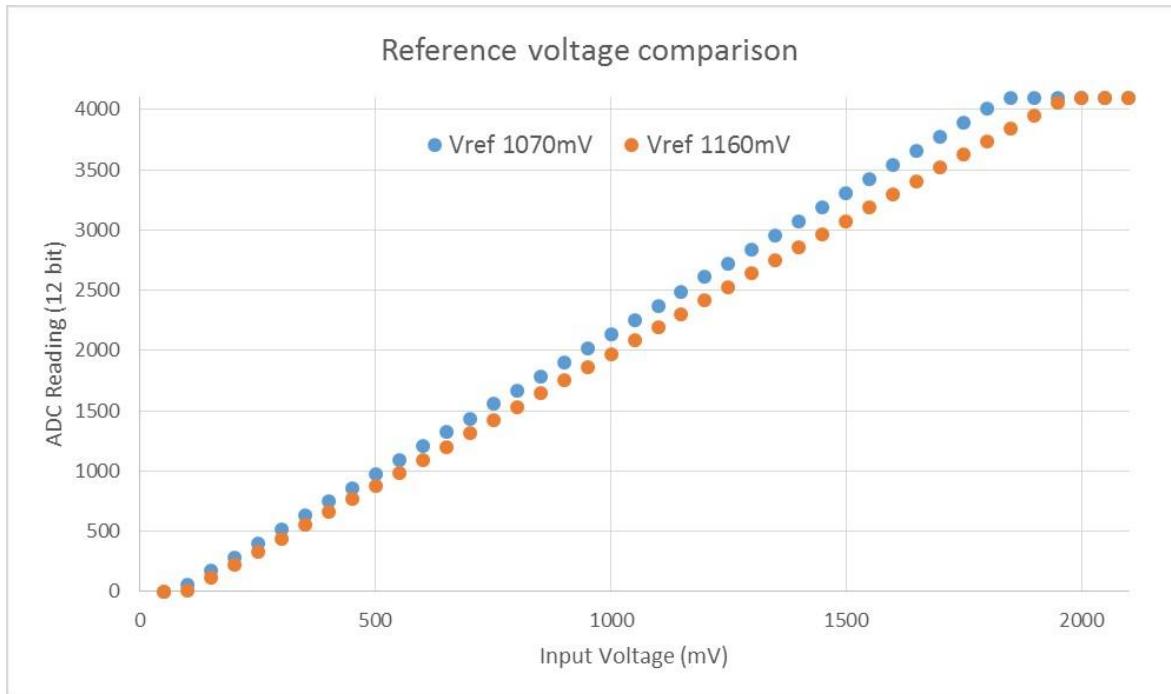
4.2.1. Analogno-digitalna pretvorba na ESP mikrokontroleru

ESP32 koristi dva analogno-digitalna pretvornika sa sukcesivnom aproksimacijom za očitavanje analognih vrijednosti preko 18 mjernih kanala (ulaznih izvoda GPIO s alternativnom funkcijom). Napon na izvodu uspoređuje s referentnim te se digitalna vrijednost u zadanoj rezoluciji pridružuje očitanoj vrijednosti. Sučelje podržava dva načina rada: *oneshot* (pretvorba se odvija na zahtjev korisnika) i *continuous* (kontinuirano pohranjivanje rezultata za obradu podataka u stvarnom vremenu). Za potrebe rada razmatrao se samo način rada *oneshot*, čije su funkcije definirane u biblioteci `esp_adc/adc_oneshot.h`. Prije dohvata rezultata potrebno je odabrati postavke mjerena pozivom funkcije `adc_oneshot_config_channel()`, kojoj se predaje referenca na odabrani SAR pretvornik (`adc1_handle` ili `adc2_handle`), odabrani mjni kanal te konfiguracija rezolucije rezultata i atenuacije mjerena. Uobičajen je raspon referentnog napona 1 – 1.2 V, a točan je iznos (ovisno o korištenoj pločici) pohranjen u *eFuse* vrijednosti te se automatski pridružuje ako je ta postavka omogućena u početnoj konfiguraciji (`CONFIG_ADC_CALI_EFUSE_VREF_ENABLE=y`). Atenuacijom se povećava merno područje ulaznog napona, a rasponi su prikazani u tablici (Tablica 4.1).

Tablica 4.1 Atenuacija i odgovarajuća mjerna područja ADC-a [17].

Atenuacija	Ulagani napon koji se može mjeriti
ADC_ATTEN_DB_0	100 mV – 950 mV
ADC_ATTEN_DB_2_5	100 mV – 1250 mV
ADC_ATTEN_DB_6	150 mV – 1750 mV
ADC_ATTEN_DB_12	150 mV – 2450 mV

U radu je odabrana 12-bitna rezolucija i 12 dB atenuacija. Nakon postavljanja kanala očitanje vrijednosti počinje pozivom funkcije `adc_oneshot_read()`, koja digitalni rezultat u zadanoj rezoluciji pohranjuje u jedan od parametara. Kako se uz stvarne mjerne veličine analognih senzora (vlažnost tla, kvaliteta zraka) najčešće vežu izmjerene vrijednosti napona, potrebno je dobiveni digitalni rezultat pretvoriti u odgovarajući iznos napona. To se može postići jednostavnim omjerom, no ESP32 nudi posebno kalibracijsko sučelje koje korisniku omogućuje dohvati preciznijih naponskih očitanja. U biblioteci `esp_adc/adc_cali.h` definirana je funkcija `adc_cali_create_scheme_line_fitting()`, kojom se stvara shema za pridruživanje digitalnih vrijednosti. Funkciji se predaju podatci o korištenom kanalu, atenuaciji i rezoluciji u vidu konfiguracije `adc_cali_line_fitting_config_t`, a pozivom se funkcije ulaznom parametru funkcije predaje referenca na novonastalu shemu. Nadalje, ta se referenca predaje funkciji `adc_cali_raw_to_voltage()`, koja digitalnoj vrijednosti pridružuje odgovarajući iznos napona. Tako se dobivaju rezultati koji u obzir uzimaju točnu vrijednost referentnog napona iz memorije *eFuse* (ako ona nije dostupna, izbacuje se upozorenje i koristi predefinirana vrijednost) [17]. Razlike u dobivenim vrijednostima u ovisnosti o dvjema različitim vrijednostima referentnog napona prikazane su slikom (**Slika 4.3**). Konačno, vrijednosti se napona mogu pridružiti senzorskim mernim rezultatima.



Slika 4.3 Prikaz odstupanja pri mjerenu s različitim referentnim naponima [17].

4.2.2. Senzor vlažnosti tla

Kapacitivni senzor vlažnosti tla (v1.2) široko je dostupan i jeftin analogni senzor. Napaja se preko pločice s 3,3 V te daje izlazni napon u efektivnom rasponu od 1,5 do 3 V (zbog toga je prikladna atenuacija ADC-a od 12 dB).

Ako se razmatraju samo jeftini komercijalno dostupni senzori, općenito postoje dva pristupa mjerenu vlažnosti tla. Prvi je direktno mjerene otpora tla između dviju sondi senzora. Kroz tlo se pušta struja malog iznosa pomoću koje se računa otpor, koji se smanjuje što je tlo vlažnije, odnosno što je veća koncentracija vode u tlu. Najveći je problem tog pristupa izravan kontakt vodljivih sondi s tlom. Kako se preko njih zatvara strujni krug, dolazi do vidljive elektrolitske korozije sondi već nakon mjesec dana korištenja (Slika 4.4), pri čemu senzor djelomično ili potpuno gubi svoju funkcionalnost [19].



Slika 4.4 Korozija sondi otporničkog senzora [19].

S druge strane, senzor je zbog načina rada izrazito osjetljiv na promjenu koncentracije iona u tlu. Tako dolazi do većih odstupanja promjenom sastava tla uz istu vlažnost, primjerice, dodatkom gnojiva. Usto, otpor kao električka veličina znatno ovisi o temperaturi, pa i najmanje oscilacije utječu na rezultat. Zato takav pristup nije prihvatljiv za potrebe rada te se umjesto toga koristi kapacitivno mjerjenje vlažnosti tla.

Senzori koji se temelje na kapacitivnoj metodi ustvari posredno mjere relativnu dielektričnu konstantu tla, koja se drastično povećava povećanjem koncentracije vode. Tipičan je iznos konstante za suho tlo 2 – 6, dok je za vodu blizu 80. Veličina preko koje se mjeri dielektričnost je rezonantna frekvencija kondenzatora određenog kapaciteta. Smanjivanjem dielektrične konstante dodatkom vode smanjuje se kapacitet, a iznos rezonantne frekvencije sklopa raste [20]. Frekvencija se zatim očitava brojačem LM555 i pretvara u analognu vrijednost; manju za veću vrijednost vlažnosti tla. U ovom pristupu nema izravnog kontakta vodljivih dijelova s tlom, pa je senzor otporan na koroziju. Također, temperaturna promjena kapaciteta u odnosu na otpor znatno je manja, pa je senzor prikladniji za stakleničke uvjete. Izgled senzora prikazan je slikom (Slika 4.5).



Slika 4.5 Kapacitivni senzor vlažnosti tla v1.2 [21].

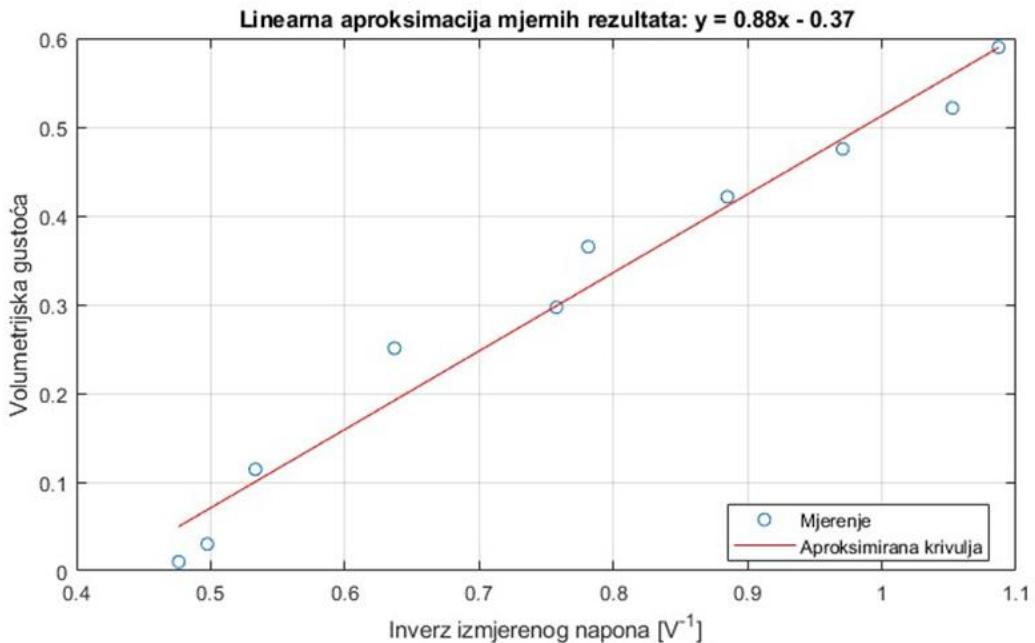
Kalibracija senzora izvodi se pridruživanjem izmjerениh vrijednosti volumetrijske vlažnosti tla odgovarajućim iznosima napona. Provedeno je deset mjerena, pri čemu se vlažnost tla (θ_v) računala preko mase mokrog (m_s) i suhog (m_d) tla te gustoće vode (ρ_w) i nasipne gustoće tla ($\rho_{d,s}$) izrazom (1):

$$\theta_v = \left(\frac{m_s - m_d}{m_d} \right) \frac{\rho_{d,s}}{\rho_w} \quad (1)$$

Detaljniji je postupak mjerena opisan u radu [20]. Izmjerene su vrijednosti potom unesene u konstante VOLTAGE_READINGS odnosno MOISTURE_READINGS u datoteci soil_moisture.h. Prepostavljen je linearni odnos između inverza naponske razine i odgovarajuće vlažnosti tla te se u funkciji soil_moisture_init() metodom najmanjih kvadrata određuju koeficijenti aproksimiranog pravca. Mjerni su rezultati prikazani u tablici (Tablica 4.2), a aproksimacija pravcem slikom (Slika 4.6). Tom je metodom dobiven izraz s korijenom srednje kvadratne pogreške (RMSE) iznosa 0.032377 (odstupanje pri mjerenu vlažnosti ispod 5 %), koja je u prikladnom rasponu za uređaje slične primjene [20]. Taj se izraz koristi u funkciji read_soil_moisture(), čijim se pozivom dohvata konačna vrijednost vlažnosti tla.

Tablica 4.2 Rezultati mjerena vlažnosti tla u pokaznom modelu.

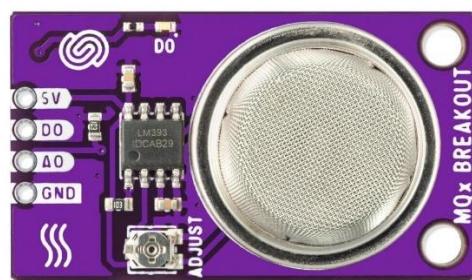
Volumetrijska vlažnost tla [%]	Izmjereni napon [V]
1,00	2,1
3,009	2,01
11,434	1,875
25,075	1,57
29,689	1,32
36,510	1,28
42,127	1,13
47,543	1,03
52,157	0,95
58,980	0,92



Slika 4.6 Aproksimirana krivulja kalibracije nacrtana alatom MATLAB.

4.2.3. Senzor plinova MQ-135

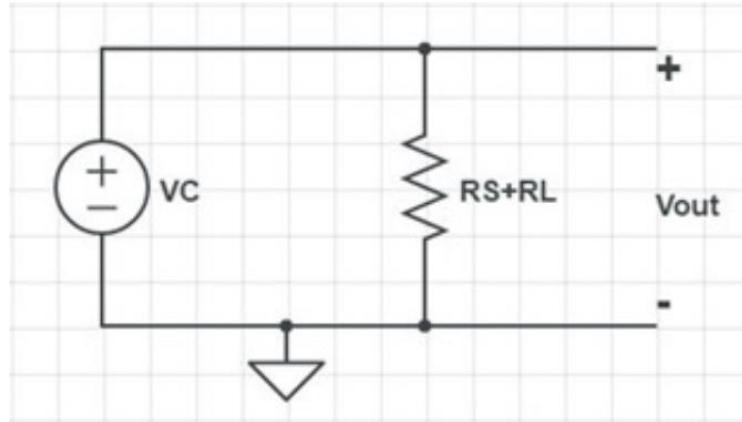
Senzori oznake MQ vrlo su dostupni i jeftini analogni senzori za raspoznavanje atmosferskih plinova. Senzori koriste metal-oksidni poluvodički materijal (najčešće SnO₂), koji mijenja otpor u kontaktu s plinom. Kako se otpor mijenja i promjenom temperature, u senzorima se postavlja i grijača žica, koja održava temperaturu senzora visokom i time smanjuje temperaturne oscilacije. Zbog toga je prije dohvata prvih točnih mjernih rezultata potrebno zagrijavati senzor određeno vrijeme koje propiše proizvođač [22]. Senzor oznake MQ-135 prikazan je na slici (Slika 4.7).



Slika 4.7 Pločica sa senzorom MQ-135 [27].

MQ-135 podešen je tako da mijenja otpor ovisno o koncentraciji amonijaka, parama alkohola, benzena, ugljikova dioksida, dima i drugih plinova. Kao osjetljivu tvar koristi SnO₂, poluvodički materijal n-tipa, kojem otpor pada u doticaju sa spomenutim

reducirajućim plinovima. Senzor se napaja preko 5 V te sadrži analogni i digitalni izlaz. Na digitalni se izlaz postavlja logička jedinica ovisno o naponu postavljenom potenciometrom te se neće daljnje razmatrati u radu. Analogni je izlazni signal u rasponu 0 – 5 V te se koristi pri izračunu otpora senzora (R_S) koji ovisi o koncentraciji određenog plina u zraku. Taj otpor zajedno s teretnim otporom (R_L) iznosa 10k zatvara izlazni strujni krug prikazan slikom (Slika 4.8). Otpor senzora računa se izrazom (1), pri čemu je V_{RL} napon na otporniku R_L odnosno mjereni analogni izlaz.

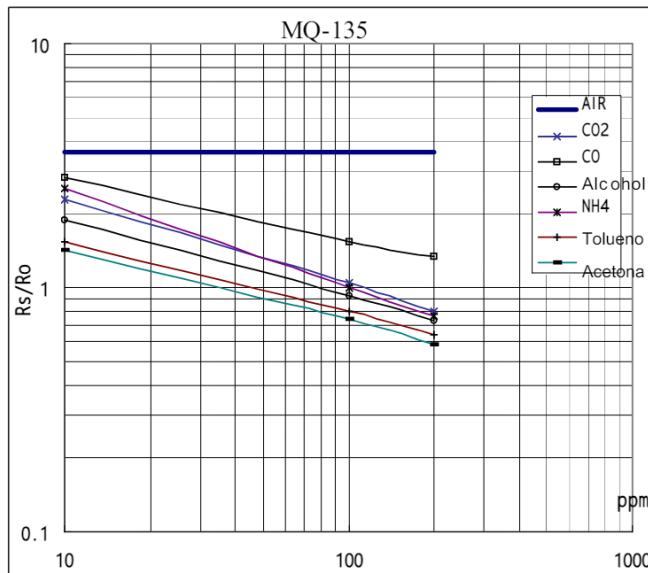


Slika 4.8 Pojednostavljena shema izlaznog strujnog kruga senzora MQ-135 [26].

$$R_S = \frac{V_C \cdot R_L}{V_{RL}} - R_L \quad (1)$$

Odgovarajuća koncentracija plinova pridružuje se danom otporu po shemi koju osigurava proizvođač. Senzor će se koristiti za mjerjenje koncentracije ugljikova dioksida, pa se razmatra kalibracijska krivulja tog plina. Kako je shema pridruživanja dana u logaritamskoj skali, krivulja se prilagođava eksponencijalnoj funkciji oblika $y = ax^b$, gdje se aproksimiraju koeficijenti a i b pomoću točaka na shemi prikazanoj slikom (Slika 4.9). Tim postupkom [25] dobiven je izraz (2), pri čemu je R_0 referentni otpor proizvođača izmјeren pri standardnim uvjetima i koncentracije 100 ppm amonijaka u zraku [22].

$$ppm = 116.6 \cdot \left(\frac{R_S}{R_0} \right)^{-2.769} \quad (2)$$



Slika 4.9 Osjetljivost senzora MQ-135 na različite atmosferske plinove [22].

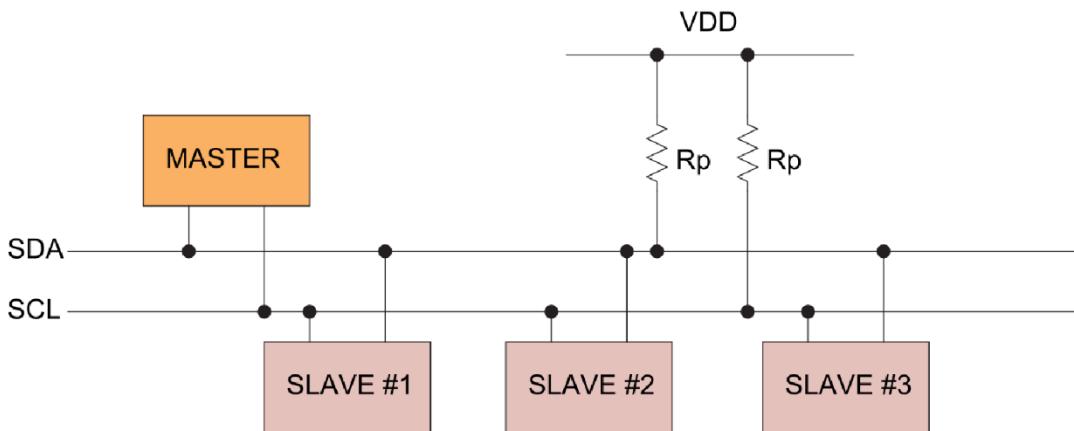
Za potrebe rada R_0 izračunat je iz izraza (1) iz izmijerenog otpora senzora R_s pri poznatoj koncentraciji ugljikova dioksida. Kako se nije provodilo mjerjenje koncentracije drugim kalibriranim uređajem, uzeta je atmosferska vrijednost od 425 ppm [24], a otpor R_0 dobiven usrednjavanjem više izmijerenih vrijednosti izlaznog napona iznosi 122264 Ω . Dobivene su vrijednosti upisane kao konstante u datoteku MQ135.h. Funkcija `MQ135_init()` inicijalizira ADC, `MQ135_read_voltage()` vraća rezultat pretvorbe u voltima, a `MQ135_read()` računa konačnu vrijednost koncentracije ugljikova dioksida u zraku pomoću izraza (2).

Valja napomenuti da se opisanim postupkom ne mogu dobiti potpuno točni rezultati. Senzori oznake MQ ne razlikuju tipove detektiranih plinova. Promjena vrijednosti ppm ne znači nužno promjenu koncentracije ugljikova dioksida u zraku, nego bilo kojeg reducirajućeg plina koji mijenja otpor senzora. Osim toga, neprecizna aproksimacija referentnog otpora R_0 također je znatan izvor pogreške. Zbog toga se senzor MQ-135 u radu može koristiti za grubo oslikavanje koncentracije ugljikova dioksida u stakleniku i kao sigurnosni mehanizam protiv požara ako se koncentracija poveća iznad dozvoljene granice. Za precizno mjerjenje koncentracija plinova u zraku koriste se skupi i precizni digitalni senzori posebno prilagođeni karakteristikama određenog plina [23].

4.3. Senzori spojeni putem I2C sučelja

4.3.1. I2C komunikacijski protokol

Razvojem sve složenijih ugradbenih sustava pojavljuje se potreba za komunikacijom sa sve većim brojem perifernih uređaja. U isto se vrijeme teži sve manjim dimenzijama uređaja, a veći broj veza povećava fizičke dimenzije pločice. Dakle, potrebno je omogućiti spajanje što većeg broja vanjskih uređaja bez nagomilavanja električnih veza. U tu svrhu 1982. tvrtka Philips Semiconductor (danasa NXP Semidonductors) razvija komunikacijski protokol I2C (engl. *I-squared-C*) [28]. Komunikacija između uređaja koji podržavaju taj protokol odvija se jednostavnom dvožičnom sabirnicom pri čemu se uređaji jedinstveno adresiraju. Prijenos podataka odvija se pri nižim brzinama (od 100 kHz do 5 MHz, najčešće 400 kHz), a maksimalan broj povezanih uređaja određen je širinom adresa (tipično 112 kod 7-bitnih adresa, znatno više kod 10-bitnih). Dvije dvosmjerne linije tipa *open drain* preko kojih se odvija komunikacija jesu SDA (serijska podatkovna linija), preko koje se šalju podatci, i SCL (serijska taktna linija), kojom je određen takt prijenosa. Uredaj koji inicira prijenos podataka naziva se *master*, a uređaji koje adresira smatraju se uređajima *slave*. Podatci koji se prenose sastoje se od adrese uređaja *slave* i kontrolnih bitova za početak, završetak, smjer i mehanizam potvrđivanja prijenosa [29]. Spajanje uređaja prikazano je slikom (Slika 4.10).



Slika 4.10 Blokovska shema spajanja uređaja pri I2C komunikaciji Pogreška! Izvor reference nije pronađen. .

Prije početka prijenosa (engl. *idle*) obje su linije priteznim otpornicima postavljene u stanje logičke jedinice. Početak prijenosa inicira *master* spuštanjem SDA linije iz stanja "1" u stanje "0" dok je linija SCL u stanju "1". Nakon *start* uvjeta *master* na liniju postavlja 7-bitnu adresu uređaja kojemu pristupa i bit kojim određuje hoće li čitati podatak s SDA linije

ili će postaviti podatak na nju. Nakon što *slave* potvrdi da je ispravno adresiran slanjem signala ACK, podatci se mogu postaviti na liniju. Ako se radi o operaciji čitanja, *slave* postavlja 8-bitne podatke, a *master* prvo potvrđuje da ih je primio slanjem signala ACK te ih potom čita. U slučaju pisanja *master* postavlja podatke na podatkovnu liniju, a *slave* potvrđuje da ih je primio te ih potom čita. Podatci se mogu čitati odnosno pisati s linije sve dok *master* signalom NACK ne označi posljednji podatak i prekine prijenos slanjem *stop* uvjeta postavljanjem SDA linije u “1” dok je SCL u “1”.

4.3.2. I2C komunikacija na ESP32 mikrokontroleru

Mikrokontroler ESP32 posjeduje dva I2C kontrolera (engl. *port*), koji se mogu konfigurirati u načinu rada *master* ili *slave*. Način rada *slave* podrazumijeva drugi mikrokontroler koji preuzima ulogu uređaja *master*, pa se neće dalje razmatrati u ovom radu. Način rada *master* jednostavno se implementira API-jima definiranim u biblioteci *i2c_master.h*. Prije prijenosa podataka potrebno je inicijalizirati uređaj *master* i dvolinijsku sabirnicu preko koje komunicira. Prvo se inicijalizira sabirnica pozivom funkcije *i2c_new_master_bus* kojoj se predaje konfiguracija *i2c_master_bus_config_t* i upravljač (engl. *handle*) *i2c_master_bus_handle_t*, koji predstavlja sabirnicu. U konfiguraciji postavlja se I2C port, broj izvoda GPIO za SDA i SCL, izvor takta (100 kHz za standardni način rada, 400 kHz za brzi način rada), period pogreške (ako pogreška u komunikaciji traje manje od zadanog perioda, moguće ju je filtrirati), pritezni otpornik linije i dr. Nakon poziva funkcije stvara se instanca konfiguirirane sabirnice koja se predaje idućoj funkciji *i2c_master_add_device()*, kojom se inicijalizira master uređaj. Funkciji je također potrebno predati konfiguraciju *i2c_device_config_t*, kojom se definira adresa uređaja kojem uređaj *master* pristupa i njegina širina (7 ili iznimno 10 bita) te brzina takta SCL. Programski je kod za inicijalizaciju sučelja prikazan programskim odsječkom 4.2.

```
i2c_master_bus_config_t i2c_mst_config = {
    .clk_source = I2C_CLK_SRC_DEFAULT,
    .i2c_port = I2C_PORT,
    .scl_io_num = I2C_MASTER_SCL_IO,
    .sda_io_num = I2C_MASTER_SDA_IO,
    .glitch_ignore_cnt = 7,
    .flags.enable_internal_pullup = true,
};

i2c_master_bus_handle_t bus_handle;
```

```

ESP_ERROR_CHECK(i2c_new_master_bus(&i2c_mst_config,
&bus_handle));

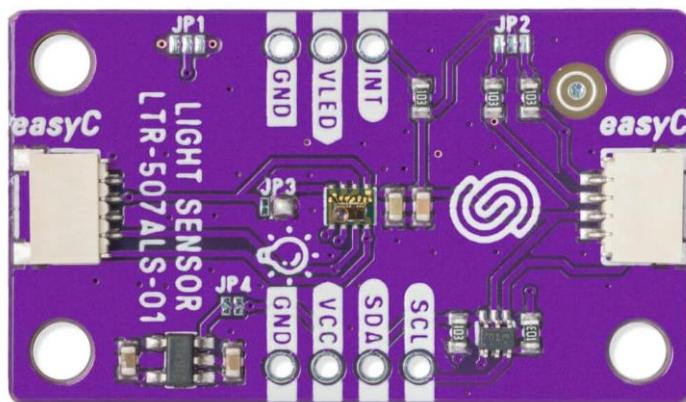
```

Kôd 4.2 – Inicijalizacija sučelja I2C.

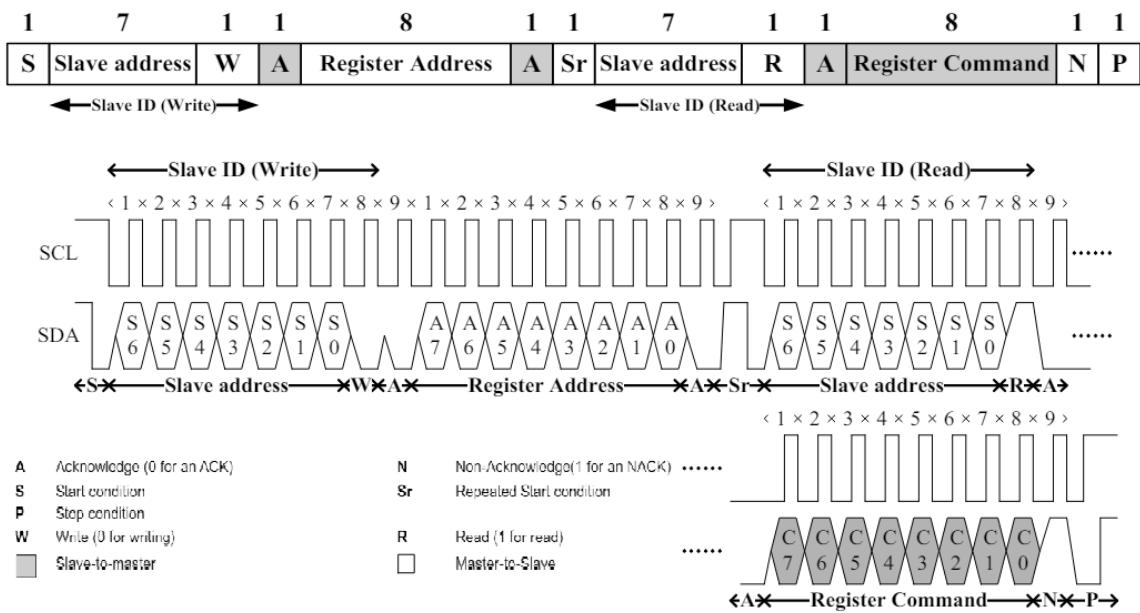
Konačno, nastaje konfigurirani uređaj *master* predstavljen s *i2c_master_dev_handle_t*, kojim se započinje prijenos podataka. Funkcijom *i2c_master_transmit()* mikrokontroler piše podatke uređaju *slave*, pozivom *i2c_master_receive()* čita se iz njega, a pomoću *i2c_transmit_and_receive()* master prvo piše pa čita podatke s linije [18].

4.3.3. Senzor ambijentnog osvjetljenja i udaljenosti LTR-507

LTR-507ALS-01 optički je digitalni senzor ambijentnog osvjetljenja i udaljenosti. Senzor pretvara analogni intenzitet svjetlosti u digitalni signal koji se odašilje sučeljem I2C. Uređaj nudi spektralni odziv sličan onome kod ljudskog oka, radi u različitim uvjetima osvjetljenja te automatski odbacuje 50 i 60 Hz treperenje, koje bi narušilo mjerjenje. Također se može koristiti i za procjenu udaljenosti s ugrađenim infracrvenim LED driverom, ali ta funkcionalnost nije implementirana u svrhe ovog rada. Mikrokontroler komunicira sa senzorom standardnim (100 kHz), brzim (400 kHz) i jako brzim (3,4 MHz) prijenosom I2C, pri čemu se senzor postavlja u stanje niske potrošnje (engl. *standby/sleep mode*) kad se na sabirnici ne odvija prijenos [31]. Prije prvog mjerjenja potrebno je postaviti početne postavke senzora. To se, kao i pristupanje mjernim podatcima, obavlja upisom određenih konstanti u podatkovne registre senzora. Pločica sa senzorom prikazana je slikom (Slika 4.11), a primjer upisa vrijednosti u podatkovni register shemom (Slika 4.12).



Slika 4.11 Pločica sa senzorom LTR-507-ALS-01 [32].



Slika 4.12 Pisanje u podatkovni registar senzora LTR-507-ALS-01 [31].

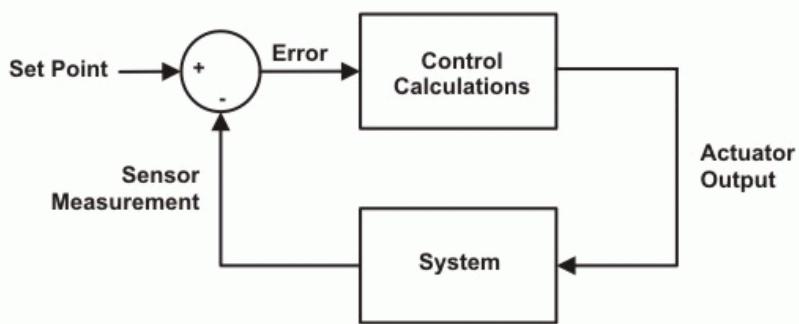
Prvo se inicijalizira sučelje I₂C uređaja na prethodno opisan način; adresa senzora je 7-bitna iznosa 0x23. Registrima za inicijalizaciju pristupa se tako da se na liniju SDA prvo pošalje 8-bitna adresa registra, a potom se iščitava ili ažurira njegova vrijednost ovisno o tome radi li se o operaciji čitanja ili pisanja. Za čitanje vrijednosti registara najprikladnija je funkcija `i2c_transmit_and_receive()`, kojom se prvo upisuje adresa registra te se čita njegova vrijednost, a za pisanje u registar koristi se funkcija `i2c_transmit()`, pri čemu se odmah prenose dva podatka (prvo adresu pa nova vrijednost registra). Na taj način u funkciji `ALS_Enable()` senzor se iz neaktivnog (*standby*) stanja postavlja u aktivno, a zatim se namještaju drugi parametri: dinamički raspon mjerenja (`setALSGain()`), rezolucija analogno-digitalnog pretvornika (`setALSBitWidth()`) te brzina očitavanja (`setALSMesRate()`). Konačno, 16-bitna cijelobrojna vrijednost intenziteta svjetlosti izražena u luxima dohvata se pozivom funkcije `LTR507_getLightIntensity()`, pri čemu se prije pristupanja podatkovnom registru provjerava ispravnost rezultata čitanjem vrijednosti statusnog registra `LTR507_ALS_PS_STATUS_REG`.

5. Aktuatori za automatsku regulaciju uvjeta

Aktuatori u pokaznom modelu implementiraju ručnu i automatsku regulaciju. Kako osvjetljenje u stakleniku ovisi o biljnoj vrsti i dnevnom ciklusu, ono nije izravno povezano sa senzorskim mjerjenjem u mjernoj jedinici lux. Zbog toga je za rasvjetu omogućena ručna regulacija, pri čemu se ona pali i gasi s obzirom na korisnikov unos. Iznimno, u korisničkoj će aplikaciji biti omogućeno automatsko upravljanje rasvjetom ovisno o vremenu tijekom dana. Na primjeru vodene pumpe (3,6 W) i električnog grijачa (50 W) ilustrirat će se osnovni principi automatske regulacije.

5.1. Algoritmi automatske regulacije

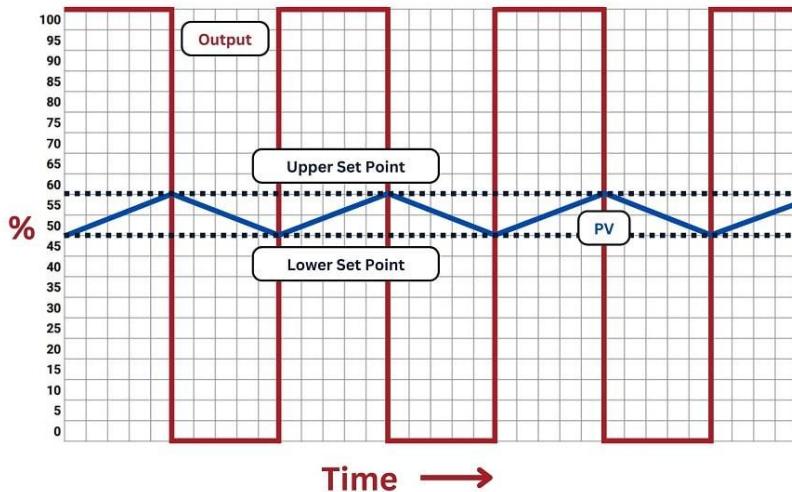
Najpopularniji tip industrijskog automatskog upravljanja jest PID regulacija. Mjerna vrijednost senzora negativnom se povratnom vezom (Slika 5.1) oduzima od korisnički postavljene vrijednosti te se rezultat provlači kroz PID regulator. PID regulator zasniva se na proporcionalnom (P), integracijskom (I) i derivacijskom (D) djelovanju, kojima se podešavaju aktuatorski parametri. Takav se pristup koristi u 95 % industrijskih primjena, nudi mogućnosti automatskog podešavanja (engl. *autotuning*) i samopodešavanja (engl. *selftuning*) parametara [33]. Problem takva pristupa jest što zahtijeva kontinuiranu kontrolu nad aktuatorskim izlazom. Kako će se u pokaznom modelu grijачem i pumpom upravljati sklopkama (relejom i tranzistorskom sklopkom), dakle bez mogućnosti kontrole snage pumpanja i jačine grijanja, PID regulacija nije primjenjiva.



Slika 5.1 Osnovna ideja automatskog upravljanja s povratnom vezom [35].

Za sustave bez mogućnosti podešavanja primjenjiva je automatska regulacija tipa *uključi-isključi* (engl. *on/off, bang-bang*). Metoda se zasniva na postavljanju gornje i donje granice mјerenog parametra te se paljenjem ili gašenjem aktuatora veličina održava u željenom

intervalu. Dakle, koristi se samo proporcionalno regulatorsko djelovanje, kojim se upravlja aktuatorima [34]. Osnovni je princip te metode prikazan slikom (Slika 5.2). Iako je ovaj tip regulatora manje precizan od prethodnog, izrazito je jednostavan za realizaciju i primjenjiv u mnogim situacijama.



Slika 5.2 Osnovni princip automatske regulacije tipa *uključi-isključi* [34].

5.2. Izvedba rješenja u sustavu za regulaciju uvjeta u stakleniku

Za pokazni je model korišten modificirani oblik regulacija tipa *uključi-isključi* bez postavljanja dviju granica. Iznad opisani način primjenjiv je za situacije u kojima se senzorsko mjerjenje provodi gotovo kontinuirano. U pokaznom će se modelu mjerene veličine očitavati rijetko zbog smanjenja potrošnje. Češća mjerjenja nisu nužna jer ne dolazi do većih oscilacija promatranih parametara, a i time se osigurava dovoljno vremena za pripremu mjernih rezultata kod sporijih senzora.

Ilustracijski je princip regulacije izrazito jednostavan: pri svakom se senzorskom očitanju provjerava uvjet prema kojem se po potrebi uključuje odgovarajući aktuator na definirano vrijeme. Dakle, pumpa i grijач će se periodički paliti svakoga intervala očitavanja ako je vlažnost tla odnosno temperatura zraka manja od korisnički definirane vrijednosti. Trajanje navodnjavanja odnosno grijanja postavlja se eksperimentalno, ovisno o radu aktuatora.

Za periodička očitavanja i aktivaciju aktuatora koriste se brojači u sklopu operacijskog sustava za rad u stvarnom vremenu pod nazivom FreeRTOS (engl. *Real Time Operating System – RTOS*) integriranog u ESP32. Funkcije za brojače definirane su u biblioteci

freertos/timers.h. Funkcijom `xTimerCreate()` stvara se brojač kojem se pridružuje ime, vrijednost do koje se broji u otkucajima (engl. *tick*), zastavica koja se postavlja na `pdTrue` ako je brojač periodičan, identifikator te funkcija koja se poziva kad brojač odbroji do zadane vrijednosti. Otkucaji se preračunavaju u milisekunde korištenjem konstante `portTICK_PERIOD_MS`. Na taj način inicijalizirani su brojači za senzore, vodenu pumpu i grijač, koji započinju brojenje pozivom funkcije `xTimerStart()`. U funkciji povratnog poziva senzora pozivaju se funkcije za mjerjenje iz prethodnog poglavlja te se uvjetno pokreću aktuatori postavljanjem odgovarajućih izvoda GPIO za upravljanje relejima. Programski kod stvaranja brojača i funkcije povratnog poziva za vodenu pumpu prikazan je programskim odsječkom 5.1.

```

esp_err_t WATER_PUMP_init(void)
{
    gpio_set_direction(WATER_PUMP_GPIO, GPIO_MODE_OUTPUT);
    WATER_PUMP_timer = xTimerCreate("WATER_PUMP_update",
        (WATER_PUMP_ACTIVE_DURATION * 1000) / portTICK_PERIOD_MS,
        pdFALSE, NULL, WATER_PUMP_update);
    if (WATER_PUMP_timer)
        return ESP_OK;
    return ESP_FAIL;
}

static void HEATER_update(TimerHandle_t handle)
{
    HEATER_BUSY = false;
    gpio_set_level(HEATER_GPIO, 0);
    esp_rmaker_param_update_and_report(
        esp_rmaker_device_get_param_by_name(Heater, "Status"),
        esp_rmaker_bool(HEATER_BUSY));
}

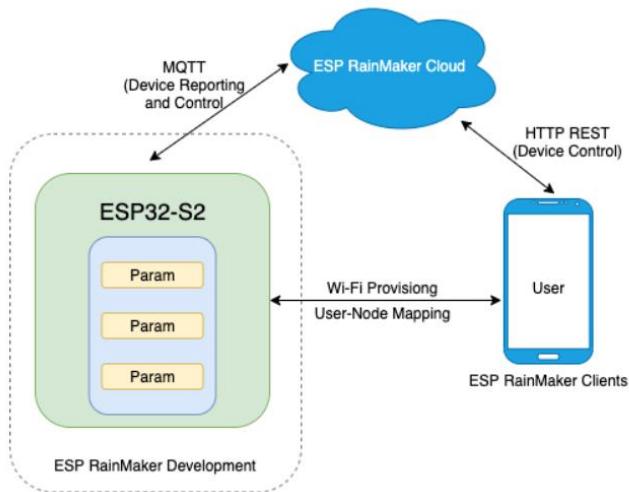
```

Kôd 5.1 – Funkcije stvaranja brojača (iznad) i povratnog poziva (ispod) vodene pumpe.

6. Platforma ESP RainMaker

ESP RainMaker platforma je koju nudi Espressif za IoT rješenja koja zahtijevaju praćenje i automatsko upravljanje uređajima na daljinu. Potpuno je prilagođena razvoju softvera na seriji čipova ESP32. Za te uređaje pruža potpuni alat za razvijanje softvera (SDK) i mobilnu aplikaciju, transparentni posrednički sloj u oblaku te alate za osobno računalo. Komunikacija uređaja s oblakom i razvoj aplikacije znatno su olakšani jednostavnim aplikacijskim programskim sučeljima integriranim u ESP-IDF [36].

Čvor je glavna logička jedinica ekosustava ESP RainMaker. On predstavlja individualni uređaj (sustav s mikrokontrolerom) koji se preko klijenta (mobilne aplikacije ili sučelja naredbenog retka) povezuje na oblak. Veza između čvora i oblaka postiže se sigurnim protokolom za razmjenu poruka u stvarnom vremenu MQTT (engl. *Message Queuing Telemetry Transport*). Veza se osigurava protokolom TLS (engl. *Transport Layer Security*), kojim se generiraju ključevi po certifikatu X.509 za pristupanje oblaku tijekom koraka primanja zahtjeva [38]. Princip komunikacije prikazan je slikom (Slika 6.1).



Slika 6.1 Princip komunikacije između čvora, klijenta i oblaka [41].

6.1. Povezivanje

Prvo je potrebno uspostaviti komunikaciju između čvora i klijenta, što se postiže u koraku postavljanja Wi-Fi mreže (engl. *Wi-Fi provisioning*). U tom koraku klijent šalje jedinstvenu korisničku oznaku i slučajno generirani tajni ključ čvoru (mikrokontroleru), a zauzvrat od njega dobiva identifikator čvora (engl. *node Id*). Oznaka se čvora zajedno s tajnim ključem

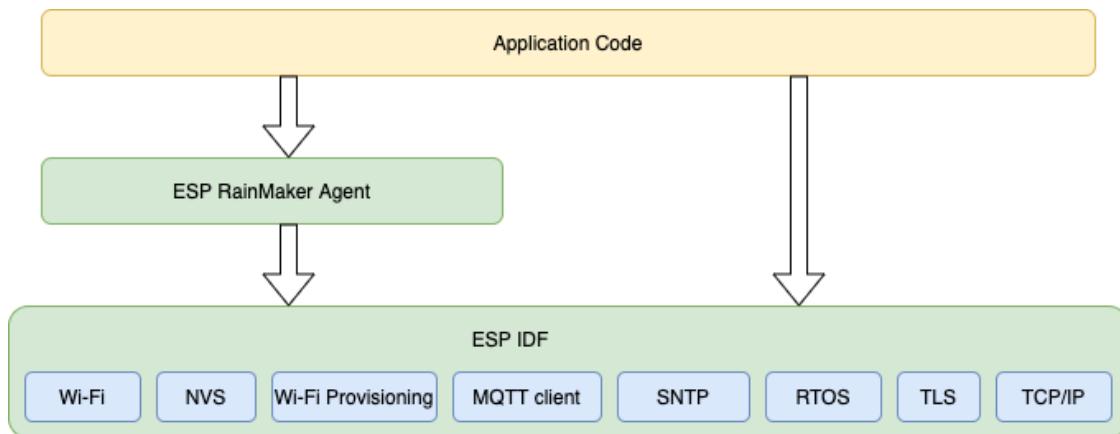
šalje oblaku usluge RainMaker. S druge strane, mikrokontroler izravno komunicira s oblakom protokolom MQTT i šalje mu iste podatke (identifikator čvora, korisničku oznaku, tajni ključ). Klijent provjerava odgovor oblaka (engl. *request Id*); ako se podatci čvora i klijenta poklapaju, povezivanje je uspješno, u protivnom se nastavlja provjeravanje do isteka dozvoljenog vremena [39].

Izravna komunikacija između čvora i oblaka moguća je samo ako je veza potvrđena. Potvrđivanje veze (engl. *claiming*) moguće je na više načina:

- samopotvrđivanje; ako je tajni ključ tvornički postavljen u memoriju *eFuse* mikrokontrolera,
- potpomognuto potvrđivanje (zadano), u kojem klijent (mobilna aplikacija), povezan s čvorom BLE vezom, razmjenjuje podatke s oblakom i omogućuje vezu,
- potvrđivanje koje vodi domaćin (engl. *host*), pri čemu korisnik uspostavlja vezu posebnim zahtjevom nakon prijave na korisnički račun [40].

6.2. Programski kod

Cijela je programska podrška pokaznog modela u potpunosti podređena aplikacijskim programskim sučeljima agenta ESP RainMaker. Korisnički je kod uokviren aplikacijskim funkcijama, koje preko agenta uspostavljaju komunikaciju s klijentom. Takva je organizacija prikazana slikom (Slika 6.2).



Slika 6.2 Proces softverskog razvoja preko agenta ESP RainMaker [37].

Struktura je projektnog direktorija sljedeća:

```
greenhouse/
  └── .vscode/
  └── build/
  └── main/
    ├── app_driver.c
    ├── app_main.c
    ├── app_priv.h
    ├── CMakeLists.txt
    ├── component.mk
    ├── device.c
    ├── device.h
    ├── DHT11.c
    ├── DHT11.h
    ├── Kconfig.projbuild
    ├── LTR-507.c
    ├── LTR-507.h
    ├── MQ-135.c
    ├── MQ-135.h
    ├── soil_moisture.c
    └── soil_moisture.h
  └── managed_components/
  └── CMakeLists.txt
  └── dependencies.lock
  └── Makefile
  └── partitions_4mb_optimised.csv
  └── partitions.csv
  └── README.md
  └── sdkconfig
  └── sdkconfig.defaults
  └── sdkconfig.defaults.esp32
  └── sdkconfig.defaults.esp32c6
  └── sdkconfig.defaults.esp32s2
  └── sdkconfig.old
```

Kôd 6.1 – Struktura projektnog direktorija.

U datoteci `app_main.c` napisan je glavni program `app_main()`, koji se izvodi na mikrokontroleru. U njemu se prvo poziva funkcija `app_driver_init()` definirana u datoteci `app_driver.c`, u kojoj se nalazi korisnički program. Korisnički se program izvodi neovisno o funkcijama sučelja RainMaker. U njemu se inicijaliziraju senzori i aktuatori, početne se vrijednosti mjerena i stanja aktuatora spremaju u globalne varijable te se pokreću brojači za senzorska očitavanja i rad aktuatora. Pri tome se koriste datoteke s imenima senzora, u kojima su implementirane funkcije iz prethodnih poglavlja. Period očitavanja, trajanje rada aktuatora i druge konstante definirane su u datoteci `app_priv.h`. Nakon pokretanja korisničkog programa slijede pozivi funkcija sučelja RainMaker.

Prvo se poziva funkcija `app_wifi_init()`, kojom se inicijalizira Wi-Fi mreža i postavke za *Wi-Fi provisioning*. Zatim se inicijalizira čvor pozivom funkcije `esp_rmaker_node_init()`, tako da mu se pridruži konfiguracija tipa

`esp_rmaker_config_t`, ime i tip. Čvor sam po sebi ne posjeduje nikakvu vrijednost, pa mu je zato potrebno pridružiti uređaj tipa `esp_rmaker_device_t`.

Uređaj u pokaznom modelu predstavlja senzor ili aktuator, kojemu se pridružuju parametri vidljivi korisniku u mobilnoj aplikaciji. Uređaji su kreirani pozivima funkcija definiranih u datoteci `device.c`. Primjerice, uređaju DHT11 pridruženo je ime te parametri temperature i vlažnosti zraka. Ti se parametri periodički ažuriraju i dostavljaju oblaku pri svakom senzorskom čitanju pozivom funkcije `esp_rmaker_param_update_and_report()`. Jednostavnim postavljanjem zastavice `PROP_FLAG_TIME_SERIES` na uređaj omogućeno je iscrtavanje vrijednosti parametara pohranjenih u oblaku. Funkcija stvaranja uređaja prikazana je programskim odsječkom 6.2. Pri stvaranju uređaja korištene su već gotove komponente grafičkog sučelja definirane u biblioteci `esp_rmaker_standard_params.h`.

```
esp_rmaker_device_t *humiture_device_create(const char
                                              *dev_name, void *priv_data, int temperature, int humidity)
{
    esp_rmaker_device_t *device =
        esp_rmaker_device_create(dev_name,
                                 ESP_RMAKER_DEVICE_TEMP_SENSOR, priv_data);
    esp_rmaker_device_add_param(device,
                                esp_rmaker_name_param_create(ESP_RMAKER_DEF_NAME_PARAM,
                                dev_name));
    esp_rmaker_device_add_param(device,
                                esp_rmaker_param_create("Temperature", NULL,
                                esp_rmaker_float((float)temperature), PROP_FLAG_READ |
                                PROP_FLAG_TIME_SERIES));
    esp_rmaker_device_add_param(device,
                                esp_rmaker_param_create("Humidity", NULL,
                                esp_rmaker_float((float)humidity), PROP_FLAG_READ |
                                PROP_FLAG_TIME_SERIES));
    esp_rmaker_device_assign_primary_param(device,
                                           esp_rmaker_device_get_param_by_name(device, "Temperature"));
    return device;
}
```

Kôd 6.2 – Stvaranje uređaja za senzor DHT11 u datoteci `device.c`

Aktuatorским uređajima u datoteci `app_main.c` također je pridružena funkcija povratnog poziva. Kad korisnik unese željenu temperaturu odnosno vlažnost tla ili uključi LED rasvjetu, promjena parametra ažurira se u mobilnoj aplikaciji te se pošalje na oblak. Oblak tada pošalje zahtjev za ulazak u funkciju povratnog poziva. U toj se funkciji prepoznaže uređaj za koji je poslan zahtjev te se unesena vrijednost postavlja za željeni aktuator. Dio koda prikazan je ispod odlomka.

```
//pridruživanje funkcije povratnog poziva
esp_rmaker_device_add_cb(Heater, actuator_write_cb, NULL);

//funkcija povratnog poziva
static esp_err_t actuator_write_cb(const esp_rmaker_device_t
*device, const esp_rmaker_param_t *param,
const esp_rmaker_param_val_t val, void *priv_data,
esp_rmaker_write_ctx_t *ctx)
{
    ...
    //postavljanje temperature termostata
    app_set_heater(val.val.i);
    ...
}
```

Kôd 6.3 – Funkcija povratnog poziva i njezino pridruživanje grijaču

Nakon pridruživanja uređaja čvoru, pokreće se agent sučelja RainMaker pozivom funkcije `esp_rmaker_start()`, koji će izvršiti korak potvrđivanja veze, uspostaviti komunikaciju protokolom MQTT, objaviti početne vrijednosti parametara uređaja ispod čvora i dr. Konačno, pozivom funkcije `app_wifi_start()` čvor će se povezati s mrežom i obaviti korak postavljanja Wi-Fi mreže ako ona nije postavljena.

U ostatku glavnog programa inicijalizira se NVS memorijска particija te se omogućuju postavke smanjene potrošnje opisane u prethodnim poglavljima.

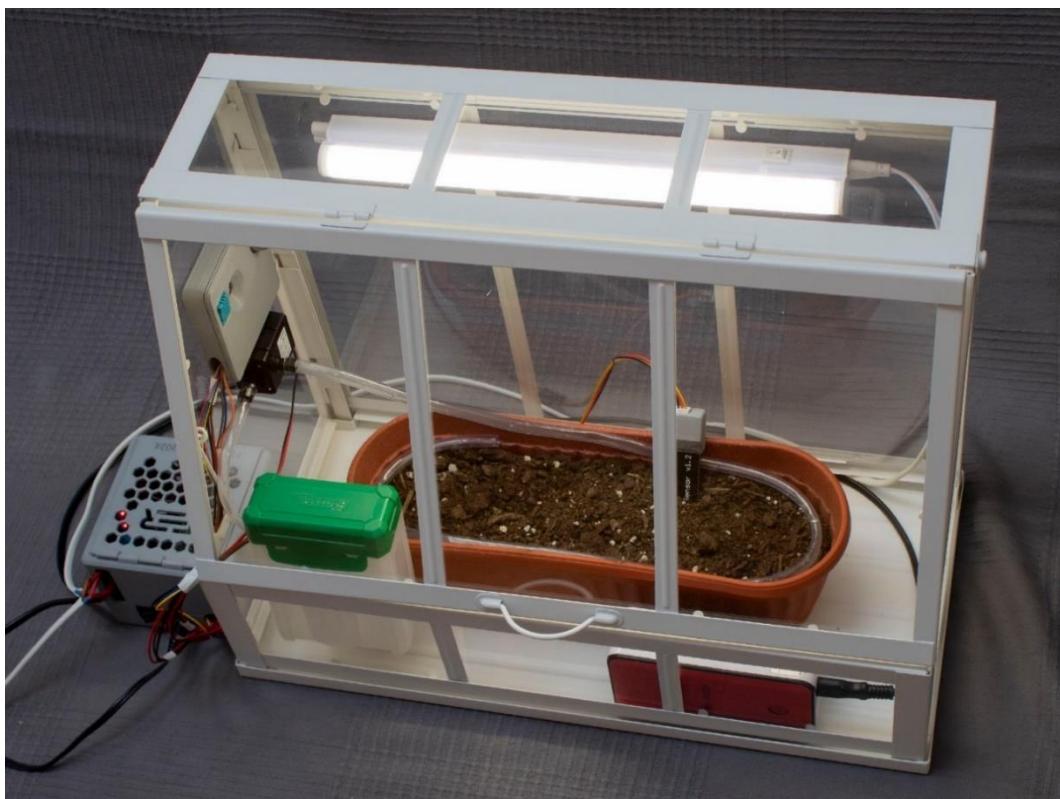
7. Pokazni model sustava za praćenje i regulaciju uvjeta u stakleniku

7.1. Sklopovalski dio sustava

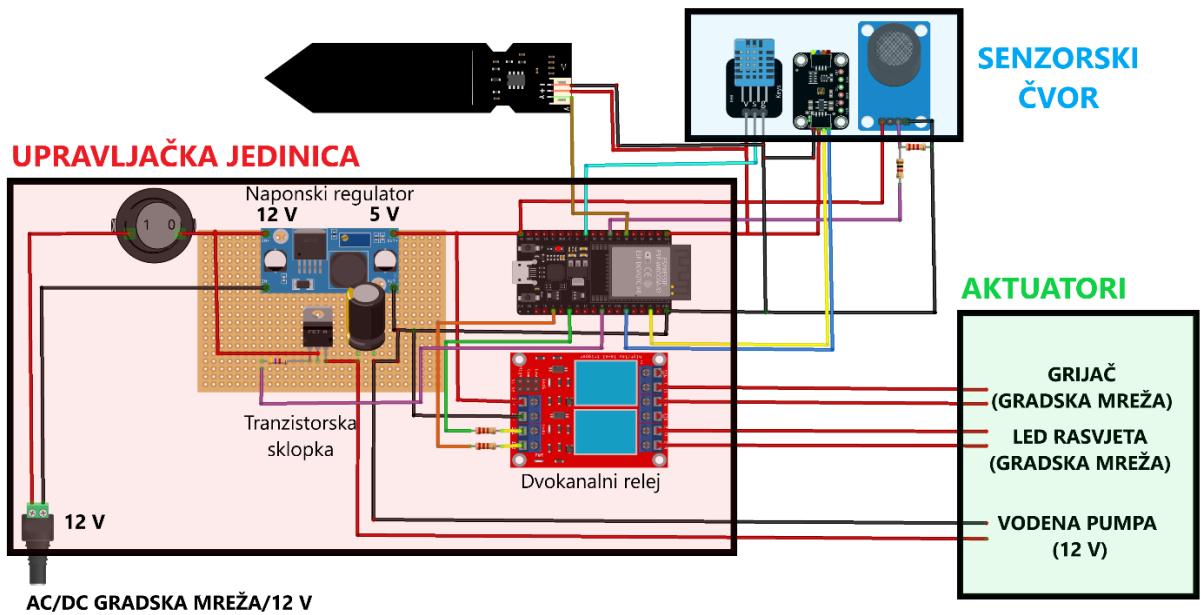
Pokazni se model (**Slika 7.1**) sastoji od tri dijela:

- upravljačke jedinice,
- senzorskog čvora,
- povezanih aktuatora.

Svi su dijelovi pričvršćeni na staklenik manjih dimenzija namijenjen za uzgoj mladica i biljaka u unutarnjem ili vanjskom prostoru. U staklenik je postavljen i spremnik vode za navodnjavanje. Dijelovi modela prikazani su slikom (**Slika 7.2**).

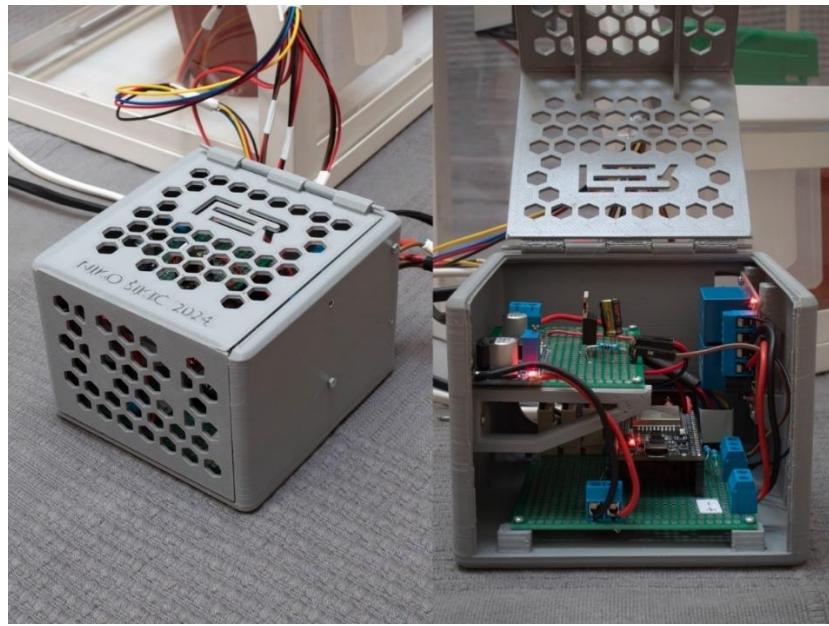


Slika 7.1 Pokazni model sustava za praćenje i regulaciju uvjeta u stakleniku.



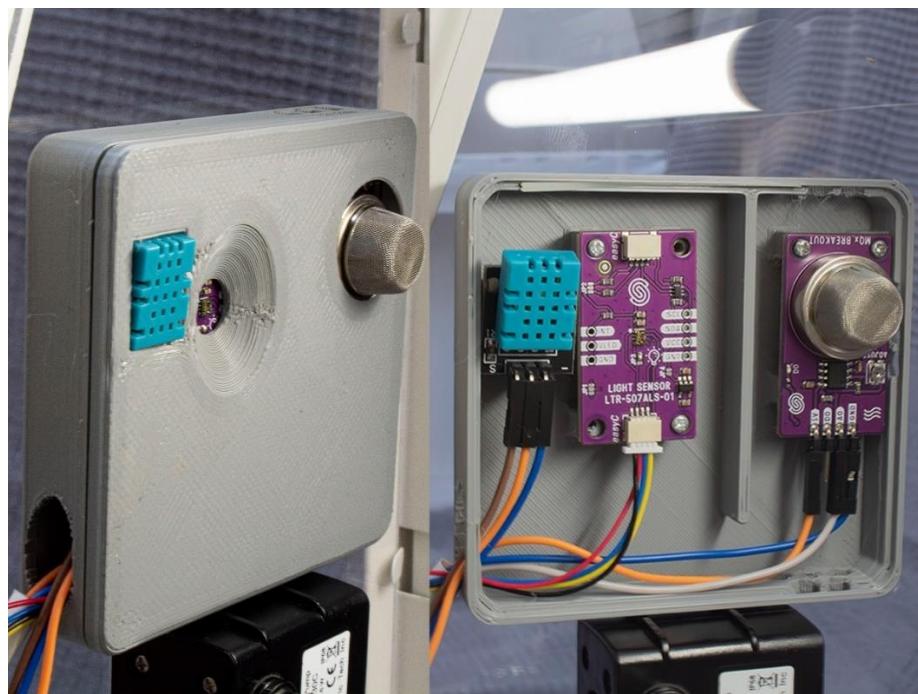
Slika 7.2 Pojednostavljena shema sustava izrađena u programu Fritzing.

Upravljačku jedinicu priказанu slikom (Slika 7.3) čine mikrokontroler, pločica s napajanjem te pločica s dvokanalnim relejom. Cijeli se sustav napaja preko AC/DC 100-230 V/12 V mrežnog adaptera. Izlaznih se 12 V regulira na 5 V propusnim regulatorom s prekidanjem struje (engl. *buck, foward, step-down*) kako bi se 5 V dovelo na istoimeni izvod mikrokontrolera i na stezaljke za napajanje releja. Na pločici s mikrokontrolerom nalaze se konektori za jednostavno spajanje senzora i aktuatora. Relejnom se pločicom upravlja LED rasvjetom i grijачem, koji se napajaju iz gradske mreže. Vodena se pumpa napaja direktno s 12 V, a upravlja se jednostavnom tranzistorskog sklopkom na pločici s napajanjem. Ona se, kao i releji, preklapa signalom s GPIO izvoda mikrokontrolera. Na kućištu upravljačke jedinice nalazi se sklopka za uključivanje i isključivanje napajanja.



Slika 7.3 Upravljačka jedinica pokaznog modela s pločicama za napajanje, mikrokontroler i releje.

U senzorskom su čvoru postavljeni senzori DHT11, LTR-507-ALS-01 i MQ-135. Senzori su postavljeni u kućište, pri čemu je senzor MQ-135 nešto udaljen od ostalih senzora zbog zagrijavanja. Kapacitivni je senzor vlažnosti tla odvojen od čvora i zaštićen posebnim kućištem. Sva su kućišta dizajnirana u besplatnom programu Blender. Senzorski je čvor prikazan slikom (Slika 7.4).

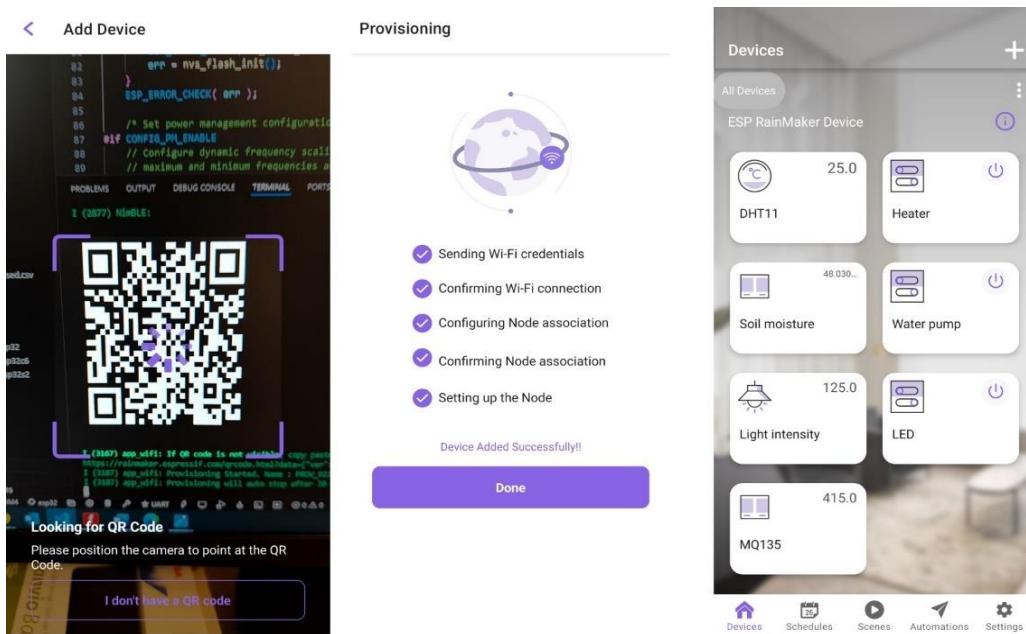


Slika 7.4 Senzorski čvor pokaznog modela sa senzorima DHT11, LTR-507-ALS i MQ-135.

7.2. Programska potpora

Programska podrška s uputama za korištenje dostupna je na poveznici https://github.com/Nixon275/Zavrsni_projekt.

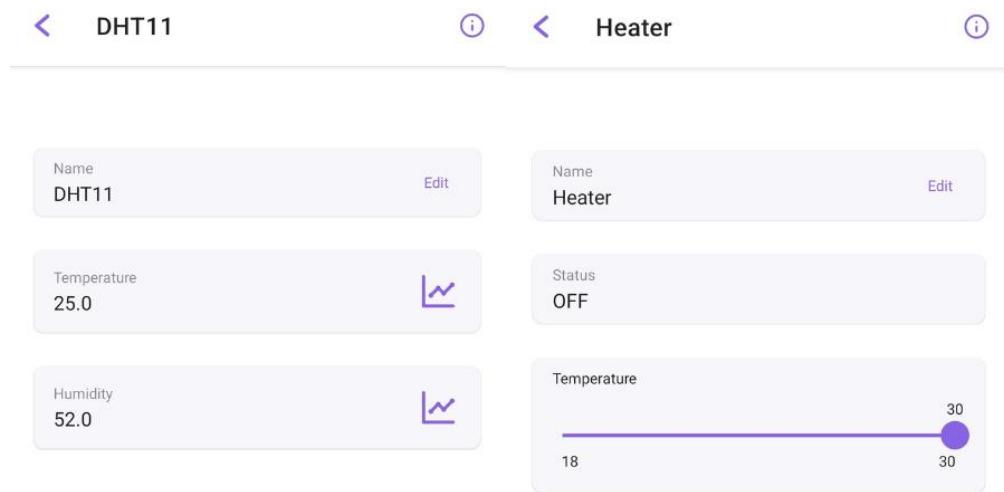
Nakon programiranja mikrokontrolera potrebno je pokrenuti aplikaciju ESP RainMaker na mobilnom uređaju, prijaviti se na korisnički račun i započeti povezivanje. Mobilni uređaj u koraku postavljanja Wi-Fi mreže mora biti povezan na mrežu na koju se želi spojiti mikrokontroler. Bluetooth također mora biti uključen zbog potpomognutog potvrđivanja veze. Najjednostavniji način povezivanja jest skeniranjem QR koda ispisanih u terminalu nakon programiranja mikrokontrolera. Aplikacija će na temelju tog koda, koji je jedinstven za svaki mikrokontroler, proslijediti identifikator čvora oblaku i obaviti povezivanje. Nakon uspješnog povezivanja čvor s povezanim uređajima postaje vidljiv u aplikaciji i nadzornoj ploči platforme ESP RainMaker (engl. *ESP RainMaker Dashboard*). Postupak povezivanja i izgled čvora s uređajima u aplikaciji prikazani su slikom (Slika 7.5).



Slika 7.5 Postupak povezivanja sustava s mrežom i izgled početnog zaslona s uređajima spojenih na čvor u aplikaciji ESP RainMaker.

Na početnom su zaslonu vidljivi svi uređaji spojeni na čvor s istaknutim posljednjim mjernim rezultatom (ako je u pitanju senzor) ili postavljenim stanjem (ako je u pitanju aktuator). Pritiskom na pojedini uređaj otvara se detaljniji prikaz (Slika 7.6), u kojem je moguće iscrtati mjerne rezultate pritiskom na ikonu pored promatranog parametra. Na

zaslonu s iscrtanim rezultatima moguće je podesiti vremenski interval praćenja, metodu iscrtavanja, prikaz srednje vrijednosti i dr.



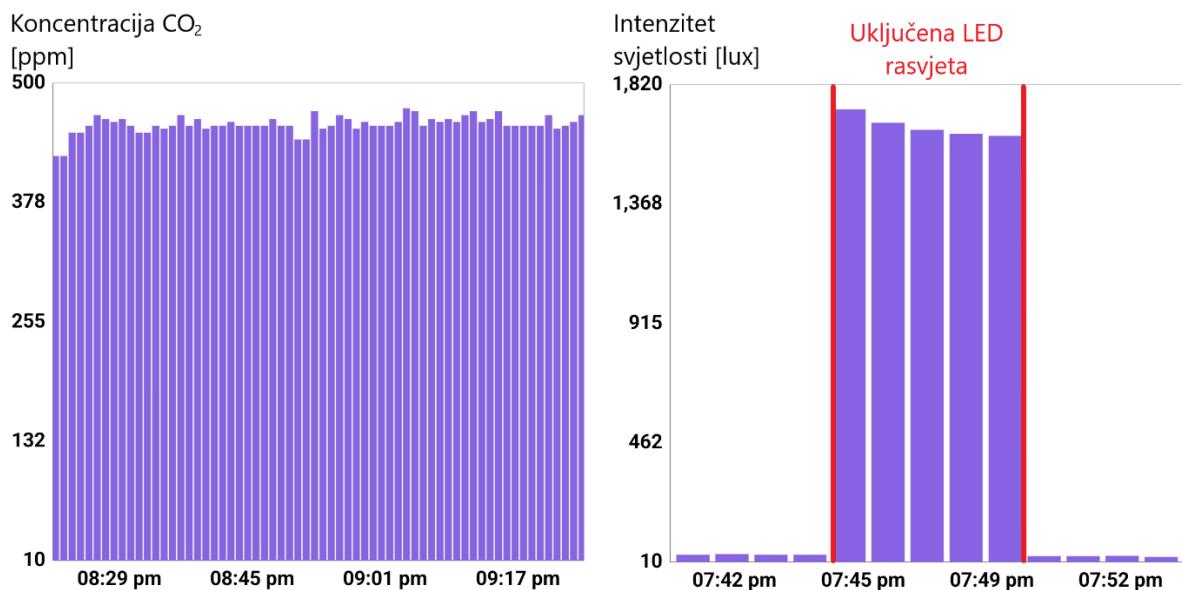
**Slika 7.6 Detaljniji prikaz uređaja za senzor DHT11 (lijevo) i za grijач (desno) u mobilnoj aplikaciji.
Zadana temperatura na grijajuću postavljena je na 30 °C.**

U detaljnijem prikazu uređaja grijajuča i vodene pumpe moguće je podesiti temperaturu odnosno vlažnost tla pomicanjem klizača u definiranom rasponu i time omogućiti automatsku regulaciju. LED rasvjetom upravlja se ručno jednostavnom sklopkom. Također je moguće podesiti raspored uključivanja uređaja odabirom izbornika za raspored (engl. *Schedules*) na početnom zaslonu. Time se može omogućiti paljenje LED rasvjete u određeno doba dana, ovisno o vrsti uzgajane biljke. Na početnom su zaslonu prisutni i drugi izbornici, poput automatizacije (engl. *Automations*), u kojem se upravlja aktuatorskim uređajima ovisno o mjernim vrijednostima. Kako je automatizacija izvedena neovisno o aplikaciji, ta se opcija nije razmatrala.

7.3. Rezultati

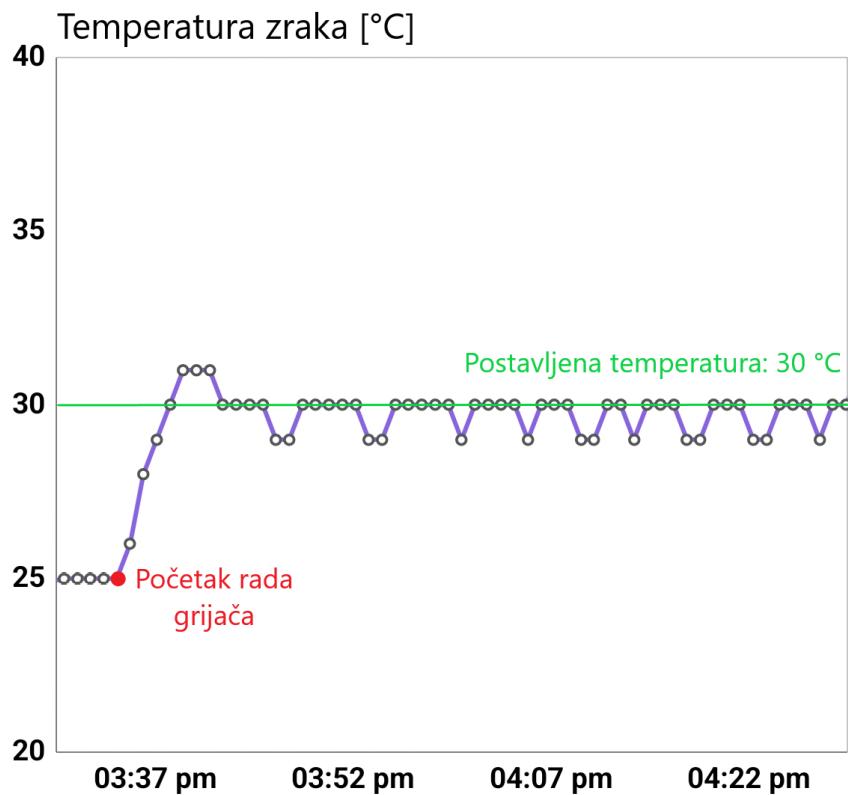
Sustav je tijekom višesatnog rada pokazao dostatnu robusnost i nije se pretjerano zagrijavao. Senzori su dostavljali mjerne rezultate na oblak platforme ESP RainMaker uz minimalno pogrešaka pri slanju. Najveće vrijednosne oscilacije uočene su kod mjernih rezultata senzora koncentracije ugljikova dioksida u zraku MQ-135, a njihov je uzrok potrebno vrijeme za zagrijavanje senzora. Odziv svakog aktuatora bio je brz i pravovremen. Na slici (Slika 7.7) prikazani su mjereni rezultati koncentracije ugljikova dioksida u zraku i promjene intenziteta

svjetlosti uslijed preklapanja sklopke uređaja LED rasvjete. Svi su podatci iscrtani u mobilnoj aplikaciji ESP RainMaker.

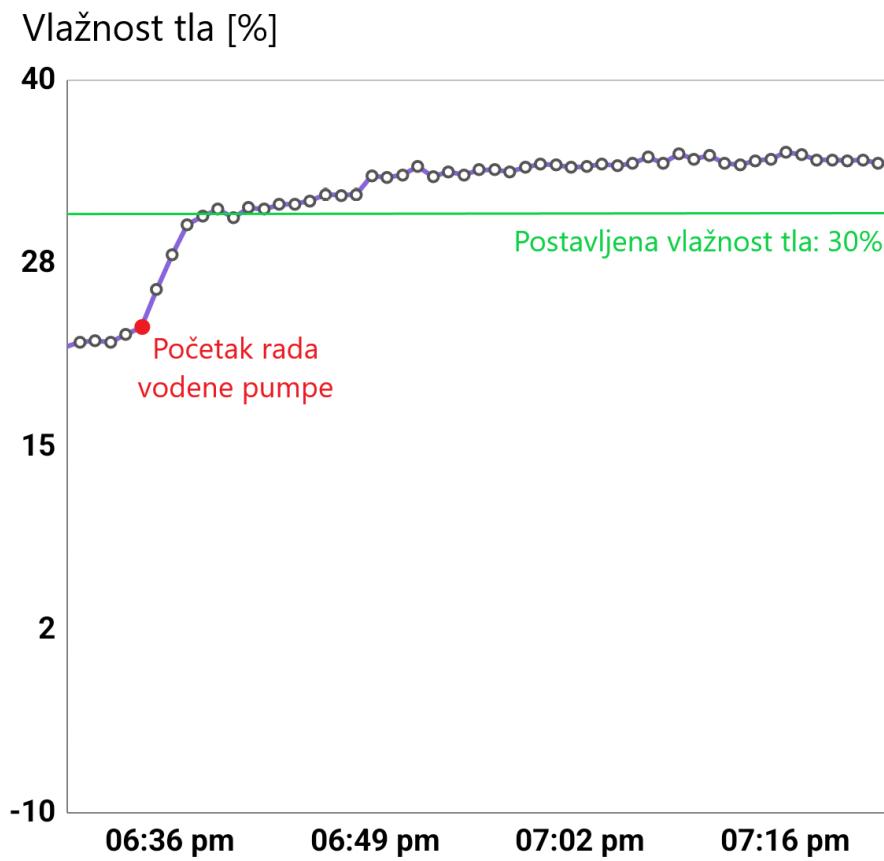


Slika 7.7 Ispitani mjerni rezultati koncentracije ugljikova dioksida u zraku (lijevo) i intenziteta svjetlosti (desno) u mobilnoj aplikaciji ESP RainMaker.

Pri automatskoj regulaciji mikroklimatskih parametara trajanje rada aktuatora prilagođeno je njihovoj učinkovitosti i periodu očitavanja senzora. **Slika 7.8** prikazuje automatsku regulaciju temperature zraka u stakleniku. Period očitavanja senzora postavljen je na 20 sekundi, a trajanje rada grijачa 15 sekundi. Željena temperatura postavljena je na 30 °C, a vrijednosti su promatrane tijekom sat vremena. Grijач je nakon dostizanja postavljene vrijednosti temperaturu održavao s maksimalnom apsolutnom pogreškom od 1 °C. Zabilježeni su mjerni rezultati i za automatsku regulaciju vlažnosti tla tijekom sat vremena. Željena je vlažnost tla postavljena na 30%, period očitavanja na 10 sekundi, a trajanje rada vodene pumpe na 5 sekundi. U iscrtanim je rezultatima prikazanim slikom (**Slika 7.9**) apsolutno odstupanje od zadane vrijednosti manje od 5%. Navedeni su rezultati zadovoljavajući za demonstraciju automatske regulacije tipa *uključi-isključi*, a do detaljnijih bi se zaključaka došlo usporedbom rezultata s mjeranjima točnijeg mjernog instrumenta, što nije provedeno u radu.



Slika 7.8 Isertani rezultati temperaturne regulacije grijačem u aplikaciji ESP RainMaker.



Slika 7.9 Isertani rezultati regulacije vlažnosti tla vodenom pumpom u aplikaciji ESP RainMaker.

8. Zaključak

Sustavi za praćenje i regulaciju mikroklimatskih uvjeta pri uzgoju biljaka u zatvorenom prostoru postaju sve razvijeniji. Moderni staklenički sustav tipično sadrži senzore i aktuatorne za upravljanje mikroklimatskim parametrima poput temperature i vlažnosti zraka, koncentracije ugljikova dioksida u zraku, vlažnosti tla i intenziteta svjetlosti. U radu je demonstrirano da se navedeni parametri mogu upravljati i jeftinim komercijalno dostupnim senzorima i aktuatorima spojenim na mikrokontroler ESP32. Za komunikaciju s digitalnim senzorima razmatrani su i objašnjeni protokoli jednožične (*I-wire*) i dvožične (I2C) komunikacije, a istraženi su i provedeni postupci kalibracije analognih senzora. Aktuatori su regulirani regulatorom tipa *uključi-isključi*, a bolje bi se upravljanje postiglo primjenom PID regulatora. Praćenje i regulacija spomenutih parametara korištenjem odabralih senzora i aktuatora postignuto je izradom pokaznog modela.

Pokazni je model cijelokupno sklopovsko i programsko rješenje koje korisniku omogućuje prikaz i kontrolu nad mikroklimatskim parametrima u stakleniku manjih dimenzija. Razvoj programske podrške znatno je olakšan korištenjem platforme ESP RainMaker, kojom se olakšava bežična komunikacija i prikaz mjernih rezultata. U programskoj su podršci također implementirane postavke smanjene potrošnje, kojima se poboljšava autonomnost ugradbenog sustava.

Izradom i objavom ovakvih i sličnih rješenja naglašava se primjenjivost sve dostupnijih tehnologija u sklopu mreže povezanih uređaja interneta stvari. Takva rješenja postaju sve češća u modernom stakleničkom uzgoju, a njihov je razvoj važan za optimalan uzgoj biljaka u poljoprivredne ili istraživačke svrhe. Njima se olakšava proces proizvodnje i osigurava kvalitetniji proizvod, što je od izrazite važnosti u današnjem društvu, u kojem resursi postaju sve oskudniji, a potražnja za njima veća.

Literatura

- [1] M. A. Abas and M. Dahlui, *Development of greenhouse autonomous control system for Home Agriculture project*, 2015 International Conference on Advanced Mechatronics, Intelligent Manufacture, and Industrial Automation (ICAMIMIA), Surabaya, Indonesia (2015), str. 12-17
- [2] Nemali, Krishna, *History of Controlled Environment Horticulture: Greenhouses*. HortScience 57.2 (2022), str. 239-246.
- [3] Bhujel, A., Basak, J.K., Khan, F. et al. Sensor Systems for Greenhouse Microclimate Monitoring and Control: a Review. *J. Biosyst. Eng.* 45 (2020), str. 341–361
- [4] Tunio, Mazhar & Gao, Jianmin. *Technological Modernization and Its Influence on Agriculture Sustainability, Aeroponics Systems in Belt and Road Countries*. (2021).
- [5] Institut Kulinarstva New York. *Hydroponic Garden*. Poveznica: <https://www.ice.edu/campuses/newyork/facilities-technology/hydroponic-garden>; pristupljeno 20. lipnja 2024.
- [6] Weiss Technik. *Fitotron Modular Plant Growth Chambers Type HGC*. Poveznica: <https://www.weiss-technik.com/en/products/detail/fitotron-modular-plant-growth-chambers-type-hgc~p27622>; pristupljeno 20. lipnja 2024.
- [7] Bhujel, A., Basak, J.K., Khan, F. et al. *Sensor Systems for Greenhouse Microclimate Monitoring and Control: a Review* . *J. Biosyst. Eng.* 45, (2020.), str. 341–361.
- [8] V. M. Quan, G. Sen Gupta and S. Mukhopadhyay, *Review of sensors for greenhouse climate monitoring*, 2011 IEEE Sensors Applications Symposium, San Antonio, TX, USA, (2011), str. 112-118.
- [9] Espressif Systems. (2021). *ESP32 Series Datasheet*. Poveznica: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf; pristupljeno 11. svibnja 2024.
- [10] Espressif Systems. (2024). *Getting Started with ESP-IDF*. Poveznica: <https://docs.espressif.com/projects/esp-idf/en/stable/esp32/get-started/index.html>; pristupljeno 11. svibnja 2024.
- [11] A. Maier, A. Sharp and Y. Vagapov, *Comparative analysis and practical implementation of the ESP32 microcontroller module for the internet of things*, 2017 Internet Technologies and Applications (ITA), Wrexham, UK, (2017.), str. 143-148.
- [12] Espressif Systems. (2024). *Getting Started with ESP32-DevKitC*. Poveznica: <https://docs.espressif.com/projects/esp-idf/en/stable/esp32/hw-reference/esp32/get-started-devkitc.html>; pristupljeno 11. svibnja 2024.
- [13] Espressif Systems. (2021). *ESP32-WROOM-32E & ESP32-WROOM-32UE Datasheet*. Poveznica: https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e_esp32-wroom-32ue_datasheet_en.pdf; pristupljeno 11. svibnja 2024.
- [14] Espressif Systems. (2024). *Wi-Fi Power Save Example*. Poveznica: https://github.com/espressif/esp-idf/blob/master/examples/wifi/power_save/README.md; pristupljeno 11. Svibnja 2024.

- [15] Components101. (2024). *DHT11 Temperature Sensor*. Poveznica: <https://components101.com/sensors/dht11-temperature-sensor>; pristupljeno 11. svibnja 2024.
- [16] Aosong Electronics. (2020). *DHT11 Technical Data Sheet*. Poveznica: <https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf>; pristupljeno 11. svibnja 2024.
- [17] Espressif Systems. (2024). *ADC API Reference*. Poveznica: <https://docs.espressif.com/projects/esp-idf/en/v4.4.8/esp32/api-reference/peripherals/adc.html>; pristupljeno 20. travnja 2024.
- [18] Espressif Systems. (2024). *ESP32 Peripherals API Reference*. Poveznica: <https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-reference/peripherals/index.html>; pristupljeno 20. travnja 2024.
- [19] M. Saleh, I. H. Elhajj, D. Asmar, I. Bashour and S. Kidess, *Experimental evaluation of low-cost resistive soil moisture sensors*, IEEE International Multidisciplinary Conference on Engineering Technology (IMCET), Beirut, Lebanon, 2016, str. 179-184
- [20] Hrisko, Joshua, *Capacitive Soil Moisture Sensor Theory, Calibration, and Testing* (2020).
- [21] SunFounder, (2024). *Ultimate Sensor Kit: Soil Moisture Sensor*. Poveznica: https://docs.sunfounder.com/projects/ultimate-sensor-kit/en/latest/components_basic/10-component_soil.html; pristupljeno 27. Svibnja 2024.
- [22] Hanwei Electronics. (2020). *MQ-135 Gas Sensor Datasheet*. Poveznica: https://www.electronicoscaldas.com/datasheet/MQ-135_Hanwei.pdf; pristupljeno 27. svibnja 2024.
- [23] Vaisala. (2021). *User Guide for CO2 Sensors*. Poveznica: <https://docs.vaisala.com/v/u/M210514EN-E/en-US>; pristupljeno 27. svibnja 2024.
- [24] CO2.Earth. (2024). *Daily CO2 Levels*. Poveznica: <https://www.co2.earth/daily-co2>; pristupljeno 27. svibnja 2024.
- [25] Gironi, D. (2014). *Cheap CO2 Meter Using MQ135 Sensor with AVR*. Poveznica: <https://davidegironi.blogspot.com/2014/01/cheap-co2-meter-using-mq135-sensor-with.html#.VJXAGsC3b>; pristupljeno 27. svibnja 2024.
- [26] K. B. Kumar Sai, S. R. Subbdareddy, A. K. Luhach, *IOT based Air Quality Monitoring System Using MQ135 and MQ7 with Machine Learning Analysis*, Scalable Computing: Practice and Experience 20, 4 (2019).
- [27] Soldered Electronics. *Air Quality Sensor MQ135 Breakout*. Poveznica: <https://soldered.com/product/air-quality-sensor-mq135-breakout/#gallery-2>; pristupljeno 27. svibnja 2024.
- [28] CiteSeerX. *Document on Sensor Applications*. Poveznica: <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=314537daa1f601f83044b25b68e2af6c8f331f3f>; pristupljeno 27. svibnja 2024.
- [29] Novitasari, I., Widiana, I. W. *Design and Development of a Smart Home System with Real-Time Monitoring and Control*, Makara Journal of Technology, 22,2 (2018), str. 89-96.

- [30] Analog Devices. *I²C Primer: What is I²C (Part 1)*. Poveznica: <https://www.analog.com/en/resources/technical-articles/i2c-primer-what-is-i2c-part-1.html>; pristupljeno 5. srpnja 2024.
- [31] Lite-On. *Datasheet for Lite-On Sensors*. Poveznica: https://hr.mouser.com/datasheet/2/239/lite-on_lite-s-a0000286413-1-1749818.pdf; pristupljeno 27. svibnja 2024.
- [32] Soldered Electronics. *Digital Light Proximity Sensor LTR-507*. Poveznica: <https://soldered.com/product/digital-light-proximity-sensor-ltr-507-breakout/#gallery-2>; pristupljeno 27. svibnja 2024.
- [33] Sveučilište u Zagrebu. *Predavanje na temu zatvorene petlje*. Poveznica: https://www.fer.unizg.hr/_download/repository/RUS_predavanje_17.pdf; pristupljeno 24. svibnja 2024
- [34] Control.com. *On/Off Control Textbook*. Poveznica: <https://control.com/textbook/closed-loop-control/onoff-control/>; pristupljeno 24. svibnja 2024.
- [35] Parallax. *PID Control Tutorial*. Poveznica: <https://learn.parallax.com/tutorials/language/pbasic/pid-control>; pristupljeno 24. svibnja 2024.
- [36] Espressif Systems. *ESP-RainMaker Documentation*. Poveznica: <https://docs.espressif.com/projects/esp-rainmaker/en/latest/esp32/#>; pristupljeno 27. lipnja 2024.
- [37] Espressif Systems. *ESP-RainMaker Firmware Development Guide*. Poveznica: <https://rainmaker.espressif.com/docs/firmware-dev.html>; pristupljeno 27. lipnja 2024.
- [38] Espressif Systems. *ESP-RainMaker Node-Cloud Communication*. Poveznica: <https://rainmaker.espressif.com/docs/node-cloud-comm.html>; pristupljeno 27. Lipnja 2024.
- [39] Espressif Systems. *ESP-RainMaker User-Node Mapping*. Poveznica: <https://rainmaker.espressif.com/docs/user-node-mapping.html>; pristupljeno 27. lipnja 2024.
- [40] Espressif Systems. *ESP-RainMaker Claiming Devices*. Poveznica: <https://rainmaker.espressif.com/docs/claiming.html>; pristupljeno 27. lipnja 2024
- [41] Espressif Systems. *Getting Started with ESP-RainMaker*. Poveznica: <https://rainmaker.espressif.com/docs/get-started.html>; pristupljeno 27. lipnja 2024.

Sažetak

Sustav za automatsko upravljanje mikroklimatskim uvjetima zatvorenog prostora za uzgoj biljaka

Kako bi se osigurao optimalan uzgoj biljaka u zatvorenom prostoru moderni staklenici implementiraju sustave za praćenje i automatsku regulaciju mikroklimatskih parametara. U radu su opisani najvažniji mikroklimatski parametri za uzgoj biljaka te najčešći senzori i aktuatori za njihovo upravljanje. Cilj rada bio je izrada pokaznog modela staklenika temeljenog na seriji mikrokontrolera ESP32 i razvojnom okviru ESP-IDF. Korištena je pločica ESP32-DevkitC V4 s integriranim funkcijama za Wi-Fi i BLE komunikaciju, uz modove rada smanjene potrošnje. Odabrani su senzori DHT11, kapacitivni senzor vlažnosti tla, MQ-135 i LTR-507-ALS-01. Za njihovu komunikaciju s mikrokontrolerom objašnjeni su protokoli *1-wire* i I2C, a provedeni su i eksperimentalna mjerena u postupku kalibracije analognih senzora. Razmatrane su metode automatske regulacije aktuatora, pri čemu je odabrana regulacija tipa *uključi-isključi*. Povezivanje sustava preko bežične mreže i pohrana podataka na oblak postignuti su korištenjem platforme ESP RainMaker. Pokazni model cjelovito je sklopovsko i programsko rješenje za automatsku regulaciju parametara u stvarnom vremenu. Takvim i sličnim IoT rješenjima povećava se produktivnost i smanjuje potreba za ljudskim radom u poljoprivrednoj proizvodnji.

Ključne riječi: staklenik, automatska regulacija, IoT, ESP32, ESP RainMaker, DHT11, MQ-135, senzor vlažnosti tla, LTR-507-ALS-01, *1-wire*, I2C

Summary

A system for automatic control of the microclimatic conditions in rooms for growing plants

To ensure optimal plant growth in enclosed spaces, modern greenhouses implement systems for monitoring and automatic regulation of microclimatic parameters. This project describes the most important microclimatic parameters for plant cultivation and the common sensors and actuators used for their management. The goal was to create a demonstration model of a greenhouse based on the ESP32 microcontroller series and the ESP-IDF development framework. Utilizing the ESP32-DevkitC V4 board, equipped with integrated Wi-Fi and BLE capabilities, the project incorporates power-saving features. The selected sensors include DHT11, a capacitive soil moisture sensor, MQ-135, and LTR-507-ALS-01. The 1-wire and I2C communication protocols with the microcontroller are explained, and experimental measurements were conducted during the calibration process of analog sensors. After evaluating various methods for automatic actuator control, *on-off* regulation was chosen. The system connects wirelessly to a network and stores data in the cloud via the ESP Rainmaker platform. The demonstration model is a comprehensive hardware and software solution for real-time automatic parameter regulation. Such IoT solutions increase productivity and reduce the need for human labor in agricultural production.

Keywords: greenhouse, automatic regulation, IoT, ESP32, ESP RainMaker, DHT11, MQ-135, soil moisture sensor, LTR-507-ALS-01, 1-wire, I2C