

TEHNIČKO VELEUČILIŠTE U ZAGREBU
STRUČNI PRIJEDIPLOMSKI STUDIJ ELEKTROTEHNIKE

Jadran Marinović

**Mobilni uređaj za mjerenje ugljikovog dioksida
u atmosferi**

ZAVRŠNI RAD br. ZR 3286

Zagreb, rujan 2023.

TEHNIČKO VELEUČILIŠTE U ZAGREBU
STRUČNI PRIJEDIPLOMSKI STUDIJ ELEKTROTEHNIKE

Jadran Marinović

JMBAG: 0246090257

**Mobilni uređaj za mjerenje ugljikovog dioksida
u atmosferi**

ZAVRŠNI RAD br. ZR 3286

Zagreb, rujan 2023.

TEHNIČKO VELEUČILIŠTE U ZAGREBU
POVJERENSTVO ZA ZAVRŠNI RAD

Zagreb, 18. srpnja 2023.

Veleučilište - **Tehničko veleučilište u Zagrebu**
odjelno
organizirano:
Predmet: **Signali i procesi**
Grana: **2.03.04 telekomunikacije i informatika**

ZAVRŠNI ZADATAK br. 3286

Pristupnik: **Jadran Marinović (0246090257)**
Studij: **Stručni prijediplomski studij Elektrotehnika**
Smjer: **Komunikacijska i računalna tehnika**

Zadatak: **Mobilni uređaj za mjerenje ugljikovog dioksida u atmosferi**

Opis zadatka:

U okviru završnog rada potrebno je izraditi sklopovsku i programsku podršku uređaja za mjerenje ugljikovog dioksida u atmosferi. Uređaj treba imati mogućnost kontinuiranog mjerenja i pohranu mjernih rezultata u SQL bazu podataka u cilju dostupnosti putem web sučelja i vizualizacije te naknadne statističke obrade rezultata mjerenja.

Sklopovska podrška treba biti temeljena na Arduino platformi uz korištenje odgovarajućeg senzora i komponenti za komunikaciju. Dodatno, uređaj treba imati mogućnost određivanja trenutne lokacije mjerenja (GPS koordinate) i prikaz rezultata mjerenja na LCD zaslonu. Programska podrška treba omogućiti jednostavno korištenje i ostale navedene funkcionalnosti uređaja.

Sklopovsku i programsku podršku potrebno je ispitati s aspekta funkcionalnosti i fleksibilnost rada u svakodnevnoj primjeni.

Zadatak uručen pristupniku: 18. srpnja 2023.
Rok za predaju rada: 22. rujna 2023.

Mentor:

Predsjednik povjerenstva za
završni rad:

dr. sc. Mladen Sokele, v. pred.

Mario Lučan, struč. spec. ing. el.

Sažetak završnog rada

Cilj ovog završnog rada je bio napraviti mobilni mjerni uređaj kojim bi se mjerila koncentracija ugljikovog dioksida na zadanoj pješačkoj ruti. Potreba za takvim uređajem postoji zbog mnogobrojnih posljedica visokih koncentracija ugljikovog dioksida na okoliš i ljudsko zdravlje. Posljedice loše kvalitete zraka na ljudsko zdravlje su najviše prisutne u gusto naseljenim i urbaniziranim sredinama zbog velike količine prometa i industrije. Uređaj automatski tijekom mjerenja šalje izmjerene podatke i lokaciju mjerenja u udaljenu MySQL bazu podataka kako bi mjerenja bila sigurno pohranjena i lako dostupna za naknadnu obradu i analizu. Naknadna obrada mjerenja i vizualizacija rezultata se vrši u Microsoft Excel programu. U ovom radu je razvijena sklopovska i programska podrška za uređaj. Sklopovska podrška se temelji na ESP 32 WROOM mikrupravljaču, a programska na Arduino frameworku. Rad se sastoji od razvoja vlastitog programa za ESP 32 mikroupravljač u Arduino frameworku, konfiguracije MySQL udaljene baze podataka i razvoja vlastite PHP skripte za unos podataka u bazu.

Ključne riječi: mjerenje kvaliteta zraka, koncentracija ugljikovog dioksida, ESP 32 mikroupravljač, MySQL baza podataka, IoT

Sadržaj

1. Uvod.....	1
2. Posljedice ugljikovog dioksida na okoliš i ljudsko zdravlje	2
3. Konceptcija sklopovske podrške uređaja	4
4. Odabir mikroupravljača	5
5. Opis korištenih senzora	7
5.1 CO ₂ senzor.....	7
5.2 Kalibracija CO ₂ senzora	9
5.3 GPS senzor.....	10
5.4 OLED zaslon	11
6. Električna shema uređaja	12
7. Programski kod	13
7.1 Dijagram toka rada.....	13
7.2 Razvojno okruženje PlatformIO	14
7.3 Metodologija pisanja koda	15
7.4 Zanimljivi dijelovi koda	16
7.5 Korištene biblioteke.....	19
7.6 MySQL baza podataka i njena konfiguracija	19
7.7 PHP skripta	21
8. Završni izgled uređaja.....	23
9. Obrada podataka	24
10. Mjerenje potoka u naselju Voltino.....	25
11. Zaključak	29
Literatura.....	30
Summary	32
Prilog.....	33
11.1 Arduino kod	33
11.2 PHP skripta	38

Popis kratica

CO ₂	Ugljikov dioksid
CSV	Format datoteke gdje su vrijednosti razdvojene zarezima (eng. <i>Comma-Separated Values</i>)
GND	Točka zajedničkog potencijala tj. referentna točka (eng. <i>Ground</i>)
GNSS	Globalni navigacijski satelitski sustav (eng. <i>Global Navigation Satellite System</i>)
GPIO	Opći ulazno/izlazni pinovi (eng. <i>General Purpose Input/Output</i>)
GPS	Američki globalni sustav za pozicioniranje (eng. <i>Global Positioning System</i>)
IDE	Integrirano razvojno okruženje (eng. <i>Integrated Development Environment</i>)
IoT	Internet stvari (eng. <i>Internet of Things</i>)
NDIR	Ne-disperzivna infracrvena tehnologija (eng. <i>Non-Dispersive Infrared</i>)
OLED	Organski svjetleći dioda (eng. <i>Organic Light-Emitting Diode</i>)
PWM	Pulsno širinska modulacija (eng. <i>Pulse Width Modulation</i>)
RX	Oznaka za prijamnik (eng. <i>Receiver</i>)
SQL	Programski jezik koji se koristi u bazama podataka (eng. <i>Structured Query Language</i>)
SSID	Naziv Wi-Fi mreže (eng. <i>Service Set Identifier</i>)
TX	Oznaka za predajnik (eng. <i>Transmitter</i>)
UART	Serijski komunikacijski protokol (eng. <i>Universal Asynchronous Receiver/Transmitter</i>)
USB	Univerzalni serijski ulaz (eng. <i>Universal Serial Bus</i>)
Wi-Fi	Tehnologija za bežično povezivanje uređaja (eng. <i>Wireless Fidelity</i>)

Popis tablica

Tablica 1	Nuspojave izloženosti ugljikovom dioksidu [3]	2
Tablica 2	Specifikacije ESP 32 WROOM modula [6].....	6
Tablica 3	Ključne specifikacije SenseAir S8 LP senzora [7].....	8
Tablica 4	Korišteni pinovi	12
Tablica 5	Korištene biblioteke	19

Popis slika

Slika 1	Dijagram koncepcije sklopovske podrške uređaja.....	4
Slika 2	Dijagram pinova na ESPduinu [5].....	5
Slika 3	Senseair S8 senzor.....	7
Slika 4	Prikaz PWM izlaza Senseair S8 na osciloskopu	7
Slika 5	Prikaz komponenata NDIR senzora [9]	9
Slika 6	NEO-6MV2 GPS modul sa antenom	10
Slika 7	0,96 inčni SSD 1306 OLED zaslon.....	11
Slika 8	Električna shema uređaja	12
Slika 9	Dijagram toka rada programa	13
Slika 10	Struktura datoteka u PlatformIO razvojnom okruženju.....	14
Slika 11	Sadržaj platformio.ini datoteke	15
Slika 12	Funkcija za spajanje na WiFi	16
Slika 13	Konfiguracija i upotreba Timera0.....	17
Slika 14	Slanje podataka sa http.post funkcijom u bazu podataka	18
Slika 15	Izgled mySQL tablice	20
Slika 16	PHP skripta	22
Slika 17	Završni izgled uređaja.....	23
Slika 18	Ruta 1. mjerenja.....	25
Slika 19	Izgled tablice sa rezultatima u Microsoft Excelu.....	26
Slika 20	Graf normaliziranih CO ₂ vrijednosti	27
Slika 21	Karta sa CO ₂ normaliziranim vrijednostima	28

1. Uvod

Industrijalizacija u zadnja tri stoljeća donijela je značajan napredak i prosperitet čovječanstvu, ali je za posljedicu imala veliko zagađenje okoliša. Fosilna goriva kao što su nafta, ugljen i zemni plin desetljećima su služila kao glavni izvor energije za industriju, gradove te transport ljudi i dobara. Međutim, njihovo izgaranje rezultira emisijom velikih količina stakleničkih plinova u atmosferu, s ugljičnim dioksidom (CO₂) kao glavnim predstavnikom. U ovom radu ćemo se baviti isključivo emisijama ugljikovog dioksida tj. zagađenjem nastalim izgaranjem fosilnih goriva. Nekontrolirano korištenje fosilnih goriva danas predstavlja veliku opasnost za okoliš i ljudsko zdravlje, pogotovo u gusto naseljenim urbanim sredinama. Zbog tog problema i namjere da ga se kvalitetno kvantificira i zabilježi, postoji potreba za mobilnim uređajem za mjerenje koncentracije ugljikovog dioksida (CO₂) na zadanoj pješačkoj ruti. Uređaj bi trebao imati mogućnosti ispisa podataka sa senzora na OLED zaslon i mogućnost trajne pohrane izmjerenih podataka i lokacije mjerenja u udaljenu SQL bazu podataka. Podatci moraju biti sigurno pohranjeni kako bi se mogla obaviti naknadna statistička obrada, vizualizacija i analiza. Sklopovska podrška se temelji na ESP 32 WROOM mikroupravljač, a programska na *Arduino frameworku*. Podaci se pohranjuju u MySQL bazu podataka na lokalnoj mreži uz pomoć vlastite PHP skripte za unos podataka. Statistička obrada i vizualizacija i mjerenja se naknadno izvršava u programu Microsoft Excel.

2. Posljedice ugljikovog dioksida na okoliš i ljudsko zdravlje

Ugljikov dioksid (CO_2) je bezbojan, bezmirisan, nezapaljiv plin. Nastaje kao posljedica izdisanja zraka živih organizama i izgaranjem fosilnih goriva kao što su nafta, ugljen i zemni plin. CO_2 ima značajnu ulogu u atmosferi i klimatskim promjenama jer je jedan od plinova koji doprinosi efektu staklenika, što može utjecati na globalnu temperaturu i ekološke sustave. Temperatura zemlje ovisi o balansu toplinske energije koja ulazi u atmosferu i energije koja se gubi u svemiru.[1] Zbog toga se CO_2 redovito prati i regulira u mnogim aspektima okolišne i energetske politike kako bi se smanjile negativne posljedice na okoliš i zdravlje ljudi.

U visokim koncentracijama, CO_2 može imati negativne učinke na zdravlje ljudi. To se posebno primjećuje u zatvorenim prostorima s lošom ventilacijom, kao što su sobe bez prozora ili prostori s lošom cirkulacijom zraka. Ugljikov dioksid kod ljudi izaziva štetne posljedice poput uspavanosti i glavobolje što rezultira povećanim krvnim tlakom, ubrzanim srčanim ritmom, asfiksijom (nedostatak kisika) i konvulzijama.[2]

TABLICA 1 NUSPOJAVE IZLOŽENOSTI UGLJIKOVOM DIOKSIDU [3]

Koncentracija u zraku [ppm]	Nuspojave
400	Prosječna vrijednost na otvorenom
400 – 1000	Vrijednosti koje su uobičajene za dobro ventilirane unutarnje prostore
1000 – 2000	Vrijednosti na koje se povezuju sa usporenosti i lošom kvalitetom zraka
2000 – 5000	Dolazi do glavobolja, umora, gubitka koncentracije i ubrzanog ritma srca
5000	Dolazi do manjka kisika i nije preporučen duži boravak u takvim prostorima
40000	Toksično okruženje i po život opasno okruženje zbog ozbiljnog manjka kisika

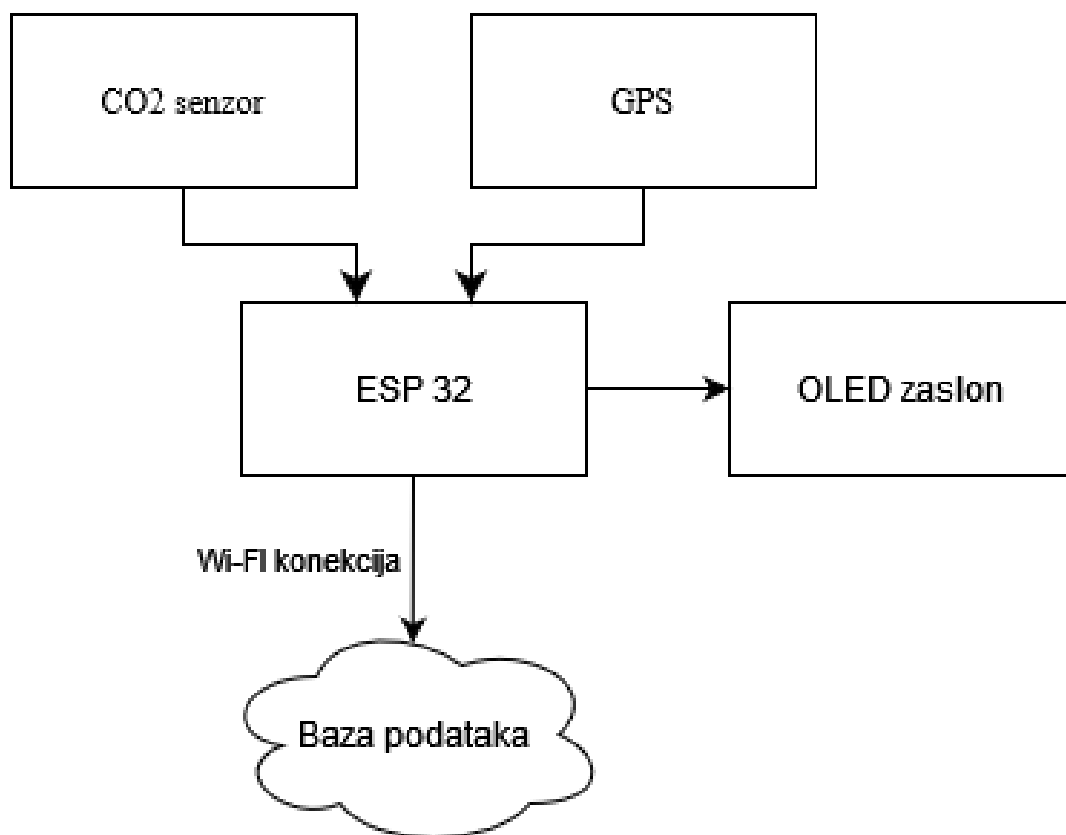
Nakon što je Europska unija i njezine članice poduzele odgovarajuće akcije i donijele politike, kvaliteta zraka znatno se poboljšala u usporedbi s sedamdesetim godinama prošlog stoljeća. Emisije štetnih plinova u atmosferu iz različitih ključnih izvora, uključujući promet, industriju i proizvodnju električne energije, sada su ograničene

zakonodavstvom i općenito su u padu, iako ne u očekivanom obimu. Ugljikov dioksid je najveći uzročnik globalnog zagrijavanja i klimatskih promjena. Klimatske promjene uključuju porast prosječne temperature Zemlje, topljenje ledenjaka i ledenih pokrivača, porast razine mora, češće i ozbiljnije ekstremne vremenske uvjete i promjene u klimatskim obrascima.

Ugljični otisak je mjera količine stakleničkih plinova, posebno ugljikovog dioksida (CO₂), koje pojedinac, organizacija, događaj ili proizvod stvaraju kao posljedicu svojih aktivnosti ili potrošnje. Ovaj koncept koristi se kako bi se procijenio ukupni doprinos emisijama stakleničkih plinova koji pridonose globalnom zagrijavanju i klimatskim promjenama.[4]

3. Konceptcija sklopovske podrške uređaja

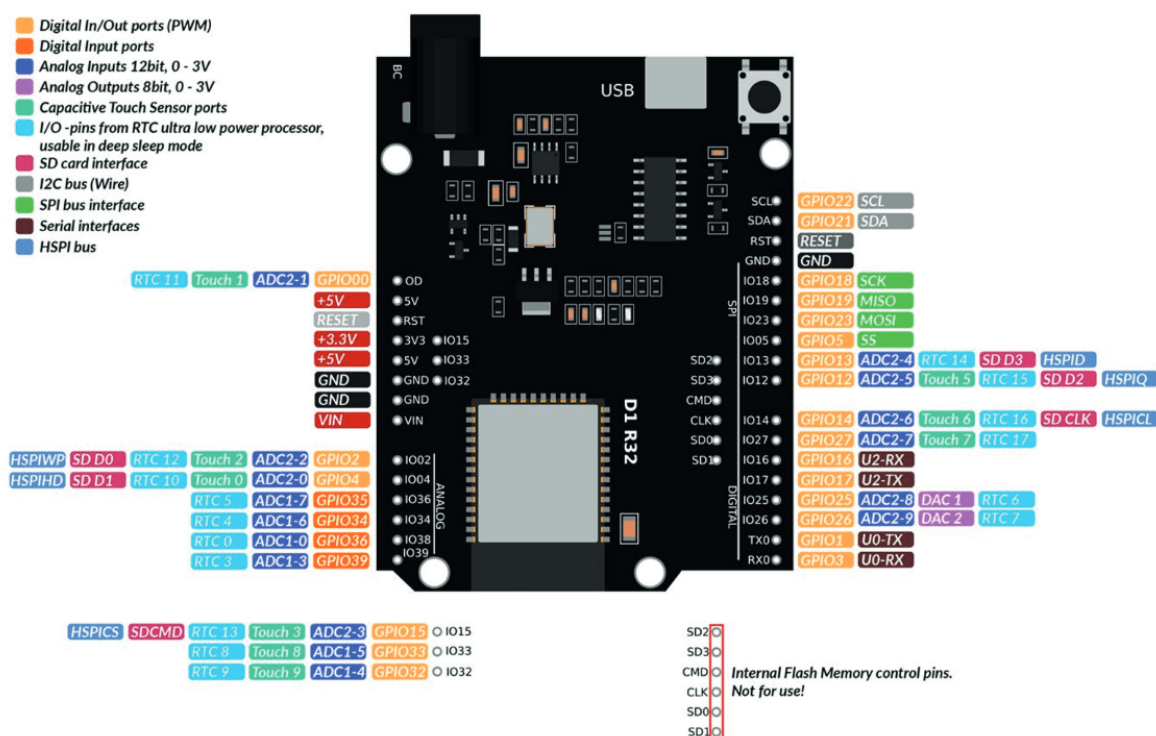
Sklopovska podrška uređaja sastoji se od mikroupravljača ESP 32 WROOM, CO₂ senzora SenseAir S8 LP, NEO-6MV2 GPS modula i OLED zaslona. Senzori komuniciraju s mikroupravljačem pomoću serijske UART komunikacije, a OLED zaslon sa SPI-om. ESP 32 vrši obradu podataka i šalje ih u udaljenu bazu podataka preko Wi-Fi mreže. Napajanje uređaja se vrši pomoću vanjske USB baterije koja je spojena na ESP 32 s micro USB kabelom.



SLIKA 1 DIJAGRAM KONCEPCIJE SKLOPOVSKE PODRŠKE UREĐAJA

4. Odabir mikroupravljača

Za ovaj projekt odabran je mikroupravljač ESP 32 proizvođača Expressif Systems u varijanti oblika Arduino UNO pločice takozvanog ESPduina. Odabrana je ta varijanta kako bi se mogli koristiti standardni Arduino *shieldovi*. Za programiranje se može koristiti standardni Arduino IDE, ali i PlatformIO. U ovom završnom radu korišten je PlatformIO zbog jednostavnijeg programiranja i fleksibilnosti.



TABLICA 2 SPECIFIKACIJE ESP 32 WROOM MODULA [6]

Kategorija	Parametar	Opis
Hardware	Sučelja	SD kartica, UART, SPI, SDIO, I2C, LED PWM, Motor PWM, I2S, IR, brojač impulsa, GPIO, kapacitivni senzor dodira, ADC, DAC, Two-Wire Automotive Interface (TWAI®), kompatibilan sa ISO11898-1 (CAN Specification 2.0)
	Integrirani kristal	40 Mhz
	Integrirani SPI flash	4 MB
	Napon napajanja	3V – 3,6V
	Potrošnja	80 mA
	Minimalna struja napajanja	500 mA
	Radna temperatura	-40 - 85°C
	Veličina čipa	18 x 25,5 x 3,1 mm
	Stupan osjetljivosti na vlagu	Level 3
Wi-Fi	Protokol	802.11 b/g/n (802.11n do 150 Mbps)
	Raspon središnje frekvencije radnog kanala	2412 ~ 2484 MHz

ESP 32 dolazi s ugrađenom Wi-Fi podrškom koja omogućava jednostavno povezivanje s bežičnim mrežama. ESP 32 ima ugrađen klasičan Bluetooth v4.2 kao i BLE (Bluetooth Low Energy). Wi-Fi koristi 2,4 GHz frekvencijsko područje te koristi 802.11 b/g/n standard koji podržava brzinu prijenosa do 150 Mbps. Također treba napomenuti da Wi-Fi i Bluetooth koriste istu antenu za komunikaciju.[6]

5. Opis korištenih senzora

5.1 CO₂ senzor

Odabrani senzor za ovaj projekt je Senseair S8 LP 0053. To je precizni relativno mali senzor namijenjen kućnoj i *hobby* uporabi. Senseair S8 ima opseg pouzdanijeg djelovanja od 400 – 2000 ppm-a (parts per milion) a prošireni opseg od 0 – 10000 ppm-a. Senzor komunicira s mikrokontrolerom preko UART sučelja sa brzinom od 9600 bita po sekundi u formatu SERIAL_8N1.



SLIKA 3 SENSEAIR S8 SENZOR

Senzor također posjeduje i PWM (pulse width modulation) izlaz koji radi na frekvenciji od 1 kHz preko kojega se mjerenjem duljine vremena vođenja može dobiti izlazna vrijednost od 0 – 2000 ppm-a.[7]



SLIKA 4 PRIKAZ PWM IZLAZA SENSEAIR S8 NA OSCOLOSKOPU

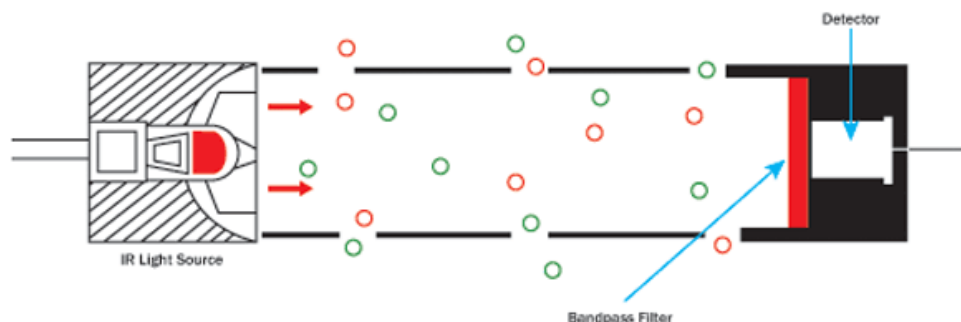
TABLICA 3 KLJUČNE SPECIFIKACIJE SENSEAIR S8 LP SENZORA [7]

Parametar	Opis
Mjereni plin	CO ₂
Princip rada	Non-dispersive infrared (NDIR)
Područje mjerenja	400 – 2000 ppm-a, 0 – 10000 ppm-a u proširenom području mjerenja
Interval mjerenja	4 sekunde
Preciznost	±40 ppm-a
Vrijeme odziva	2 minute
Radna temperatura	0 – 50°C
Radna vlažnost zraka	0 – 85%
Temperatura pohrane	-40 - 70°C
Dimenzije	33,9 x 19,8 x 8,7 mm
Težina	<8 g
Napon napajanja	4,5 – 5,25 V
Potrošnja	300 mA vršno, <18 mA prosječno
Vijek trajanja	15+ godina
Seriska komunikacija	UART, Modbus, RS485
PWM izlaz	1kHz, 0 – 100% duty cycle 0 – 2000ppm-a
Održavanje	Bez potrebnog održavanja

Senzor koristi NDIR (Non-Dispersive Infrared) tehnologiju mjerenja. NDIR senzor sastoji se od infracrvenog izvora, detektora, optičkog filtera, plinske ćelije i elektronike za obradu signala.[5] No najvažnija komponenta unutar NDIR senzora je optički filter koji selektivno propušta samo određenu valnu duljinu. Na primjer, NDIR senzori za mjerenje koncentracije ugljikovog dioksida koriste optički filter koji propušta valnu duljinu od 4,67 μm .

Prednosti NDIR tehnologije nad tradicionalnim elektrokemijskim senzorima su iznimno značajne. Budući da se temelji na optičkom mjerenju, NDIR tehnologija omogućava duži životni vijek senzora i smanjuje troškove održavanja. NDIR senzori su precizni, stabilni i pouzdani u mjerenju koncentracije različitih plinova. Ova tehnologija često se koristi za mjerenje koncentracije plinova poput ugljikovog dioksida (CO₂) i metana

(CH₄) u raznim aplikacijama, uključujući mjerenje kvalitete zraka, industrijske procese i sigurnosne sustave za detekciju plinova. [8]



SLIKA 5 PRIKAZ KOMPONENATA NDIR SENZORA [9]

NDIR senzori plina detektiraju pad intenziteta infracrvenog signala koji je proporcionalan koncentraciji plina.

5.2 Kalibracija CO₂ senzora

Senzor SenseAir S8 LP tijekom svog radnog vijeka zahtjeva kalibraciju kako bi se osigurao pouzdan rezultat mjerenja. SenseAir S8 LP posjeduje ABC (automatic baseline calibration) koja automatski podešava vrijednost temeljenu na minimalnoj izmjerenoj vrijednosti u zadnjih 7 do 14 dana. Novi senzori ponekad daju potpuno krive vrijednosti CO₂ i tada je potrebna manualna kalibracija

Manualna kalibracija SenseAir S8 senzora može se izvršiti na dva načina, softverski ili hardverski. Softwareski način se obavlja slanjem posebne komande na UART ulaz senzora s mikroupravljačem. Hardwareski način se izvršava tako da se bCAL_in pin senzora kratko spoji s točkom zajedničkog potencijala (GND) na 4 do 8 sekundi za kalibraciju na 400 ppm-a ili na preko 8 sekundi na kalibraciju od 0 ppm-a.[7]

Tokom razvoja uređaja senzor je izgubio točnost i morala se provesti kalibracija u programu Microsoft Excel.

5.3 GPS senzor

GPS (Global Positioning System) ili bolje rečeno GNSS (Global Navigation Satellite Systems) je sustav za precizno određivanje lokacije na zemlji pomoću satelita u orbiti.

U ovom završnom radu korišten je senzor NEO-6MV2 švicarskog proizvođača u-blox. GPS senzor je ključna komponenta u ovom radu zbog pružanja precizne lokacije obavljenog mjerenja. NEO-6MV2 GPS modul pri određivanju lokacije koristi isključivo američki GPS sustav za razliku od novijih modela koje imaju mogućnosti spajanja i na ruski GLONASS sustav radi dodatnog poboljšanja preciznosti. Sam NEO-6MV2 GPS modul je zalemljen na pločicu koja sadrži popratne komponente poput stabilizatora napona, filtera i drugih komponenata za omogućavanje pouzdanog rada GPS senzora. Također sadrži pinove za napajanje i output pinove RX (receive) i TX (transmit). Podatci o lokaciji sa senzora se šalju na odabrani mikrokontroler tj. ESP 32 pomoću UART sabirnice.

UART (Universal Asynchronous Receiver/Transmitter) je široko korišten standard dvosmjerne serijske komunikacije koja omogućava pouzdan prijenos podataka između dva uređaja. U ovom slučaju UART sabirnica koristi SERIAL_8N1 format komunikacije s mikrokontrolerom tj. 8 podatkovnih bitova bez dodatnog paritetnog bita i jedan stop bit. UART komunikacija je relativno sporih 9600 bita/s i ona označava brzinu koliko podataka se može prenijeti u sekundi. Brzina od 9600 bita/s je uobičajena brzina komunikacije za senzore ovog tipa i također ju koristi SenseAir S8 senzor. [10]



SLIKA 6 NEO-6MV2 GPS MODUL SA ANTENOM

NEO-6MV2 koristi vanjsku aktivnu keramičku patch antenu dimenzija 25 x 25 x 6.5 mm. Antena radi na frekvenciji od 1561 do 1602 MHz i ima RHCP (right handed circular polarization) polarizaciju. Vanjska antena je spojena s GPS modulom s IPX

konektorom. Antena ima niskošumno pojačalo tj. LNA (low noise amplifier) koje daje pojačanje od 28 dB. [11]

5.4 OLED zaslon

U ovom radu korišten je SD1306 0,96 inčni I2C OLED zaslon. Zaslon je monokromatski, dimenzija 128 × 64 pixela. OLED zasloni zahtijevaju pozadinsko osvjetljenje i zato imaju malu potrošnju te imaju jako dobar kontrast što ih čini dobrim za svijetle uvjete. Zaslon koristi I2C komunikacijski protokol. I2C (Inter-Integrated Circuit) je hardverski serijski protokol i koristi dva pina SCL i SDA. Pin SCL (Serial Clock) prenosi serijski radni takt, a SDA (Serial Data) prenosi podatke dsa mikroupravljača. SD1306 OLED zaslon za napajanje koristi 5 V

Za komunikaciju s mikroupravljačem se najčešće koriste Adafruit_GFX.h i Adafruit_SSD1306.h biblioteke. Ion za napajanje koristi 5 V

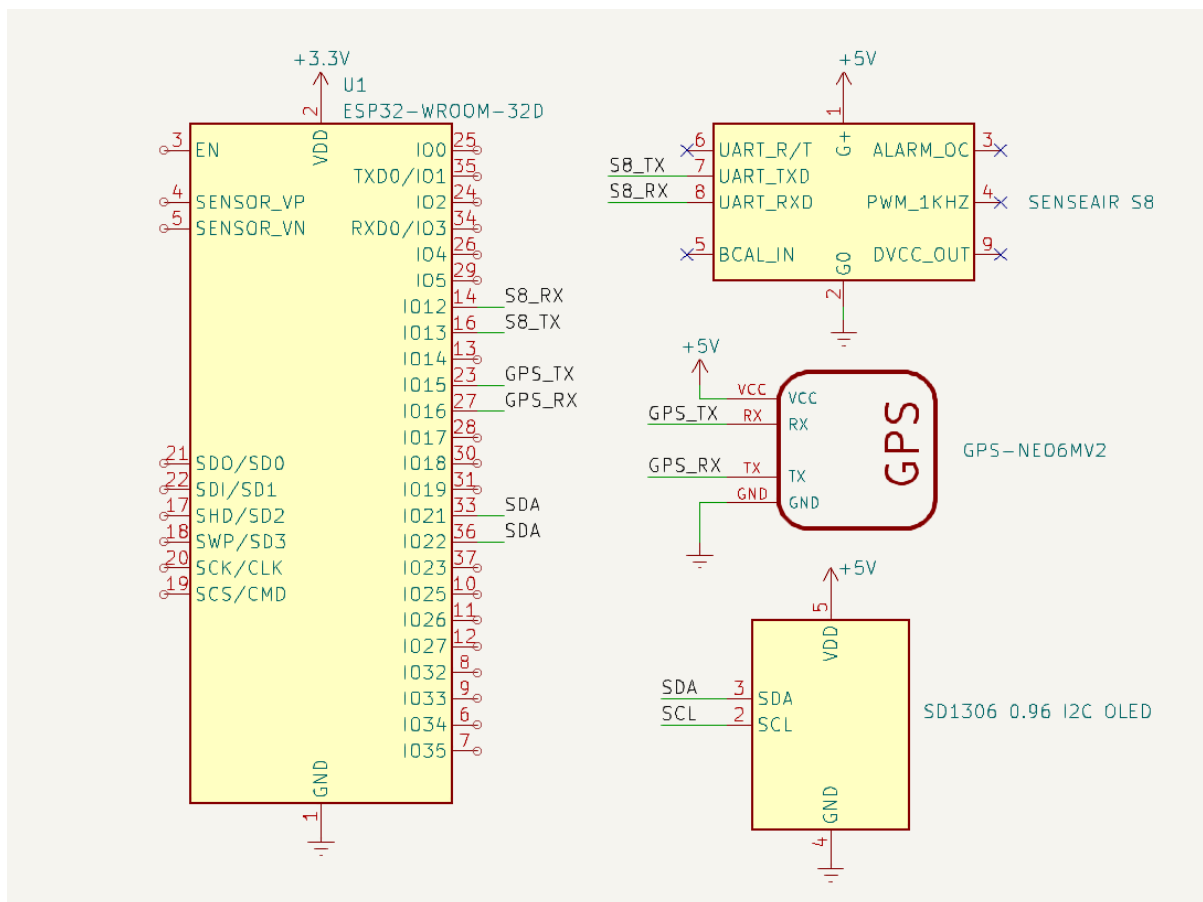
U ovom radu se koristi OLED zaslon za prezentaciju rezultata mjerenja i dijagnostiku uređaja. Dijagnostika podrazumijeva ispis problema s konekcijom s mrežom i s udaljenom bazom podataka.



SLIKA 7 0,96 INČNI SSD 1306 OLED ZASLON

6. Električna shema uređaja

Električna shema ovog uređaja se sastoji od konekcije izabranog mikroupravljača tj. ESP 32 sa izabranim senzorima. Senzori SenseAir S8 i NEO-6MV2 su spojeni pomoću serijske UART komunikacije dok je OLED zaslon spojen sa I2C serijskim protokolom.



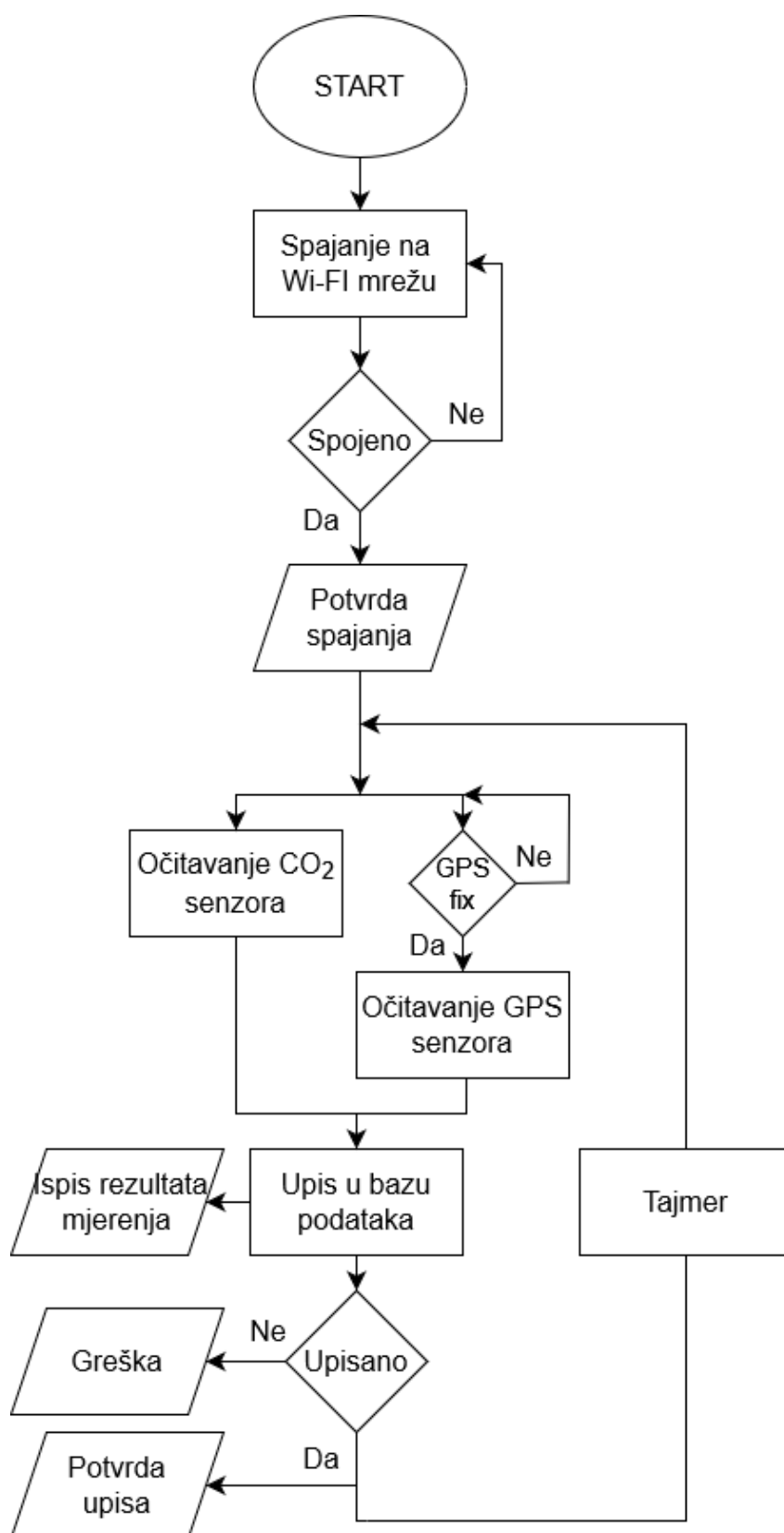
SLIKA 8 ELEKTRIČNA SHEMA UREĐAJA

TABLICA 4 KORIŠTENI PINOVI

PIN	Funkcija
PIN 2	Interna LED dioda
PIN 12	S8 TX
PIN 13	S8 RX
PIN 16	GPS RX
PIN 17	GPS TX
PIN 21	SDA za OLED
PIN 22	SCL za OLED

7. Programski kod

7.1 Dijagram toka rada



SLIKA 9

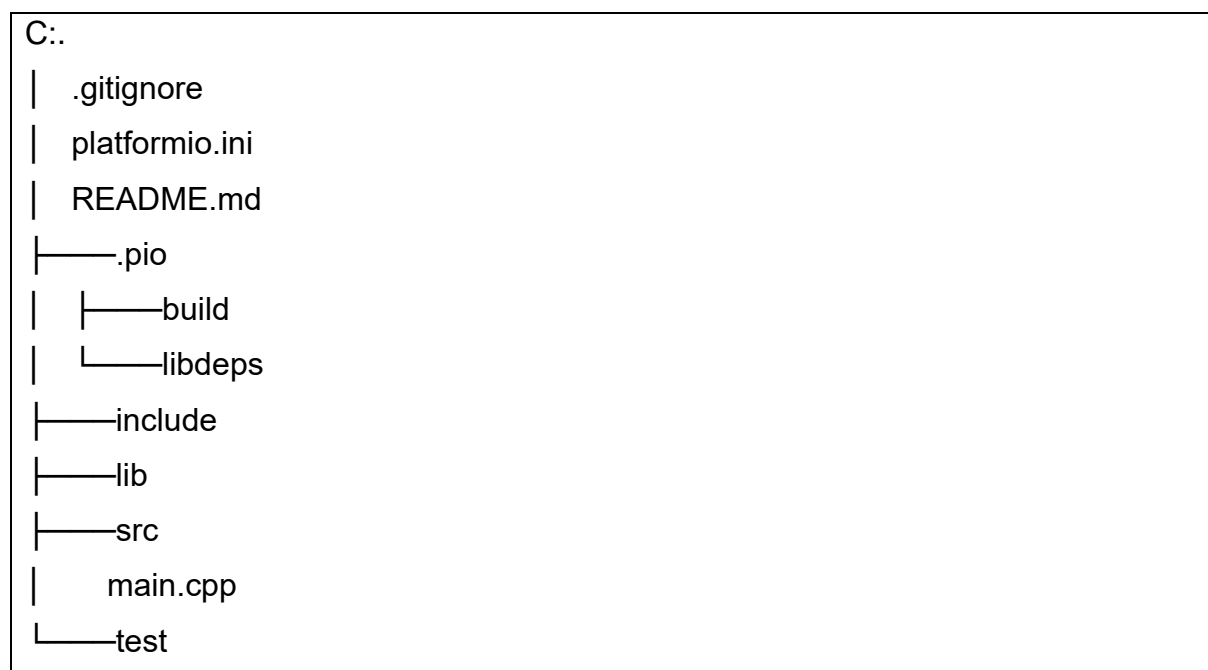
DIJAGRAM TOKA RADA PROGRAMA

7.2 Razvojno okruženje PlatformIO

PlatformIO je cross-platform i cross-architecture profesionalno razvojno okruženje namijenjeno razvoju softwarea za embedded sustave. PlatformIO IDE je ekstenzija za Visual Studio Code (VSCode) koji je jedan od najkorištenijih open source uređivača koda. VSCode za razliku od standardnog Arduino IDE-a donosi moderno razvojno okruženje koji programeru ubrzava i olakšava razvoj softwarea. Visual Studio Code za razliku od Arduino IDE-a automatski prepoznaje greške u kodu te ima mogućnost rada sa više datoteka odjednom. PlatformIO donosi i mogućnosti debugginga, kompajliranja koda i automatskog flashanja zadanog mikrokontrolera. [12]

Također postoje mnogobrojne ekstenzije tj. proširenja koja su napravljene od strane samih korisnika programa. PlatformIO podržava više Frameworkova tj. SDK-a (Static Code Analysis) poput Arduino frameworka, Espressif, FreeRTOS-a ...

Pri izradi novog PlatformIO projekta nakon podešavanja parametara poput korištenje pločice i frameworka otvara se prazan projekt sa sljedećom strukturom datoteka:



SLIKA 10 STRUKTURA DATOTEKA U PLATFORMIO RAZVOJNOM OKRUŽENJU

Glavni kod za odabrani mikroupravljač se piše u main.cpp datoteci. Također, projekt sam stvara .gitignore datoteku kao bi sustav za verzioniranje pri pospremanju različitih verzija preskočio datoteke koje se nalaze u .pio direktoriju. Direktorij .pio sadržava sve

dodane biblioteke i njihove pripadajuće primjere itd. Također ignorira sve datoteke koje nastaju kao posljedica kompajliranja programskog koda.

PlatformIO za podešavanje postavki koristi konfiguracijsku datoteku `platformio.ini` u kojoj se podešavaju jednostavnije postavke poput korištene platforme, *frameworka*, korištene pločice, brzine serijske komunikacije s računalom i zahtjevnije postavke poput korištenih biblioteka i brzine radnog takta procesora. U `platformio.ini` datoteci se koristi sintaksa python programskog jezika.

```
[env:esp32dev]
platform = espressif32
board = esp32dev
framework = arduino
monitor_speed = 115200
upload_speed = 921600
build_unflags =
    -DBOARD_HAS_PSRAM
lib_deps =
    tinyu-zhao/TinyGPSPlus-ESP32@^0.0.2
    adafruit/Adafruit GFX Library@^1.11.7
    adafruit/Adafruit SSD1306@^2.5.7
```

SLIKA 11 SADRŽAJ PLATFORMIO.INI DATOTEKE

7.3 Metodologija pisanja koda

Arduino je *open – source* platforma namijenjena izradi elektroničkih projekata. Arduino se sastoji od fizičke pločice tj. mikroupravljača i softwarea IDE-a i *frameworka*. Kod se piše u pojednostavljenoj verziji C++ - a što pojednostavljuje razvoj. Također Arduino kompatibilne razvojne pločice ne zahtijevaju zaseban hardwareski programer nego se kod stroji kod na mikroupravljač snima preko micro USB kabela. Arduino je postao platforma popularna među početnicima, ali i naprednijim korisnicima zbog velike količine različitih biblioteka. [13]

Programski kod je napisan u PlatformIO razvojnom okruženju s Arduino frameworkom. Glavni kod ovog projekta pohranjen je u datoteci `main.cpp`. Kod se sastoji od više logičkih cjelina i pri njegovom pisanju se nastojalo napisati ga čitljivo i lako za naknadno mijenjanje. Upotrebom `#define` naredbi omogućeno je lako mijenjanje varijabli kao što su pinovi različitih komponenata i podatci kao što su ime mreže (SSID) i lozinka iste

koje je potrebno striktno definirati (hardkodirati). Mnoge radnje koje se trebaju izvršavati više puta ili one koje zahtijevaju puno popratnog koda su izvađene van iz glavnih funkcija `setup()` i `loop()` kako bi se povećala čitljivost samog koda.

Tijekom razvoja koda korišten je sustav za verzioniranje Git za potrebu praćenja različitih verzija i sigurnosnog kopiranja u udaljeni GitHub repozitorij.

7.4 Zanimljivi dijelovi koda

```
void SpajanjeNaWiFi() {
    WiFi.mode(WIFI_OFF);
    delay(1000);

    WiFi.mode(WIFI_STA);

    WiFi.begin(SSID, PASSWORD);
    Serial.println("Spajanje na WiFi");

    while (WiFi.status() != WL_CONNECTED){
        delay(500);
        Serial.print(".");
    }
    Serial.print("Spojeno na: ");
    Serial.print("Ime mreže : "); Serial.println(SSID);
    Serial.print("IP address: "); Serial.println(WiFi.localIP());
}
```

SLIKA 12 FUNKCIJA ZA SPAJANJE NA WiFi

U dijelu koda na slici 12. mikroupravljač ESP 32 se spaja na WiFi mrežu s funkcijom iz biblioteke `WiFi.h`. U `WiFi.begin` se upisuju prethodno definirano ime mreže (SSID) i njena lozinka. Zatim se tijekom uspostavljanja konekcije ispisivanjem (“.”) pokazuje da se zaista ta konekcija pokušava uspostaviti i ispisuje se do trenutka uspostave same konekcije. Na kraju se ispisuje u terminal tj. serial monitor ime mreže na koju se mikrokontroler spojio i IP adresa koju mu je *router* dodijelio.


```

void IRAM_ATTR Timer0_ISR(){
    upis_flag = true;
}

//konfigurirano na 30 sekundi
Timer0_Cfg = timerBegin(0, 64000, true);
timerAttachInterrupt(Timer0_Cfg, &Timer0_ISR, true);
timerAlarmWrite(Timer0_Cfg, 37500, true);
timerAlarmEnable(Timer0_Cfg);

```

SLIKA 13 KONFIGURACIJA I UPOTREBA TIMERA0

Konfiguracija timera se vrši u „setup()“ funkciji s naredbom „timer.Begin“ u kojoj se zadaje korišteni timer tj. Timer0, prescaler na 64000 i smjer brojanja (na više ili niže). Prescaler na ESP 32 ima rezoluciju od 2^{16} . Funkcija „timerAttachInterrupt()“ povezuje prekidnu rutinu „Timer0_ISR“ s „Timer0“ tajmerom. U sljedećoj liniji koda se odabire broj impulsa unutarnjeg *clocka* od nakon kojeg se poziva prekidna rutina timera 0. ESP 32 za razliku od većine mikroupravljača posjeduje zasebni *clock* za tajmere od 80 Mhz. Broj impulsa koje odgovara vremenu od 30 sekundi (t_{T0}) sprema se u registar TCNT0. Rezolucija svih timera na ESP 32 je 2^{64} . Konfiguracija parametara *timera* provodi se na temelju jednadžbe (1):

$$t_{T0} = \frac{\text{Prescaler}}{\text{clock}} \cdot (\text{max vrijednost timera} - TCNT0_0) \quad (1)$$

```

if (gps.location.isValid() && upis_flag == true){

    String postData = "latitude=" + String(latitude,6) +
                      "&longitude=" + String(longitude,6) +
                      "&co2=" + String(CO2);

    HTTPClient http;
    http.begin(SERVER_PATH);
    http.addHeader("Content-Type", "application/x-www-form-urlencoded");
}

```

```

        int httpCode = http.POST(postData);
        String payload = http.getString();

        Serial.println("CO2: " + String(CO2));
        Serial.print("geografska duzina: "); Serial.println(latitude, 6);
        Serial.print("geografska sirina: "); Serial.println(longitude, 6);
        Serial.print("URL: "); Serial.println(SERVER_PATH);
        Serial.print("Data: "); Serial.println(postData);
        Serial.print("httpCode: "); Serial.println(httpCode);
        Serial.print("payload: "); Serial.println(payload);
        Serial.println("-----");
    );
    upis_flag = false;
}

```

SLIKA 14 SLANJE PODATAKA SA HTTP.POST FUNKCIJOM U BAZU PODATAKA

Na slici 14. vidimo funkciju za konekciju mikroupravljača s udaljenom bazom podataka. Ova funkcija se izvršava samo ako GPS ima „fix“ tj. dobiva položajne signale od satelita i ako je „timer0“ postavio bool varijablu „upis_flag“ u stanje „true“. Podaci koje želimo poslati u udaljenu bazu podataka pospremamo u varijablu „postData“ koja je tipa „String“. Pozivanjem funkcije „http.begin“ uspostavljamo http konekciju sa serverom. „http POST“ metodom šaljemo izmjerene podatke u udaljenu MySQL bazu podataka. Posljedna radnja koja se događa je ispis dijagnostičkih podataka na serial monitor poput poslanog stringa, http koda i dijagnostike PHP skripte.

7.5 Korištene biblioteke

TABLICA 5 KORIŠTENE BIBLIOTEKE

Ime biblioteka	Opis
SPI.h Wire.h HardwareSerial.h	Biblioteke potrebne za ostvarivanje seriske UART komunikacije
HTTPClient.h WiFi.h WiFiClientSecure.h	Biblioteke potrebne za konekciju sa lokalnom mrežom i MySQL serverom
TinyGPSPlus.h	Biblioteka za dekodiranje NMEA rečenica dobivenih sa GPS modula
Adafruit_GFX.h Adafruit_SSD1306.h	Adafruit biblioteke za upravljanje OLED SSD1306 zaslonom i prikazivanje različitih grafika
lozinke.h	Vlastita biblioteka za pohranu osjetljivih podataka tj. lozinke WiFi mreže na koju se ESP 32 spaja
pinovi.h	Vlastita biblioteka koja je namjenjena pohrani definicijama korištenih pinova za senzore i oled zaslon

7.6 MySQL baza podataka i njena konfiguracija

Baza podataka je sustav za pohranu i organizaciju različitih vrsta podataka. Početni podaci često započinju u Excel proračunskim tablicama, ali kako te tablice rastu, često se suočavamo s problemima kao što su redundancija i nedosljednosti podataka. Takvi podaci postaju sve teže i teže čitljivi u tabličnom obliku, a ograničeni su i mogućnostima pretraživanja i izvlačenja specifičnih podataka za analizu. Kada se pojave ovi izazovi, korisno je premjestiti podatke u udaljenu bazu podataka. Baza podataka je spremnik objekata u kojem možemo organizirati više tablica i omogućuje nam efikasno upravljanje i pohranu podataka. [14]

Za ovaj projekt odabrana je MySQL baza podataka zbog jednostavnosti upotrebe i inicijalne konfiguracije. Za pokretanje same baze podataka i pripadajućeg Apache web servera koristi se program XAAMP Control Panel koji je PHP razvojno okruženje. XAAMP se izvršava na stolnom računalu na Windows operacijskom sustavu na lokalnoj mreži (localhost).

Konfiguracija same baze podataka izvršava se u grafičkom web sučelju MySQL-a unutar web preglednika. Za početak potrebno je napraviti novu bazu podataka a zatim novu tablicu. Pri konfiguraciji preporučeno je koristiti ID stupac s uključenom „AUTO_INCREMENT“ opcijom koja inkrementira broj retka tj. dodaje svakom retku njegov broj. Također treba paziti na tip podatka koji se dodaje u svaki stupac što znači da se cjelobrojni unosi stavljaju u INT (*Integer*) tip a decimalnu u *Float* . Stupac „DateTime“ ima uključenu opciju „current_timestamp()“ koja dodaje vrijeme dodavanja pojedinog unosa u bazu podataka u formatu godina-mjesec-datum sat.

Za potrebe ovog projekta tablica ima 5 stupaca.

1. ID stupac – broj stupca, *Integer*
2. *Datatype* – vrijeme zapisivanja
3. CO2 – vrijednost CO₂ senzora, *Integer*
4. *Latitude* – geografska širina, *Float*
5. *Longitude* – geografska dužina, *Float*

<div><div></div><div></div><div></div></div>			<div><div></div>id<div></div>1</div>	datetime	latitude	longitude	co2
<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div> Edit <div><div></div><div></div><div></div></div> Copy <div><div></div><div></div><div></div></div> Delete	246	2023-09-17 20:35:12	45.8062	15.9773	405	
<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div> Edit <div><div></div><div></div><div></div></div> Copy <div><div></div><div></div><div></div></div> Delete	247	2023-09-17 20:35:37	45.8062	15.9773	432	
<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div> Edit <div><div></div><div></div><div></div></div> Copy <div><div></div><div></div><div></div></div> Delete	248	2023-09-17 20:36:09	45.8063	15.9773	455	
<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div> Edit <div><div></div><div></div><div></div></div> Copy <div><div></div><div></div><div></div></div> Delete	249	2023-09-17 20:36:42	45.8063	15.9772	471	
<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div> Edit <div><div></div><div></div><div></div></div> Copy <div><div></div><div></div><div></div></div> Delete	250	2023-09-17 20:37:15	45.8062	15.9772	542	
<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div> Edit <div><div></div><div></div><div></div></div> Copy <div><div></div><div></div><div></div></div> Delete	251	2023-09-17 20:37:39	45.8062	15.9772	533	
<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div> Edit <div><div></div><div></div><div></div></div> Copy <div><div></div><div></div><div></div></div> Delete	252	2023-09-17 20:38:11	45.8062	15.9772	644	
<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div> Edit <div><div></div><div></div><div></div></div> Copy <div><div></div><div></div><div></div></div> Delete	253	2023-09-17 20:38:44	45.8062	15.9772	578	
<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div> Edit <div><div></div><div></div><div></div></div> Copy <div><div></div><div></div><div></div></div> Delete	254	2023-09-17 20:39:08	45.8062	15.9772	621	
<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div> Edit <div><div></div><div></div><div></div></div> Copy <div><div></div><div></div><div></div></div> Delete	255	2023-09-17 20:39:41	45.8062	15.9772	625	

SLIKA 15 IZGLEDE MYSQL TABLICE

7.7 PHP skripta

PHP skripta u okviru ovog projekta igra ključnu ulogu u cijelom sustavu. Njezina svrha je obrada i interpretacija „HTTP POST“ zahtjeva koje uređaj šalje u bazu podataka . Ova obrada podataka omogućuje skripti da izvuče važne informacije iz zahtjeva i smjesti ih na odgovarajuća mjesta u MySQL bazi podataka. PHP programski jezik je odabran za ovu svrhu zbog jednostavnosti programiranja, dostupnosti materijala na internetu i podrške u za MySQL bazu podataka.

```
<?php
/*
    ~\xampp\htdocs\co2_projekt
*/

$hostname = "localhost";
$username = "root";
$password = "";
$database = "sensor_db";

$conn = mysqli_connect($hostname, $username, $password, $database);

if (!$conn) {
    die("konekcija je neuspješna" . mysqli_connect_error());
}

echo "konekcija sa bazom je OK<br>";

if(isset($_POST["latitude"]) && isset($_POST["longitude"]) &&
isset($_POST["co2"])) {

    $lat = $_POST["latitude"];
    $long = $_POST["longitude"];
    $co2 = $_POST["co2"];

    //test vrijednosti
    /*
    $lat = 33;
    $long = 666;
    $co2 =323400;
    */

    $sql1 = "INSERT INTO co2_tablica (latitude, longitude, co2) VALUES
    (\".$lat.\",\".$long.\",\".$co2.\")";
```

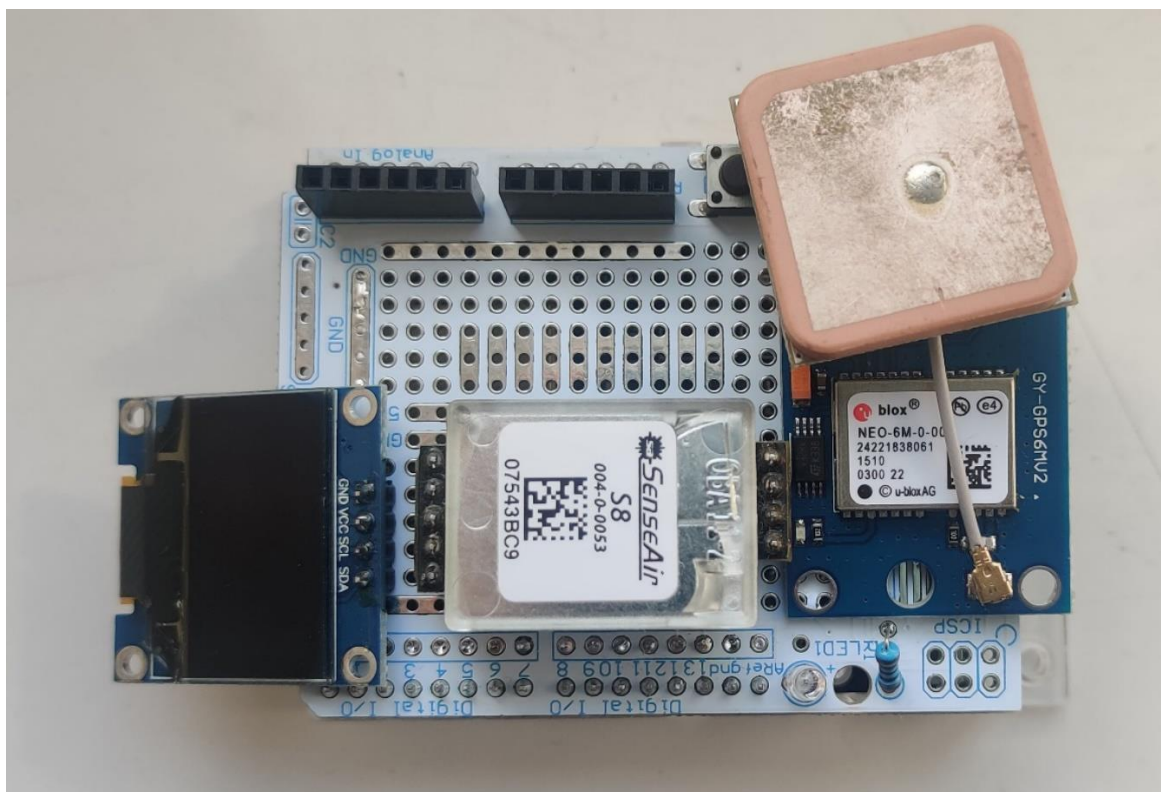
```
if (mysqli_query($conn, $sql1)) {  
    echo "\nnova vrijednost je uspjesno dodana";  
} else {  
    echo "Error: " . $sql1 . "<br>" . mysqli_error($conn);  
}  
  
}  
  
?>
```

SLIKA 16 PHP SKRIPTA

Na početku PHP skripte definirane su osnovni parametri baze podataka kao što su lokacija servera, login podaci i ime baze podataka. Default korisničko ime je root a lozinka je prazan string i ti podaci nisu mijenjani. Naredbom `mysql_connect` povezujemo PHP skriptu i bazu podataka i rezultat pokušaja konekcije ispisuje se na web pregledniku i serial konzoli mikroupravljača. PHP skripta se nalazi u `xampp/htdocs/test.php` datoteci i također je možemo manualno testirati u web pregledniku odlaskom na adresu „localhost/co2_projekt/test.php“. Kada skripta primi http post naredbu izvršava se „INSERT“ naredba i nova vrijednost se dodaje u odgovarajući stupac. Također nas o uspješnosti unosa skripta obavještava kroz konzolu tj. serial monitor.

8. Završni izgled uređaja

Mjerni uređaj napravljen u ovom radu je zalemljen na Arduino shield za izradu prototipova koji je zatim nataknut na mikrokontroler. Svi spojevi su zalemljeni s donje strane mjernog uređaja. Na slici 17. se vidi završni izgled napravljenog uređaja.



SLIKA 17 ZAVRŠNI IZGLED UREĐAJA

9. Obrada podataka

Obrada mjerenja vrši se u programu Microsoft Excel nakon izvoza izmjerenih mjerenja iz MySQL baze podataka u CSV (*comma separated value*) formatu. Izvoz baze mora biti izvršen prema predlošku "CSV for Excel" kako bi podatke pohranjene unutar datoteke excel mogao razvrstati u stupce. Nakon što preuzmemo podatke, potrebno je zamijeniti sve decimalne točke (".") zarezima (",") u programu poput Notepada, jer Microsoft Excel koristi zareze kao decimalni separator u izračunima. Nakon toga potrebno je otvoriti novu excel datoteku i unijeti podatke u sam excel tako da pod karticom "Data" odaberemo opciju "From Text/CSV". Odabirom CSV datoteke s računala otvara se prozor u kojem kliknemo "Transform Data". Zatim moramo pregledati da li tip podataka u Excelu odgovara stvarnom tipu podataka. Pritiskom na "Close & Load" dobivamo tablicu sa našim podacima koji su spremni za daljinu statističku obradu i vizualizaciju. Mapiranje podataka vrši se sa funkcijom "3D Map".

10. Mjerenje potoka u naselju Voltino

Mjerenje rute potoci obavljeno je 18.9.2023 u periodu od 19.10h do 20.05h. Mjerenje je obavljeno na biciklu s izrađenim mjernim uređajem. Mjerni uređaj je bio spojen preko bežične mreže mobilnog uređaja na laptop s bazom podataka. Također je uređaj bio spojen s micro USB kabelom na laptop radi napajanja. Pri pokretanju mjerenja trebalo je voditi računa o IP adresi laptopa koja je spremljena na mikroupravljač da bi se podatci uspješno spremili u bazu podataka.obavljala u slučaju pronalaska bolje



SLIKA 18 RUTA 1. MJERENJA

Na slici 18. se vidi ruta mjerenja koju smo dobili s GPS modulom uređaja i vizualizirali u Microsoft Excelu s funkcijom "3D Map". CO2 senzor tijekom mjerenja nije bio

ispravno kalibriran te je bilo potrebno naknadno normaliziranje rezultata mjerenja. Normalizacija rezultata u naknadnoj obradi je napravljena radi zadržavanja mogućnosti potencijalne promjene formule po kojoj se sama obavljala u slučaju pronalaska bolje.

id	datetime	latitude	longitude	co2	co2 normaliziran
448	19:10:26	45,8003	15,9283	13365	398
449	19:10:48	45,8003	15,9283	21069	411
450	19:11:07	45,8003	15,9283	12599	406
451	19:11:26	45,8003	15,9283	21069	411
452	19:11:44	45,8003	15,9283	21069	411
453	19:12:02	45,8003	15,9283	21069	411
454	19:12:21	45,8003	15,9283	21069	411
455	19:12:40	45,8002	15,9283	11313	406
456	19:12:59	45,8001	15,9283	21069	411
457	19:13:17	45,8001	15,9283	21069	411
458	19:13:36	45,8001	15,9283	21069	411
459	19:13:54	45,8002	15,9283	11316	406
460	19:14:12	45,8001	15,9283	21069	411
461	19:14:31	45,8000	15,9285	21069	411
462	19:14:50	45,7996	15,9285	11314	406
463	19:15:09	45,7992	15,9285	21334	411
464	19:15:27	45,7991	15,9285	11829	406
465	19:15:46	45,7990	15,9285	21069	411
466	19:16:05	45,7991	15,9282	11313	406
467	19:16:23	45,7992	15,9276	11315	406
468	19:16:42	45,7992	15,9272	12332	406
469	19:17:01	45,7993	15,9269	19532	410

SLIKA 19 IZGLED TABLICE SA REZULTATIMA U MICROSOFT EXCELU

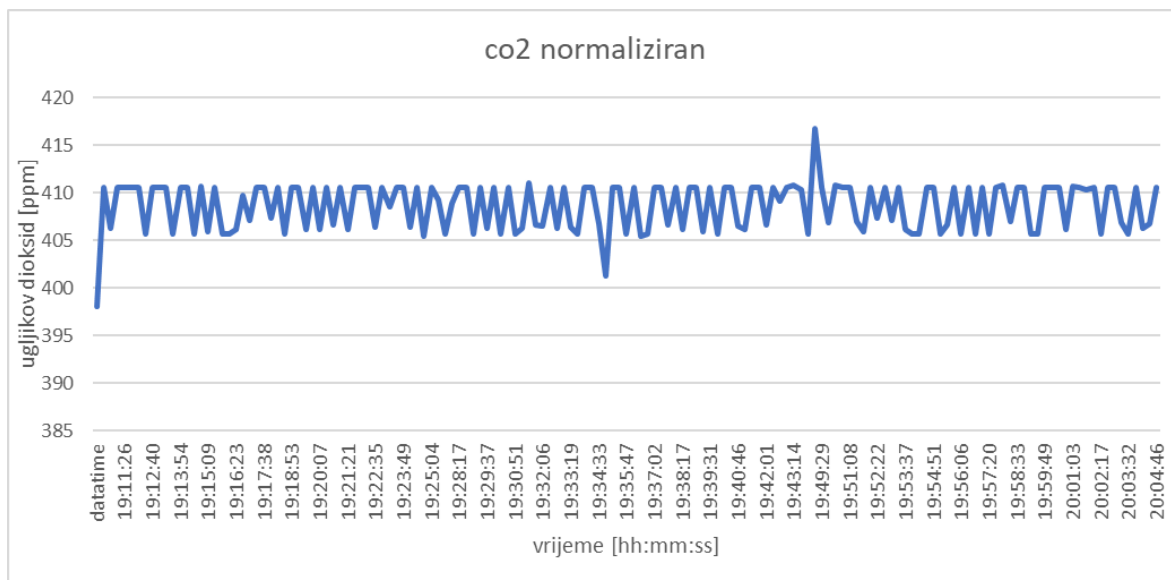
U stupcu co2 normaliziran se vide realniji rezultati mjerenja nego kakvi su originalno bili na izlazu senzora. Normalizacija co2 rezultata izračunata je po formuli (2).

$$=([@co2] - \text{srednja vrijednost}) / \text{opseg senzora} + \text{vrijednost na otvorenom} \quad (2)$$

Korištene vrijednosti za “opseg senzora” je 2000 a “vrijednost na otvorenom prostoru” je 400.

Nažalost mi za dobivene rezultate ne možemo biti sigurni da su točni zato što nemamo drugi referentni senzor za usporedbu dobivenih rezultata.

Na slici 20. vidimo graf normaliziranih vrijednosti CO₂ i možemo doći do zaključka da nema velikih odstupanja i velikih zagađenja jer je mjerenje većinski obavljeno na pješačkom putu uz potoke.



SLIKA 20 GRAF NORMALIZIRANIH CO₂ VRIJEDNOSTI



Slika 21 Karta sa CO₂ normaliziranim vrijednostima

Na sljedećoj karti je stupac co2 normaliziran korišten kao informacija visine. Možemo zaključiti s karte da se najveće oneobičenje može zabilježiti na križanjima većih cesta. Veliko zagađenje je zabilježeno na križanjima Zagrebačke ceste i Sokolske ulice te križanju Baštijanove i Golikove ulice

11. Zaključak

U okviru ovog završnom radu izrađen je mobilni uređaj za mjerenje koncentracije ugljikovog dioksida (CO₂) na zadanoj pješačkoj ruti. Za mjerni uređaj izrađena je sklopovska i programska podrška. Potreba za takvim mjernim uređajem postoji zbog mnogobrojnih negativnih posljedica visokih koncentracija ugljikovog dioksida na ljudsko zdravlje i okoliš. Posljedice loše kvalitete zraka na ljudsko zdravlje su najviše prisutne u gusto naseljenim i urbaniziranim sredinama zbog velike količine prometa i industrije.

Uređaj automatski tijekom mjerenja šalje izmjerene podatke i lokaciju mjerenja u udaljenu MySQL bazu podataka kako bi mjerenja bila sigurno pohranjena i lako dostupna za naknadnu obradu i analizu. Također tijekom mjerenja uređaj ispisuje rezultate mjerenja na OLED zaslon i dijagnostičke podatke vezano za uspješnost povezivanja s Wi-Fi mrežom i slično. Sklopovska podrška se temelji na ESP 32 WROOM mikrupravljaču, a programska na Arduino frameworku. Rad se sastojao od razvoja vlastitog programa za ESP 32 mikroupravljač u Arduino frameworku, konfiguracije MySQL udaljene baze podataka i razvoja vlastite PHP skripte za unos podataka u bazu.

U sklopu rada izmjerena je koncentracija ugljikovog dioksida na ruti potoci u naselju Voltino u kojem se može zaključiti da su najveće koncentracije ugljikovog dioksida posljedice gustog prometa na velikim raskrižjima. Naknadna obrada mjerenja i vizualizacija rezultata izvršena je u Microsoft Excel programu.

Literatura

- [1] EPA, *Basics of Climate Change* <https://www.epa.gov/climatechange-science/basics-climate-change> (pristupano 20.9.2023.)
- [2] C. Johnson, *The Health Effects of CO2 & How To Protect Yourself* <https://learn.kaiterra.com/en/air-academy/health-effects-of-co2> (pristupano 20.9.2023.)
- [3] Wisconsin Department of Health Services, *Carbon Dioxide* <https://www.dhs.wisconsin.gov/chemical/carbondioxide.htm> (pristupano 20.9.2023.)
- [4] European Environment Agency, *Čišći zrak doprinosi ljudskom zdravlju i smanjuje posljedice klimatskih promjena* <https://www.eea.europa.eu/hr/articles/cisci-zrak-doprinosi-ljudskom-zdravlju> (pristupano 20.9.2023.)
- [5] *ESPduino 32 Pinout* <https://designtech.blogs.auckland.ac.nz/d1-r32-esp32/> (pristupano 20.9.2023.)
- [6] *ESP 32 WROOM 32 Datasheet* https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32_datasheet_en.pdf (pristupano 20.9.2023.)
- [7] *Seansair S8 LP Datasheet* <https://senseair.com/products/size-counts/s8-lp/> (pristupano 20.9.2023.)
- [8] *Operating principle of NDIR* <https://www.figaro.co.jp/en/technicalinfo/principle/ndir-type.html> (pristupano 20.9.2023.)
- [9] LWIR Systems, *Optical Filters Open Up New Uses for MWIR*, <https://www.photonics.com/a56392> (pristupano 20.9.2023.)
- [10] *NEO6MV2 GPS module Datasheet* https://content.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_%28GPS.G6-HW-09005%29.pdf (pristupano 20.9.2023.)
- [11] *GNSS antenna* <https://www.newark.com/molex/206640-0001/antenna-patch-1-597-1-607ghz-4/dp/62AC0252> (pristupano 20.9.2023.)
- [12] *What is PlatformIO* <https://docs.platformio.org/en/latest/what-is-platformio.html> (pristupano 20.9.2023.)

- [13] *What is Arduino* <https://learn.sparkfun.com/tutorials/what-is-an-arduino/all>
(pristupano 20.9.2023.)
- [14] *Microsoft osnove baza podataka* <https://support.microsoft.com/hr-hr/office/osnove-baza-podataka-a849ac16-07c7-4a31-9948-3c8c94a7c204>
(pristupano 20.9.2023.)

Summary

The aim of this thesis was to create a mobile measuring device for measuring the concentration of carbon dioxide along a predefined pedestrian route. The need for such a device arises due to the numerous consequences of high concentrations of carbon dioxide on the environment and human health. The effects of poor air quality on human health are most pronounced in densely populated and urbanized areas due to heavy traffic and industrial activities. During measurements, the device automatically sends the measured data and the measurement location to a remote MySQL database to ensure secure storage and easy access for subsequent processing and analysis. The post-processing of measurements and visualization of results are done in the Microsoft Excel program. In this work, both hardware and software support for the device have been developed. The hardware support is based on the ESP 32 WROOM microcontroller, while the software is based on the Arduino framework. The work includes the development of a custom program for the ESP 32 microcontroller in the Arduino framework, configuration of a remote MySQL database, and the development of a custom PHP script for data entry into the database.

Keywords: air quality measurment, concentrations of carbon dioxide, ESP 32 microcontroller, MySQL database, IoT

Prilog

11.1 Arduino kod

pinovi.h

```
//pinovi za GPS
#define GPS_RX 16
#define GPS_TX 17

// pinovi za S8
#define S8_RX 13
#define S8_TX 12
```

lozinke.h

```
datoteka za pohranu osjetljivih podataka
*/
#define SSID ""
#define PASSWORD ""
#define SERVER_PATH ""
//ne zaboravi promijeniti IP
```

main.cpp

```
#include <Arduino.h>
#include <SPI.h>
#include <Wire.h>
#include <HttpClient.h>
#include <WiFi.h>
#include <WiFiClientSecure.h>
#include <TinyGPSPlus.h>
#include <HardwareSerial.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <SPI.h>

#include <pinovi.h>
#include <lozinke.h>
```

```

#define SERIAL_BRZINA 115200
#define LED_INTERNI 2

hw_timer_t *Timer0_Cfg = NULL;

// za GPS
HardwareSerial neogps(1); //uart port 1
TinyGPSPlus gps;

float longitude, latitude;

// za CO2 senzor
byte CO2req[] = {0xFE, 0X44, 0X00, 0X08, 0X02, 0X9F, 0X25};
byte CO2out[] = {0, 0, 0, 0, 0, 0, 0};

//za oled
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define OLED_RESET -1
#define SCREEN_ADDRESS 0x3C //0x3D ako ne radi
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);

void Alarm(){
    for (int8_t i = 0; i < 3; i++){
        digitalWrite(LED_INTERNI, HIGH);
        delay(500);
        digitalWrite(LED_INTERNI, LOW);
        delay(500);
    }
}

void RequestCO2(){

    while (!Serial1.available()){

        Serial1.write(CO2req, 7);
        delay(50);
    }

    uint8_t timeout = 0;
    while (Serial1.available() < 7){

        timeout++;
        if (timeout > 10){

            while (Serial1.available())

```

```

        Serial1.read();
        break;
    }
    delay(50);
}

for (uint8_t i = 0; i < 7; i++)
    CO2out[i] = Serial1.read();
}

unsigned long CO2count(){

    int high = CO2out[3];
    int low = CO2out[4];
    unsigned long val = high * 256 + low;
    return val * 1;
}

void SpajanjeNaWiFi() {
    WiFi.mode(WIFI_OFF);
    delay(1000);

    WiFi.mode(WIFI_STA);

    WiFi.begin(SSID, PASSWORD);
    Serial.println("Spajanje na WiFi");

    while (WiFi.status() != WL_CONNECTED){
        delay(500);
        Serial.print(".");
    }
    Serial.print("Spojeno na: ");
    Serial.print("Ime mreže : "); Serial.println(SSID);
    Serial.print("IP adresa: "); Serial.println(WiFi.localIP());
}

bool upis_flag = false;

void IRAM_ATTR Timer0_ISR(){
    upis_flag = true;
}

void setup(){

    pinMode(LED_INTERNI, OUTPUT);

    Serial.begin(SERIAL_BRZINA);

```

```

Serial1.begin(9600, SERIAL_8N1, S8_RX, S8_TX);
neogps.begin(9600,SERIAL_8N1, GPS_RX, GPS_TX);

display.begin(SSD1306_SWITCHCAPVCC,SCREEN_ADDRESS);
display.clearDisplay();
display.setTextSize(2);
display.setTextColor(WHITE);
display.setCursor(0,0);
display.println("Start");
display.display();

delay(1000);

//konfigurirano na 30 sekundi
Timer0_Cfg = timerBegin(0, 64000, true);
timerAttachInterrupt(Timer0_Cfg, &Timer0_ISR, true);
timerAlarmWrite(Timer0_Cfg, 37500, true);
timerAlarmEnable(Timer0_Cfg);

if(!display.begin(SSD1306_SWITCHCAPVCC, SCREEN_ADDRESS))
    Serial.println(F("SSD1306 nije spojen"));

SpajanjeNaWiFi();
}

void loop(){

    RequestCO2();
    unsigned long CO2 = CO2count();
    delay(2000);

    //za mysql bazu podataka
    if(WiFi.status() != WL_CONNECTED) {
        SpajanjeNaWiFi();
    }

    //za tinyGPS
    boolean newData = false;
    for (unsigned long start = millis(); millis() - start < 1000;) {
        while (neogps.available()) {
            if (gps.encode(neogps.read())) {
                newData = true;
            }
        }
    }
}

```

```

if (gps.location.isValid() == 1) {

    latitude = gps.location.lat();
    longitude = gps.location.lng();

    //Serial.println(gps.location.lat(), 6);
    //Serial.println(gps.location.lng(), 6);

    delay(5000);
}

if (gps.location.isValid() && upis_flag == true){

    String postData = "latitude=" + String(latitude,6) +
                      "&longitude=" + String(longitude,6) +
                      "&co2=" + String(CO2);

    HTTPClient http;
    http.begin(SERVER_PATH);
    http.addHeader("Content-Type", "application/x-www-form-urlencoded");

    int httpCode = http.POST(postData);
    String payload = http.getString();

    Serial.println("CO2: " + String(CO2));
    Serial.print("geografska duzina: "); Serial.println(latitude, 6);
    Serial.print("geografska sirina: "); Serial.println(longitude, 6);
    Serial.print("URL: "); Serial.println(SERVER_PATH);
    Serial.print("Data: "); Serial.println(postData);
    Serial.print("http Code: "); Serial.println(httpCode);
    Serial.print("primljeni podaci: "); Serial.println(payload);
    Serial.println("-----");
");

    display.clearDisplay();
    display.setTextSize(2);
    display.setTextColor(WHITE);
    display.setCursor(0,0);
    display.println(CO2);
    display.display();

    if (httpCode == 200){

```

```

        display.setTextSize(2);
        display.setTextColor(WHITE);
        display.setCursor(0,15);
        display.println("upis je ok");
        display.display();
    }else{
        display.setTextSize(2);
        display.setTextColor(WHITE);
        display.setCursor(0,15);
        display.println("nije upisano");
        display.display();
    }

    delay(10000);

    upis_flag = false;
}
}

```

11.2 PHP skripta

```

<?php
/*
    ~\xampp\htdocs\co2_projekt
*/

$hostname = "localhost";
$username = "root";
$password = "";
$database = "sensor_db";

$conn = mysqli_connect($hostname, $username, $password, $database);

if (!$conn) {
    die("konekcija je neuspješna" . mysqli_connect_error());
}

echo "konekcija sa bazom je OK<br>";

if(isset($_POST["latitude"]) && isset($_POST["longitude"]) &&
isset($_POST["co2"])) {

```

```

$lat = $_POST["latitude"];
$long = $_POST["longitude"];
$co2 = $_POST["co2"];

//test vrijednosti
/*
$lat = 33;
$long = 666;
$co2 = 323400;
*/

$sql1 = "INSERT INTO co2_tablica (latitude, longitude, co2) VALUES
('.$lat.', '.$long.', '.$co2.')";

if (mysqli_query($conn, $sql1)) {
    echo "\nnova vrijednost je uspjesno dodana";
} else {
    echo "Error: " . $sql1 . "<br>" . mysqli_error($conn);
}
}

?>

```

