

UNIVERZITET CRNE GORE
ELEKTROTEHNIČKI FAKULTET U PODGORICI



NIKOLA MRKIĆ

**PRILOG RAZVOJU PAMETNIH SENZORA ZA
KUĆNU AUTOMATIZACIJU**

MAGISTARSKI RAD

PODGORICA 2020

PODACI I INFORMACIJE O MAGISTRANDU

Ime i prezime: **Nikola Mrkić**

Datum i mjesto rođenja: **30.06.1968 god. Nikšić**

Naziv završenog osnovnog studijskog programa i godina završetka studija:

Elektronika,telekomunikacije i racunari 2002 godina

INFORMACIJE O MAGISTARSKOM RADU:

Naziv postiplomskog studija : **Odsjek računari**

Naziv rada : **Prilog razvoju pametnih senzora za kućnu automatizaciju**

Fakultet na kojem je rad odbranjen: **Elektrotehnički fakultet ,Podgorica**

UDK,OCJENA I ODBRANA MAGISTARSKOG RADA

Datum prijave magistarskog rada: **26.06.2006 godine**

Datum sjednice naučno-nastavnog vijeća na kojoj je prihvaćena tema: **11.07.2006 godine**

Komisija za ocjenu teme i podobnosti magistranda u sastavu:

Prof.dr Djordjije Jovanović ,

Prof.dr. Srdjan Stanković,

Prof.dr. Radovan Stojanović

Mentor : **Prof.dr Radovan Stojanović**

Komisija za ocjenu rada:

Prof.dr Budimir Lutovac ,

Prof.dr. Radovan Stojanović,

Prof.dr. Neđeljko Lekić

Komisija za odbranu rada:

Prof.dr Budimir Lutovac ,

Prof.dr. Radovan Stojanović,

Prof.dr. Neđeljko Lekić

Datum odbrane:

Datum promocije:

Sažetak:

Ideja kućne automatizacije je stara više od jednog vijeka. Aktuelnost istraživanja u ovoj oblasti postaje dinamična i izazovna tokom poslednjih dvije decenije, sa razvojem modernih informaciono komunikacionih tehnologija. Ove tehnologije posjeduju potencijal da riješe problem ekonomske opravdanosti i korisnosti sistema kućne automatizacije i učine ih, konačno, široko upotrebljivim. Predmet istraživanja ostaje projektovanje sistema kućne automatizacije koji će optimizovati sledeće zahtjeve: tehničko upotrebne karakteristike, cijenu koštanja, jednostavnost rukovanja, održivost i mogućnost nadogradnje.

U tom kontekstu, u okviru ovog rada, sprovedena su istraživanja na temu projektovanja optimalnog sistema kućne automatizacije, koji će uzeti u obzir sve zahtjeve koje mora da ispuni upotrebljivi sistem ove vrste, a da se pri tom zasniva na pristupačnim, savremenim i jeftinim tehnologijama sensorika, mjernih i izvršnih elektronskih kola i Interneta. U tom cilju sprovedena je deskriptivna, kvalitativna i kvantitativna (testna) studija slučaja projektovanja (Internet of Things) IoT sistema kućne automatizacije prihvatljive cijene koštanja i zadovoljavajućih performansi, sa primjerom primjene u oblasti uštede energije za potrebe domaćinstava i ostalih objekata.

Studija je sumirala odgovarajuća profesionalna i istraživačka znanja iz oblasti kućne automatizacije u pogledu topologija, tehnologija, povezivanja, oblasti primjena i protokola. Predložen je i opisan integrisani sistem kućne automatizacije koji predstavlja široku platformu sa više mogućih scenarija upotrebe senzoričke, hardvera, komunikacija i softvera, a sve u cilju izbora optimalnog rešenja. Predložen je sistem kućne automatizacije baziran na IoT principu koji u kombinaciji sa klasičnim sistemima predstavlja veoma jeftino i efikasno rešenje. Koristi NodeMCU jeftinu IoT platformu otvorenog koda na čipu i kombinaciji sa otvorenim IoT platformama. Data su odgovarajuća hardverska i softverska rešenja IoT sistema kućne automatizacije sa primjerima izbora hardvera, projektovanja softvera i njihove integracije. Izvršeno je testiranje predloženog IoT sistema kućne automatizacije na primjeru kontrole potrošnje električne energije u domaćinstvu na konkretnoj lokaciji u definisanom vremenskom intervalu. Dodatno, ostala znanja vezana za sisteme kućne automatizacije su sistematizovana, uz elaboraciju testiranih primjera.

Ključne riječi: kućna automatizacija, pametna kuća, IoT, NodeMCU, ThingSpeak, senzori, Internet, energija

Abstract:

The idea of home automation is more than a century old. The relevance of research in this area has become dynamic and challenging over the last two decades, with the development of modern information and communication technologies. These technologies have the potential to solve the problem of economic justification and usefulness of home automation systems and make them, finally, widely used. The subject of research remains the design of home automation systems that will optimize the following requirements: technical usability characteristics, cost price, ease of operation, sustainability and upgradeability.

In this context, within this paper, research was conducted on the design of an optimal home automation system, which will take into account all the requirements that must be met by a usable system of this type, based on affordable, modern and inexpensive sensor technologies, measuring and executive electronic circuits and the Internet. To this end, a descriptive, qualitative and quantitative (test) case study of the design of the Internet of Things IoT home automation system at an acceptable cost and satisfactory performance, with an example of application in the field of energy saving for households and other facilities.

The study summarized relevant professional and research knowledge in the field of home automation in terms of topologies, technologies, connectivity, applications and protocols. An integrated home automation system was proposed and described, which represents a wide platform with several possible scenarios for the use of sensors, hardware, communications and software, all with the aim of choosing the optimal solution. A home automation system based on the IoT principle has been proposed, which in combination with classic systems represents a very cheap and efficient solution. It uses the NodeMCU cheap open source IoT platform on a chip and combined with open IoT platforms. Appropriate hardware and software solutions of IoT home automation systems are given with examples of hardware selection, software design and their integration. The testing of the proposed IoT home automation system was performed on the example of control of electricity consumption in the household at a specific location in a defined time interval. Additionally, other knowledge related to home automation systems is systematized, with the elaboration of tested examples.

Keywords: home automation, smart home, IoT, NodeMCU, ThingSpeak, sensors, Internet, energy

Sadržaj:

Sažetak:.....	2
Abstract:	4
Sadržaj:.....	5
Glava I	8
1 Uvod	8
Glava II.....	12
2. Kućna automatizacija-pregled	12
2.1 Pregled i prednosti	12
2.2 Sistem i upravljanje	12
2.2.1 Centralizovana kontrola	13
2.2.2 Decentralizovana kontrola	13
2.2.3 Polucentralizovana kontrola.....	14
2.3 Topologija.....	14
2.3.1 Topologija zvijezda	14
2.3.2 Topologija sabirnice.....	15
2.3.3 Topologija stabla.....	15
2.4 Međusobno povezivanje	16
2.4.1 Žičani sistemi	16
2.4.1.1 KNX / EIB	16
2.4.1.2 RS-485	17
2.4.1.3 CAN sabirnica	17
2.4.1.4 1-žična sabirnica	17
2.4.1.5 Ethernet.....	18
2.4.1.6 Powerline	18
2.4.2 Bežični sistemi	18
2.4.2.1 WLAN	19
2.4.2.2 ZigBee	19
2.4.2.3 EnOcean	19
2.4.2.4 IrDA.....	19
2.5 Zadaci.....	20
2.5.1 Osvjetljenje	20
2.5.5 Audio i Video.....	20
2.5.2 Roletne	21
2.5.3 Grijanje.....	22

2.5.4 Sigurnost	22
2.5.5 Malo komplikovaniji oblici pametne kuće.....	23
2.5.5.1 Pametne kuhinje	23
2.5.5.2 Pametna kupatila	24
2.5.5.3 Pametni vrtići.....	25
2.5.5.4 Pametna njega kućnih ljubimaca	26
2.5.5.5 Više pametnih opcija	26
2.5.5.6 Kućna automatizacija za medicinsku njegu.....	27
2.5.6 Je li automatizacija kuće sigurna?.....	28
2.5.7 Praćenje.....	29
Glava III.....	30
3. Integrisani sistem kućne automatizacije	30
3.1 Uvod.....	30
3.2 Arhitektura predloženog rešenja.....	31
3.2.1 Micro_Lan (1_Wire)	32
3.3 Bežični “smart senzori” za KA	35
3.3.1 Vrste bežičnih komunikacionih tehnologija za «smart» senzore	36
3.3.1.1 RF senzori.....	36
3.3.1.2 Bluetooth	36
3.3.1.3 Infrared	37
3.3.1.4 ZigBee	37
3.3.2 Arhitektura predloženog bežičnog smart «senzora» za kućnu automatizaciju	37
3.3.2.1 Senzori i ulazno kolo za kondicioniranje (front end).....	38
3.3.2.2 Mikrokontroler:	38
3.3.2.3 Komunikacioni gateway:	40
3.3.2.4 RF moduli za kućnu automatizaciju	40
3.3.2.5 ASK-RF.....	41
3.3.2.6 FSK-RF.....	43
3.3.2.7 RS232 preko RFa	45
3.4 Žičani PLC «smart senzori» za kućnu automatizaciju.....	46
3.4.1 Uvod.....	46
3.4.2 PLC model	47
3.4.3 PLC parametri i njihov opis	47
3.4.3.1 Impedansa.....	48
3.4.3.2 Šum.....	48
3.4.3.3 Slabljenje	49
3.4.3.4 Stojeći talas.....	50
3.4.4 Uobičajeni PLC protokoli	50

3.4.4.1 X10	50
3.4.4.2 Lonworks	52
3.4.4.3 CEBus	52
3.4.4.4 HomePlug	53
3.4.4.5 CENELEC	53
GLAVA IV	55
4. Kućna automatizacija bazirana na IoT principu	55
4.1 Kratki pregled kućne automatizacije bazirane na IoT principu	55
4.2 Pregled nekih od radova sistema kućne automatizacije baziranih na IoT	55
4.3 Moderni sistemi na čipu i ploči za potrebe IoT kućne automatizacije	56
4.3.1 ESP8266	56
4.3.1.1 Specifikacija ESP8266:	57
4.3.2 ESP32	58
4.3.2.1 Tehnička specifikacija ESP32	58
4.3.2.2 Pinout na ESP32 NodeMCU	59
4.3.2.3 Napajanje ESP32 NodeMCU	61
4.3.2.4 Režimi napajanja i potrošnja energije ESP32.....	63
4.4 Referentne IoT platforme.....	65
4.4.1 IFTTT vs. Zapier	66
4.5 ThingSpeak	68
4.5.1 ThingSpeak aplikacije	69
4.6 Kratak rezime komercijalnih IoT platform	70
4.7 Princip projektovanja IoT rešenja za kućnu automatizaciju bazirano na nodeMCUs	71
Glava V	74
5. Predloženo rješenje kućne automatizacije na bazi Sistema na čipu i dostupnih platformi	74
5.1 Uvod.....	74
5.2. DHT11 Senzor	76
5.3 HTTP zahtjevi: GET i POST	77
5.3.1 HTTP GET	78
5.3.2 HTTP POST	78
5.3.3 HTTP GET/POST sa ESP32	79
5.3.3.1 ESP32 HTTP GET: preko URL-a i JSONa.....	79
5.3.3.2 ESP32 HTTP POST: URL Encoded, JSON Data Object, Plain Text.....	80
5.3.3.3 HTTP POST URL Encoded	80
5.3.3.4 HTTP POST JSON Object	81
5.3.3.5 HTTP Plain Text.....	81
5.4 Slanje podataka sa ESP32 na ThingSpeak pomoću HTTP protokola	81

5.4.1 API zahtjevi (requests) za ThingSpeak	84
5.4.2 Cod za ESP32 baziran na HTTP POST i ThingSpeak server	84
5.4.3 Dvosmjerno slanje podataka	87
5.4.3.1 TalkBack API commands	88
5.5 ESP32 programski interfejs za podršku ThingSpeak in TalkBack	89
5.5.1 Program ESP32	90
5.5.2.Kod	90
5.6 Predloženi scenario kućne automatizacije	95
5.6.1 MECOnet IoT demonstration.....	99
5.6.2 Apps MATLAB Visualizations (vizuelizacija MATLAB aplikacije)	101
5.6.3 MATLAB Code	101
Glava VI	103
6.Rezultati testiranja	103
Glava VII	114
7 Zaključak	114
Glava VIII.....	115
8 Literatura	115
Bibliografija:.....	118

Glava I

1 Uvod

Iako je ideja kućne automatizacije stara više od jednog vijeka, realne pametne kuće i objekti su se pojavili u posljednje dvije decenije. Kućni aparati kao prvi uređaji kućne automatizacije datiraju od početka 20-tog vijeka, kada se pojavio prvi usisivač na električni pogon, 1907. godine. Ubrzo nastaju prvi frižideri, mašine za veš, pegle i ostali uređaji, koji imaju za cilj da olakšaju život ljudi, ne samo u njihovim domaćinstvima, već i u drugim radnim prostorima. Takvi uređaji u početku su bili privilegija bogatijih domaćinstava, a kasnije i ostalih. Prvim kućnim računarom označavamo pojavu ECHO IV računara koji je nastao krajem 60-tih godina. Ovaj pametni uređaj mogao je da izračuna listu kupovine, kontroliše temperaturu kuće, uključuje i isključuje uređaje. Suviše je bio komplikovan i nije dugo trajao. Sve do kraja 90-tih godina i početka 2000-tih kućna automatizacija je bila pretežno diktirana u dva pravca, povećanja sigurnosti (alarmni sistemi) i sistemi regulacije potrošnje energije (termostatskog tipa). Treba istaći i da sistemi gerantologije ili staranja o starijoj populaciji ili osjetljivim grupama smatraju se dosta uspješnom primjenom kućne automatizacije.

Današnje pametne kuće više se bave sigurnosnim i zelenim tehnologijama. Oni teže da budu održivi sistemi, pomažu da naši domovi ne troše nepotrebnu energiju, da nam olakšaju planiranje vremena i automatizuju proces življenja, ali ne na uštrb kvalitetu života. Takođe, pomažu da ostvarimo stepen zaštite kroz različite sisteme obezbedjenja. Trenutni trendovi u automatizaciji kuća uključuju daljinsko upravljanje mobilnim uređajima, automatizovana svijetla, automatizovano podešavanje potrošnje energije, uređaje za zakazivanje, obaveštenja o mobilnom telefonu / e-pošti / tekstu, daljinski video nadzor i niz drugih aplikacija. Intenzivni razvoj kućne automatizacije počinje razvojem Interneta i savremenih žičnih i bežičnih komunikacija i podržan je softverima različite namjene i složenosti. Praktično ne postoje savremene tehnologije koje se ne koriste u kućnoj automatizaciji ili pametnim kućama.

Kućnu automatizaciju trebamo posmatrati sa više aspekata: potreba korisnika, izazova projektanta-inženjera, cijene koštanja i pouzdanosti rada. Sa stanovišta projektanta sistemi kućne automatizacije su izazovan problem u pogledu izbora tehnologija koje će se primijeniti, njihove integracije. Npr. U nekim slučajevima jednostavnije je koristiti proste tehnologije prekidačkog tipa, nego internet bazirane tehnologije. U drugim situacijama, sistem može biti projektovan, jedino upotrebom složenih tehnologija, koje sa druge strane komplikuju upotrebu. Cilj ovog rada jeste predlaganje optimalnog rešenja sistema kućne automatizacije sa stanovišta hardvera i softvera, uzimajući u obzir zahtjeve i mogućnosti korisnika. Akcenat se daje na senzoriku

sistema kućne automatizacije, jer ona predstavlja glavnu komponentu u projektovanju sistema. Jednostavno, na osnovu raspoložive i najjeftinije tehnologije treba predložiti odgovarajući sistem kućne automatizacije koje će biti što je moguće jednostavnija i pouzdaniji za kornika.

Doprinos teze treba tražiti u nekoliko stručnih i istraživačkih pravaca.

- Odgovarajuća profesionalna znanja iz oblasti kućne automatizacije su sumarizovana u pogledu topologija, tehnologija, povezivanja, oblasti primjena i protokola.
- Predložen je i opisan Integrirani sistem kućne automatizacije koji predstavlja široku platformu sa više mogućih scenario upotrebe senzoričke, hardvera i softvera.
- Predložen je sistem kućne automatizacija baziran na IoT principu koji u kombinaciji sa klasičnim i integriranim sistemima može predstavljati optimalno rešenje u velikom broju slučajeva.
- Data su odgovarajuća hardverska i softverska rešenja IoT sistema kućne automatizacije sa primjerima izbora hardvera, projektovanja softvera i njihove integracije.
- Izvršeno je testiranje predloženog IoT sistema kućne automatizacije na konkretnom primjeri i preliminarni rezultati testiranja elaborirani i diskutovani.
- Veliki broj senzorskih, hardverskih i softverskih rešenja je demonstriran u svrhu korišćenja sa stanovišta projektanata sistema kućne automatizacije.
- Ostala znanja vezana za sisteme kućne automatizacije su sistematizovana.

Teza je organizovana na sledeći način.

U Glavi II se daje pregled kućne automatizacije. u pogledu pregleda i prednosti, sistema upravljanja, topologija, načina povezivanja, medijuma prenosa (žičani, bežični) kao i konkretnih primjera upotrebe.

U Glavi III se opisuje Integrirani sistem kućne automatizacije, koji se detaljno opisuje sa stanovišta Micro_Lan-a, RF, Bluetooth, Infra-red, Zig-Bee, senzora, ulaznih kola, kola za kondicioniranje, mikrokontrolerskih sistema, komunikacionih gateway-a (RF, ASK-RF, RS232 over RF ili BT). Zatim, pažnja se posvećuje žičanim sistemima, sa stanovištva upotrebe PLC-ova, analize impedanse, šuma i slabljenja. Elaboriraju se uobičajeni PLC protokoli X10, Lonworks, CEBus, HomePlug, CENELEC itd.

U Glavi IV detaljnije se razradjuju sistemi kućne automatizacije bazirani na IoT principu sa akcentom na na moderne sisteme kućne automatizacije koji upotrebljavaju NodeMCU, klase ESP8266 i ESP32. Njihovo povezivanje i integracija sa komercijalnim IoT platformama je takodje prikazana u ovoj glavi.

Predloženo rešenje kućne automatizacije na bazi Sistema na čipu i dostupnih IoT platformi je tema Glave V. Obradjuje se upotreba HTTP protokola u sistemima kućne automatizacije kao i slanje podataka na komercijalne platforme upotrebom ovih tehnologija. Predlaže se ESP32

programski interfejs za podršku ThingSpeak in TalkBack, vizuelizacija preko Matlaba. Opis konkretnog pilot sistema i rezultati njegovog preliminarnog testiranja daju se u Glavi VI.

Slijede Zaključak, korištena literatura i bibliografija.

Glava II

2. Kućna automatizacija-pregled

2.1 Pregled i prednosti

Kućna automatizacija je podskup automatizacije građevinskog prostora gdje se automatizacija zgrada uglavnom koristi u industrijskim i javnim objektima s ciljem smanjenja troškova energije i zaposlenih. Suprotno tome, automatizacija domova koristi se u stambenim zgradama ili se jednostavno naziva "kućama" za kontrolu, nadzor i optimizaciju električnih uređaja u domaćinstvima s naglaskom na posebne zahtjeve. Glavni ciljevi su povećati udobnost, fleksibilnost, poboljšati sigurnost, i smanjiti troškove. Udobnost se dodaje automatizacijom svakodnevnih ljudskih potreba sa lako objašnjivim i dobro dizajniranim korisničkim sistemom. Poboljšanje sigurnosti postiže se korištenjem vanjskih kamera i mehanizama upozorenja poput kontakata vrata ili prozora kako bi se pravovremeno reagovalo u slučaju pokušaja provale. Sigurnošću se bave senzori koji otkrivaju opasnost po život i imovinu, na primjer detektori dima. Cilj "niski troškovi" ima dva značenja. Prvo, troškovi za kupovinu i instalaciju sistema trebali bi biti niski tj dostupni po povoljnim cijenama. Drugo, pametnim upravljanjem potrošnje energije smanjuju se računi koji se moraju platiti. [1]

Neki tipični zadaci u sistemu kućne automatizacije su, na primjer, upravljanje svjetlima, roletnama, hladjenjem i grijanjem, kao i nadzor temperaturnih senzora, detektora dima ili kontakata prozora. Osnovna riječ "pаметan" u sistemu kućne automatizacije označava da se iza sistema nalazi inteligentna kontrola koja može pokrenuti određene radnje na predstojećim događajima, na primjer otvoriti roletnu i upaliti svjetlo kad neko uđe u sobu. [2]

Korisnički pristup ili kako je opšte prihvaćeno "vizualizacija" kućne automatizacije treba biti intuitivan i jednostavan način za upotrebu. Stanovniku omogućuje pristup svim uređajima putem zajedničkog korisničkog interfejsa. Takođe postoji mogućnost daljinskog upravljanja kućnim funkcijama putem Interneta bilo gdje i bilo kada se korisnik nalazio, na primjer s mjesta odmora putem računara ili pametnog telefona.

2.2 Sistem i upravljanje

Tipični sistem automatizacije kuće sastoji se od senzora, aktivatora i hardverskih kontrolera. U osnovi postoje dva različita načina upravljanja sistemom kućne automatizacije, centralizovana i

decentralizovana kontrola, ali takođe su moguća i neka kombinovana rješenja. Struktura sistema određuje gdje se nalazi inteligencija, što znači da je glavni dio upravljačkog softvera smješten u jednoj komponenti, više njih ili u svim komponentama. U sljedećem odjeljku međusobno se upoređuju različite strukture i raspravlja se o prednostima i nedostacima. [2]

2.2.1 Centralizovana kontrola

U centralizovanom sistemu samo jedna komponenta sadrži inteligenciju i generiše akcije. Signali senzora (prekidač, senzor osvjetljenja, senzor pokreta, temperaturni senzor itd.) šalju se glavnoj komponenti i tada ona odlučuje koje radnje moraju izvršiti određeni aktivatori (zatvarač, prekidač releja, zatamnivač itd.) . Glavna komponenta je bitna za sistem i ako se pokvari ili ukloni, više ništa neće raditi. Prednost ovog sistema je u tome što senzorima i pokretačima nije potrebna velika inteligencija, a izgrađeni su jednostavno. U slučaju promjena konfiguracije, potrebno je ažurirati samo softver u glavnoj komponenti. Troškovi proširenja takve instalacije niži su nego kod decentralizovanog sistema jer moduli senzora i aktivatora sadrže manje inteligencije. Sami početni troškovi su veći zbog glavnog kontrolera .

Glavna komponenta traži senzore za nove podatke u redovnim vremenskim intervalima, što se naziva i anketiranje. Što znači da je sistem jedan master sistem. Što je više robova (slaves) u sabirnici (bus), potrebno je više vremena da se zatraži informacija od svakog priključka i da se vrati naredba . Druga mogućnost je da senzori mogu automatski slati podatke o promjeni. To bi tada bio sistem s više voditelja s prednošću što je u sabirnici manje protoka informacija. [2]

2.2.2 Decentralizovana kontrola

U decentralizovanom sistemu svaka komponenta sadrži inteligenciju i šalje naredbe pravo na sabirnicu. Pokretači preslušavaju naredbe u sabirnici i odlučuju je li im određena naredba zanimljiva kako bi izvršili radnju. Takav se sistem sastoji od uređaja za napajanje i ulaza i izlaza. Dakle, cijena malog sistema je niska, ali proširenja su većeg opsega nego kod centraliziranog sistema. Ako neka komponenta otkáže, to neće uticati na rad ostalih komponenti. Aktivnost sabirnice zavisi od broja događaja, a ne o broja senzora na sabirnici. [2]

2.2.3 Polucentralizovana kontrola

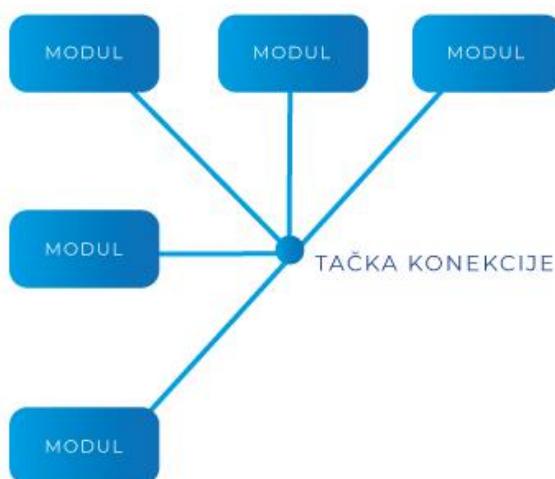
Polucentralizovani sistem sastoji se od nekoliko modula (npr. Modul za prebacivanje svjetla, modul za prigušivanje, modul okidača, razni moduli senzora, itd.) , koji su samostalni i u njima je smještena inteligencija. Dakle, ako su ovi moduli povezani preko sabirnice, inteligencija se raspoređuje između različitih izlaznih modula i svaki izlazni modul ima samo inteligenciju koja kontroliše vlastite aktivnosti. Sistem se sastoji od podsistema s vlastitim kontrolerom. Dakle, ako jedan modul zakaže, drugi moduli i dalje mogu obavljati funkciju. [2]

2.3 Topologija

U svakom sistemu kućne automatizacije postoji određeni način ožičenja između jedinica. U praksi to znači gdje treba instalirati žice. Da bi se garantovao ispravan rad, pravila se moraju strogo poštovati u skladu s odabranom topologijom, znači svaka žica ima svoju poziciju.

2.3.1 Topologija zvijezda

U topologiji zvijezda svaki modul ima vlastitu žicu do centralne tačke povezivanja (slika2.1).

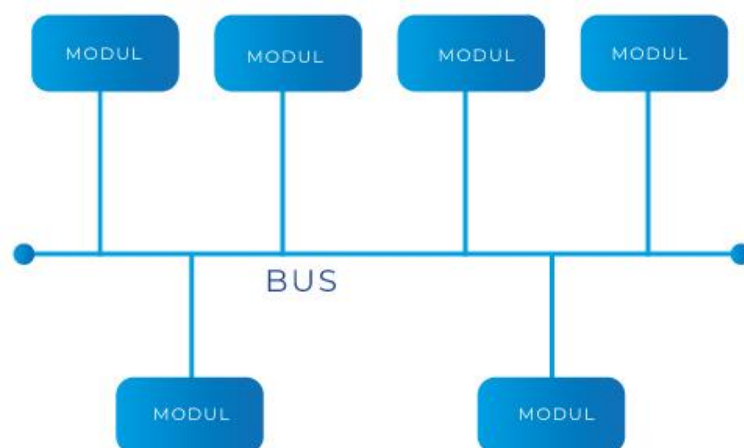


Slika 2.1: Topologija zvijezda

Prednost u tome što ako je jedna žica prekinuta modul zakačen na njoj neće raditi, ali svi ostali moduli hoće. Nedostatak je što je potrebno puno ožičenja i što ima puno veza na centralnoj tački. [2]

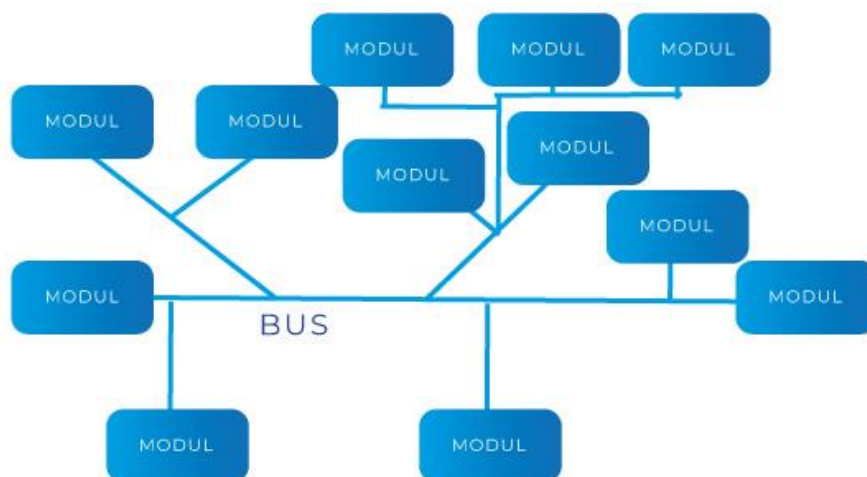
2.3.2 Topologija sabirnice

U topologiji sabirnice žica vodi od jednog modula do drugog. Odvajanje do više modula nije dopušteno (slika 2.2). Često je potreban završni otpor na krajevima sabirnice kako bi se izbjegle refleksije što dokazuje dosadašnja praksa. Žičane instalacije su mnogo manje dužine u odnosu na topologiju zvijezde, ali ako se sabirnica prekine, više od jednog modula više neće raditi. [2]



Slika 2.2: Topologija sabirnice

2.3.3 Topologija stabla



Slika 2.3: Topologija stabla

Topologija stabla je kombinacija topologija zvijezda i sabirnice (slika 2.3). Program instalacije može napraviti bilo koju vrstu grana, osim petlji. Rezultat ovakve topologije je velika fleksibilnost. Ako se žičana veza prekine više od jednog modula neće raditi. [2]

2.4 Međusobno povezivanje

Puno je mogućnosti kako se sistem kućne automatizacije u praksi realizuje za prenos naredbi i podataka između različitih komponenti. U osnovi se pravi razlika između "žičanih" i "bežičnih" sistema. U sljedećem odjeljku procjenjuje se nekoliko tehnologija sabirnica s obzirom na njihova svojstva.

2.4.1 Žičani sistemi

Na sabirnici zasnovanoj na žici podaci se prenose preko jedne ili više žica. Postoji nekoliko standardizovanih sabirnica posebno razvijenih za automatizaciju objekata. Postoji i puno sabirnica koji nisu i razvijene u tu svrhu, ali se lako mogu prilagoditi aplikaciji kućne automatizacije i samim tim su često u upotrebi. U sljedećem odjeljku razmatraju se najpopularnije tehnologije žičane sabirnice.

2.4.1.1 KNX / EIB

KNX je popularni standardizirani mrežni komunikacijski protokol za automatizaciju zgrada koji se temelji na proširenom EIB-ovom komunikacijskom snopu. KNX podržava različite komunikacijske medije poput ožičenja upletenih parova (TP), mrežne mreže, radio (KNX-RF), infracrvenu vezu i Ethernet (KNXnet / IP). Zbog standardizacije KNX je nezavistan o bilo kojoj određenoj hardverskoj platformi i postoji puno proizvođača u takozvanoj KNX asocijaciji.

Medij koji se najčešće koristi je upletena parica (dvije žice) s diferencijalnom signalizacijom brzinom od 9600 bit / s. Dvije dodatne žice (napajanje i uzemljenje) napajaju sabirnicu s 30 V DC. KNX sabirnica je višestruka glavna sabirnica u kojoj je kontrola pristupa medijima CSMA / CA (Carrier Sense Multiple Access with Collision Avoidance) i tako omogućuje dužinu fizičkog segmenta do 1000 m sa slobodno odabranom topologijom, ali petlje nisu dopuštene. KNX takođe implementira korekciju greške s ponovnim prenosom kako bi osigurao tačan prenos podataka. Za povezivanje sa KNX sabirnicom potrebni su posebni moduli interface, jer KNX standard nije dostupan u javnosti. Ti su moduli skupi i ponekad rijetko dostupni. [3]

2.4.1.2 RS-485

RS-485 (poznat i kao EIA-485) standard je interface za diferencijalni i serijski prenos podataka s visokom otpornošću na elektromagnetske smetnje. Dvije žice koriste se za prenos jednog bita s poludupleksnim prenosom. Korištenjem četiri žice moguć je i full-duplex prenos. Na strani prijemnika tumači se razlika ove dvije žice. Prednost ove tehnologije je što interferencije uobičajenog načina rada nemaju uticaja. Pa je u bučnim okruženjima moguća udaljenost do 1200 m. [4]

Zbog činjenice da RS-485 samo određuje fizička svojstva, korisnik se mora pobrinuti za ispravljanje grešaka i protokol koji bi mogao biti vrlo specifičan za aplikaciju. Kada se koristi multi-master sistem, aplikacija mora da vodi računa i o kontroli pristupa sabirnici. Moguće su brzine prenosa podataka do 12 Mbit / s, a IC-ovi primopredajnika su jeftini. Neke tipične primjene za RS485 su automatizacija zgrada, upravljanje svjetlom na pozornici (DMX512).

2.4.1.3 CAN sabirnica

Mreža upravljačkog područja (CAN) prvobitno je razvijena 1983. godine u firmi Robert Bosch GmbH sa ciljem smanjenja ožičenja za međusobno povezivanje upravljačkih i senzorskih uređaja u vozilu. Danas se CAN koristi i u mnogim drugim primjenama poput industrijske ili medicinske automatizacije. CAN je dvožična diferencijalna i sabirnica polja zasnovana na porukama gdje je cijeli protokol, uključujući kontrolu pristupa medijima i zaštitu od grešaka, već implementiran u hardver. Ne mora da se brine o sudarima na sabirnici ako dva čvora istovremeno šalju informaciju. CAN se obično definiše za brzine prenosa podataka od 10 Kbit / s do 1 Mbit / s, što odgovara udaljenostima od 5 km do najmanje 40 m. Zbog ovih svojstava, ova sabirnica savršeno odgovara za prenos podataka u bučnom okruženju s otpornošću na elektromagnetske smetnje uobičajenog načina rada. Cijena CAN hardvera je prihvatljiva. Na tržištu je moguće pronaći puno mikrokontrolera s integrisanim CAN modulima.

2.4.1.4 1-žična sabirnica

1-Wire je jednožična komunikacijska sabirnica uređaja koju je razvio Dallas Semiconductor za povezivanje senzora na jednostavan način. Riječ je o sabirnici male brzine u kojoj svaki uređaj ima jedinstveni 64-bitni identifikator. Za anketiranje uređaja radi dobijanja podataka potreban je jedan glavni sabirnik. 1-žična sabirnica jeftin je i jednostavan način povezivanja senzora i aktera, ali budući da postoji samo jedna žica, elektromagnetska izobličenja mogu lako poremetiti

komunikaciju preko dugih žica. Ovaj ga nedostatak čini neprikladnim za sistem kućne automatizacije jer je tamo pouzdanost jedan od glavnih aspekata. [5]

2.4.1.5 Ethernet

Takođe postoji mogućnost korištenja standardne Ethernet mreže, kako se koristi u LAN-u (Local Area Network), za međusobno povezivanje različitih komponenti koje se koriste u sistemu kućne automatizacije. Ethernet je multi-master sistem. Kao takav pruža veliku propusnost koja je savršena za streaming aplikacije poput velikih video ili audio signala. Ipak, za prenos jednostavnih upravljačkih naredbi ili podataka senzora, previše je režijskih troškova uzrokovanih TCP / IP stack što je potrebno za postavljanje veze i prenos podataka. Pa je potrošnja energije uzrokovana Ethernet-ovim primopredajnicima, kao i složenost hardvera i softvera prilično velika. [6]

2.4.1.6 Powerline

Powerline opisuje tehniku korištenja postojećih dalekovoda instaliranih u zgradi za prenos podataka. Pomoću posebnog adaptera, koji se naziva i modem napajanja, podaci se modulišu na napajanje, a podaci se primaju demodulacijom signala s napajanja. Tipična frekvencija modulacije je 2-30 MHz. Te visoke frekvencije na liniji napajanja mogu rezultovati poremećajima ostalih usluga u istom frekvencijskom opsegu. Svjetiljke za uštedu energije ili prebacivanje napajanja mogu negativno uticati na podatkovni signal na liniji napajanja. Za prenošenje signala sabirnice na sve tri faze električnih instalacija potreban je uređaj za fazno spajanje. Da bi se osigurala sigurnost podataka, treba koristiti šifrovanje podataka jer su podaci dostupni na svakoj utičnici. Primjer za implementaciju protokola kućne automatizacije koji se temelji na elektroenergetskoj komunikaciji je X10. [7]

2.4.2 Bežični sistemi

Kada se koristi bežični sistem, potrebe za žičanom instalacijom ne postoje što je prednost ako postoji postojeća zgrada koja bi nakon toga trebala biti automatizovana. Ipak, sigurnost je vrlo važan aspekt u sistemu kućne automatizacije i bežični sistemi mogu se napadati i ugroziti bez dodirivanja žica.

2.4.2.1 WLAN

Bežična lokalna mreža (WLAN) standard je lokalne radio mreže. Slično Ethernetu, režijski troškovi su ogromni u poredjenju s propusnošću koja se koristi za prenos podataka senzora i upravljačkih naredbi koji se koriste u sistemu kućne automatizacije. Energija koja se koristi za prenos podataka prilično je velika. WLAN je prikladan, na primjer, za audio ili video streaming.

2.4.2.2 ZigBee

ZigBee je standard komunikacijskog protokola za radio uređaje male snage koji se temelji na IEEE 802 standardu. Dizajniran je za jedinstvene potrebe jeftinog bežičnog senzora i upravljačke mreže male snage. Standard je razvio ZigBee Alliance², a u izradi je učestvovalo više od 230 proizvođača. Razvoj još uvijek pokreće ovu organizaciju, a proizvodi ZigBee već se nalaze na tržištu. Postoji mnogo aplikacija za ZigBee koje se mogu koristiti za automatizaciju zgrada. Senzori, aktuatori i kontrola grade skup mreža u cijeloj zgradi. [8].

2.4.2.3 EnOcean

EnOcean je bežična tehnologija za prikupljanje energije koju je razvio EnOcean Alliance³ s mnogim uključenim proizvođačima. Prikupljanje energije tehnika je u kojoj malu količinu energije koja je potrebna za prenos kratkog podatkovnog okvira generišu elektromagnetski generatori, piezo generatori, solarne ćelije ili termo spojnice. To znači da većina komponenti može raditi bez baterije što ih čini bez održavanja. Primjene za ove uređaje su, na primjer, prekidači, kontakti za prozore ili drugi senzori koji bežično prenose svoje podatke na baznu stanicu. Nedostatak ove tehnologije je nedostatak sigurnosti. Svi okviri mogu se jednostavno snimiti prikladnim prijemnim uređajem. [9].

2.4.2.4 IrDA

Mnogi uređaji s daljinskim upravljanjem koriste prenos podataka infracrvenim povezivanjem (IrDA). Stoga bi moglo biti prikladno upravljati tim uređajima infracrvenim primopredajnikom povezanim s upravljačkom jedinicom kućne automatizacije. Ipak, udaljenost i brzina prenosa podataka su ograničeni. Budući da je kompletan sistem automatizacije kuće raspoređen po

cijeloj zgradi, nemoguće je povezati sve komponente optičkim pristupom. U svakom slučaju, IrDA se i dalje može koristiti za lokalno upravljanje uređajima.

2.5 Zadaci

Ovaj odjeljak opisuje tipične zadatke sistema kućne automatizacije. Naravno, postoji više mogućih zadataka, ali fokus je na onim najvažnijim.

2.5.1 Osvjetljenje

Uključivanje ili prigušivanje svjetla osnovna je funkcija svake električne instalacije u kući. Ipak, njime upravlja i sistem kućne automatizacije, ali s tom razlikom što se vrijednost može dodati određenom prekidaču za svjetlo. Na primjer, ako osoba uđe u dnevnu sobu i želi da odgleda film pritiskom na tipku, sva svjetla se prebacuju ili prigušuju na željeni nivo osvetljenja za gledanje filma. Još jedan primjer bio bi taster kad izlazimo iz kuće gdje se sva svjetla isključuju pritiskom samo jednog prekidača. Dodavanjem senzora intenziteta svjetlosti i detektora pokreta, osvjetljenje u sobi može se prilagoditi ukupnim intenzitetom svjetlosti ili prisutnošću osobe.

Automatizacija svjetla: Kada je riječ o automatizaciji svjetla, postoje dvije mogućnosti. Prvo je instalacija pametnih prekidača za svjetlo. Pametnim prekidačima svjetla ne samo da se može upravljati pomoću aplikacije, već mnogi nude razne mogućnosti osvetljenja. Druga opcija pametnog osvetljenja su pametne sijalice. Pametne sijalice pretvaraju obična svjetla u pametna i nude više mogućnosti osvetljenja, poput prigušivanja ili nježnog svjetla, ali su skuplje jer im je potrebna kupovina pretvarača za rad sa pametnim uređajima.. Zamjenom starih sijalica za pametne može se automatizovati osvetljenje.

2.5.5 Audio i Video

Izvori zvuka i slike mogu se distribuirati u jednu ili više prostorija. U takvom sistemu sa više soba može se odabrati kanal, a jačina zvuka u svakoj sobi može se kontrolisati posebno. Tipični audio izvori su internetski radio, mp3, CD itd., A tipični video izvori su streaming putem interneta, DVD itd

Pametni zvučnici: Skoro svaka porodica u kući ima zvučnik. Gotovo je nemoguće današnji život bez zvučnika. Pametni zvučnici više ne služe samo za slušanje muzike, iako to osnovna

funkcija. Mogu pretraživati internet, javljati vrijeme, objavljivati vijesti, raditi kao personalni asistent i mogu biti centralni kontrolni sistem cijelog sistema Smart-Home. S pravim pametnim zvučnicima, gotovo svim aspektima kućne automatizacije može se kontrolisati samo glasom, pitajući Alexa¹ koja je na ulaznim vratima, upotrebom Siri² da se ugasi svjetlo koje ste ostalo upaljeno na spratu ili smanjiti grijanje s Googleom. Kada se pametni zvučnik želi koristiti kao čvorište za automatizaciju kuće, treba provjeriti jesu li uređaji koji se kupuju kompatibilni s zvučnikom, pošto naravno nijesu svi.

Pametni televizori: Pametni televizori izvrsni su jer je osnovi moguće gledati sve što se želi, kad god se želi, bez obzira ima li se kablovska ili ne. Pametni TV obično se odnosi na sam televizor, ali u ovom slučaju to je krovni naziv za sve IoT uređaje za streaming videozapisa. U današnje vrijeme postoji mnoštvo opcija kada je u pitanju streaming TV. Postoje doslovno pametni TV koji emituje Netflix, Hulu, Amazon itd. Direktno sa same jedinice ili se može dobiti sistem za uparivanje s normalnim televizorom. Mnogi od ovih uređaja mogu se umrežiti sa pametnim zvučnicima i centralnom kontrolnom jedinicom..

Video zvona na vratima: Video zvona ne samo da osjećaju i upozoravaju kad je osoba na ulaznim vratima, već mogu pokazati i video snimke ko je tamo i može koristiti interfonsku tehnologiju za razgovor sa svojim posjetiocem. Više se ne mora fizički odlaziti do vrata ili vikati "ko je tamo?" ili "uđite" s druge strane kuće tj može se vidjeti sa pametnog uređaja ili čak započeti razgovor.

2.5.2 Roletne

Za roletne koristi se sličan pristup kao i za upravljanje svjetlima. Na primjer, pritiskom na taster roletne se mogu pomjeriti u željeni položaj. U mnogim sistemima kućne automatizacije moguće je istovremeno kontrolisati nekoliko roletni, na primjer za svaku sobu ili sprat. Takva mogućnost osigurava stanovniku puno vremena svaki dan ako nije potrebno ručno kontrolisati svako zatvaranje.

Prozori za privatnost: Pametni prozori za privatnost mogu se instalirati ili biti jednostavni kada se postave preko postojećih prozora. Mogu se promijeniti iz normalnog prozora u neprovidni prozor kroz koji niko ne može vidjeti, s daljinskog upravljača ili sa pametnog uređaja. Mnogi prozori takođe štite od štetnih UV zraka i rade kao izolatori kako bi održavali temperaturu.

¹ Alex-Amazon voicebot

² Siri-iphone

Pametne roletne: Pametne roletne mogu se podizati i zatvarati sa telefona ili drugih uređaja kako odgovara želji korisnika. Mnogi se takođe mogu postaviti prema rasporedu tako da svako jutro mogu obradovati sunčevom svjetlošću u isto vrijeme.

2.5.3 Grijanje

Grijanje, ventilacija i klimatizacija kontrolisani sensorima temperature i vlažnosti često su uključeni u sistem kućne automatizacije. Zavisno o unutrašnje i spoljne temperature, grijanje se podešava tako da ima unaprijed zadatu vrijednost temperature u sobi. Na primjer, stanovnik može isključiti grijanje kad odlazi na godišnji odmor i ponovo ga daljinski uključiti nekoliko sati prije dolaska kući. To štedi novac i pomaže očuvanju životne sredine.

Pametni termostati: S pametnim termostatom, može se pratiti grijanje ma gdje god ko bio. Može se paljenje i gašenje grijanja podesiti prema rasporedu tako da se toplotna energija ne troši kad nikoga nema kod kuće. Pametni termostati imaju opciju senzora koji automatski mijenja temperaturu kad osjete da je potrebna.

Pametni utikači: Pametni utikači samo se uključe u uobičajene utičnice i tada se sve što se njima napaja može se lako uključiti i isključiti sa pametnog uređaja i sama prostorija postaje automatizovana i pametnija..

2.5.4 Sigurnost

U sistemu kućne automatizacije senzori alarma koriste se za otkrivanje promjena stanja i zatim odlučuju mora li se alarm upaliti ili ne, zavisno da li ima potrebe. Često korišteni senzori su, na primjer, magnetski kontakti za prozore ili vrata, detektori dima, detektori pokreta, senzori za lomljenje stakla itd., dovoljno za zaštitu od izbijanja požara ili neovlaštenog pristupa objektu. Jedna od mogućnosti za sprečavanje neovlaštenog ulaska od provalnika je simulacija prisutnosti. To znači da se svjetla uključuju i gase kako bi simulirala prisutnost domaćina.

Video nadzor: Praćenje nekretnina vlasništva sa pametnog uređaja tokom čitave godine, non stop, uz video nadzor koji je uvijek budan. Sigurnosni sistem zasnovan na aplikaciji omogućava da se može potvrditi u trenutku upozorenja pristupanjem snimcima na pametnom uređaju s bilo koje kamere u prostoriji.

Senzori: Senzori najčešće se koriste na vratima i prozorima koji mogu biti potencijalna ulazna mjesta u dom. Senzori koji se temelje na aplikacijama mogu na pametni uređaj poslati upozorenje da je neko koristio jedan od ulaznih otvora.

Automatske brave: Poznato je da većina provalnika ulazi kroz otključana vrata ili prozor. Upotrebom automatizovanih brava za zaključavanje vrata sa pametnog uređaja sa bilo kojeg mjesta i bilo kada može se izvršiti provjera pa čak i više puta da li je zaključano.

Alarmni sistem: Ako se dogodi najgore i desi se provala u domu ili firmi, može se postaviti alarm koji se aktivira. Većina sistema se postavlja tako da automatski upozorava.

Senzori za požar, CO i poplave: Provala nije jedina prijetnja domu ili poslu. Vatra, plin i voda takođe mogu nanijeti ozbiljnu štetu. Pametni detektori re / CO nude utišavanje aplikacije i šalju obavještenja na jedan ili više telefona ako su aktivirani. Senzori poplave rade na istom principu kao i detektori dima .

2.5.5 Malo komplikovaniji oblici pametne kuće

2.5.5.1 Pametne kuhinje

Iako nam tehnologija još uvijek nije osigurala robota kuvara, ovi pametni kuhinjski uređaji toliko olakšavaju kuvanje i čišćenje. Mnoge od njih se lako mogu povezati sa pametnim zvučnicima ili telefonom .

Pametni frižideri: Pametni frižideri imaju različite mogućnosti. Mogu da odgovore koliko se često koriste, upozore ako su vrata ostala otvorena, obavijeste koje proizvode je potrebno kupiti, a neki modeli čak imaju i TV. Pametni frižideri mogu se isporučiti s jednom od ovih mogućnosti, u kombinaciji s njima ili sa svima njima. Neki frižideri prikazuju ove podatke direktno na samom uređaju, a drugi sve podatke šalju u aplikaciju. Većina frižidera kompatibilnih s aplikacijama takođe će pokrenuti dijagnostiku kako bi se osiguralo da sve radi bez problema.

Pametni šporeti: Pametni šporeti omogućuju kontrolu nad bilo kojim mjestom. Postoje čak i pametni šporeti kojima se može upravljati putem Alexa . Neke aplikacije takođe omogućuju traženje recepata i zakazivanje vremena i temperature potrebnih za redosled kuvanja i pečenja, ali često je ipak potrebno pritisnuti start što znači da je ljudski faktor neophodan. Većina pametnih šporeta barem će dopustiti da se kontroliše tajmer i pokreće dijagnostika.

Pametne mašine za pranje sudja: Pametne mašine za pranje sudja rade na vrlo sličnim aplikacijama poput pametnih šporeta i pametnih frižidera kompatibilnih s aplikacijama. To je zato što ih proizvode iste kompanije (npr. LG, GE, Whirlpool itd.). Pomoću ovih aplikacija moguće je provjeriti status pranja, dobiti signal kad se pranje završi i zaustaviti , pokrenuti

punjenje. Slično šporetu i frižideru, većina ovih aplikacija takođe će pokretati dijagnostiku na kontrolnim jedinicama ako je potrebno.

Pametno pranje veša: Pranje veša nije nužno kućni posao, ali budući da su ove aplikacije iste kao i prethodni uređaji, ts. Pametne mašine za pranje i sušenje veša su baš poput mašine za pranje sudja. Može se pokrenuti, pauzirati, zaustaviti pranje, provjeriti status, primiti obaveštenje kada je proces završen i pokrenuti dijagnostiku vrlo paktično sa pametnog uređaja.

Pametna jela: Mogu se kupiti pametni tanjiri i viljuške koji prate unos kalorija i povezuju se s aplikacijom koja će pomoći u praćenju pravilne ishrane.

Pametni spori šporet: Upravljanje pametnim šporetom sa telefona ili tableta je takodje vrlo ostvarljivo danas. Može se provjeriti temperatura, promijeniti jelo, ili ih jednostavno uključiti i isključiti.

Pametni lonac za kavu: Većina lonaca za kavu može se postaviti na tajmer da se počne kuvati kada je potrebno. Pametnim loncima može se reći da se kuvaju sa pametnog telefona tako da kad god se želi kava može se pokrenuti lonac.

Pametne kante za smeće: Većina pametnih kanti za smeće jednostavno je bez dodira i kontroliše mirise. Postoji jedna pametna kanta koja uključuje skener bar koda. Dok se baca hrana, skenira se bar kod, a kanta za smeće dodaje je na popis namirnica na pametnom uređaju.

Robotski usisavači: Imati u kući sobaricu robota možda je još prilično daleko, ali robotski usisivač je vrlo koristan uređaj. Njim se može upravljati s pametnih uređaja, uključujući pametne zvučnike. Kao da posjedovanje robotskog usisavača nije dovoljno, neki od tih usisavača imaju vlastitu vještačku inteligenciju (AI) poput pametnih zvučnika, a nekolike vrste usisivača čak mogu prosljediti slike i videa pravo na pametni uređaj.

2.5.5.2 Pametna kupatila

Čak se i najosnovnije svakodnevene ili neuobičajene potrebe mogu poboljšati u pametnom domu. Tehnologija pametnog kupatila učinit će kupatilo više od sobe koje se moraju i inače koriste tako da se i u taj dio kuće provodi više vremena.

Pametni toalet: Pametni toalet poboljšava osnovne funkcije, senzor koji otvara i zatvara poklopac, automatsko čišćenje wc šolje. Imaju i neke drugačije funkcije, poput ugrađenog bidea, grijanih sjedala i još mnogo toga.

Pametna ogledala: Na primjer, zamislite da perete zube i gledate televiziju u ogledalu. Mogu se čak imati i pametna ogledala s osvjetljenjem aktiviranim dodiranjem i površinama bez magle.

Pametni tuševi: Pametni tuševi imaju različite mogućnosti. Mogu se uključiti iz udobnog kreveta, pripremiti savršenu postavku tuširanja, upaliti muziku ili čak televizijski program.

2.5.5.3 Pametni vrtići

Novorođenče je jedno od najboljih i najstresnijih iskustava u životu. Zahvaljujući pametnim vrtićima, bebe su sigurnije nego ikad, a roditelji se mogu malo opustiti i posvetiti sebi.

Video monitori: Savremeni dječji monitori potpuno su se promijenili od kada se video kamere mogu lako ugraditi. Sada, umjesto samo slušanja bebinog plača, beba može se vizualno provjeriti iz bilo koje sobe u bilo kom trenutku. Većina monitora se poveže sa telefonom putem aplikacije, no neki jednostavno imaju odgovarajući monitor koji se lako može premjestiti iz sobe u sobu.

Pametno ljuljanje: Na tržištu već postoji nekoliko različitih pametnih kolijevki koje se razlikuju u mogućnostima usluga koje pružaju. Najmanje što mogu je da ljuljaju na raznim brzinama kojima može upravljati pametni uređaj. Mnogi također mogu emitovati prijatan zvuk koji bi umirio bebu ili imaju ugrađeno snimanje i reprodukciju govora majke.

Pametni zdravstveni monitori: Postoji mnogo različitih monitora za bebe koji prate njihovo disanje, otkucaje srca, kretanje, pa čak ponekad i temperaturu. Jedni su postavljeni i rade u pametne jastučice i ćebad za krevetiće a drugi su prenosivi i mogu se koristiti i na putovanjima . Ovakvi monitori ne samo da pomažu u sprječavanju SIDS-a(sindrom iznanedene bebine smrti), već mogu da prate promjene pelena, navike spavanja i još mnogo toga, a zatim te podatke šalju u aplikaciju na pametnom uređaju.

Pametni stolovi za presvlačenje: Pametnom postolje za krevetac koje djeluje kao presvaka za bebe, i pametni stočić mjeri težinu bebe kako bi se osiguralo pravilno hranjenje.

Pametni mobini telefon: Pametni mobilni telefon može osjetiti u kojoj je fazi spavanja beba i reaguje na to zvukom, pokretom i svjetlošću koja umiruje dijete i omogućava lijepo spavanje. Postoji nekoliko uređaja koji koriste svjetlost, boju i zvuk za smirivanje djeteta u snu.

Senzori: Mnogi roditelji koriste senzore oko jaslica ili spavaće sobe svoje djece za dodatno obezbedjenje. Kada se senzori aktiviraju, šalju upozorenja na pametni uređaj. Mogu se postaviti ne samo na spoljnim ulaznim vratima, već i na unutrašnjim , tako da uvijek upozorava kada neko ulazi ili se čak kreće po sobi. Temperaturni senzori još su jedan pametan izbor, pazeći da u sobi uvijek bude savršena temperatura za vašu bebu.

2.5.5.4 Pametna njega kućnih ljubimaca

Svakodnevna briga o krznenom ljubimcu jednostavnija je nego ikad pomoću pametne tehnologije. Najbolji dio ovih naprava je što olakšavaju život kućnim ljubimcima bilo da su ostali sami u kući ili suvlasnici prisutni.

Video nadzor: Upotreba video monitora je korisna za praćenje aktivnosti ljubimaca kada su ostali sami kod kuće da ne pričinu neku štetu. Posebni monitori za kućne ljubimce idu u oba smjera kako bi se vlasnik i ljubimac mogli vidjeti i čuti, a neki čak dopuštaju da se dijele poslastice kao nagrada.

Pametni otirači: Mnogi vlasnici kućnih ljubimaca imaju lijep namještaj koji žele da bude sačuvan. Pametnim prostirkama ili igračkama koriste se razne tehnike kako bi kućnim ljubimcima dala jasna upozorenja da se drže izvan zabranjenih soba i namještaja. Posebno korisno za obuku novih kućnih ljubimaca ili održavanje dresure starijih tvrdoglavih.

Pametna vrata za kućne ljubimce: Pametna vrata za kućne ljubimce rade s pametnim ključem koji ide na ogrlicu ljubimca sa opcijama da otvaraju vrata za ulazak i izlazak iz kuće. Između ostalog kako ne bi neželjena bića završila u domu. Vrata se zaključavaju i kada se ne želi da ljubimac izlazi.

Pametne hranilice: Pametne hranilice automatski dijele hranu i vodu za kućne ljubimce. Pomoću aplikacije na telefonu zakazuje se hranjenje i kontroliše veličinu porcija za ljubimce.

2.5.5.5 Više pametnih opcija

Pametni ventilatori: Pametni ventilatori, poput termostata, mogu se kontrolisati iz aplikacije na telefonu, postaviti na raspored ili mogu da se upale kada su potrebni.

Pametni podovi: Podno grijanje je sve više zastupljeno i taj luksuz podova sa kontrolisanom temperaturom koji se mogu automatizovati sa pametnih uređaja postaje gotovo uobičajan u gradnji objekata. Iako se ta tehnologija razvija još nema puno proizvoda za automatizovane grijane podove, ali mnogi pametni termostati mogu se programirati i za upravljanje podnim grijanjem.

Automatizirane prskalice: Imati lijep i zelen travnjak mnogo je lakše s automatizovanim sistemom prskalica kojima upravlja aplikacija. Lako se odredi vrijeme koje prskalice i kada teba da se upale radi redovnog održavanja.

Pametni bazen: Na primjer, ako imate bazen, znate koliko mu je potrebno za održavanje. Aplikacije pametnih bazena daljinski mjere hemikalije u bazenu, omogućuju automatizaciju

određenih postupaka čišćenja i planiranje redovnog održavanja bazena , temperature tj. spremnosti za trenutnu upotrebu.

Pametni kreveti: Pametnim madracem kontroliše se položaj, mekoća, pa čak i toplina kreveta. Neki madraci imaju dodane pogodnosti, poput postavki masaže, ugrađenih budilica i ugrađenog nadzora spavanja koji šalje informacije o navikama spavanja (nemiran san, hrkanje) kako bi pomogao da se ima što bolji odmor od svakodnevnih aktivnosti .

2.5.5.6 Kućna automatizacija za medicinsku njegu

Pomoć voljenima da vode svakodnevno nezavistan i siguran život.

Mnogo porodica se same brinu o ostaralim roditeljima i to je vrlo teška i odgovorna aktivnost.. Njega u kući može biti vrlo skupa, a ustanove za starije osobe sigurno oduzimaju nezavisnost stare osobe i osjećaj ostavljenosti od porodice i tako dovode do novih problema.

Kućna automatizacija nudi pristupačne alternative koje omogućuju mir i nezavisnost i osjećaj sigurnosti ostarale osobe, praćenje uz pomoć aplikacija što manje ili onoliko koliko se smatra potrebnim. Neke od mogućnosti za medicinsku njegu automatizacije kuće su:

Tasteri za medicinsko upozorenje: Kod starijih osoba se često desi da izgube ravnotežu, padnu i da ne mogu ustati . Dugmad za medicinska upozorenja su lagana , vodootporna , koja osoba može nositi oko vrata, na ruci ili samo nositi pri ruci. Dugme se spaja na interfonski sistem koji kada se aktivira pritiskom, poziva medicinski nadzorni centar. Ako je potrebno, neko iz pozivnog centra može da upozori koga treba ili hitnu pomoć za pacijenta. Neki tasteri su osjetljivi na pada te mogu čak osjetiti kada je korisnik pao i automatski upozoriti call centar.

Video nadzor: Video nadzor omogućuje snimak osobe kojoj je potrebna medicinska njega iz praktičnosti pametnog uređaja. Video nadzorom može se steći mir i sigurnost, znajući da je posmatrana osoba sigurna kod kuće.

Praćenje aktivnosti: Postoji više različitih načina za praćenje aktivnosti i ustaljenih radnji voljene osobe. Dozatori lijekova mogu da upozore kada je pacijent uzeo ili nije uzeo dnevne doze. Veza sa frižiderom putem senzora upozorava da li osoba redovno uzima jelo ili pije potrebne tečnosti. Takođe se mogu nadzirati ostale kućni uređaji u smislu sigurnosti da vrata nijesu ostala otvorena, da šporet nije uključen ili da slavina ne radi.

Sigurnost za bolesnike s Alzheimerovom i demencijom: Većina oboljelih od Alzheimerove i demencijske bolesti odluta daleko od svog mjesta stanovanja i poznate životne sredine. Kad se to dogodi, to je zastrašujuće i vrlo teško iskustvo za njih i za članove porodice . Kućni

sigurnosni sistemi mogu pomoći da se to ne dogodi, prateći oboljelu osobu i obavještavajući na pametnom telefonu .

2.5.6 Je li automatizacija kuće sigurna?

Ne može seni za jednu tehnologiju da je 100% sigurna. Automatizacija kuće nije ništa drugačija od ostalih računarskih sistema. Potrebno je razmisliti o mnogo stvari prilikom ulaganja u automatizaciju kuće i biti pripremljen da neće sve uvijek biti kako treba.

Ako nisu isključeni, pametni zvučnici uvijek slušaju.

Privatnost: Kada je riječ o pametnim kućama, mora se uzeti u obzir činjenica da tako povezana firma kompanijama daje mogućnost prikupljanja podataka o njoj samoj. uvijek će biti krivac pametni zvučnik. Mnogi pametni zvučnici ostaju stalno uključeni, što znači da stalno slušaju. Oni pamte i uče iz svega što se izgovori, kako bi bolje zadovoljili potrebe samog korisnika, ali i da bi prodali više proizvoda. Na primjer, ako pitate Alexa nešto za mašinu za pranje veša, slijedeći put kada budete surfovali webom dobićete mnoštvo oglasa za mašine za pranje veša. Isto tako, ova vrsta upada nije jedinstvena pametnim zvučnicima, pa čak ni asistentima poput Alexa ili Siri. To se događa danas i kada se korisniku usluga nešto sviđa na Facebooku, potražuje nešto na googleu ili koristi svoju debitnu ili kreditnu karticu za kupovinu. Danas je takva umreženost sa svim i svakim, sve su prikladnije usluge ne samo za korisnika usluga, već i za kompanije koje pokušavaju prodati sve i svašta. Sve dok se na to trezveno gleda uz odgovarajuću zaštitu može se uživati u pogodnostima pametnog doma.

Cyber sigurnost : Cybersecurity vrlo je važno pitanje možda i najozbiljnije koje mnogi ljudi ne shvataju da bi se tako odnosili prema njemu. Tržište Interneta stvari (IoT) stalno raste i razvija se, velikom brzinom, a samim tim i stariji uređaji postaju ranjiviji i manje zaštićeni. Važno je da se korisnik odnosno potrošač mora se naučiti zaštititi , u slučaju da programeri to ne mogu. Prvi korak je prije svega informisanost . Pročitati upustva na uređajima prije nego se donese odluka o kupovini. Upoznati se sa sigurnosnim prijetnjama s kojima bi se mogli suočiti i koje odgovornosti je programer za internetske stvari preuzeo na sebe kada je u pitanju zaštita interesa potrošača. Mrežna sigurnost još je jedan važan dio zaštite cyber sigurnosti pametnog doma. Započnje se sa šifrovanjem lozinki, najvažnijim dijelom mrežne sigurnosti. Potrebno je koristiti jake lozinke sa više različitih znakova i često ih mijenjati.

Ista preporuka za lozinku vrijedi za bilo koji alarmni sistem Touch podloge. Kad je riječ o odabiru lozinke, što više znakova je najsigurnija i tu nema dileme . Čak i ako se čini jedinstvenom i vrlo pametnom, ako se lozinka temelji na bilo kakvim ličnim podacima, neki lopov upoznat sa životom i navikama vlasnika mogao bi lako da je otkrije. Slijedeći korak je

osigurati da su računari, tableti i telefoni zaštićeni firewalls , antivirusnim softverom i da uvijek ažuriraju svoje operativne sisteme kako bi se pratio razvoj softvera.

I na kraju, važno je da distributeri uređaja mogu kontaktirati korisnika ako postoje problemi s njihovim proizvodima. To nije problem ako se kupuje direktno od distributera (npr. Nest, Phillips itd.), Ali ako se kupuje preko treće strane (npr. Target, BestBuy itd.), mora se i treba ići na mrežu i registrovati svaki kupljeni uređaj. Razlog zbog kojeg je važno biti dostupan tim kompanijama jest taj da mogu kontaktirati direktno ako ikad povrijede sigurnost, povuku proizvodnju ili promijene nešto bitno na svom proizvodu.

Prekidi napajanja: Bilo koji sistem koji se pokreće putem Interneta neće da radi kad nestane struje. Sistemi kućne sigurnosti i medicinske uzbune koji imaju rezervne baterije za nestanak struje i prolaze kroz telefonske linije bit će u radu, ali ako i telefonske linije ne rade, oni će takođe biti u kvaru. Bilo koji sistem trebao bi biti siguran u bilo kojoj situaciji, pod uslovom da ima dovoljno spremljene rezervne izvore električne enrgije.

Nereagirajuće žrtve: Pitanje koje se često postavlja jeste što će se dogoditi da korisnik tastera za medicinsko upozorenje ne reaguje i / ili ne može ostvariti komunikaciju u hitnim slučajevima. Većina pružaoca medicinskih usluga sa korisnikom mora imati gotove planove za hitne slučajeve, šta i kako se ponašati u takvoj situaciji.

2.5.7 Praćenje

Potrebno je stalno nadgledati i vizualizirati podatke senzora korisniku. Tako će se uvijek imati uvid u dešavanja na samom sistemu kućne automatizacije bez obzira za koju se namjenu koristi.

Zanimljivi podaci senzora mogu biti:

- Temperatura solarnog kolektora
- Temperatura vode grijanja
- Potrošnja energije
- Brzina vjetra
- Jačina svjetlosti

Glava III

3. Integrirani sistem kućne automatizacije

3.1 Uvod

U ovoj glavi biće opisan Integrirani Sistem Kućne Automatizacije (ISKA). ISKA predstavlja široku KA platformu sa nekoliko mogućih scenaria:

MicroLAN mrežu sa potrebnim interfejsima.

WSN mrežu sa potrebnim interfejsima.

«Power line» komunikacionu mrežu (PLK).

Koji će od datih scenarija biti upotrijebljen zavisi od više faktora koji se definišu samim projektnim zadatkom, proisteklim iz konkretnih potreba investitora kao i infrastrukturnim i materijalnim resursima. Kako ISKA predstavlja generalnu platformu čija je obrada veoma zahtjevan zadatak u okviru jednog rada, njen opis je dat u vidu generalne arhitekture dok su detaljnije obradjena rešenja vezana za WSN i PLK.

Micro_LAN je dat u okviru opisa 1_Wire mreže na primjeru korištenih temperaturnih senzora u sklopu predloženog «smart senzora» na bazi PLK.

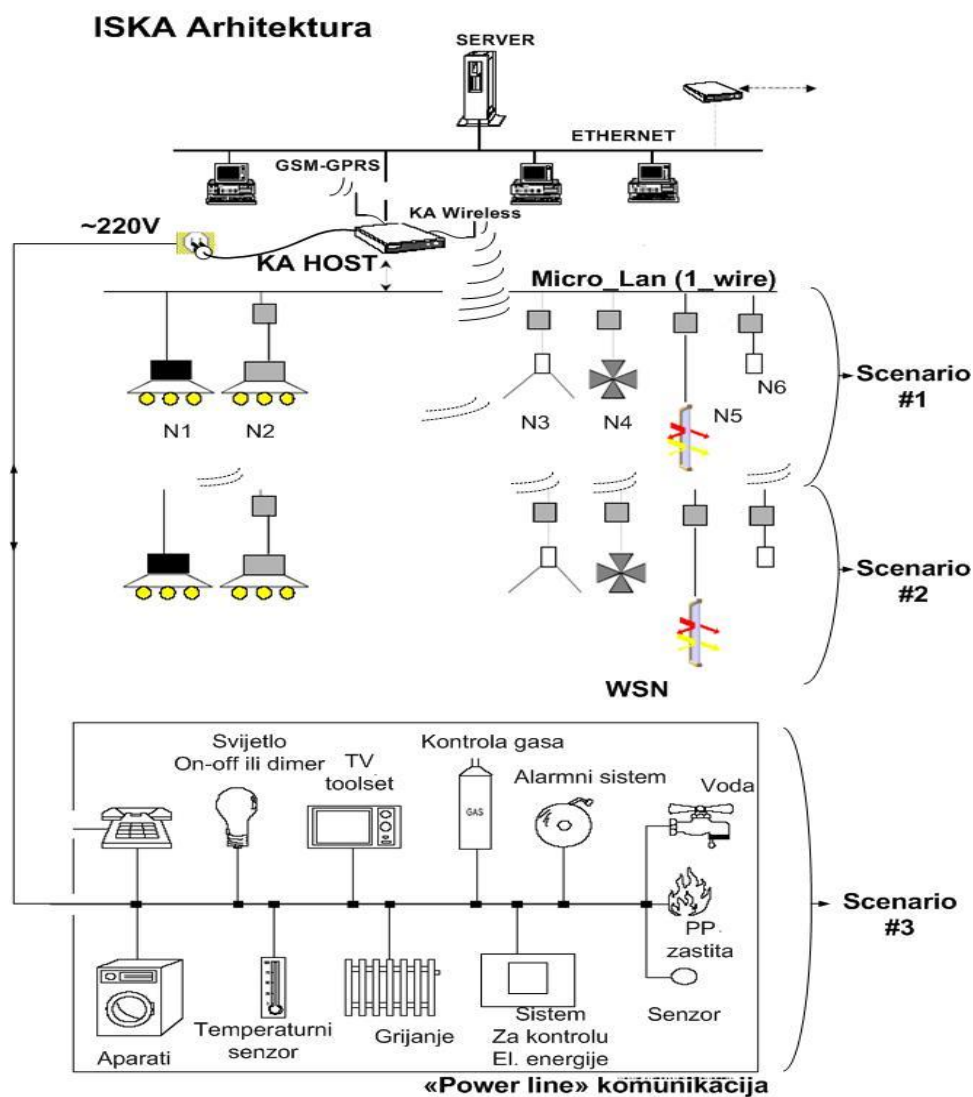
U okviru WSN biće detaljno obradjena rešenja WS predajnika i WS prijarnika kao i WS primo-predajnika bazirana na AM i FM tehnikama, dok će se u kratkim crtama dati i BlueTooth konfiguracija. Pored arhitekture i konkretnih šema predajnika i prijarnika prezentira se i njihovo povezivanje na PC konfiguracije kao i mikrokontrolere opšte namjene kao što su 8 bitni AVR RISCovi.

PLK se obradjuje na nivou najnovijih rešenja raspoloživih u ovoj oblasti baziranih na ST seriji modema . Ovdje se daju arhitektura i detalji PLK predajnika i prijarnika opšte namjene kao i način njihovog povezivanja na PC i mikrokontrolerske konfiguracije. Elaboriraju se konkretni primjeri ON-OFF temperaturne i svjetlosne regulacije pogodne za potrebe povećanja energetske efikasnosti. Takodje se diskutuju osnovna softverska rešenja za testiranje PLK sistema.

Pored hardverske i softverske arhitekture o okviru ove glave se daju i neki od rezultata preliminarog testiranja prvenstveno u okviru provjere funkcionalnosti pojedinih komponenta. Data rešenja se mogu i parcijalno upotrijebljivati u okviru djelimičnog sistema kućne automatizacije.

3.2 Arhitektura predloženog rešenja

Kada se govori o savremenom sistemu kućne automatizacije mora se uzeti u obzir njihova integrisanost i kompatibilnost sa postojećim tehnologijama. Takav sistem treba da je umrežen u LAN i WAN okruženje kao i da posjeduje mogućnost autonomnog rada. Jedno od mogućih rešenja integrisanog sistema je dato na slici 3.1.



Slika 3.1: ISKA arhitektura, N označava "node" ili "slave".

Kao što se vidi, glavnu komponentu predstavlja host kućne automatizacije označen kao KA HOST. Njegova funkcija je višestruka:

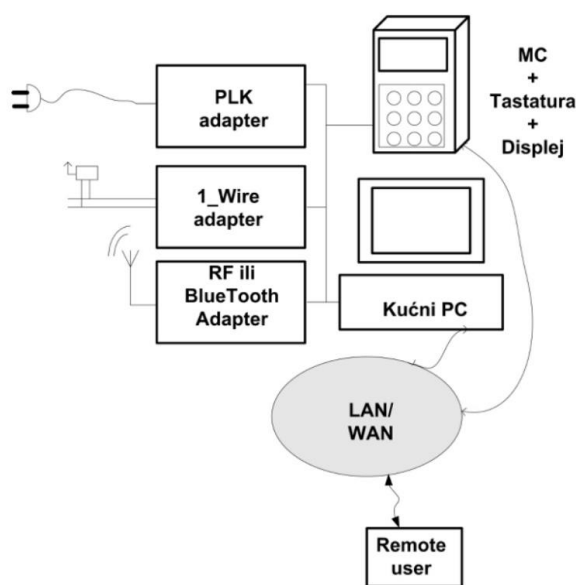
Omogućava konekciju prema LAN i WAN okruženju kako bi se moglo upravljati sistemom kućne automatizacije posredstvom Interneta.

Omogućava konekciju prema GSM i GPRS domenima kako bi se moglo upravljati sistemom kućne automatizacije primjenom mobilne telefonije.

Omogućava komunikaciju unutar komponenti sistema kućne automatizacije po jednom od scenarija implementaciju gdje je uključen i odgovarajući protokol komunikacije.

Posjeduje mogućnost autonomnog rada što podrazumijeva mogućnost direktnog setovanja parametara, posredstvom aplikacije, i samo-regulaciju unutar Sistema . Ovaj slučaj je koristan u vreme prekida WAN ili LAN konekcije

Ulogu KA HOST u opštem slučaju može preuzeti kućni PC što je u suštini najekonomičnije rešenje. Tada se na njega povezuju različiti adapteri kojim se implementira odgovarajući protocol, slika 3.2. Umjesto PCja može se upotrijebiti i microcontroler opremljen tastaturom i displejom što je nepovoljnije rešenje zbog teškoća u spajanju MC baziranog sistema na unutrašnju i spoljašnju mrežu.



Slika 3.2 Rešenje KA hosta

Tokom našeg projekta korišten je PC računar kao KA host zbog pogodnosti koje pruža u procesu razvoja prototipova kao u snimanju značajnih rezultata prilikom eksperimentisanja. U nastavku se daje kratak opis Micro_Lan scenario i detaljnije se elaboriraju WSN i PLK rešenja.

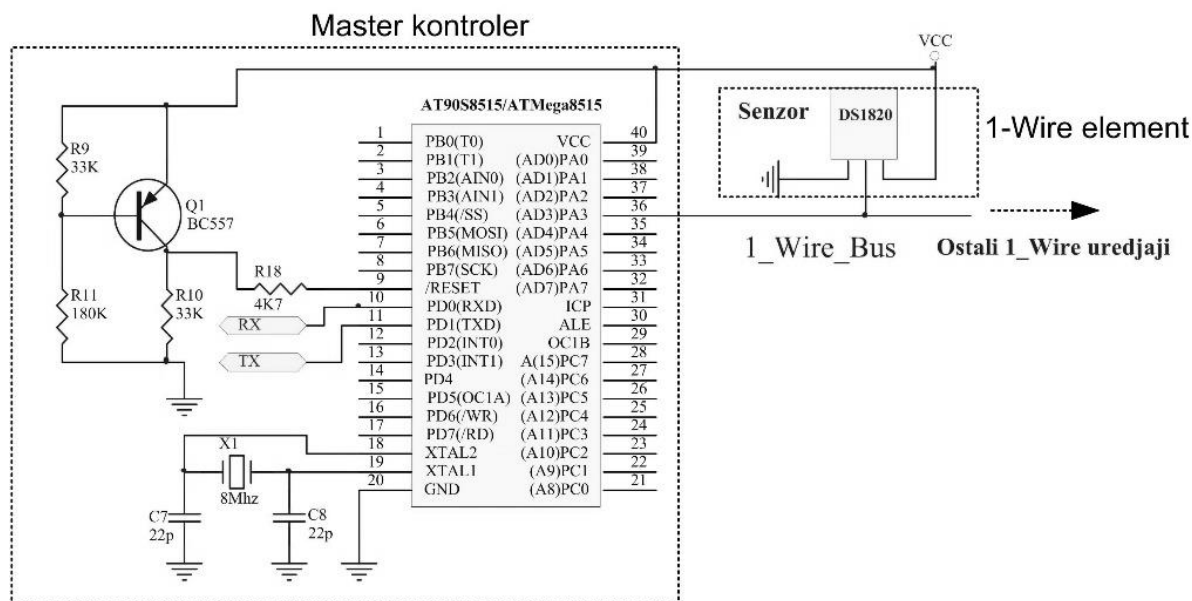
3.2.1 Micro_Lan (1_Wire)

Kao što je gore napomenuto, akcenat ovog magistarskog rada nije na Micro_Lan sistemima, međjutim ovdje će oni biti opisani u kratkim crtama na primjeru temperaturnog senzora DS1820 (Dallas-Maxim) koji se upotrebljavama u okviru inteligentnog PLK senzora. Biće razmatrana njegova integracija na RISC mikrokontrolere opšte namjene kakav je ATmega16 (Atmel).

1-Wire periferije su jeftini i fleksibilni uređaji koji se lako povezuju na 1-Wire bus (npr. CAT 5 kabal). Koriste 1 I-O pin kao intefejs za dvosmjernu komunikaciju, dok su druga dva provodnika predvidjena za masu i opciono napajanje. Napajanje je najčešće „parazitno“, preko data pina, a rijetko preko dodatog provodnika. Postoji širok spektar 1-Wire uređaja, za raznovrsne namjene: temperaturni senzori, pojačavači, «on-off» detektori, digitalni I/O uređaji, memorijske komponente, A/D i D/A konvertori, itd. Svi oni prepoznaju 1-Wire komunikacioni protokol i sadrže jedinstven identifikacioni kod (ID).

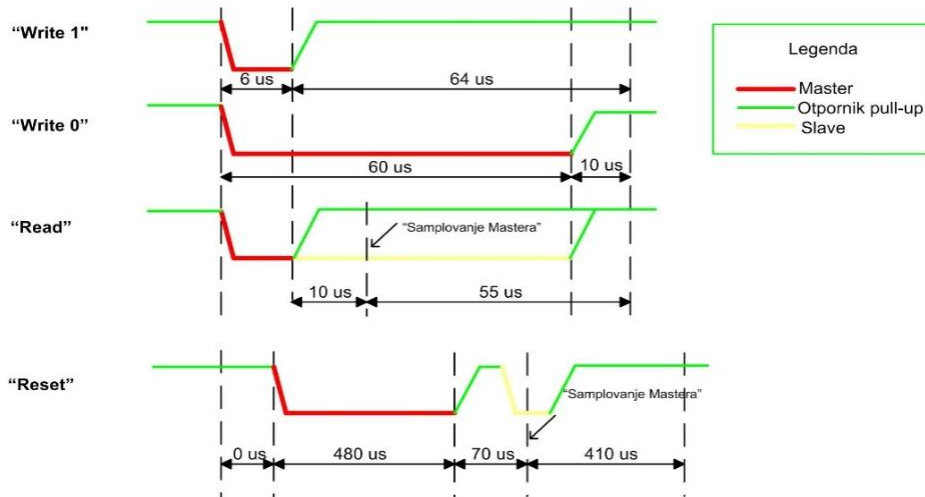
1-Wire mreža se sastoji od 3 elementa: bas master kontrolera, žica i konektora i 1-wire elmenata (slave). Mreža pripada grupi open-drain master/slave arhitektura koja upotrebljava pull-up otpornik prema naponu napajanja. To je rigidni komunikacioni protocol zato što slejvovi ne mogu medjusobno komunicirati bez autorizacije mastera. Unutar svakog slejva postoji jedinstvena sekcija sa 64bitnim registrom koji sadrži jedinstveni identifikacioni broj koji služi kao adresa. [12]

Na slici 3.3 je dat primjer jednostavne 1-Wire mreže gdje ulogu slejva ima temperaturni sensor DS1820 a mastera 8bitni AVR RISC μ P ATmega16. Dati sensor ima temperaturni opseg od -55°C do 135°C sa tačnošću $\pm 0.5^{\circ}\text{C}$.



Slika 3.3: 1_Wire temperaturni sensor spojen na AT90S8515 mikrokontroler

1-Wire bas je u praznom hodu na visokom nivou. Komunikacija preko 1-Wire basa je podijeljena u vremenske slotove od $60\mu\text{s}$ u kojem se prenosi 1 bit podatka. Svaki bit prenosa, bez obzira na smjer, mora biti iniciran od strane mastera. Uvijek počinje povlačenjem basa na niski nivo (nulom), koji će sinhronizovati logiku tajminga za svaku jedinicu. Postoji pet osnovnih komandi za komunikaciju na 1-Wire basu: “Write 1”, “Write 0”, “Read”, “Reset” i “Presence”. Njihov tajming je dat na slici 3.4.



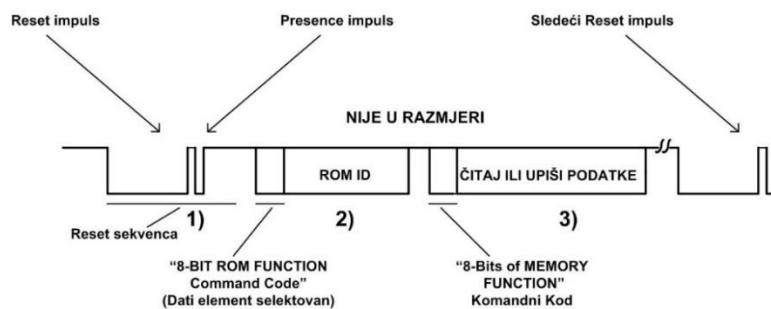
Slika 3.4: Tajming osnovnih 1-Wire operacija

Tipična komunikaciona 1_Wire sekvenca je data na slici 3.5. Sastoji se iz sledećih sub-sekvenci:

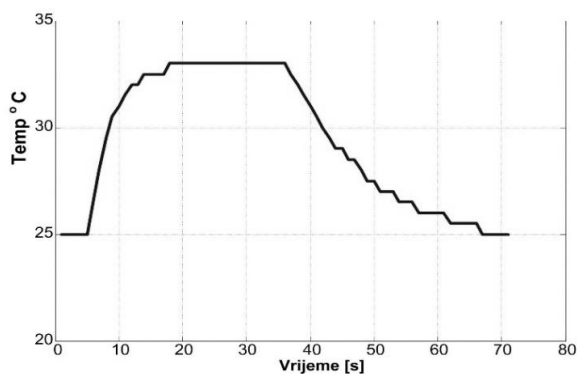
Inicijalizacije: Sastoji se od reset impulsa emitovanog od strane mastera i presence impulsa emitovanog od slavea.

ROM comandi: Omogućava masteru da izabere jedan od prisutnih slaveova i da očita 64-bitni ROM kod koga šalje slave u cilju identifikacije.

DS1820 funkcijskih komandi: Koje omogućavaju masteru da inicijalizuje temperaturnu konverziju, ta čita ili upisuje na određenju memorijsku lokaciju i da određuje mod napajanja



Slika 3.5: Primjer 1-Wire sekvence

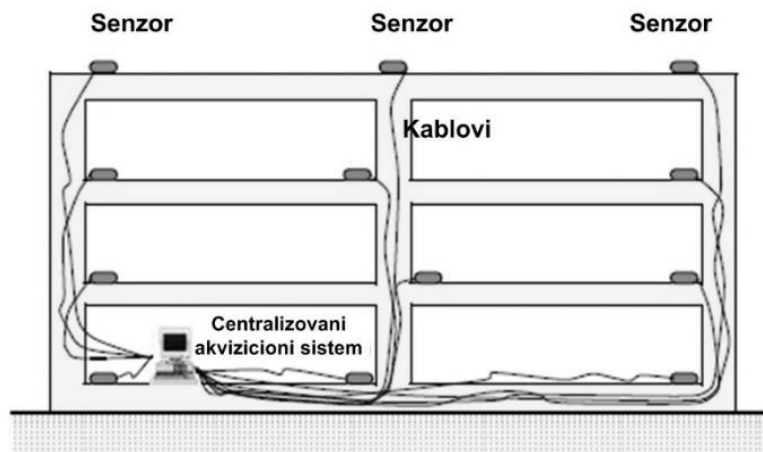


Slika 3.6: Odziv temperaturnog senzora DS1820 spojenog na AT90S8515

3.3 Bežični “smart senzori” za KA

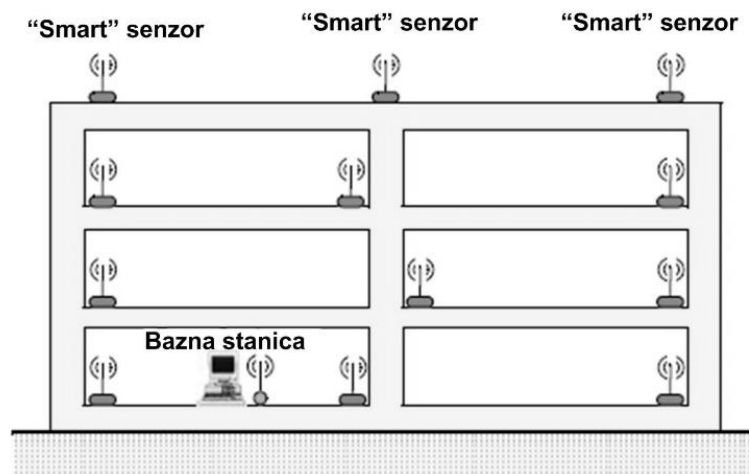
Svjedoci smo sve češće primjene bežičnih sistema (wireless) u oblasti kućne automatizacije. Na ovaj način se ostvaruje značajna finansijska ušteda jer oko 80% novca prilikom instalacije sistema kućne automatizacije se troši na postavljanje i ugradnju kablova. Tehnika bežične automatizacije počiva na hardversko-softverskim rešenjima komunikacije perifernih uređaja (intelegentnih senzora) sa centralnim sistemom (kućnim kompjuterom). Sam senzor može biti realizovan na više načina uključujući standardni ili mikroprocesorski princip. Osim senzora važnu ulogu igraju i protokol komunikacije kao i hardver i softver na strani centralnog sistema. Mjerenja temperature, osvjjetljenja, vlažnosti su samo neki od primjena upotrebe bežičnih senzora.

U ovom Paragrafu je opisan način realizacije AM-FM senzora za kućnu automatizaciju opremljenim prostim ali efikasnim komunikacionim protokolima. Ovi senzori se upotrebljavaju u slučajevima gdje je potrebno redukovati troškove žičanog povezivanja periferija i centralnih sistema. Takodje se upotrebljavaju u slučajevima nepostojanja mrežnog napajanja ili pak tamo gdje je potrebna stalna mobilnost senzora (od mjesta do mjesta gdje nije lako pristupiti nekom od žičanih puteva prenosa). Primjer klasične realizacije senzora za kućnu automatizaciju i njihovo povezivanje na centralni računar je dat na slici 3.7. Kako se može vidjeti ovo je dosta neprikladno rešenje jer zahtijeva veliki broj kablova i dosta naporno, «glomazno» ožičenje.



Slika 3.7: Centralizovani (kablovski sistem) kućne automatizacije

Za razliku od sistema na slici 3.7 sistem automatizacije baziran na bežičnim sensorima, slika 3.8, je mnogo fleksibilniji i jednostavniji za ugradnju a vremenom mu i cijena postaje prikladnija. Bazna stanica komunicira sa «Smart» sensorima bežično uz odgovarajući komunikacioni protokol koji može biti različite složenosti.



Slika 3.8: Moderni wireless sistem KA

Pošto svi «smart» senzori bez obzira dali se radi o žičanim ili bežičnim sadrže odgovarajuće zajedničke komponente kao što je procesorsko kolo, izvor napajanja, kola senzorskih pretvarača to će u nastavku biti prvo izloženi moduli po kojima se razlikuju a to su komunikacioni moduli.

3.3.1 Vrste bežičnih komunikacionih tehnologija za «smart» senzore

Današnji bežični senzori koriste više komunikacionih tehnologija. RF senzori (AM i FM) koji funkcionišu na 27 MHz, 433 MHz, 868 MHz ili 900 MHz. Na 2,4 GHz i 5,4 GHz. Takođe je prisutna komunikacija pomoću 802.XX i Bluetootha. ZigBee je zastupljen na tri frekvencije 868 MHz, 915 MHz i 2,4 GHz. Takođe na 2,4 GHz postoji wireless USB a ređe se koriste infrared.

3.3.1.1 RF senzori

RF bežični sistem je efikasan u pogledu cijene koštanja. Počiva na FSK i ASK modulaciji i obuhvata širi opseg frekvencija. Najčešće radi na 412, 433 i 866 MHz. Omogućava snage do nekoliko mW sa dometom od nekoliko desetina do nekoliko stotina metara.

3.3.1.2 Bluetooth

Podržava multimedijalni saobraćaj i prenos podataka. Radi na 2,4 GHz u ISM (*Industrial-Scientific-Medicine*) opsegu. Obuhvata spektar (83.5 MHz) i podijeljen je u 79 komunikacijskih kanala širine 1 MHz. Pri komunikaciji primopredajnici „prelaze” sa kanala na kanal na pseudo

slučajan način. Kanal je podijeljen na vremenske odsječke u trajanju 625 ms, a za svaki odsječak se određuje drugačija frekvencija prelaska. Cijena koštanja je do 4 puta veća nego kod infrared ili RF prenosa.

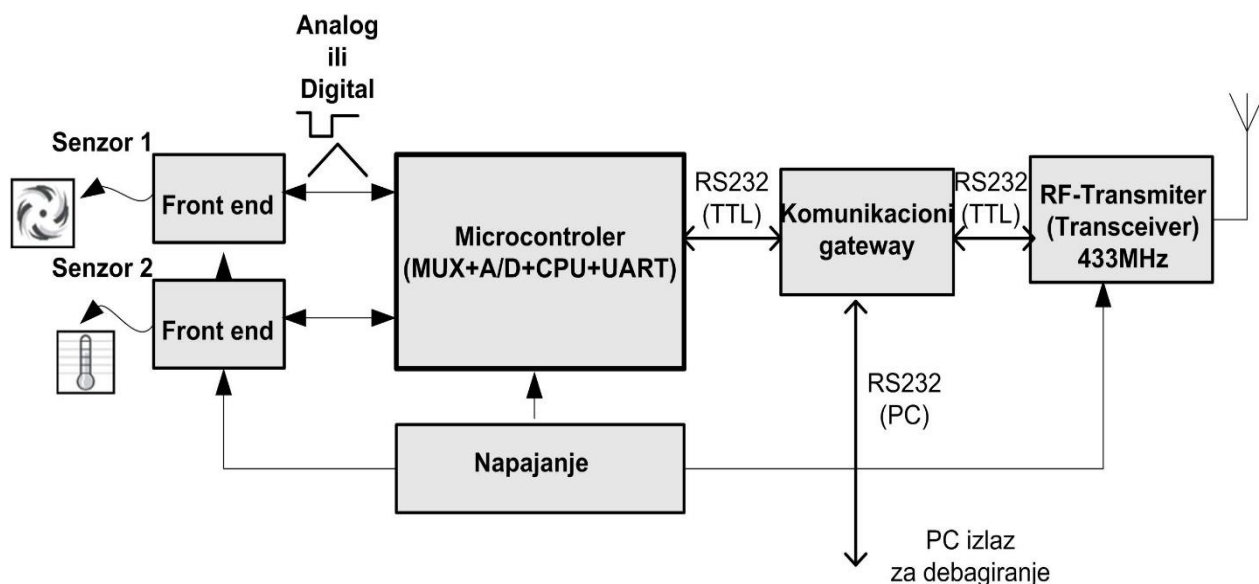
3.3.1.3 Infrared

IR wireless je korišćenje bežične tehnologije u sistemu ili uređajima koji razmenjuju podatke preko infrared radijacije. Infrared funkcioniše u *“line-of-sight”* modu što znači da mora postojati putanja optičke vidljivosti između predajnika i prijemnika ili u *„diffuse“* modu bez uslova optičke vidljivosti. IR wireless ne prolazi kroz prepreke, npr. zidovi, što ograničava komunikaciju unutar prostorija zgrada.

3.3.1.4 ZigBee

Za razliku od Bluetooth-a koji zahtjeva 10 Mips i 128 KB na 16 bitnom procesoru ZigBee zahtjeva 10 Mips i 4 do 32 KB na 8 bitnom procesoru, što ga čini prikladnijim za kućnu automatizaciju. Pri sadašnjem stanju tehnologije ne postoji prirodna podrška ZigBee-a na kućnim računarima.

3.3.2 Arhitektura predloženog bežičnog smart «senzora» za kućnu automatizaciju

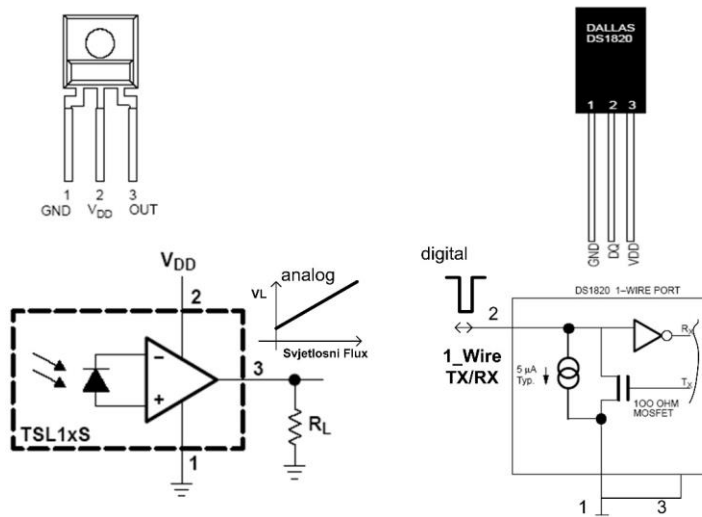


Slika 3.9: Blok šema RF baziranog „smart senzora”

Slika 3.9 predstavlja blok šemu RF baziranog «smart» senzora za kućnu automatizaciju. Sastoji se od senzora, ulaznog pojačavačkog kola (front end), mikrokontrolera, komunikacionog gatewaya, izvora napajanja i RF modula. Dati moduli su integrisani na štampanoj ploči, a u savremenoj tehnologiji mogu biti integrisani i na čipu.

3.3.2.1 Senzori i ulazno kolo za kondicininrinja (front end)

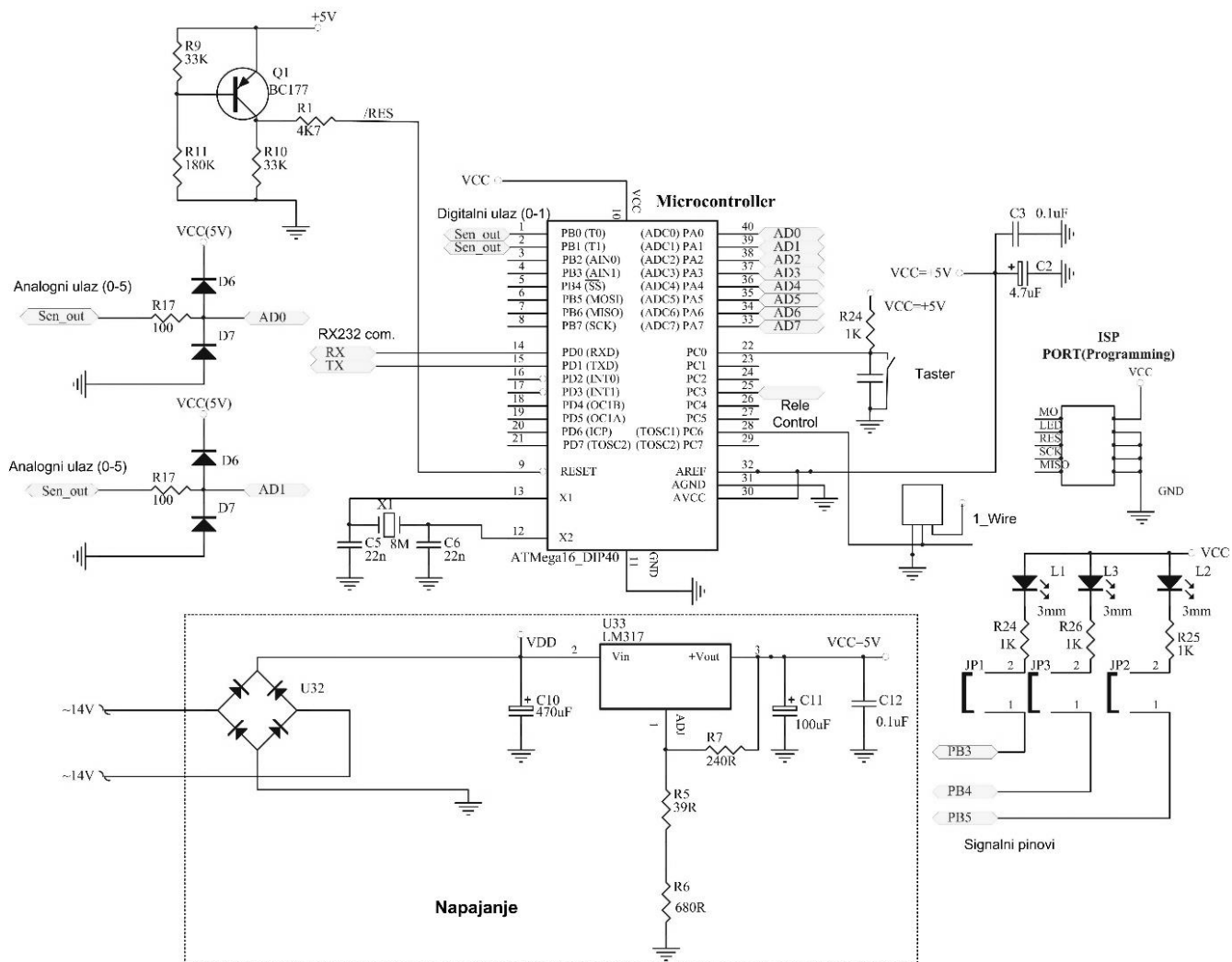
Ulazna mjerna veličina se uzima sa senzora koji mogu biti različitog tipa. Pomoću front enda signal sa senzora se može pretvoriti u digitalni ili analogni ekvivalent. Ako se radi o analognom ekvivalentu onda je on naponskog (0-5V, 0-10V) ili strujnog oblika (4mA-20mA). Obično se senzor i elektronski dio integrišu u jedan čip koji predstavlja senzorski čip. Takvi čipovi predstavljaju najčešće korištene senzore u procesu kućene automatizacije. Na slici 3.10 je dat senzor TSL1XS koji pretvara osvjetljaj u izlazni napon i može se smatrati veoma pogodnim u slučajevima regulacije osvjetljenja primjenom kućne automatizacije. On na izlazu daje analognu veličinu koja se vodi na A-D ulaze mikrokontrolera. Drugi primjer predstavlja već pomenuti temperaturni senzor (termostat) DS1820 koji se digitalno upravlja. Ovaj senzor se veže na digitalni ulaz-izlaz mikrokontrolera kojim se emulira 1_Wire protokol.



Slika 3.10: Primjeri integrisanih senzora primijenjenih u kućnoj automatizaciji

3.3.2.2 Mikrokontroler:

Mikrokontrolersko kolo u suštini predstavlja mikrokontroler. Mikrokontrolersko kolo je prikazano na slici 3.11.

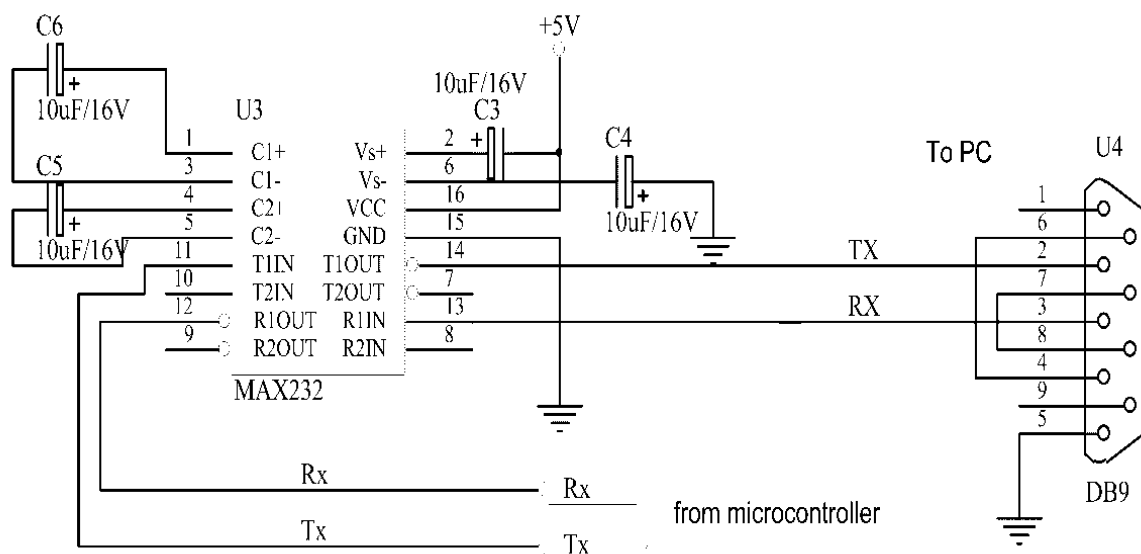


Slika 3.11: Mikrokontrolersko kolo

Njegovo jezgro predstavlja mikrokontroler ATmega16 koji se pogoni na 8MHz, a čije su karakteristike gore elaborirane. U ovoj fazi projekta koriste se 2 10bitna A-D ulaza, dok su ostali predviđeni za buduća proširenja. Sistem je ISP programabilan (In System Programable) što omogućava uploadovanje mikrokontrolerskog koda direktno na ploči. Nekoliko I-O pinova je upotrijebljeno za signalizacijske izlaze, dok je serijska komunikacija omogućena preko pinova PD0 i PD1 (Rx, Tx). Za napajanje mikrokontrolerskog kola kao i analognog «front enda» koristi se standardni 3-pinski regulator LM317. Kao konvertor RS232 TTL nivoa u standardni (+12V, -12V) koristi se drajver realizovan pomoću MAX232, prikazan na slici 3.12. Pošto se mikrokontroler napaja uni-polarno (0-5V) to su svi izlazi pojačavača/pretvarača prilagodjeni ovom nivou. Naime, svi signali imaju ugrađeni ofset i maksimalnu amplitudu od 2.5V. Radni parametri kao što su frekvencija odabiranja, broj odbiraka, izbor pre-filtra, terminator i ostali mogu biti smješteni u EEPROM memoriji mikrokontrolera ili zadati od strane hosta u postupku inicijalizacije.

3.3.2.3 Komunikacioni gateway:

Ovaj gateway predstavlja prelaz izmedju mikrokontrolera i RF komunikacionig modula. Pošto većina RF komunikacionih modula ima TTL kompatibilan ulaz izlaz to znači da se oni direktno mogu vezati na RXD-TXD (RS232) pinove mikrokontrolera.



3.12 Komunikacioni gateway

Medjutim, u svrhu debugovanja i eksperimentisanja nekada je potrebno komunikacioni modul direktno vezati na serijski port PC računara. Tada je potrebno ugraditi translator standardnog RS232 nivou (+12V, -12V) na TTL nivo i obrnuto. U tu svrhu poželjno je koristiti hardverski drajver, npr., MAX 232 ili MAX 202. Šema takvog drajvera je prikazana na slici 3.12. Kao što će se kasnije vidjeti svi naši razvijeni RF moduli posjeduju ovaj drajver (gateway).

3.3.2.4 RF moduli za kućnu automatizaciju

Mikrokontroler sa hostom može, pored ostalog, komunicirati i putem RF-a. U našem slučaju prenos podataka na ovaj način je vršen primjenom dvije modulacione tehnike:

ASK modulacije (Amplitude-Shift Keying),

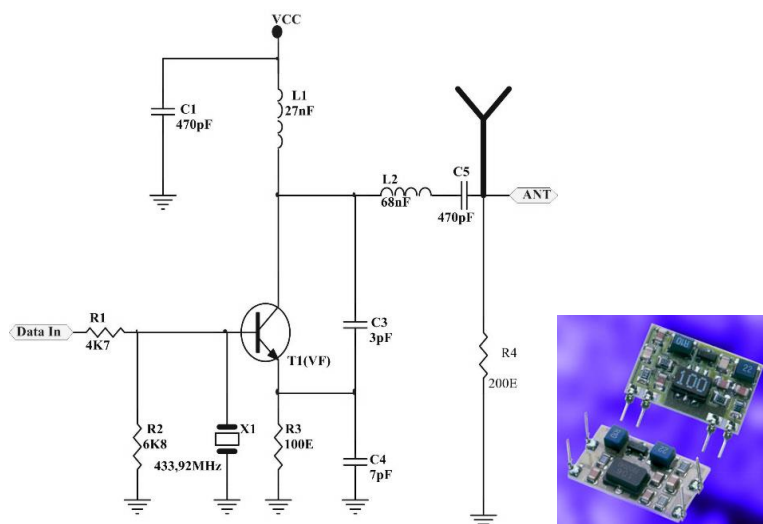
FSK modulacije (Frequency-Shift Keying).

U oba slučaja frekvencija nosioca iznosi 433.92 MHz.

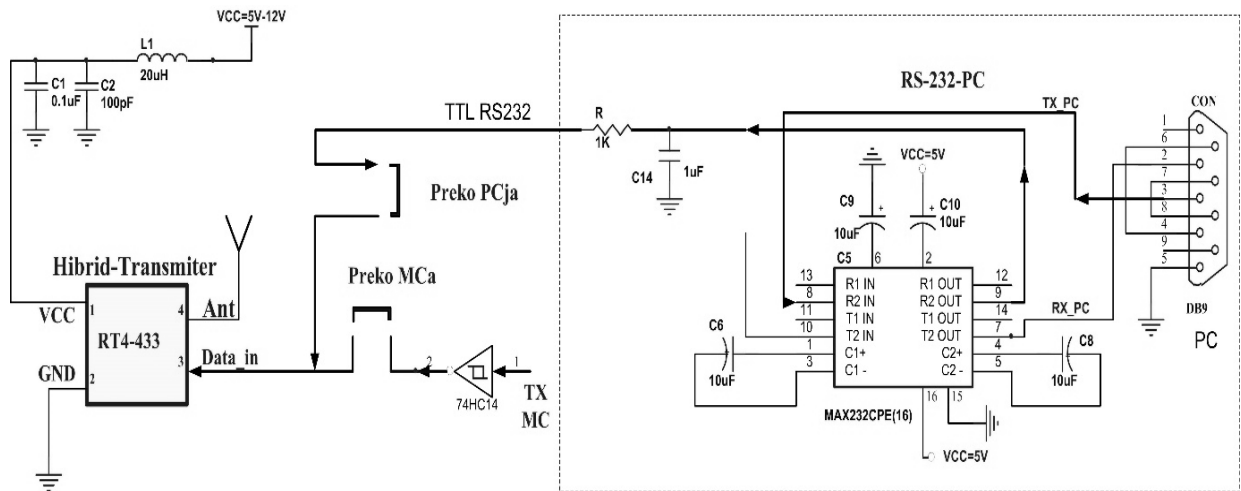
3.3.2.5 ASK-RF

Postoji više varijanti sklopova koji koriste ASK modulaciju a koji se koriste u bežičnim senzorskim aplikacijama (uglvanom za komercijalnu upotrebu), ali u poslednje vrijeme, u pogledu arhitekture, dominiraju rešenja bazirana na SAW rezonatoru. Naime, frekvencija oscilovanja jednotranzistorskog oscilatora koji u suštini predstavlja RF predajnik striktno je regulisana rezonantnom frekvencijom SAW rezonatora datog na slici 3.13(a) (komponenta X1). Na ovom principu radi i sklop korišten u ovom radu. Takođe, omogućen je visok Q faktor ($Q > 10000$) uz veoma jednostavnu topologiju kola. Cijena SAW rezonatora je veoma niska i danas se može naći za manje od 0.75\$.

U ovom projektu kao ASK-RF predajnik korišten je četvoropinski hibridni predajnik AM-RT4-433 [40], slika 3.13(b). Pomoću ovog predajnika se mogu prenositi digitalni podaci do frekvencije 4KHz, a posjeduje CMOS/TTL data ulaz. Domet mu je oko 70m u otvorenom prostoru (30m u zatvorenom). Posjeduje veoma malu strujno potrošnju, oko 4mA uz napon napajanja od 5-12V. Na slici 3.14 je prikazana šema korištenog predajnika. Digitalni podaci za prenos se dovode u CMOS/TTL formi u slučaju direktnog vezivanja na pin TX_C_TTL (J2 zatvoren J1 otvoren) ili putem RS232, pomoću bafera MAX232 (standardni DB9 konektor, J1 otvoren, J2 zatvoren), u slučaju korištenja sistema u konekciji sa PC konfiguracijom. Korištena je standardna $\lambda/4$ antena.

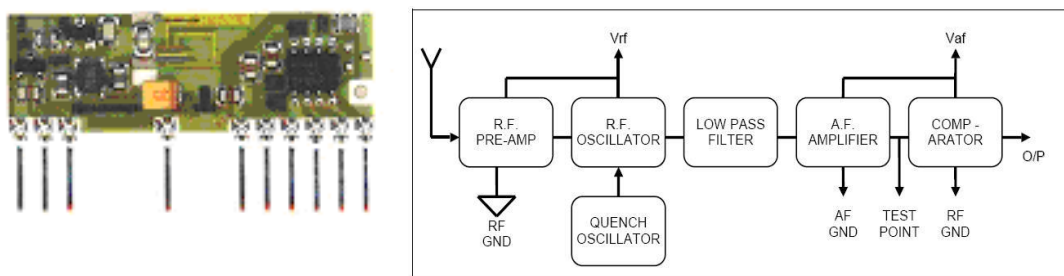


Slika 3.13: a) Šema SAW RF oscilatora, b) njegova hibridna realizacija.

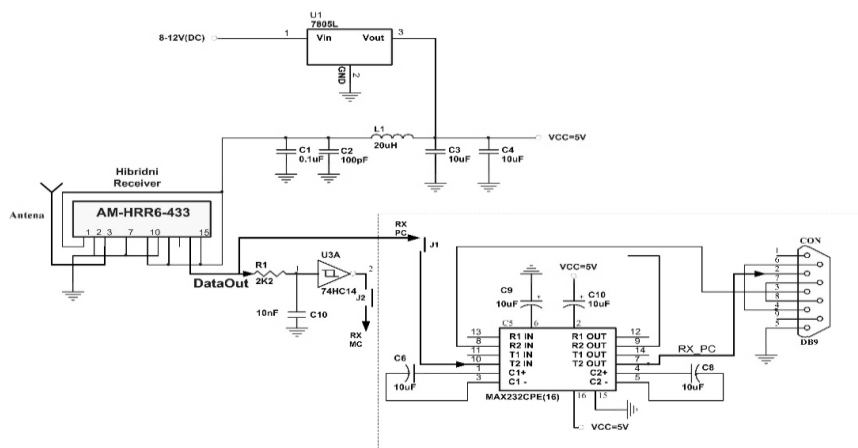


Slika 3.14: Šema ASK-RF Predajnika korištenog u radu

Kao prijemnik korišten je super-regenerativni hibridni modul AM-HRR6-433 ([40]). Ovaj hibrid posjeduje veoma visoku frekvencijsku stabilnost u širokom temperaturnom dijapazonu. Upotrebom laserski trimovanih komponenti omogućen je operativni mod bez dodatnog podešavanja. Prijemnik proizvodi CMOS/TTL kompatibilni izlaz i zahtijeva samo konekciju na antenu i napon napajanja. Konzumacija HRR6 modula je veoma mala i iznosi 0.5mA. Prijemni opseg mu se kreće u granicama 50m (indoor). Izgled modula kao i njegova blok šema prijemnika su dati na slici 3.15 . Na slici 3.16 je data šema projektovanog prijemnog modula za oba slučaja konekcije: direktno na mikrokontroler (izlaz C2) ili na COM port PC računara (DB9 konektor).



Slika 3.15: a) izgled AM-HRR6-433 prijemnog modula, b) njegova blok šema



Slika 3.16: ASK-RF prijemnik baziran na hibridnom modulu AM-HRR6-433.



Slika 3.17: AM predajnik i AM prijemnik

3.3.2.6 FSK-RF

Za prenos signala putem FSK modulisanog RF signala korišten je modul BIM2-433. Ovoj modul integriše FM predajnik i FM prijemnik i kao takav je idealan za niže-dometne RF linkove i za brzine do 40kbit/s. Posjeduje brzu promenu TX/RX moda (<1ms). Pored ovoga, njegove značajne karakteristike se mogu sumirati u:

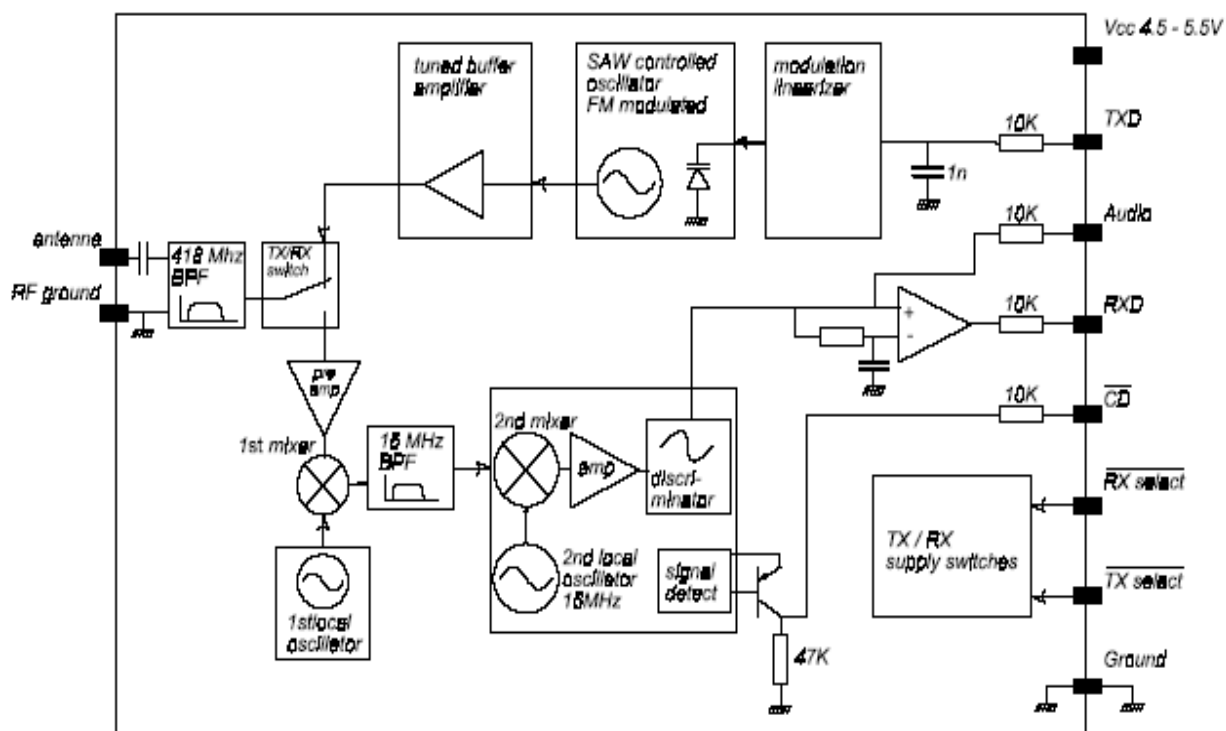
- Minijaturno PCB pakovanje,
- Kompatibilan sa ETS 300-220 na 433.92 MHz (BiM-433-40),
- SAW kontrolisana FM transmisija na -6dBm ERP,
- 107dBm prijemna osjetljivost,
- Jednostruko 4.5 to 5.5V napajanje sa strujom < 15mA (TX ili RX),
- Half duplex prenos podataka do 40 kbit/s,
- Pouzdan prijem do 50 metara u “in-building” i 200m “out-building”,
- Direktno povezivanje na 5V CMOS logiku, itd,

Izgled BIM2-433 modula je dat na slici 3.18.



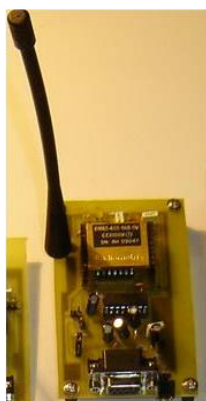
Slika 3.18: Izgled modula BIM2-433.

Blok šema ovog modula data je na slici 3.19.

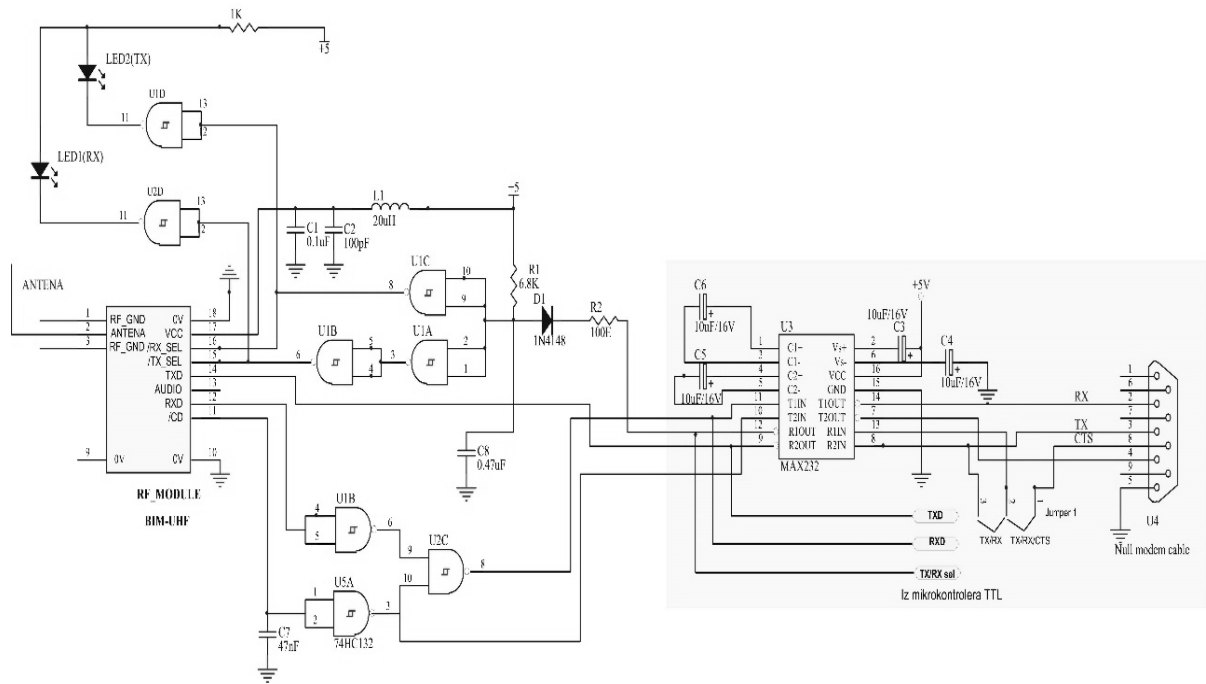


Slika 3.19: Blok šema modula BIM2-433.

Prilikom dizajniranja primopredajnika baziranog na ovom modulu za potrebe ove teze, pošlo se od činjenice da se isti može direktno povezati na RS232 port u TX-RX i CTS-RTS modu. Na slici 21 se daje takva šema, a mod prenosa se podešava odgovarajućim džemperom “Jumper/SER”. Treba napomenuti da u slučaju TX-RX konekcije kolo pravilno radi samo kada se signali šalju sa odgovarajućom preambulom i odgovarajućim odnosom “nula”/“jedinica”, odnosno “mark”/“space”. To se postiže kolom (vremenskom konstantom) kojeg čine R1, C8, D1 i R2 u kombinaciji sa odgovarajućim invertorima.



Slika 3.20: FM-RF primopredajnik realizovani modul



Slika 3.21: FM-RF primopredajnik šema

3.3.2.7 RS232 preko RFa

Kao što je prethodno istaknuto, a što se da vidjeti iz priloženih šema, prenos fizioloških signala do udaljenog hosta se vrši po principu “RS232 preko RF-a”. Taj pristup je izabran iz razloga velike rasprostranjenosti RS232 protokola, koji je faktički postao nezamenljiv standard, kao i zbog toga što veliki broj razvojnih softvera ili gotovih paketa podržava ovaj protokol. Pored toga, većina familija modernih mikroprocesora posjeduju integrisani UART, a postoje i jednostavni adapteri prema protokolima kao što USB. Tokom rada na projektu, podaci su slati različitim brzinama od 2400 bps do 57 kbs, a pri komponovanju protokola vodilo se računa o najpovoljnijoj vrijednosti “bit-error”-a, za odgovarajući “mark”/”space” (sumarni odnos impulsa i pauze) odnos. Preliminarna ispitivanja su pokazala da se za manje brzine prenosa i na manjim rastojanjima može upotrebljavati izvorni RS232 protokol, dok za brzine prenosa veće od 19200bps podaci se moraju komponovati u odgovarajući paket. Najpovoljniji rezultati su dobijeni sa paketom sledećeg oblika:

- preambula (55h ili AAh),
- 1 ili 2 bajta vrijednosti FFh,
- startnim bajtom poruke 01h,
- kodovanim bajtovima koji predstavljaju podatke i
- 16bitnim CRC-om, koji se dodaje na kraju paketa.

Bi-phase bit coding (Manchester) metod je korišten pri pakovanju podataka poruke. Ovaj kod daje dobre rezultate pri redudansi 100% (mark/space=50/50).

3.4 Žičani PLC «smart senzori» za kućnu automatizaciju

3.4.1 Uvod

PLC (Power line communication) koristi postojeće energetske kablove za prenošenje podataka. Ovakav način prenošenja podataka je relativno stara ideja i primijenjen je 1838 u cilju daljinskog mjerenja električne energije, a 1897 godine imamo već prvi patent povezan sa datom problematikom. 1920 godine dva patenta su prijavljena preko Američke telefonske i telegrafске kompanije pod nazivom «Prenos nosioca preko napojnih vodova». Na prvi pogled se može pomisliti da je ova ideja dosta eksploatisana, međjutim to nije slučaj i napojni vodovi se veoma rijetko koriste za prenos informacija.

Postoje dvije glavne oblasti PLC primjena – jedna za broadband Internet, tj. pristup Internetu iz kuće i druga za potrebe kućne automatizacije. U našim istraživanjima smo se pretežno bazirali na PLC za potrebe KA.

Kućne i kancelarijske mreže obično koriste Ethernet ili “wireless” sklopove. Ethernet omogućava velike brzine prenosa podataka, ali zahtijeva specijalni kabal 5 kategorije poznat kao CAT5 kojeg je potrebno instalirati u čitavom stanu-kući. Bežični sklopovi trenutno postaju mnogo popularniji i rade prilično pouzdano, ali za sada omogućavaju dosta ograničene brzine i nijesu imuni na fizičke prepreke (zidove, armirani beton, itd).

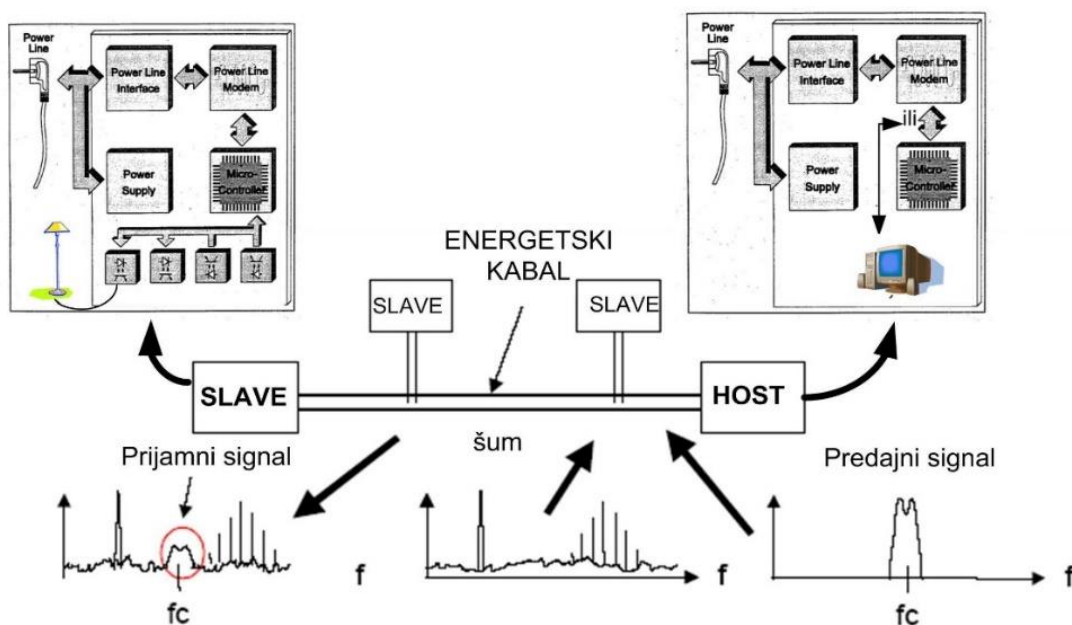
Najveća prednost PLC je ta što ne zahtijeva posebno ožičenje koje već danas predstavlja glavnu tržišnu stavku za određivanje cijene sistema kućne automatizacije. Dodatna prednost je široka pristupačnost samoj mreži jer skoro svaka prostorija posjeduje najmanje tri do četiri priključna mjesta (utičnice 220V). Što znači, gdje god imamo utičnicu imamo i pristup KA mreži.

PLC tehnologija je kasnila sa implementacijom zato što su provodni kablovi bili pretežno projektovani za distribuciju energije čija frekvencija iznosi 60Hz. Takav medijum je dosta nezgodan za prenos visokofrekventnih signala zato što pored ostalog posjeduje promenljivu impedansu, značajan šum, izraženo slabljenje koje se mijenja od tipa uređaja priključenog na mrežu.

U cilju boljeg poznavanja PLCa sledeća sekcija daje globalni pregled komunikacionog sistema što podrazumijeva osnovne izmjerene performanse, metode prenosa podataka, diskusiju elemenata sistema i njihovu integraciju

3.4.2 PLC model

Slika 3.22 prikazuje pojednostavljeni PLC sistem. Kao što se vidi glavni zadatak ovakvog sistema je da prenese digitalne informacije, obradi ih i vrati povratnu informaciju za bilo koja dva elementa uključena u sistem. Sastoji se od Hosta i više slejvova. Host može biti kompatibilni PC računar ili mikrokontroler. Glavni interfejs za prenos informacija predstavlja PLC modem koji se veže na osnovno kolo i hosta i slavea. Zadatak modema je da prihvati-pošalje digitalne signal iz hosta-slavea izvrši njegovu modulaciju-demodulaciju.



Slika 3.22: Pojednostavljeni PLC system

Iz više razloga koji će biti elaborirani u sledećemo poglavlju prijemni signal je veoma malog intenziteta i superponiran je sa značajnim šumom.

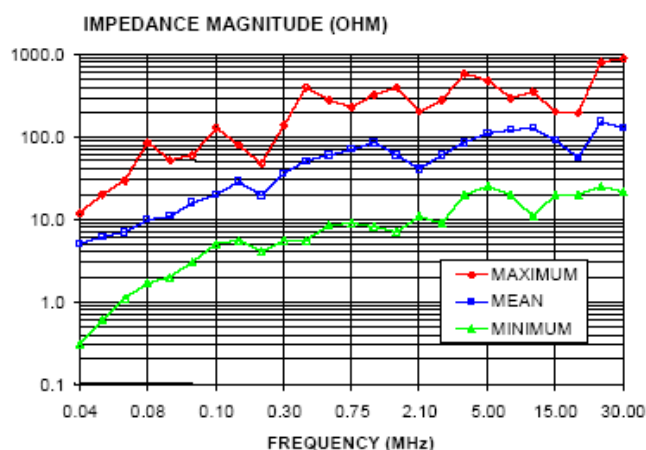
3.4.3 PLC parametri i njihov opis

Komunikacioni medij koji se koristi u PLC svrhe sadrži više paralelno spojenih provodnika i uređaja. Svi kućanski aparati kao i kancelarijski uređaji predstavljaju opterećenje koje se mora

uzeti u obzir prilikom analize komunikacionog kanala koji se koristi za potrebe kućne automatizacije. Pored ovih opterećenja značajnu ulogu igraju i opterećenja u čitavoj zgradi jer se pretpostavlja da distributivni transformator napaja više od jednog stana. Ipak, prilikom projektovanja ovih sistema uticaji šuma i impedance mogu se smatrati najkritičnijim. Pored njih značajnu ulogu igraju slabljenje i pojava nosećeg talasa.

3.4.3.1 Impedansa

Energetska niskonaponska mreža ima veoma promjenljivu impedansu koja zavisi od nekoliko više faktora. Značajno istraživanje u ovoj oblasti sproveli su Malack i Engstrom (IBM Electromagnetic Compatibility Laboratory), koji su mjerili RF impedansu 86 komercijalnih distributivnih sistema naizmjenične struje u 6 evropskih zemalja, slika 3.23. Ova mjerenja pokazuju da se impedansa mreže povećava sa frekvencijom i da na 100kHz iznosi 1.5 to 8 Ohm. Ekvivalentna impedansa je određena sa dva parametra – opterećenja spojenog na mrežu i impedance distributivnih transformatora. U posljednje vrijeme ovdje se uključuje i komponenta koja dolazi od strane EMI filtera koji se primenjuju u savremenim generacijama kućnih aparata (frižiderima, mašinama za pranje veša, televizijskim prijamnicima itd.). Ekvivalentna impedansa je pretežno induktivnog karaktera. Za tipično omsko opterećenje slabljenje signala se kreće od 2 do 50dB pri 150kHz što zavisi od distributivnog transformatora i veličine samog opterećenja. Tako da veliki potrošači kao što su termo peći ili bojlerski grijači mogu značajno oslabiti signal nosioca. Medjutim, nekad je moguće ekvivalentnim kapacitivnim i induktivnim opterećenjem proizvesti rezonantno kolo što može značajno oslabiti signal u širem frekventnom opsegu.

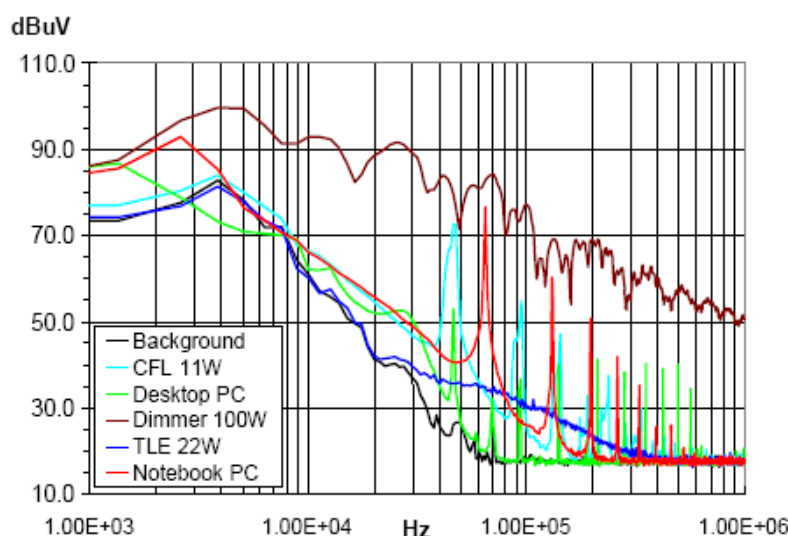


Slika 3.23:Zavisnost impedanse mreže(Malacak i Engstrom)

3.4.3.2 Šum

Aparati spojeni na isti sekundar distributivnog transformatora su pretežno izvori šuma kod PLC. Primarni izvori šuma su triaci korišteni kod dimera, drajvera za upravljanje motorima,

prekidačkim izvorima napajanja, balastima za fluorescentne lampe itd. Takav se šum javlja u obliku harmonijskog multipla od 50Hz. Univerzalni motori koji se mogu sresti kod miksera ili bušilica takodje proizvode šum, ali on nije tako značajan kao u slučaju dimera koji su permanetnog karaktera. Zadnjih godina se pojavljuju dva izvora veoma intenzivnog šuma, kompaktne fluorescentne lampe (poznate kao štedljive sijalice), Compact Fluorescent Lamps (CFL) , i punjači za baterije malih potrošača kao što su notebookovi i mobilni telefoni. U mnogim slučajevi dominantne frekvencije šuma se kreću u opsegu od 10KHz do 150KHz. Često se impulse koji prouzrokuju šum javljaju oko prolaska sinusoide kroz nulu (zero crossing) jer se u toj tački vrši sinhronizaciju kod velikog broja impulsnih regulatora. Donja slika 3.24 pokazuje spektar signala šuma snimljen pomoću izolacionog transformatora sa VARIACom, spektralnog analizera HP4395A visokofrekvencijski izolovanog preko transformatora 2mH transformer (1:1) i kondenzatora 1uF. Slika 3.24 daje šum okruženja laboratorije u kojoj su sprovedeni eksperimenti i koja pretežno sadrži šum od personalnih računara.



Slika 3.24:Spektar šuma nekih od značajnih potrošača (Cantone)

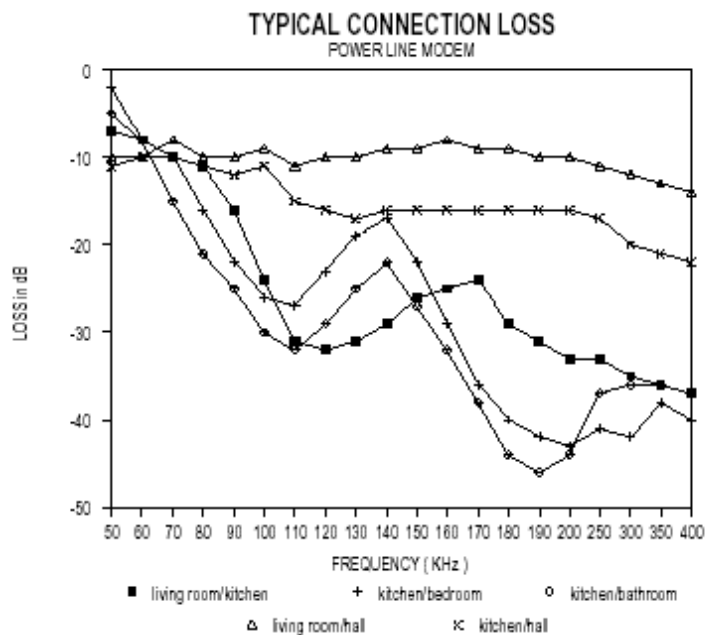
3.4.3.3 Slabljenje

Kao što je prikazano u gornjem tekstu slabljenje signala nosioca, a samim tim i digitalnog ekvivalenta poruke je funkcija više parametara. Medjutim, to slabljenje je i funkcija dužine vodova kao i njihove putanje. Tako, npr. za signal čiji je nosioc 130KHz ono iznosi:

10dB do 15 dB u slučajevima da su predajnik i prijarnik spojeni na istu fazu.

20dB do 30 dB u slučajevima kada se nalaze na različitim fazama, a indukcija signala se ostvaruje preko razvodne table koja unosi 10dB do 20dB dodatnog slabljenja.

Treba napomenuti da slabljenje zavisi i od topologije mreže na koju je sistem instaliran. Donja slika 3.25 daje dijagram slabljenja za različite frekvencije i različite putanje signala.



Slika 3.25: Statičko kašnjenje u funkciji frekvencije nosioca i njegovog puta (CHAFFANJON)

3.4.3.4 Stojeći talas

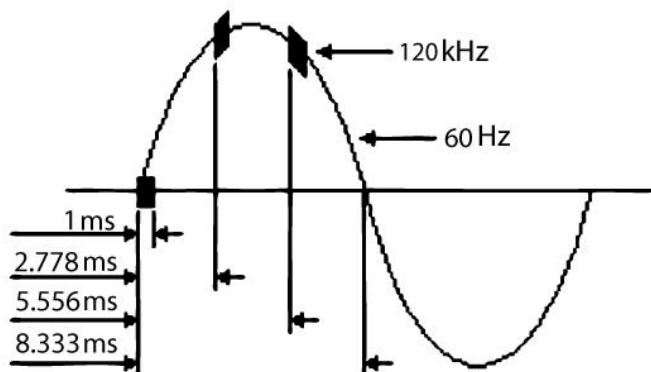
Stojeći talas može nastati u slučajevima kada se fizička dimenzija vodova približava $1/8$ talasne dužine nosioca. Za frekvencije 100 i 150kHz to približno iznosi 375 i 250m. Ovdje se mora uzeti dužina svih paralelno vezanih vodova sa sekundarne strane transformatora preko kojih se vrši komunikacija, jer je poznato da su više objekata stanovanja vezana na jedan transformator. Tako da pri projektovanju sistema treba izbjegavati da ta dužina sumarno iznosi više od 250m.

3.4.4 Uobičajeni PLC protokoli

3.4.4.1 X10

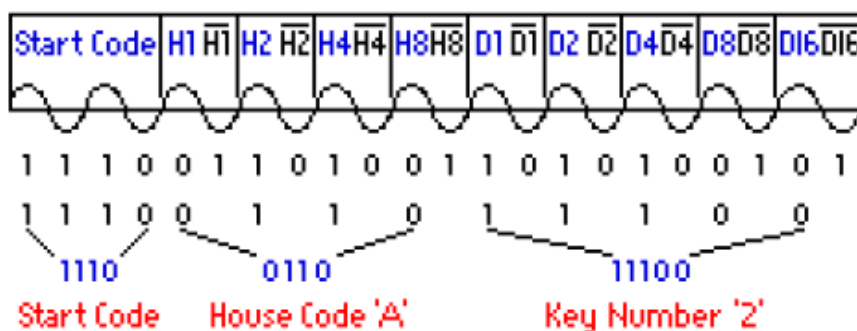
X10 protokol je jedno od najstarijih rešenja iz oblasti standardizovane KA. Patentiran je od strane kompanije Scotland - Pico Electronics sa prvim prodatim proizvodom 1978. X10 se koristi za uključenje i kontrolu osvetljenja, termostata, garažnih vrata, televizijskih prijemnika i antena itd. X10 je jednostavan i vrlo primenljiv. Koristi "off keying" (OOK) tehniku za prenos podataka. 120kHz burst signal se prenosi svake milisekunde i sinhronizovan je sa prolaskom

kroz nulu (zero-crossing) i to za pozitivnu i negativnu poluperiodu. Binarna "1" se reprezentuje burstom čije je trajanje 1ms, a koji se neposredno javlja poslije prolaska kroz nulu (sa maksimalnim kašnjenjem od 200us). Logička "0" obrnuto. Slika 3.26 prikazuje vremenski tajming burst signala u odnosu na prolazak kroz nulu i to za trofazni sistem.



Slika 3.26: X10 vremenski dijagram za 50 Hz signal

X10 paket zauzima 11 perioda mrežnog napona. Počinje sa startnim paketom koji ima vrijednost 1110 i koji zauzima 4 poluperiode, 2 ciklusa. Sledeća dva ciklusa sadrže takozvani «kućni kod», House code, i sledećih pet ciklusa sadrže kod broja ili (1 do 16) ili funkcijskog koda (on-off). Poruka se prenosi uvijek duplo sa dva ciklusa pauze, što znači da jedan X10 paket zauzima $2 \times 11 + 3 = 25$ mrežnih perioda, slika 3.27



Slika 3.27: X10 paket format

Prema tome X10 je protocol koji ima malu brzinu protoka. Prenosi se 11 bita za vrijeme od $25 \times 20 \text{ms} = 500 \text{ms}$, što znači bit-rate od $50 \times 11 / 25 = 22 \text{bits/sec}$ i pogodan je za trivijalne aplikacije a jedna od prednosti mu je jeftina realizacija.

3.4.4.2 Lonworks

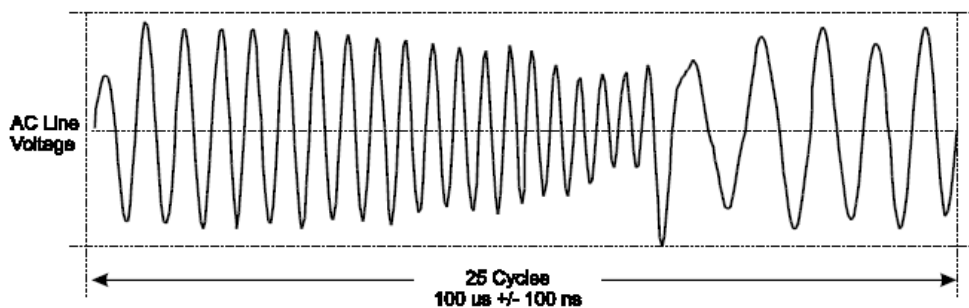
Lonworks je mrežni protokol kreiran od strane Echelon Corporation i dizajniran je da omogući komunikaciju između kontrolnih uređaja i nodova. Svaki nod u mreži kao što su prekidač, sensor, motor, detektor pokreta obavlja jednostavni zadatak, dok cjelokupna mreža predstavlja dosta složen sistem. Raniji standardi za Lonworks su koristili spreadspectrum modulaciju koja je pogodna u šumovitim sredinama gdje dominantni šum zauzima jednu od frekvencija. Koristi širi komunikacioni opseg od 100Khz – 400Khz, ali u zadnje vrijeme se izbjegava područje preko 150Khz zbog potencijalnih interferencija sa radio frekvencijama.

Zbog toga Echelonov zadnji proizvod PLT-22 primopredajnik radi na principu Dual Carrier Frequency sa upotrebom DSP procesora koji omogućava adaptivnu korekciju nosioca i podataka, otklanjanje impulsnog šuma. Ovaj primopredajnik komunicira upotrebom BPSK u frekventnim opsezima 125Khz-140Khz primarni i 110Khz-125Khz sekundarni. Brzina komunikacije može biti do 5kbs što je mnogo brže od X10 protokola.

3.4.4.3 CEBus

1984, Electronic Industries Alliance (EIA), je započela sa radom na formiranju jedinstvenog protokola za komunikaciju uređaja potrošačke elektronike u kući. Stoga je i sam protokol nazvan the Consumer Electronic Bus (CEBus). Specifikacija protokola uključuje komunikaciju posredstvom više medija kao što su energetske vodovi, ukrštene parice, koaksijalni kablovi, infra-red, radio i fiber optic. Set protokola koji pripada ovoj grupi je označen kao EIA-600, a svoju potpunu specifikaciju doživio je 1992. godine.

CEBus koristi non return to zero (NRZ) širinsko impulsnu modulaciju. Osnovna su četiri simbola : '1', '0', EOF i EOP. Ovi simboli se kodiraju upotrebom "chirp" spread spectrum tehnike u opsegu od 100 kHz do 400 kHz gdje se signal nosioca swipuje duž ovog frekventnog opsega. Uzlazni i silazni swip nosioca traje 100 μ s I to u slučaju kodiranje "1" za "0" to je 200 μ s. Za slučajnu sekvencu srednje trajanje je 150 μ s, za bit rate od 7.5 kbps. Izgled "chirp" za trajanje od 100 μ s je dato na slici 28 .



Slika 3.28: CEBus spektrum širenja “chirp”

3.4.4.4 HomePlug

HomePlug je ne-profitni konzorcium formiran 2000 od strane desetak vodećih IT kompanija zainteresovanih za razvoj “high-speed” PLC mreža. Njegovo članstvo se danas proširilo na oko 80 kompanija uključujući one koje se bave razvojem poluprovodničkih čipova kao i sistem integracijom. Cilj ovog konzorcijuma je da kreira jedan otvoreni standard za ovu svrhu. U junu 2001, HomePlug 1.0 je publikovan.

HomePlug je mnogo kompleksniji od gornjih tehnologija i koristi OFDM

(Orthogonal Frequency Division Multiplexing) tehniku u opsegu od 4.49 do 20.7 MHz podijeljeno na odgovarajuće pod opsege. U opsegu od 0 - 25 MHz, postoji prostor za 128 sub-nosioca.

Nekoliko proizvođača je testiralo datu tehnologiju i dobijeni rezultati su obećavajući. Npr. Testiranje u okviru 500 kuća je pokazalo da 80% instalacija može podržati brzine veće od 5 Mbps i 98% veće od 1 Mbps, što je već dovoljno za zahtjevne aplikacije kao što su AV (Audio Video). Pošto ovaj standard uključuje složene matematičke algoritme sa minimizacijom greške njegova realizacija je za sada dosta skupa, mada na srednjim brzinama već postaje konkurentno rešenje.

3.4.4.5 CENELEC

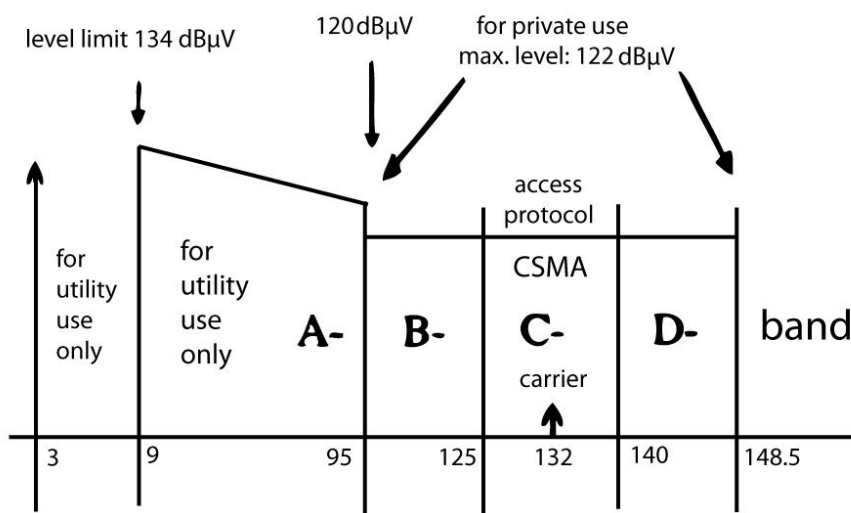
CENELEC EN 50-065 jeste Evropski standard za kućnu automatizaciju (dio European Home System – EHS) i koristi frekventni opseg 95kHz do 148.5kHz za privatne “narrow-band” PLC sisteme. Sistem kompatibilan sa ovim standardom u suštini samo zahtijeva pridržavanje ovog opsega bez obzira o kakvoj se modulaciji radi. CENELEC standard je prikazan na donjoj slici 3.29.

U okviru CENELEC protokola podaci se prenose u digitalnoj formi koja je mnogo više otporna na šum nego analogna. Neke od osnovnih modulacionih šema koje se primenjuju su Amplitude Shift Keying (ASK), Frequency Shift Keying (FSK), Minimum Frequency Shift Keying (MFSK) i Phase Shift Keying (PSK), dok je OFDM u procesu ispitivanja.

U okviru projekta koji je pratio izradu ovog rada korištena je MFSK modulacija iz dva razloga (i) zbog boljih performansi u odnosu na šum i (ii) zato što većina jeftinih integrisanih PLC modema koristi ovu modulaciju. Napomenimo da su najpoznatiji integrisani modemi koji podržavaju CENELEC standard (i) TDA5050 ASK power line modem, (ii) ST7537/8 FSK power line modems i (iii) AS5501/02 FSK power line modem. Mi smo se bazirali na ST7537/8 modemima iz razloga što je on i u pogledu frekvencije i modulacije najoptimalnije CENELEC rešenje

CENELEC EN 50 065

Signaling on Low-Voltage Electrical Installations



Slika 3.29: CENELEC standardi

PLC je standardizovani medijum EHS specifikacije. Definiše 2400 half duplex MFSK protocol. EHS PLC se poklapa sa CENELEC EN 50.065-1 i upotrebljava C-band sa CSMA (Carrier Sense Multiple Access). Centralna frekvencija je 132.5 kHz sa devijacijom za prenos podataka od 0.6 kHz. Logička "1" se reprezentuje sa donjom frekvencijom 131.9kHz, a logička "0" sa gornjom frekvencijom od 133.1 kHz. Kao što definiše CENELEC naponski opseg izlaznog modulisanog signala ne smije preći 116 dB.

GLAVA IV

4. Kućna automatizacija bazirana na IoT principu

4.1 Kratki pregled kućne automatizacije bazirane na IoT principu

Živimo u svijetu Interneta, gdje će se svaki fizički objekt kontrolisati a i biti povezan s internetom i na taj način komunicirati sa svima. Istraživanja pokazuju da će uređaji koji su povezani s Internetom stvari (IoT) instalirani širom svijeta od 2015. do 2025. godine iznositi 75,44 milijarde [13]. Pojavom softvera za prepoznavanje glasa, kao što je Amazon Alexa, porasla je potražnja za kućnom automatizacijom [14]. Glavni cilj je razviti način da se brzo, kvalitetano i ekonomski povoljno postojeće kuće pretvore u pametnu kuću, tj proizvesti i predložiti jeftini sistem s manje stvarne potrošnje energije temeljen na IoT-u. Upotrebom takvog sistema registrovana osoba može kontrolisati svoje kućne uređaje s bilo kojeg mjesta i bilo kada u realnom vremenu. Takođe, registrovana osoba može nadgledati sve svoje kućne električne uređaje. Prilagoditi privatni server za praćenje i kontrolu sistema. Server komunicira s ESP32 Wi-Fi modulom. Procjenom server na osnovu dobijene informacije, registrovana osoba može uključiti / isključiti svoje kućne uređaje. A kako je to server privatne kuće, on je takodje autonoman i tajan. Kućni server stvoren je na temelju OSI modela mrežne arhitekture. Kako se očekuje da će svjetsko tržište sistema kućne automatizacije narasti sa 32,11 milijardi USD u 2015. na 78,27 milijardi USD do 2022. godine, na CAGR od 12,46 između 2016. i 2022. pomoću IoT sistema [15]. Dakle, važno je osjetiti promjenu nadolazeće tehnološke ere i napraviti promjene u informatičkom životu. Zato će se u ovom radu predložiti sistem u kojem korisnik može koristiti ekonomičan sistem pametne kućne automatizacije.

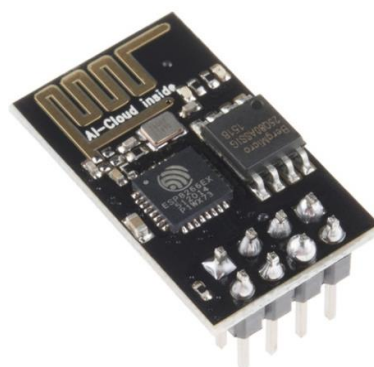
4.2 Pregled nekih od radova sistema kućne automatizacije baziranih na IoT

U ovom sistemu kućne automatizacije odvijaju se mnoga istraživanja. Autor sa svojim koautorima predstavio je sistem naprednog kombinovanog daljinskog praćenja zdravlja, kućne automatizacije i alarmnog sistema temeljenog na IOT-u, gdje su autori predstavili sistem za daljinsko zdravlje i kućnu automatizaciju s alarmnim sistemom [15,16]. Grupa istraživača predložili su sistem nazvan dizajn sistema za pametne kuće kontrolisane glasom koji se temelji na fonemu, gdje su autori istraživali različite vrste glasovnih signala i koristeći te glasovne signale predložili su sistem automatizacije kuće [17]. Autor predstavlja rad o sistem automatizacije pametnih kuća koji koristi IP, Bluetooth, GSM i Android [18]. Dva inženjera su predstavili sistem koji se naziva automatizacija domova korištenjem Edge Computing internet

of things, gdje su predstavili koncept Edge Computing IOT u sistemu kućne automatizacije [19]. Grupa autora je u ovom radu predstavila papirnati sistem kućne automatizacije zasnovan na IoT-u pomoću ličnog asistenta [20] . U ovom djelu je predložen sistem za kućnu automatizaciju koji koristi platformu za praćenje temeljenu na IoT-u, gdje su autori predstavili jedan osnovni koncept kako se automatizacija kuće može izvesti pomoću IoT-a [21]). I mnoga druga istraživanja predstavila su različite radove na ovom sistemu kućne automatizacije. Takođe, mnogi drugi istraživači još uvijek rade na ovom ogromnom polju.

4.3 Moderni sistemi na čipu i ploči za potrebe IoT kućne automatizacije

4.3.1 ESP8266



Slika 4.1 ESP8266 WiFi modul

ESP8266 WiFi modul samostalni je SOC sa integrisanim TCP / IP protokolom stek koji svakom mikrokontroleru može dati za pristup svoju WiFi mrežu, slika 4.1. ESP8266 je sposoban ili da hostuje aplikaciju ili da uradi offloading svih Wi-Fi mrežnih funkcija iz drugih aplikacija procesora.

Svaki modul ESP8266 isporučuje od prodavca se unaprijed programiranim s firmwareom skupa AT naredbi, što znači da se može jednostavno spojiti na svoj Arduino uređaj i dobiti otprilike kao mnogo WiFi mogućnosti kao što nudi WiFi Shield. Sve se to može uraditi praktično odmah.

Modul ESP8266 izuzetno je ekonomski isplativa ploča s velikim mogućnostima koje jos uvijek nijesu dostigle svoj vrhunac upotrebe. Ovaj modul ima dovoljno moćnu ugrađenu mogućnost obrade i skladištenja što mu omogućuje integrisanje sa sensorima i ostalim aplikacijama specifičnih uređaja putem svojih GPIO-a s minimalnim razvojem unaprijed i minimalnim opterećenjem tokom izvođenja. Njegov visoki stepen integrisanosti na čipu omogućuje za

minimalne spoljne sklopove, uključujući front-end modul, dizajniran je da zauzimaju minimalno površinu PCB-a.

ESP8266 podržava APSD za VoIP programe i Bluetooth interfejs za koegzistenciju, sadrži samokalibrisanu RF dozvolu radi u svim radnim uslovima i ne zahtijeva spoljne RF dijelove. Postoji gotovo neograničen broj informacija dostupnih za ESP8266. U sledećem naslovu Karakteristike mogu se vidjeti mnogi resursi koji će pomoći u korištenju ESP8266, i uputstva o tome kako transformisati ovaj modul u IoT.

4.3.1.1 Specifikacija ESP8266:

- 802,11 b/g/n
- Wi-Fi Direct (P2P), soft-AP
- Integrisani niz TCP/IP protokola
- Integrisani TR prekidač, balun, LNA, pojačalo za napajanje i odgovarajuća mreža
- Integrisani PLLS, regulatori, DCXO i jedinice za upravljanje napajanjem
- +19.5dBm izlazna snaga u režimu 802.11b mod
- Struja curenja <10uA
- 1MB Flash memorija
- Integrisani 32-bitni CPU sa malo napajanja može se koristiti kao procesor aplikacije
- SDIO 1.1 / 2.0, SPI, UART
- STBC, 1×1 MIMO, 2×1 MIMO
- A-MPDU & A-MSDU agregation & 0.4ms interval čuvara
- Buđenje i prenos paketa < 2ms
- Standby potrošnja energije < 1.0mW (DTIM3)

4.3.2 ESP32



Slika 4.2 ESP32 mikrokontroler

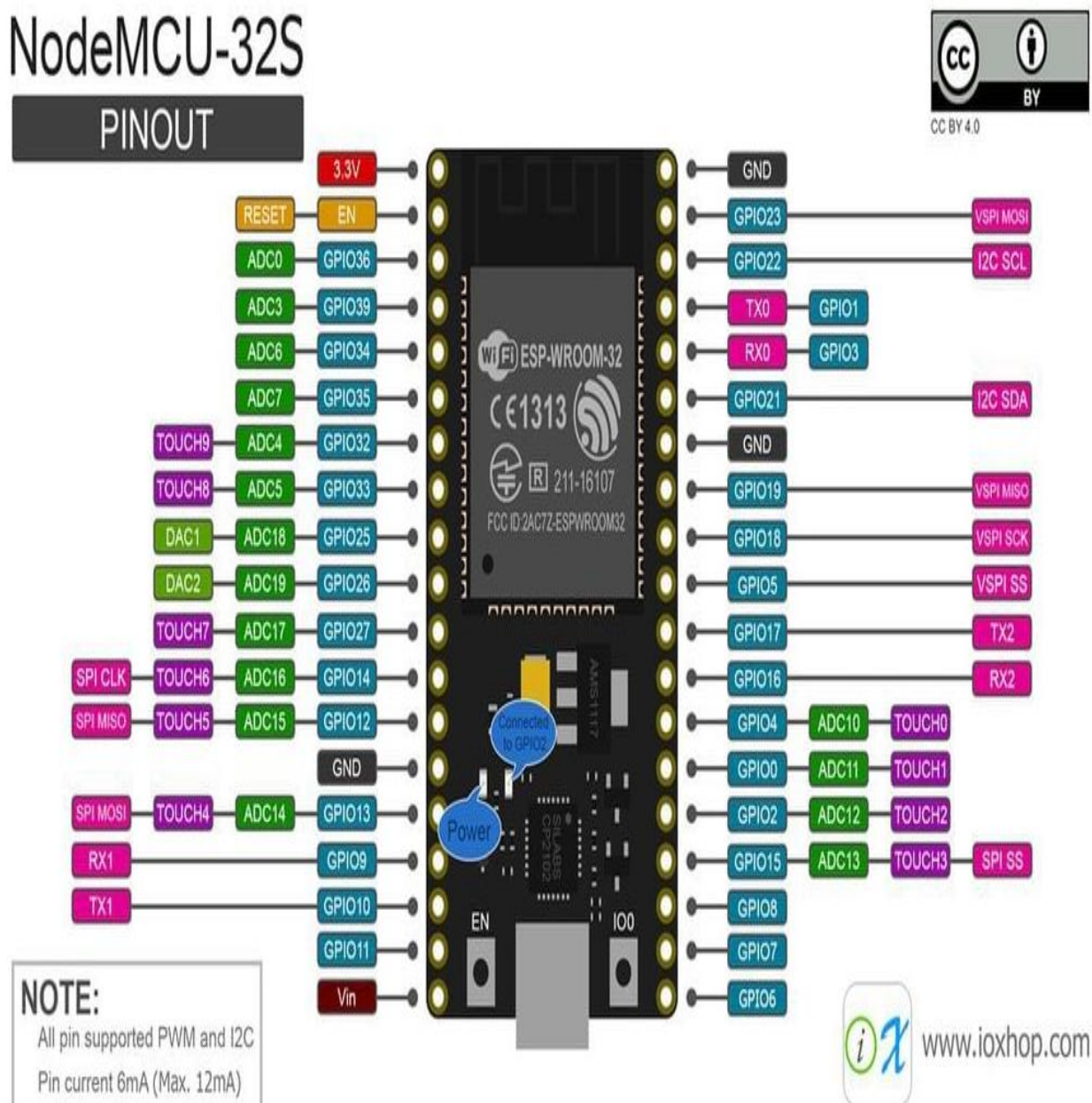
ESP32 je moćan 32-bitni mikrokontroler sa integrisanim Wi-Fi mrežom, punim TCP/IP nizom za internet konekciju i Bluetooth 4.2. prikazan na slici 4.2. Zbog niske cijene koštanja u kombinaciji sa velikom snagom i mogućnošću povezivanja ESP32 sa mnogim drugim elektronskim uređajima, mikrokontroler je veoma pogodan za IoT projekte.

4.3.2.1 Tehnička specifikacija ESP32

- TTensilica Xtensa 32-bitni LX6 mikroprocesor sa 2 jezgra
- Napajanje: 2.3V – 3.6V
- Trenutna potrošnja: 20 μ A – 240mA. In DeepSleep-Mode only 5 μ A
- Operativni temperaturni opseg: -40°C – 125°C
- Eksterna fleš memorija: podržano je do 16 MB
- Interfejsi
 - o UART/SDIO/SPI/I2C/I2S/IR daljinski upravljač
 - o 36 programibilnih U/I pinova max 20mA
 - o 2 analogni unos 0V do 1V sa 12 bitnom rezolucijom
 - o svi ulazi tolerišu maksimalno 3.6V
- Mreža
 - o WiFi
 - WiFi 802.11 b/g/n 2.4 GHz sa WPA/WPA2 PSK
 - ipv4 i ipv6 iz Arduino Core 2.5.0

- UDP i TCP sa 5 istovremenih veza maksimalno
 - Propusni opseg: 150 do 300 kByte/s
 - Latency: < 10ms
- o Bluetooth: v4.2 BR/EDR i Bluetooth Low Energy (BLE).

4.3.2.2 Pinout na ESP32 NodeMCU



Slika 4.3: Pinout na ESP32 NodeMCU

Raspored pinova i njihovo značenje je dato na Tabeli 4.3T

<i>NodeMCU oznaka</i>	Dodir	PWM	Ime	Opis
3.3V				Napajanje 3.3V, 400mA
EN			OMOGUĆITI	Low = nije napajanje High =omogućeno napajanje, vući 12kΩ
SVP		x		Pin za digitalni unos sa PWM-om
SVN		x		
P34		x		
P35		x		
P32	TOUCH9	x		Digitalni U/I pin sa senzorom osetljivim na dodir i PWM-om
P33	TOUCH8	x		
P25		x		Digitalni analogni konvertor (DAC_1)
P26		x		Digitalni konvertor (DAC_2)
P27	TOUCH7	x		Digitalni U/I pin sa dodirom i PWM-om
P14	TOUCH6	x		
P12	TOUCH5	x		
GND				Zemlja
P13	TOUCH4	x		Digitalni U/I pin sa dodirom i PWM-om
SD2		x		Digitalni U/I pin sa PWM-om
SD3		x		Napajanje 5V-9V, 400mA
CMD		x		
5V		x		Digitalni U/I pin sa PWM-om
CLK		x		
SD0		x		Digitalni U/I pin sa dodirom i PWM-om
SD1		x		
P15	TOUCH3	x		Digitalni U/I pin sa PWM-om
P2	TOUCH2	x		Zemlja

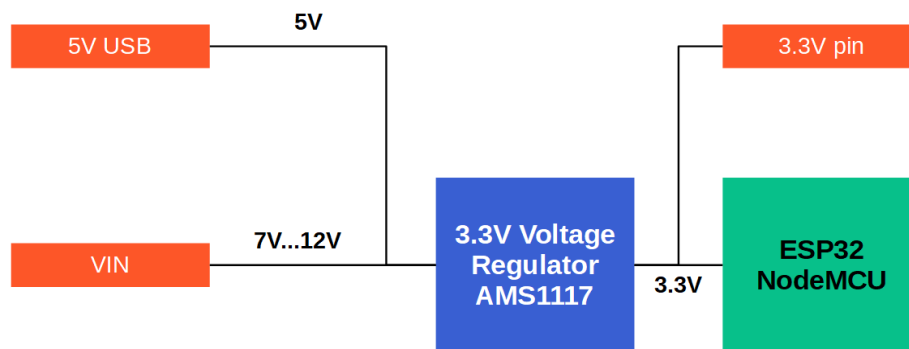
P0	TOUCH1	x		
P4	TOUCH0	x		
P16		x		Digitalni U/I pin sa dodirrom i PWM-om
P17		x		Digitalni U/I pin sa PWM-om
P5		x	SS	
P18		x	SCK	Digital I/O pin i SCK (Serijski sat) za SPI komunikaciju
P19		x	MISO	Digital U/O pin i MISO (Master In Slave Out) za SPI komunikaciju
GND				Zemlju
P21		x		Digitalni U/I pin sa PWM-om
RX		x		Serijski unos ESP-a povezanog sa USB-UART-om
TX		x		Serijski izlaz ESP-a povezan sa USB-UART-om
P22		x		Digitalni U/I pin sa PWM-om
P23		x	MOSI	Digital U/O pin i MOSI (Master Out Slave In) za SPI komunikaciju
GND				Zemlja

Tabela 4.1T: Raspored pinova na ESP32 NodeMCU

4.3.2.3 Napajanje ESP32 NodeMCU

Osnovna potreba je znati koje mogućnosti ESP32 ima za snabdijevanje strujom, tj koji su različiti nivoi napona na PCB-u, počevši od iglica za snabdijevanje i sa završetkom se na samom mikroprocesoru.

Sledeća slika 4.4 prikazuje šematsku sliku nivoa napona na ESP32 NodeMCU. Dodatne tabele prikazuju specifikacije mikroprocesora i regulatora napona.



Slika 4.4: Šema nivoa napona na ESP32 NodeMCU

Mikrokontroler	Minimalni napon	Tipičan napon	Maksimalan napon
ESP32	2.3V	3.3V	3.6V

Tabela4.2T Napon mikrokontrolera

Napon regulatora	Izlazni napon	Maksimalni ulazni napon	Maksimalna izlazna struja
AMS1117	3.3V	15V	1A

Tabela4.3T Napon regulatora

Prva i takođe najlakša mogućnost za napajanje je 5V USB kabl. Ali pošto ESP32 radi na 3.3V, postoji ugrađeno regulatorno tijelo za napon da transformiše 5V USB konekciju na željeni 3.3V. 3.3V pin NodeMCU PCB se takođe napaja iz ove veze.

Druga mogućnost je da se koristi VIN pin NodeMCU kao ulaz za napajanje. Regulator napona AMS1117 ima maksimalni ulazni napon od 15V. U ovom slučaju regulator proizvodi dosta toplotne energije jer regulator nema toplotnu sudoperu ili ventilator za hlađenje toplote. Zato se napon između 7V i 12V preporučuje kada se ESP32 napaja VIN pin-om.

Postoji i više opcija za napajanje ESP32 sa različitim vrstama isključenih baterija. Ovo je posebno interesantno kada se želi izgraditi projekat koji je nezavisan od normalnog napajanja, na primjer, meteorološka stanica na otvorenom. Ako je maksimalni napon baterije veći od

maksimalnog napona ESP32 (3,6V), mora da se koristi regulator napona da bi se smanjio napon na 3,3V. Izlaz regulatora napona je zatim povezan sa 3.3V pinom table ESP32.

Preporuka za napajanje baterije je LiFePO4 baterija, jer ne treba dodatni regulator napona između ESP32 i baterije i one se mogu više puta puniti. Takođe LiFePO4 baterije imaju kapacitet do 6.000mAh, slično LiPo i Li-jonskim baterijama. Takve baterije izgradjenom projektu daju dug vijek trajanja u kombinaciji sa režimom napajanja. Dodatno je korisno što smanjuju potrošnju struje na minimum.

4.3.2.4 Režimi napajanja i potrošnja energije ESP32

Režimi napajanja i potrošnja energije ESP32 su prikazani na Tabeli 4.4T .

Kada je Wi-Fi omogućen, čip se prebacuje između aktivnih i modemskih režima spavanja. Zbog toga se potrošnja energije mijenja u skladu sa tim.

RTC = Sat u realnom vremenu

ULP ko-procesor = Ultra-Low-Power Co-procesor

Aktivni režim

Radio sa čipovima je na programu. Čip može da prima, prenosi ili sluša.

Modem-režim spavanja

CPU je operativan i sat je konfigurisan. Wi-Fi/Bluetooth bazna traka i radio su onemogućeni. U režimu modema i spavanja, CPU učestalost se automatski menja. Učestalost zavisi od opterećenja procesora i perifernih uređaja koji se koriste.

Režim svetlosnog spavanja

CPU je pauziran. Rade RTC memorija i RTC periferni uređaji, kao i ULP ko-procesor. Svi događaji buđenja (MAC, host, RTC tajmer ili spoljni prekidi) probudiće čip.

Režim dubokog spavanja

Samo se napajaju RTC memorija i RTC periferni uređaji. Podaci o Wi-Fi i Bluetooth vezi skladište se u RTC memoriji. ULP ko-procesor je funkcionalan.

Režim hibernacije

Unutrašnji oscilator od 8 MHz i ULP ko-procesor su onemogućeni. RTC memorija za oporavak se napaja. Samo jedan RTC tajmer na sporom satu i određeni RTC GPIO-i su aktivni. RTC tajmer ili RTC GPIOs mogu da probude čip iz režima hibernacije.

Režim napajanja	Opis			Potrošnja struje
Aktivno (RF radi)	WiFi Tx paket			180mA ~ 240mA
	WiFi/BT Tx paket			130 mA
	WiFi/BT Rx i slušanje			95mA ~ 100mA
Modem-stanje spavanja	CPU se napaja	240 MHz	Čipovi dvostrukim jezgrom	30mA ~ 68mA
			Jedno jezgro čipova	N/A
	160 MHz	Čipovi dvostrukim jezgrom	27mA ~ 44mA	
		Jedno jezgro čipova	27mA ~ 34mA	
	80 MHz	Čipovi dvostrukim jezgrom	20mA ~ 31mA	
		Jedno jezgro čipova	20mA ~ 25mA	
Svetlo-spavanje			0.8mA	
Dubok san	ULP ko-procesor se napaja			150μA
	uLP senzorski praćeni obrazac			100μA @ 1% duty
	RTC tajmer + RTC memorija			10μA
Hibernacija	RTC tajmer samo			5μA
Isključeno napajanje	CHIP_PU je podešen na nizak nivo, čip se napaja.			0.1μA

Tabela 4.4T Režimi napajanja i potrošnja energije ESP32

4.4 Referentne IoT platforme

Migracija alata i aplikacija u oblak (cloud), rast društvenih mreža i nedavno širenje Internet of Things (IoT) platformi odveli su naše uređaje, usluge i podatke na internet. Iako ovi sistemi nude RESTful API³ (skup je programskih instrukcija za pristup web softverskoj aplikaciji) koji omogućavaju programerima da integrišu takve usluge zajedno kako bi obezbedili korisnu funkcionalnost ("Ako su vrata otključana, onda isključite rernu", "Alexa tell SmartThings to unlock the door", "Add new posts you like on Instagram to Dropbox"), krajnji korisnici nemaju lak način da kreiraju sopstvene integracije. Stoga vidimo pojavu nove klase programskih okvira koji nude jednostavan, ali efikasan model okidača koji krajnjim korisnicima omogućava da sami kreiraju korisne integracije. Korisnici povezuju okidače (npr. ako je brava na vratima angažovana) i radnje (npr. isključite rernu) zajedno da biste kreirali automatizovane zadatke. S obzirom na njihovu jednostavnost i efektivnost, istraživačka zajednica je istražila različite aspekte ovih modela [26–28].

Ovi okviri omogućavaju korisnicima da postignu takvu automatizaciju koristeći apstrakciju kanala. Kanal predstavlja funkcije usluge na mreži u okviru za delovanje okidača i koristi protokol OAuth⁴ autorizacije za bezbedno povezivanje sa RESTful API-jem usluge na mreži. Stoga kanali predstavljaju prostor funkcionalnosti koji je moguć u ovim programskim okvirima. Prethodni rad je istraživao prostor kombinacija okidača (ili recepata) koje je kreirao korisnik. Et al. je analizirao efektivnost programiranja okidača u pametnim domovima [28].

Et al. je takođe napravio snimak 200.000+ recepata IFTTT-a (If-This-Then-That) i okarakterisao prostor recepata koje je kreirao korisnik [29].

Nasuprot tome, prostor moguće funkcionalnosti u takvim okvirima karakterišemo analizom svojstava njihovih kanala. Fokusiramo se na IFTTT [22] i Zapier [23], dva popularna programska okvira za akciju okidača. IFTTT podržava jedan okidač i akcionu kombinaciju u svakom zadatku i više je fokusiran na potrošačke usluge (npr. Vijesti, društvene mreže) i uređaje (npr. kontrola životne sredine, aplikante) sa ciljem da budu jednostavni za korišćenje. Zapier [23] sa druge strane omogućava kreiranje složenijih zadataka i usmjerava svoju uslugu na poslovne potrebe kao što su upravljanje projektima i automatizacija marketinga.

³ (Rest stands for Representational State Transfer. To je arhitektonski stil i pristup komunikaciji koji se koristi u razvoju web usluga. REST je postao logičan izbor za izgradnju API-ja. Omogućuje korisnicima efikasno povezivanje i interakciju s uslugama u oblaku. API (Application Programming Interface)

⁴ (OAuth 2.0 je otvoreni protokol za autorizaciju koji omogućuje aplikacijama pristup međusobnim podacima. Na primjer, aplikacija za igre može pristupiti podacima korisnika u aplikaciji Facebook, ili aplikacija temeljena na lokaciji može pristupiti podacima korisnika aplikacije Foursquare itd.)

4.4.1 IFTTT vs. Zapier

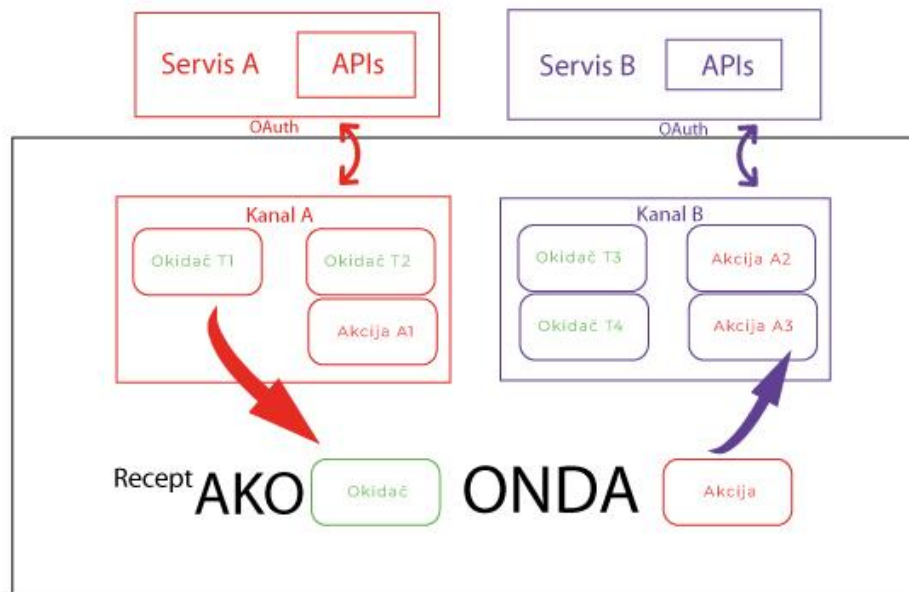
IFTTT je predmet obožavanja svih generacija mladih i starih koje koriste Internet. Potpuno je besplatan i vrlo jednostavan za upotrebu. Nije loše ponoviti da je IFTTT skraćena za 'If This-Then-That' i besprekorno opisuje koncept cloud automatizacije: na primjer, postavljenu fotografiju na Instagramu direktno prosljeđuje na Facebook feed, uz unaprijed definisan post uz upustva koje slijedeće korake treba preduzeti uz vrlo prilagodjen i dopadljiv interfejs.

Zapier, je stvoren za veće kompanije što ne znači uslovno rečeno i manje da ga ne mogu kristiti. Interfejs je komplikovaniji od IFTTT-evog, a i potrebno je platiti veći dio usluga koje pruža. Ovaj servis bez problema upravlja višestrukim accountima što nije slučaj sa IFTTT-om. Zapier nudi i veći broj servisa uz one uobičajene kao što su Twitter, Facebook, Gmail, Instagram i njima slični, tu su i neki manje poznati poput WooCommercea, Ducksboarda, TelAPI-a te FogBugza.

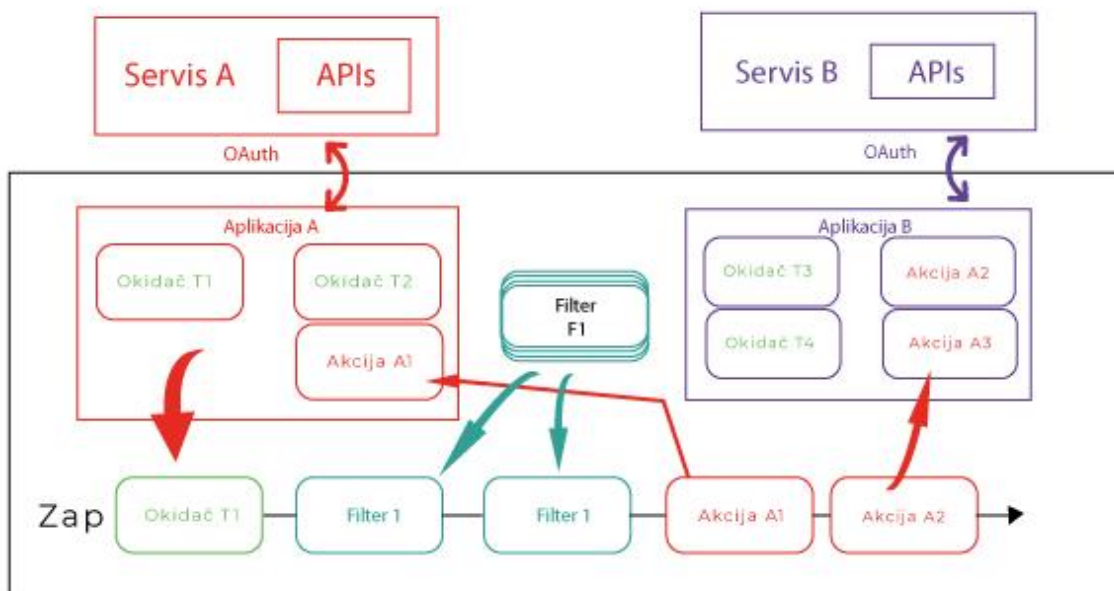
Cilj IFTTT i Zapier okvira je da obezbede jednostavan način za ne-programere da automatizuju aktivnosti na više platformi integriranjem njihovih funkcionalnosti. Stoga ova dva okvira dijele mnoge karakteristike dizajna. Slika 4.5 pruža pregled ovih okvira. Oba okvira se obično povezuju sa različitim uslugama i uređajima tako što prvo nabave oznaku OAuth [24]. Ovaj simbol omogućava ovim okvirima pristup API-jem koje pruža usluga i započinje zadatke bez dalje intervencije korisnika. Svaka usluga je predstavljena u ovim okvirima kao kolekcija funkcija. Ova apstrakcija je poznata kao kanal u IFTTT-u i aplikacija u Zapieru. U svakom kanalu postoje dva tipa funkcija: okidači i radnje. Okidač je događaj koji se dešava u povezanoj usluzi na mreži. "Datoteka je otpremljena na disk jedinicu u oblaku" ili "SmartLock je otključan" su primeri okidača. Radnja je operacija (ili skup operacija) koja postoji u API-u usluge na mreži. Primjeri radnji uključuju "uključivanje ili isključivanje povezane rene" ili "slanje SMS-a.

Oba okvira omogućavaju korisnicima da kombinuju okidače i radnje za definisanje različitih zadataka. Ovi zadaci su poznati kao Recepti u IFTTT i Zaps u Zapieru. IFTTT ograničava svaki zadatak na jedan okidač i radnju. Ovo obezbeđuje pojednostavljeni model koji je lako pratiti za krajnjeg korisnika. Zapier sa druge strane dozvoljava da se više radnji veže za jedan okidač. Ovo omogućava Zapieru da podrži složenije operacije kao što su pretrage i izmena postojećih podataka. Zapier takođe obezbeđuje niz filtera koji obezbeđuju dodatnu uslovnu (i/ili) kontrolu nad komponentama okidača (npr. ako "prijem e-poruke" deluje kao okidač, filteri mogu da uključuju "tema sadrži X" i/ili "je od Y"). Ovo omogućava mnogo bogatiju prilagodljivost, ali potencijalno može da preplavi tipičnog korisnika. Postoje i druge karakteristike koje čine sisteme drugačijim jedni od drugih. IFTTT obezbeđuje društvenu komponentu za svoje zadatke gde korisnici mogu međusobno da dijele svoje recepte i daje statistiku o tome koliko ljudi koristi

recept. Ove interakcije korisnika proučavao je vaš et al. [28]. Takođe obezbeđuje mobilnu aplikaciju i za Android i za iOS platforme koje omogućavaju da se mobilni servisi dodaju kao kanali. Zapier sa druge strane pruža programerima mogućnost da definišu sopstvene prilagođene kanale.



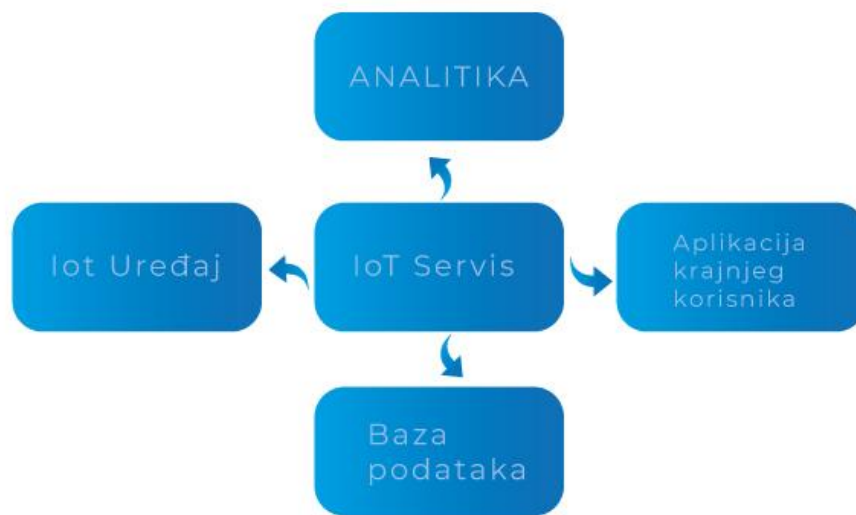
(a) IFTTT



(b) Zapier

Slika 4.5: Pregled dizajna IFTTT-a(a) i Zapiera(b). Dok dva dizajna dijele mnoge karakteristike (npr. definiciju kanala/aplikacija i način na koji se povezuju sa uslugama), oni nude različite nivoe prilagodljivosti u svojoj definiciji zadatka (npr. recepti/zaps).

Internet of Things(IoT) je sistem 'povezanih stvari', slika 4.6. Stvari se uglavnom sastoje od ugrađenog operativnog sistema i mogućnosti komunikacije sa internetom ili susjednim stvarima. Jedan od ključnih elemenata generičkog IoT sistema koji premosti razne 'stvari' je IoT servis. Interesantna implikacija od 'stvari' koje čine IoT sistemi je da stvari same po sebi ne mogu ništa da urade. Najosnovnija funkcija trebalo bi biti da imaju sposobnost da se povežu sa drugim 'stvarima'. Ali stvarna moć IoT-a je upregnuta kada se stvari povežu sa 'uslugom' bilo direktno ili preko drugih 'stvari'. U takvim sistemima usluga igra ulogu nevidljivog menadžera obezbeđivanjem mogućnosti koje se kreću od jednostavnog prikupljanja i praćenja podataka do složene analitike podataka. Dijagram ispod ilustruje gde se IoT usluga uklapa u IoT ekosistem:



Slika4.6: Internet of Things(IoT)

Jedna od takvih IoT platformi za aplikacije koja nudi veliki izbor analiza, praćenja i protivakcijskih mogućnosti je 'ThingSpeak'.

4.5 ThingSpeak

ThingSpeak je platforma koja pruža razne usluge isključivo usmjerene na izgradnju IoT aplikacija. ThingSpeak nudi mogućnosti prikupljanja informacija u realnom vremenu, obradu istih i vizuelizaciju prikupljenih podataka u obliku dijagrama. Takođe pruža mogućnost kreiranja dodatnih komponenti i aplikacija za saradnju sa veb uslugama, društvenom mrežom i drugim API-jem.

Osnovni element ThingSpeak-a je 'ThingSpeak Channel'. Kanal skladišti podatke koje prikupljamo u ThingSpeak i sastoji se slijedećih elemenata:

- 8 polja za skladištenje podataka bilo kog tipa – Polja se koriste za skladištenje podataka dobijenih iz senzora ili sa nekog ugrađenog uređaja.
- 3 polja lokacije - Može se koristiti za skladištenje geografske širine, dužine i nadmorske visine. Veoma korisno za praćenje uređaja za selidbu, putovanja, razne sportske aktivnosti i sl.
- 1 polje statusa - Kratka poruka za opisivanje podataka uskladištenih u kanalu.

Da bi se koristio ThingSpeak, mora se izvršiti prijava i onda napraviti kanal. Kada je oformljen kanal, mogu da počnu da se šalju podaci, što omogućava ThingSpeak-u da ih preuzima i obradjuje.

4.5.1 ThingSpeak aplikacije

ThingSpeak obezbeđuje aplikacije koje mogu na lak način da se integrišu sa veb uslugama, društvenim mrežama i drugim API-jem. Neke od aplikacija koje je obezbedio ThingSpeak su:

- ThingTweet – Integracija ThingSpeak sa twitterom omogućava da se objavljuju poruke . Ovo u suštini predstavlja TwitterProxy koji ponovo usmjerava postove na twitter.
- ThingHTTP - povezivanje sa Veb uslugama i podržava metode GET, PUT, POST i DELETE HTTP.
- TweetControl – Ostvaruje kontrolu, tj može se nadgledati sopstveni Twitter feedovi za određenu ključnu riječ, a i obraditi zahtjev. Kada se određena ključna riječ nađe u twitter feedu, može da se koristi ThingHTTP da bi se ostvarila povezanost sa drugom Veb uslugom ili izvršila određena radnja.
- React - Slanjem tvita ili aktiviranjem ThingHTTP zahtjeva Channel izvrši reakciju na određeni uslov .
- TalkBack - Ova aplikacija omogućava da se komande stavljaju u red, a zatim dozvoljava uređaju da odradjuje po ovim komandama u redu po kojem su postavljene u čekanje.
- Timecontrol - Upotrebom ove aplikacije, može da se odradi ThingTweet, ThingHTTP ili TalkBack u zakazano vrijeme u budućnosti. Takođe može da se koristi da bi se omogućilo da se u određeno vrijeme tokom cijele nedelje navedene radnje dešavaju i ispunjavaju tačno na vrijeme.

Pored navedenog, ThingSpeak omogućava da se kreira ThingSpeak aplikacija kao plugin-ove koristeći HTML, CSS i JavaScript koje mogu ugraditi unutar veb sajta ili unutar postavljenog ThingSpeak kanala.

4.6 Kratak rezime komercijalnih IoT platform

Jedan od ključnih elemenata IoT sistema je IoT usluga. ThingSpeak je jedna takva platforma za aplikacije koja nudi najrazličitije funkcije. U samom ThingSpeak-a je kanal koji se može koristiti za skladištenje i obradu podataka prikupljenih iz popularno rečeno iz 'stvari'. ThingSpeak takođe obezbeđuje razne aplikacije za integraciju sa veb uslugama, drugim API-jem i društvenim mrežama i pruža mogućnost kreiranja aplikacija kao pluginova. To je odlična platforma sa velikim mogućnostima za istraživanje integracije Interneta stvari.

IFTTT, vodeća platforma za integraciju sa 670 popularnih brendova i koju koristi preko 22 miliona potrošača, danas je najavila IFTTT Pro, novu pretplatu koja prevazilazi IFTTT-ovaj original If This Then That koncept. Pro takođe uključuje uvodnu pretplatu "odredi sopstvenu cijenu", koja se sada pridružuje IFTTT-tom standardnom free-forever planu. Od 2010. godine, IFTTT nudi jednostavan način potrošačima da kombinuju popularne brendove i usluge u svom životu, u jedinstvene integracije i automatizacije, poznate kao Apleti. Pro sada prevazilazi jedan okidač, jednu radnju koja nudi višestepenih apleta koji mogu da ispituju podatke iz više izvora pre nego što pokrenu više radnji. Na primer, Pro može da kreira aplet koji svake večeri ispituje njihov Google Calendar i Slack, pre nego što utvrdi da li da uključi svoja Philips Hue svetla i reprodukuje njihov omiljeni Spotify spisak numera. Pro pokreće sa mnogim od najboljih funkcija koje su tražili IFTTT-ti korisnici napajanja, uključujući:

- Višestepeni apleti
- Upiti i uslovna logika
- Više radnji
- Brže izvršenje apleta
- Podrška

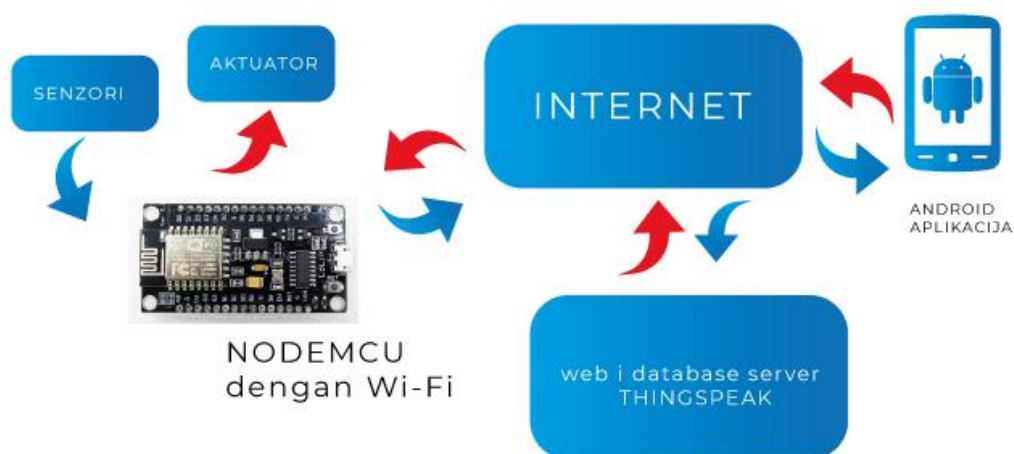
"Pro-ove moćne nove funkcije prvi put su razvijene za naš preduzetnički proizvod, IFTTT Connect, koji smo lansirali još u januaru", objasnio je Linden Tibbets, osnivač i CEO kompanije IFTTT. "Profesionalci sada imaju istu sposobnost da stvore sofisticirane integracije kao inovativni IFTTT Connect klijenti kao što su Husqvarna, iRobot i Qapital. Uspjeh poslovanja našeg preduzeća omogućava nam da se vratimo svojim korjenima i korisničkoj zajednici, sa Pro

predstavlja polaznu tačku za ono što će biti tekući razgovor. Naša namjera je da budemo fleksibilni u smislu funkcija, pa čak i cijena kako se Pro razvija tokom vremena."

IFTTT-ove rešenja mogu globalna preduzeća da iskoriste za povezivanje i prerastu svoje proizvode u integrisane i suštinske usluge, dramatično smanjujući njihove troškove razvoja, istovremeno povećavajući kompatibilnost i doživotnu vrednost. IFTTT je standard povezivanja i alternativa niskom kodu za izgradnju sopstvenih integracija u kući. Preduzeća kao što su Amazon, Bosch, Google, Husqvarna, ING, iRobot i Samsung veruju IFTTT-u za svoja rešenja za integraciju.

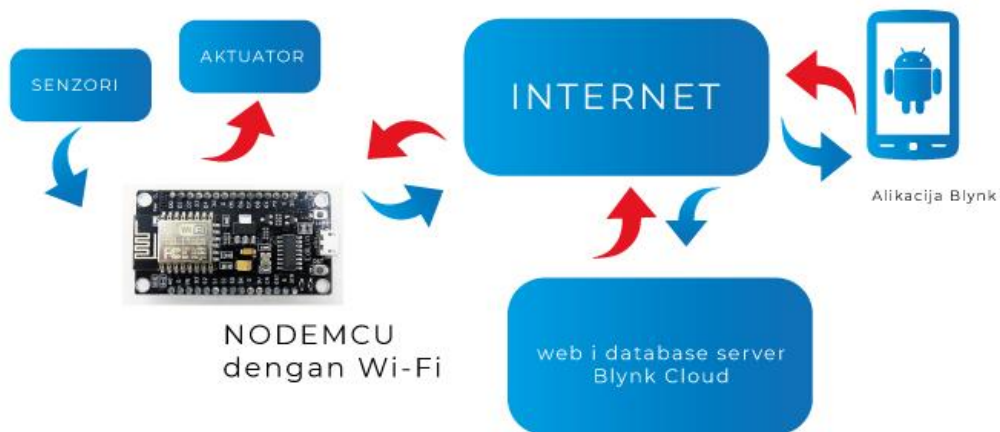
4.7 Princip projektovanja IoT rešenja za kućnu automatizaciju bazirano na nodeMCUs

Projektovanje ili dizajniranje se radi fazno . Prva faza je izrada samog hardvera a druga izrada potrebnog softvera. Dizajn hardvera i dizajn softvera. Dizajn hardvera je priprema hardverskih komponenti koje će biti iskorišćene kao što su NodeMCU, Wi-Fi modul ESP8266-01, senzori i aktuatori itd. Dizajn softvera je u obliku projektovanja programa (kodova) za Arduino. Dizajniranje tj pravljenje same Android aplikacije koja je sve za razvoj medija. Sledeće što je potrebno je dizajnirati laboratorijski list i kratak priručnik za podršku radu IoT rešenja. Za hardverski modul koji je zasnovan na NodeMCU a koristi se ThingSpeak onda je potrebno dizajnirati hardverski modul koji će realizovati strujno kolo sa sensorima, aktuatorima, mikrokontrolerom (uključen Wi-Fi modula ESP8266), koji se može prikazati na slici 4.7.

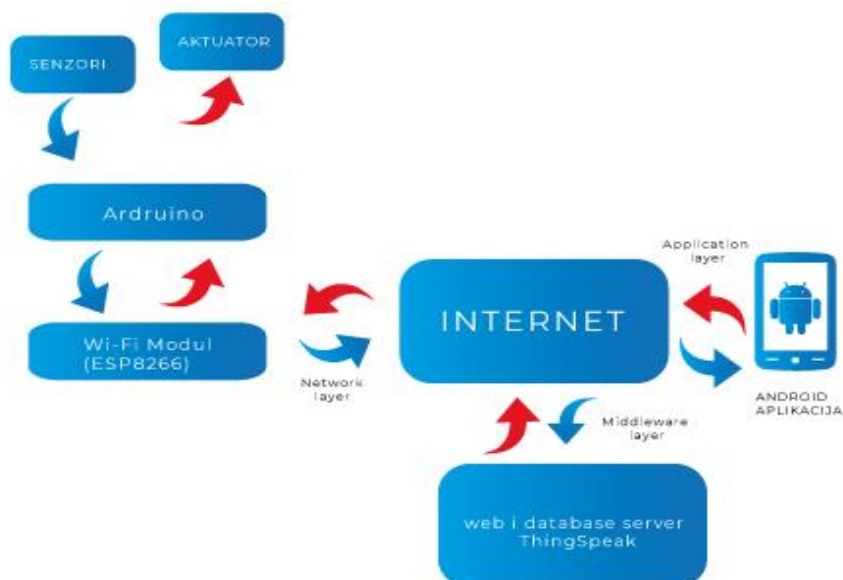


Slika 4.7. Dizajn hardverskog modula zasnovan na NodeMCU sa ThingSpeak-om

Podatke koji dolaze sa senzora će dobiti NodeMCU, a zatim će biti poslani ThingSpeak-u bazu podataka u oblaku. Ako postoji pristup internet onda svako može da pristupi i vidi podatke sa ovih senzora, bilo gdje i u bilo kom trenutku. To se može uraditi pomoću Android pametnog telefona, Android tableta ili bilo kog Android uređaja. Android korisnici mogu da kontrolišu aktuatoru pomoću slanja kontrolnih podataka u ThingSpeak pomoću pametnog telefona. Pored toga, podatke će pročitati NodeMCU putem hotspot Wi-Fi mreže, koja će se zatim koristiti za kontrolu aktuatora. Android aplikacija za čitanje i pisanje na ThingSpeak je izgrađena pomoću MIT App Inventor. Koristeći Blynk aplikaciju srednjeg softvera, šema hardverskog modula zasnovanog na NodeMCU-u može biti nacrtana kao što je prikazano na slici 4.8 . Blynk će objediniti programiranje u srednjem softveru, aplikaciji a dijelom u mrežnom sloju. Blynk je način za povezivanje između fizičke pločice i hardvera i mobilne aplikacije. Tada putem mobilne aplikacije se može upravljati pločicom (npr. zadati temperaturu) ili očitavati podatak sa nje, izmjerenu temperature.



Slika 4.8. Dizajn hardverskog modula zasnovan na NodeMCU sa Blynk aplikacijom



Slika 4.9 Dizajn hardverskog modula baziranog na Arduino i ESP8266 za ThingSpeak

Dizajn hardverskog modula zasnovanog na Arduino i ESP8266-01 sa ThingSpeak je kao što je prikazano na slici 4.9. To je principu isto kao hardverski modul zasnovan na NodeMCU (Slika 4.7), samo u ovoj konfiguraciji postoje mikrokontroler i razdvojeni Wi-Fi Modul

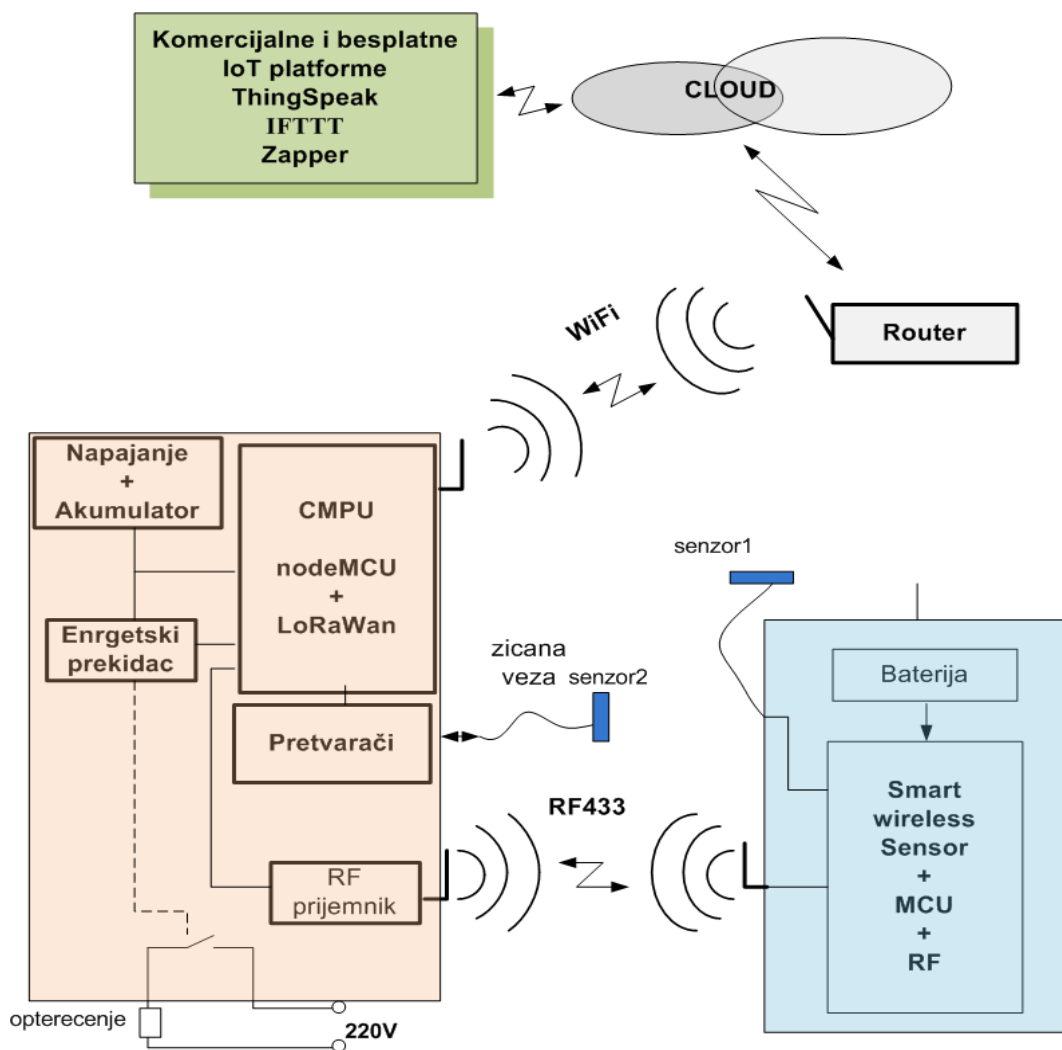
Postoji još i šema za hardverske module zasnovane na Arduino i ESP8266 sa Blynk aplikacijom koja je ista kao šema na slici 4.8, samo za komponentu NodeMCU se zamenjuje Arduinoom i ESP8266 Wi-Fi moduli.

Glava V

5. Predloženo rješenje kućne automatizacije na bazi Sistema na čipu i dostupnih platformi

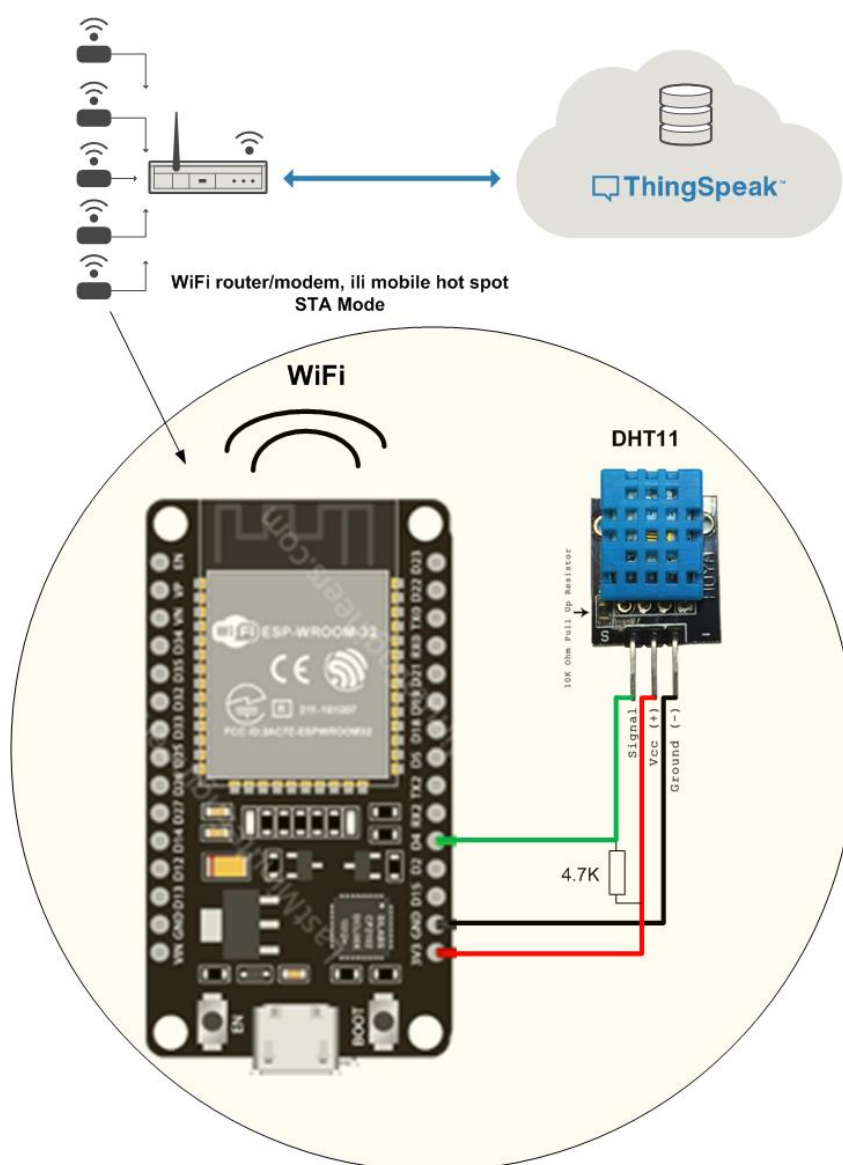
5.1 Uvod

Razvoj savremene tehnologije mikroprocesora i komunikacija omogućava implementaciju jeftinih i veoma efiksanih sistema kućne automatizacije. Centralna procesorska jedinica može biti realizovana pomoću NodeMCU kontrolera, dok senzori mogu žičano ili bežično komunicirati sa njom. Jedan od takvih scenarija je predložen i u ovom radu. Njegova blok šema je prikazana na Slici 5.1.



Slika 5.1: Predložena šema sistema kućne automatizacije

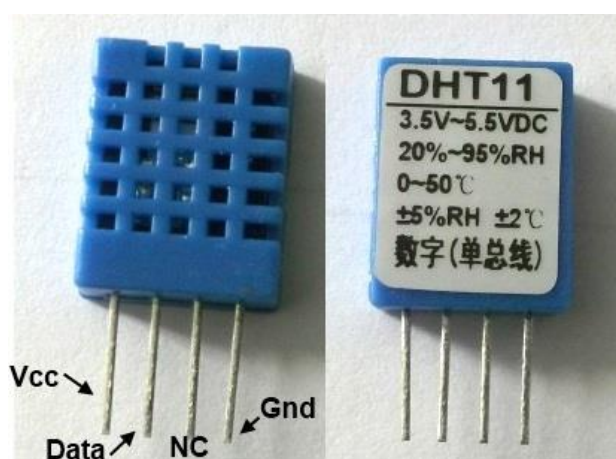
CMPU predstavlja **Centralnu Mikroprocesorsku Jedinicu (Central Microprocessor Unit)** baziranu na nodeMCU ESP32 koji je opisan u gornjem poglavlju. Kao što je rečeno ovaj sistem na čipu ima sve karakteristike savremenih mikrokontrolera uz mogućnost različitih oblika komunikacije od serijske RS232 komunikacije pa do BLE (BlueTooth Low Energy). Senzori sa CMPU mogu komunicirati žičano ili bežično. CMPU sa kućnim WiFi ruterom (spotom) komunicira preko WiFi konekcije, STA (Station Mode) modu, a potom preko klada sa serverom na kojem je smještena IoT platforma. To može biti neka od komercijalnih IoT platformi kakve su ThingSpeak, IFTTT, Zapper ili pak IoT platforme dizajnirane prema zahtjevima korisnika.



Slika 5.2: Sistem kućne automatizacije baziran na ESP32, NodeMCU i senzoru temperature DHT11

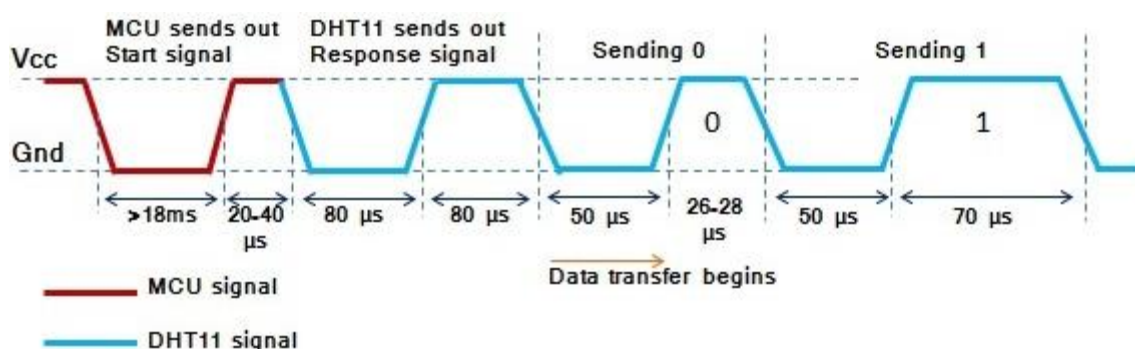
5.2. DHT11 Senzor

Temperatura i relativna vlažnost vazduha u domu i na radnom mestu znatno utiču na kvalitet života i produktivnost. Prvi korak ka optimizaciji ova dva parametra ambijenta je njihovo pouzdano merenje. DHT11 služi za paralelno merenje temperature i vlažnosti vazduha. To je trenutno najjeftiniji senzor dostupan na tržištu koji pruža kalibrisane digitalne izlaze za temperaturu i relativnu vlažnost vazduha, slika 5.3. Dostupan je u 4-pinskom kućištu, a radi pri naponima između 3,5 V i 5,5 V. U stanju je da meri temperaturu u opsegu od 0° do 50°C sa preciznošću od $\pm 2^\circ\text{C}$, a vlažnost vazduha u opsegu od 20-95% sa preciznošću od $\pm 5\%$, što ga čini pogodnim za unutrašnju upotrebu. Senzor ima svoj 1-wire protokol kojim komunicira sa mikrokontrolerom.



Slika 5.3: Senzor DHT11 za mjerenje temperature i vlažnosti

Vremenski dijagram, slika 5.4, prikazuje protokol kojim senzor komunicira sa mikrokontrolerom. Crvenom bojom prikazan je signal sa mikrokontrolera, a plavom signal sa senzora.



Slika 5.4. Vremenski dijagram protokola za komunikaciju

Mikrokontroler inicira prenos podataka slanjem signala “start” i ukoliko je ostvarena komunikacija, dobija odzivni signal. Treba napomenuti da nakon slanja signala “start”, pin mikrokontrolera mora biti rekonfigurisan kao ulazni pin. Kada jednom dobije odziv sa senzora mikrokontroler postane spreman za prijem signala. Senzor tada šalje 5 bajtova podataka kontinualno (kod ovih bajtova, prvi bit je najznačajniji). Ovih pet bajtova podataka se sastoje od: Podaci = Cijeli broj RV + Decimalni broj RV + Cijeli broj Temp. + Decimalni broj Temp. + bajt provjere

Kod DHT11 senzora decimalni brojevi temperature i vlažnosti su uvek jednaki nuli (rezolucija pri mjerenju temperature je 1°C, a pri merenju relativne vlažnosti 1%). Stoga, prvi i treći bajt zapravo daju numeričke vrednosti izmerene relativne vlažnosti vazduha i temperature. Poslednji bajt je bajt za proveru koji govori o tome da li je prenos podataka obavljen bez grešaka. Ako su prva četiri bajta preneti uspešno, tada će poslednji bajt biti jednak sa poslednjih osam bitova sume prva četiri bajta.

DHT11 je na ESP32 modul spojen kao na slici 5.2.

Za čitanje sa DHT senzora koristićemo DHT.h biblioteku od Adafruit-a. Da bismo koristili ovu biblioteku, potrebno je instalirati biblioteku Adafruit Unified Sensor. Slijede koraci softverske podrške rada DHT senzora na ESP32

5.3 HTTP zahtjevi: GET i POST

Hypertext Transfer Protocol (HTTP) je dizajniran da omogući komunikaciju između klijenta i servera. HTTP radi kao protokol zahtjeva i odgovora između klijenta i servera. Npr. klijent (pregledač) šalje HTTP zahtjev serveru, a server klijentu vraća odgovor. Odgovor sadrži informacije o statusu zahteva i može sadržati i određeni traženi sadržaj.

U našem slučaju klijent je ESP32, a host IoT server sa odgovarajućim API.

- ESP32 (klijent) šalje HTTP zahtev web serveru;
- Server vraća odgovor ESP32 (klijentu);
- Odgovor sadrži informacije o statusu zahtjeva i može sadržati i traženi sadržaj.

5.3.1 HTTP GET

GET se koristi kao zahtjev za prijem podataka podataka sa određenog servera. GET je jedna od najčešćih HTTP metoda. Potrebno je napomenuti da se URL , zahtjev i parametri sadrže u jednom stringu, npr-

```
GET /update-sensor?temperature=value1
```

Ili jednostavno možemo zahtijevati vraćanje JSON objekta, npr:

```
GET /get-sensor
```

Karakteristike GET zahtjeva:

GET zahtjevi se mogu keširati

GET zahtjevi ostaju u historiji browsera

GET zahtjevi se mogu obeležiti

GET zahtjevi se ne koriste kada se radi sa osjetljivim podacima

GET zahtjevi imaju ograničenje dužine

GET zahtjevi se koriste samo za traženje podataka ne izmjenu-promjenu

5.3.2 HTTP POST

POST se koristi za slanje podataka na server za kreiranje /ažuriranje resursa. Podaci poslani serveru sa POST-om čuvaju se u tijelu HTTP zahtjeva:

```
POST /update-sensor HTTP/1.1
Host: example.com
api_key=api&sensor_name=name&temperature=value1&humidity=value2&pressure=value3
Content-Type: application/x-www-form-urlencoded
```

Ili

```
POST /test/demo_form.php HTTP/1.1
Host: example.com
name1=value1&name2=value2
```

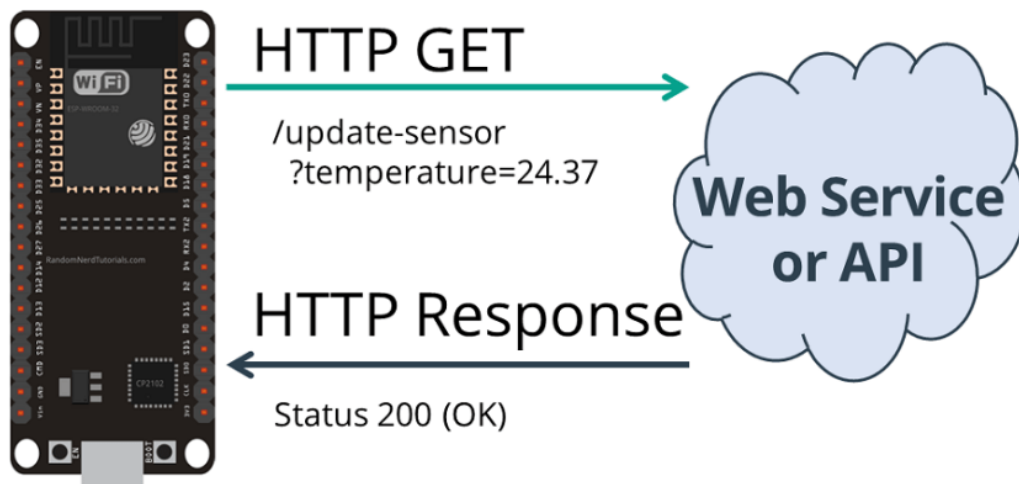
U tijelu zahtjeva može se poslati JSON objekat:

```
POST /update-sensor HTTP/1.1
Host: example.com
{api_key: "api", sensor_name: "name", temperature: value1, humidity:
value2, pressure: value3}
Content-Type: application/json
```

5.3.3 HTTP GET/POST sa ESP32

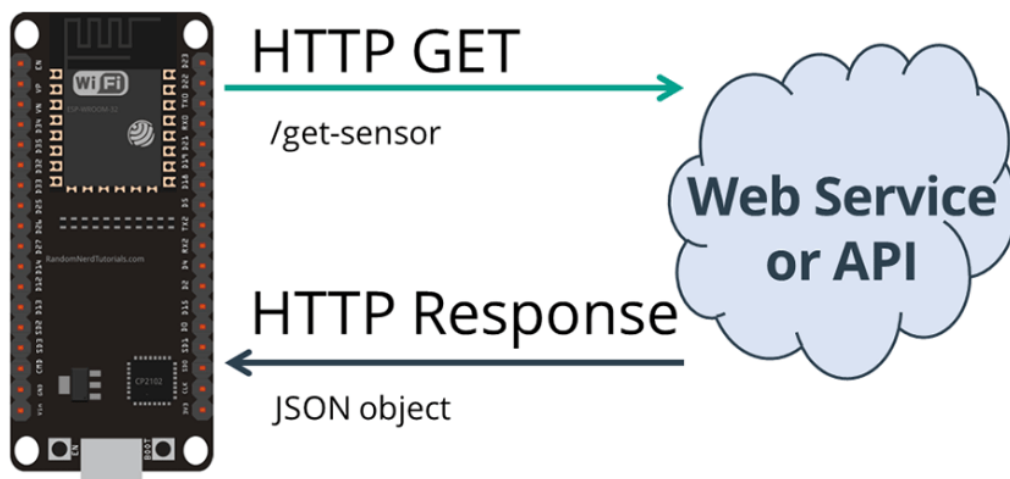
5.3.3.1 ESP32 HTTP GET: preko URL-a i JSONa

URL(Uniform Resource Identifier) je ustvari web adresa., oblika http ili https. U ovom slučaju ESP32 upućuje HTTP GET zahtjev prema serveru. Ova vrsta zahtjeva takođe se može koristiti vraćanje JSON objekta.



Slika 5.5 HTTP GET preko URL-a

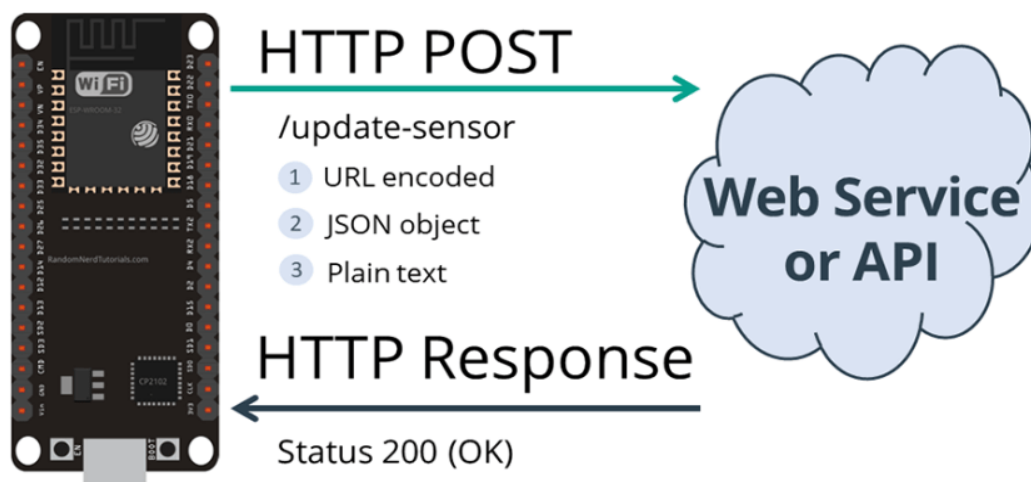
Sledeći primjer pokazuje kako napraviti HTTP GET zahtev da biste dobili JSON objekat i dekodirali ga pomoću ESP32. Mnogi API-ji vraćaju podatke u JSON formatu.



Slika 5.6 HTTP GET preko JSON

5.3.3.2 ESP32 HTTP POST: URL Encoded, JSON Data Object, Plain Text

Sa ovim primjerom, ESP32 može da postavlja HTTP POST zahtjeve koristeći tri različite vrste tjelesnih zahtjeva: kodirani URL, JSON objekat ili običan tekst. To su najčešće metode i treba da se integrišu sa većinom API-ja ili veb usluga.



Slika 5.7 HTTP POST: URL Encoded, JSON Data Object, Plain Text

5.3.3.3 HTTP POST URL Encoded

HTTP POST zahtjev tipe URL enkodiran je sledeci.

```
POST /update-sensor HTTP/1.1
Host: 192.168.1.106:1880
api_key=tPmAT5Ab3j7F9&sensor=BME280&value1=24.25&value2=49.54&value3=1005.14
Content-Type: application/x-www-form-urlencoded
```

Sa sledećim oblikom ESP32 koda:

```
// Your Domain name with URL path or IP address with path
http.begin(serverName);

// Specify content-type header
http.addHeader("Content-Type", "application/x-www-form-urlencoded");

// Data to send with HTTP POST
```



```
String httpRequestData =  
"api_key=tPmAT5Ab3j7F9&sensor=BME280&value1=24.25&value2=49.54&value3=1005.14";  
  
// Send HTTP POST request  
int httpResponseCode = http.POST(httpRequestData);
```

5.3.3.4 HTTP POST JSON Object

HTTP POST zahtjev u formi JSON objekta je sledeći:

```
POST /update-sensor HTTP/1.1  
Host: example.com  
{api_key: "tPmAT5Ab3j7F9", sensor_name: "BME280", temperature: 24.25;  
humidity: 49.54; pressure: 1005.14}  
Content-Type: application/json
```

Koristite sledeći isječak:

```
http.addHeader("Content-Type", "application/json");  
  
int httpResponseCode =  
http.POST("{\"api_key\":\"tPmAT5Ab3j7F9\", \"sensor\":\"BME280\", \"value1\": \"24.25\", \"value2\": \"49.54\", \"value3\": \"1005.14\"}");
```

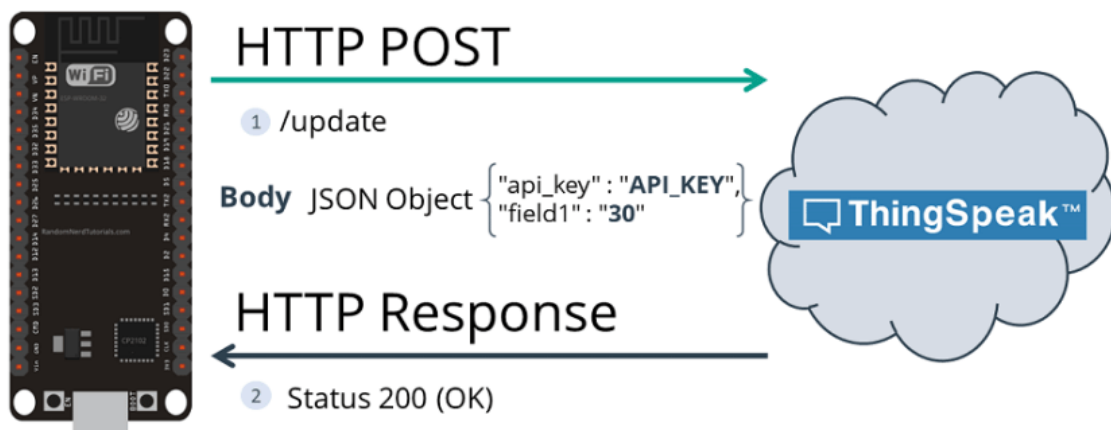
5.3.3.5 HTTP Plain Text

Ako želimo da šaljemo “plain text” ili vrijednost, onda kod izgleda na sledeći način:

```
http.addHeader("Content-Type", "text/plain");  
  
int httpResponseCode = http.POST("Hello, World!");
```

5.4 Slanje podataka sa ESP32 na ThingSpeak pomoću HTTP protokola

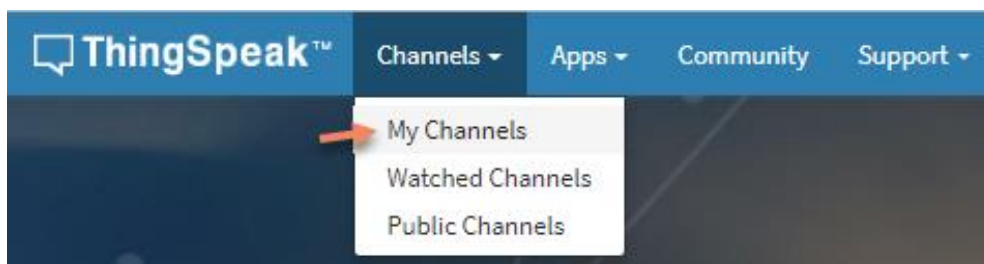
ESP32 šalje vrijednost na ThingSpeak server pomoću HTTP POST zahtjeva. ThingSpeak ima besplatni API koji nam omogućava da šaljete ili preuzimate podatke pomoću HTTP-a. Naravno, poslati podaci mogu biti vizuelizovani u obliku grafika. Kao primjer na server ćemo poslati slučajne vrijednosti, ali u stvarnoj aplikaciji koristimo stvarno očitavanje senzora.



Slika 5.8 HTTP POST zahtjev za Thing Speak

Da biste koristili ThingSpeak API, potreban je API ključ (API key), koji je pridružen svakom formiranom kanalu. Treba slijediti sledeće korake da biste kreirali kanall.

1. Sign In na ThingSpeak™ koristeći MathWorks® account.
2. Kliknuti **Channels > MyChannels**.



3. Na Channels stranici, odabrati **New Channel**.
4. Check boxove za Fields 1–3 i setovati ih na sledeći način:
 - **Name:** Dew Point Measurement
 - **Field 1:** Temperature (F)
 - **Field 2:** Humidity
 - **Field 3:** Dew Point

New Channel

Name

Description

Field 1

Field 2

Field 3

Field 4

Show Video

YouTube
 Vimeo

Video URL

Show Status

5. Kliknuti na dugme **Save Channel** na dnu **Settings**.

Onda vidjeti ove kartice:

- **Private View** :Ova kartica prikazuje informacije o vašem kanalu koje samo vi možete da vidite.
- **Public View**: Ako odaberete da kanal učinite javno dostupnim, koristite ovu karticu da biste prikazali izabrana polja i vizuelizacije kanala.
- **Channel Settings** : Ova kartica prikazuje sve opcije kanala koje ste podesili na kreiranju. Kanal možete da uređujete, brišete ili brišete sa ove kartice.
- **Sharing** : Ova kartica prikazuje opcije deljenja kanala. Kanal možete da postavite kao privatn, dijeljen sa svima (javnim) ili da ga dijelite sa određenim korisnicima.
- **API Keys** : Ova kartica prikazuje API tastere kanala. Koristite tastere za čitanje i pisanje na kanalu.

- **Data Import/Export** :Ova kartica vam omogućava uvoz i izvoz podataka kanala.

Kanal je dostupan za buduću upotrebu tako što se klikne na dugme **Channels >My Channels**

5.4.1 API zahtjevi (requests) za ThingSpeak

Write a Channel Feed

```
GET  
https://api.thingspeak.com/update?api_key=E02JL30LMWQP1GDD&field1=0
```

Read a Channel Feed

```
GET https://api.thingspeak.com/channels/1171888/feeds.json?results=2
```

Read a Channel Field

```
GET  
https://api.thingspeak.com/channels/1171888/fields/1.json?results=2
```

Read Channel Status Updates

```
GET https://api.thingspeak.com/channels/1171888/status.json
```

Channel ID: **1171885**

api_key=E02JL30LMWQP1GDD

5.4.2 Cod za ESP32 baziran na HTTP POST i ThingSpeak server

Koriste se biblioteke WiFi.h i HTTPClient.h.

```
#include <WiFi.h> //ukljucivanje potrebnih biblioteka  
#include <HTTPClient.h>  
  
const char* ssid = "SSID home wi-fi";  
const char* password = "wi-fi PASSWORD";  
  
// Domain Name sa punim URL putem za HTTP POST Request  
const char* serverName = "http://api.thingspeak.com/update";  
// Service API Key  
String apiKey = "API_KEY";
```

```

//Delay konstante
unsigned long lastTime = 0;
unsigned long timerDelay = 10000;

void setup() {
  Serial.begin(115200);
  WiFi.begin(ssid, password);
  Serial.println("Connecting");
  while(WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.print("Connected to WiFi network with IP Address: ");
  Serial.println(WiFi.localIP());
}

void loop() {
  //Salji HTTP POST zahtjev svakih 10 sekundi
  if ((millis() - lastTime) > timerDelay) {
    //Check WiFi connection status
    if(WiFi.status()== WL_CONNECTED){
      HTTPClient http;

      // Domain URL path ili IP address path
      http.begin(serverName);

      // Specify content-type header
      http.addHeader("Content-Type", "application/x-www-form-
urlencoded");
      // Data to send with HTTP POST
      String httpRequestData = "api_key=" + apiKey + "&field1=" +
String(random(40));
      // Send HTTP POST request
      int httpResponseCode = http.POST(httpRequestData);

      /*
      // If you need an HTTP request with a content type:
application/json, use the following:
      http.addHeader("Content-Type", "application/json");
      // JSON data to send with HTTP POST
      String httpRequestData = "{\"api_key\":\\"" + apiKey +
"\", \"field1\":\\"" + String(random(40)) + "\"}";
      // Send HTTP POST request
      int httpResponseCode = http.POST(httpRequestData);*/

      Serial.print("HTTP Response code: ");
      Serial.println(httpResponseCode);

      // Free resources
      http.end();
    }
    else {

```

```

    Serial.println("WiFi Disconnected");
  }
  lastTime = millis();
}
}
}

```

Ili bez biblioteke HTTPClient.h.

```
#include <WiFi.h>
```

```

String apiKey = "BE7U15RJYY57AV91";           // Enter your Write API key from ThingSpeak
const char *ssid = "Ashish";                  // replace with your wifi ssid and wpa2 key
const char *pass = "12345678";
const char* server = "api.thingspeak.com";
WiFiClient client;
void setup()
{
  Serial.begin(115200);
  delay(10);
  Serial.println("Connecting to ");
  Serial.println(ssid);
  WiFi.begin(ssid, pass);
  while (WiFi.status() != WL_CONNECTED)
  {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");
}
void loop()
{
  int h = 0;
  float t =0;

  h = hallRead(); //read value for h
  t = ((temperature_sens_read()-32)/1.8);      //changing temperature parameter to celsius
  if (client.connect(server,80))              // "184.106.153.149" or api.thingspeak.com
  {
    String postStr = apiKey;
    postStr += "&field1=";
    postStr += String(h);
    postStr += "&field2=";
    postStr += String(t);
    postStr += "\r\n\r\n";

    client.print("POST /update HTTP/1.1\n");
    client.print("Host: api.thingspeak.com\n");
    client.print("Connection: close\n");
    client.print("X-THINGSPEAKAPIKEY: "+apiKey+"\n");
    client.print("Content-Type: application/x-www-form-urlencoded\n");
    client.print("Content-Length: ");
    client.print(postStr.length());
  }
}

```

```

client.print("\n\n");
client.print(postStr);

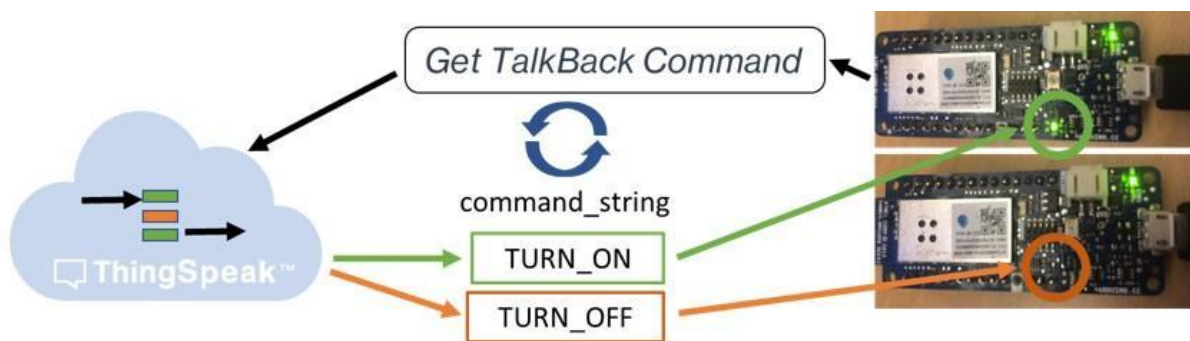
Serial.print("Hall: ");
Serial.println(h);
Serial.print("Temperature:");
Serial.print(t);
Serial.println(" C");

Serial.println("%. Send to Thingspeak.");
}
client.stop();
Serial.println("Waiting...");
delay(10000);
}

```

5.4.3 Dvosmjerno slanje podataka

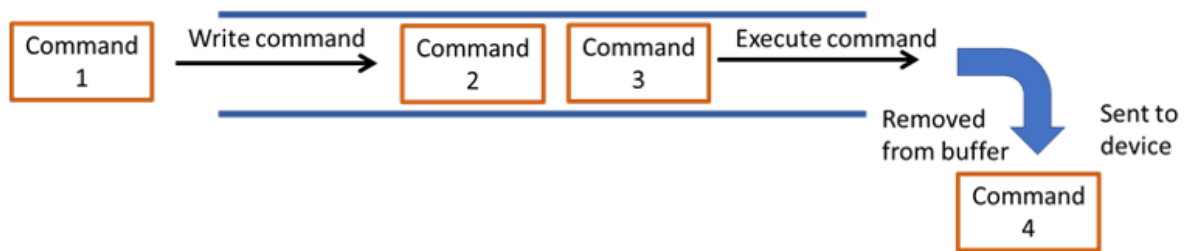
Gore navedeno slanje podataka na ThingSpeak server je jednosmjernog karaktera. Nekada je potrebno ostvariti povratnu spregu, tj. klijentu slati odgovarajuće naredbe, kao uključi rele, on-off itd... To je naročito zgodno kod kontrole alarmnih stanja. U tim slučajevima se upotrebljava TalkBack Api ThingSpeaka.



Slika 5.9 Dvosmjerno slanje podatka na Thing Speak

U ovom primjeru, zapisujete TURN_ON ili TURN_OFF u TalkBack red čekanja i uređaj menja stanje LED uređaja tako da se podudara sa komandom. Koristite aplikaciju ThingSpeak TalkBackApp da biste uskladištili komande za uređaj. Možete da koristite Web interfejs ili HTTP komande za upisivanje komandi uređaja na listu. Na listi možete da uskladištite do 8000 komandi. Svaki put kada uređaj pročita komandnu listu, on pročita jednu komandu i ukloni najnoviju komandu sa liste.

TalkBack Queue (FIFO Buffer)



Slika 5.10 TalkBack Queue. skladištenje komandi

5.4.3.1 TalkBack API commands

Add a TalkBack Command

```
POST https://api.thingspeak.com/talkbacks/40211/commands.json  
api_key=7Z2R7QTDSUM19TNL
```

Get a TalkBack Command

```
GET  
https://api.thingspeak.com/talkbacks/40211/commands/COMMAND\_ID.json?api\_key=7Z2R7QTDSUM19TNL
```

Update a TalkBack Command

```
PUT https://api.thingspeak.com/talkbacks/40222/commands/COMMAND\_ID.json  
api_key=7Z2R7QTDSUM19TNL
```

Execute the Next TalkBack Command

```
POST https://api.thingspeak.com/talkbacks/40222/commands/execute.json  
api_key=7Z2R7QTDSUM19TNL
```

Update a Channel and Execute the Next TalkBack Command

```
POST https://api.thingspeak.com/update.json  
field1=70  
api_key=  
talkback_key=7Z2R7QTDSUM19TNL
```

Get the Last Executed Command

GET

https://api.thingspeak.com/talkbacks/40222/commands/last.json?api_key=7Z2R7QTDSUM19TNL

Delete a TalkBack Command

DELETE

https://api.thingspeak.com/talkbacks/40222/commands/COMMAND_ID.json
api_key=7Z2R7QTDSUM19TNL

Delete All TalkBack Commands

DELETE <https://api.thingspeak.com/talkbacks/40222/commands.json>
api_key=7Z2R7QTDSUM19TNL

List All TalkBack Commands

GET

https://api.thingspeak.com/talkbacks/40222/commands.json?api_key=7Z2R7QTDSUM19TNL

TalkBack ID: 40222

api_key=7Z2R7QTDSUM19TNL

5.5 ESP32 programski interfejs za podršku ThingSpeak in TalkBack

Preduslov je da se mora formirati makar jedan kanal na ThingSpeaku na gore opisani način.

Kreirajte **Channel** kao što je prikazano u fascikli **Collect Data in New Channel** u novom kanalu i zapišite API ključ za pisanje. Takođe je potrebno da podesite **TalkBack**. Idite na **Apps > TalkBacks** i odaberite **New TalkBack**.

Add Commands to TalkBack Queue (**Dodaj komande u red za povratne razgovore**).

Komande možete dodati u Red za TalkBack na jedan od dva načina.

- * Koristite web interfejs ThingSpeak TalkBack da biste dodali komande u Red čekanja za TalkBack. TalkBack možete da konfigurirate tako da ima do 8000 komandi.

* Koristite ThingSpeak API. Možete da koristite HTTP POST zahtjev da biste dodali komandu u red. U sledećem postu zamijenite TALKBACK_ID, YOUR_TALKBACK_API_KEY, TALKBACK_COMMAND i POSITION_NUMBER odgovarajućim vrijednostima za kanal.

```
POST https://api.thingspeak.com/talkbacks/TALKBACK_ID/commands
api_key=YOUR_TALKBACK_API_KEY
command_string=TALKBACK_COMMAND
position=POSITION_NUMBER
```

5.5.1 Program ESP32

1)Preuzminje sa interneta najnovijeg Arduino® IDE.

2) Instaliranje ESP32 core.

Više informacija potražiti u članku https://github.com/espressif/arduino-esp32/blob/master/docs/arduino-ide/boards_manager.md.

3)U meniju Tools izabiranje odgovarajućeg porta i table u Arduino IDE- u. Ovaj primjer se testira pomoću opcije Sparkfun ESP32 Thing.

4) Nalijepiti šifru u Arduino IDE (Paste the code into the Arduino IDE). Dodati informacije o Wi-Fi mreži, TalkBack API ključ i broj TalkBack .

5) Programirati uređaj, a zatim gledati serijski monitor i LED da bi se posmatrale promjene kada se komande ispunjavaju. Svaka izvršena komanda biće uklonjena sa liste. Potrebno je da dodati još komandi na listu nakon što se one utroše.

5.5.2.Kod

1)Početi pisati kod tako što će se unijeti odgovarajuće biblioteke i definisati promenljive. Unijeti mrežni SSID i lozinku. Unijeti broj kanala i TalkBack parametre: myTalkBackID i myTalkBackKey.

```
/*
FetchCommandFromTalkBack
```

Description: Checks a TalkBack queue every 60 seconds and set the state of the build in LED according

to the latest command fetched. Turn the LED on and off by using the commands TURN_ON and TURN_OFF.

The TalkBack documentation can be found at <https://www.mathworks.com/help/thingspeak/talkback-app.html>.

Hardware: ESP32-based boards

Notes:

- Requires installation of EPS32 core. See https://github.com/espressif/arduino-esp32/blob/master/docs/arduino-ide/boards_manager.md for details.

- Select the target hardware from the Tools > Board menu

Copyright 2018, The MathWorks, Inc.

```
*/
```

```
#include <WiFi.h>
```

```
char ssid[] = <enter your SSID>; // Your network SSID (name)
```

```
char pass[] = <enter your password>; // Your network password
```

```
WiFiClient client;
```

```
unsigned long myTalkBackID = <enter your TalkBack ID>;
```

```
const char * myTalkBackKey = <enter your TalkBack API key>;
```

2)U funkciji Setup pokrenuti LED i pokrenuti serijski monitor.

```
void setup() {  
    pinMode(LED_BUILTIN, OUTPUT); // Set up LED  
    Serial.begin(115200); // Initialize serial  
    WiFi.mode(WIFI_STA);  
}
```

3)U glavnoj petlji započinje se uspostavljanjem veze sa lokalnom Wi-Fi mrežom. Kreiranjem POST poruke sa ispravnim parametrima. Poslati POST zahtev, provjeriti rezultat i provjeriti da li postoji komanda TalkBack. Sačekati 60 sekundi i ponovo provjeriti red.

```
void loop() {

    // Connect or reconnect to Wi-Fi
    if(WiFi.status() != WL_CONNECTED){
        Serial.print("Attempting to connect to SSID: ");
        Serial.println(String(ssid));
        while(WiFi.status() != WL_CONNECTED){
            WiFi.begin(ssid, pass);
            Serial.print(".");
            delay(5000);
        }
        Serial.println("\nConnected.");
    }

    // Create the TalkBack URI
    String tbURI = String("/talkbacks/") + String(myTalkBackID) +
String("/commands/execute");

    // Create the message body for the POST out of the values
    String postMessage = String("api_key=") + String(myTalkBackKey);

    // Make a string for any commands that might be in the queue
    String newCommand = String();

    // Make the POST to ThingSpeak
    int x = httpPOST(tbURI, postMessage, newCommand);
    client.stop();

    // Check the result
    if(x == 200){
        Serial.println("checking queue...");
        // check for a command returned from TalkBack
```

```

if(newCommand.length() != 0){

    Serial.print(" Latest command from queue: ");
    Serial.println(newCommand);

    if(newCommand == "TURN_ON"){
        digitalWrite(LED_BUILTIN, HIGH);
    }

    if(newCommand == "TURN_OFF"){
        digitalWrite(LED_BUILTIN, LOW);
    }
}
else{
    Serial.println(" Nothing new.");
}

}
else{
    Serial.println("Problem checking queue. HTTP error code " +
String(x));
}

delay(60000); // Wait 60 seconds to check queue again
}

```

4)Koristiti funkciju httpPOST da bi se pročitala sledeća talkBack komanda.

```

// General function to POST to ThingSpeak
int httpPOST(String uri, String postMessage, String &response){

    bool connectSuccess = false;
    connectSuccess = client.connect("api.thingspeak.com",80);

```

```

if(!connectSuccess){
    return -301;
}

postMessage += "&headers=false";

String Headers = String("POST ") + uri + String(" HTTP/1.1\r\n") +
    String("Host: api.thingspeak.com\r\n") +
    String("Content-Type:application/x-www-form-
urlencoded\r\n") +
    String("Connection: close\r\n") +
    String("Content-Length: ") +
String(postMessage.length()) +
    String("\r\n\r\n");

client.print(Headers);
client.print(postMessage);

long startWaitForResponseAt = millis();
while(client.available() == 0 && millis() - startWaitForResponseAt
< 5000){
    delay(100);
}

if(client.available() == 0){
    return -304; // Didn't get server response in time
}

if(!client.find(const_cast<char *>("HTTP/1.1"))){
    return -303; // Couldn't parse response (didn't find HTTP/1.1)
}

int status = client.parseInt();
if(status != 200){
    return status;
}

```

```

}

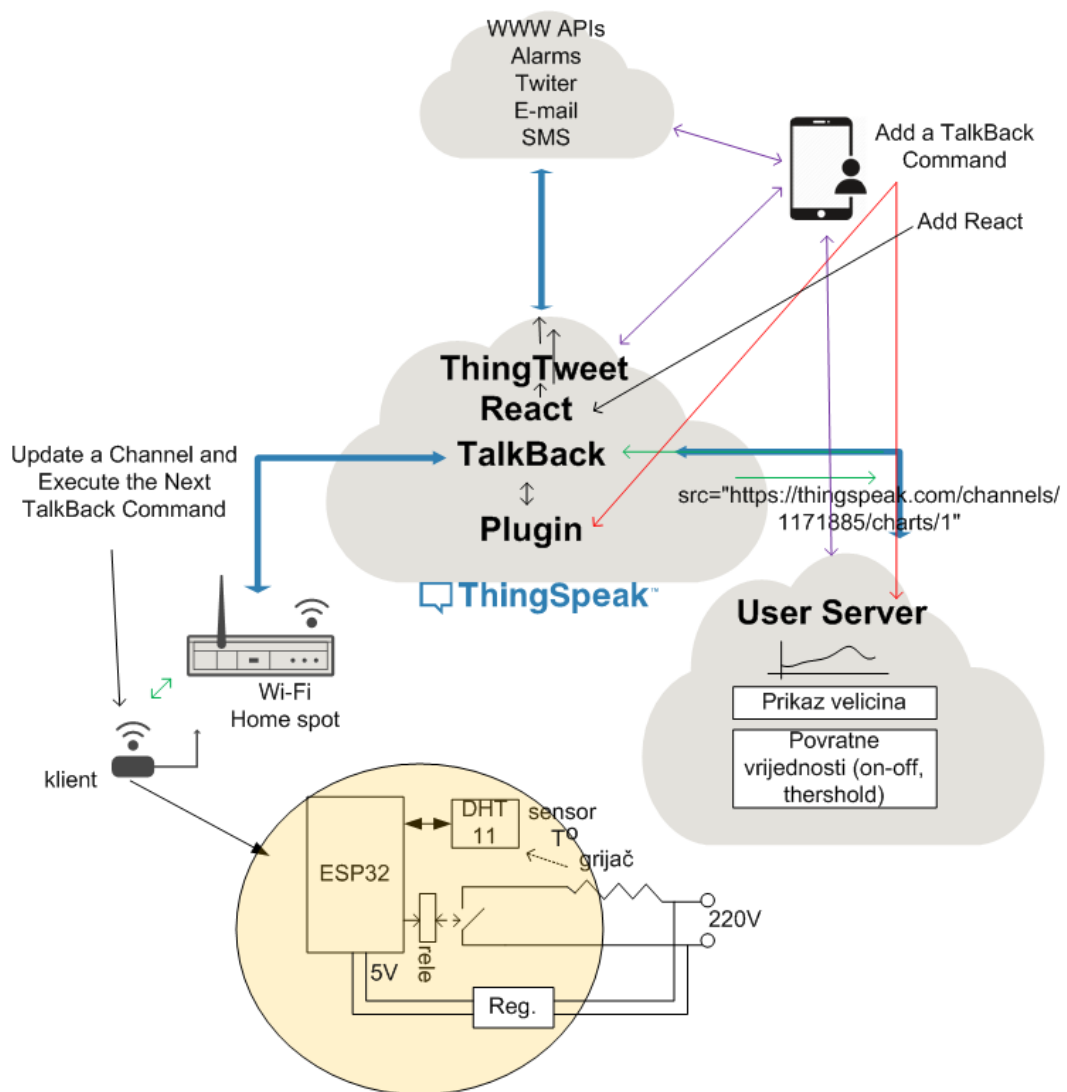
if(!client.find(const_cast<char *>("\n\r\n"))){
    return -303;
}

String tempString = String(client.readString());
response = tempString;

return status;
}

```

5.6 Predloženi scenario kućne automatizacije



Slika 5.11 Predloženi scenario kućne automatizacije

Predloženi scenario kućne automatizacije se sastoji od klienta baziranog na ESP32 koji preko Wi-Fi mreže komunicira sa kućnim Wi-Fi ruterom i preko njega sa WWW na kojem se nalazi ThingSpeak IoT Server ili slični IoT server. Prethodno su na ThingSpeaku napravljeni korisnički kanali koji imaju svoj API key i TalckBack, koji takodje ima svoj API key.

POST naredbom se šalju izmjerene veličine temperature i vlažnosti sa ECP32 na ThingSpeak Server gdje se prikazuju na otvorenom kanalu. Ujedno se sa datog kanala čita zadatka komanda (ako je ima) preko TalkBack APIa. Data komanda se zadaje ili preko Plugin APIa smještenog na ThingSpeak kanalu ili User servera koji šalje komandu ThingSpeak serveru i unutar njega TalkBack APIu. React API služi za obavještanje ili alarmiranje korisnika i može komunicirati sa drugim APIama koji su smešteni na ThingSpeak serveru ili na drugim WWW serverima, email, google, IFFFT itd.

Princip rada je sažet u sledećim koracima, gdje se daje i odgovarajući kod ili pseudo kod.

KORAK 1: Update a Channel and Execute the Next TalkBack Command

Ova operacija ima za cilj da pošalje pročitane vrijednosti analognih veličina, u našem slučaju temperature i vlažnost, pomoću ESP32 klienta, prema ThingSpeak serveru, kao i da očita određenu komandu koja je njemu namijenjena sa ThingSpeak servera. Izvršava se pomoću POST naredbe implementirane u kodu ESP32.

```
connectSuccess = client.connect("api.thingspeak.com",80);  
  
String Headers = String("POST /update HTTP/1.1\r\n") +  
String("Host: api.thingspeak.com\r\n") +  
String("Content-Type: application/x-www-form-urlencoded\r\n") +  
String("Connection: close\r\n") +  
String("Content-Length: ") + String(postMessage.length()) +  
String("\r\n\r\n");  
  
// Update a Channel and Execute the Next TalkBack Command  
String postMessage = String("field1=") + String(number1) +  
String("&field2=") + String(number2) +
```



```
String("&field3=") + String(number3) +  
String("&field4=") + String(number4) +  
String("&api_key=") + String(myApiKey) +  
String("&talkback_key=") + String(myTalkBackKey);  
client.print(Headers);  
client.print(postMessage);
```

KORAK 2: Add a TalkBack command

Ova operacija ima za cilj da zada povratnu komandu prema ESP32 koja može biti reakcija na pročitane analogne veličine ili jednostavno on-off komanda povezana sa pragom odlučivanja datih veličina, u našem slučaju ako temperatura i vlažnost predju neku vrijednost. Dodavanje komande se vrši na ThingSpeak klijentu na dva načina:

- Preko Plugin interfejsa u samom ThingSpeaku ili preko
- Eksternog www servera

U oba slučaja se koristi HTTP POST u odgovarajućem HTML ili JavaScript kodu

Plugins

Name

Send alarm value

HTML

```
<html>
```

```
<head>
```

```
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
```

```
<script type="text/javascript"  
src="//ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js"></script>
```

```
<style type="text/css">
```

```

body { background-color: white; }

</style>

<script type="text/javascript">

function lightOn(){

var x = document.getElementById("value").value;

var y= "MECOIoTALCH1=";

var e="END";

var z=y+x.toString()+e;

var
t='https://api.thingspeak.com/talkbacks/40164/commands.json?api_key=J626Z5V22
08BF9LV&command_string=';

var com=t+z+ '&position=1'

alert(com);

$.post(com);

}

</script>

<title>send_value</title>

</head>

<body>

<h3> MECOnet IoT demonstration</h3>

<p> -200=off, 200=on, other values threshold value</p>

<label for="value">Send value </label> <input id="value"
name="fname" type="text"><br>

<button onclick="lightOn()">Send</button>

</body>

</html>

```

CSS

```
<style type="text/css">
  body { background-color: #ddd; }
</style>
```

JavaScript

```
<script type="text/javascript">
  document.write('<div>javascript active</div>');
</script>
```

Eksterni web server, u našem slučaju <http://meconet.me>

<http://www.meconet.me/IoT-ThingSpeak>

5.6.1 MECOnet IoT demonstration

-200=off, 200=on, other values threshold value

Send value
Send

```
<meta http-equiv="content-type" content="text/html; charset=UTF-8"><script
type="text/javascript"
src="//ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js"></script>

<style type="text/css">body { background-color: white; }

</style>

<script type="text/javascript">

function lightOn(){

var x = document.getElementById("value").value;

var y= "MECOIoTALCH1=";

var e="END";

var z=y+x.toString()+e;
```

```

var
t='https://api.thingspeak.com/talkbacks/40164/commands.json?api_key=J626Z5V22
08BF9LV&command_string=';

var com=t+z+ '&position=1'

alert(com);

$.post(com);

}

</script>

<title></title>

<h3>MECOnet IoT demonstration</h3>

<p>-200=off, 200=on, other values threshold value</p>

<label for="value">Send value </label> <input id="value" name="fname"
type="text" /><br />

<button onclick="lightOn()">Send</button>

```

Prikaz dijagrama izmjerenih veličina i stanja kontrolnih veličina

Dijagramima se može pristupiti na 2 načina

- **Preko ThingSpeak URL, vezanog za korisnikov kanal, ili**
- **Preko Eksternog URL-a koju prenosi diagram sa ThingSpeak URLa**

Pristupanje vizualiaciji preko ThingSpeak URLa. Public view kanala je jednostavno dostupan, npr:

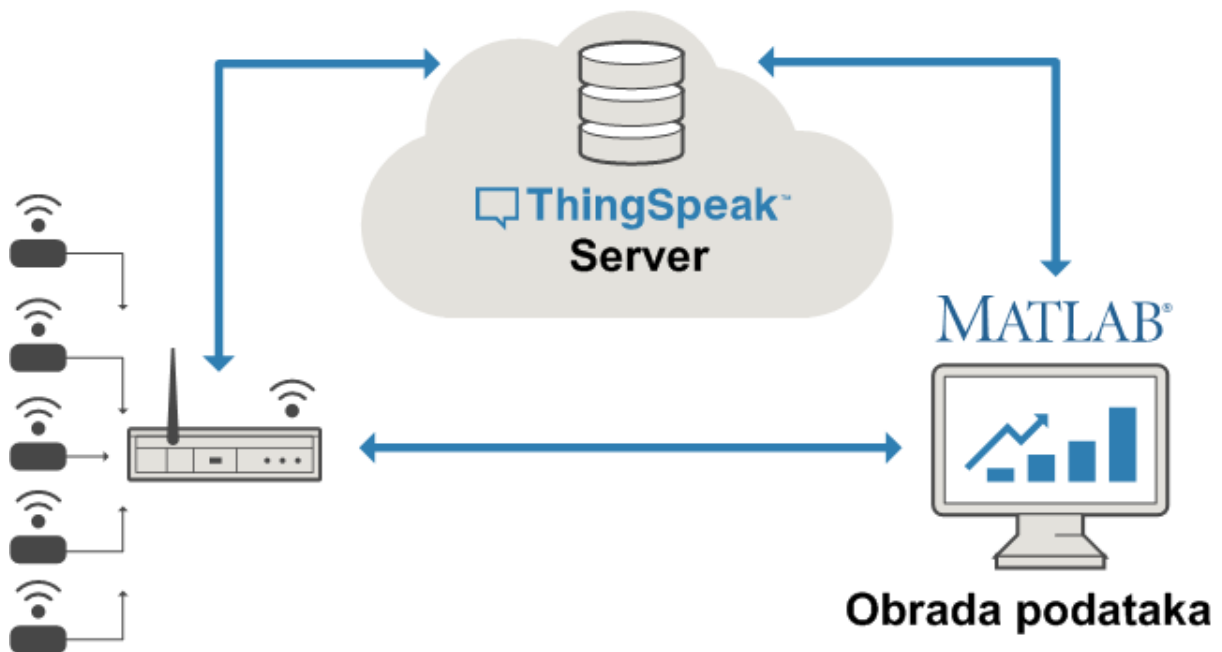
<https://thingspeak.com/channels/1171885>

Pristupanje vizualiaciji preko eksternog URLa. Public view kanala je dostupan preko pozivanja u html kodu eksternog sajta. Dijagrami npr. mogu biti smješteni u HTML obejka “iframe”, npr.

<http://www.meconet.me/IoT-ThingSpeak>

```
<p><iframe height="260"
src="https://thingspeak.com/channels/1171885/charts/1?bgcolor=%23ffffff&color=%23d62020&dynamic=true&results=60&type=line&update=15"
style="border: 1px solid #cccccc;" width="300"></iframe>
```

5.6.2 Apps MATLAB Visualizations (vizuelizacija MATLAB aplikacije)



Name

Slika 5.12 Vizualizacija ThingSpeak preko MATLAB aplikacije

Koristiti histogram da bi se razumjela varijacija(promjena) podataka.

5.6.3 MATLAB Code

```
% Read temperature for the last 10 hours from a ThingSpeak channel and  
% visualize temperature variations using the MATLAB HISTOGRAM function.
```

```
readChannelID = 1171885;
```

```
% Temperature Field ID  
TemperatureFieldID = 2;
```

```
% Channel Read API Key  
% If your channel is private, then enter the read API  
% Key between the '' below:
```

```
readAPIKey = '1IN00E6WEC5LVZG';
```

```
% Find mean value of temperature every min
```

```
tempF = thingSpeakRead(readChannelID,'Fields',TemperatureFieldID,...  
'NumMinutes',1*1, 'ReadKey',readAPIKey);  
bar(mean(tempF));  
%datetick('x','MM');  
ylabel('Mean value of each minute');
```

Save and Run

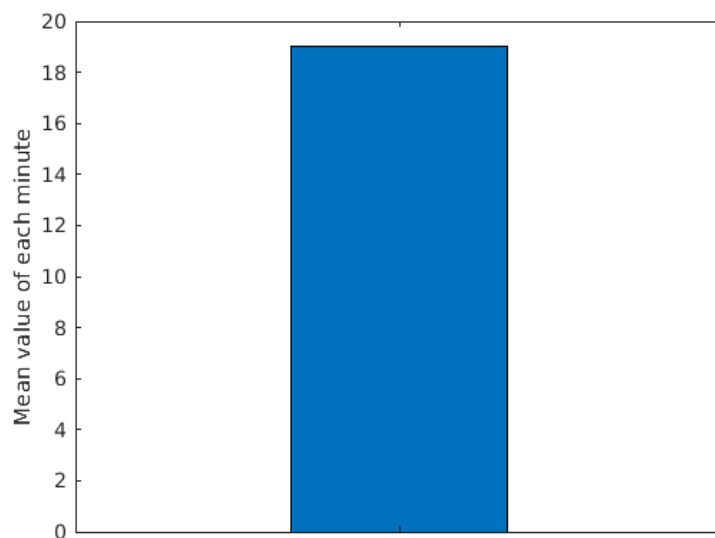
Save

Create a public URL:

https://thingspeak.com/apps/matlab_visualizations/370887



MATLAB Plot Output



Slika 5.13 Dijagram ThingSpeak preko MATLAB aplikacije

Glava VI

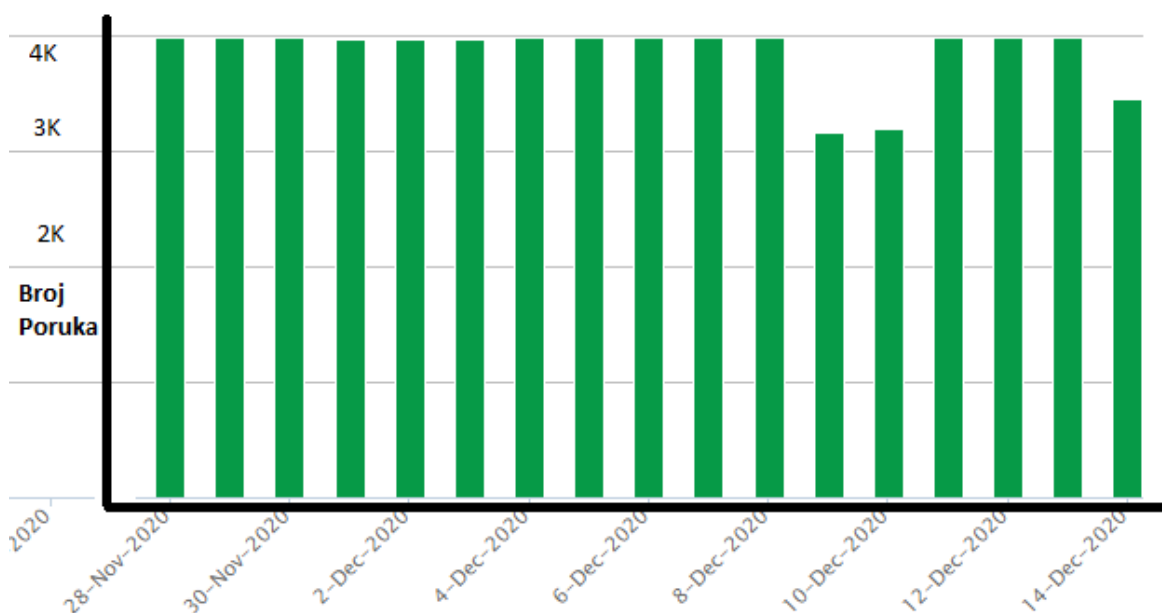
6.Rezultati testiranja

Testiranje predloženog sistema je izvršeno u periodu od 10/03/2020 do 15/12/2020. Ukupan broj kanala poslatih i preostalih poruka prema ThingSpeak kanalu “IoT MECOnet Experiments”, <https://thingspeak.com/channels/1171885>, je je dat u Tabeli 6.1T.

Tabela 6.1T: Broj poslatih poruka

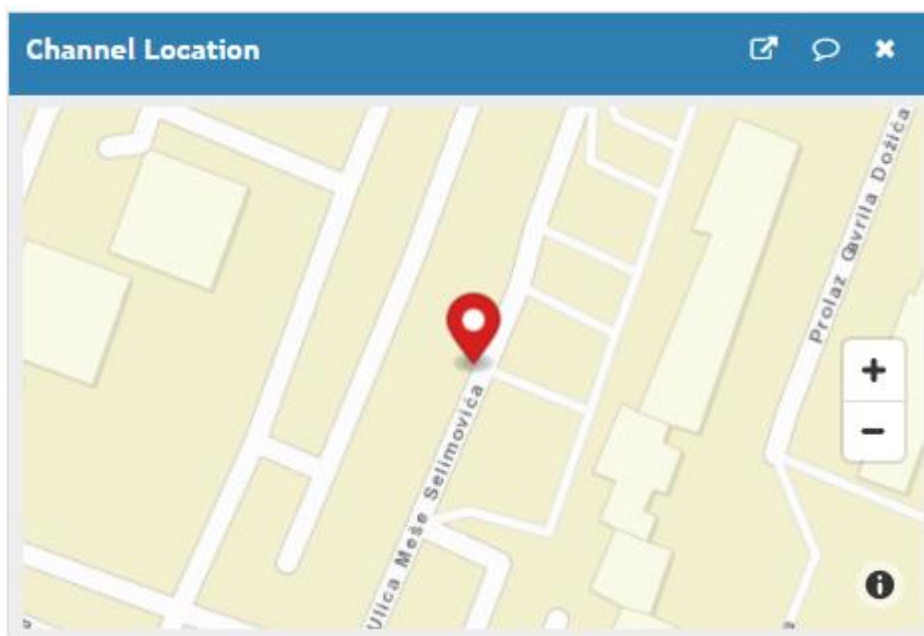
Poruke/kanali	Broj poslatih poruka	Frekvencija slanja
Broj poruka	163103	20sec
Broj kanala	4	20sec

Dnevno je slato oko 4000 poruka što se vidi na donjoj slici.



Slika 6.1 Dnevni protok poruka od uređaja prema ThingSpeak kanalu i www.meconet.me serveru.

ThingSpeak web server je obradio sve pristigle poruke i kontrolisao grejno tijelo snage 2KW koje se nalazilo na lokaciji u Podgorici, ul Meše Selimovića 12, čija je mapa prikazana na sledećoj slici 6.2.



Slika 6.2 Lokacija ThingSpeak web servera

Server koji je preuzimao informacije sa ThingSpeak-a je bio <http://meconet.me> koji posjeduje API prema server ThingSpeak.

Parametri koje je prihvatao i obradivao testni kanal su dati u Tabeli 6.2T

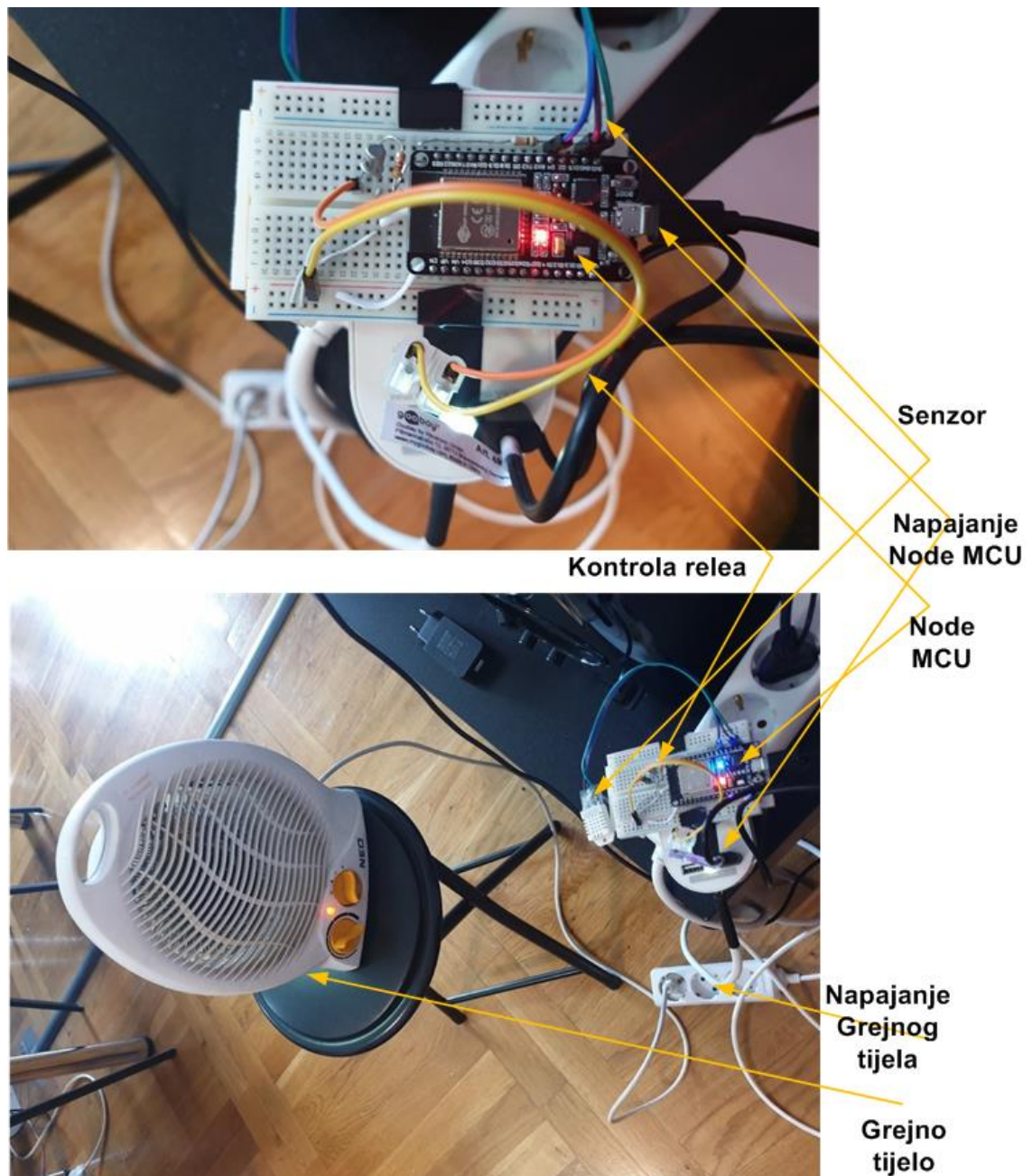
Tabela 6.2T Parametric testnog kanala

Ime kanala	IoT MECOnet Experiments
Opis kanala	Humadity and Temperature at MECOnet Location measured by ESP32 and sent each 20 seconds to Thing Speak. The feedback, control signals are sent from ThingSpeak to MECOnet location
Polje 1 (Field 1)	Humidity in %
Polje 2 (Field 2)	Temperature in Cel
Polje 3 (Field 3)	Control
Polje 4 (Field 4)	Temp Threshold
Polje 5	Send alarm (threshold) value
Polje 6	Use histogram to understand variations
Polje 7	Channel Location

Fotografija objekta regulacije na lokaciji je prikazana na donjoj slici . Prototip set se sastoji od:

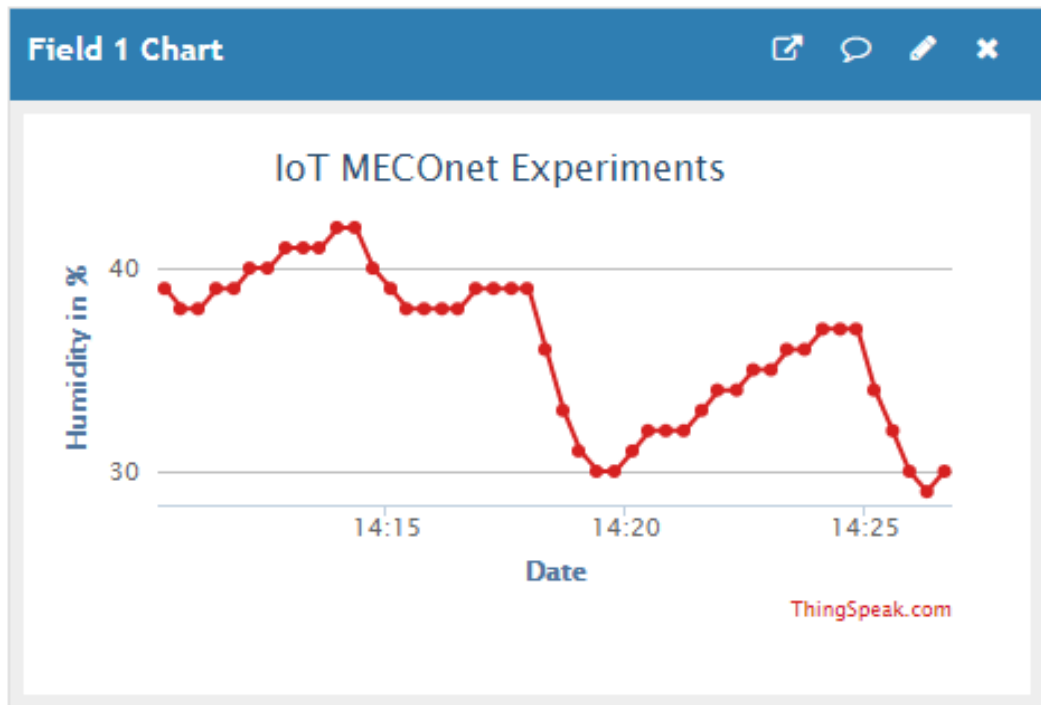
1. Node MCU

2. Senzora
3. Kola za napajanje Node MCU
4. Kola za kontrolu relea koji uključuje-isključuje grejno tijelo
5. Napajanje grejnog tijela
6. Grejnog tijela, 2KW grejača fena. Ova kombinacija je upotrijebljena da bismo mogli pratiti efekat zagrijavanja i ujedno sušenja vazduha, promjene vlažnosti.

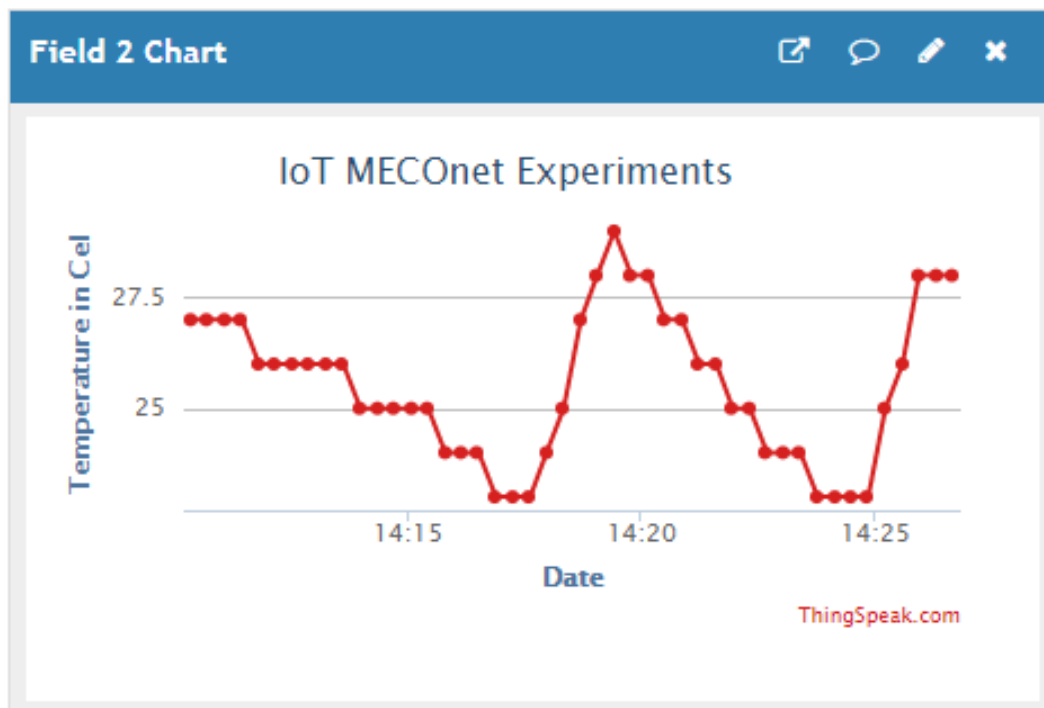


Slika 6.3: Detalji eksperimenta i prototip uređaja.

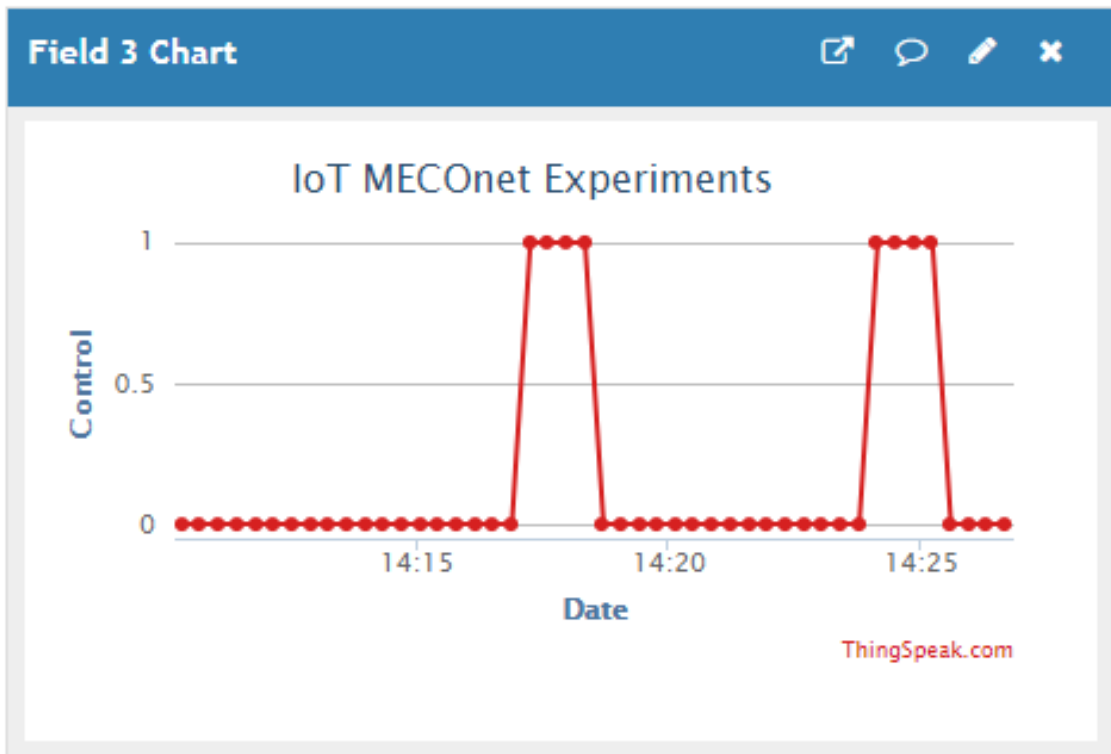
Donje slike prikazuju realno stanje na ThingSpeak kanalu u segment regulacije po gornjim poljima (Fields).



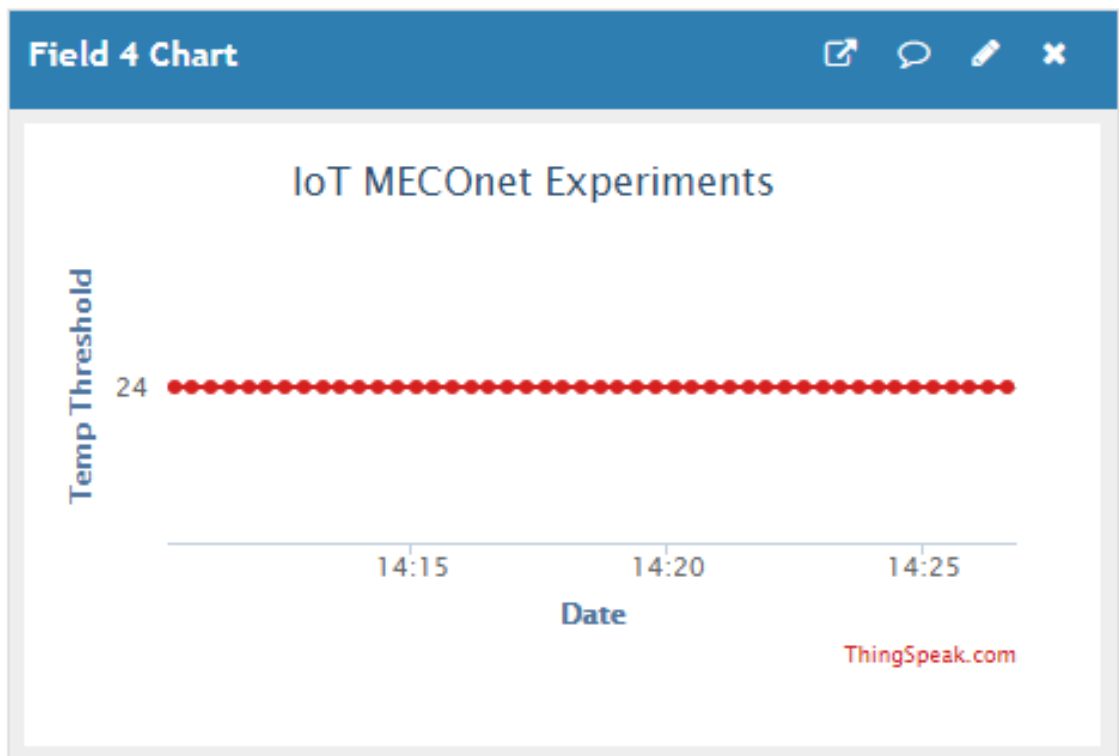
Slika 6.4: Dijagram promjene vlažnosti u određenom vremenskom intervalu.



Slika 6.5: Dijagram promjene temperature u određenom vremenskom intervalu



Slika 6.6: Kontrolni signal paljenja-gašenja grejnog tijela u određenom vremenskom interval



Slika 6.7: Zadana referentna vrijednost temperature

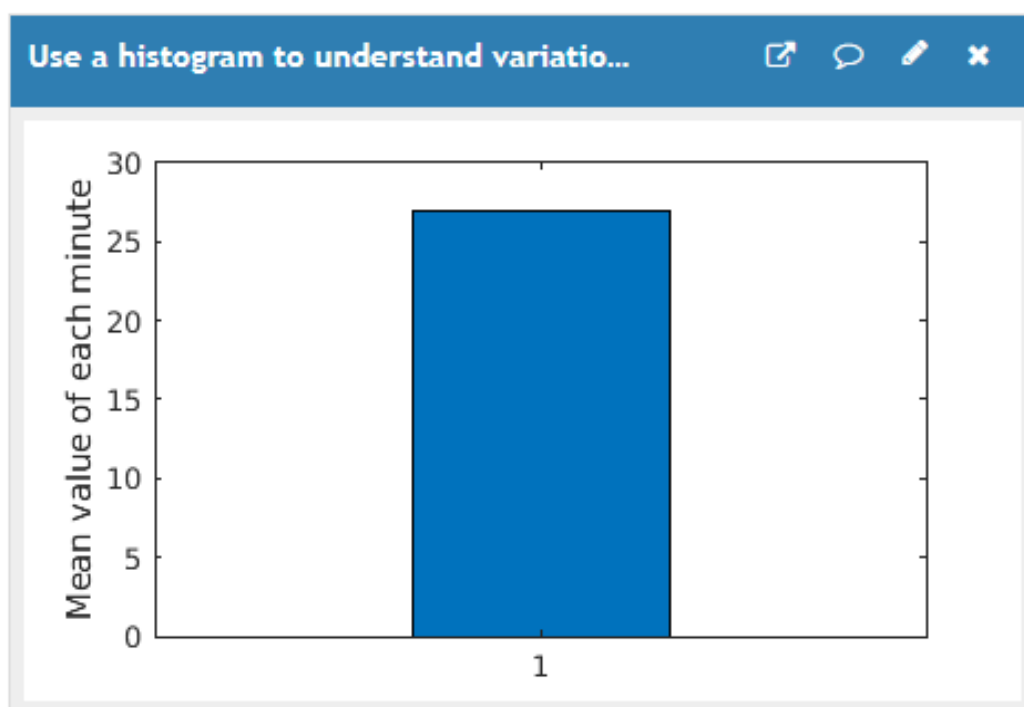
Send alarm value 🔗 💬 ✕

MECOnet IoT demonstration

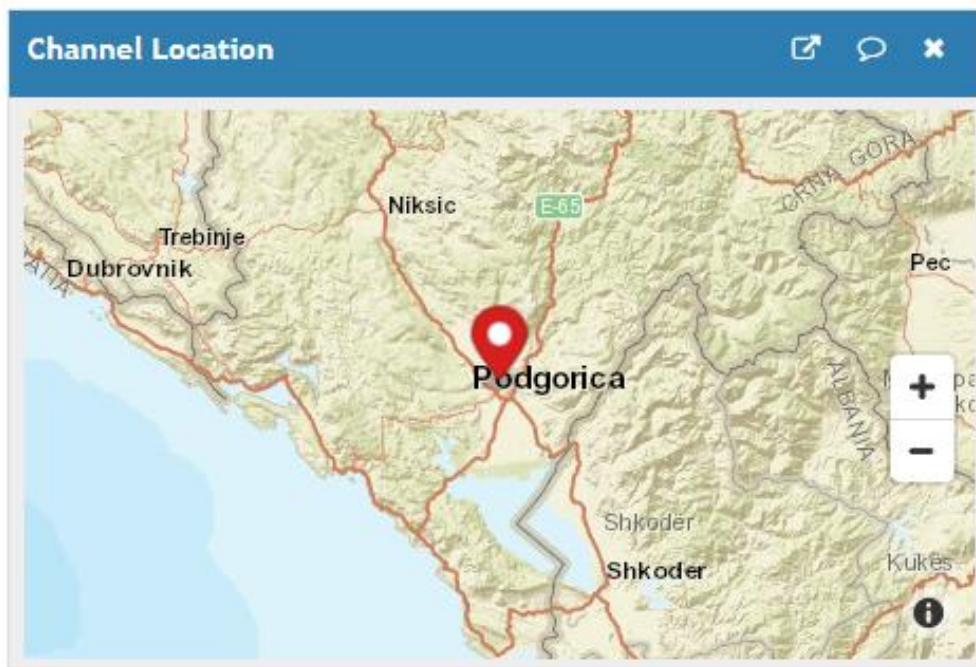
-200=off, 200=on, other values threshold value

Send value

Slika 6.8: Polje za definisanje referentne temperature

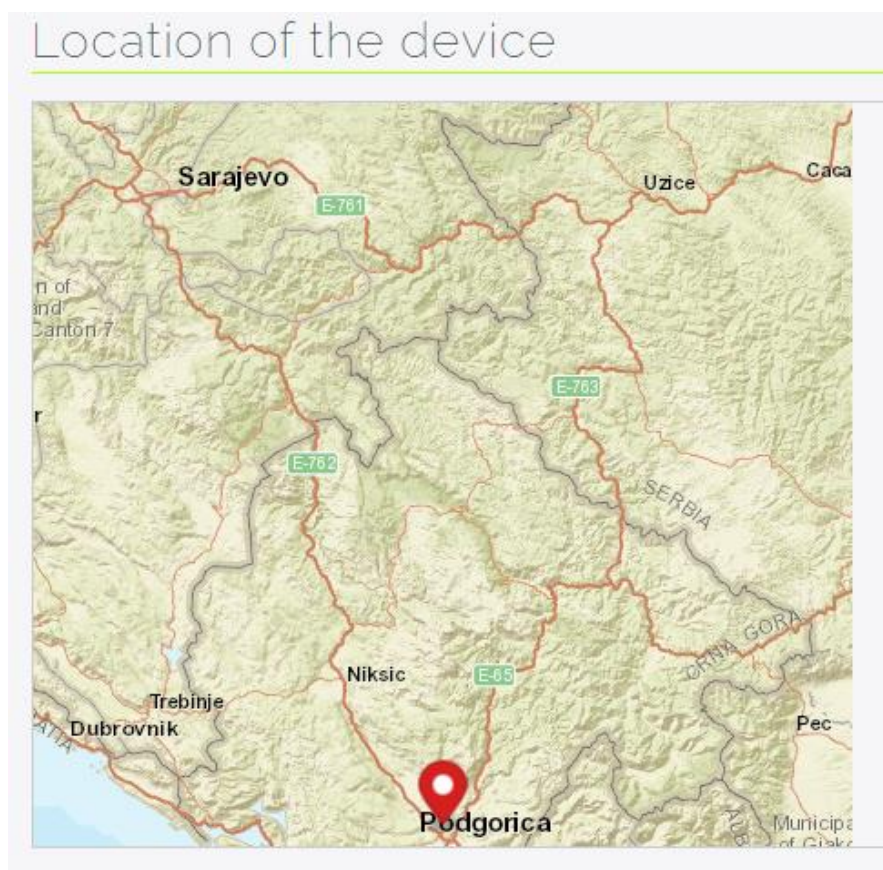


Slika 6.9: Srednja vrijednost temperature za vrijednost jednog minuta, nakon 3 odbirka poslatih prema ThingSpeak server, 3x 20sec-

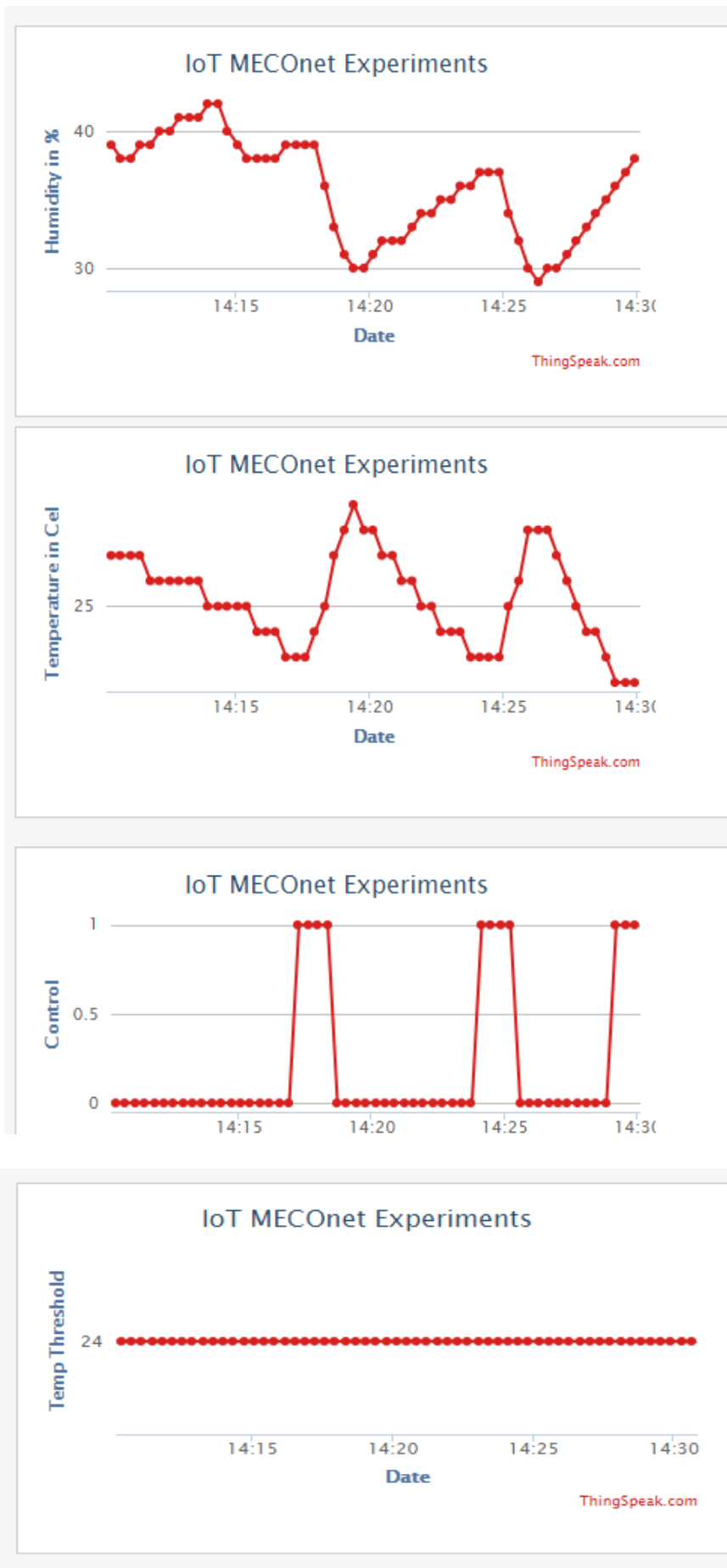


Slika 6.10: Lokacija senzora

Korisnički server koji vizualizuje podatke sa ThingSpeak IoT servera je <http://meconet.me> , preko korisničke stranice <http://www.meconet.me/IoT-ThingSpeak>



Slika 6.11: Lokacija prikazana na korisničkoj web stranici



Slika 6.12: Princip regulacije temperature na IoT objektu prikazan na korisničkoj web stranici.

IoT Fetch from ThingSpeak

The last values are shown from ThingSpeak Channel

Data fetched from [MECOnet ThingSpeak](#) every 10 seconds

H %: <input type="text" value="34"/>	T Celsius: <input type="text" value="23"/>
--------------------------------------	--

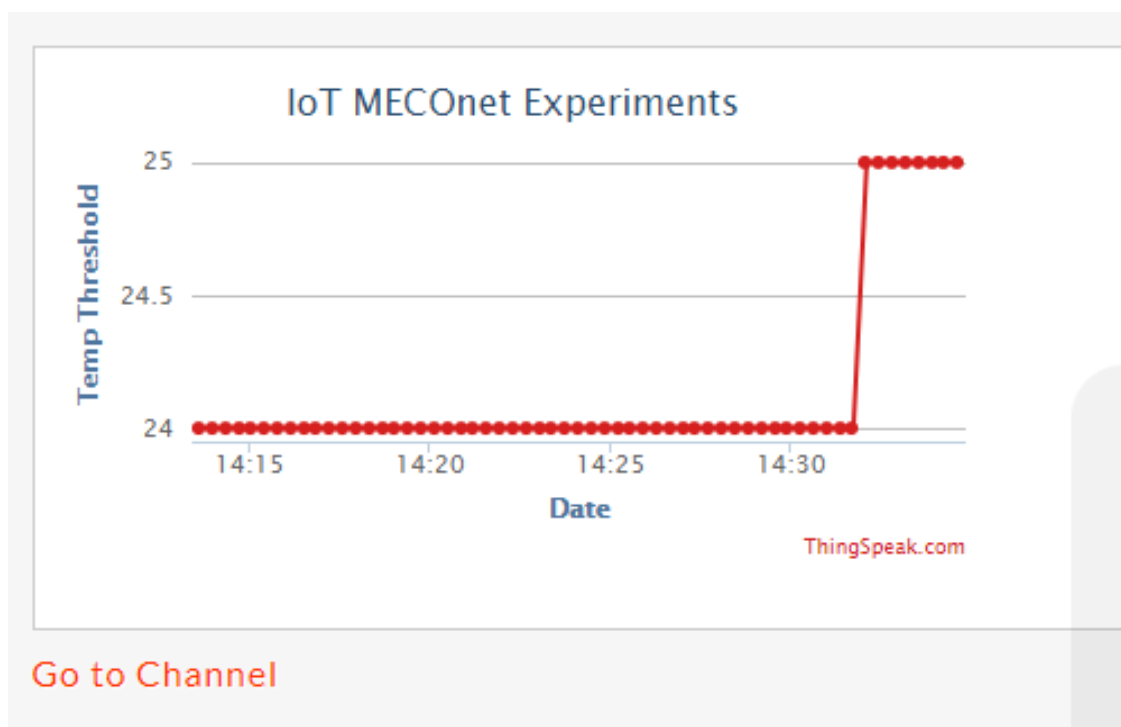
Thu Dec 10 2020 14:35:42 GMT+0100 (Central European Standard Time)

MECOnet IoT demonstration

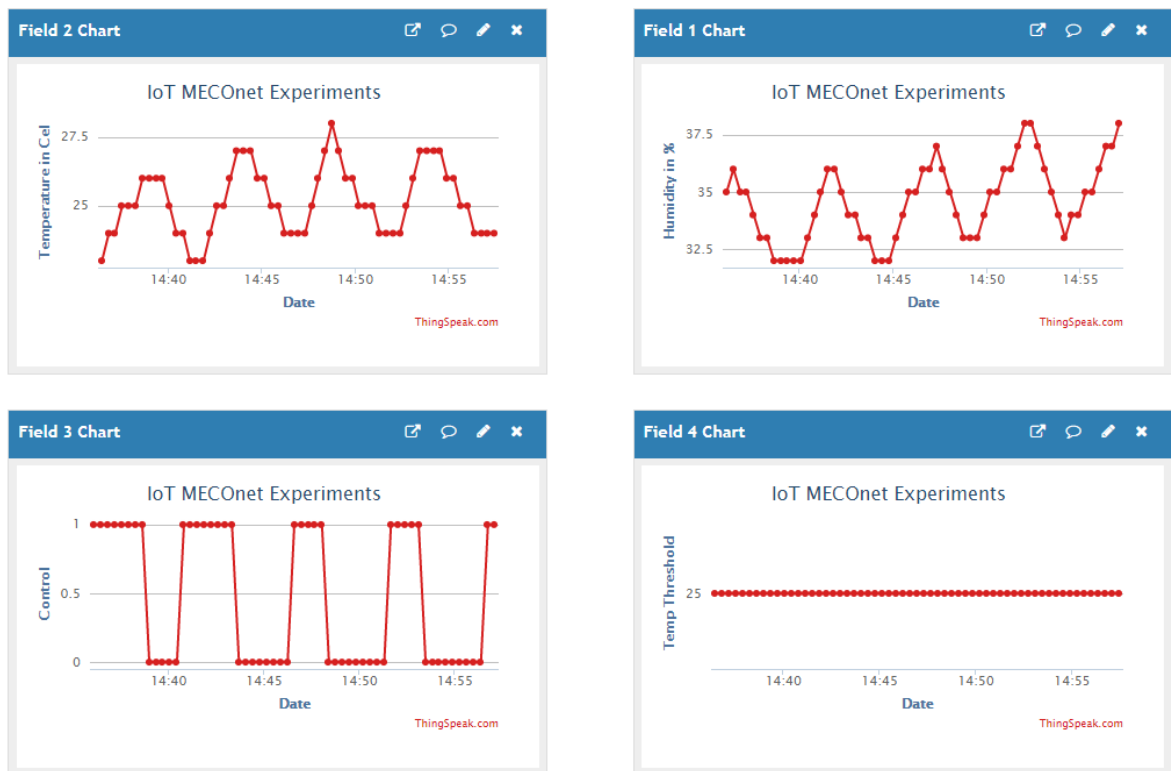
-200=off, 200=on, other values threshold value

Send value

Slika 6.13: Princip ekstrakcije trenutne vrijednosti mjerenih veličina i forma zadavanja referentne vrijednosti temperature.



Slika 6.14: Promjena referentne vrijednosti temperature



Slika 6.15: Efekti promjene referentne temperature na ostale dijagrame.

U slučaju da sistem ne radi, tj. poruke od uređaja (klijenta se ne šalju prema) ThingSpeak serveru, aktivira se alarmiranje preko React, ThingTwit metoda, što se vidi na donjoj slici, kada klijent nije slao podatke više of 10min. Korisnik je primio twitter poruku.

The screenshot shows the 'React Settings' configuration page in the ThingSpeak web interface. The page has a blue header with the ThingSpeak logo and navigation links for 'Channels', 'Apps', and 'Support'. Below the header, there are tabs for 'Apps', 'React', and 'React 3 | Edit'. The main content area contains the following settings:

- React Name:** React 3
- Condition Type:** No Data Check
- Test Frequency:** Every 10 minutes
- Condition:** If channel IoT MECOnet Experiments (11T18RS) has not been updated for 10 minutes
- Action:** ThingTweet
- then tweet:** ThingSpeak ne prima podatke
- using Twitter account:** RadovanStojan13
- Options:** Run action only the first time the condition is met

On the right side, there is a 'Help' section titled 'React Settings' with a list of instructions:

- React Name:** Enter a unique name for your React.
- Condition Type:** Select a condition type corresponding with your data. A channel can hold numeric sensor data, text, strings, status updates, or geographic location information.
- Test Frequency:** Choose whether to test your condition every time data enters the channel or on a periodic basis.
- Condition:** Select a channel, a field and the condition for your React.
- Action:** Select ThingTweet, ThingHTTP, or MATLAB Analysis to run when the condition is met.
- Options:** Select when the React runs.

A 'Learn More' link is located below the help text.

Slika 6.16: Setovanje React-ThingTweet aplikacija



Slika 6.17: Primijeni Tweet u slučaju da ThigTweet ne prima podatke

Proces testiranja greške u komunikaciji analiziran je tokom testnog perioda. Od ukupno 163103 poruka uspješno je procesirano 163073 poruka.

Glava VII

7 Zaključak

Projektovanje sistema kućne automatizacije predstavlja izazovan problem sa stanovišta inženjerske prakse. Današnje pametne kuće trebaju da se baziraju na pametnim, sigurnim, tehnologijama i treba da budu održivi sistemi i posjeduju odgovarajući stepen neophodne zaštite. Trenutni trendovi u automatizaciji kuća zahtijevaju da oni budu internet (daljinski) bazirani i da su podržani mobilnim tehnologijama. Pri projektovanju sistema kućne automatizacije trebamo uzimati u obzir više aspekata: potrebu korisnika, cijenu koštanja, pouzdanosti rada i mogućnost jednostavne nadogradnje u budućnosti u smislu praćenja dostupnih tehnologija.

U ovom radu je predloženo optimalno rešenja sistema kućne automatizacije sa stanovišta senzoričke hardvera i softvera. Na osnovu raspoložive i najjeftinije tehnologije predlože se odgovarajući sistem kućne automatizacije. U tom smislu sumiraju odgovarajuća profesionalna znanja iz oblasti kućne automatizacije u pogledu topologija, tehnologija, povezivanja, oblasti primjena i protokola. Predlože se Integrirani sistem kućne automatizacije koji predstavlja skalabilnu platformu u pogledu upotrebe senzoričke, hardvera i softvera. Kao odgovor na moderna vremena predlaže se sistem kućne automatizacija baziran na IoT principu koji u kombinaciji sa klasičnim i integriranim sistemima može predstavljati optimalno rešenje u velikom broju slučajeva. Daju su odgovarajuća hardverska i softverska rešenja IoT sistema kućne automatizacije sa primjerima izbora hardvera, projektovanja softvera i njihove integracije, upotreba nodeMCU u kombinaciji sa postojećim internacionalnim serverima za IoT. Izvršeno je testiranje predloženog IoT sistema kućne automatizacije na konkretnom primjeri i preliminarni rezultati testiranja elaborirani i diskutovani.

Dodatno, veliki broj senzorskih, hardverskih i softverskih rešenja je demonstriran u svrhu korišćenja sa stanovišta projektanata sistema kućne automatizacije.

Ostala znanja vezana za sisteme kućne automatizacije su sistematizovana i mogu biti korištena u praksi.

Na kraju, cilj rada je bio da ponudi realan i koristan izvor informacija za upoznavanje sa sistemima kućne automatizacije, kao i da budućim projektantima olakša izbor optimalnog rešenja, dajući više primjera i rezultata njihovog testiranja.

Glava VIII

8 Literatura

- [1] K. Pohl und E. Sikora, „Overview of the Example Domain: Home Automation“, in Software Product Line Engineering, Berlin/Heidelberg: Springer-Verlag, S. 39-52.
- [2] G. Kaiser, „eCourse on Integrated Home Systems | Leonardo ENERGY“, 13-Mai-2008. [Online]. Available: <http://www.leonardo-energy.org/ecourse-integrated-home-systems>. [Accessed: 18-Okt-2011].
- [3] W. Köhler, „Simulation of a KNX network with EIBsec protocol extensions“, 2008. [Online]. Available: <https://www.auto.tuwien.ac.at/thesis/pdf/THESIS0002.pdf>. [Accessed: 06-Nov-2011].
- [4] A. Jan, „Designing RS-485 Circuits“, Juni-1999.[Online].Available: http://www.embeddedsys.com/subpages/resources/images/documents/microsys_art_RS485.pdf. [Accessed: 18-Nov-2011].
- [5] Dallas Semiconductor, „1-Wire Standard“. [Online]. Available: <http://www.maximic.com/products/ibutton/ibuttons/standard.pdf>. [Accessed: 18-Nov-2011].
- [6] H. Köhler, „Verwendung eines 8-Bit Microcontroller zur Ethernet Vernetzung in der Hausautomation“, 2003. [Online]. Available: <http://www.ifas.htwkleipzig.de/easytoweb/download/Verwendung%20eines%208bit%20Microcontrollers%20zur%20Ethernet%20Vernetzung%20in%20der%20Hausautomation.pdf>. [Accessed: 18-Nov-2011].
- [7] K. Dostert, „Powerline-Kommunikation“. [Online]. Available: <http://www.net-imweb.de/pdf/Dostert.pdf>. [Accessed: 18-Nov-2011].
- [8] ZigBee Alliance, „ZigBee Technology“. [Online]. Available: <http://www.zigbee.org>. [Accessed: 18-Nov-2011].
- [9] EnOcean Alliance, „EnOcean Alliance“. [Online]. Available: <http://www.enoceanalliance.org/>. [Accessed: 18-Nov-2011].
- [10] Sutherland, J, “As Edge speeds increase, wires become transmission lines”, EDN pp 75-93 Oct 14, 1999
- [11] Awtrey, Dan, 1-WIRE IN A TRANSMISSION LINE ENVIRONMENT 4/26/2004

- [12] Dan Awtrey, Staff Engineer, Automatic Information Group, Dallas Semiconductor
MicroLAN Design Guide
- [13] I. billions), "IoT: number of connected devices worldwide 2012-2025 | Statista", Statista, 2018. [Online]. Available: <https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/>. [Accessed: 01- Aug- 2018].
- [14] "Internet Subscribers in Bangladesh January, 2018 | BTRC", Btrc.gov.bd, 2018. [Online]. Available: <http://www.btrc.gov.bd/content/internet-subscribers-bangladesh-january-2018>. [Accessed: 01- Aug- 2018].
- [15] "Domicile - An IoT Based Smart Home Automation" System International Conference on Robotics,Electrical and Signal Processing Techniques (ICREST) 2019
- [16] "Advanced IOT based combined remote health monitoring, home automation and alarm system - IEEE Conference Publication", Ieeexplore.ieee.org, 2018. [Online]. Available: <https://ieeexplore.ieee.org/document/8301659/>. [Accessed: 01- Aug- 2018].
- [17] "Design of a phoneme based voice controlled home automation system - IEEE Conference Publication", Ieeexplore.ieee.org, 2018. [Online]. Available: <https://ieeexplore.ieee.org/document/8307835/>. [Accessed: 01- Aug- 2018].
- [18] "Smart Home automation system using IR, bluetooth, GSM and android - IEEE Conference Publication", Ieeexplore.ieee.org, 2018. [Online]. Available: <https://ieeexplore.ieee.org/document/8313770/>. [Accessed: 01- Aug- 2018].
- [19] "Home automation using edge computing and Internet of Things - IEEE Conference Publication", Ieeexplore.ieee.org, 2018. [Online]. Available: <https://ieeexplore.ieee.org/document/8355544/>. [Accessed: 01- Aug- 2018].
- [20] "IOT based home automation by using personal assistant - IEEE Conference Publication", Ieeexplore.ieee.org, 2018. [Online]. Available: <https://ieeexplore.ieee.org/document/8358401/>. [Accessed: 01- Aug- 2018].
- [21] "Smart-home automation using IoT-based sensing and monitoring platform - IEEE Conference Publication", Ieeexplore.ieee.org, 2018. [Online]. Available: <https://ieeexplore.ieee.org/document/8372548/>. [Accessed: 01- Aug- 2018].
- [22] Ifttt. <https://ifttt.com/>, 2016. Last accessed: July 2016.
- [23] Zapier. <https://zapier.com/>, 2016. Last accessed: July 2016.
- [24] Internet Engineering Task Force. RFC6749 - The OAuth 2.0 Authorization Framework, 2012.

- [25] M. W. Newman, A. Elliott, and T. F. Smith. Providing an integrated user experience of networked media, devices, and services through enduser composition. In *International Conference on Pervasive Computing*, pages 213–227. Springer, 2008.
- [26] J. F. Pane, B. A. Myers, et al. Studying the language and structure in non-programmers' solutions to programming problems. *International Journal of Human-Computer Studies*, 54(2):237–264, 2001.
- [27] B. Ur, E. McManus, M. Pak Yong Ho, and M. L. Littman. Practical trigger-action programming in the smart home. In *SIGCHI Conference on Human Factors in Computing Systems, CHI'14*, 2014.
- [28] B. Ur, M. Pak Yong Ho, S. Brawner, J. Lee, S. Mennicken, N. Picard, D. Schulze, and M. L. Littman. Trigger-action programming in the wild: An analysis of 200,000 ifttt recipes. In *SIGCHI Conference on Human Factors in Computing Systems, CHI'16*, 2016.
- [29] M. Walch, M. Rietzler, J. Greim, F. Schaub, B. Wiedersheim, and M. Weber. homeblox: making home automation usable. In *Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication*, pages 295–298. ACM, 2013.
- [30] Somayya M., et al. 2015. Internet of Things (IoT): A Literature Review. *Journal of Computer and Communications*, 3,164-173
- [31] Tutorialspoint. 2016. Internet of Things, Simple Easy Learning. Tutorial Point (I) Pvt.Ltd.
- [32] Huansheng Ning. 2013. *Unit and Ubiquitous Internet of Things*. CRC Press Taylor & Francis Group
- [34] Zeinab Kamal, et.al. 2017. *Internet of Things Applications, Challenges and Related Future Technologies*. World Scientific News.
- [33] Shivangi Vashi et.al.2017. Internet of Things: A Vision, Architectural Elements and Security Issues. *International Conference on I-SMAC*, 2017.
- [34] M. Athar & R. Asnawi. 2017. Monitoring Kecepatan dan Arah Angin Pada Pembangkit Listrik Tenaga Bayu dengan Konsep Internet of Things. *Seminar Nasional Pendidikan Teknik Elektro X*.
- [35] M. Athar & R. Asnawi. 2017. Monitoring Kecepatan dan Arah Angin Pada Pembangkit Listrik Tenaga Bayu dengan Konsep Internet of Things. *Seminar Nasional Pendidikan Teknik Elektro X*.
- [36] Salsabillan Ulfa Tian & R. Asnawi. 2017. Prototipe Sistem Monitoring Parameter Pembangkit Listrik Tenaga Surya Berbasis Internet of Things. *Proyek Akhir Program Studi Teknik Elektro, Jurusan Pendidikan Teknik Elektro, FT UNY*

- [37] Ardis Bany Sutrisno & R.Asnawi, 2017. Monitoring Parameter Pembangkit Listrik Tenaga Surya Dengan Pendekatan Konsep Internet Of Things. Proyek Akhir Prodi Teknik Elektro, JPTE FT UNY.
- [38] ----- . What is Arduino?. Diakses dari <https://store.arduino.cc/usa/arduino-uno-rev3> pada [15 Januari 2019](#)

Bibliografija:

- [1] Igor Dragaš, Nikola Mrkić i Radovan Stojanović,“Primjeri realizacije AM-R baziranih senzora za kućnu automatizaciju i telemetriju“ , IT Žabljak , Crna Gora, Februar 2006 god.
- [2] Radovan Stojanović, Nikola Mrkić, Tafa Žilbert, Rade Gardašević, Konstantinos Perakis, “PhysiLAB –fleksibilni sistem za praćenje fizioloških signala“ , ETRAN 2006, Beograd , Srbija, Jun 2006.god.
- [3] Radovan Stojanović, Nikola Mrkić, Tafa Žilbert, Svetlana Stojanović and Konstantinos Perakis,“PHYSILAB AN INEXPENSIVE WEARABLE SYSTEM FOR MONITORING PHYSIOLOGICAL SIGNALS“ International conference ESBME 2006, Patris, Greece,jul 2006.god.
- [4] Tafa Žilbert, Radovan Stojanović, Konstantinos Perakis i Nikola Mrkić, ,“ MONITORING OF PHYSIOLOGICAL SIGNALS USING BLUETOOTH“ International conference ESBME 2006, Patris, Greece,jul 2006.god.