



Univerza v Mariboru

Fakulteta za energetiko

NAPRAVA ZA PRESTREZANJE, DEKODIRANJE IN
POSREDOVANJE SPOROČIL POZIVNIKOV ZA GSM
MOBILNO OMREŽJE

Diplomsko delo

Študent: Matic Krajšek
Študijski program: Visokošolski študijski program 1. stopnje Energetika
Mentor: izr. prof. dr. Peter Vrtič
Lektorica: Anica Krčmar, predmetna učiteljica slovenskega in srbohrvaškega jezika

Velenje, september 2019



Univerza v Mariboru

Fakulteta za energetiko

Hočevarjev trg 1
8270 Krško, Slovenija

Številka: 05-368-40/2019 JK
Kraj in datum: Krško, 22.08.2019

Na osnovi 330. člena Statuta Univerze v Mariboru (Uradni list RS, št. 29/2017–UPB12, 32/2019) izdajam:

SKLEP O ZAKLJUČNEM DELU

Maticu Krajšku, študentu visokošolskega strokovnega študijskega programa 1. stopnje **ENERGETIKA**, se dovoljuje izdelati zaključno delo.

Mentor: **izr. prof. dr. PETER VIRTič**

Naslov zaključnega dela:

NAPRAVA ZA PRESTREZANJE, DEKODIRANJE IN POSREDOVANJE SPOROČIL POZIVNIKOV ZA GSM MOBILNO OMREŽJE

Naslov zaključnega dela v angleškem jeziku:

A DEVICE FOR INTERCEPTION, DECODING AND FORWARDING OF PAGER MESSAGES FOR GSM MOBILE NETWORK

Rok za izdelavo in oddajo zaključnega dela je **30.09.2019**. Zaključno delo je potrebno izdelati skladno z »Navodili za izdelavo zaključnega dela na 1. in 2. stopnji študija UM FE« in ga v 3 spiralno vezanih izvodih oddati v Referat za študentske zadeve FE UM najkasneje 14 dni pred zagovorom. Hkrati se odda tudi izjava mentor-ja/-ice (in morebitnega somentor-ja/-ice) o ustreznosti zaključnega dela.

Pravni pouk:

Zoper ta sklep je možna pritožba na Senat Fakultete za energetiko v roku 10 delovnih dni od dneva prejema sklepa.



Dekan:
red. prof. dr. Bojan Štumberger

Obvestiti:
✓ - Matic Krajšek
- izr. prof. dr. PETER VIRTič
- arhiv

ZAHVALA

*Zahvaljujem se mentorju izr. prof. dr. Petru Vrtiču za
pomoč in usmerjanje pri izdelavi diplomskega dela.
Iskrena hvala moji mami za spodbudo in podporo med študijem.*

NAPRAVA ZA PRESTREZANJE, DEKODIRANJE IN POSREDOVANJE SPOROČIL POZIVNIKOV ZA GSM MOBILNO OMREŽJE

Ključne besede: pozivniki, posredovanje, GSM omrežje, POCSAG

UDK: 004.777:[621.395.721.5+621.396.7] (043.2)

Povzetek

V diplomskem delu smo izdelali napravo za posredovanje sporočil pozivnikov za GSM mobilno omrežje. Seznanili smo se z vso potrebno tehnologijo, razvijanjem Windows IoT aplikacij, izdelali program, razvili grafični vmesnik in po komponentah sestavili napravo. Uporabnik lahko sedaj sporočila, ki so bila poslana na pozivnik, prejema tudi na svojem mobilnem telefonu.

A DEVICE FOR INTERCEPTION, DECODING AND FORWARDING OF PAGER MESSAGES FOR GSM MOBILE NETWORK

Key Words: pagers, forwarding, GSM network, POCSAG

UDK: 004.777:[621.395.721.5+621.396.7] (043.2)

Abstract

In our thesis we designed a device for interception, decoding and forwarding of pager messages for GSM mobile network. We have acquainted ourselves with all the technology we needed, developing Windows IoT applications, building a program, developing a graphical user interface, and assembling the device by components. The user can now also receive pager messages on their mobile phone.

KAZALO VSEBINE

1 UVOD	1
2 SISTEM ZAŠČITE IN REŠEVANJA	2
3 DIGITALNA FREKVENČNA MODULACIJA	4
3.1 FSK MODULACIJA.....	4
3.2 FSK DEMODULACIJA.....	7
3.2.1 FM detektorski demodulator	7
3.2.2 Filtrirni demodulator.....	7
4 POCSAG PROTOKOL	9
4.1 CAP ALI RIC KODA.....	9
4.2 PREAMBULA.....	10
4.3 KODNA BESEDA	10
4.3.1 Struktura kodne besede naslova.....	10
4.3.2 Struktura kodne besede sporočila	11
4.3.3 Struktura kodne besede nedejavnosti	12
4.4 FSC KODA.....	12
4.5 ALFA NUMERIČNA SPOROČILA	12
4.6 NUMERIČNA SPOROČILA.....	13
4.7 POCSAG PAKET.....	14
5 UART	15
5.1 ZAČETNI BIT.....	16
5.2 PODATKOVNI OKVIR	17
5.3 PARNOSTNI BIT	17
5.4 KONČNI BITI.....	17
6 RADIJSKA POSTAJA ISKRA TR40M.....	18

6.1 OPIS PREDELAVE	19
7 FSK DEMODULATOR EXAR XR-2211A.....	21
7.1 DOLOČITEV FREKVENCE PIKE IN PRESLEDKA.....	23
7.2 PRERAČUN VREDNOSTI ZUNANJIH KOMPONENT	24
7.3 KONČNO VEZJE	26
8 GSM MODEM.....	27
8.1 KONČNO VEZJE	28
9 RASPBERRY PI.....	30
9.1 PROGRAMIRANJE.....	31
10 PRIMER UPORABE	37
10.1 IZPIS SPOROČILA NA LCD ZASLONU	37
10.2 IZPIS SPOROČILA NA TESTNI TELEFONSKI ŠTEVILKI.....	37
11 KONČNI IZDELEK.....	39
12 SKLEP	42
VIRI IN LITERATURA	43
PRILOGE.....	45

KAZALO SLIK

Slika 3.1: Odnos med binarnim prenosom podatkov in oddajanim signalom [10]	4
Slika 3.2: Parametri za opis FSK signala [10]	6
Slika 3.3: Tvorjenje nekoherentnega FSK signala [9]	6
Slika 3.4: FM – detektorski demodulator, blokovna shema [10]	7
Slika 3.5: Filtrirni demodulator, blokovna shema [10]	8
Slika 4.1: Preambula v binarni obliki	10
Slika 4.2: Struktura kodne besede naslova	11
Slika 4.3: Struktura kodne besede sporočila	12
Slika 4.4: Struktura POCSAG sporočila	14
Slika 5.1: Serijski prenos podatkov [4]	15
Slika 5.2: Povezava dveh mikro-krmilnikov z uporabo UART	16
Slika 5.3: UART paket	16
Slika 6.1: Radijska postaja Iskra TR40M	19
Slika 6.2: Vnos podatkov v program za programiranje radijske postaje	20
Slika 7.1: Blokovna shema demodulatorja [7]	21
Slika 7.2: Konfiguracija kontaktov [7]	22
Slika 7.3: Frekvenčna analiza avdio signala	23
Slika 7.4: Shema vezja demodulatorja [7]	24
Slika 7.5: Končno vezje demodulatorja	26
Slika 8.1: Shema vezja stabilizatorja napetosti in priključitve modema	28
Slika 8.2: Končno vezje modema	29
Slika 9.1: Mikroračunalnik Raspberry Pi 2 model B [14]	30
Slika 9.2: Grafični vmesnik – prva stran »Terminal«	31
Slika 9.3: Grafični vmesnik – druga stran »Demodulacija«	32
Slika 9.4: Grafični vmesnik – tretja stran »Modem«	32
Slika 10.1: Izpis sporočila na LCD zaslonu	37
Slika 10.2: Izpis sporočila na testni telefonski številki	38
Slika 11.1: Čelna plošča – risba	39
Slika 11.2: Primer komadov iz umetne mase, natisnjenih s 3D tiskalnikom	40
Slika 11.3: Notranja razporeditev komponent naprave	41

Slika 11.4: Končni izdelek..... 41

KAZALO TABEL

Tabela 4.1: Primer uporabe funkcijskih bitov	11
Tabela 4.2: ASCII tabela	13
Tabela 4.3: BCD Tabela	13
Tabela 6.1: Iskra TR40M - tehnični podatki	18
Tabela 7.1: Opis kontaktov [7]	22
Tabela 8.1: SIM800L – tehnični podatki [19]	27
Tabela 9.1: Sporočilo POCSAG [11]	33
Tabela 9.2: Prikaz uporabljenih AT komand.....	36

UPORABLJENE KRATICE

ZaRe	–	Zaščita in Reševanje
FSK	–	Frequency – Shift Keying
POCSAG	–	Post Office Code Standardisation Advisory Group
UART	–	Universal Asynchronous Receiver – Transmitter
GSM	–	Global System for Mobile communications
LCD	–	Liquid – Crystal Display
VCO	–	Voltage Controlled Oscillator
SIM	–	Subscriber Identity Module
USB	–	Universal Serial Bus
GPRS	–	General Packet Radio Service
FSC	–	Frame Synchronization Code
BCD	–	Binary – Coded Decimal
FM	–	Frequency Modulation
PLL	–	Phase – Locked Loop
BCH	–	Bose - Chaudhuri - Hocquenghem Error Correcting Code
HDMI	–	High Definition Multimedia Interface

1 UVOD

V Sloveniji se v sistemu radijskih zvez zaščite in reševanja (ZaRe) uporablja sistem tihega alarmiranja, njegova primarna naloga je pošiljanje kratkih besedilnih sporočil. Ta so predvsem namenjena gasilcem in reševalcem (naslovníkom), ki jih prejmejo na pozivnik [16].

Ob prijavi izrednega dogodka na telefonsko številko 112 mora dispečer v regijskem centru za obveščanje obvestiti naslovníka na najbolj zanesljiv način, da se bo ta nanj lahko odzval. To stori preko analognega omrežja, ki ga sestavljata terminal za vnos sporočila in oddajnik, ki preko radijskih signalov pošlje sporočilo. Čeprav se nam ta tehnologija zdi zastarela, pa na trgu trenutno ni alternative, ki bi bila robustnejša in zanesljivejša. Slabost tega sistema je, da mora naslovník ves čas imeti pozivnik pri sebi.

Danes si življenje brez mobilnega telefona zelo težko predstavljamo. Telefoniranje ni več prvotni namen, saj se je tehnologija razvila do te mere, da ga lahko uporabljamo tudi kot osebni računalnik. Postal je obvezen predmet našega vsakdana.

Odločili smo se, da bomo poskusili odpraviti slabost pozivnika, tako da bomo sporočilo s pomočjo radijske postaje prestregli, ga s pomočjo strojne in programske opreme dekodirali in ga s pomočjo GSM (angl. Global System for Mobile communications) modema posredovali na telefonsko številko mobilnega telefona.

V prvem delu naloge predstavimo vse potrebne tehnologije in protokole, uporabljene pri izdelavi naprave za prestrezanje, dekodiranje in posredovanje sporočil pozivnikov za GSM mobilno omrežje. V drugem delu predstavimo praktični del naloge, ki je sestavljen iz štirih ključnih delov: prestrezanje signalov s pomočjo radijske postaje, demodulacija s pomočjo strojne opreme, programsko dekodiranje z uporabo mikroračunalnika in posredovanje dekodiranega sporočila na GSM omrežje s pomočjo GSM modema. V zadnjem delu naloge predstavimo primer uporabe v praksi, ugotovitve ter možne izboljšave.

2 SISTEM ZAŠČITE IN REŠEVANJA

V Sloveniji se na področju zaščite in reševanja uporablja sistem radijskih zvez ZaRe. Gre za največji profesionalni analogni sistem radijskih zvez v državi. Njegova izgradnja se je pričela leta 1994 in je v lasti Ministrstva za obrambo, Uprave Republike Slovenije za zaščito in reševanje. Deli se na dva podsistema: na sistem radijskih zvez in sistem osebne klica. Celotna mreža je sestavljena iz 133 repetitorskih postaj, ki omogočajo 95-odstotno pokritost z uporabo stacionarnih in 100-odstotno pokritost z uporabo mobilnih repetitorskih postaj. Sistem ZaRe je izgrajen za delo v izrednih okoliščinah, kar pomeni, da morebitni izpad posameznih elementov ne vpliva na funkcionalnost celotnega sistema. S pomočjo daljinsko vodenih repetitorjev in z možnostjo takojšnjega popravila oziroma zamenjave poškodovanih elementov upravljalca zagotavlja nemoteno delovanje celotne mreže [16].

Mrežo sestavlja 68 radijskih kanalov v frekvenčnem obsegu od 168 do 174 MHz. Vsak ima svojo dodeljeno vlogo in je sestavljen iz sprejemne frekvence (Rx), oddajne frekvence (Tx) in dveh subtonov za sprejem in oddajanje, ki služita kot osnovna zaščita pred zlorabo in za preprečevanje motenj. Radijsko zvezo lahko uporabnik vzpostavi z drugim uporabnikom neposredno ali posredno (repetitorsko). Kanali so razdeljeni v tri skupine. Prvo skupino sestavlja 32 kanalov, ki so označeni z zaporedno številko kanala in kraja regijskega centra (primer: 01 REG CE). Ti kanali so semiduplexni, kar pomeni, da delujejo preko repetitorja. Kanala 31 in 32 sta rezervirana za mobilni repetitor. Drugo skupino sestavlja 18 simpleksnih kanalov, to pomeni, da so namenjeni vzpostavitvi neposredne zveze. Označeni so z zaporedno številko kanala in z oznako ZARE SI (primer: 35 ZARE SI). Njihove frekvence so enake izhodnim frekvencam repetitorjev. V tretjo skupino spada 18 simpleksnih kanalov. Označeni so z zaporedno številko kanala in z oznako ZARE SI (primer: 56 ZARE SI). Njihove frekvence so enake sprejemnim frekvencam repetitorjev. Za organizacijo prometa in dodeljevanje prostih kanalov uporabnikom skrbi regijski center za obveščanje.

Sistem osebne klica deluje tako, da se eden od kanalov uporablja za pošiljanje kratkih sporočil. Ta so namenjena imetnikom sprejemnikov – pozivnikov. Sistem deluje v okviru

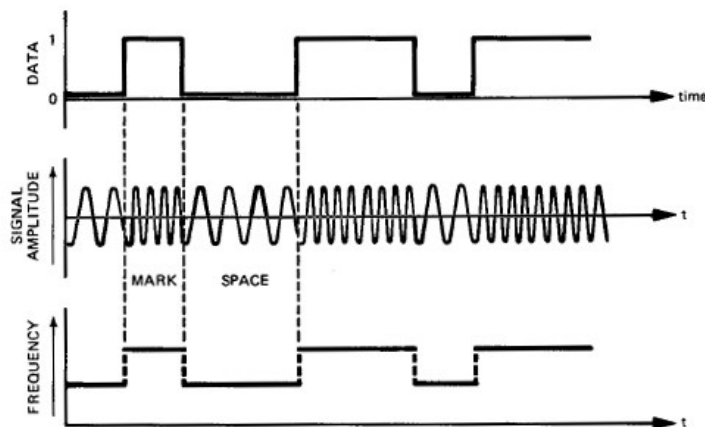
celotne države na enotni frekvenci, kar pomeni, da je možno poslati sporočilo na katerikoli pozivnik, če le poznamo njegov naslov. Sporočila na lokalni ravni pošilja regijski center za obveščanje, na državni pa Center za obveščanje Republike Slovenije. Možno je tudi pošiljanje sporočil s strani uporabnika. Za kodiranje sporočil se uporablja POCSAG protokol, ki omogoča pošiljanje tekstovnih sporočil do 250 znakov, tudi šumnikov in sličnikov. Podrobnejša struktura sporočila bo opisana v nadaljevanju [1].

3 DIGITALNA FREKVENČNA MODULACIJA

3.1 FSK MODULACIJA

Digitalna frekvenčna modulacija (angl. Frequency - Shift Keying) – FSK je vrsta frekvenčne modulacije, pri kateri se za prenos logičnih stanj uporabljata vsaj dve različni frekvenci. Zgodovinsko gledano bi lahko primerjali FSK modulacijo z Morsejevo abecedo, kjer se za prenos informacij uporabljata »pika« (angl. mark) in »presledek« (angl. space), kjer »pika« predstavlja logično stanje 1 in višjo frekvenco, »presledek« pa logično stanje 0 in nižjo frekvenco.

FSK modulacija je ena najpogostejših oblik digitalne modulacije, saj se uporablja skoraj povsod: za komunikacijo med računalniki, GSM mobilna omrežja, za daljinsko upravljanje (vrat, avtomobilov,...), za prepoznavanje telefonske številke klicatelja na stacionarnih telefonih, za telemetrijo,... Poznamo več vrst FSK modulacije, med katerimi je najosnovnejši primer binarne FSK modulacije. Slika 3.1 prikazuje odnos med binarnim prenosom podatkov in oddajanim signalom [8].



Slika 3.1: Odnos med binarnim prenosom podatkov in oddajanim signalom [10]

Minimalni čas logičnega stanja se imenuje »dolžina elementa«. Vrednosti znašajo med 5 in 22 milisekund, vendar se v praksi uporablja tudi čas, ki je manjši od 1 mikrosekunde oziroma večji od 1 sekunde. Dolžino elementa lahko opišemo tudi drugače. Uporabimo lahko izraz »hitrost prenosa podatkov« (angl. baud rate) [10].

$$\text{hitrost prenosa podatkov} = \frac{1}{\text{dolžina elementa [v sekundah]}} \quad (2.1)$$

FSK signal običajno opisujemo s frekvenco zamika (angl. Shift frequency) in s sredinsko frekvenco (angl. center frequency). Frekvenca zamika je razlika med frekvenco, ki predstavlja logično stanje 1 (f_1), in frekvenco, ki predstavlja logično stanje 0 (f_2). Običajno se vrednosti gibljejo med 50 in 1000 Hz [10].

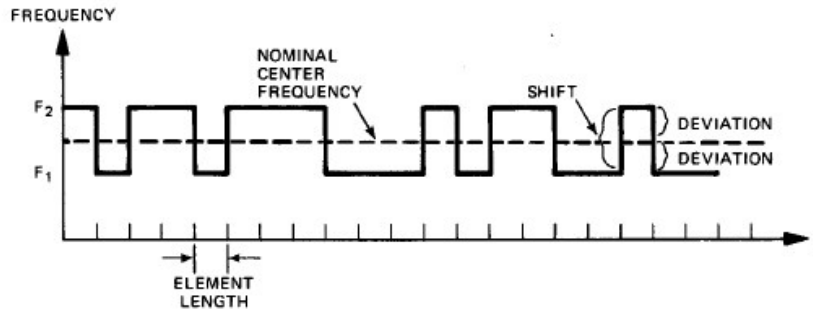
$$\text{Frekvenca zamika} = |f_2 - f_1| \quad (2.2)$$

Sredinska frekvenca se nahaja na sredini frekvence f_1 in f_2 .

$$\text{Sredinska frekvenca} = \frac{f_2 + f_1}{2} \quad (2.3)$$

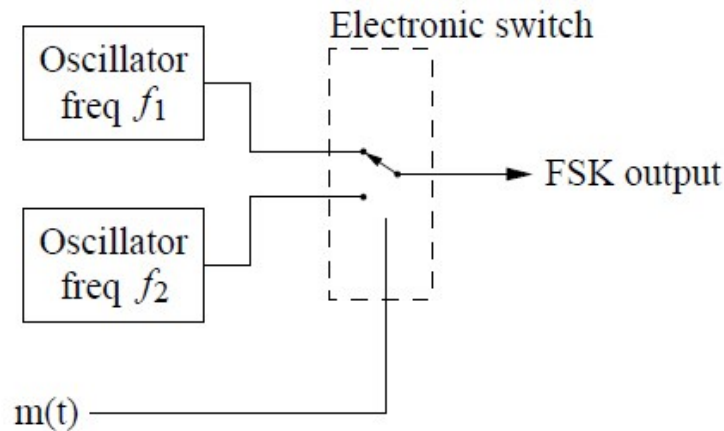
Občasno se uporablja tudi izraz »deviacija«, ki izhaja iz frekvenčne (angl. Frequency Modulated) – FM modulacije. Ta je enaka absolutni vrednosti razlike med sredinsko frekvenco in frekvenco f_1 ali f_2 . Numerično je enaka tudi polovici frekvence zamika. Slika 3.2 prikazuje najpogostejše parametre, s katerimi opisujemo FSK signal [10].

$$\text{Deviacija} = \frac{\text{frekvenca zamika}}{2} = \left| \frac{f_2 - f_1}{2} \right| \quad (2.4)$$



Slika 3.2: Parametri za opis FSK signala [10]

FSK signal se lahko tvori koherentno ali nekoherentno. Koherentno pomeni, da ima faza frekvence f_1 ali f_2 fiksno fazno razmerje proti referenci. Primer nekoherentnega tvorjenja FSK signala v praksi je prikazan na sliki 3.3 [10].



Slika 3.3: Tvorjenje nekoherentnega FSK signala [9]

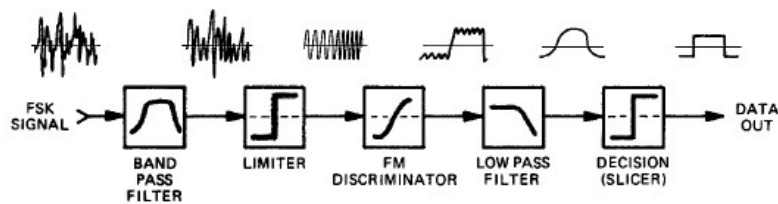
Tvorjenje signala se izvaja tako, da preklapljammo med dvema frekvenčnima oscilatorjema. Ker prehod med oscilatorjema ni v fazi, se pojavi motnja. Koherentni signal je veliko odpornejši na pojav napake, se pa v praksi kljub temu večkrat pojavlja nekoherentno tvorjenje FSK signala, saj je izdelava tega enostavnejša [10].

3.2 FSK DEMODULACIJA

Metode demodulacije lahko razvrstimo v dve kategoriji: FM detektorski demodulator in filtrirni demodulator.

3.2.1 FM detektorski demodulator

Ta vrsta demodulatorja tretira FSK signal kot preprost FM signal z binarno modulacijo. Funkcijska blokovna shema je prikazana na sliki 3.4. Demodulator najprej odstrani interference, ki so izven željenega frekvenčnega pasu. Omejeni signal na izhodu je nato zaznan s FM detektorjem, kar tvori pozitivni izhod za logično stanje 1 in negativni izhod za logično stanje 0. Zaznan signal potuje preko nizkoprepustnega filtra, kjer se odstranijo komponente šuma in neželene frekvence. FM detektorski demodulatorji so bili v preteklosti zelo popularni, saj je njihova izvedba dokaj enostavna, so pa zaradi širokega vhodnega frekvenčnega spektra nezanesljivi, saj je možnost pojava napake pri demodulaciji velika. V zadnjem času se srečujemo s Phase – Locked – Loop (PLL) demodulatorji. Ti imajo zelo podobno zmogljivost kot FM detektorski demodulatorji, vendar se v nekaterih okoliščinah, kadar se gradijo za točno določene signale, lahko izkažejo za boljše [10].

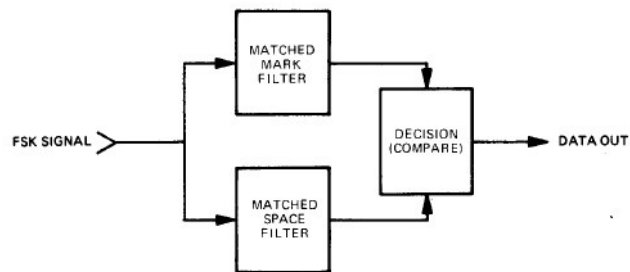


Slika 3.4: FM – detektorski demodulator, blokovna shema [10]

3.2.2 Filtrirni demodulator

Filtrirni demodulator je izgrajen tako, da se upoštevajo parametri vhodnega signala, na podlagi katerih se konfigurira demodulator, da ta deluje čim bolj optimalno. S tem se poskuša zmanjšati možnost pojava napake pri demodulaciji. Na pravilno izgradnjo filtra ne

vplivajo samo parametri signala, ampak tudi vrste motenj, ki nanj delujejo. Funkcijska blokovna shema je prikazana na sliki 3.5. V demodulatorju se izhoda filtra za logično stanje 1 in 0 primerjata in če je izhod filtra za logično stanje 1 večji od tistega za logično stanje 0, se v odločitvenem bloku odloči, da se na izhodu demodulatorja pojavi logično stanje 1. Takšna vrsta demodulatorja je najbolj zaželen, saj so filtri optimizirani na vhodni signal, kar pomeni, da se možnost pojava napake pri demodulaciji zmanjša [10].



Slika 3.5: Filtrirni demodulator, blokovna shema [10]

4 POCSAG PROTOKOL

Post Office Code Standardisation Advisory Group – POCSAG protokol se uporablja za prenos podatkov pri pozivnikih. Ime izvira iz Velike Britanije, kjer so se leta 1976 pričeli srečevati inženirji iz vsega sveta, da bi razvili nov protokol za omrežje pozivnikov. Takrat je bila uporaba le-teh zelo popularna, zato je posledično bila pokritost s signalom zagotovljena po celotnih državah. Protokol so inženirji razvili pod okriljem Britanske pošte (angl. British Post Office), ki je v tistem času upravljala z vso telekomunikacijsko infrastrukturo v državi. Uvedba protokola je bila velik korak naprej, saj je ta omogočal daljšo življenjsko dobo baterije pozivnika in povečal skupno število naročnikov v omrežju. Poznamo dve vrsti pozivnikov: numeričnega, ki prikazujejo samo številke, in alfa numeričnega, ki poleg številke prikazuje tudi besedilo.

Hitrosti prenosa podatkov so standardizirane: 512, 1200 ali 2400 bit/s. Najdaljšo komunikacijsko razdaljo omogoča hitrost 512 bit/s. Sporočila se pošiljajo strukturirano. Pošiljanje se prične z dolgo preambulo, kar omogoča pozivnikom, da periodično vključijo sprejemnik in pričnejo s poslušanjem, ko je preambula zaznana.

POCSAG sporočilo je sestavljeno iz 576 bitov dolge preambule in iz niza paketov. Ti so sestavljeni iz ene ali več kodnih besed. Vsak paket je sestavljen iz 32-bitne kode, ki se uporablja za identificiranje začetka paketa, imenovane Frame Synchronization Code – FSC, in osmih naslovnih okvirjev. Naslovni okvir je sestavljen iz dveh 32 bitov dolgih kodnih besed [11].

4.1 CAP ALI RIC KODA

CAP (angl. Channel Access Protocol) ali RIC (angl. Radio Identity Code) koda je unikatna koda, ki se dodeli posameznemu pozivniku. Če ta zazna svojo CAP kodo pri dekodiranju sporočila, uporabnika opozori s piskom, tonom ali vibracijo [11].

4.2 PREAMBULA

Je sestavljena iz 576 bitov logičnih enk in ničel, ki med sabo alternirajo. Slika 4.1 prikazuje delček preambule. Njen namen je omogočiti pozivniku dodatno varčevanje z energijo. To je v praksi rešeno tako, da se pozivnikov sprejemnik vključuje za samo delček sekunde in se v tem času preveri, če je mogoče zaznati preambulo. Če jo je mogoče zaznati, ostane sprejemnik vključen in se prične sinhronizirati z njeno hitrostjo prenosa podatkov. Vsi nadaljnji paketi imajo enako hitrost prenosa podatkov kot preambula [11].

Skupaj: 576 bitov

101010101010101010101010101010101010

Slika 4.1: Preambula v binarni obliki

4.3 KODNA BESEDA

Poznamo tri različne vrste kodnih besed: kodna beseda naslova (angl. address code word), kodna beseda sporočila (angl. message code word) in kodna beseda nedejavnosti (angl. idle code word). Standardizirana dolžina kodne besede je 32 bitov [11].

4.3.1 Struktura kodne besede naslova

Kodna beseda naslova, ki jo prikazuje slika 4.2, je sestavljena iz 32 bitov. Stanje prvega bita je logično 0, kar ga tudi razlikuje od kodne besede sporočila. Biti, ki se nahajajo na zaporednem mestu od 2 do 19, so naslovni biti. CAP koda je sestavljena iz teh 18 naslovnih bitov in treh bitov, ki ponazarjajo lokacijo okvirja znotraj paketa – skupaj 21 bitov, kar omogoča 2 097 152 unikatnih CAP kod.

Biti na zaporednem mestu 20 in 21 sta funkcijska bita (angl. source identifier bits). Ponazarjata nam vir sporočila. Primer uporabe prikazuje tabela 4.1.

Tabela 4.1: Primer uporabe funkcijskih bitov

Bit 20	Bit 21	Funkcija
0	0	Določi operater
0	1	
1	0	
1	1	

Biti na zaporednem mestu od 22 do 31 se imenujejo BCH (angl. Bose - Chaudhuri - Hocquenghem Error Correcting Code) biti. POCSAG uporablja BCH kodo za popravilo napake. Ta omogoča sprejemniku, da lahko popravi največ dva napačna bita. Bit na zaporednem mestu 32 je namenjen preverjanju parnosti [11].



Slika 4.2: Struktura kodne besede naslova

4.3.2 Struktura kodne besede sporočila

Sporočilo, ki je bilo poslano na pozivnik, je sestavljeno iz ene kodne besede naslova in med 0 in n kodnih besed sporočila. Sporočilo se konča, ko kodni besedi sporočila sledi kodna beseda naslova ali pa kodna beseda nedejavnosti.

Kodna beseda sporočila, ki jo prikazuje slika 4.3, se vedno prične s stanjem prvega bita logično 1 in je vedno locirana takoj za kodno besedo naslova. Biti, ki se nahajajo na zaporednem mestu od 2 do 21, so biti sporočila. Ti vsebujejo besedilo, ki je bilo poslano. Biti na zaporednem mestu od 22 do 31 so BCH biti, bit na zaporednem mestu 32 pa je namenjen preverjanju parnosti [11].



Slika 4.3: Struktura kodne besede sporočila

4.3.3 Struktura kodne besede nedejavnosti

Kodna beseda nedejavnosti je rezervirana beseda, ki se uporablja za zapolnitev okvirjev, ki ne vsebujejo kodne besede sporočila ali kodne besede naslova. Če vsebuje okvir samo kodno besedo naslova (pri tonskih pozivnikih), se uporabi kodna beseda nedejavnosti za zapolnitev 64-bitnega okvirja. Njena heksadecimalna vrednost je 0x7A89C197 oziroma binarno [11]:
 0111 1010 1000 1001 1100 0001 1001 0111_(BIN)

4.4 FSC KODA

FSC (angl. Frame Synchronization Code) koda je 32-bitna standardizirana koda, namenjena identifikaciji začetka vsakega paketa. Njena heksadecimalna vrednost je 0x7CD215D8 oziroma binarno [11]:
 0111 1100 1101 0010 0001 0101 1101 1000_(BIN)

4.5 ALFA NUMERIČNA SPOROČILA

So zaporedja števil, črk in simbolov, pretvorjena v 7-bitne ASCII znake. Kodna beseda sporočila ima 20 bitov, rezerviranih za te znake. Ker je velikost enega znaka 7 bitov, pomeni, da se zadnji bit tretjega znaka prenese v naslednjo kodno besedo sporočila. Zadnja kodna beseda se zapolni z znaki, ki jih ni mogoče natisniti. To so na primer »end of message«, »end of text« ali »null«. Tabela 4.2 prikazuje ASCII znake v povezavi z bitim.

Najdaljša dolžina standardnega alfa numeričnega sporočila je 40 znakov, v nekaterih izvedbah omrežij do 250 znakov [11].

Tabela 4.2: ASCII tabela

ASCII tabela				Bit 7	0	0	0	0	1	1	1	1
				Bit 6	0	0	1	1	0	0	1	1
				Bit 5	0	1	0	1	0	1	0	1
Bit 4	Bit 3	Bit 2	Bit 1									
0	0	0	0	NULL	DLE	SP	0		P		p	
0	0	0	1	SOH	DC	!	1	A	Q	a	q	
0	0	1	0	STX	DC	"	2	B	R	b	r	
0	0	1	1	ETX	DC	#	3	C	S	c	s	
0	1	0	0	EOT	DC	\$	4	D	T	d	t	
0	1	0	1	ENQ	NAK	%	5	E	U	e	u	
0	1	1	0	ACK	SYN	&	6	F		f	v	
0	1	1	1	BEL	ETB	'	7	G	W	g	w	
1	0	0	0	BS	CAN	(8	H	X	h	x	
1	0	0	1	HT	EM)	9	I	Y	i	y	
1	0	1	0	LF	SUB	*	:	J	Z	j	z	
1	0	1	1	VT	ESC	+	;	K		k		
1	1	0	0	FF	FS	,	<	L		l		
1	1	0	1	CR	GS	-	=	M		m		
1	1	1	0	O	RS	.	>	N	^	n		
1	1	1	1	SI	US	/	?	O	_	o	DEL	

4.6 NUMERIČNA SPOROČILA

Numerična sporočila se pošiljajo kot 4-bitne BCD (angl. Binary – Coded Decimal) vrednosti. Tabela 4.3 prikazuje numerično vrednost v povezavi z ASCII znakom.

Tabela 4.3: BCD Tabela

Numerična vrednost	ASCII	ASCII znak	Numerična vrednost	ASCII	ASCII znak
0x0	0	0x30	0x8	8	0x38
0x1	1	0x31	0x9	9	0x39

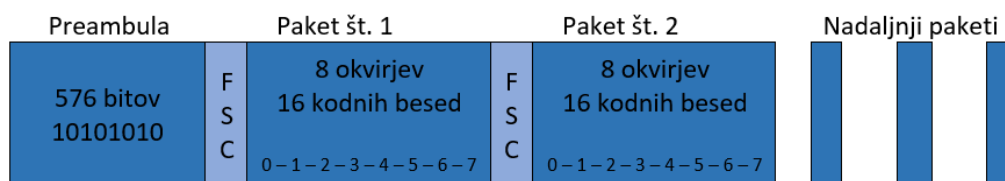
»se nadaljuje«

»nadaljevanje«

Numerična vrednost	ASCII	ASCII znak	Numerična vrednost	ASCII	ASCII znak
0x2	2	0x32	0xA	*	0x2A
0x3	3	0x33	0xB	U	0x55
0x4	4	0x34	0xC	space	0x20
0x5	5	0x35	0xD	-	0x2D
0x6	6	0x36	0xE)	0x29
0x7	7	0x37	0xF	(0x28

4.7 POCSAG PAKET

Paket se začne s FSC kodo, ki ji sledi 8 okvirjev. Vsak okvir je sestavljen iz dveh kodnih besed. V paketu se lahko pošlje največ 16 naslovnih kodnih besed. Vsak okvir ima lahko dve kodni besedi naslova ali dve kodni besedi sporočila ali dve kodni besedi nedejavnosti. Slika 4.4 prikazuje strukturo sporočila [11].



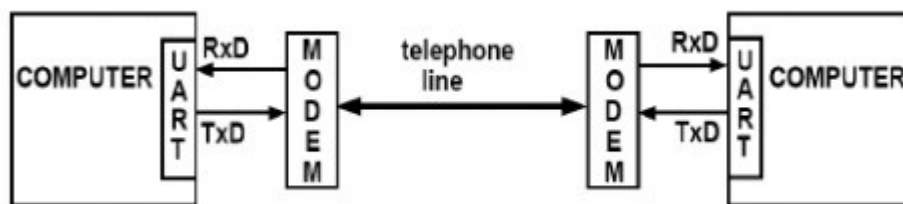
Slika 4.4: Struktura POCSAG sporočila

5 UART

UART (angl. Universal Asynchronous Receiver – Transmitter) se imenuje integrirano vezje, ki je sprogramirano tako, da nadzoruje standardizirani računalniški vmesnik RS-232C in nanj priključene naprave, ki uporabljajo serijsko komunikacijo, med katerimi so najpogostejši modemi. UART omogoča tudi nekaj naslednjih osnovnih operacij:

- pretvorba bajtov, ki jih dobi iz računalnika, v enoten serijski niz bitov, pripravljenih za oddajanje;
- pri sprejemanju podatkov pretvori serijski niz bitov v bajte;
- pri sprejemanju preveri bit za preverjanje parnosti;
- pri oddajanju doda začetni in končni mejnik podatkov (začetni in končni bit), v primeru sprejemanja pa ju odstrani;
- pri računalniku nadzoruje tipkovnico in miško.

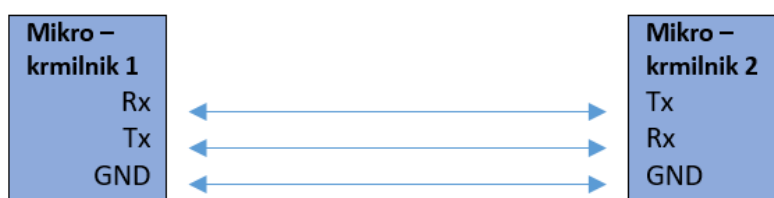
Slika 5.1 prikazuje shematski prikaz serijskega prenosa podatkov med računalnikoma, ki sta med sabo priključena preko telefonske linije in modema.



Slika 5.1: Serijski prenos podatkov [4]

Tri glavne komponente, ki sestavljajo UART, so: nadzor oddajanja, nadzor sprejemanja in generator hitrosti prenosa podatkov (angl. baud rate generator). V načinu oddajanja UART vzame osem bitov paralelnih podatkov in jih pretvori v serijski niz bitov z začetnim bitom logično 0 in končnim bitom logično 1. V načinu sprejemanja UART sprva zazna začetni bit, sprejme niz n bitov, kjer je $5 < n < 9$, in jih pretvori v paralelni način v trenutku, ko zazna

končni bit. Ker se prenos podatkov izvaja asinhrono, pomeni, da se hitrost procesorja ne oddaja posebej, zato mora UART sinhronizirati vhodni podatkovni niz z lokalno hitrostjo procesorja. Slika 5.2 prikazuje osnovni način povezave dveh mikro-krmilnikov z uporabo UART.



Slika 5.2: Povezava dveh mikro-krmilnikov z uporabo UART

Nastavitev UART je v večini primerov dokaj enostavna. Izbrati je potrebno hitrost prenosa podatkov, parnost, število podatkovnih bitov in število končnih bitov. Najpogostejša nastavitev je 8 podatkovnih bitov brez parnosti in eden končni bit. Hitrost prenosa podatkov izberemo glede na namen uporabe. V praksi se za ožičeno povezavo uporablja 9600 bitov na sekundo. UART pošilja podatke v paketu, ki je prikazan na sliki 5.3. Vsak paket vsebuje začetni bit, 5 do 9 podatkovnih bitov, parnostni bit in 1 ali 2 končna bita [4].



Slika 5.3: UART paket

5.1 ZAČETNI BIT

Linija za oddajanje podatkov – Tx drži visoko napetostno raven, ko ta ne prenaša podatkov. Da se prenos lahko začne, mora UART, ki oddaja, napetost na liniji Tx pritisniti na nizko napetostno raven za en časovni cikel takta procesorja. Ko sprejemni UART zazna tranzicijo z visoke na nizko napetostno raven, prične z branjem bitov v podatkovnem okvirju.

5.2 PODATKOVNI OKVIR

Ta vsebuje dejanski podatek, ki se prenaša. Njegova dolžina je med 5 in 8 bitov. Če se parnostni bit ne uporablja, je lahko njegova dolžina največ 9 bitov.

5.3 PARNOSTNI BIT

Namen parnostnega bita je omogočiti sprejemnemu UART, da zazna, če se je kakšen bit med prenosom spremenil. Ti se lahko spremenijo iz različnih vzrokov, kot so elektromagnetno valovanje, različne hitrosti prenosa podatkov na sprejemni in oddajni strani, dolge razdalje... Ko se parnostni bit ujema s podatki, to pomeni, da med prenosom le-h ni bilo napake.

5.4 KONČNI BITI

UART, ki oddaja, signalizira konec prenosa podatkov tako, da napetost na liniji Tx pritisne na visoko napetostno raven za čas trajanja najmanj dveh bitov.

6 RADIJSKA POSTAJA ISKRA TR40M

Radijska postaja iskra TR40M je mobilni radijski sprejemnik – oddajnik, s katerim lahko vzpostavljamo simpleksne in semiduplexsne zveze. Njeno frekvenčno področje delovanja je od 146 do 174 MHz pri razmiku med kanali (frekvenčni korak) 12,5 kHz. Funkcijo krmilnika postaje ima integrirano vezje M2716, ki ima EPROM (angl. Erasable Programmable Read – Only Memory) spomin, ki ga je možno reprogramirati z ustrežno opremo. Splošni tehnični podatki so podani v tabeli 6.1. Slika 6.1 prikazuje radijsko postajo [5].

Tabela 6.1: Iskra TR40M - tehnični podatki

LASTNOST:	VREDNOST:	LASTNOST:	VREDNOST:
Frekvenčno področje:	od 146 do 174 MHz	Število kanalov:	12
Vrsta modulacije:	Frekvenčna (FM)	Razmik med kanali:	12,5 kHz
Stabilnost frekvence:	$\pm 1,5$ kHz	Temperatura skladiščenja:	od -25 °C do +45 °C
Temperatura obratovanja:	od -25 °C do +45 °C	Impedanca antene:	50 Ω
Napajanje:	13,8 V nominalno 11 V minimalno 16 V maksimalno	Masa:	1,2 kg
Dimenzije:	40 x 180 x 188 mm	Poraba:	Sprejem: 170 mA Oddaja: 3 A
Izhodna VF moč:	10 W	Izhodna NF moč:	2 W



Slika 6.1: Radijska postaja Iskra TR40M

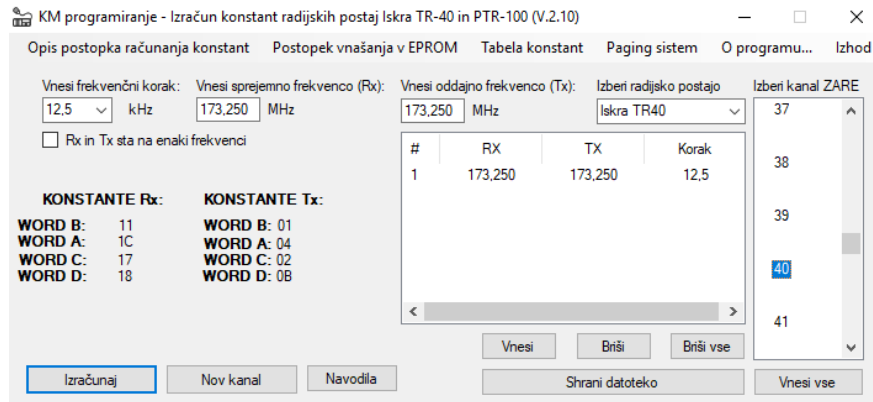
6.1 OPIS PREDELAVE

Radijsko postajo smo najprej reprogramirali na ustrezen ZaRe kanal. Ker ima krmilnik M2716 EPROM spomin, smo ga odstranili iz tiskanega vezja in ga s pomočjo ultravijolične žarnice izbrisali. Za lažje programiranje smo ustvarili program za programiranje radijskih postaj Iskra TR-40 in PTR-100. Ta na podlagi vhodnih podatkov (frekvenčni korak, sprejemna in oddajna frekvenca, model radijske postaje) generira binarno datoteko, ki vsebuje potrebne parametre za programiranje krmilnika.

Program je z dovoljenjem GNU tudi javno objavljen na povezavi:

<https://sourceforge.net/projects/iskra-tr40-in-ptr100-program/> [15.7.2019]

Radijska postaja ima na voljo 12 kanalov. Programirali smo samo prvega. Postopek dela prikazuje slika 6.2.



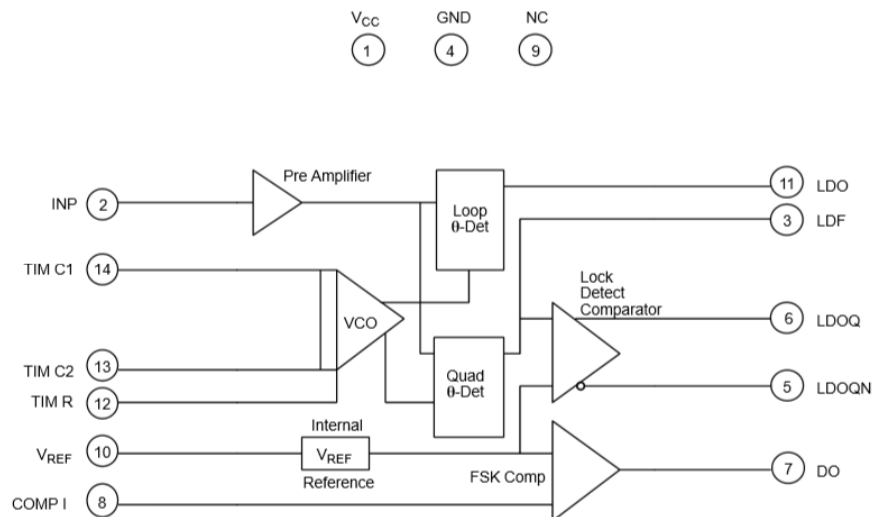
Slika 6.2: Vnos podatkov v program za programiranje radijske postaje

Krmilnik M2716 smo vstavili v programator in ga s pomočjo programske opreme in binarne datoteke, ki smo jo ustvarili, sprogramirali. Po končanem programiranju smo ga vstavili v tiskano vezje radijske postaje.

Za uspešno demodulacijo FSK signala potrebujemo vir avdio signala. S preučevanjem vezja radijske postaje smo ugotovili, da ta uporablja FM PLL (angl. Phase – Locked Loop) detektorsko integrirano vezje SL6601C. S preučevanjem podatkovnega lista proizvajalca smo ugotovili, da se frekvenčno demoduliran avdio signal pojavi na kontaktu (angl. pin) 8 [5], [12].

7 FSK DEMODULATOR EXAR XR-2211A

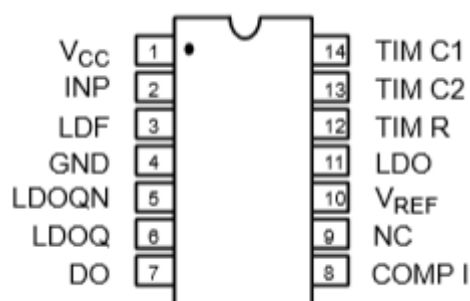
XR-2211A FSK demodulator proizvajalca Exar uporablja PLL sistem, ki je posebej konstruiran za uporabo v komunikacijski tehnologiji. Ima širok napajalni napetostni razpon od 4.5 do 20 V. Frekvenčno področje delovanja je od 0.01 do 300 kHz. Vhodni analogni signal ima lahko vrednosti od 10 mV do 3 V, njegov izhod pa lahko direktno priključimo na UART vhod. Vezje sestavljajo PLL za sledenje vhodnemu signalu znotraj željenega frekvenčnega področja, fazni detektor, ki zazna nosilno frekvenco, in napetostni primerjalnik, ki skrbi za demodulacijo. Za določitev sredinske frekvence, hitrost prenosa podatkov in ostalih parametrov se uporabljajo zunanje komponente (upori, kondenzatorji...). Slika 7.1 prikazuje blokovno shemo demodulatorja.



Slika 7.1: Blokovna shema demodulatorja [7]

PLL je v demodulatorju izgrajen iz vhodnega predojačevalnika, analognega množilnika in napetostno kontroliranega oscilatorja (angl. Voltage Controlled Oscillator) – VCO. Predojačevalnik je uporabljen kot omejevalnik vhodnega signala tako, da drži vhodni signal, ki presega 10 mV efektivne napetosti, ves čas na konstantni napetostni ravni. Fazni detektor se v vezju obnaša kot digitalna »ekskluzivna ali« vrata. Njegov izhod je seštevek in razlika

frekvenc vhodnega signala in VCO izhoda. VCO je tako rekoč tokovno kontroliran oscilator, ki se mu normirani vhodni tok nastavlja z uporabo R_0 proti masi in pogonski tok z uporabo R_1 , ki je povezan s faznim detektorjem. Slika 7.2 opisuje postavitev kontaktov, tabela 7.1 pa njihov opis. Zaradi večje preglednosti je opis kontaktov v angleškem jeziku [7].



Slika 7.2: Konfiguracija kontaktov [7]

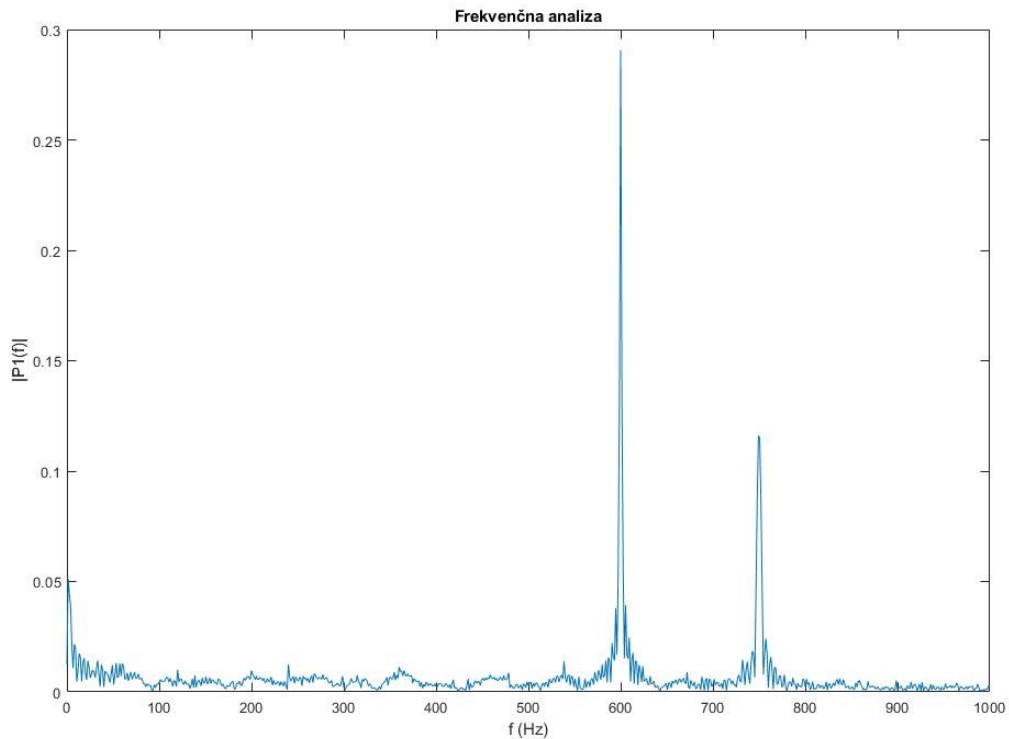
Tabela 7.1: Opis kontaktov [7]

Kontakt:	Simbol:	Opis:
1	V _{CC}	Positive power supply
2	INP	Analog input
3	LDF	Lock detect filter
4	GND	Ground
5	LDOQN	Lock detect output »not«
6	LDOQ	Lock detect output
7	DO	Data output
8	COMP I	FSK comparator input
9	NC	Not connected
10	V _{REF}	Internal voltage reference
11	LDO	Loop detect output
12	TIM R	Timing resistor input
13	TIM C ₂	Timing capacitor input
14	TIM C ₁	Timing capacitor input

7.1 DOLOČITEV FREKVENCE PIKE IN PRESLEDKA

Za določitev frekvence pike in presledka smo uporabili računalniški program Matlab [17]. S pomočjo računalnika smo posneli FSK moduliran POCSAG avdio signal, ki je bil prestrežen na radijski postaji. V programu Matlab smo uporabili funkcijo »fft«, s katero smo s pomočjo Fourierjeve transformacije analizirali avdio signal. Slika 7.3 prikazuje frekvenčno analizo, na kateri je prikazana amplituda v odvisnosti od frekvence. Razvidno je, da je signal sestavljen iz dveh frekvenc, ki izrazito izstopata. Ugotovili smo, da je frekvenca pike $F_1 = 750$ Hz, frekvenca presledka pa $F_2 = 600$ Hz.

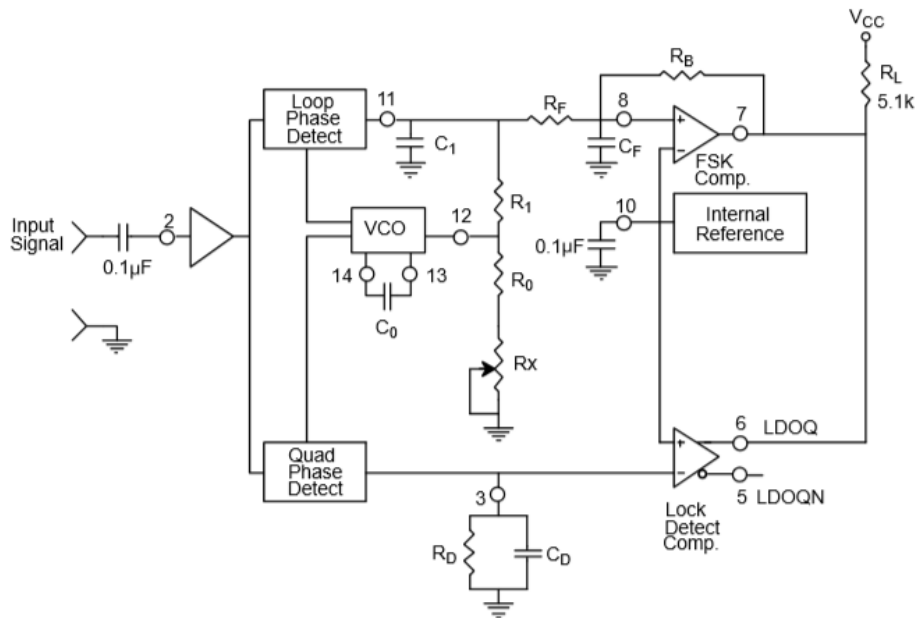
Določitev hitrosti prenosa podatkov je bila relativno enostavna. Ogledali smo si nastavitve enega od pozivnikov, ki preverjeno delujejo na omrežju. Ta znaša 1200 bit/s.



Slika 7.3: Frekvenčna analiza avdio signala

7.2 PRERAČUN VREDNOSTI ZUNANJIH KOMPONENT

Proizvajalec v podatkovnem listu navaja postopek preračuna vrednosti uporov in kondenzatorjev, ki so potrebni za pravilno delovanje demodulatorja. Shema vezja, ki ga bomo uporabili, prikazuje slika 7.4. Potrebujemo 7 ključnih vrednosti: R_0 , R_1 , C_0 , C_1 , C_F , frekvenco pike f_0 in frekvenco presledka f_1 . Vse vrednosti elementov, razen vrednosti R_0 , se lahko zaokrožijo na najbližjo standardno vrednost [7].



Slika 7.4: Shema vezja demodulatorja [7]

- Izračunamo PLL sredinsko frekvenco, kjer je $f_0 = F_1$, $f_1 = F_2$:

$$f_0 = \sqrt{F_1 * F_2} = \sqrt{750 * 600} = 670,82 \text{ Hz} \quad (3.1)$$

- Izberemo vrednost upora R_0 , ki mora biti med 10 k Ω in 100 k Ω . Izberemo 25 k Ω . Potenciometer R_x se uporablja za fino nastavitvev sredinske frekvence. Izbrali bomo vrednost 10 k Ω .

$$R_0 = R_0 + \frac{R_x}{2} = 25000 + \frac{10000}{2} = 30 \text{ k}\Omega \quad (3.2)$$

- Izračunamo vrednost C_0 :

$$C_0 = \frac{1}{R_0 * f_0} = \frac{1}{25000 * 670,82} = 5,96 * 10^{-8} F \quad (3.3)$$

Zaokrožimo na standardno vrednost 0,068 μ F

- Izračunamo R_1 . Ta vrednost nam določa pasovno širino sledenja, kjer je $f_0=F_1$, $f_1=F_2$.

$$R_1 = \frac{R_0 * f_0}{(F_1 - F_2)} * 2 = \frac{25000 * 670,82}{(750 - 600)} * 2 = 223,606 k\Omega \quad (3.4)$$

Zaokrožimo na standardno vrednost 220 k Ω .

- Izračunamo C_1 , kjer je $x = 0.5$:

$$C_1 = \frac{1250 * C_0}{R_1 * x^2} = \frac{1250 * 0,068 * 10^{-6}}{220 * 10^3 * 0,5^2} = 1,545 * 10^{-9} F \quad (3.5)$$

Zaokrožimo na standardno vrednost 1,55 nF.

- Izračunamo R_F :

$$R_F = R_1 * 5 = 220000 * 5 = 1,1 M\Omega \quad (3.6)$$

- Izračunamo R_B :

$$R_B = R_F * 5 = 5,5 M\Omega \quad (3.7)$$

- Izračunamo R_{SUM} :

$$\frac{(R_F + R_1) * R_B}{(R_F + R_1 + R_B)} = \frac{(1,1M + 220k) * 5,5M}{(1,1M + 220k + 5,5M)} = 1,065 M\Omega \quad (3.8)$$

Zaokrožimo na standardno vrednost 1,1 M Ω .

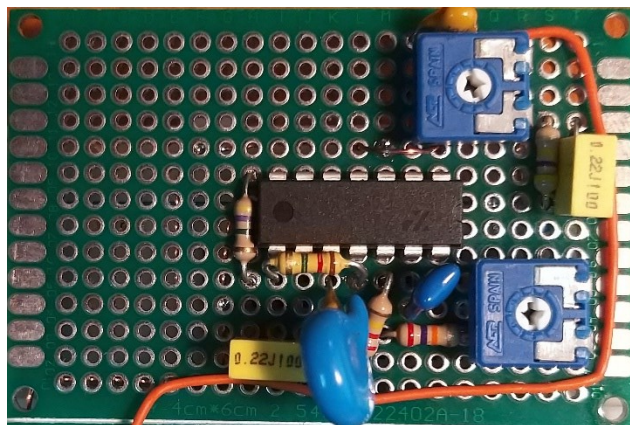
- Izračunamo C_F , kjer je hitrost prenosa podatkov = 1200 bit/s:

$$C_F = \frac{250}{(R_{SUM} * \text{hitrost prenosa podatkov})} = \frac{250}{(1,1M * 1200)} = 0,1894 \mu F \quad (3.9)$$

Zaokrožimo na standardno vrednost 0,22 μ F.

7.3 KONČNO VEZJE

Slika 7.5 prikazuje končno vezje, ki smo ga izdelali na pripravljene ploščici za tiskana vezja. Na vhodni kontakt 2 XR-2211A FSK demodulatorja smo dovedli avdio signal iz kontakta 8 integriranega vezja SL6601C radijske postaje Iskra TR40M.



Slika 7.5: Končno vezje demodulatorja

8 GSM MODEM

Uporabili smo SIM800L GSM modem. Gre za majhen GSM/GPRS modul velikosti 15,8 x 17,8 x 2,4 mm, ki deluje na frekvencah 850 MHz, 900 MHz, 1800 MHz in 1900 MHz.

Ponaša se s sledečimi lastnostmi [19], te so:

- podpora 5 x 5 x 2 tipkovnice
- UART vrata
- USB vrata (za prenos programske opreme)
- avdio kanal (priklop mikrofona in zvočnika)
- reža za SIM kartico

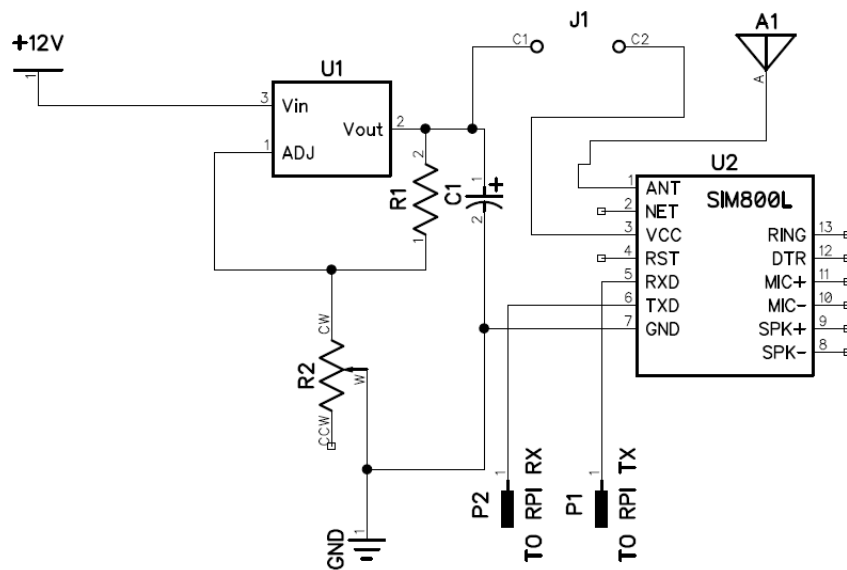
V tabeli 8.1 so prikazane nekatere tehnične podrobnosti naprave.

Tabela 8.1: SIM800L – tehnični podatki [19]

Lastnost:	Opis:
Napajalna napetost	od 3,4 V do 4,4 V
Varčevanje z energijo	0,7 mA v načinu spanja
Moč oddajanja	2 W pri frekvenci 850 MHz in 900 MHz 1W pri frekvenci 1800 MHz in 1900 MHz
Delovna temperatura:	Od -40 °C do 85 °C
Temperatura skladiščenja:	Od -45 °C do 90 °C
UART vrata	asinhronska komunikacija hitrosti prenosa podatkov od 1200 bit/s do 115200 bit/s podpora AT komand

Ker potrebujemo modem samo za pošiljanje SMS (angl. Short Message Service) sporočil, bomo uporabili samo kontakte, ki jih potrebujemo za priklop modema na mikroračunalnik Raspberry Pi preko UART. Potrebovali bomo sledeče kontakte: GND, TXD, RXD, V_{CC} in priključek za anteno.

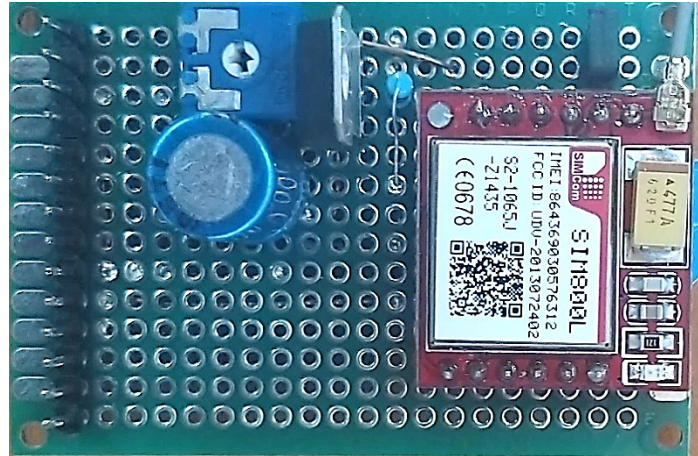
Ker potrebuje modem napajalno napetost od 3,4 do 4,4 V, smo uporabili stabilizator napetosti LM317T. Slika 8.1 prikazuje shemo vezja za stabilizacijo napetosti in priključitev modema. Stabilizatorju napetosti U_1 smo na izhodni strani dodali $100\mu\text{F}$ kondenzator C_1 , ki skrbi za zagotovitev energije ob vklopnem tokovnem pojavu. Trimer R_2 , katerega vrednost znaša $10\text{ k}\Omega$, nam omogoča umeritev izhodne napetosti stabilizatorja na zahtevan nivo (izhodno napetost smo nastavili na $3,8\text{ V}$), mostiček (angl. jumper) – J_1 pa nam omogoča odklop napajalnega kontakta modema od stabilizatorja napetosti za potrebe umerjanja le-te. Vrednost upora R_1 znaša $240\ \Omega$, A_1 predstavlja GSM anteno, P_1 in P_2 pa UART povezavo z mikroračunalnikom Raspberry Pi.



Slika 8.1: Shema vezja stabilizatorja napetosti in priključitve modema

8.1 KONČNO VEZJE

Slika 8.2 prikazuje končno vezje, ki smo ga izdelali na pripravljene ploščici za tiskano vezja.



Slika 8.2: Končno vezje modema

9 RASPBERRY PI

Za dekodiranje POCSAG protokola in pošiljanje SMS sporočil smo uporabili mikroračunalnik Raspberry Pi 2 model B. Procesor s taktom 900 MHz in 1 GB bralno-pisalnega (angl. Random Access Memory) – RAM spomina, UART priključkom ter LCD (angl. Liquid – Crystal Display) priklopom so zagotovili vse naše potrebe. Slika 9.1 prikazuje mikroračunalnik, ki se ponaša s sledečimi lastnostmi [14]:

- Broadcom BCM2837 Arm7 Quad Core Procesor pri delovnem taktu 900 MHz
- 1 GB RAM
- 4 x USB vtičnica
- stereo izhod, kompozitni video izhod
- HDMI vtičnica
- CSI priključek za kamero
- DSI priključek za LCD ekran na dotik
- micro SD vtičnica za spominsko kartico
- micro USB vtičnica za napajanje
- 40-kontaktni vhodno-izhodni priključek
- priključek za internet
- 1 x UART



Slika 9.1: Mikroračunalnik Raspberry Pi 2 model B [14]

Ker ima mikroračunalnik na voljo le en UART, mi pa potrebujemo dva – enega za prejemanje podatkov iz demodulatorja, drugega pa za komunikacijo z GSM modemom – smo kot drugi UART uporabili USB v UART pretvornik, ki smo ga vstavili v eno od USB vtičnic na mikroračunalniku.

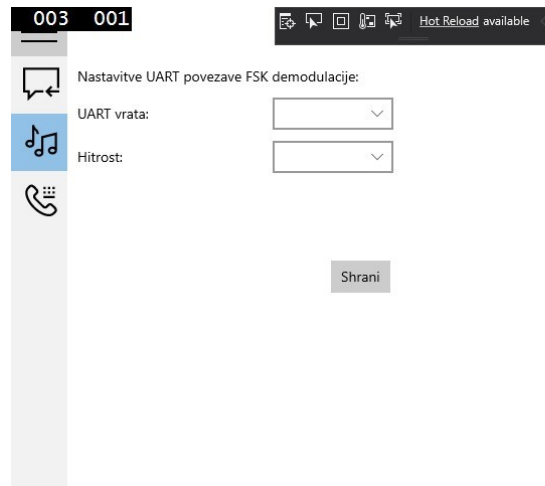
9.1 PROGRAMIRANJE

Za programiranje grafičnega vmesnika in implementacijo dekodiranja POCSAG sporočila smo uporabili Microsoft Visual Studio 2019 community edition. Raspberry Pi podpira več različnih operacijskih sistemov; mi smo uporabili Microsoft Windows IoT. Za začetek je bilo potrebno dobiti sliko operacijskega sistema, ki smo jo našli na Microsoftovi spletni strani in jo s pomočjo namenskega programa prenesli na spominsko kartico. Raspberry Pi smo priključili na lokalno ožičeno omrežje, saj se program, ki smo ga ustvarili, na mikroračunalnik prenaša preko spleta. Izdelali smo preprost grafični vmesnik, ki ima na levi strani meni, s katerim lahko preklapljam med tremi različnimi stranmi. Osnovna stran, poimenovana »Terminal«, nam prikazuje dekodirano RIC številko, funkcijska bita in dekodirano sporočilo, ki je bilo pred tem prestreženo in demodulirano. Kako je videti slika, je predstavljamo na sliki 9.2. Podatki v zgornjem delu slik grafičnega vmesnika so del razhroščevalnega načina mikroračunalnika in so brezpredmetni.



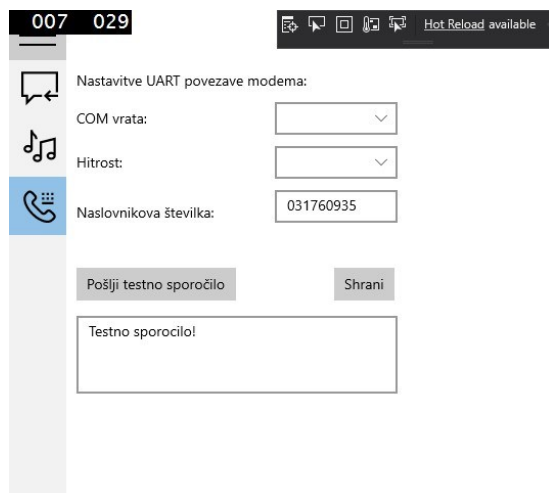
Slika 9.2: Grafični vmesnik – prva stran »Terminal«

Druga stran, poimenovana »Demodulacija«, nam prikazuje osnovne nastavitve UART vhoda, ki se uporablja za sprejemanje podatkov iz demodulatorja. Stran lahko vidimo na sliki 9.3.



Slika 9.3: Grafični vmesnik – druga stran »Demodulacija«

Tretja stran nam prikazuje osnovne nastavitve UART vhoda, ki se uporablja za komunikacijo z GSM modemom, okno za vnos naslovnikove telefonske številke in možnost pošiljanja testnega sporočila. Izgled strani lahko vidimo na sliki 9.4.



Slika 9.4: Grafični vmesnik – tretja stran »Modem«

V nadaljevanju je bilo potrebno izdelati program za dekodiranje POCSAG sporočila. V tabeli 9.1 je prikazan primer sporočila, kodiranega z uporabo protokola POCSAG, ki vsebuje CAP kodo z vrednostjo 1234567 in z besedilom »TEST« .

Tabela 9.1: Sporočilo POCSAG [11]

#	Sporočilo:							
1	7CD215D8							
2	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197
3	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	4B5A1A25	95A393FC
4	7CD215D8							
5	CA902290	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197
6	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197
7	7CD215D8							
8	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197
9	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197
10	7CD215D8							
11	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197
12	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197	7A89C197

Program deluje tako, da ves čas posluša UART vhod, kamor podatki prihajajo serijsko in kodirano v ASCII znake. Ker se vsako POCSAG sporočilo prične s preambulo, bo ta čakal na trenutek, ko te ne bo več na sprejemu. V nadaljevanju morata biti izpolnjena dva pogoja: da je na sprejemu FSC koda, katere heksadecimalna vrednost je 7CD215D8, in da so bili biti pred njo preambula.

FSC koda ponazarja začetek vsakega paketa, konec preambule pa začetek sporočila. Če povežemo to trditev s tabelo 9.1, se sedaj nahajamo na zaporedni številki vrstice 1. Heksadecimalne vrednosti 7A89C197 predstavljajo kodno besedo nedejavnosti. V nadaljevanju čaka program na prvi podatek, ki je različen od kodne besede nedejavnosti, hkrati pa šteje, koliko je bilo okvirjev znotraj paketa, saj se ta podatek uporablja pri sestavljanju CAP kode. Enega izmed paketov v tabeli 9.1 predstavlja besedilo na zaporedni številki vrstice 1, 2 in 3, enega izmed okvirjev pa besedilo na zaporedni številki vrstice 2, prvi in drugi stolpec. Prvi podatek, ki bo teoretično prispel, je kodna beseda naslova. V tabeli

9.1 se nahaja v vrstici 3 in ima heksadecimalno vrednost 4B5A1A25. Če to vrednost pretvorimo v binarno obliko, dobimo:

$$4B5A1A25_{(HEX)} = 0100\ 1011\ 0101\ 1010\ 0001\ 1010\ 0010\ 0101_{(BIN)}$$

Bit na prvem mestu ponazarja začetek kodne besede naslova, biti od mesta 2 do 19 pa so naslovni biti in se uporabljajo pri sestavljanju CAP kode. Da jo iz sporočila dekodiramo, najprej izpišemo naslovne bite:

$$100\ 1011\ 0101\ 1010\ 000_{(BIN)}$$

Ker se kodna beseda naslova nahaja na sedmem mestu (štetje pričnemo z 0), se ta podatek pretvori v binarno obliko.

$$7_{(DEC)} = 0111_{(BIN)}$$

CAP koda je sestavljena iz 18 naslovnih bitov in 3 bitov, ki ponazarjajo lokacijo okvirja znotraj paketa. Dobili smo 4 lokacijske bite, zato zanemarimo tistega, ki je najmanj pomemben, in dobimo:

$$111_{(BIN)}$$

Zdaj lahko naslovne bite združimo z lokacijskimi in dobimo:

$$10010\ 1101\ 0110\ 1000\ 0111_{(BIN)}$$

Pretvorimo iz binarne v decimalno vrednost:

$$10010\ 1101\ 0110\ 1000\ 0111_{(BIN)} = 1234567_{(DEC)}$$

CAP ali RIC koda je dekodirana, njena vrednost pa znaša 1234567. Nadaljujemo z dekodiranjem besedila sporočila. Teoretično kodni besedi naslova sledi kodna beseda sporočila. Ker je v posamezni kodni besedi sporočila prostora za največ dva 7-bitna ASCII znaka, ki predstavljata dejansko besedilo sporočila, se v primeru več znakov biti prenesejo v naslednjo kodno besedo sporočila. V tabeli 9.1 se kodna beseda sporočila nahaja v vrstici 3 in ima heksadecimalno vrednost 95A393FC. Če to vrednost pretvorimo v binarno obliko, dobimo:

$$95A393FC_{(HEX)} = 1001\ 0101\ 1010\ 0011\ 1001\ 0011\ 1111\ 1100_{(BIN)}$$

Bit na prvem mestu ponazarja začetek kodne besede sporočila, biti od mesta 2 do 21 pa so ASCII znaki. Da dekodiramo besedilo iz kodne besede sporočila, najprej izpišemo bite sporočila:

0010 1011 0100 0111 0010_(BIN),

vzamemo prvih 7 bitov in jih s pomočjo tabele 4.2 pretvorimo v znak:

0010101_(BIN) = T,

vzamemo drugih 7 bitov in jih s pomočjo tabele 4.2 pretvorimo v znak:

1010001_(BIN) = E.

Ostalo je 6 bitov. Vzamemo naslednji paket in ga pretvorimo v binarno obliko:

CA902290_(HEX) = 1100 1010 1001 0000 0010 0010 1001 0000_(BIN).

Izpišemo bite sporočila:

1001 0101 0010 0000 0100.

Vzamemo prvi bit, ga pripišemo k šestim bitom, ki so ostali v prejšnji kodni besedi sporočila in dobljene bite s pomočjo tabele 4.2 pretvorimo v znak:

1100101_(BIN) = S.

Vzamemo naslednjih 7 bitov in jih s pomočjo tabele 4.2 pretvorimo v znak:

0010101_(BIN) = T.

Vzamemo naslednjih 7 bitov in jih s pomočjo tabele 4.2 pretvorimo v znak:

0010000_(BIN) = »EOT«.

Znak »EOT« je kratica za konec oddajanja (angl. End Of Transmission) in ponazarja konec sporočila. Program jo zazna in preneha z dekodiranjem, saj vsi nadaljnji podatki predstavljajo kodno besedo nedejavnosti, katere namen je zapolniti okvirje, ki ne vsebujejo podatka.

CAP koda in sporočilo se izpišeta na strani »Terminal«. Sledi pošiljanje SMS sporočila. Ker modem uporablja standardizirane AT komande za komunikacijo preko UART, je implementacija relativno enostavna. Tabela 9.2 prikazuje komande in vrstni red pošiljanja preko UART, kjer komanda 1 postavi modem v tekstovni način, komanda 2 določi telefonsko številko prejemnika, kjer x ponazarja telefonsko številko, komanda 3 besedilo sporočila, komanda 4 pa ASCII znak za CTRL-Z .

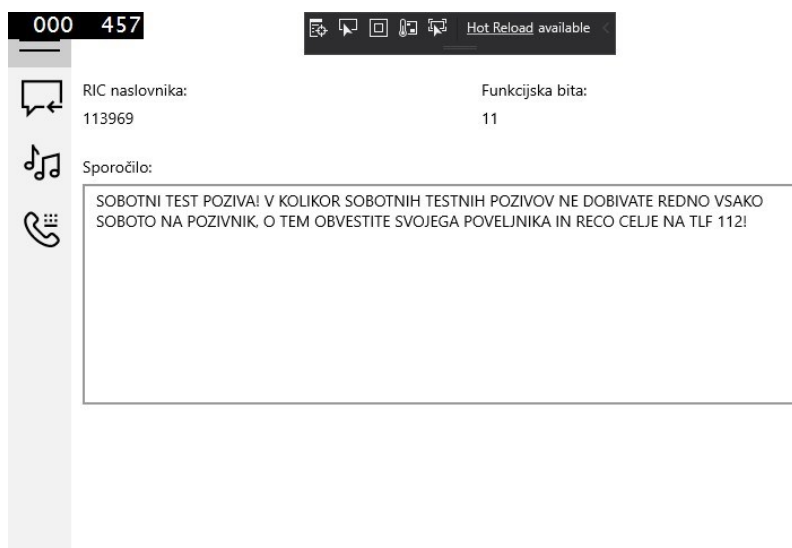
Tabela 9.2: Prikaz uporabljenih AT komand

#	Komanda:
1	AT+CMGF=1
2	AT+CMGS="\x\r"
3	TEST
4	Char(26)

10 PRIMER UPORABE

10.1 IZPIS SPOROČILA NA LCD ZASLONU

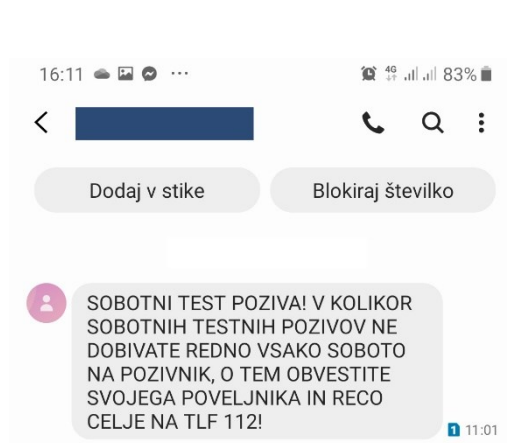
Slika 10.1 prikazuje dekodirano POCSAG sporočilo, izpisano na LCD zaslonu.



Slika 10.1: Izpis sporočila na LCD zaslonu

10.2 IZPIS SPOROČILA NA TESTNI TELEFONSKI ŠTEVILKI

Slika 10.2 prikazuje POCSAG sporočilo, ki je bilo posredovano v GSM omrežje.



Slika 10.2: Izpis sporočila na testni telefonski številki

11 KONČNI IZDELEK

Vse komponente, ki smo jih uporabili pri izdelavi naprave za posredovanje sporočil pozivnikov na GSM mobilno omrežje, smo postavili v kovinsko ohišje. Ker smo potrebovali čelno ploščo za vgradnjo LCD zaslona in ostalih komponent, smo se odločili, da jo bomo s pomočjo programa Solidworks [18] narisali in risbo posredovali zunanjemu izvajalcu, ki jo je lasersko izrezal iz kovine. Slika 11.1 prikazuje risbo plošče v pomanjšanem merilu.



Slika 11.1: Čelna plošča – risba

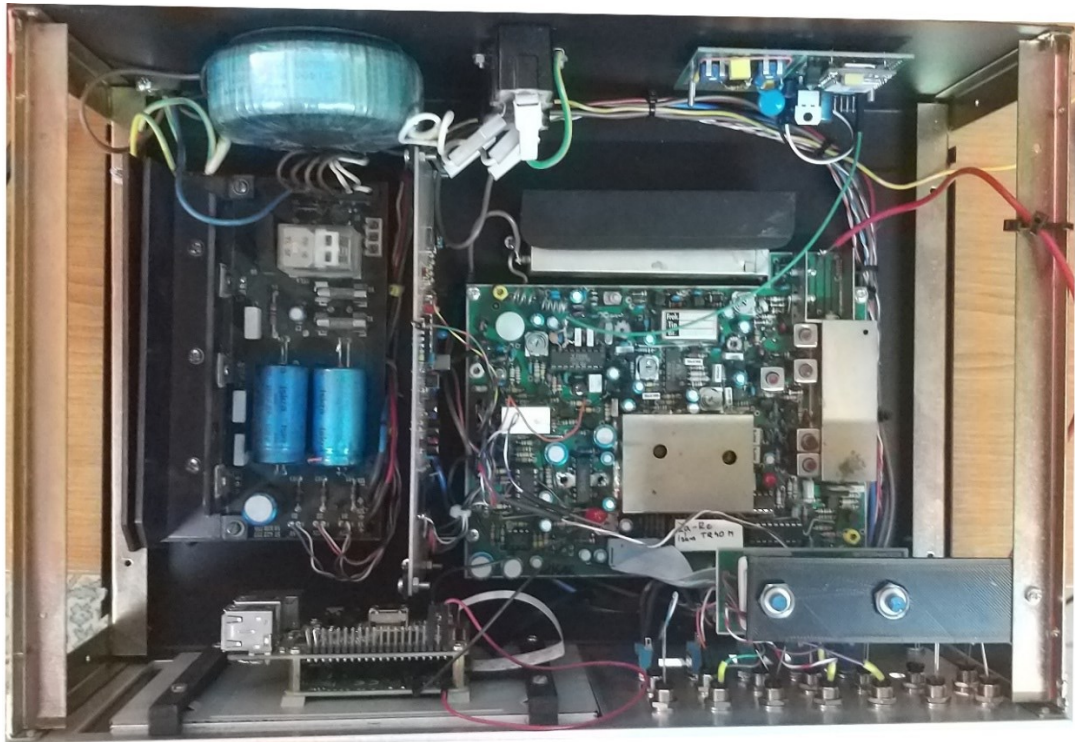
Pri vgradnji komponent v ohišje smo potrebovali nekaj komadov, izdelanih iz umetne mase. Ti so nam pomagali pritrditi komponente na svoje mesto. S pomočjo programa Solidworks smo jih narisali v 3D obliki in jih s pomočjo 3D tiskalnika natisnili. Slika 11.2 prikazuje eno od komponent radijske postaje, za katero smo narisali in natisnili dva pritrdilna nosilca črne barve.



Slika 11.2: Primer komadov iz umetne mase, natisnjenih s 3D tiskalnikom

Na sliki 11.3 in 11.4 je predstavljen končni izdelek. Čelno ploščo sestavlja 5 stikal, 14 LED diod, LCD zaslon in mikrofonski priključek. Zaradi možnosti razširitve funkcionalnosti naprave smo dodali nekaj LED diod in stikal. Dodali smo tudi odprtini, namenjeni naknadni vgradnji serijskega in USB priključka.

Notranjost sestavljajo napajalno vezje s transformatorjem in prenapetostno zaščito, GSM modem, FSK demodulator ter radijska postaja in njej pripadajoče komponente. Te smo pritrdili na ohišje s pomočjo vijakov, povezali vse signalne in napajalne kable in kovinsko ohišje ustrezno izolirali. Izbira kovinskega ohišja je bila dobra tudi zato, ker deluje kot Faradayeva kletka. Tako smo preprečili možnost pojava zunanjih interferenc na ključne komponente naprave.



Slika 11.3: Notranja razporeditev komponent naprave



Slika 11.4: Končni izdelek

12 SKLEP

Zastavljeni problem smo uspešno rešili. Naprava, ki smo jo izdelali, je uspešno prestregla, demodulirala, dekodirala in posredovala sporočilo na GSM omrežje. Zaradi uporabe zmogljivega mikroročunalnika smo dopustili možnost nadgradnje programske opreme. Dodali bi lahko možnost filtriranja in posredovanja sporočil več prejemnikom in ne samo enemu. Z iskanjem ključnih besed v sporočilu bi lahko dodali v napravo releje, preko katerih bi lahko avtomatsko odprli vrata garaže, vključili krovno sireno gasilskega doma, vključili luči ipd. Možnosti za razširitev je veliko.

Ugotovili smo, da je protokol POCSAG zanesljiv način brezžičnega prenašanja podatkov in bi se lahko uporabljal tudi v druge namene. Na področju energetike bi bil primeren za uporabo v črpališčih javnega vodovoda, pri čistilnih napravah, malih hidro elektrarnah in drugih napravah, kjer je potrebna komunikacija na daljavo za namene monitoringa ali obveščanja o napakah. Implementacija sistema bi bila relativno enostavna, saj bi od strojne opreme na sprejemni strani potrebovali le sprejemno radijsko postajo, FSK demodulator in računalnik s serijskim priključkom, na oddajni pa oddajno radijsko postajo, FSK modulator in računalnik s serijskim priključkom. Sistem bi bilo tudi možno izvesti tako, da bi bila komunikacija dvosmerna. Komercialne postaje imajo oddajno moč približno 10 W, kar omogoča brezžično komunikacijo do nekje 30 km, odvisno od terena, vremena in ostalih fizikalnih lastnosti. To je več kot dovolj, na primer v Občini Žalec, kjer so nekatera črpališča pitne vode in naprave v radiju od 10 do 20 km od središča mesta. S takšnim sistemom bi zmanjšali stroške, nastale zaradi potrebe po dostopu do internetnega omrežja za namene komunikacije.

Z napravo, ki smo jo izdelali, smo zadovoljni, saj je izpolnila naše zastavljene cilje.

VIRI IN LITERATURA

- [1] Marinko Valentin, Tavčar Boštjan: Sistem radijskih zvez in tihega alarmiranja, revija Ujma, Ministrstvo za obrambo: Uprava Republike Slovenije za zaščito in reševanje, Ljubljana, 1995.
- [2] Xiong Fuqin: Digital modulation techniques, Boston; London: Artech House, 2000.
- [3] Ozimek Igor, Hrovat Andrej: Concurrent POCSAG and TETRA paging, WSEAS transactions on communications, Athens: WSEAS press, 2005.
- [4] Umakanta Nanda, Sushant Kumar Pattnaik: Universal Asynchronous Receiver and Transmitter (UART). Dostopno na:
https://www.researchgate.net/publication/308988751_Universal_Asynchronous_Receiver_and_Transmitter_UART [15.7.2019].
- [5] Iskra Elektrozveze n.sol.o., radijska postaja TR40M/2, Izdanje 10881k, založba neznana, leto izdaje neznano.
- [6] Windows 10 IoT Core & Raspberry Pi. Dostopno na:
<https://devblogs.microsoft.com/premier-developer/getting-started-with-windows-10-iot-core-raspberry-pi-3b/> [15.7.2019].
- [7] Exar XR-2211A FSK demodulator. Dostopno na:
<https://www.maxlinear.com/Files/Documents/XR2211Av104.pdf> [15.7.2019].
- [8] FSK – Frequency shift keying. Dostopno na:
www.eng.auburn.edu/~tropical/courses/TIMS-manuals-r5/TIMS%20Experiment%20Manuals/Student_Text/Vol-D1/D1-07.pdf
[15.7.2019].
- [9] Frequency shift keying. Dostopno na:
https://www.dip.ee.uct.ac.za/~nicolls/lectures/eee482f/13_fsk_2up.pdf [15.7.2019].
- [10] FSK: Signals and Demodulation. Dostopno na:
<http://edge.rit.edu/edge/P0914/public/FSK.pdf> [15.7.2019].
- [11] The POCSAG paging protocol. Dostopno na:
[https://www.raveon.com/pdffiles/AN142\(POCSAG\).pdf](https://www.raveon.com/pdffiles/AN142(POCSAG).pdf) [15.7.2019].

- [12] SL6601C FM detector. Dostopno na: <https://pdf1.alldatasheet.com/datasheet-pdf/view/144131/ETC1/SL6601C.html> [15.7.2019].
- [13] Program Audacity. Dostopno na: <https://www.audacityteam.org/> [15.7.2019].
- [14] Raspberry Pi 2 model B. Dostopno na: <https://www.raspberrypi.org/products/raspberry-pi-2-model-b/> [15.7.2019].
- [15] Stabilizator napetosti LM317T. Dostopno na: <https://www.st.com/resource/en/datasheet/lm217.pdf> [15.7.2019].
- [16] Ministrstvo za obrambo, uprava Republike Slovenije za zaščito in reševanje. Dostopno na: <http://www.sos112.si/slo/page.php?src=pe32.htm> [15.7.2019].
- [17] Program Matlab. Dostopno na: <https://www.mathworks.com/products/matlab.html> [15.7.2019].
- [18] Program Solidworks. Dostopno na: <https://www.solidworks.com/> [15.7.2019].
- [19] SIM800L GSM/GPRS modem. Dostopno na: https://img.filipeflop.com/files/download/Datasheet_SIM800L.pdf [15.7.2019].

PRILOGE

**PRILOGA A: IZJAVA O AVTORSTVU IN ISTOVETNOSTI TISKANE IN
ELEKTRONSKE OBLIKE ZAKLJUČNEGA DELA**

**Priloga A – IZJAVA O AVTORSTVU IN ISTOVETNOSTI TISKANE IN ELEKTRONSKE OBLIKE
ZAKLJUČNEGA DELA**

UNIVERZA V MARIBORU
Fakulteta za energetiko
(ime članice UM)

IZJAVA O AVTORSTVU IN ISTOVETNOSTI TISKANE IN ELEKTRONSKE OBLIKE ZAKLJUČNEGA DELA

Ime in priimek študent-a/-ke: Matic Krajšek
Študijski program: Visokošolski študijski program 1. stopnje Energetika
Naslov zaključnega dela: Naprava za prestrezanje, dekodiranje in posredovanje sporočil
pozivnikov za GSM mobilno omrežje

Mentor: izr. prof. dr. Peter Vrtič
Somentor: _____

Podpisan-i/-a študent/-ka Matic Krajšek

- izjavljam, da je zaključno delo rezultat mojega samostojnega dela, ki sem ga izdelal/-a ob pomoči mentor-ja/-ice oz. somentor-ja/-ice;
- izjavljam, da sem pridobil/-a vsa potrebna soglasja za uporabo podatkov in avtorskih del v zaključnem delu in jih v zaključnem delu jasno in ustrezno označil/-a;
- na Univerzo v Mariboru neodplačno, neizključno, prostorsko in časovno neomejeno prenašam pravico shranitve avtorskega dela v elektronski obliki, pravico reproduciranja ter pravico ponuditi zaključno delo javnosti na svetovnem spletu preko DKUM; sem seznanjen/-a, da bodo dela deponirana/objavljena v DKUM dostopna široki javnosti pod pogoji licence Creative Commons BY-NC-ND, kar vključuje tudi avtomatizirano indeksiranje preko spleta in obdelavo besedil za potrebe tekstovnega in podatkovnega rudarjenja in ekstrakcije znanja iz vsebin; uporabnikom se dovoli reproduciranje brez predelave avtorskega dela, distribuiranje, dajanje v najem in priobčitev javnosti samega izvirnega avtorskega dela, in sicer pod pogojem, da navedejo avtorja in da ne gre za komercialno uporabo;
- dovoljujem objavo svojih osebnih podatkov, ki so navedeni v zaključnem delu in tej izjavi, skupaj z objavo zaključnega dela;
- izjavljam, da je tiskana oblika zaključnega dela istovetna elektronski obliki zaključnega dela, ki sem jo oddal/-a za objavo v DKUM.

Uveljavljam permissivnejšo obliko licence Creative Commons: _____ (navedite obliko)

Datum in kraj:
26. 8. 2019

Podpis študent-a/-ke:

