Učenje Morseovog koda s Arduino platformom

Avramović, Nemanja

Undergraduate thesis / Završni rad

2018

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:259370

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2025-01-02**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA

Sveučilišni studij

UČENJE MORSEOVOG KODA S ARDUINO PLATFORMOM

Završni rad

Nemanja Avramović

Osijek, 2018.

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Obrazac Z1P - Obrazac za ocjenu završnog rada na preddiplomskom sveučilišnom studiju

Osijek, 05.09.2018.

Odboru za završne i diplomske ispite

Prijedlog ocjene završnog rada

Ime i prezime studenta:	Nemanja Avramović
Studij, smjer:	Preddiplomski sveučilišni studij Elektrotehnika
Mat. br. studenta, godina upisa:	4029, 22.09.2017.
OIB studenta:	22861514334
Mentor:	Doc.dr.sc. Tomislav Matić
Sumentor:	Dr.sc. Ivan Vidović
Sumentor iz tvrtke:	
Naslov završnog rada:	Učenje Morseovog koda s Arduino platformom
Znanstvena grana rada:	**Arhitektura računalnih sustava (zn. polje računarstvo)**
Predložena ocjena završnog rada:	Izvrstan (5)
Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 3 bod/boda Jasnoća pismenog izražavanja: 3 bod/boda Razina samostalnosti: 3 razina
Datum prijedloga ocjene mentora:	05.09.2018.
Datum potvrde ocjene Odbora:	12.09.2018.

Potpis mentora za predaju konačne verzije rada u Studentsku službu pri završetku studija:	Potpis:
	Datum:

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK**

IZJAVA O ORIGINALNOSTI RADA

Osijek, 16.09.2018.

Ime i prezime studenta:	Nemanja Avramović
Studij:	Preddiplomski sveučilišni studij Elektrotehnika
Mat. br. studenta, godina upisa:	4029, 22.09.2017.
Ephorus podudaranje [%]:	4%

Ovom izjavom izjavljujem da je rad pod nazivom**: Učenje Morseovog koda s Arduino platformom**

izrađen pod vodstvom mentora Doc.dr.sc. Tomislav Matić

i sumentora Dr.sc. Ivan Vidović

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.
Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

SADRŽAJ

1. UVOD 1

1.1. Zadatak završnog rada 1

2. MORSEOV KOD 2

2.1. Što je Morseov kod? 2

2.2. Način rada 3

2.3. Primjena u komunikacijskim sustavima 4

3. ARDUINO SUSTAV 5

3.1. Arduino IDE 5

3.2. Arduino UNO 6

3.3. I^2C LCD modul 7

3.4. Ostale elektroničke komponente 8

4. REALIZACIJA SUSTAVA I TESTIRANJE 9

4.1. Generiranje i prepoznavanje Morseovog koda 9

4.2. Elektronička shema sustava 10

4.3. Shema razvijenog sustava 11

4.4. Programska podrška 12

4.4.1. Dekoder 13

4.4.2. Enkoder 14

4.4.3. Učilica 14

4.4.4. Test 15

4.5. Testiranje i izgled prototipa 16

5. ZAKLJUČAK 18

LITERATURA 19

SAŽETAK 20

ABSTRACT 21

ŽIVOTOPIS 22

PRILOZI 23

1. UVOD

Cilj ovog završnog rada je napraviti Morseovu učilicu na Arduino platformi, koja će omogućiti lagano i brzo učenje Morseove abecede. Ideja se temelji na prepoznavanju i generiranju Morseovih znakova te učenje istih. Sustav će se sastojati od „*Dekodera*“, „*Enkodera*“, „*Učilice*“ i završnog „*Testa*“. „*Dekoder*“ će prepoznavati uneseni Morseov kod te ga pretvarati u alfanumeričke znakove uz popratni zvuk Piezo zujalice, dok će *„Enkoder“* generirati Morseov kod preko Piezo zujalice iz prethodno unesenih slova i brojeva. „*Učilica*“ će služiti za učenje svakog slova i broja pojedinačno te će naizmjenično koristiti funkcije dekodera i enkodera. Nakon završene učilice svatko će moći pristupiti rješavanju „*Testa*“, koji će se sastojati od 10 naizmjenično zadanih alfanumeričkih znakova, koje će osoba, uz prethodno usvojeno znanje, pokušati generirati u Morseov kod. Čitav sustav će biti smješten u drveno kućište.

1.1. Zadatak završnog rada

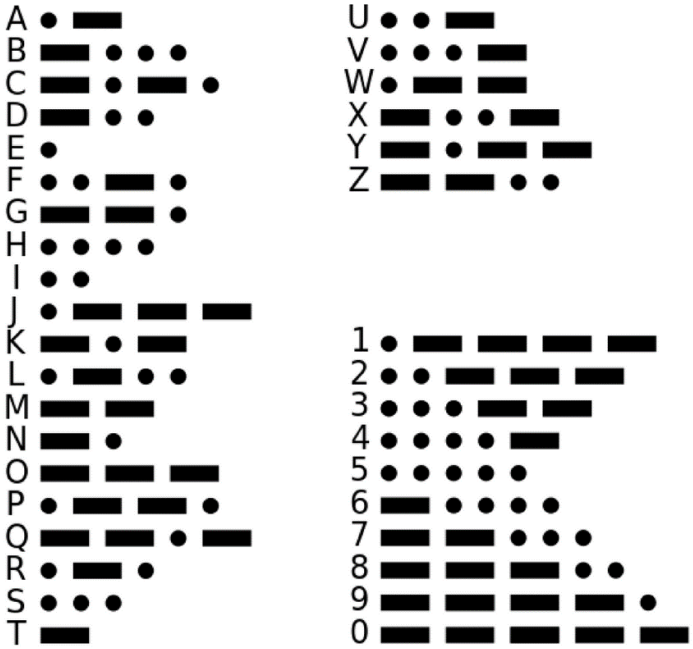
Potrebno je istražiti postupak generiranja i prepoznavanja Morseovog koda. Nakon toga je potrebno razviti sustav zasnovan na Arduino platformi za učenje Morseovog koda te to sve prikazati na LCD ekranu.

2. MORSEOV KOD

U sljedećiim potpoglavljima detaljno će se opisati što je Morseov kod i kako je nastao. Bit će detaljno opisana ideja Morseovog koda te način rada, kao i njegov doprinos i primjena u komunikacijskim sustavima.

2.1. Što je Morseov kod?

Morseov kod predstavlja način kodiranja slova engleske abecede, brojeva, interpunkcijskih znakova i neke službene kratice kao niz dugih i kratkih zvučnih ili svjetlosnih impulsa. Svaki znak Morseove abecede sastoji se od jedinstvenog niza točkica, crtica (eng. *„dots", „dashes"*) i razmaka između istih. Na slici **2.1** prikazani su znakovi engleske abecede i arapskih brojeva.



Sl. 2.1: Morseovi znakovi za englesku abecedu i arapske brojeve.

Morseov kod je dobio naziv prema njegovom izumitelju, Samuel Finley Breese Morseu. Samuel Morse je pridonijeo razvoju telegrafa te je Morseov kod ubrzo postao jedan od najpoznatijih kodova korišten u žičnoj i radio-telegrafiji. Iskusni poznavaoci Morseovog koda mogu direktno, pomoću sluha ili gledajući u nadolazeće svjetlosne impulse, razumjeti primljene kodove. Međutim, postoje i elektromagneti s pisačem, koji prijemne impulse ispisuju na vrpcu u obliku točkica i crtica prema [1].

2.2. Način rada

Svaki Morseov znak sastoji se od jedinstvenog niza točkica i crtica i razmaka između njih. Vrijeme trajanja točkice iznosi jednu vremensku jedinicu dok vrijeme trajanja crtice iznosi tri vremenske jedinice. Trajanje vremenske jedinice, koja se najčešće mjeri u milisekundama, razlikuje se od osobe do osobe ovisno o brzini i sposobnosti odašiljanja Morseovih znakova. Nakon svake odaslane točkice ili crtice slijedi razmak, čije vrijeme je jednako trajanju jedne jedinice. Slova unutar riječi su razmaknuta u trajanju od tri jedinice dok vrijeme razmaka između pojedine riječi iznosi sedam jedinica. Prvi električni telegrafski sustav se temeljio na ostvarivanju žične komunikacije odašiljanjem samo električnog impulsa i razmaka između njih, koji su kontrolirali elektromagnet smješten na prijemnoj strani telegrafskog sustava. Morseov originalni telegrafski prijemnik se temeljio na mehaničkom satu, koji je pomicao papirnatu traku. Primanjem električnih impulsa elektromagnet je spuštao olovku na papirnatu vrpcu, a prekidom strujnog kruga opruga je povlačila olovku u početno stanje prema [2]. Na slici **2.2** prikazan je originalni Morseov koder i dekoder.

Sl. 2.2: Originalni Morseov koder i dekoder.

Brzina prijenosa Morseovog koda mjeri se u broju prenesenih riječi u minuti, wpm (eng. *words per minute*). Vješti poznavaoci Morseovog koda u glavi mogu da dešifriraju i do 50-60 riječi u minuti. U hitnim slučajevima Morseov kod se može poslati improviziranim metodama koje se mogu uključiti i isključiti što ga ujedno čini jednim od najjednostavnijih i najsvestranijih metoda telekomunikacija. Najpoznatiji signal za opasnost je SOS koji je u Morseovom kodu šifriran kao tri točkice, tri crtice i tri točkice („ . . . - - - . . . “) koji je međunarodno priznat ugovorom prema [3].

2.3. Primjena u komunikacijskim sustavima

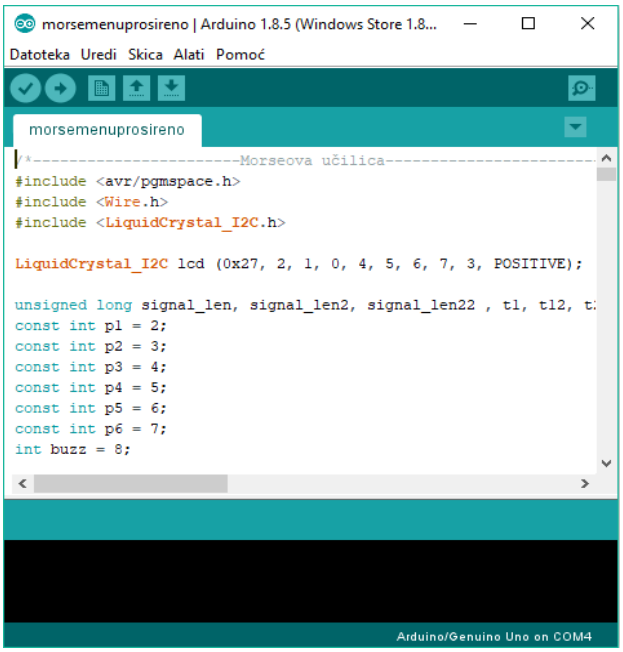
Morseov kod je ispočetka bio namijenjen za civilne svrhe s ciljem povećanja brzine komunikacije na velike udaljenosti na što jednostavniji način, upotrebom impulsa električne struje. S vremenom Morseov kod postaje glavno sredstvo komunikacije za vojne svrhe. Radio-telegrafija Morseovim kodom našla je svoju široku primjenu tijekom Drugog svjetskog rata. Prijenos poruka između ratnih brodova i pomorskih baza zaraćenih strana te ratnih izviđačkih zrakoplova i mornarica bio je od izuzetne važnosti. S obzirom na to da glasovni radio sustavi na brodovima u tadašnje vrijeme nisu bili razvijeni, radio-telegrafija omogućavala je prijenos poruka na velike udaljenosti. U civilnom zrakoplovstvu se također koristi Morseov kod, ali u današnje vrijeme piloti ne moraju biti vrsni poznavaoci Morseovog koda jer prijemne poruke dešifriraju električni dekoderi i prikazuju na LCD ekranima prema [2]. Također u moreplovstvu se pomoću svjetionika navigiraju brodovi, koji se nalaze na udaljenostima od nekoliko desetina kilometara od svjetionika. Danas se Morseov kod primjenjuje i u medicini kod olakšavanja komunikacije ljudi s poteškoćama, kao što su gluhi ili slijepi. Oni mogu primati Morseov kod preko „kožne zujalice“ (eng. *skin buzzera*), uređaja koji se nalazi ispod kože i prenosi impulse pojedinog znaka. Isto tako i kod ljudi koji su doživjeli moždani udar i izgubili sposobnost pokreta i pričanja mogu s treptajima očiju komunicirati s osobama, koje poznaju Morseov kod.

3. ARDUINO SUSTAV

Nadolazeća potpoglavlja će pobliže objasniti rad čitavog sustava. Objasnit će se korištena programska podrška kao i komponente korištene u izradi čitavog sustava te njihova međusobna komunikacija.

3.1. Arduino IDE

Arduino IDE predstavlja sučelje koje služi za programiranje raznih projekata, koji se potom prenose na Arduino uređaj. Uz sinkronizirani rad Arduino programa i Arduino uređaja moguće je napraviti vrlo složene i korisne projekte. Svaki program napisan u Arduino programu naziva se skica (eng. „*sketch*“) te treba sadržavati dvije funkcije, a to su funkcije „*setup()*“ i „*loop()*“ prema [4]. Nakon prenošenja programa na uređaj i njegovog pokretanja program kreće s izvođenjem funkcije „*setup()*“, u kojoj se definiraju svi potrebni ulazi i izlazi, nakon koje ulazi u funkciju „*loop()*“ u kojoj dolazi do izvršavanja cilja projekta. Na slici **3.1** prikazano je sučelje Arduino programa.



Sl. 3.1: Sučelje Arduino programa.

3.2. Arduino UNO

Arduino UNO je programabilna pločica koja se temelji na ATmega328P mikrokontroleru. Mikrokontoler u sebi sadrži na tisuće i tisuće malih tranzistora, koji su odgovorni za odvijanje svih procesa u Arduinu. Tranzistor se smatra jednim od najvažnijih tehnoloških izuma ikada. Nalazi se u svim modernim uređajima prema [5]. Ploča je opremljena sa 14 digitalnih ulazno/izlaznih pinova, 6 analognih ulaznih pinova, 16 MHz kvarcnog kristala, USB konektora, konektora za napajanje, tipke za resetiranje i ICSP konektora. Arduino UNO se može napajati preko USB

Sl. 3.2: Arduino UNO.

konektora ili preko baterije putem konektora za vanjsko napajanje prema [6]. Dimenzije Arduina UNO su 68.6 mm * 53.4 mm. Na slici **3.2** se vidi izgled Arduina UNO.

Arduino UNO ima granični napon u rasponu 6-20 V, ali zbog izbjegavanja mogućih oštećenja preporučeni napon iznosi 7-12 V. Dodatnu zaštitu posjeduje sam Arduino UNO kada se na USB ulaz dovede struja veća od 500 mA. Tada osigurač prekida konekciju sve dok se kratki

Tab. 3.1: Tehničke specifikacije Arduina UNO.

Mikrokontroler	Atmega328P
Operativni napon	5 V
Preporučeni iznos napajanja	7-12 V
Granični iznos napajanja	6-20 V
Digitalni U/I pinovi	14
PWM digitalni U/I pinovi	6
Analogni ulazni pinovi	6
DC struja na U/I pinovima	20 mA
DC struja na 3.3V pinu	50 mA
Flash memorija	32 KB od kojih je 0.5 KB za bootloader
SRAM	2 KB
EEPROM	1 KB
Takt procesora	16 MHz

spoj ne ukloni. U tablici **3.1** prikazane su tehničke specifikacije Arduina UNO.

3.3. I2C LCD modul

I2C LCD modul je zaslon s tekućim kristalom koji je najčešće korišteni zaslon prilikom pravljenja projekata na Arduino platformi. Zaslon uvelike olakšava očitavanje svih potrebnih izlaznih parametara kao što su temperatura, tlak ili rezultati nekih drugih složenih procesa. Ovi zasloni se jednostavno povezuju s Arduinom preko I2C sabirnice. I2C sabirnica je serijska sinkrona sabirnica. Ona omogućuje prenošenje bitova jedan za drugim, što je za razliku od paralelne komunikacije puno sporiji način komuniciranja, ali mnogo jednostavnije izveden. Serijska komunikacija koristi samo nekoliko linija za komunikaciju dok paralelna komunikacija zahtjeva od 8 do 20 linija. I2C komunikacija prisutna je kod mikoupravljača, LCD zaslona, EEPROM i RAM memorije, pojačala s programabilnim pojačanjem itd prema [7]. LCD modul ima integrirani sklop, koji upravlja serijskom komunikacijom. Njegovi ulazno/izlazni podaci su ograničeni na dvije dvosmjerne linije, SDA (serijska podatkovna linija) i SCL (serijska linija sata), koje se spajaju na analogne pinove Arduina. Modul se sastoji od dva dijela, zaslona, koji može prikazivati 16x2 ili 20x4 znakova čije pozadinsko osvjetljenje može biti zeleno s crnim slovima ili plavo s bijelim slovima, i pločice, koja se nalazi na dnu zaslona i upravlja I2C komunikacijom smanjujući sve izlazne na četiri linije SDA, SCL, GNDi VCC prema [8]. Na slici **3.3** prikazan je I2C LCD modul Prije samog programiranja potrebno je pravilno spojiti pinove Arduina i pinove LCD modula i to se može vidjeti u tablici **3.2**.

Tab. 3.2: Pravilno spajanje Arduina i Lcd modula.

Arduino UNO	LCD modul
GND	GND
5V	VCC
A4	SDA
A5	SCL

Sl. 3.3: I2C LCD modul.

3.4. Ostale elektroničke komponente

Od ostalih elektroničkih komponenti korišteni su otpornici, tipkala i Piezo zujalica. Otporik je dvoprilazna, pasivna elektronička komponenta. On se opire protjecanju struje tako da apsorbira dio električne energije te dio energije otpušta u obliku topline. Na taj način ograničava jakost struje kroz sklop. Prema Ohmovom zakonu, otpor predstavlja odnos napona između dvije priključnice i jakosti struje, koja protječe kroz te priključnice. Mjerna jedinica za otpor je Ohm (Ω). U ovom sustavu korišteni su otpornici od 10 kΩ spajani na tipkalo i otpornik od 100 Ω koji je spojen na Piezo zujalicu radi ograničavanja glasnoće Piezo zujalice. Tipkalo je mala elektronička komponenta, koja služi kao sklopka ili prekidač. Svako tipkalo ima četiri nožice, dvije lijeve i dvije desne. Kada se napon dovede npr. na gornju lijevu nožicu tipkala i kada je tipkalo nepritisnuto struja ne prolazi kroz sklopku i napon se nalazi samo na gornjoj i donjoj lijevoj nožici. Čim se tipkalo pritisne ono propušta struju kroz sve četiri nožice. Tipkala su spojena preko *pull-down* otpornika od 10 kΩ radi izbjegavanja pojave neželjenog stanja tipkala prema [9]. Piezo zujalica je mali zvučni uređaj, koji se najčešće koristi u računalima, pisačima, fotokopirnim uređajima, alarmima itd. Zujalica može biti aktivna i pasivna. Aktivna zujalica sadrži unutarnji oscilator, dok pasivna nema. Vidno su jako slične, ali ih možemo razlikovati po donjem dijelu uređaja, koji je kod aktivnog crn, a kod pasivnog se vidi zelena boja tiskane pločice prema [10]. Na slici **3.4** su prikazani otpornik, tipkalo i zujalica.

Sl. 3.4: Otpornik, tipkalo i zujalica.

4. REALIZACIJA SUSTAVA I TESTIRANJE

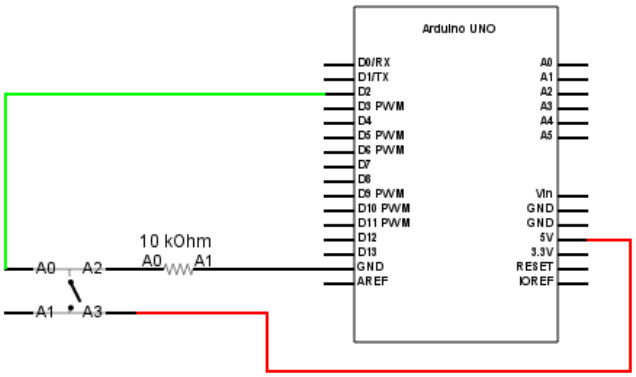
Realizacija Morseove učilice temelji se na izradi prije svega električne sheme na osnovu koje se izrađuje fizički model, koji se sastoji od 10 tipkala, jednog posebnog tipkala za generiranje Morseovog koda, I^2C zaslona, Arduina UNO. Nakon izrade fizičkog modela potrebno je bilo napisati program, koji omogućava sinkroniziran rad čitavog sustava. Morseova učilica zamišljena je da se sastoji od uvodnog dijela, u kojem se osobu ukratko uvodi u Morseov kod, i izbornika koji se sastoji od četiri odvojena dijela. Prvi je *Dekoder* u kojem će se izvoditi dekodiranje Morseovog koda kojeg će osoba kreirati u slova. Drugi je *Enkoder* u kojem će se napisana riječ enkodirati u zvuk Morseovog koda. Treći dio, *Učilica,* je zamišljen da osobu nauči svaki pojedini Morseov znak, dok će u posljednjem dijelu, *Testu,* osoba provjeravati naučeno znanje. Zadnji korak realizacije sustava se temelji na testiranju i izradi kućišta za čitav sustav.

4.1. Generiranje i prepoznavanje Morseovog koda

Generiranje Morseovog koda predstavlja proces enkodiranja znakova u jedinstveni odnos točkica i crtica reprezentiran kraćom i dužom zvučnom signalizacijom. Otipkavanjem Morseovog koda u pravilnim vremenskim razmacima generiramo šifriranu poruku. Generiranje u Morseovoj učilici obavljati će se pritiskanjem posebnog tipkala predviđenog za to. Prepoznavanje Morseovog koda je proces obrnut od generiranja, što znači da prepoznavanje očitava kodiranu šifriranu poruku i dekodira je u slova. Po uzoru na stare enkodere i dekodere Morseovog koda, razvijena je Morseova učilica, koja će u *Dekoderu* u realnom vremenu istovremeno generirati i prepoznavati šifrirani znak, jedan po jedan. Algoritmi koji će to omogućavati detaljnije će biti opisani u poglavlju *Programska podrška*.

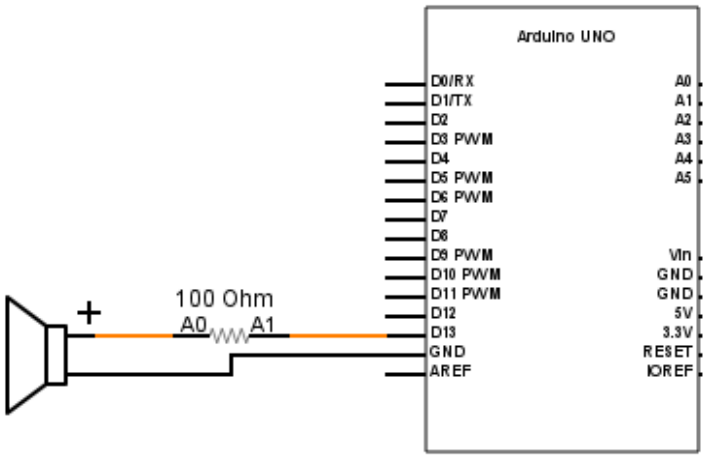
4.2. Elektronička shema sustava

Elektronička shema sustava je izrađena u web aplikaciji *Schematics.* Shema čitavog sustava nije zahtjevna te je izrada iste bila jednostavna. Prvo je izrađena shema spajanja Arduina i tipkala. Ukupno je bilo potrebno spojiti 11 tipkala na pinove Arduina. S obzirom na to da svako tipkalo ima grupirane dvije lijeve i dvije desne nožice, gornja lijeva nožica svakog tipkala je spojena na uzemljene, odnosno GND pin Arduina dok su svaka donja lijeva nožica pojedinog tipkala spojene na po jedan digitalni pin Arduina. Gornja desna nožica pojedinog tipkala spojena je na pin Arduina, koji daje napon od 5 V. Zbog održavanja trenutnih stanja tipkala te izbjegavanja pojavljivanja grešaka prilikom rada sustava, svaka lijeva gornja nožica pojedinog tipkala je na uzemljenje spojena preko otpornika od 10 kΩ, koji se naziva *pull-down* otpornik. Na slici **4.1** se je prikazana shema spajanja Arduina i tipkala.



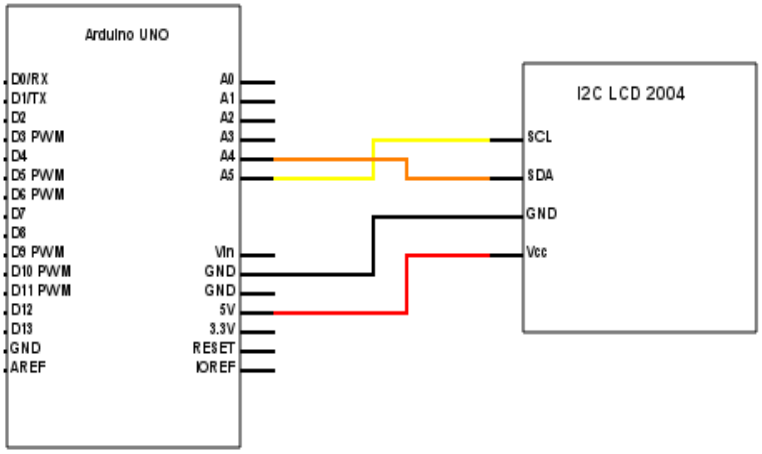
Sl. 4.1: Shema spajanja Arduina i tipkala.

Iduća shema, koja se izrađivala, je bila shema spajanja Piezo zujalice, odnosno malog signalizirajućeg zvučnog uređaja, i Arduina. Zujalica je spojena na posljednji digitalni pin preko otpornika od 100 Ω, kao prema slici **4.2**, kako bi se smanjila glasnoća zujalice jer je jako glasna.



Sl. 4.2: Shema spajanja Arduina i Piezo zujalice.

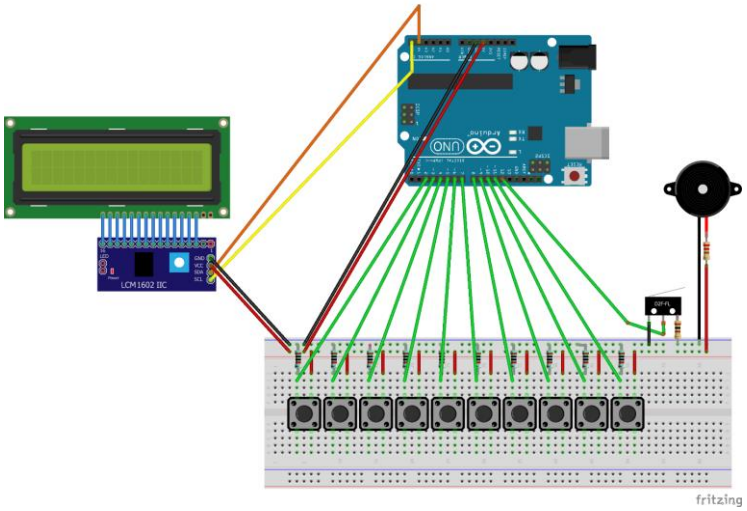
Posljednja shema koja je izrađivana je shema spajanja Arduina i I^2C zaslona. Zahvaljujući I^2C serijskoj komunikaciji, spajanje zaslona i Arduina je svedeno zapravo samo na dvije linije, ne računajući linije za napajanje i uzemljenje. SDA pin LCD modula, koji je odgovoran za prijenos podataka, je spojen na analogni pin Arduina, A4. SCL pin LCD modula, koji je zaslužan za prijenos iznformacije o vremenu, je spojen na Arduinov analogni pin A5, kao što se vidi na slici **4.3**.



Sl. 4.3: Shema spajanja Arduina i LCD modula.

4.3. Shema razvijenog sustava

Shema razvijenog sustava napravljena je u programu pod nazivom *Fritzing*, koji je besplatan za preuzeti. Pomoću njega se mogu lako, brzo i slikovito napraviti sheme funkcionalnih sustava. Na slici **4.4** je prikazana shema razvijenog sustava.



Sl. 4.4: Shema razvijenog sustava.

4.4. Programska podrška

Glavni dio čitavog sustava je programski dio, koji se sastoji od niza funkcija i algoritama zaslužnih za generiranje i prepoznavanje Morseovih znakova. Na početku programa definirane su biblioteke, konstante i varijable. Biblioteka „Wire.h“ korištena je da omogući komunikaciju Arduina s I^2C sklopovima. „LiquidCrystal_I2C.h“ biblioteka je korištena da omogući funkcionalnost I^2C zaslona, dok je biblioteka „avr/pgmspace.h“ korištena radi omogućavanja funkcije *F() makro*, koja će detaljnije biti opisana u nastavku. U funkciji *setup()* nalazi se naredba za aktiviranje LCD zaslona, definiranje ulaznih i izlaznih pinova nakon čega program ulazi u *loop()* funkciju, od koje sve počinje. Nakon uključivanja Morseove učilice, u uvodnom dijelu se prikazuje nekoliko rečenica, koje osobu uvode u način rada Morseovog koda, tj. odnos vremenskog trajanja točkice i crtice te trajanje razmaka između točkice i crtice, pojedinih slova i riječi. Nakon uvodnog dijela od osobe se zahtijeva da definira svoju točkicu i crticu, tako što će pritisnuti tipkalo za generiranje Morseovog koda. Očitana točkica će imati toleranciju kasnijeg prepoznavanja od -50% očitanog vremena do +20% očitanog vremena, dok će crtica imati toleranciju od -20% očitanog vremena do +50% očitanog vremena. Nakon toga će se pristupiti izborniku, u kojem će se moći izabrati *Dekoder, Enkoder, Učilica* ili *Test*, kao što se vidi na slici **4.5**.

Sl. 4.5: Izbornik Morseove učilice.

S obzirom da čitav program sadrži preko 800 linija koda te uz to niz funkcija *lcd.print()* kojima se prikazuje velika količina teksta na zaslonu, došlo je do problema s memorijom i prekidanja rada čitavog sustava jer Arduino UNO ima 2 KB statičke radne memorije, a veličina programa je iznosila nešto manje od 2000 B. Problem je riješen pomoću funkcije *F() makro* koja omogućava da se sav željeni tekst, koji se prikazuje na zaslonu, sprema u flash memoriju, koja kod Arduina UNO iznosi 32 KB prema [11]. Veličina čitavog programa je pomoću spomenute funkcije smanjena na polovinu prvobitne vrijednosti, čime je omogućena funkcionalnost sustava.

4.4.1. Dekoder

U *Dekoderu* će biti prikazano prebacivanje Morseovih znakova u slova i brojeve uz popratnu zvučnu signalizaciju. Moći će se vidjeti kako se direktno iz generiranog Morseovog znaka na LCD zaslonu očitavaju slova i brojevi jedan po jedan odmah nakon što osoba unese željeni Morseov znak. Slovo ili broj će biti prepoznati samo u slučaju ako su vremenski odnos točkice, crtice i razmaka u skladu s pravilima Morseovog koda. Na slici **4.6** se vidi zaslon *Dekodera*.

Sl. 4.6: Zaslon *Dekodera*.

Problem prepoznavanja je riješen tako što je na samom početku definiran jedan statični string *letters[]* u kojeg su spremljeni znakovi engleske abecede i brojeva jedan za drugim, ali u obliku Morseovih znakova. Nakon toga je bilo potrebno precizno mjeriti pritisnuto stanje Morseovog tipkala. S obzirom na to da je frekvencija oscilatora 16 MHz precizno mjerenje je bilo moguće, za kojeg je korištena funkcija *millis()*, koja mjeri vrijeme u milisekundama počevši s pokretanjem Arduina. U trenutku pritiska Morseovog tipkala trenutno vrijeme se pripiše jednoj vremenskoj varijabli te čim se tipkalo otpusti novo vrijeme se pripiše drugoj vremenskoj varijabli. Rezultat razlike te dvije vremenske varijable jeste vrijeme, koje se analizira. Ako je vremenska duljina pritiska veća od donje granice tolerancije za točkicu i manja od gornje granice tolerancije za točkicu u novi pripisani string *morse[]* bit će dodana točkica. U skladu s tim, ako je vrijeme veće od gornje granice točkice i manje od donje granice crtice biti će prepoznata i dodana crtica. Donja granica tolerancije za crticu je u ovom slučaju stavljena kao gornja granica točkice s namjerom da se izbjegne neočitavanje ni jednog znaka jer ako su gornja granica točkice i donja granica crtice različite često dolazi do neočitavanja crtice jer se vrijeme mjeri u milisekundama i teško je, gotovo nemoguće, da osoba uspije dva puta zaredom da generira crticu u vremenskim intervalima od – 20% do + 50% očitanog vremena, kao što je generirala u uvodu Morseove učilice. Zato je očitavanje crtice namješteno da kreće kada očitavanje točkice prestaje. Sve dok je vrijeme između generiranja točkice ili crtice manje od 3 dužine točkice, generirana točkica ili crtica se sprema u string *morse[]*. Ako je vrijeme čekanja između generiranja točkice i crtice veće od 3 dužine točkice, što odgovara razmaku između slova, string *morse[]* odlazi u funkciju *converter()*, koja

generirani Morseov znak pretvara u slovo ili broj nakon čega briše string *morse[]* te ga priprema za čuvanje novog Morseovog znaka. U dekoderu će se još moći koristiti tri tipke, jedna za pravljenje razmaka, druga za brisanje pojedinog slova i treća za brisanje čitavih riječi i postavljanje kursora na početak.

4.4.2. Enkoder

Nakon pristupa *Enkoderu* u prvom redu se može vidjeti kursor, koji služi za pokazivanje trenutnog mjesta za željeno slovo, a u zadnja dva reda mogu se vidjeti slova, brojevi, razmak i strelica, koja služi za brisanje. To se može vidjeti na slici **4.7**. *Enkoder* će prikazati obrnut postupak od *Dekodera*, što znači šifriranje poruke i prebacivanje u zvučni Morseov kod pritiskanjem posebnog gumba za generiranje čitave poruke. Kretanje po slovima bit će moguće pomoću pet tipkala za gore, dole, lijevo, desno te tipke za biranje željenog slova. Problem enkodiranja je riješen tako što je svakom slovu i broju pripisan njegov odgovarajući Morseov znak u string *morse*. Točkice i crtice pojedinog znaka su dodavane tako da je nakon svakog izabranog slova ili broja u string *morse* dodan pripadajući Morseov znak te na kraju znak „ | “ koji je označavao prekid između pojedinog Morseovog slova. Na primjer, riječ SOS u stringu *morse* će biti sačuvana kao „

Sl. 4.7: Zaslon *Enkodera*.

. . . | - - - | . . . | “. Svaka uspravna crta između slova označava dužinu razmaka u trajanju dužine tri točkice. Ako se u stringu nalazi razmak između riječi, on će, prema pravilima za Morseov kod, trajati sedam dužina točkice. Svaka točkica i crtica će biti generirana u Morseov kod po uzoru na generiranu točkicu i crticu svake pojedine osobe u uvodnom dijelu Morseove učilice, što znači da će se zvučno generirane riječi razlikovati od osobe do osobe.

4.4.3. Učilica

Učilica služi za učenje pojedinog znaka Morseove abecede. Radi na principu pokušaja i pogreški. Svako pojedino slovo i broj će biti prikazano na zaslonu te će uz slovo biti prikazan njegov ekvivalentni Morseov znak, uz koji će biti prisutna i zvučna signalizacija, koja će uvelike olakšati učenje Morseovog koda. Ako osoba pogrešno generira traženi znak, od nje se traži da

pokuša ponovo sve dok ne uspije pravilno generirati. Tek nakon ispravnog generiranja znaka, prelazi se na iduće slovo ili broj. Učilica koristi naizmjenično algoritme *Enkodera* i *Dekodera*. Prvo koristi funkcije *Enkodera* za enkodiranje pojedinog slova u zvučni oblik Morseovog koda, kako bi se osobi što bolje predstavio pojedini znak u Morseovom kodu. Nakon toga osoba generira traženi znak, a zatim *Učilica* koristi funkcije *Dekodera*, kojima provjerava da li je osoba ispravno generirala Morseov znak. Svako slovo će biti generirano, kao i u enkoderu i dekoderu, na osnovu prethodno generiranih točkica i crtica. Na slici **4.8** prikazan je zaslon *Učilice*.

Sl. 4.8: Zaslon *Učilice*.

4.4.4. Test

Test je zadnji dio Morseove učilice, u kojemu osoba može provjeriti naučeno znanje. Sastoji se od zadavanja deset naizmjeničnih alfanumeričkih znakova te provjere istih. Za izbacivanje naizmjeničnih brojeva, pomoću kojih se bira slovo ili broj za provjeru, nije bilo dovoljno koristiti samo funkciju *random (min,max)*, zato što bi se svaki put nakon ulaska u *Test* pojavljivali isti naizmjenično odabrani alfanumerički znakovi. Problem je riješen upotrebom funkcije *randomSeed (analogRead (A0))* koju je potrebno napisati u *setup()* funkciji prema [12]. *Test* koristi algoritme *Dekodera* za provjeru točnosti generiranog Morseovog znaka. Točnost ovisi o tome da li osoba generira prema pravilima za način rada Morseovog koda te da li je trajanje generirane točkice i crtice u skladu s generiranim točkicama i crticama u uvodnom dijelu. Sa svakim točnim generiranjem osoba osvoji po jedan bod. Na kraju *Testa* se, prema osvojenom broju bodova,

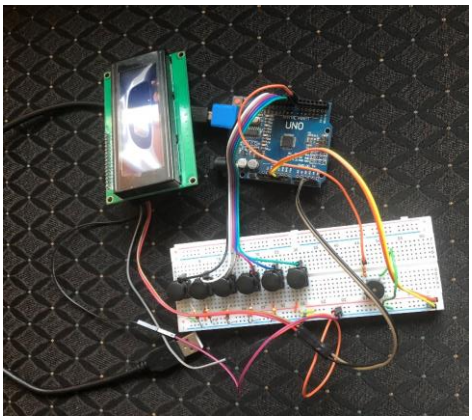
Sl. 4.9: Zaslon Testa.

prikazuje jedna od pet poruka, u kojima se čestita na uspješnosti ili savjetuje osobu da ponovo pristupi učilici. Na slici **4.9** prikazan je zaslon *Testa*.

4.5. Testiranje i izgled prototipa

Prvo testiranje je sprovedeno u *Dekoderu* za svako pojedinačno slovo i broj. Nakon pisanja koda za dekodiranje Morseovih znakova u alfanumeričke znakove, svaki Morseov znak, pojedinačno, je generiran na temelju dostupne Morseove abecede te dekodiran u alfanumerički znak. Nakon uspješnosti tog testa, sprovedeno je testiranje *Enkodera*, u kojem je bilo potrebno unesenu riječ, pritiskom na tipku za generiranje, kodirati u Morseovu poruku reprezentiranu zvukom Piezo zujalice. Testiranje je provedeno s više riječi i svaki je test rezultirao uspješnim prepoznavanjem. Kombiniranjem algoritama *Dekodera* i *Enkodera*, u *Učilici* i *Testu* testiranje je sprovedeno na način da odmah nakon generiranog traženog slova u *Učilici* ili *Testu* prikazuje se poruka o uspješnosti generiranja, čime je testiranje privedeno kraju.

Prototip sustava je izrađen kombinacijom vodiča i eksperimentalne pločice, kao što se može vidjeti na slici **4.10**. Početna verzija prototipa je funkcionirala, ali se dešavalo da se ponekad izgubi kontakt prilikom pomjeranja i prenošenja eksperimemntalne pločice s jednog mjesta na drugo. Dešavao se gubitak komunikacije Arduina s LCD modulom ili pojedinog tipkala s Arduinom. Zbog toga je odlučeno da prototip bude smješten u svoje kućište i time ujedno poprimi ljepši izgled.



Sl. 4.10: Izgled početne verzije prototipa.

Za izradu kućišta uzeta je drvena kutija, koja se može otvarati s gorje strane. Na vratima s gornje strane izbušene su po četiri rupice za svako tipkalo, dvije rupice za mali zvučnik te šira rupa za I^2C komunikaciju između LCD modula i Arduina. Prvih pet tipkala s lijeve strane bit će korišteni za kretanje i odabir željene stavke u izborniku ili slova u *Enkoderu*. Iduće tri tipke bit će korištene u *Dekoderu* i redom s lijeva na desno će označavati tipku za brisanje pojedinog znaka, odvajanje i brisanje cijelog zaslona. Predzadnja tipka će se koristiti u *Enkoderu* i služit će za

enkodiranje željene riječi, dok će zadnja tipka s desne strane označavati tipku za povratak iz pojedine stavke izbornika. Donja crvena tipka bit će korištena u *Dekoderu*, *Učilici* i *Testu* za dekodiranje pojedinih alfanumeričkih znakova. Donje dvije rupice LCD modula su zašarafljene na kutiju tako da LCD zaslon stoji odmah iznad tipkala. Sa stražnje strane kućišta izbušene su dvije rupice, koje služe za napajanje Arduina. Kroz lijevu rupicu prolazi USB kabal, kojim se Arduino može spojiti na laptop ili računar te dodatno programirati, dok kroz desnu rupicu prolazi adapter, pomoću kojeg se Arduino može uključiti na 220 V utičnicu te napajati bez potrebe za laptopom. Na slici **4.11**. prikazana je konačna verzija prototipa.

Sl. 4.11: Konačna verzija prototipa.

5. ZAKLJUČAK

U ovom završnom radu napravljena je Morseova učilica na Arduino platformi. Prije svega je bilo potrebno točno proučiti način rada Morseovog koda, nakon čega je razvijena ideja o funkcioniranju čitavog sustava. Nakon razvijanja ideje, izrađena je elektronička shema te su nabavljene komponente. Spajanjem svih komponenti u jednu funkcionalnu cjelinu završio je prvi dio izrade sustava. Drugi dio izrade sustava temeljio se na programiranju. Kombinacijom niza funkcija i algoritama postignuto je da program uspješno izvršava sve tražene zadatke. Zbog izbjegavanja gubljenja kontakta te neočekivanih prekida u radu sustava, praktičnije upotrebe i ljepšeg izgleda čitav sustav je smješten u drvenu kutiju, koja je praktična za prijenos i upotrebu. Cilj Morseove učilice je da pomogne drugima da nauče Morseov kod na jednostavan i zanimljiv način te približi sliku o tome na koji način su ljudi komunicirali, u ne tako dalekoj prošlosti.

LITERATURA

[1] Samuel Morse: Činjenice koje niste znali o slavnom izumitelju, http://studentski.hr/vijesti/na-danasnji-dan/samuel-morse-cinjenice-koje-niste-znali-o-slavnom-izumitelju, kolovoz 2018.

[2]Morseov kod, https://en.wikipedia.org/wiki/Morse_code, kolovoz 2018.

[3] 1908: SOS prihvaćen kao međunarodni signal za opasnost, http://history.info/on-this-day/1908-sos-adopted-as-international-distress-signal, kolovoz 2018.

[4] Arduino razvojno okruženje, https://www.arduino.cc/en/guide/environment, kolovoz 2018.

[5] T. Olsson, Arduino Wearables, Technology in Action, 2012.

[6] Arduino UNO Rev3, https://store.arduino.cc/usa/arduino-uno-rev3, kolovoz 2018.

[7] Davor Bonači, I^2C sabirnica, Seminarski rad, http://spvp.zesoi.fer.hr/seminari/2007/seminari/DavorBonaci_I2Csabirnica.pdf, kolovoz 2018.

[8] LCD 16x2 – Korištenje LCD zaslona na bazi tekućeg kristala putem I^2C komunikacije, https://www.meccanismocomplesso.org/en/lcd1602-utilizzare-un-display-a-cristalli-liquidi-lcd-con-arduino-tramite-i2c/, kolovoz 2018.

[9] Pull-up / Pull-down Otpornik, https://e-radionica.com/hr/blog/2015/08/19/pull-up-pull-down-otpornik/, kolovoz 2018.

[10] Aktivna zujaloca, http://www.merlin10.com/blog/2016/05/03/aktivna-zujalica/, kolovoz 2018.

[11] Arduino memorija, https://playground.arduino.cc/Learning/Memory, kolovoz 2018.

[12] Funkcija randomSeed(), https://www.arduino.cc/reference/en/language/functions/random-numbers/randomseed/, kolovoz 2018.

SAŽETAK

Naslov: Učenje Morseovog koda s Arduino platformom

U ovom završnom radu napravljena je Morseova učilica, koja na jednostavan i zanimljiv način može naučiti dekodiranje pojedinog Morseovog znaka. Temeljena je na Arduino platformi. Putem digitalnih pinova omogućena je komunikacija s Arduinom preko tipkala dok je komunikacija između Arduina i LCD modula uspostavljena I^2C serijskom komunikacijom. Glavni dio čitavog sustava je programski dio, koji je zaslužan za neometan i sinkronizirani rad čitavog sustava. Morseova učilica je smještena u drveno kućište radi sigurnijeg rada, estetskog izgleda i praktičnosti upotrebe.

Ključne riječi: Arduino, Morseov znak, LCD modul, I^2C serijska komunikacija, programski dio

ABSTRACT

Title: Learning Morse code with Arduino platform

In this final paper was made Morse's learning device, which in a simple and interesting way can learn to decode an each Morse sign. It is based on Arduino platform. Over digital pins, it is possible to communicate with Arduino via push-buttons while communication between Arduino and LCD module is established by I^2C serial communication. The main part of the whole system is a program that is responsible for undisturbed and synchronized operation of the whole system. Morse's learning device is situated in a wooden case for safer work, aesthetic appearance and practicality.

Keywords: Arduino, Morse sign, LCD module, I^2C serial communication, program

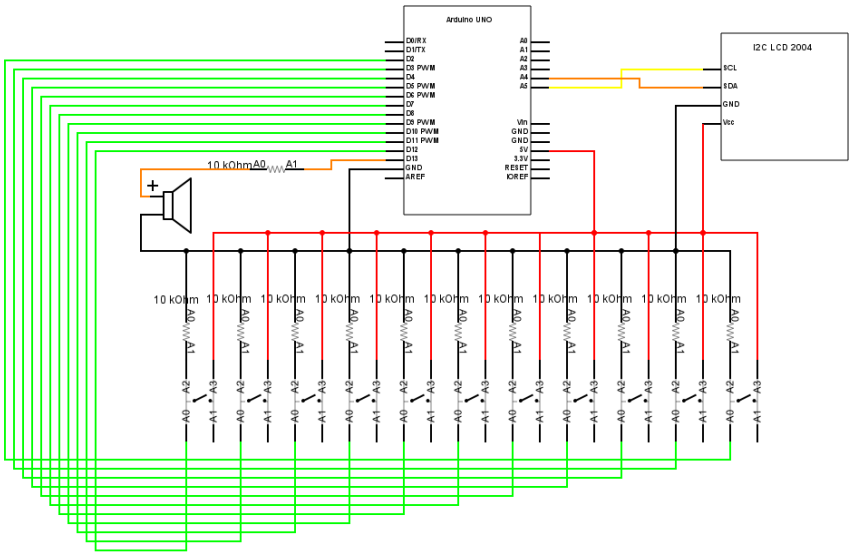
ŽIVOTOPIS

Nemanja Avramović rođen je u Vukovaru 06.11.1996. Do četvrte godine života živi u Vukovaru nakon čega se seli u Borovo. Pohađao je Osnovnu školu „Borovo“ od prvog do osmog razreda. Osnovu školu završio je s odličnim uspjehom 2011 godine. Nakon završetka osnovne škole, iste godine upisuje Gimnaziju Vukovar, smjer Opća gimnazija. Gimnaziju završava s odličnim uspjehom 2015. godine nakon čega iste godine upisuje Elektrotehnički fakultet Osijek na preddiplomskom studiju elektrotehnike. Nakon završetka prve godine preddiplomskog studija elektrotehnike, na drugoj godini izabire smjer Komunikacije i Informatika.

PRILOZI

CD s programskim kodom i shemama sustava

P. **1**



P. 1: Elektronička shema čitavog sustava.