

# Identifikacija i pristup primjenom RFID tehnologije

---

**Katić, Tamara**

**Master's thesis / Diplomski rad**

**2018**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Split, University of Split, Faculty of science / Sveučilište u Splitu, Prirodoslovno-matematički fakultet**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:166:298129>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2025-02-18**

*Repository / Repozitorij:*

[Repository of Faculty of Science](#)



UNIVERSITY OF SPLIT



**SVEUČILIŠTE U SPLITU**  
**PRIRODOSLOVNO-MATEMATIČKI FAKULTET**

**TAMARA KATIĆ**

**IDENTIFIKACIJA I PRISTUP PRIMJENOM**  
**RFID TEHNOLOGIJE**

**DIPLOMSKI RAD**

**SPLIT, 2018**

**SVEUČILIŠTE U SPLITU**  
**PRIRODOSLOVNO-MATEMATIČKI FAKULTET**

**STUDIJ: INFORMATIKA I TEHNIKA**

**IDENTIFIKACIJA I PRISTUP PRIMJENOM**  
**RFID TEHNOLOGIJE**

**DIPLOMSKI RAD**

**MENTOR:**

**Dr.sc. Vladimir Pleština, docent**

**STUDENT:**

**Tamara Katić**

**SPLIT, 2018**

# Temeljna dokumentacijska kartica

Diplomski rad

Sveučilište u Splitu

Prirodoslovno-matematički fakultet

Odjel za politehniku

Ruđera Boškovića 33, 21000 Split, Hrvatska

## IDENTIFIKACIJA I PRISTUP PRIMJENOM RFID TEHNOLOGIJE

Tamara Katić

### SAŽETAK

Prilikom izrade ovog diplomskog rada proučavala sam probleme odlaganja otpada u suvremenom svijetu te izračun komunalne naknade za građane. Zajedno sa kolegicom Antonelom Barnjak izradila sam prototip pametnog sustava za naplatu usluga sakupljanja i odvoza otpada koji svakom korisniku u realnom vremenu omogućuje uvid u njegovu potrošnju. Problem pravednijeg izračuna naknade za odvoz otpada riješile smo na način da se svaki korisnik prijavljuje putem RFID sustava za identifikaciju prilikom odlaganja otpada te mu se pri tome izračunava težina otpada koju ubacuje u kantu i na tome temelji njegov mjesečni račun. Projekt se temelji na korištenju Arduino Nano pločice koja je lako dostupna i jednostavna za rad pa se njom mogu koristiti i potpuni početnici, a za identifikaciju korisnika koristila sam RFID – RC522 čitač sa pripadajućim karticama za kontrolu pristupa. Danas se javlja sve veća potreba za ovakvim sustavima sa što višom razinom automatizacije procesa rada. Osmislile smo cjenovno prihvatljiv autonoman sustav koji zadovoljava današnje potrebe korisnika za pravednijim načinom naplate odvoza otpada te koji ima razne mogućnosti za nadogradnju i poboljšanja u budućnosti.

**Ključne riječi:** RFID tehnologija, Arduino, LCD ekran, Solenoid brava, konstrukcija, program

Rad je pohranjen u knjižnici Prirodoslovno-matematičkog fakulteta, Sveučilišta u Splitu

**Rad sadrži:** 54 stranice, 18 slika, 3 tablice i 26 literaturnih navoda.

Izvornik je na hrvatskom jeziku.

**Mentor:** **Dr. sc. Vladimir Pleština**, docent Prirodoslovno-matematičkog fakulteta,  
Sveučilišta u Splitu

**Neposredni voditelj:**

**Hrvoje Turić**, *predavač Prirodoslovno-matematičkog fakulteta, Sveučilišta u Splitu*

**Ocjenjivači:** **Dr. sc. Vladimir Pleština**, *docent Prirodoslovno-matematičkog fakulteta, Sveučilišta u Splitu*

**Hrvoje Turić**, *predavač Prirodoslovno-matematičkog fakulteta, Sveučilišta u Splitu*

**Dr. sc. Tomislav Matić**, *izvanredni profesor Prirodoslovno-matematičkog fakulteta, Sveučilišta u Splitu*

Rad prihvaćen: **u rujnu, 2018. godine**

## **Basic documentation card**

Graduate thesis

University of Split

Faculty of Science

Department of polytechnics

Ruđera Boškovića 33, 21000 Split, Croatia

### **IDENTIFICATION AND ACCESS USING RFID TECHNOLOGY**

Tamara Katić

#### **ABSTRACT**

Throughout making of this thesis I have been studying the problems of waste disposal in the modern world and calculation of communal fee for citizens. Together with my colleague, Antonela Barnjak, I have developed a prototype of a billing system for collecting and disposal of waste that enables each user to have real-time insight into their consumption. The problem of a fairer calculation of the waste disposal fee was solved by each user logging in with RFID identification system when disposing of the waste. The monthly bill is based on the calculation of the weight of the waste that is placed into the bin. The project is based on the use of the Arduino Nano microcontroller, which is easily accessible and simple to operate, so that it can be used by complete beginners, and RFID - RC522 reader with corresponding cards for access control and user identification. Today there is an increasing need for such systems with the highest level of automation of work processes. We have designed an affordable autonomous system that meets today's needs for a fairer way of charging waste disposal and has many possibilities for upgrades and improvements in the future.

**Key words:** RFID technology, Arduino, LCD display, Solenoid lock, construction, program

Thesis deposited in library of Faculty of Science, University of Split

**Thesis consists of:**

54 pages, 18 pictures, 3 tables and 26 references.

Original language: Croatian

**Mentor:** **Vladimir Pleština, Ph.D.** *Assistant professor of Faculty of Science, University of Split*

**Supervisor:** **Hrvoje Turić**, *Lecturer of Faculty of Science, University of Split*

**Reviewers:** **Vladimir Pleština, Ph.D.** *Assistant professor of Faculty of Science, University of Split*

**Hrvoje Turić**, *Lecturer of Faculty of Science, University of Split*

**Tomislav Matić, Ph.D.** *Associate Professor of Faculty of Science, University of Split*

Graduate thesis accepted: **in September of 2018.**

# Sadržaj

<b>1. UVOD.....</b>	<b>1</b>
<b>2. ARDUINO .....</b>	<b>3</b>
2.1. ARDUINO NANO .....	5
<b>3. RFID TEHNOLOGIJA .....</b>	<b>9</b>
3.1. RFID SUSTAV .....	10
3.2. RFID TRANSPONDERI.....	11
3.2.1. <i>Podjela transpondera prema načinu napajanja.....</i>	<i>14</i>
3.2.2. <i>Podjela transpondera prema načinu fizičke izvedbe.....</i>	<i>16</i>
3.2.3. <i>EPCglobal podjela transpondera .....</i>	<i>17</i>
3.3. PRIMJENA RFID TEHNOLOGIJE .....	19
<b>4. IZRADA KONSTRUKCIJE.....</b>	<b>21</b>
<b>5. SPAJANJE KOMPONENTI SA ARDUINO PLOČICOM.....</b>	<b>23</b>
5.1. SPAJANJE LIQUIDCRYSTAL EKRANA SA IIC ADAPTEROM I ARDUINO PLOČICE.....	23
5.2. SPAJANJE RFID ČITAČA SA KARTICAMA ZA KONTROLU PRISTUPA I ARDUINO PLOČICE....	25
5.3. SPAJANJE SOLENOID BRAVE I ARDUINO PLOČICE.....	27
5.4. SPAJANJE ARDUINO KOMPATIBILNOG MOS DRIVING MODULA I ARDUINO PLOČICE .....	29
<b>6. PROGRAMSKA PODRŠKA.....</b>	<b>31</b>
<b>7. ZAKLJUČAK .....</b>	<b>36</b>
<b>LITERATURA.....</b>	<b>38</b>
<b>POPIS SLIKA I TABLICA .....</b>	<b>40</b>
<b>PRILOG 1 - SHEMA SPAJANJA SVIH KORIŠTENIH KOMPONENTI SA ARDUINOM NANO .....</b>	<b>41</b>
<b>PRILOG 2 – PROGRAMSKI KOD .....</b>	<b>42</b>



# 1. Uvod

U suvremenom svijetu često se javlja problem odlaganja otpada. Uz različite pokušaje povećanja svijesti građana o potrebi za recikliranjem, problem se javlja i prilikom naplate komunalnih usluga za skupljanje i odvoz komunalnog otpada. Na području gradova Splita, Kaštela i Solina te općina Podstrana, Dugopolje i Klis se za održavanje čistoće i odlaganje komunalnog otpada brine Trgovačko društvo Čistoća d.o.o. Ukupni broj stanovnika u gore navedenim jedinicama lokalne samouprave iznosi cca. 300 000 stanovnika. Svi stanovnici koji su vlasnici ili korisnici zgrada, stanova, poslovnih ili drugih prostorija, najmoprimci, zakupci i drugi korisnici stambenih i poslovnih prostora, kao i proizvođači otpada na navedenim područjima, dužni su koristiti usluge sakupljanja, odvoza i odlaganja komunalnog otpada. Obračunska osnovica za plaćanje usluga zbrinjavanja komunalnog otpada iz kućanstva na području grada Splita je volumen otpada. Za stambene prostore cijena usluge iznosi 0.194 kn po litri otpada, dok su za poslovne prostore, ovisno o kategorizaciji njihove djelatnosti, cijene usluge u rasponu od 0.194 kn po litri otpada do 1.940 kn po litri otpada. S obzirom na to da građani nemaju vlastite kante već zajedničke kontejnere, naknada za komunalni otpad obračunava se po potrebnom broju odvoza po kontejneru, neovisno o pojedinačnoj potrošnji svakog građanina.

Ideja za temu ovog diplomskog rada nastala je prilikom proučavanja samog sustava naplate za usluge odvoza komunalnog otpada. Kolegica Antonela Barnjak i ja smo pokušale osmisliti pravedniji način naplate usluge u kojem bi se mjesečna naknada korisnicima određivala prema stvarnoj težini korisnikovog stvorenog otpada, umjesto fiksnog iznosa naknade kao što je to slučaj do sada. Razvijajući našu ideju, odlučile smo se u sklopu diplomskog rada osmisliti i izraditi prototip pametnog sustava za naplatu usluga sakupljanja i odvoza otpada koji bi svakom korisniku u realnom vremenu mogao omogućiti uvid u njegovu potrošnju. Kolegica Barnjak se bazirala na osiguravanju toga da sustav funkcionira i bez pristupa električnoj energiji, uz pomoć solarnog sustava, te na samom izračunu korisnikove potrošnje prema težini njegovog otpada. Jedan od ključnih elemenata autonomne inteligentne kante za smeće koju smo izradile je i identifikacija samog korisnika. U ovom radu pobliže ću opisati sustav za identifikaciju i pristup koristeći tehnologiju radiofrekvencijske identifikacije (RFID tehnologija, engl. *radio frequency*

*identification*) kako bi se za pojedinog korisnika mogla definirati količina otpada koju je on odložio.

RFID je naziv za tehnologiju koja koristi radio valove kako bi automatski identificirali objekte. Velika raznolikost RFID sustava omogućuje izrazito veliki broj primjena, koji s vremenom i tehnološkim napretkom sve brže raste. Danas je moguće primijeniti RFID tehnologiju u cjelokupnoj čovjekovoj okolini, od odjeće, preko automobila i vlakova do kućnih ljubimaca, pa čak i u same ljude, što obećava brojne pogodnosti i nove, do sada neslućene, mogućnosti lagodnijeg i efikasnijeg obavljanja svih svakodnevnih poslova. Fleksibilnost i sveprisutnost RFID sustava su neki od glavnih razloga zbog kojih sam se odlučila koristiti upravo tu tehnologiju prilikom identifikacije korisnika sustava naplate za usluge odvoza komunalnog otpada. U nastavku rada detaljnije ću opisati funkcioniranje RFID tehnologije i način njene primjene prilikom izrade ovog sustava. Uz to, pobliže ću definirati i sve ostale komponente koje sam koristila, njihovo povezivanje u jednu cjelinu, kao i programsku podršku koja omogućuje njihov zajednički rad.

## 2. Arduino

Arduino je platforma otvorenog koda (eng. *open source*) za kreiranje elektroničkih prototipova bazirana na sklopovlju i programskom paketu koji je fleksibilan i jednostavan za korištenje [1]. Sastoji se od skupa elektroničkih komponenti (eng. *hardware*) i programske podrške (eng. *software*) koji se mogu jednostavno povezivati u složenije cjeline. Nastao je u Italiji 2003. godine kada je grupa studenata Instituta za interaktivni dizajn u Ivreji osmislila cjenovno pristupačan i jednostavan način za sve početnike i stručnjake da uređajem komuniciraju sa okolinom uz pomoć raznih senzora i aktuatora. Tijekom godina Arduino je postao „mozak“ velikog broja projekata, od svakodnevnih objekata do kompleksnih znanstvenih instrumenata, koje su osmislili različiti umjetnici, dizajneri, elektroničari i svi oni koji su zainteresirani za kreiranje interaktivnih objekata ili okruženja.

Početni tim izumitelja Arduino platforme činili su Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino i David Mellis. Njegovi izumitelji su izdvojili glavne razloge uspješnosti Arduino platforme, po kojima se ona razlikuje od tada korištenih mikrokontrolera [1-2]:

- Cijena - Arduino ploče su relativno jeftine u usporedbi sa ostalim mikrokontrolerima. Najjeftiniji model Arduina (Arduino UNO) moguće je naručiti po cijeni od 22\$.
- Više-programski - Arduino softver, tzv. IDE, možemo koristiti na Windows, Macintosh OSX i Linux operacijskim sustavima, dok je većina ostalih mikrokontrolera ograničena na rad na Windows operacijskom sustavu.
- Jednostavno programsko okruženje - Arduino softver je jednostavan za korištenje početnicima, ali i dovoljno fleksibilan da bude zanimljiv naprednim korisnicima. Učiteljima je također pristupačan jer se temelji na proceduralnom programskom okruženju pa se studenti koji su već upoznati sa takvim okruženjima lako mogu prilagoditi i na Arduino IDE. Samo programiranje pločice ne zahtijeva dodatan dio hardvera, tzv. programator, kao što to zahtijevaju ostali mikrokontrolerski sustavi, već je dovoljan USB kabel koji se može povezati sa svakim računalnom, bez obzira na operacijski sustav.

- *Open source* platforma sa mogućnošću proširenja programskog okruženja - dozvoljeno je dijeljenje i preuređivanje programskog koda. Programski jezik može biti proširen uz pomoć C++ biblioteka, a oni koji žele razumjeti njegove tehničke detalje mogu proučiti programski jezik AVR C na kojem je baziran jezik Arduina. Također, za one naprednije, moguće je i izravno dodavati AVR C kod u programe bazirane na Arduinu.
- *Open source* hardver sa mogućnošću proširenja - planovi Arduino ploča su objavljeni pod „Creative Commons“ licencom, što znači da svi vještiji dizajneri mogu stvarati vlastite verzije modula, poboljšavati ih i improvizirati. Upravo na taj način je nastala i prva hrvatska Arduino kompatibilna pločica pod nazivom CROduino [3]. Čak i relativno neiskusni korisnici mogu napraviti maketnu verziju modula kako bi bolje razumjeli na koji način funkcionira i uštedjeli novac.

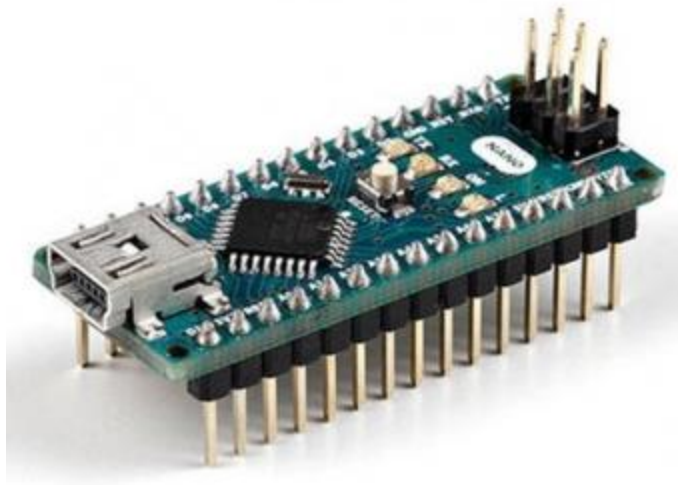
Sastoji se od hardver dijela koji je elektronički programibilni strujni krug, poznat i kao mikrokontroler, te softver dijela koji se naziva IDE (eng. *Integrated Development Environment*) kojeg korisnik pokreće na svom računalu te iz njega programira i upravlja samom pločicom. Hardver se sastoji od pločice otvorenog koda dizajnirane oko 8-bitnog Atmel AVR mikrokontrolera ili 32-bitnog Atmel ARM-a koja vrši interakciju s računalima putem USB priključka. Na pločici uz mikrokontroler nalazimo i razne komplementarne komponente koje olakšavaju programiranje i interakciju s drugim krugovima. Također, velika prednost Arduina je to što se njegovu pločicu može povezati i sa nizom izmjenjivih modula poznatih pod nazivom štitovi (eng. *shields*) koji imaju različite dodatne funkcije [4]. Štitovi nam mogu pružiti mogućnost spajanja na Internet, kontroliranja elektromotora, korištenja GPS-a, pretvaranja Arduina u glazbeni instrument i još puno drugih mogućnosti. Neki od najčešće korištenih štitova su: Arduino Ethernet Shield, Arduino Wi-Fi Shield, MP3 Player Shield, Ardumoto Motor Driver Shield i drugi.

Softver Arduina, odnosno Arduino IDE čini standardni kompajler i pokretač operativnog sustava (eng. *bootloader*) koji se nalazi na samoj pločici, po čemu je Arduino posebit jer ostali mikrokontroleri u svrhu *bootloadera* zahtijevaju eksterno programiranje. *Bootloader* je program koji se pokreće sa uključivanjem Arduina ili pritiskom na tipku reset te pojednostavljuje

preuzimanje programa sa računala na *flash* memoriju Arduino pločice. IDE sadrži i uređivač koda koji ima različite mogućnosti kao što su označavanje koda, uparivanje zagrada, automatsko uvlačenje linija i slično.

## 2.1. Arduino Nano

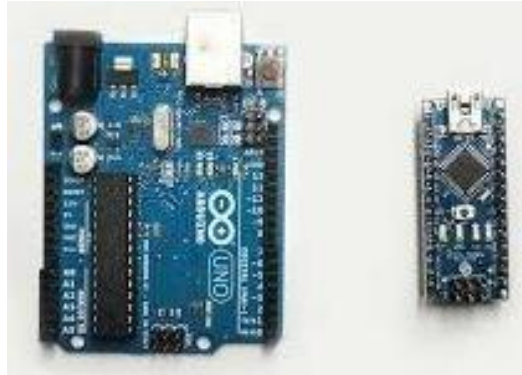
Razlikujemo više modela Arduino pločica koje možemo podijeliti po razinama [5]. Najpoznatije pločice i moduli za Arduino početnike su Arduino Uno, Arduino Leonardo, Arduino 101, Arduino Micro, Arduino Nano te Arduino Mini. Uz njih postoje još i one sa poboljšanim značajkama kao što su Arduino Mega, Arduino Zero, Arduino Due, Arduino Mo, Arduino Mo Pro te prijenosne pločice kao što su Arduino Gemma, Lilypad Arduino USB, Lilypad Arduino Main Board itd.



Slika 1. Arduino Nano pločica [9]

Prilikom izrade ovog projekta korišten je Arduino Nano (slika 1), relativno mala pločica težine 7 g, koja se bazira na mikrokontroleru ATmega 328 sa 32 KB *Flash* memorije za pohranjivanje koda, od čega 2 KB koristi *bootloader* [6]. Arduino Nano koristi isti mikrokontroler kao i Arduino Uno koji je najpoznatija i najčešće korištena Arduino pločica. Osim razlike u veličini (slika 2), što sugestira i samo ime pločica, Arduino Nano se razlikuje i po

rasporedu komponenti na pločici, kao i po dva dodatna analogna pina koje sadrži. Arduino Uno za spajanje sa računalom i napajanje koristi USB priključak, dok za isto Arduino Nano koristi Mini-B USB priključak [7].

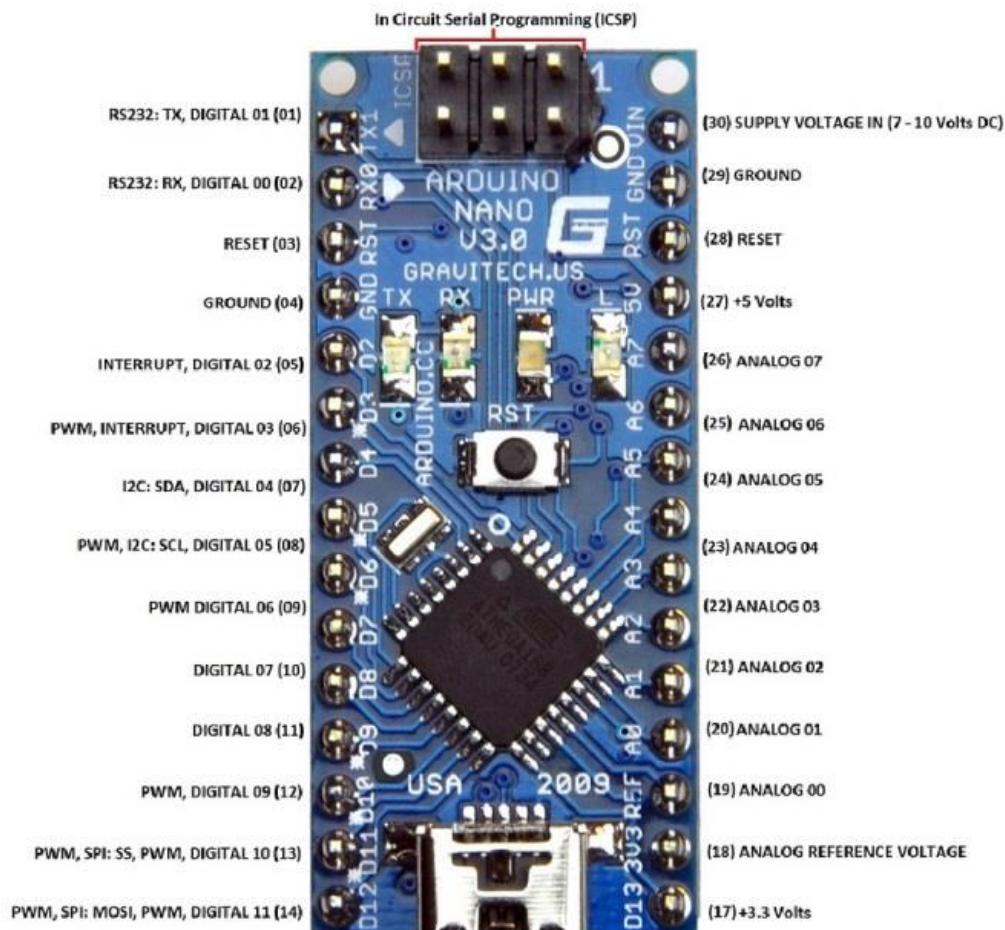


Slika 2. Usporedba veličine pločice Arduino Uno (lijevo) i Arduino Nano (desno) [21]

Na pločici Arduino Nano nalazi se mikrokontroler, 8 analognih ulaza i 14 digitalnih ulaza / izlaza od kojih se 6 može koristiti za PWM izlaze (digitalni pinovi 3, 5, 6, 9, 10 i 11) kao što je naznačeno na slici 3 [8]. PWM izlazi koriste se kako bi simulirali analogni izlaz upotrebom impulsno širinske modulacije na digitalnom pinu uz pomoć funkcije *analogWrite()* koja nam vraća 8-bitnu vrijednost. Uz to, pronalazimo i Mini-B USB priključak, ICSP zaglavlje, tipku za reset i LED diode. Moguće ga je napajati putem Mini-B USB priključka, 5V stabiliziranim vanjskim napajanjem (pin 27) ili 6-20 V nestabiliziranim vanjskim napajanjem (pin 30). Izvor napajanja automatski se odabire na najveći izvor napona. [9] Preporučeni napon napajanja je između 7 V i 12 V. Mikrokontroler ATmega 328 ima 32 KB memorije od kojih 2 KB koristi *bootloader*, 2 KB SRAM radne memorije i 1 KB EEPROM memorije koju je moguće izbrisati i ponovno programirati.

Svaki od 14 digitalnih pinova na Arduino Nano pločici se koristi pri naponu od 5 V i može biti konfiguriran kao ulaz (eng. *input*) ili kao izlaz (eng. *output*). Za to se primjenjujemo funkcije *pinMode()*, *digitalWrite()* i *digitalRead()*. Pri pokretanju Arduina svi pinovi inicijalno su postavljeni kao ulazni pinovi. Svaki od pinova može primiti ili dati maksimalno 40 mA te ima unutrašnji *pull-up* otpornik jačine 20-50 k $\Omega$  koji je prema zadanim postavkama isključen [9]. Razlikujemo dva stanja digitalnog signala: HIGH i LOW. Za pinove koji imaju maksimalnu vrijednost napona od 5 V kažemo da su u stanju HIGH, dok su oni koji su spojeni na napon od 0 V ili uzemljenje

(GND) u stanju LOW [8]. Neki od digitalnih pinova imaju specijalizirane funkcije. Na primjer, pin 13 ima ugrađenu LED diodu koja svijetli kada je vrijednost pina postavljena na HIGH, a ugašena je kada je vrijednost pina LOW.



*Slika 3. Pinovi Arduino Nano pločice [8]*

Arduino Nano ima 8 analognih ulaza, od kojih svaki osigurava 10-bitnu rezoluciju, tj. 1024 različite vrijednosti. Mogu imati bilo koju vrijednost između 0 V i 5 V. Arduino pločica ima ugrađenu i tipku za reset (vidimo na slici 3). Pritiskom na nju trenutačno povezujemo reset pin sa uzemljenjem te ponovno pokrećemo koji je učitan na pločici. Ova karakteristika je jako korisna u slučajevima kada se učitan kod ne ponavlja, ali ga je potrebno testirati više puta. Uz fizički dodanu tipku za reset, Arduino Nano ima i opciju automatskog softver reseta, tj.

ponovnog pokretanja programa ukoliko je pločica povezana sa računalom [9]. Dodatne tehničke specifikacije i podatci o Arduino pločici navedeni su u tablici 1.

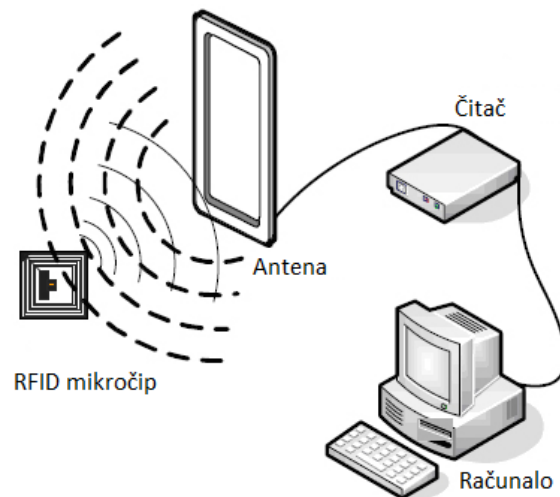
Mikrokontroler:	ATmega328
Radni napon:	5 V
<i>Flash</i> memorija	32 KB (2 KB koristi <i>bootloader</i> )
SRAM	2 KB
EEPROM	1 KB
Takt	16 MHz
Analogni pinovi	8
Digitalni ulazno / izlazni pinovi	22 (od čega su 6 PWM pinovi)
Struja prema ulazno / izlaznim pinovima	40 mA
Ulazni napon (preporučeni)	7 - 12 V
Ulazni napon (ograničenja)	6 – 20 V
PWM izlazi	6
Potrošnja energije	19 mA
Veličina pločice	18 x 45 mm
Težina	7 g

*Tablica 1. Tehničke specifikacije pločice Arduino Nano [9]*



### 3. RFID tehnologija

Tehnologija radiofrekvencijske identifikacije (RFID tehnologija, engl. *radio frequency identification*) je naziv za tehnologije koje koriste radio valove kako bi automatski identificirali objekte. RFID uređaji služe istoj svrsi kao i bar kod ili magnetska traka na poleđini kreditne kartice, pruža jedinstveni identifikator za taj objekt. Baš kao što bar kod ili magnetska traka moraju biti očitani kako bi se dobile informacije, tako i RFID uređaj mora biti skeniran kako bi se pronašle identifikacijske informacije. Njegova prednost je što, za razliku od bar koda i magnetske trake, RFID uređaj ne mora biti smješten u neposrednoj blizini samog čitača već, ovisno o specifikacijama pojedinačnog uređaja, može funkcionirati i na udaljenostima od nekoliko metara. Ovu tehnologiju primjenjujemo u aplikacijama gdje je potrebna sigurna i jednostavna identifikacija, te dugotrajnost i otpornost identifikatora na različite specifične utjecaje okoline, a gdje nije potrebna izravna vidljivost. U većini okruženja RFID postiže 99.5 % do 100 % očitavanja u prvom skeniranju [10].



Slika 4. Ilustracija RFID sustava [22]

Radiofrekvencijska komunikacija temelji se na stvaranju elektromagnetskih valova u odašiljačima i njihovom otkrivanju na udaljenom prijemu. Postoji nekoliko metoda identifikacije objekata, no najčešća je pohranjivanje identifikacijskog serijskog broja ili neke druge informacije na mikročip koji zajedno s antenom čini RFID transponder. Transponder komunicira s čitačem putem radio signala, jednosmjerno ili dvosmjerno, a čitač je povezan s

računalom ili računalnom mrežom na kojemu se nalazi baza podataka (slika 4). Jednostavna identifikacijska oznaka pohranjena na transponderu u ovoj bazi povezana je s informacijama o označenom proizvodu [11].

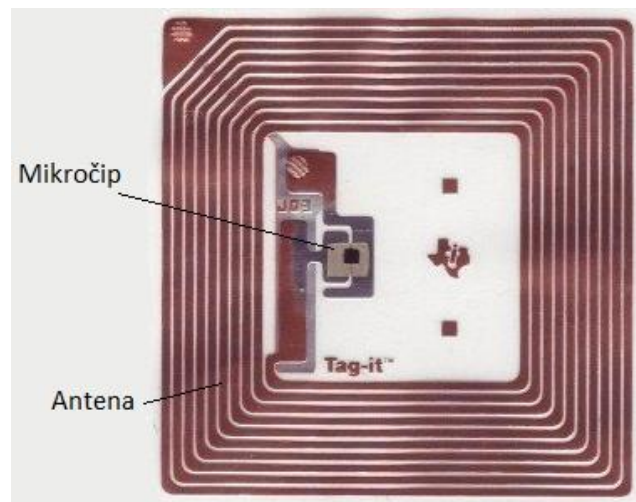
### 3.1. RFID sustav

RFID sustav sastoji se od tri dijela: računalo, čitača sa dekoderom za interpretiranje podataka i transpondera koji je programiran sa svim podacima za identifikaciju [12], [14]. Antena transpondera je putem radiofrekvencijskih valova vezana za odašiljač signala, odnosno čitač. Kada se čitač primakne frekvencijskom području antene, on postaje aktivan i šalje anteni podatke koji su programirani u njegovu memoriju. Zatim čitač prima signal antene, dekodira ga i šalje informaciju u računalo. Razlikujemo sustave u kojima jedna antena prima i odašilje signal, od onih u kojima jedna odašilje, a druga antena prima signal.

Čitač ima zadatak komunicirati sa transponderima te prenositi podatke dalje do računala. Međusobno ih razlikujemo po složenosti ovisno o tipu transpondera s kojima trebaju komunicirati, po korištenim frekvencijama i funkcijama koje moraju zadovoljiti. Najjednostavniji čitači omogućuju čitanje samo jedne vrste transpondera, koristeći samo jednu frekvenciju i jedan protokol, dok oni složeniji koriste različite protokole, omogućuju odabir podataka, provjeru i ispravljanje grešaka. RFID čitači su najčešće stalno aktivni, konstantno odašiljući energiju radio signalom u potrazi za transponderima koji su im ušli u domet. Kod nekih primjena ovo je nepotrebno, a ako se radi o baterijski napajanim uređajima može čak biti i nepoželjno sa stajališta uštede energije. Zbog toga je neke čitače moguće postaviti tako da odašilju radio impulse kao odgovor na neki vanjski podražaj. Kada je signal transpondera primljen i dekodiran, prema protokolu „Command Response“ (naredba – odgovor), čitač će na ponovljeno slanje signala odgovoriti instrukcijom transponderu da prestane emitirati. Neka od bitnijih obilježja čitača su radna frekvencija (HF ili UF), podrška za razne protokole transpondera, mogućnost umrežavanja više čitača, mogućnost upravljanja višestrukim antenama itd. [11], [13].

## 3.2. RFID transponderi

Riječ transponder izvedena je od termina „transmitter / responder“ prema funkciji tog uređaja koji na transmisiju čitača odgovara podatkom. RFID transponderi se sastoje od dvije osnovne komponente, od mikročipa i antene, koji su najčešće smješteni u kućište otporno na utjecaj okoline (slika 5). Mikročip sadrži radio prijammnik, radio modulator za slanje odgovora čitaču, upravljačku logiku, memoriju i sustav za upravljanje napajanjem. Svaki RFID transponder sadrži određenu veličinu EEPROM memorije (električno izbrisive programibilno ispisne memorije) u kojoj su pohranjeni podaci o proizvodu.



Slika 5. RFID transponder [13]

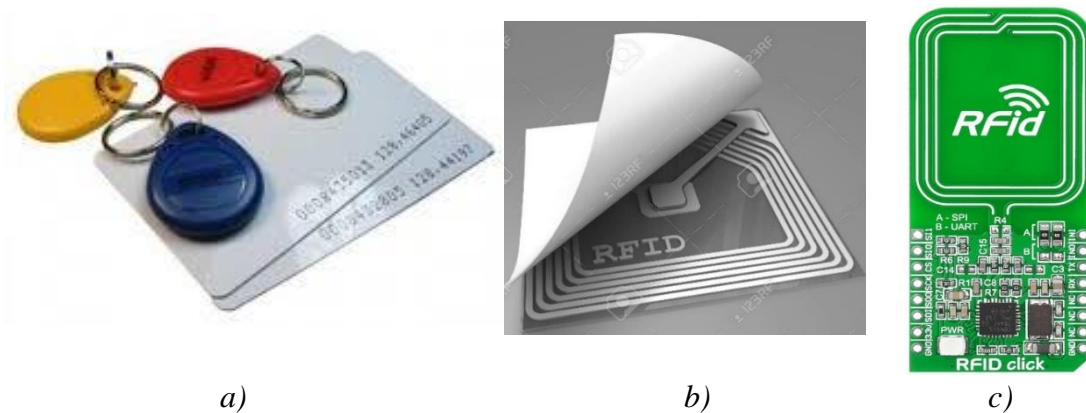
Transponderi se međusobno razlikuju po radiofrekvencijskim valnim duljinama putem kojih komuniciraju, po obliku, veličini, protokolu ili jeziku, jačini, načinu pohranjivanja podataka, itd. Neke od najčešće korištenih podjela kod transpondera su sljedeće [11]:

- Načinu napajanja:
  1. Pasivni,
  2. Djelomično aktivni,
  3. Aktivni

- Mogućnosti čitanja i zapisivanja na transponder:
  1. *Read Only* (R) – transponderi koje je moguće samo čitati (u procesu proizvodnje dobivaju svoj jedinstveni serijski broj koji više nije moguće promijeniti),
  2. *Write Once Read Many* (WORM) – transponderi koji omogućuju jednostruko programiranje (korisnik sam programira memoriju transpondera, ali podatak može upisati samo jednom, nakon čega on ostaje permanentno pohranjen),
  3. *Read/Write* (R/W) – najnapredniji oblik transpondera s mogućnošću višestrukog programiranja (obično imaju jedinstven i trajan serijski broj kojem se dodaju zapisani podaci). Takvi transponderi korisni su u kompleksnijim aplikacijama, ali budući da su skuplji, nisu praktični za označavanje jeftinijih proizvoda.
  
- Korištenim frekvencijama [10], [15]:
  1. Niske frekvencije (LF, eng. *Low Frequency*) – globalno korištene frekvencije, rade u rasponu od 100 do 500 kHz, a najčešće na 125 kHz. Imaju najniže cijene, ali i najkraći domet signala i najmanje brzine očitavanja i prijenosa,
  2. Visoke frekvencije (HF, eng. *High Frequency*) – rade u rasponu od 10 do 50 MHz, a najčešće na 13.56 MHz. Nižeg su ili srednjeg cjenovnog ranga, kratkog do srednjeg dometa signala, srednje brzine očitavanja i prijenosa. Najčešće se koriste u sustavima elektronskih karata, beskontaktnog plaćanja, praćenja pošiljki, itd.
  3. Ultravisoke frekvencije (UHF, eng. *Ultra High Frequency*) – rade u rasponu od 433 do 930 Mhz i 2.45 GHz. Ovim frekvencijama nije moguće globalno pristupati, već postoje značajna ograničenja kod njihove upotrebe. Imaju najveći domet signala, veće brzine prijenosa, ali i cijene. Kod ovih transpondera ne smije biti prepreke između njih i čitača jer UHF radio-val ne prodire tako dobro kroz materijale te zahtijeva više energije za transmisiju nego val niže frekvencije.

Dometa signala RFID transpondera prije svega je vezan uz korištenu frekvenciju, ali i uz vrstu napajanja. Tako je tipičan dometa transpondera koji radi na niskim frekvencijama do 30 cm, na visokim frekvencijama do 1 m, a na UHF (eng. *Ultra High Frequency*) frekvencijama do 6 m. Upotrebom aktivnih, baterijski napajanih, izvedbi transpondera navedeni dometa mogu se povećati.

- Fizičkoj izvedbi (slika 6):
  1. RFID etikete (eng. *tag*),
  2. RFID naljepnice
  3. RFID tiskane pločice.



Slika 6. Fizičke izvedbe transpondera: a) etikete, b) naljepnica i c) tiskana pločica [23]-[25]

Nositelj informacije u obliku etikete, naljepnice ili tiskane pločice obično se postavlja na objekt, ambalažu, paletu, kontejner ili na sam proizvod, tako da može putovati s njime i na svakom koraku ga identificirati. Podaci pohranjeni u transponderu mogu biti raznovrsni – svakako će identificirati proizvod na traci, robu u tranzitu, lokaciju, vozilo, također i životinju ili osobu, ali mogu predstavljati i instrukcije o daljnjem postupanju.

Transponderi mogu imati memoriju koju čini samo jedan bit. Primjer sustava koji koristi takve transpondere je sustav za elektroničko praćenje artikala (eng. EAS, *Electronic Article Surveillance*) u trgovini. Takvi RFID transponderi ne sadrže nikakve posebne informacije već služe kako bi se razni proizvodi osigurali od krađe. EAS transponderi nemaju jedinstveni serijski broj, već samo prikazuju prisutnost kada prolaze kroz signalno frekvencijsko polje, te im je potreban samo jedan bit da bi pokrenuo alarm jednom kad je pobuđen u polju čitača.

Za pohranu serijskog broja, po mogućnosti zajedno s kontrolnim bitovima, dovoljno je 128 bitova. Serijski, odnosno identifikacijski broj može pohraniti proizvođač ili sam korisnik unutar svoje aplikacije. Veći kapaciteti memorije, do 512 bita, uvijek su programibilni. Kod takvih transpondera, osim same identifikacije, korisnik može pohraniti razne podatke o označenom objektu, upute za daljnje postupke u nekom procesu ili rezultate ranijih akcija nad objektom. Transponderi s 64 kilobita memorije obično nose datoteke s podacima organiziranim u polja koja se mogu selektirati tijekom procesa čitanja. Za većinu aplikacija dovoljan je 96-bitni serijski broj, a transponder će na kraju puta proizvoda koji je njime označen ionako biti odbačen.

### **3.2.1. Podjela transpondera prema načinu napajanja**

RFID transponderi za rad trebaju energiju, iako u izuzetno malim količinama. Prema načinu napajanja, dijelimo ih na tri vrste: pasivne, djelomično aktivne i aktivne transpondere.

#### **Pasivni transponderi**

Pasivni RFID transponderi nemaju unutrašnji izvor napajanja već za napajanje sklopovlja koriste energiju prikupljenu iz dolaznog radio signala određenih svojstava. Kako nemaju unutrašnji izvor napajanja pasivni transponderi moraju imati mogućnost skladištenja energije tijekom primanja signala, na primjer kondenzator, kako bi se omogućilo slanje odgovora kada čitač prestane odašiljati ili čitač odašilje tijekom cijele komunikacije što znači da transponder

odgovara na različitoj frekvenciji. Ovisno o veličini i dizajnu antene, te o odabranoj radio frekvenciji, domet signala je od oko 10 cm pa do nekoliko metara.

Razlikujemo tri tehnike energetskog uparivanja čitača i pasivnih transpondera [11]:

1. Elektromagnetska indukcija: primjenjuje se na malim udaljenostima. Antene čitača i transpondera građene su kao zavojnice s velikim brojem zavoja, te zajedno tvore transformator. Električna struja u zavojnici antene čitača stvara magnetsko polje koje inducira struju u zavojnicama antene transpondera. Čitač komunicira s transponderom modulirajući amplitudu, frekvenciju ili fazu vala nosioca. Transponder s čitačem komunicira variranjem opterećenja na svojoj anteni čime utječe na napon na čitačevoj anteni.
2. Reflektiranje: koristi se na većim udaljenostima. Do toga dolazi kada se elektromagnetski val reflektira od neke površine natrag prema odašiljatelju. Količina reflektirane energije ovisi o tome koliko dobro reflektirajuća površina rezonira s dolaznim elektromagnetskim valom. Izmjeničnim paljenjem i gašenjem reflektiranja transponder stvara uzorak kojega čitač uočava.
3. Elektrostatičko uparivanje: najrjeđe korištena metoda energetskog uparivanja. Kod ovog pristupa antene čitača i transpondera djeluju kao nabijene ploče. Dodavanje elektrona na antenu čitača uklanja elektrone s antene transpondera, i obrnuto.

Nepostojanje unutrašnjeg izvora napajanja omogućuje izvedbe pasivnih transpondera vrlo malenih dimenzija, prikladnih za ugradnju u naljepnice ili pod kožu. Moguće su i izrazito jeftine izvedbe što ih čini najpopularnijima kod masovnih primjena u trgovačkim lancima.

## **Djelomično aktivni transponderi**

Djelomično aktivni RFID transponderi sadrže bateriju koja napaja mikročip, dok za napajanje antene koriste energiju prikupljenu iz signala čitača. Upravo zbog takve strukture napajanja, kao zbog radnih karakteristika koje su djelomično aktivno, odnosno djelomično pasivne, takvi transponderi se nalaze u posebno kategoriji između pasivnih i aktivnih transpondera s pojedinim prednostima i manama obje skupine.

## **Aktivni transponderi**

Aktivni RFID transponderi posjeduju unutrašnji izvor napajanja koji se koristi za napajanje sklopovlja i za emitiranje radio signala. Takvi izvori napajanja, tj. baterije, imaju ograničen vijek trajanja, uobičajeno nekoliko godina, ovisno o uvjetima okoline i korištenju. Određene vrste aktivnih transpondera mogu imati i zamjenjivu bateriju. Veća snaga emitiranog radio signala, omogućena vlastitim napajanjem, čini ove transpondere mnogo učinkovitijim u zahtjevnim radnim okolinama kao što su voda (uključujući ljude i životinje koji su velikim postotkom građeni od vode) ili metal te na većim udaljenostima. Uz to, skuplji su i veći od pasivnih transpondera, ali imaju i bolji imunitet na buku te bržu transmisiju podataka u području visoke frekvencije. Obično mogu funkcionirati u vrlo velikom rasponu temperatura od  $-50^{\circ}\text{C}$  do  $+70^{\circ}\text{C}$  [10]. Mnogi aktivni transponderi upotrebljivi su na udaljenostima do nekoliko stotina metara i imaju životni vijek baterije do 10 godina. RFID transponderi su do sada integrirani sa raznim sensorima, npr. sa sensorima temperature, vlažnosti, vibracija, svjetla i radijacije.

### **3.2.2. Podjela transpondera prema načinu fizičke izvedbe**

Prema načinu fizičke izvedbe, RFID transpondere dijelimo u tri kategorije: RFID etikete (eng. *tag*), RFID naljepnice i RFID tiskane pločice (eng. PCB – *Printed Circuit Board*) [11].

#### **RFID etikete**

RFID etikete se proizvode u različitim veličinama, oblicima, s različitim kapacitetima memorije i različitim fizičkim karakteristikama. Mogu biti dovoljno male da se smjeste pod kožu životinje, mogu biti oblika čavla ili vijka za označavanje drvene građe ili u obliku kreditne kartice za korištenje u aplikacijama kontrole pristupa. Veliki plastični privjesci za sprečavanje krađe odjeće u trgovinama također su RFID etikete, a slični su i vrlo otporni transponderi u obliku bloka kojima se označavaju kontejneri u internim procesima proizvodnje, ili radni strojevi



i kamioni u svrhu praćenja i održavanja. Gotovo svi su nekom vrstom kućišta zaštićeni od udaraca, kemikalija, vlage i prašine.

## **RFID naljepnice**

Barkod kao tehnologija automatske identifikacije u upotrebi je već desetljećima i vrlo je dobro afirmirana. Međutim, barkod naljepnice nakon što su jednom otisnute, ne mogu više biti promijenjene, a da bi je skener pročitao mora mu biti u vidnom polju. Nova generacija *smart* naljepnica opremljena je RFID tehnologijom i nadilazi neka od glavnih ograničenja tradicionalnog barkoda. Integrirani elektronički sklop sadrži memoriju i može biti programiran ili reprogramiran korištenjem radiovalova. *Smart* naljepnice imaju očitu prednost pred tradicionalnim barkod naljepnicama u aplikacijama gdje je potrebna kombinacija efikasnosti čitanja i vizualna, ljudskom oku razumljiva informacija.

## **RFID tiskane pločice**

Tiskana pločica (eng. PCB - *Printed Circuit Board*) je namijenjena ugradnji u proizvod ili ambalažu. Prednosti su joj niža cijena i sposobnost podnošenja uvjeta okoline koje RFID naljepnice ne bi podnijele.

### **3.2.3. EPCglobal podjela transpondera**

EPCglobal vodeća je standardizacijska organizacija na području RFID tehnologija. Ona je definirala dvije podjele RFID uređaja, na šest klasa i u dvije generacije. Navedene podjele dane su u sljedećim dvjema tablicama.

<b>EPC klasa</b>	<b>Definicija</b>	<b>Programiranje</b>
Klasa 0	Moguće samo čitanje	Programira ih proizvođač
Klasa 1	Jednostruko programiranje	Programira ih korisnik
Klasa 2	Višestruko programiranje	Programibilni
Klasa 3	Djelomično aktivni transponderi	
Klasa 4	Aktivni transponderi	
Klasa 5	Čitači	

*Tablica 2. Podjela RFID uređaja na klase [11]*

<b>Karakteristika</b>	<b>Prva generacija</b>	<b>Druga generacija</b>
Frekvencija	860-930 MHz	860-960 MHz
Memorijski kapacitet	64 ili 96 bita	96-256 bita
<i>Field</i> programibilnost	Da	Da
Reprogramibilnost	klasa 0 – samo čitanje klasa 1 – jednostruko programiranje višestruko programiranje	/
Ostale karakteristike	/	brži i pouzdaniji čitači, bolje poštivanje ostalih globalnih standarda

*Tablica 3. Generacije RFID čipova [11]*

### 3.3. Primjena RFID tehnologije

Još od pojave proizvodnje u velikim razmjerima, metode brze identifikacije pomažu prilikom ubrzanja procesa rukovanja robom i materijalima. Povijesno gledajući, jeftine i jednostavne tiskane naljepnice su bile glavna osobina prerađivačke tehnologije. Sedamdesetih godina prošlog stoljeća, veliki korak naprijed u etiketiranju proizvoda napravila je pojava barkodova koji su omogućili automatizaciju i standardizaciju procesa identifikacije. Iako su jeftini u proizvodnji, barkodovi imaju veliki broj ograničenja zbog kojih se razvila alternativna tehnologija označavanja proizvoda, RFID.

Danas su tri velike organizacije predvodnici usvajanja RFID tehnologije u velikoj mjeri: Wal-Mart, Tesco i Ministarstvo obrane SAD-a [16]. Svaka od njih ima cilj ponuditi konkurentnije cijene uz korištenje RFID-a pomoću koje znatno smanjuju operativne troškove pojednostavljuvanjem praćenja zaliha, prodaje i narudžbi. Kada se koristi u kombinaciji sa računalnim bazama podataka i upravljanjem zalihama, povezanih putem digitalne komunikacijske mreže, RFID može odrediti pojedinačne stavke dok se kreću između tvornica, skladišta, vozila i trgovina.

Uz navedeno, RFID ima mnoge primjene koje često ne bi očekivali. Primjer toga su pametne garderobe koje su opremljene s interaktivnim RFID kioscima. Skeniranjem artikala odjeće, kupci mogu pristupiti podacima o proizvodu, pronaći slične alternative i dati povratne informacije. Time se trgovcima povećavaju šanse da njihovi kupci pronađu ono što traže, ali isto tako i dobiju pristup pouzdanoj tehnologiji koja smanjuje mogućnost krađe odjeće. RFID koriste i kockarnice koje u poker čipove ugrađuju RFID transpondere čime prate potrošnju igrača i koriste te informacije kako bi ih zadržali u igri što dulje s pravilno tempiranim pićima i različitim uslugama. Na taj način dolazi do značajne prevencije gubitaka i osigurava se stara izreka prema kojoj kuća uvijek pobijedi. Uz to, RFID transponderi se ugrađuju i u golf loptice kako bi se smanjio njihov gubitak na golf terenima i skratilo vrijeme njihovog sakupljanja. Primjenjuju se i u knjižnicama u kojima su praktičniji od klasičnog barkoda, koji zahtijevaju linijsku vidljivost, jer se RFID oznake mogu čitati iz više kutova. Time je proces naplate i prijave te vođenje evidencije stanja knjiga na policama znatno jednostavniji i brži.

2002. godine došlo je do značajnog napretka za široku primjenu RFID tehnologije kada je Philips razvio novi standard koji je rezultirao NFC tehnologijom (eng. *Near-Field Communication*). Takva tehnologija predstavlja spajanje aktivnog signaliziranja između mobilnih uređaja koji koriste blisko povezivanje uz primjenu pristupa koji je kompatibilan sa čitanjem postojećih pasivnih RFID proizvoda. Cilj novog NFC standarda je osigurati mehanizam pomoću kojeg bežični mobilni uređaji mogu komunicirati s uparenim uređajima u neposrednoj blizini (do 20 cm) umjesto da se oslanjaju na mehanizme otkrivanja radio-standarda kratkog dometa kao što su Bluetooth i Wi-Fi [16]. 2004. godine Nokia je najavila mobitel Nokia 3200 (slika 7), prvi mobilni uređaj koji je ima ugrađen NFC čitač.



*Slika 7. Nokia 3200 s ugrađenim NFC čitačem na stražnjoj strani kućišta [16]*

Danas se NFC tehnologija koristi u aplikacijama kao način plaćanja. Google Wallet jedan je od najpoznatijih servisa koji povezuje željeni račun s NFC uređajem, te podržava veliki broj kartičnih kuća pa se ova usluga već poprilično ukorijenila u razvijenim zemljama. Uz primjenu u plaćanju, NFC tehnologija se koristi i u sustavima za otključavanje i paljenje automobila. Takvi sustavi rade identično kao i sustavi za plaćanje - za otvaranje vrata ili otključavanje automobila potrebno je dotaknuti NFC prijemnik koji prima signale sa pametnog telefona i jedinstveno nas identificira. Na taj način se uz novčanike, eliminiraju i ključevi.

## 4. Izrada konstrukcije

Prilikom izrade ovog prototipa potrebno je bilo izraditi konstrukciju same kante na koju ćemo kasnije postaviti sve komponente koje osiguravaju ispravan rad sustava. Cjelokupan popis materijala i alata potrebnog za izradu konstrukcije kante naveden je u nastavku.

Materijali za konstrukciju:

- 2 x pleksiglas ploča 15 x 36 cm (debljine 3 mm)
- 2 x pleksiglas ploča 15 x 29 cm (debljine 3 mm)
- 2 x pleksiglas ploča 60 x 36 cm (debljine 3 mm)
- 2 x pleksiglas ploča 60 x 29 cm (debljine 3 mm)
- 2 x pleksiglas ploča 28.5 x 35.5 cm (debljine 3 mm)
- 1 x pleksiglas ploča 26 x 34 cm (debljine 3 mm)
- 1 x pleksiglas ploča 30 x 36 cm (debljine 3 mm)
- 2 x metalna spojnica
- Metalni L profili
- Plastična kanalica (duljine 60 cm)
- Vijci i matice

Potreban alat:

- Bušilica i pripadajuća svrdla
- Odvijač i kliješta
- Pištolj za vruće lijepljenje

Izrada konstrukcije započeta je međusobnim povezivanjem pleksiglas ploča i metalnih L profila korištenjem vijaka i matica. Na samim pločama bilo je potrebno probušiti rupe za vijke kako bi ih mogli učvrstiti L profilima, za što smo koristili bušilicu sa odgovarajućim svrdlima. Ploče veličine 60 x 36 cm i veličine 60 x 29 cm su na dužim stranicama spojene L profilima tako da čine okvir kante oblika kvadra. Na vrh tog okvira postavili smo ploču veličine 30 x 36 cm kao poklopac jer je zbog jedne duže stranice koja prelazi rub kante korisniku jednostavnije njome

rukovati i otvarati poklopac. Jednu od stranica gornje ploče duljine 36 cm smo pričvrstili za okvir uz pomoć dvije metalne spojnice koje su pomične pa omogućuju gibanje samog poklopca.



*Slika 8. Konstrukcija kante*

Na dno ovog okvira smo postavili dodatni kvadratni dio unutar kojeg smo pohranili sve komponente koje trebaju biti na sigurnom i van dohvata korisnika. Tom dijelu su stranice činile ploče veličina 15 x 36 cm i 15 x 29 cm, a njegova donja i gornja ploha čine 2 ploče veličine 28.5 x 35.5 cm. Gornja ploha ovog dijela je ujedno i samo dno okvira koji smo prethodno sastavili sa stranicama veličine 60 x 36 cm i 60 x 29 cm. Sve stranice dodatnog donjeg dijela kante povezali smo metalnim L profilima pomoću vijaka i matica, a zatim na jednak način to spojili i sa ostatkom okvira. Nakon što smo sve navedene ploče učvrstili i osigurali da kanta bude stabilna, na tu konstrukciju smo pomoću pištolja za vruće lijepljenje postavili i međusobno spojili sve korištene komponente. Način spajanje tih komponenti detaljno je objašnjen u sljedećem poglavlju. Sve vodiče koji su povezivali komponente smještene na poklopcu sa ostatkom komponenti koje su smještene na dno kante smo postavili unutar plastične kanalice. Kanalicu smo fiksirali uz pomoć pištolja za vruće lijepljenje uz rub duže stranice ploče veličine 60 x 29 cm. Nakon što su sve komponente bile fiksirane, za kraj je bilo potrebno dodati posljednju ploču, veličine 26 x 34 cm, koja je postavljena na sami senzor vage te se postavljanjem otpada na nju vrši pritisak na senzor. Krajnji izgled izgrađene konstrukcije prikazan je na slici 8.

## 5. Spajanje komponenti sa Arduino pločicom

Prilikom izrade ovog projekta bilo je potrebno međusobno povezati sve komponente sa Arduino Nano pločicom. Korišteni su sljedeći elementi:

- LiquidCrystal ekran sa LCM1602 IIC adapterom,
- RFID – RC522 sa pripadajućim karticama za kontrolu pristupa,
- Solenoid brava 12V DC,
- Arduino kompatibilni mos driving modul VMA411.

Shema koja prikazuje na koji način su međusobno povezane sve komponente nalazi se u Prilogu 1, a zbog lakšeg razumijevanja postupak spajanja komponenti biti će podijeljen u nekoliko grupa koje će biti objašnjene u sljedećim potpoglavljima.

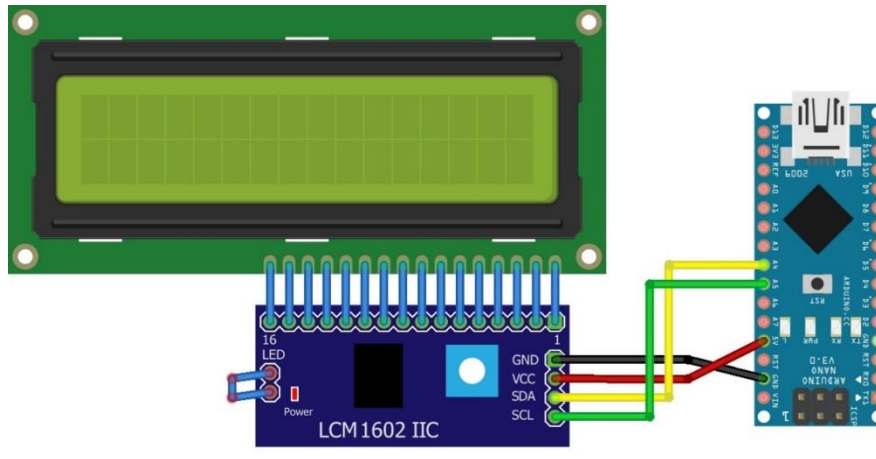
### 5.1. Spajanje LiquidCrystal ekrana sa IIC adapterom i Arduino pločice

LCD (engl. *Liquid Crystal Display*) monitor je vjerojatno jedan od najkorištenijih modula jer često postoji potreba za prikazivanje određenih podataka korisniku. Vrlo je jednostavan za korištenje i omogućuje nam ispisivanje bilo kakvih informacija na lako čitljivom ekranu. Postoje dvije vrste LCD ekrana: ekrani s paralelnom komunikacijom i ekrani sa serijskom komunikacijom. U izradi ovog projekta sam koristila onaj s paralelnom komunikacijom jer su najrasprostranjeniji i najdostupniji. Korišteni ekran je veličine 16 x 2, što znači da ima mogućnost ispisa po 16 znakova u 2 retka, ima plavo pozadinsko osvjetljenje i bijela slova te mu je za rad potreban napon od 5V.

Rad LCD ekrana bazira se na tekućim kristalima od kojih svaki tvori jedan piksel te su oni poredani ispred nekog svjetlosnog izvora.. Kristali se nalaze između dva polarizatora čije su osi polarizacije okomite jedna na drugu te zakreću molekule kristala ako postoji potencijal između njih. Promjenom veličine upravljačkog napona, zakreću se molekule kristala te se u ovisnosti o

iznosu upravljačkog napona mijenja njihova zakrenutost, odnosno količina svjetlosti koja može proći kroz polarizirajući filter. Na ovaj način kontrolira se razina osvijetljenosti slike na ekranu.

Uz LCD ekran koristila sam i IIC adapter koji pojednostavljuje spajanje LCD ekrana i Arduino pločice jer značajno smanjuje broj potrebnih vodiča za povezivanje. Shema spajanja LCD ekrana, IIC adaptera i Arduina Nano prikazana je na slici 9.



Slika 9. Shema spajanja LCD ekrana, IIC adaptera i Arduina Nano

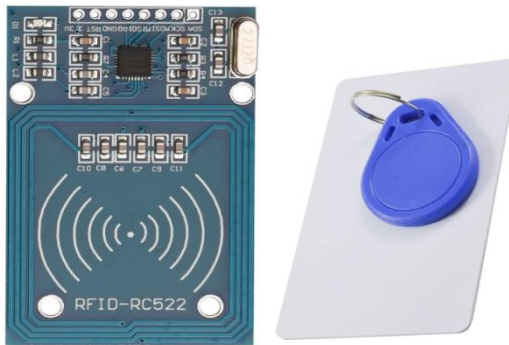
Koristila sam LCM1602 IIC adapter sa 16 pinova koje je za početak bilo potrebno lemljenjem paralelno povezati sa LCD ekranom koji također ima 16 slobodnih pinova. Nakon toga Arduino Nano povezujemo sa IIC adapterom na sljedeći način (slika 9):

- GND pin sa adaptera spajamo sa GND pinom sa Arduino pločice
- VCC pin sa adaptera spajamo sa 5V pinom sa Arduino pločice
- SDA (eng. *Serial Data Line*) pin sa adaptera spajamo sa A4 analognim pinom sa Arduino pločice
- SCL (eng. *Serial Clock Line*) pin sa adaptera spajamo sa A5 analognim pinom sa Arduino pločice



## 5.2. Spajanje RFID čitača sa karticama za kontrolu pristupa i Arduino pločice

Prilikom izrade ovog sustava odabrala sam RFID – RC522 čitač koji omogućava bežičan prijenos informacija jer je lako dostupan, pouzdan na malim udaljenostima i otporan na kvarove. Riječ je o integriranom krugu za beskontaktnu komunikaciju na 13.56 MHz. Ovaj čip upravlja antenom koja je dizajnirana za čitanje i slanje podataka. Bazna stanica, tj. RFID čitač sastoji se od antene, RF modula te kontrolne jedinice. RFID tagovi i kartice imaju sličnu konstrukciju, no bez aktivnih komponenti, što znači da ne sadrže nikakve baterije niti izvore napajanja. Sastoje se samo od antene i mikročipa [17].



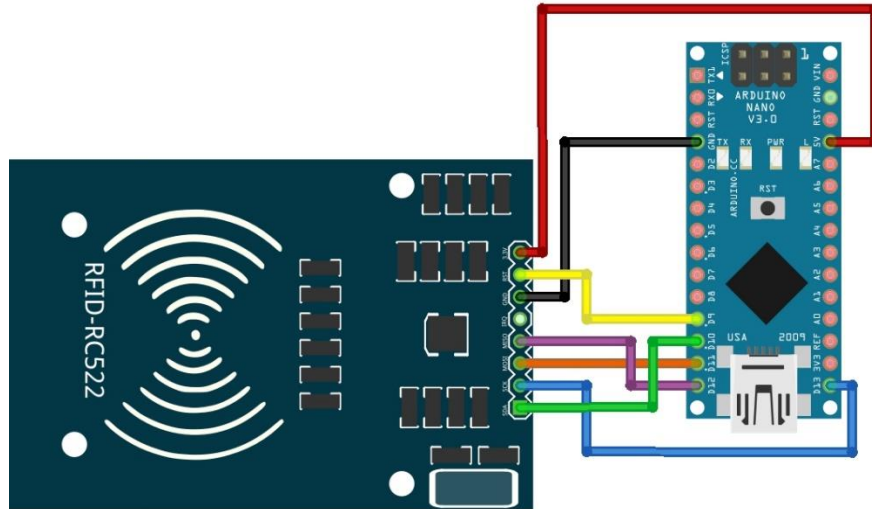
*Slika 10. RFID – RC522 čitač i njegove kartice za kontrolu pristupa [17]*

Neke od glavnih karakteristika odabranog čitača su:

- Radni napon: 3.3 V
- Struja: od 13 mA do 26 mA
- Integrirani krug: Philips MF RC522
- Radna frekvencija: 13.56 MHz
- Dimenzije: 66 x 40 x 7 mm
- Brzina prijenosa informacija: max. 10 Mbit/s
- Daljina očitavanja od senzora : <5cm (s tagovima koji dolaze uz ovaj čitač)

- Dolazi s jednom pasivnom RFID karticom i jednim privjeskom (slika 10)

Kartice koje smo koristili za kontrolu pristupa imaju memoriju od 1 KB EEPROM-a i radni vijek od 100000 upisivanja. Koristimo ih pri radnim temperaturama od -10 °C do +75 °C i pri frekvenciji od 13.56 MHz.



*Slika 11. Shema spajanja RFID – RC522 čitača i Arduina Nano*

Korišteni RFID čitač na sebi ima ukupno 8 pinova. Povezala sam ih sa Arduino Nano pločicom na sljedeći način (slika 11):

- SDA pin sa čitača spajamo sa pinom D10 sa Arduino pločice
- SCK pin sa čitača spajamo sa pinom D13 sa Arduino pločice
- MOSI pin sa čitača spajamo sa pinom D11 sa Arduino pločice
- MISO pin sa čitača spajamo sa pinom D12 sa Arduino pločice
- IRQ pin sa čitača ne spajamo
- GND pin sa čitača spajamo sa pinom GND sa Arduino pločice
- RST pin sa čitača spajamo sa pinom D9 sa Arduino pločice
- 3.3V pin sa čitača spajamo sa pinom 5V sa Arduino pločice

### 5.3. Spajanje Solenoid brave i Arduino pločice

Solenoid brava je u osnovi elektromagnet, izrađen od velike zavojnice od bakrene žice sa dijelom metala u sredini (slika 12). Prilikom napajanja namotaja zavojnice, dolazi do povlačenja metalnog dijela u središte zavojnice. Upravo zbog ovoga dolazi do otključavanja brave, tj. uvlačenja njenog pomičnog dijela sa jednog kraja. Solenoid koji sam koristila prilikom izrade ovog projekta je elektronička brava sa vanjskim kosim metalnim dijelom. U stanju mirovanja, brava je aktivna, tj. u zaključanom položaju, prilikom čega ne koristi nikakvu struju. Korištenjem napona od 9 do 12 V DC, kosi metalni dio se uvlači te je brava u otključanom stanju.



*Slika 12. Solenoid brava u aktivnom stanju [18]*

Karakteristike korištene brave su sljedeće:

- Radni napon: 12 V DC
- Potrebna struja: 500 mA (pri naponu od 9V) - 650 mA (pri naponu od 12V)
- Dimenzije: 2.4 cm x 6.7 cm x 2.8 cm

S obzirom da je za rad brave potreban značajno veći napon nego što to može pružiti Arduino Nano, potrebno ju je bilo spojiti sa korištenom akumulatorskom baterijom s naponom 12 V preko Arduino kompatibilnog mos modula VMA411. To sam napravila tako što sam jedan od dva moguća izvedena vodiča iz brave spojila na V+ izlaz mos modula, a drugi na V- izlaz modula. Iako su označeni crvenom i crnom bojom, za sam rad brave nije važno na koji vodič ćemo spojiti pozitivni, a na koji negativni izlaz modula.



*Slika 13. Raspored komponenti na poklopcu kante*

Na samom vrhu prednje stranice kante sam sa unutrašnje strane postavila komad metala u obliku „L profila“, a Solenoid bravu sam postavila na unutrašnju stranu poklopca kante. Brava je pozicionirana na način da joj u zaključanom stanju taj metalni komad čini prepreku i onemogućava podizanje samog poklopca, dok se za otključavanje uvlači pomični dio brave te se kanta može otvoriti. Na vanjskoj strani poklopca kante postavljen je solarni panel kojim napajamo samu bateriju te RFID čitač i LCD ekran kako bi korisnici imali što jednostavniji pristup za prijavu RFID karticom i očitavanje podataka sa ekrana. Na slici 13. vidljiv je raspored svih komponenti sa vanjske strane poklopca kante.

## 5.4. Spajanje Arduino kompatibilnog mos driving modula i Arduino pločice

Prilikom izrade ovog projekta bilo je nužno kontrolirati rad trošila, odnosno brave, pri većoj struji nego koju koristi sam Arduino, a da pri tome ne riskiram kvar Arduina i ostalih komponenti. Prvotno sam koristila modul s relejom pomoću kojeg manjim naponom kontroliramo protok struje većeg napona na trošilu. Međutim prilikom testiranja rada sustava, primijetila sam da se nakon nekog vremena počeo javljati problem resetiranja rada Arduino pločice, a samim time i cjelokupnog programa zbog prevelikog opterećenja releja i trošila.

Zbog toga sam se odlučila za korištenje Arduino kompatibilnog mos driving modula VMA411 (slika 14) kojim sam upravljala radom Solenoid brave. Ovaj modul ima sljedeće karakteristike:

- Radni napon: 3 do 5 VDC
- Maksimalan napon opterećenja: 0 do 24 VDC
- Maksimalna struja opterećenja: 5 A
- Ulazni pin: SIG pin
- Korišteni MOSFET tranzistor: IRF520
- Dimenzije: 34 x 21 x 16 mm



Slika 14. Arduino kompatibilni mos driving modul VMA411 [26]

Takav modul koristi se kao elektronički prekidač tamo gdje nam je potrebno pomoću određenog napona propuštati i kontrolirati protok struje različitog napona. Na sebi ima naponski upravljiv MOSFET IRF520 tranzistor koji na sebi ima tri elektrode: uvodnu, odvodnu i

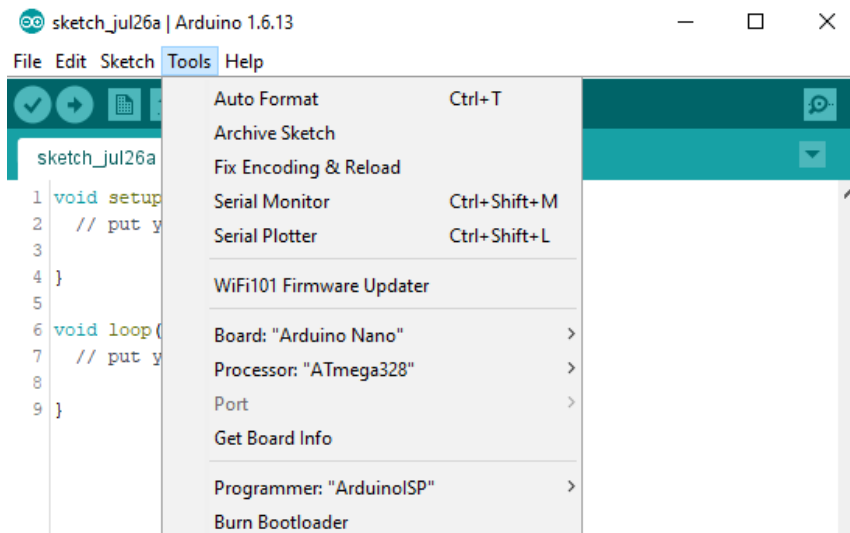
upravljačku elektrodu. Ovim tranzistorom je protok struje između uvodne i odvodne elektrode kontroliran pomoću napona koji je doveden na upravljačku elektrodu MOSFET-a. Kad nema napona između uvodne i upravljačke elektrode, otpor između uvodne i odvodne nožice je jako visok što se može gledati kao da je strujni krug otvoren, tj. da struja ne protječe strujnim krugom. Dovođenjem određenog napona između upravljačke i uvodne elektrode, smanjuje se navedeni otpor i time se omogućuje protok struje, tj. možemo reći da je strujni krug zatvoren. [19]

Na upravljačku elektrodu spojiti ćemo Arduino koji će digitalnim pinom kontrolirati protok struje između uvodne elektrode, na koju ćemo spojiti Leoch LPC12-8.0 bateriju s kapacitetom 8,0 Ah i naponom 12V, te odvodne elektrode na koju spajamo Solenoid bravu. Na ovaj način postigli smo to da naponom sa Arduino pločice kontroliramo u kojim trenucima će brava biti u otključanom, odnosno zaključanom stanju. Spajanje navedenih komponenti na modul izvela sam na sljedeći način (vidljivo i u Prilogu 1):

- V+ i V- izlaze mos modula spajamo sa dva izlaza Solenoid brave
- VIN i GND pinove mos modula spajamo sa dva izlaza baterije
- SIG pin mos modula spajamo sa D5 pinom sa Arduino pločice
- VCC pin mos modula spajamo sa 5V pinom sa Arduino pločice
- GND pin mos modula spajamo sa GND pinom sa Arduino pločice

## 6. Programska podrška

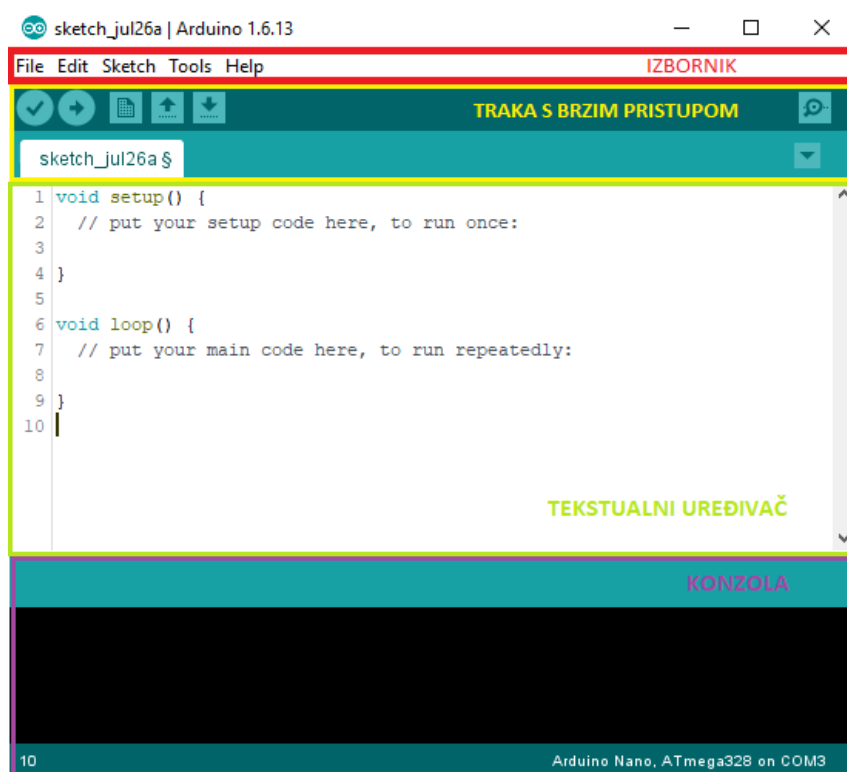
Za upravljanje Arduino pločicom koristi se softverski paket Arduino IDE koji je moguće preuzeti sa službene stranice [www.arduino.cc](http://www.arduino.cc). Prilikom izrade ovog diplomskog rada korištena je verzija Arduino 1.6.13 koju je moguće instalirati na operacijske sustave Windows, Mac OS X i Linux. Programsko okruženje Arduina bazirano je na jezicima Java i Processing te se isto može primjenjivati na svim verzijama Arduino pločica. Processing je programski alat koji omogućuje pisanje aplikacija na pojednostavljenoj verziji jezika Java koji zbog svoje jednostavnosti omogućuje da se u njemu snađu i potpuni početnici u programiranju.



Slika 15. Odabir parametara Arduino pločice u Arduino IDE sučelju

Nakon instalacije programa Arduino IDE na naše računalo, potrebno je postaviti sve parametre na vrijednosti pločice koju koristimo u radu. U izborniku „Tools“ mijenjamo vrijednosti pločice, serijskog porta, procesora i programatora (slika 15). Za vrijednost pločice odabiremo Arduino Nano iz ponuđene liste podržanih pločica, kao procesor odabiremo ATmega 328, označavamo serijski port koji je dodijeljen našoj pločici te za kraj odabiremo ArduinoISP kao vrijednost programatora. Program napisan u sučelju Arduino IDE naziva se skica (eng. *sketch*) te se u njemu definira sve što želimo da pločica napravi.

Samo sučelje programa sastoji se od četiri dijela: izbornika, trake s brzim pristupom, tekstualnog uređivača te konzole koji su prikazani na slici 16 [20]. Tekstualni uređivač čini prostor u kojem pišemo programski kod, dok nam konzola prikazuje osnovne podatke o samom programu.



Slika 16. Dijelovi sučelja programa Arduino IDE

Traka s brzim pristupom sadrži naredbe koje najčešće koristimo prilikom pisanja programa:

- „Verify“ – izgled ikone: kvačica, provjerava moguće greške u kodu i pretvara ga u niz instrukcija koje pločica razumije.
- „Upload“ – izgled ikone: strelica prema desno, provjerava moguće greške u kodu a zatim ga učitava na povezanu Arduino pločicu
- „New“ - izgled ikone: list papira, otvara novi prazan program
- „Open“ – izgled ikone: strelica prema gore, otvara izbornik za učitavanje postojećih programa



- „Save“ – izgled ikone: strelica prema dolje, sprema trenutni program u mapu „Sketchbook“
- „Serial Monitor“ – izgled ikone: povećalo, otvara prozor serijskog monitora.

U Arduino programskom okruženju koriste se biblioteke (eng. *libraries*) koje nam omogućuju korištenje dodatnih funkcionalnosti u skicama, npr. funkcionalnosti povezane s radom određenog elementa. Kako bi unutar skice koristili određenu biblioteku potrebno ju je odabrati iz izbornika “Sketch” uz pomoć opcije “Include Library” ili na vrh skice navesti znak “#” te nakon njega ime biblioteke. S obzirom na to da je za upravljanje radom RFID čitača i LCD ekrana potreban niz dodatnih funkcija, unutar skice je bilo potrebno pozvati i njihove biblioteke. Za korištenje odabranog RFID čitača potrebne su biblioteke SPI.h i MFRC522.h, a za LCD ekran sa njegovim IIC adapterom potrebne su biblioteke Wire.h i LiquidCrystal\_I2C.h koje sam pozvala na sljedeći način:

```
//biblioteke RFID citaca
#include <SPI.h>
#include <MFRC522.h>

//biblioteke lcd ekrana
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
```

Iako imamo različite načine pisanja koda, svaka skica za Arduino pločice mora se sastojati od najmanje dva dijela, „setup()“ i „loop()“ metode, koje se automatski stvaraju prilikom otvaranja svake nove skice. „Setup()“ blok naredbi izvršava se samo jednom, odmah nakon što se programski kod pokrene na Arduino pločici. Unutar njega se inicijaliziraju varijable, pokreće serijska komunikacija, određuje „PinMode“ itd. Nakon što se jednom izvrši sve iz „setup()“ bloka naredbi, program prelazi na „loop()“ blok. Sam naziv metode govori nam da se naredbe unutar nje izvršavaju kao u petlji, nakon što se redom izvrši sve do posljednje naredbe unutar bloka, ponovno se vraćamo na početak petlje te se blok izvršava iznova sve dok je Arduino uključen.

Sav programski kod nalazi se u Prilogu 2, a u nastavku ću detaljnije objasniti sve metode koje se u njemu nalaze. Unutar „setup()“ metode inicijalizirala sam serijsku komunikaciju te rad RFID čitača. Također, odredila sam karakteristike ispisa na LCD ekranu te postavila kursor na

poziciju za ispis prvog znaka. Solenoid bravu sam postavila u zaključano stanje te sam postavila digitalni pin 5 u izlazni pin kojim će se kasnije upravljati mos modulom, odnosno otključavanjem brave. Unutar metode „loop()“ se za početak vrši provjera da li RFID čitač očitava prisutnu novu karticu sa kojom je moguće serijski komunicirati. Ukoliko to nije slučaj, uz pomoć naredbe „return“ se izvršavanje „loop()“ metode pokreće od početka. Na taj način osiguravamo da RFID čitač bez prekida vrši očitavanje i provjere prisutnosti novih kartica.

```

Scan PICC to see UID, SAK, type, and data blocks...
Card UID: F0 63 FD 73
Card SAK: 08
PICC type: MIFARE 1KB

Sector Block  0 1 2 3  4 5 6 7  8 9 10 11 12 13 14 15 AccessBits
15   63 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF [ 0 0 1 ]
     62 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
     61 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
     60 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]

14   59 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF [ 0 0 1 ]
     58 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
     57 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
     56 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]

13   55 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF [ 0 0 1 ]
     54 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
     53 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
     52 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]

12   51 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF [ 0 0 1 ]
     50 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
     49 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]

```

*Slika 17. Prikaz dijela strukture podataka koji su pohranjeni na RFID karticama*

U slučajevima kada RFID čitač očita novu karticu, unutar „loop()“ metode poziva se novu metodu „pristup\_poznatom\_karticom()“. Kada se približi kartica, čitač očitava 1024 bajta informacija koji su podijeljeni u 16 sektora i 4 bloka (slika 17). Uz to, svaka kartica ima svoj jedinstveni UID broj koji se sastoji od četiri heksadekadska broja te ga u ovoj metodi pohranjujemo u varijablu „ocitano“.

Nakon toga, unutar „for“ petlje uspoređujemo UID trenutno očitane kartice sa onima koji su pohranjeni u varijabli „kartice\_administratori“ i u nizu „odobreni\_UID“. Ukoliko trenutna očitana kartica odgovara administratorskoj kartici, te ukoliko je kanta napunjena, na ekran će se

ispisati potvrda o pražnjenju kante. Ona će se otključati uz pomoć mos modula kako bi je mogli isprazniti te se resetira stanje ukupne težine u njoj postavljajući varijablu „ukupno“ na vrijednost nule. Kada kanta nije napunjena, očitavanjem administratorske kartice na LCD ekranu će se ispisati poruka da je kanta prazna.

U slučajevima kada kanta nije puna, očitano karticu uspoređujemo sa onima koje su pohranjene u nizu „odobreni\_UID“. Onda kada je očitana kartica jedna od kartica čiji je UID odobren za pristup kanti, na ekran se ispisuje poruka o odobrenom pristupu te se kanti postavljanjem digitalnog pina koji upravlja mos modulom u 1 šalje signal za otključavanje brave. Nakon što korisnik u kantu postavi svoj otpad, vrši se očitavanje težine koju je on ubacio te izračun korisnikove potrošnje o čemu će više u svom radu objasniti kolegica Barnjak. Ako očitana kartica ne pripada tom nizu, te nije administratorska kartica, prilikom njenog očitavanja se povećava vrijednost varijable „brojac“. Na kraju „for“ petlje dolazimo do jedinog preostalog slučaja. Usporedbom vrijednosti „brojaca“ i varijable „velicina\_niza“ koja označava broj odobrenih kartica dolazimo do slučaja kada očitana kartica nije jednaka niti jednoj od kartica koje su odobrene te se tada korisniku ispisuje poruka o odbijenom pristupu i zaključava kanta. Kako se metoda „pristup\_poznatom\_karticom()“ poziva svaki put kada RFID čitač očita prisutnost nove kartice prilikom izvršavanja „loop()“ metode, za svaku karticu koju korisnik približi našem čitaču će se izvršiti provjera koje radnje je potrebno izvršiti, ovisno o njenom UID-u i stanju popunjenosti kante.

## 7. Zaključak

Prilikom izrade ovog diplomskog rada proučavala sam probleme odlaganja otpada u suvremenom svijetu te izračun komunalne naknade za građane. Zajedno sa kolegicom Antonelom Barnjak izradila sam prototip pametnog sustava za naplatu usluga sakupljanja i odvoza otpada koji svakom korisniku u realnom vremenu omogućuje uvid u njegovu potrošnju. Problem pravednijeg izračuna naknade za odvoz otpada riješile smo na način da se svaki korisnik prijavljuje putem RFID sustava za identifikaciju prilikom odlaganja otpada te mu se pri tome izračunava težina otpada koju ubacuje u kantu i na tome temelji njegov mjesečni račun.



*Slika 18. Izgled gotovog prototipa sa RFID sustavom za identifikaciju korisnika*

Projekt se temelji na korištenju Arduino Nano pločice koja je lako dostupna i jednostavna za rad pa se njom mogu koristiti i potpuni početnici. Za identifikaciju korisnika koristila sam RFID – RC522 čitač sa pripadajućim karticama za kontrolu pristupa. Uz to koristila sam još i sljedeće komponente: Solenoid bravu za otključavanje kante čijim radom je upravljao Arduino kompatibilni mos modul te LCD ekran sa IIC adapterom za ispis potrebnih podataka. U prethodnim poglavljima je detaljno objašnjeno na koji način spajamo sve komponente te kako ih

povezujemo programskim kodom da bi izvršavale zadane radnje, a na slici 18. je prikazano kako sustav finalno izgleda.

Ovaj prototip pruža dobru osnovu za proširenja i nadogradnje u budućnosti. Moguće ga je i nadograđivati na način da se kante međusobno povežu u zajednički sustav koji bi imao integriranu bazu podataka te bi korisnici imali pristup odlaganju otpada na različitim lokacijama, ali sa jedinstvenim izračunom potrošnje. Također, na taj način bi bilo jednostavno korisniku omogućiti uvid u njegovu potrošnju u bilo kojem trenutku i na bilo kojem mjestu putem mobilne ili web aplikacije koja bi bila povezana sa istom bazom podataka. U takvu aplikaciju bi se mogao integrirati i dojavni sustav koji obavještava službu za gospodarenje otpadom o razini popunjenosti pojedinačne kante u svakom trenutku te njenoj potrebi za pražnjenjem. U budućnosti bi bilo potrebno sustav izraditi od otpornijih materijala koji mogu izdržati različite vanjske uvjete i promjene temperature te tako da su sve komponente zaštićene unutar okvira kante, gdje im korisnici fizički ne mogu pristupiti niti ih oštetiti prilikom rukovanja.

Danas se javlja sve veća potreba za ovakvim sustavima sa što višom razinom automatizacije procesa rada. Unatoč tome što postoje razne mogućnosti za nadogradnju i poboljšanja u budućnosti, ipak smatram da smo osmislile kvalitetan i cjenovno prihvatljiv autonoman sustav koji zadovoljava današnje potrebe korisnika za pravednijim načinom naplate odvoza otpada.

# Literatura

- [1] Arduino introduction, <https://www.arduino.cc/en/Guide/Introduction>
- [2] Upoznajte Arduino, <http://www.poslovnih.hr/blog/elektronika-za-pocetnike-upoznajte-arduino-1313>
- [3] Croduino, <https://e-radionica.com/hr/blog/2015/10/08/sto-je-arduino-i-croduino/>
- [4] Arduino štitovi, <https://learn.sparkfun.com/tutorials/arduino-shields>
- [5] Podjela Arduino pločica, <https://www.arduino.cc/en/Main/Products>
- [6] Mikrokontroler ATmega 328, <https://www.microchip.com/wwwproducts/en/ATmega328>
- [7] Usporedba Arduino pločica, <https://core-electronics.com.au/tutorials/compare-arduino-boards.html#nano>
- [8] Pinovi pločice Arduino Nano, <http://www.circuitstoday.com/arduino-nano-tutorial-pinout-schematics>
- [9] Specifikacije Arduino Nano pločice, <https://store.arduino.cc/arduino-nano>
- [10] Automatska identifikacija: RFID, <http://www.marco.hr/tehnologije/tehnologije-RFID.htm>
- [11] CARNet CERT u suradnji s LS&S: „RFID identifikacija“, CARNet, 2007
- [12] RFID sustav, <http://www.technovelgy.com/ct/Technology-Article.asp?ArtNum=2>
- [13] Funkcioniranje RFID sustava, <http://materijali.grf.unizg.hr/media/RFID%20tehnologija.pdf>
- [14] Krunoslav Žubrinić: „Korištenje sustava za radiofrekvencijsku identifikaciju u poslovanju“, LAUS CC, 2004.
- [15] RFID frekvencije, <http://www.radio-electronics.com/info/wireless/radio-frequency-identification-rfid/low-high-frequency-bands-frequencies.php>
- [16] Roy Want: „An Introduction to RFID technology“, IEEE CS and IEEE ComSoc, 2006

- [17] Spajanje RFID čitača, <https://e-radionica.com/hr/blog/2018/07/26/kkm-rfid-reader-mfrc-522/>
- [18] Solenoid brava, <https://www.adafruit.com/product/1512>
- [19] MOSFET tranzistor, <https://e-radionica.com/hr/blog/2017/12/10/sto-je-to-mosfet/>
- [20] Arduino IDE sučelje, <https://www.arduino.cc/en/Guide/Environment>
- [21] Usporedba veličina Arduino pločica, <https://www.quora.com/What-is-the-difference-between-the-Arduino-Nano-and-the-Arduino-Uno>
- [22] Ilustracija RFID sustava, <https://www.epc-rfid.info/rfid>
- [23] RFID etiketa, <https://www.robotshop.com/ca/en/seedstudio-125khz-rfid-tags.html>
- [24] RFID naljepnica, [https://www.123rf.com/photo\\_27517580\\_rfid-sticker.html](https://www.123rf.com/photo_27517580_rfid-sticker.html)
- [25] RFID tiskana pločica, <https://libstock.mikroe.com/projects/view/739/rfid-click-example>
- [26] Arduino modul VMA411, <https://www.velleman.eu/products/view/?id=435580>

## Popis slika i tablica

### Popis slika:

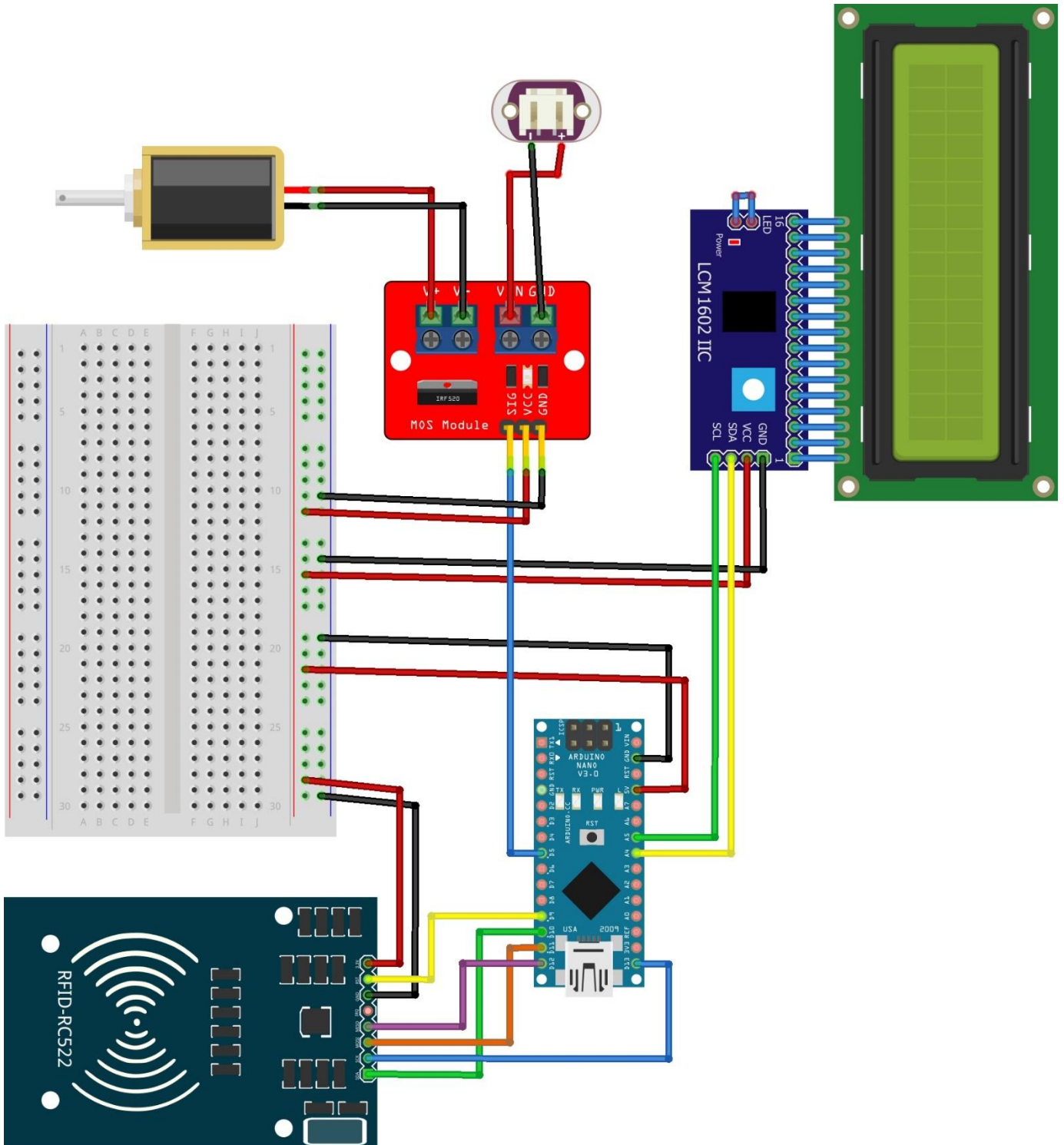
Slika 1. Arduino Nano pločica [9] .....	5
Slika 2. Usporedba veličine pločice Arduino Uno (lijevo) i Arduino Nano (desno) [21] .....	6
Slika 3. Pinovi Arduino Nano pločice [8].....	7
Slika 4. Ilustracija RFID sustava [22].....	9
Slika 5. RFID transponder [13].....	11
Slika 6. Fizičke izvedbe transpondera: a) etikete, b) naljepnica i c) tiskana pločica [23]-[25] ....	13
Slika 7. Nokia 3200 s ugrađenim NFC čitačem na stražnjoj strani kućišta [16] .....	20
Slika 8. Konstrukcija kante .....	22
Slika 9. Shema spajanja LCD ekrana, IIC adaptera i Arduina Nano .....	24
Slika 10. RFID – RC522 čitač i njegove kartice za kontrolu pristupa [17].....	25
Slika 11. Shema spajanja RFID – RC522 čitača i Arduina Nano.....	26
Slika 12. Solenoid brava u aktivnom stanju [18].....	27
Slika 13. Raspored komponenti na poklopcu kante.....	28
Slika 14. Arduino kompatibilni mos driving modul VMA411 [26].....	29
Slika 15. Odabir parametara Arduino pločice u Arduino IDE sučelju .....	31
Slika 16. Dijelovi sučelja programa Arduino IDE.....	32
Slika 17. Prikaz dijela strukture podataka koji su pohranjeni na RFID karticama .....	34
Slika 18. Izgled gotovog prototipa sa RFID sustavom za identifikaciju korisnika .....	36

### Popis tablica:

Tablica 1. Tehničke specifikacije pločice Arduino Nano [9] .....	8
Tablica 2. Podjela RFID uređaja na klase [11].....	18
Tablica 3. Generacije RFID čipova [11].....	18



# Prilog 1 - Shema spajanja svih korištenih komponenti sa Arduinom Nano



## Prilog 2 – Programski kod

```
/* * DIPLOMSKI RAD
 */
//biblioteke RFID citaca
#include <SPI.h>
#include <MFRC522.h>

#include <HX711.h> //biblioteka analogno-digitalnog pretvornika

//biblioteke lcd monitora
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);
//povezivanje lcd monitora i IIC adaptera
HX711 scale (A0,A1);

//varijable za senzor težine
long prosjek = 0;
float val = 0;
long vrijednost = 0;
int prije = 0;

//varijabla za pohranu UID kartice
String ocitano = "";

//varijable za ultrazvucni senzor
const int trigPin = 6;
const int echoPin = 7;
long vrijeme;
float udaljenost = 16; //kanta je inicijalno prazna

#define SS_PIN 10
#define RST_PIN 9

#define mos_modul 5

MFRC522 mfrc522(SS_PIN, RST_PIN); // kreiraj MFRC522 instancu
#define velicina_niza 2
String odobreni_UID[velicina_niza] = {"F0 63 FD 73", "D0 67 99 7C"};
//UID kartica sa odobrenim pristupom
String kartice_administratori = "D0 67 99 7C"; //UID administratorske
kartice

int ukupno = 0; //ukupna težina kante
int korisnik_ukupnaTezina[velicina_niza]; // = {0,0};
int korisnik_dodanaTezina[velicina_niza]; // = {0,0};
float potrosnja[velicina_niza]; // = 0;

bool otkljucavanje = false;
```

```

void setup() {
  pinMode(trigPin, OUTPUT); // postavlja trigPin kao izlaz
  pinMode(echoPin, INPUT); // postavlja echoPin kao ulaz

  Serial.begin(9600); //inicijaliziraj serijsku komunikaciju

  digitalWrite(trigPin, HIGH); //slanje trig signala
  delayMicroseconds(10); //odaslani trig signal traje 10 µs = 0.1
ms
  digitalWrite(trigPin, LOW); //prestanak slanja trig signala

  vrijeme = pulseIn(echoPin, HIGH); //Primanje reflektiranog trig signala,
u mikrosekundama
  udaljenost = (vrijeme*0.034) / 2; //Racunanje udaljenosti senzora od
predmeta, u centimetrima
//udaljenost = (vrijeme putovanja signala *
brzina zvuka kroz zrak) / 2
  delay(100);
  Serial.println(udaljenost);

  lcd.clear();
  lcd.begin(16,2); //ispis 16 znakova u 2 reda
  lcd.setCursor(0,0); //postavljanje kursora, tj.prvog znaka na prvo
mjesto prvog reda

  SPI.begin(); //inicijaliziraj SPI sabirnicu
  mfrc522.PCD_Init(); //pokreni MFRC522

  digitalWrite(mos_modul,0); //brava je početno zaključana

  lcd.print("Mjerenje tezine");
  delay(1000);
  kalibracija(); //metoda za kalibraciju senzora težine

  lcd.clear();
  lcd.print("Ocitajte karticu");
  pinMode(mos_modul, OUTPUT); //pin mos modula (pin 5) postavljamo u
izlazni pin
}

void loop() {

  // Provjere - izadi iz programa ako nema prisutne nove kartice
  //ili ako nije moguće serijski komunicirati sa njom
  if (!mfrc522.PICC_IsNewCardPresent())
  { return;}
  if (!mfrc522.PICC_ReadCardSerial())
  { return;}

  popunjenost_kante();
  ocitano ="";
  pristup_poznatom_karticom();
}

```

```

lcd.clear();
lcd.print("Ocitajte karticu");

otkljucavanje = false;
digitalWrite(mos_modul,0); //pošalji signal za zaključavanje kante
delay(2000);
}

//pristup korisnika sa karticom koju vec imamo pohranjenu (provjera preko
UID)
void pristup_poznatom_karticom()
{
    int brojac = 0;
    lcd.setCursor(0,0);
    lcd.print ("UID kartice:");
    byte znak;

    //petlja kojom se dolazi do UID broja kartice -> pohranjuje se u
varijablu ocitano
    for (byte i = 0; i < mfrc522.uid.size; i++)//i (od 0 do 3) za 4 bloka u
kojima je pohranjen UID
    {
        ocitano.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));
        ocitano.concat(String(mfrc522.uid.uidByte[i], HEX));
        lcd.print(mfrc522.uid.uidByte[i], HEX); //ispis UID broja kartice
koju je trenutno ocitao
    }
    ocitano.toUpperCase();
    ocitano.remove(0,1);

    //prema ocitanoj kartici izvrši potrebne radnje
    for (int i = 0; i < velicina_niza; i++)
    {
        //ako je kanta puna, cekaj administratorovo praznjenje
        if (udaljenost < 15 && ocitano == kartice_administratori)
        {
            //administrator vrši samo praznjenje kante u slucaju kad je ona
napunjena
            //ako je ocitani UID kartice jednak kartici administratora
            //i kanta je napunjena otključaj kantu i resetiraj ukupno stanje
            if (ukupno > 0){
                lcd.clear();
                lcd.print("Administrator");
                otkljucavanje = true;
                digitalWrite(mos_modul,1); //pošalji signal za otključavanje
kante
                lcd.setCursor(0,1);
                lcd.print("Praznjenje kante");
                delay(1000);
                ukupno = 0; //resetiranje tezine cijele kante - kanta je
ispraznjena
                prije = 0;
                lcd.clear();
            }
        }
    }
}

```

```

        lcd.print(String("Ukupna tezina:"));
        lcd.setCursor(0,1);
        lcd.print(String(float(ukupno/1000)) + String("kg."));
        delay(2000);
    }
    else{
        lcd.clear();
        lcd.print("Administrator");
        lcd.setCursor(0,1);
        lcd.print("Kanta je prazna");
        delay(2000);
    }
}

//ako kanta nije puna, provjeravaj ocitane kartice
else if (udaljenost >= 15 && ocitano == odobreni_UID[i])
{
    lcd.clear();
    lcd.print("Odobren pristup");
    otkljucavanje = true;
    digitalWrite(mos_modul,1); //pošalji signal za otključavanje
kante
    delay(1000);

    vrijednost = scale.get_value(1);
    ukupno=((vrijednost-prosjek)/val)-2*((vrijednost-
prosjek)/val));

    //TEZINE ZA CIJELU KANTU
    lcd.clear();
    lcd.print("Ukupna tezina");
    lcd.setCursor(0,1);
    lcd.print(String(ukupno) + "g.");
    delay(2000);

    int ubaceno = ukupno - prije;
    prije = ukupno;

    //TEZINE POJEDINACNO PO KORISNIKU
    lcd.clear();
    lcd.print("Korisnik " + String(i+1));
    delay(2000);
    korisnik_dodanaTezina[i] = ubaceno;
    korisnik_ukupnaTezina[i] = korisnik_ukupnaTezina[i] + ubaceno;

    lcd.clear();
    lcd.print("Ukupni otpad");
    lcd.setCursor(0,1);
    lcd.print("korisnika:" + String(korisnik_ukupnaTezina[i]) + "g");
    delay(2000);
    lcd.setCursor(0,0);
    lcd.print("Dodani otpad");
    lcd.setCursor(0,1);

```

```

lcd.print("korisnika:" + String(korisnik_dodanaTezina[i]) +
"g");
delay(2000);
lcd.clear();

//NAPLATA POTROSNJE - 1 kg korisnikove tezine naplacuje se 1.50
kn
if (korisnik_ukupnaTezina[i] <= 0)
{
    potrosnja[i] = 0;
}
else
{
    potrosnja[i] = korisnik_ukupnaTezina[i] * 0.0015;
}
lcd.setCursor(0,0);
lcd.print("Potrosnja: ");
lcd.setCursor(0,1);
lcd.print(String(float(potrosnja[i])) + "kn");
delay(3000);
return;
}
else if (ocitano != odobreni_UID[i]) //ako ocitana kartica nije
jedna od odobrenih povecaj brojac
{
    brojac = brojac + 1;
}
}

//ako se ocitana kartica ne podudara sa niti jednom karticom
//iz niza odobrenih kartica brojac ce biti jednak velicini niza
if (brojac == velicina_niza)
{
    lcd.clear();
    lcd.println("Odbijen pristup");
    otkljucavanje = false;
    digitalWrite(mos_modul,0); //pošalji signal za zaključavanje
kante
    delay(2000);
}
}

void kalibracija(){
    lcd.clear();
    lcd.print("Molim pricekajte");
    lcd.setCursor(0,1);
    lcd.print("kalibraciju vage");
    prosjek = scale.read_average(70); //spremanje prosječne vrijednosti iz
niza od 70 očitavanja
    vrijednost = scale.get_value(1); //trenutno očitana vrijednost
    vrijednost = prosjek - vrijednost; //izračun početnog stanje vage

    lcd.clear();

```

```

lcd.print("Postavite 100 g");
delay(3000);
for(int i=0;i<70;i++)
{
    vrijednost = scale.get_value(1);
    val += prosjek - vrijednost;
}

val = val/100.0;
val = val/100.0;
}

void popunjenost_kante(){
    if (otkljucavanje == false)
    {
        digitalWrite(trigPin, HIGH); //slanje trig signala
        delayMicroseconds(10);      //odaslani trig signal traje 10 µs = 0.1
ms
        digitalWrite(trigPin, LOW); //prestanak slanja trig signala

        vrijeme = pulseIn(echoPin, HIGH); // Primanje reflektiranog trig
signala, u mikrosekundama
        udaljenost = (vrijeme*0.034) / 2; // Racunanje udaljenosti senzora od
predmeta, u centimetrima
//udaljenost = (vrijeme putovanja
signala * brzina zvuka kroz zrak) / 2
        delay(100);
        Serial.println(udaljenost);

        if (udaljenost < 15){
            lcd.clear();
            lcd.print("Kanta je puna");
            delay(1000);
            lcd.clear();
            lcd.print("Cekanje");
            lcd.setCursor(0,1);
            lcd.print("praznjenja");//praznjenje kante vrsi samo administrator
unutar metode pristup_poznatom_karticom()
            delay(2000);
        }
    }
}
}

```