Pčelarska SMS vaga

Schneider, Sebastian

Master's thesis / Diplomski rad

2019

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:200:271012

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-27**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

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

Sveučilišni studij

Pčelarska SMS vaga

Diplomski rad

Sebastian Schneider

Osijek, 2019

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Obrazac D1: Obrazac za imenovanje Povjerenstva za obranu diplomskog rada

Osijek, 19.09.2019.

Odboru za završne i diplomske ispite

Imenovanje Povjerenstva za obranu diplomskog rada

Ime i prezime studenta:	Sebastian Schneider
Studij, smjer:	Diplomski sveučilišni studij Elektrotehnika, smjer Komunikacije i informatika'
Mat. br. studenta, godina upisa:	D-1086, 19.09.2018.
OIB studenta:	32520755842
Mentor:	Izv. prof. dr. sc. Davor Vinko
Sumentor:	
Sumentor iz tvrtke:	
Predsjednik Povjerenstva:	Izv. prof. dr. sc. Vanja Mandrić-Radivojević
Član Povjerenstva:	Luka Filipović
Naslov diplomskog rada:	Pčelarska SMS vaga
Znanstvena grana rada:	**Elektronika (zn. polje elektrotehnika)**
Zadatak diplomskog rada:	Zadatak diplomskog rada je razviti pčelarsku SMS vagu koristeći Atmel studio ili Arduino IDE platformu za realizaciju. Pčelarska SMS vaga treba periodički (svakih 24h) slati SMS poruku s podacima o vlažnosti i temperaturi zraka te masu same košnice. Isti podatke je na zahtjev korisnika potrebno prikazati na LCD zaslonu vage. Za više informacija javiti se mentoru: davor.vinko@ferit.hr Temu rezervirao Sebastian Schneider
Prijedlog ocjene pismenog dijela ispita (diplomskog 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: 2 bod/boda Jasnoća pismenog izražavanja: 3 bod/boda Razina samostalnosti: 3 razina
Datum prijedloga ocjene mentora:	19.09.2019.

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, 25.09.2019.

Ime i prezime studenta:	Sebastian Schneider
Studij:	Diplomski sveučilišni studij Elektrotehnika, smjer Komunikacije i informatika'
Mat. br. studenta, godina upisa:	D-1086, 19.09.2018.
Ephorus podudaranje [%]:	24%

Ovom izjavom izjavljujem da je rad pod nazivom**: Pčelarska SMS vaga**

izrađen pod vodstvom mentora Izv. prof. dr. sc. Davor Vinko

i sumentora

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 diplomskog rada 2

2. PČELARSTVO 3

2.1. Proizvodi pčele 4

2.2. Pčelarski alat i oprema 5

2.3. Pčelarska vaga 5

3. DIZAJN SKLOPA 7

3.1 Opis komponenti 7

3.1.1 Arduino 7

3.1.2. Senzor temperature i vlage zraka - DHT22 8

3.1.3. Senzor za mjerenje mase – HX711 pretvornik i Loadcell 9

3.1.4. Ekran – LCD 16x2 Display 11

3.1.5. I2C Adapter za LCD 16x2 ekran 11

3.1.6. Stvarno-vremenski sat – DS3231 RTC (engl. *Real-Time-Clock*) 12

3.1.7. GPRS/GSM Shield 13

3.2 Izrada sheme sklopa 17

4. IZRADA PROGRAMSKOG KODA 18

4.1 Primjer koda za mjerenje temperature i vlage (DHT22) 18

4.2 Primjer koda za mjerenje mase 20

4.3 Primjer koda za slanje SMS poruke (GPRS/GSM, AT komande) 22

4.4 Izrada vlastitog programskog koda 25

5. IZRADA PČELARSKE SMS VAGE 30

6. ZAKLJUČAK 35

LITERATURA 36

POPIS I OPIS UPOTREBLJENIH KRATICA 37

SAŽETAK 38

ŽIVOTOPIS 40

PRILOZI 41

1. UVOD

U većini slučajeva pčelari nisu u mogućnosti provoditi većinu svog vremena pokraj pčelinjaka kako bi znali što se s pčelinjakom događa. Jedan od načina kojim bi mogli pratiti što se s pčelinjakom događa je korištenje pčelarske SMS vage. Pčelarska SMS vaga omogućila bi slanje informacija neovisno o lokaciji pčelara i pčelinjaka. Kupovina SMS vagu drastično bi se smanjili troškovi putovanja pčelara. Korištenjem pčelarske SMS vage košnica se ne mora otvoriti kako bi se znalo da li je puna, tj. koliki je prinos meda. Mjerenjem mase zaključuje se da li je košnica puna. Prestankom rasta mase zaključuje se da je ispaša loša te na osnovu toga premješta se pčelinjak na novu lokaciju. Inače se stavljaju sve košnice na jednu ili više lokacija. Iz tog razloga najbolje je imati jednu pčelarsku SMS vagu na jedan pčelinjak.

U ovom diplomskom radu potrebno je bilo opisati potrebu i uporabu pčelarske SMS vage te ju realizirati preko Atmel studio ili Arduino IDE (engl. Integrated Development Environment) platforme. Pčelarska SMS vaga treba periodički (svakih 24h) slati SMS poruku s podacima o vlažnosti i temperaturi zraka te masu same košnice. Također ti podaci trebaju se ispisati na LCD display ako korisnik želi saznati podatke na mjestu gdje je vaga postavljena. Nakon izrade pčelarske vage trebaju se napraviti testna mjerenja mase kako bi se mogla utvrditi točnost same vage. Proces izrade pčelarske SMS vage bilo je potrebno dokumentirati.

Prilikom izrade pčelarske SMS vage koristila se isključivo Arduino IDE platforma iz razloga postojećih biblioteka za potrebne senzore te jednostavnosti.

U drugom poglavlju govori se o osnovama pčelarstva, pčelarskim proizvodima, osnovnoj opremi i priboru u koje pripada pčelarska vaga te je opisana njena primjena. Zatim u trećem poglavlju opisane su karakteristike i informacije komponente korištene pri izradi pčelarske SMS vage. U četvrtom poglavlju opisani su korišteni programski kodovi i napravljen je vlastiti programski kod. U posljednjem poglavlju opisan je postupak izrade same pčelarske SMS vage.

1.1 Zadatak diplomskog rada

Zadatak diplomskog rada je razviti pčelarsku SMS vagu koristeći Atmel studio ili Arduino IDE platformu za realizaciju. Pčelarska SMS vaga treba periodički (svakih 24h) slati SMS poruku s podacima o vlažnosti i temperaturi zraka te masu same košnice. Iste podatke je na zahtjev korisnika potrebno prikazati na LCD zaslonu vage.

2. PČELARSTVO

Pčelarstvo je grana stočarstva koja se bavi uzgojem pčela s ciljem dobivanja njihovog proizvoda te oprašivanjem biljaka. Pčelari su osobe koje se bave pčelarstvom i najčešće ne ovise o svom zemljištu te često niti nemaju svoje zemljište. Ukoliko ga imaju tada je to zemljište vrlo malo. Pčelinjak je mjesto na kojem se nalaze košnice. Postoje dvije vrste pčelinjaka:

1. Pokretni pčelinjaci – košnice se nalaze unutar preuređenog kamiona, prikolice ili kontejnera. Pčelinjak se premješta s jedne na drugu lokaciju.
2. Stacionarni pčelinjaci – košnice su postavljene na jednu lokaciju unutra odgovarajućih kućica.

Sl. 1.1. Pokretni pčelinjak

2.1. Proizvodi pčele

Pod pčelinje proizvode spadaju: med, pelud, matična mliječ, propolis, pčelinji otrov i pčelinji vosak.

Med je gust i sladak sok koji proizvodi pčela medarica od nektara koji sakuplja s cvjetovima biljaka. Pčele medarice zatim ga spremaju u sače gdje med sazrijeva. Nakon sazrijevanja med se vrca. Vrcanje je proces u kojem se med odvaja iz saća. Sastav meda se razlikuje ovisno o cvjetovima od kojih pčela skuplja nektar. Osnovni sastojci su identični kod svih vrsta meda: aminokiseline, minerali, visoko vrijedne organske kiseline kao jabučna, mravlja, limunska, jantarna, octena kiselina, razni derivati klorofila, pigmenti, inulin, vosak te elementi vitamina B [1].

Sl.2.1. Med [1]

Matična mliječ je ljepljiva, kremasta tvar bijele boje koju luče pčele medarice i koristi se kao hrana ličinkama u koloniji.

Propolis je smjesa smeđe boje koja se koristi kao zakrpa pukotinama na košnici. Pčela propolis prikuplja s biljnih sokova, pupoljaka stabala ili drugih prirodnih izvora. Propolis se koristi u medicinske svrhe jer djeluje protiv bakterija, virusa, upala i gljivica te jača imunološki sustav.

Pčelinji otrov je izlučevina žalca pčele te pripada u najjače biološke otrove. Nalazi svoju korist u medicini i koristi se u malim količinama. Koristi se u borbi velikog broja bolesti [2].

Sl.2.2. Propolis [2]

Pčelinji vosak je prirodni vosak koje pčele koriste kako bi izgradile saće u koje pohranjuju med.

2.2. Pčelarski alat i oprema

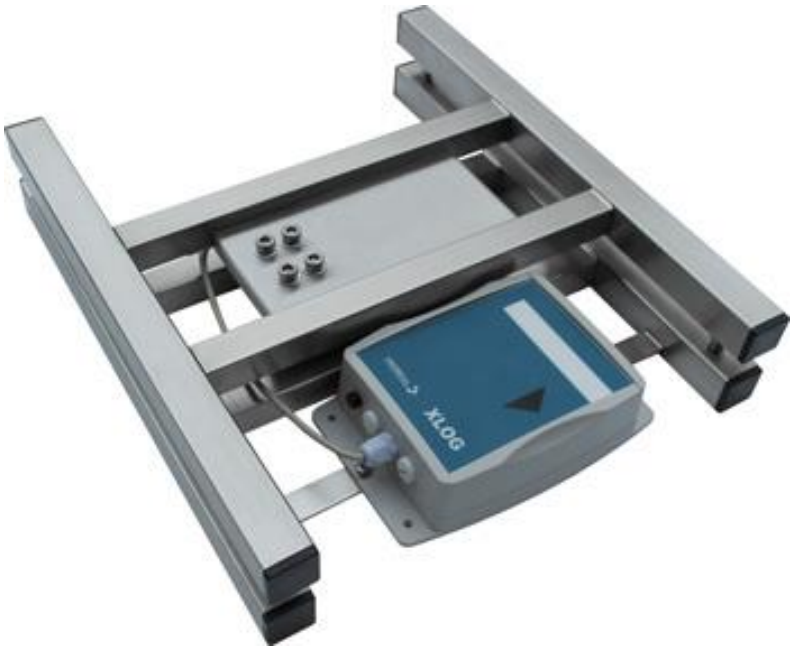
Osnovni alat i oprema koju bi pčelar trebao imati kako bi mogao raditi s pčelama je: pčelarske rukavice, pčelarsko odijelo, pčelarski šešir, gljiva goba (sredstvo za dimljenje), dimilica za pčele, dlijeto pčelarsko, kliješta za vađenje okvira (saća) iz košnice, vrcalica, pčelarska vilica, posuda sa sitom za vrcalicu i pčelarska vaga.

2.3. Pčelarska vaga

Pčelarska vaga dio je pčelarske opreme koja služi prikazivanju stanja u košnici i izvan nje. Na pčelarsku vagu postavlja se košnica u kojoj se nalazi nešto veća populacija pčela od preostalih košnica u pčelinjaku. Na taj način će dobiveni podaci biti najkorisniji za pregled stanja na cijeli pčelinjak. U današnje vrijeme najčešće se koriste elektroničke ili digitalne vage radi svoje jednostavnosti i jednostavnog očitanja te se također koriste i standardne mehaničke vage.

Pčelarsku vagu korisno je imati u pčelinjaku tijekom cijele godine. Pomoću vage može se provjeriti prinos meda bez otvaranja košnice. Pčelarskom vagom najbolje se može utvrditi kada se smanjuje ispaša ili kada potpuno prestane što govori da je vrijeme za promjenu lokacije pčelinjaka. Smanjivanje ili povećanje mase ne pokazuje nužno odraz trenutne ispaše ili potrošnje zalihe hrane već utjecaj na izgradnju saća, povećanje broja pčela i razvoj legla. Kako bi se dobile najispravnije informacije o masi košnice potrebno je masu mjeriti u isto vrijeme dana, što je najčešće u večernjim satima kada se sve pčele vrate u košnicu.

Suvremeni pčelari sve više koriste električne pčelarske vage koja ima mogućnost slanja SMS poruke. Ovisno o proizvođaču mogu se nadodati drugi dodaci: senzor temperature, senzor za temperaturu unutrašnjosti košnice, senzor za vagu zraka, mogućnost spajanja na električnog pastira, senzor kiše, senzor za mjerenja jačine vjetra, magnetni kontakt alarm, žiroskop – detektor pokreta.



Sl.2.3. GPRS/SMS Pčelarska vaga XLOG bee [3]

Pčelari nisu uvijek u mogućnosti svaki dan boraviti u pčelinjaku te im je zbog toga potrebna pčelarska SMS vaga pomoću koje će dobivati informacije iz pčelinjaka. To je jako korisno, ali ne i jeftino rješenje. Pčelinjak se u nekim slučajevima nalazi po par stotina kilometara od pčelara, no ipak informacije o pčelinjaku dolaze na mobilni uređaj pčelara te po njima može vidjeti da li ispaša još traje i kolika je. Cijena takve pčelarske SMS vage ovisi o proizvođača do proizvođača te o kvaliteti i mogućnostima, a iznosi od 300€ do 800€.

3. DIZAJN SKLOPA

Postojeće pčelarske vage imaju visoku cijenu te je odlučeno da se izrade vlastita rješenja. U nastavku su opisane komponente koje su korištene za izradu pčelarske SMS vage. Pčelarska SMS vaga realizirana je pomoću Arduino Uno razvojne pločice s ATMEGA328P mikroupravljačem.

3.1 Opis komponenti

3.1.1 Arduino

Glavni dio sklopa je Arduino Uno mikroupravljač. Arduino Uno sadrži ATMEGA328P mikroupravljač. To je integrirani krug iz ATMEL porodice integriranih krugova. ATMEGA328P u sebi sadrži CPU, ALU, FLASH memoriju, SRAM memoriju, EPROM te također sadrži registre i podržava više različitih komunikacijskih protokola. ATMEGA328P u sebi sadrži i *bootloader* koji omogućuje programiranje putem UART (engl. *Universal Asynchronous Receiver-Transmitter*) komunikacije. Neke dodatne karakteristike su [4]:

- 32K bytea FLASH memorija
- 2K bytea SRAM(statička radna memorija)
- 1K bytea EEROM memorija
- 1-20 MHz radni takt(kristalni oscilator ili vanjski generator takta)
- 20MIP-a na taktu od 20MHz
- UART/USART komunikacija
- 8 bitni Timer0, 16 bitni Timer1 i 8 bitni Tmer2
- 10 bitni ADC s 8/6 ulaznih kanala
- 6 PWM kanala
- SPI komunikacija
- I2C komunikacija
- Mod za uštedu energije (engl. *Sleep mode*)
- Prekid programa na vanjsku pobudu (engl. *Interrupt*)
- Interni RC oscilator od 8MHz

Sl.3.1. Arduino Uno razvojna pločica [5]

3.1.2. Senzor temperature i vlage zraka - DHT22

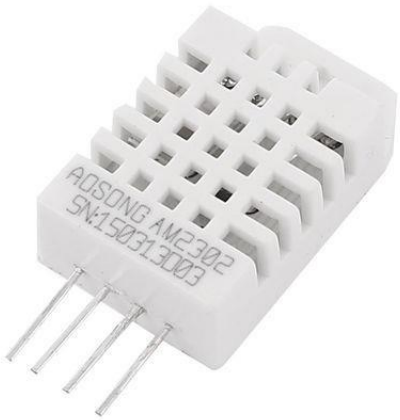
Kako bi se moglo izmjeriti parametre iz vanjskog svijeta potrebni su senzori (npr. temperature zraka, vlagu zraka i masa). Senzori predaju neku električnu veličinu (napon, struja i otpor) mikroupravljaču te je tu vrijednost potrebno pretvoriti kako bi se mogla dobiti odgovarajuća veličina. U projektu svi senzori predaju digitalni signal na izlaz.

Mjerenje temperature i vlage zraka izvodi se pomoću senzora DHT22. Senzor radi na naponu napajanja od 3.3V do 5.5V istosmjernog izvora. Mjerenje temperature izvedeno je preko termistora, dok za mjerenje vlage koristi kapacitivni senzor koji količinu vlage u zraku pretvara u promjenu vrijednosti kapaciteta navedenog senzora. Razlog korištenja ovog senzora je prihvatljiva cijena, dobra točnost, jednostavan prijenos informacija (koristi samo jednu žicu za prijenos informacije za temperature i vlage zraka), mala potrošnja. Jedina mana je vrlo sporo očitavanje u trajanju od 2 sekunde.

Karakteristike DHT22 senzora su sljedeće [6]:

- Visoka preciznost
- Prijenos informacija jednom žicom
- Prijenos podataka na velike udaljenosti (preko 100m)
- Područje mjerenja: Temperatura- od -40°C do +80°C, Vlaga- 0-100%RH

- Točnost : Temperatura- ±0.5°C, Vlaga- ±5%
- Napon napajanja 3.3 VDC - 5.5 VDC
- Potrošnja struje: Stanje čekanja: 40-50 µA, Mjerenje: 1.5 mA
- Trajanje mjerenja: 2 sekunde



Sl.3.2 DHT22 senzor temperature i vlage zraka [6]

3.1.3. Senzor za mjerenje mase – HX711 pretvornik i Loadcell

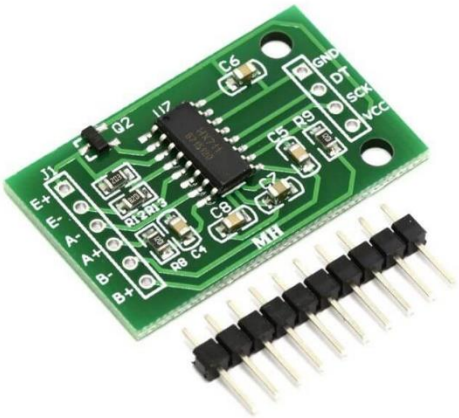
Za mjerenje težine korišten je senzor mase (engl. *Load cell*) i HX711 ADC (engl. *Analog-to- Digital Converter*) pretvornik. Senzori mase su zapravo mjerni pretvornici težine čiji se otpor mijenja promjenom sile pritiska na senzor. Pretvara silu pritiska u promjene el. otpora koji se može izmjeriti. Kada se primjene vanjske sile primjene tada pritisak (engl. *Stress*) i naprezanje (engl. *Strain*) predstavljaju rezultate. Pritisak se definira kao unutarnja sila, dok je naprezanje definirano kao deformacija i pomak koji se javljaju. Naprezanje se može uzrokovati silom, momentom, promjenom strukture materijala, pritiskom, toplinom i sl. Sastoji se od žičane mreže niti vezane tankim slojem ljepila (epoksi smola) na površini mjernog senzora. Prilikom opterećenja površine mijenja se njena duljina te se u isto vrijeme mjenja i otpor. Kod izbora mjernog senzora mora se paziti na njegove karakteristike, npr. njegovo maksimalno opterećenje i temperaturna osjetljivost. Kako bi se izmjerilo opterećenje senzor mase treba se spojiti na Arduino razvojnu pločicu koja omogućuje mjerenje samo vrlo male promjene otpora. Koriste se četiri senzora mase, tako da je svaki na jednom kutu vage te su spojeni u wheatstoneov most.

Zatim su spojeni na HX711. HX711 je analogno digitalni pretvornik s pojačalom koji pojačava signal 128 puta. ADC mjerenje otpora pretvara u digitalni podatak koji se šalje mikroupravljaču koji te podatke odrađuje [7].

Karakteristike HX711 ADC-a [8]:

- Preciznost ADC: 24-bit
- Pojačanje: 128 za A kanal, 64 za B kanal
- Napon: 2.7V - 5.5V
- Struja: < 1.5mA
- Brzina očitanja: 10SPS(samples per second) ili 80SPS
- Dimenzije: 22 x 25 mm

Sl.3.3 Senzor mase (engl. *loadcell*) [9]



Sl.3.4. HX711 ADC pretvornik [10]

3.1.4. Ekran – LCD 16x2 Display

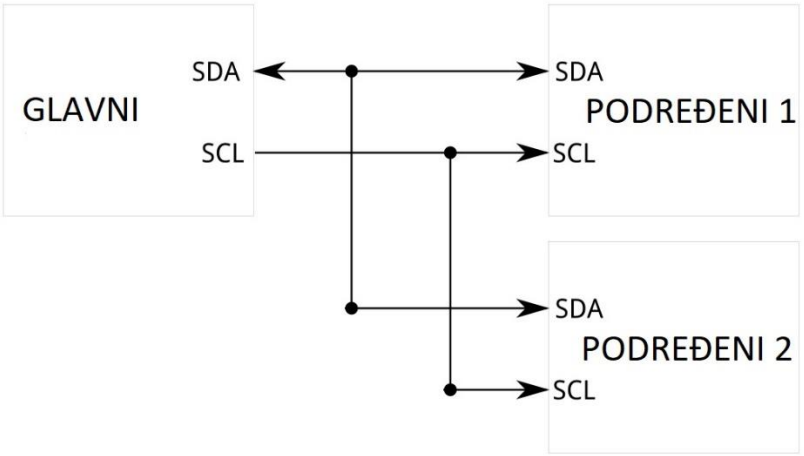
Kako bi se moglo prikazati rezultate koristio se LCD 16x2 ekran s HD44780 kontrolerom. HD44780 kontroler omogućuje prikazivanje teksta, simbola i vrijednosti. Sposoban je pokazati 16 znakova u 2 reda.

Sl.3.5. LCD 16x2 ekran [11]

3.1.5. I2C Adapter za LCD 16x2 ekran

I^2C adapter omogućuje povezanost LCD 16x2 ekrana pomoću I^2C komunikacije. I^2C komunikacijom se zauzima samo dva pina na Arduino Uno razvojnoj pločici umjesto sedam pinova. Pinovi koji se koriste su A4 i A5. I^2C protokol (engl. *The Inter-integrated Circuit*) je protokol koji omogućuje komunikaciju više podređenih (engl. *Slaves*) integriranih krugova s glavnim (engl. *Master*) integriranim krugom [12].

Sl.3.6. I2C adapter [13]



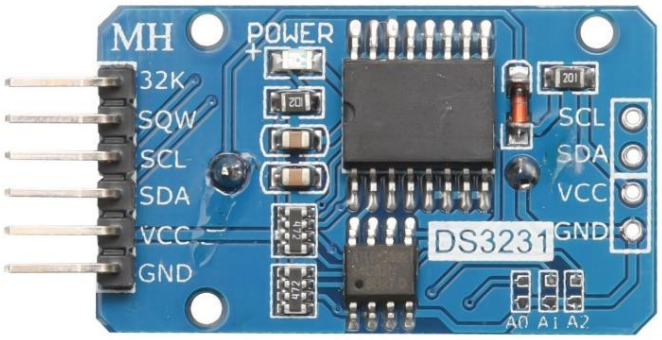
Sl.3.7. Primjer I2C komunikacije [12]

3.1.6. Stvarno-vremenski sat – DS3231 RTC (engl. *Real-Time-Clock*)

DS3231 RTC je elektronički uređaj koji precizno prati vrijeme. Koristi kristalni oscilator od 32.768 kHz. Takav kristalni oscilator također se koristi u kvarcnim satovima. Precizno prati sekunde, minute, sate, dan, mjesec i godinu. Također ima ugrađen i temperaturno kompenzirani kristalni oscilator (engl. *Temperature Compensated Crystal Oscillator*). On osigurava dodatnu stabilnost prilikom promjene temperature. S ovim RTC-om se koristi CR2032 baterija koja napaja RTC u slučaju nestanka prekida napajanja na njemu. DS3231 isto koristi I^2C komunikaciju [14].

Karakteristike su [14]:

- Napon napajanja: 3.3V do 5.5V
- Struja: 170µA
- Komunikacija: 400kHz (I^2C)
- Temperatura: -40ºC do 85ºC



Sl.3.8. DS3231 RTC [14]

3.1.7. GPRS/GSM Shield

GPRS shield baziran je na SIM900 od tvrtke SIMCOM i kompatibilan je s Adruino platformom. GPRS shield omogućuje komunikaciju koristeći GSM ćelijsku mrežu. Shield omogućuje slanje SMS-a, MMS-a te pozive preko UART-a. Na shield-u se preko UART-a šalju AT komande pomoću kojih se upravlja GPRS modul. Također module ima i 12 GPIO pinova, 2 PWM pinova i ADC pin sa SIM900 modula [15].

Karakteristike [15]:

- Quad-Band 850/900/1800/1900 MHz – radi na svim GSM mrežama u svijetu
- GPRS multi-slot klasa 10/8
- GPRS mobilna stanica klase B
- Kompatibilno sa GSM faza 2/2+
- Klasa 4 (2W @ 850 / 900 MHz)
- Klasa 1 (1W @ 1800 / 1900 MHz)
- Kontrola pomoću AT komanda: - Standardne komande: GSM 07.07 & 07.05
  - Napredne komande: SIMCOM AT komande
- SMS (engl. *Short Message Servise*)- slanje male količine podataka putem mreže
- Ugrađen TCP/UDP stog- omogućuje prijenos podataka na internet server (engl. *Web server*)
- RTC podrška

- Odabir serijskog porta
- 3.5mm priključak za zvučnike i slušalice
- Mala potrošnja energije – 1.5mA u stanju mirovanja (engl. *Sleep mode*)
- Radna temperatura: od -40ºC do 85ºC

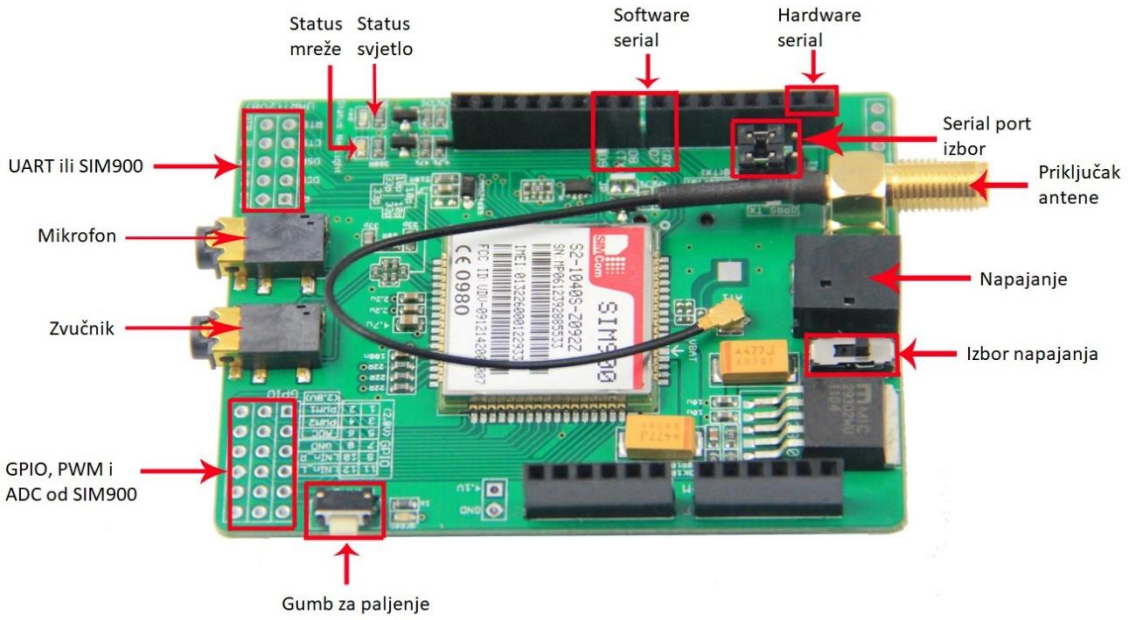
Primjena GPRS modula [15]:

- Daljinsko upravljanje uređajima
- Daljinsko upravljanje vremenskim stanicama ili bežičnim senzorima
- Sustav praćenja vozila s GPS modulom

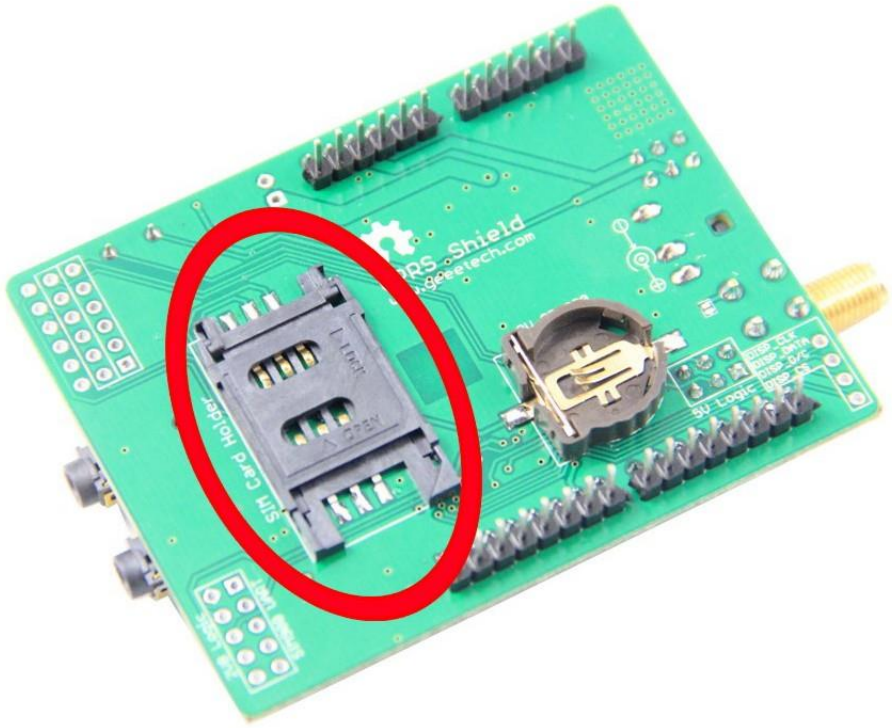
Sl.3.9. GPRS shield-a [15]

Opis hardvera GPRS shield-a [15]:

- Izbor napajanja - izbor između vanjskog napajanja i napajanja s Arduino razvojne pločice (5V)
- Napajanje – vanjsko napajanje - 4.8~5VDC
- Priključak antene – spajanje vanjske antene
- Serial port izbor - omogućuje izbor između softver i hardver serijske komunikacije
- Hardware serial – D0/D1 pinovi na GPRS shield-u
- Software serial – D7/D8 pinovi na GPRS shield-u
- Status svjetlo – govori da li je uključen SIM900 modul
- Status mreže – govori da li je SIM900 spojen na mrežu
- UART ili SIM900 - UART pinovi iz SIM900 modula
- Mikrofon – služi za pozive
- Zvučnik – služi za pozive
- GPIO, PWM i ADC od SIM900 – GPIO,PWM i ADC pinovi od SIM900 modula
- Gumb za paljenje – služi za uključivanje SIM900 modula



Sl. 3.10. Opis GPRS shield-a [15]



Sl. 3.11. Ulaz za SIM karticu [15]

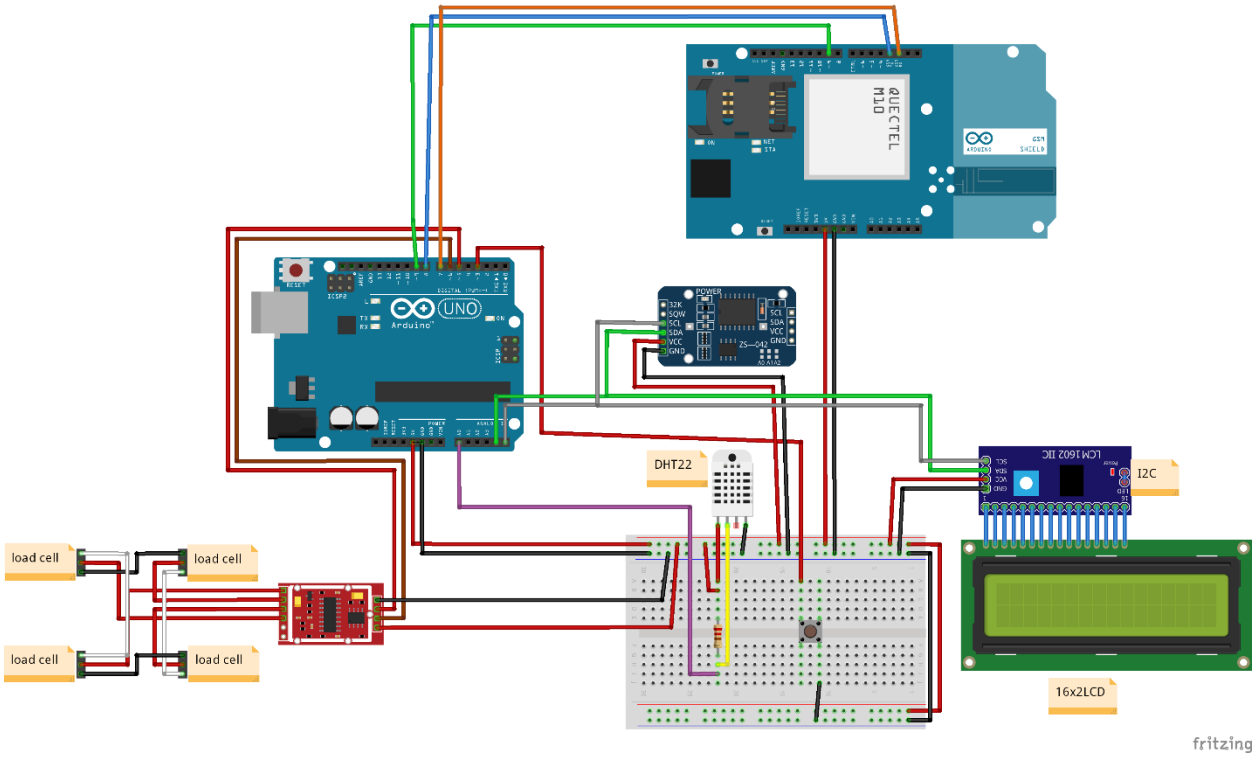
GPRS shield ima indikacijska LED svjetla (napajanje (zelena), status (plava), status mreže (crvena)) pomoću kojih korisnik može saznati radno stanje shield-a. Detaljne informacije o led svjetlima pišu u tablici:

Led svjetlo	Status	Opis
Napajanje (zelena)	Upaljena	Upaljen GPRS Shield
	Ugašena	Ugašen GPRS Shield
Status (plava)	Upaljena	Upaljen SIM900
	Ugašena	Ugašen SIM900
Status mreže (crvena)	64ms Upaljen/800ms Ugašen	SIM900 nije registriran na
	64ms Upaljen/3000ms Ugašen	SIM900 je registriran na mrežu
	64ms Upaljen/300ms Ugašen	GPRS komunikacija
	Ugašena	SIM900 ne radi

Tab 3.1. Opis LED svjetala [15]

3.2 Izrada sheme sklopa

Nakon izbora potrebnih komponenata osmišljena je shema sklopa pomoću programskog alata Fritzing [16] koji omogućava jednostavno crtanje shema koje koriste Arduino razvojne pločice. Na slici 3.12. prikazan je sklop koji se koristi u ovom projektu. Detaljni prikaz sheme se nalazi u prilogu P.2.1.



Sl.3.12. Shema pčelarske SMS vage

4. IZRADA PROGRAMSKOG KODA

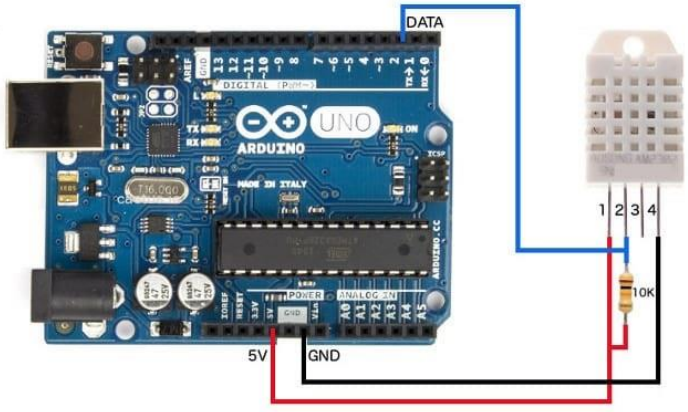
Za izradu programskog koda za pčelarsku SMS vagu korišten je Arduino IDE programski alat. Arduino IDE je programski alat koji se primjenjuje za programiranje Arduino baziranim razvojnim pločicama pomoću *C* ili *C++* programskog jezika. Arduino IDE je dizajniran na način da ga razumiju početnici te je besplatan. Unutar Arduino IDE sučelja potrebno je definirati dvije glavne funkcije:

- *void setup()* - funkcija unutar koje se postavljaju početne varijable i postavke. Ova funkcija se izvršava samo jednom prilikom pokretanja Arduina
- *void loop()* – funkcija koja se poziva odmah poslije void setup() funkcija. Void loop je beskonačna petlja koja se ponavlja sve dok se Arduino ne ugasi.

Prije izrade glavnog koda potrebno je bilo napraviti ispitivanje svih pojedinačnih komponenti na način da se naprave ili nađu testni kodovi za te komponente. Prije izrade koda potrebno je bilo isprobati kako radi svaka komponenta kako bi se upoznao njihov način rada.

4.1 Primjer koda za mjerenje temperature i vlage (DHT22)

Prije pokretanja testnog koda za DHT22 potrebno ga je pravilno spojiti na Arduino razvojnu pločicu. Način spajanja DHT22 senzora kako bi se koristio navedeni primjer koda prikazan je na Slici 4.2. Priključak 2 se spaja na jedan ulazni priključak na Arduino razvojnoj pločici (upisuje se unutar koda pod DHTPIN) dok se priključak 1 spaja a +5V, a priključak 4 na GND. Također prije korištenja potrebno je bilo instalirati odgovarajuću DHT22 biblioteku.



Sl.4.1. Primjer spajanja DHT22 senzora [17]

Primjer programskog koda preuzet je iz primjera koji se nalazi unutar same DHT biblioteke. Primjer mjeri vrijednost temperature u stupnju celzijusa (C°), vlagu zraka u postotku i temperaturu zraka u stupnjevima stupanju fahrenheit (F°) te zatim te izmjerene vrijednosti ispisuje na *Serial Monitor*.

```
#include "DHT.h"
#define DHTPIN 2  // definiranje pina na koji se spaja DHT
#define DHTTYPE DHT22

DHT dht(DHTPIN, DHTTYPE);  // inicijalizacija DHT senzora
void setup()
{
  Serial.begin(9600);
  Serial.println("DHTxx test!");
  dht.begin();
}
void loop()
{
delay(2000); // odgoda u trajanju 2 sekunde
// očitanje vlage u postotku
float h = dht.readHumidity();
// očitovanje temperature u Celzijusima
float t = dht.readTemperature();
// očitovanje temperature u Fahrenheit
float f = dht.readTemperature(true);

// ispitivanje očitanja, ako nisu dobra mjerenje se
// ponavlja
if (isnan(h) || isnan(t) || isnan(f))
{ Serial.println("Failed to read from DHT sensor!");
return;
}
float hif = dht.computeHeatIndex(f, h);
float hic = dht.computeHeatIndex(t, h, false);
```

```
Serial.print("Humidity: ");
Serial.print(h);
Serial.print(" %\t");
Serial.print("Temperature: ");
Serial.print(t);
Serial.print(" *C ");
Serial.print(f);
Serial.print(" *F\t");
Serial.print("Heat index: ");
Serial.print(hic);
Serial.print(" *C ");
Serial.print(hif);
Serial.println(" *F");
}
```

4.2 Primjer koda za mjerenje mase

Kako bi se kod u primjeru mogao koristiti potrebno ga je pravilno spojiti na Arduino razvojnu pločicu. Koriste se četiri senzora mase tako da je svaki na jednom kutu vage te su spojeni u wheatstoneov most se zatim spajaju na HX711 analogno digitalno pretvornik. HX711 ima priključke +5V i GND koji se spajaju na istoimene priključke na Arduino razvojno pločicu. Priključci DT i SCK se mogu definirani unutar samog koda. U ovom primjeru priključci DT i SCK su spojeni na 5 i 6 Arduino razvojne pločice. Prije samog korištenja senzora potrebno je bilo instalirati HX711 biblioteku. Poslije instalacije biblioteke potrebno je napraviti kalibraciju pomoću koje se dobije kalibracijski faktor kako bi se ispravno mjerila masa [18].

```
#include "HX711.h"
#define DOUT 5 //pin na koji se spaja DT pin s HX711
#define CLK 6 //pin na koji se spaja SCK pin s HX711
HX711 scale(DOUT, CLK); //konstruktor za našu biblioteku za HX711 ADC
//faktor kalibracije koji treba odrediti (nije točan za trenutni load cell) njega će se
//mijenjati kroz Serial monitor kako bi se odredilo pravi faktor
float calibration_factor = -96650;
```

```
void setup()
{
Serial.begin(9600);
//ispisuje se postupak kalibracije na Serial monitor
Serial.println("HX711 Kalibracija");
Serial.println("Uklonite utege sa vage");
Serial.println("Nakon sto pocnu ocitanja stavite poznati uteg na vagu");
Serial.println("Sada treba mijenjati faktor kalibracije, kako bi dobili tocno ocitanje");
Serial.println("Pritisnite a,s,d,f za povecanje faktora kalibracije za 10,100,1000,10000");
Serial.println("pritisnite z,x,c,v za smanjenje faktora kalibracije za 10,100,1000,10000");
Serial.println("Pritisnite t za tariranje");
scale.set_scale();
scale.tare(); //postavljanje vage na 0(tariramo)
}
void loop()
{
scale.set_scale(calibration_factor); //postavlja se faktor kalibracije
Serial.print("Ocitavanje: ");
Serial.print(scale.get_units(), 3);  // dobiva se masa zaokružena na 3 decimale
Serial.print(" kg");                 //ispisuje se masa u kilogramima
Serial.print(" Faktor kalibracije: ");
Serial.print(calibration_factor);    //ispisuje se faktor kalibracije
Serial.println();
//ako je korisnik nešto unio putem Serial monitora onda se izvršava
//daljnji kod ovisno o poslanom slovu mijenjamo faktor kalibracije za
//određenu vrijednost

if(Serial.available())
{
char temp = Serial.read();
if(temp == '+' || temp == 'a')
calibration_factor += 10;
else if(temp == '-' || temp == 'z')
calibration_factor -= 10;
else if(temp == 's')
calibration_factor += 100;
else if(temp == 'x')
calibration_factor -= 100;
else if(temp == 'd')
calibration_factor += 1000;
else if(temp == 'c')
calibration_factor -= 1000;
else if(temp == 'f')
calibration_factor += 10000;
else if(temp == 'v')
calibration_factor -= 10000;
else if(temp == 't')
scale.tare();                //ako je poslano slovo t onda vaga se postavlja na 0(tariramo)
}
}
```

Nakon pronalaska faktora kalibracije potrebno je bilo napraviti primjer koda koji će mjeriti i ispisivati masu unutar Serial monitor-a. Za mjerenje koristi se funkcija *scale.get_units();*.

```
#include "HX711.h"
#define DOUT 5
#define CLK 6

HX711 scale(DOUT, CLK);
float calibration_factor = -35300;

void setup()
{
Serial.begin(9600);
Serial.println("Uklonite sve utege s vage");
Serial.println("Pritisnite t za tariranje");
scale.set_scale();
scale.tare();                //postavlja vagu na 0
}
void loop()
{
scale.set_scale(calibration_factor); //postavlja se faktor kalibracije koji je deklariran
                                     //na početku
Serial.print("Ocitavanje: ");
Serial.print(scale.get_units(), 3);  //dobiva se masa zaokružena na 3 decimale
Serial.print(" kg");                 //ispisuje se masa u kilogramima
}
```

Funkcija *scale.get_units(),3;* učitava vrijednost s HX711 pretvornika te dijeli s faktorom kalibracije kako bi dobila odgovarajuću masu u kilogramima. Zatim se vrijednost mase u kilogramima ispisuje u *Serial monitor* [18].

4.3 Primjer koda za slanje SMS poruke (GPRS/GSM, AT komande)

Kako bi se testirao GPRS/GSM modul potrebno je upotrijebiti sljedeći kod. Ovaj kod služi za komunikaciju GPRS/GSM modula s računalom putem terminalnog sučelja zvanog Serial monitor.

```
#include <SoftwareSerial.h>

SoftwareSerial mySerial(7, 8);

void setup()
{
  mySerial.begin(19200); // GPRS baud rate
  Serial.begin(19200); // GPRS baud rate
}

void loop()
{
  if (mySerial.available())
  Serial.write(mySerial.read());
  if (Serial.available())
  mySerial.write(Serial.read());
}
```

Nakon prebacivanja prethodnog koda na Arduino razvojnu pločicu potrebno je otvoriti Serial monitor te postaviti brzinu prijenosa (engl. Baud rate) na 19200 bps unutar Serial monitora. GPRS/GSM modul se mora upaliti prije samog korištenja. Paljenje modula se izvodi tako da se pritisne PWR tipka te se drži 10 sekundi. Također paljenje se može izvesti putem programskog koda, ali prije je potrebno prespojiti kontakt JP. Prespajanjem tog kontakta paljenje se izvodi preko izlaza D9. Nakon prespajanja kontakta JP paljenje se izvodi pomoću sljedeće funkcije [15].

```
void powerUp()
{
 pinMode(9, OUTPUT);
 digitalWrite(9,LOW);
 delay(1000);
 digitalWrite(9,HIGH);
 delay(2000);
 digitalWrite(9,LOW);
 delay(3000);
}
```

Sl.4.2. Kontakt JP [15]

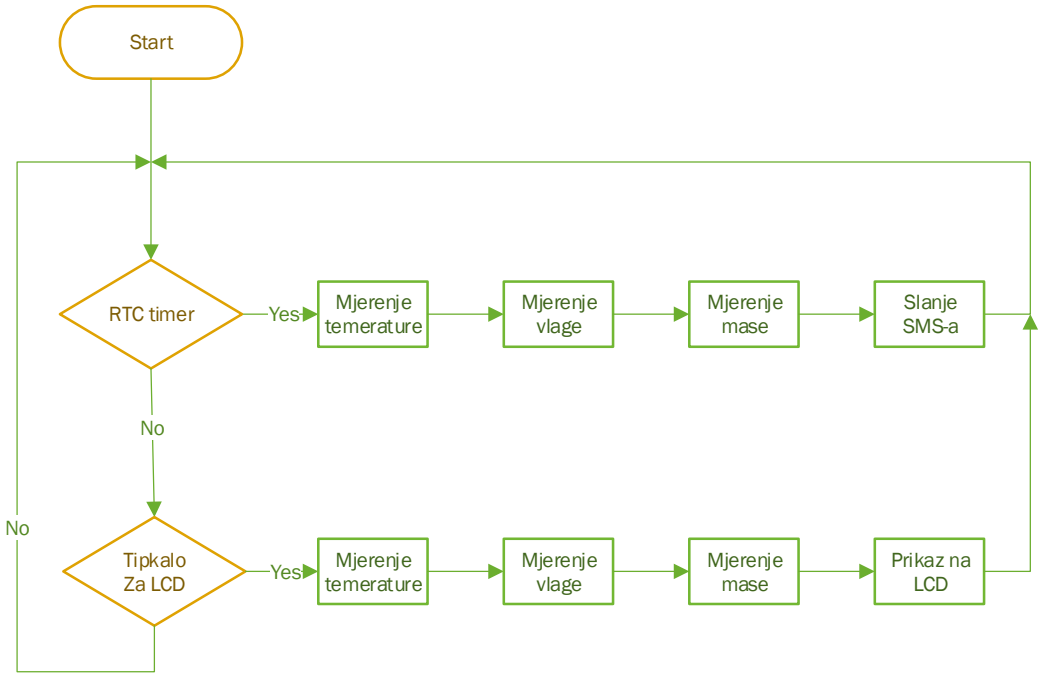
Za upravljanje GPRS/GSM modula koriste se AT komande. Kako bi se pravilno komuniciralo s modulom potrebno je poslati komandu **AT+IPR=19200**. Za provjeru o pravilnoj komunikaciji upisuje se komanda „**AT**“. Ukoliko je uspostavljena veza GPRS/GSM modul vraća odgovor „**OK**“.

Kako bi se poslala SMS poruka potrebno je izvesti sljedeće naredbe:

1. **AT+CMGF=1** – modulu se omogućuje slanje SMS na dva načina, kao tekst ili binarno. Upisivanjem 1 omogućuje se slanje poruke kao tekst.
2. **AT+CSCA=“+385970703“** – broj SMS centra tele operatora
3. **AT+CMGS=“+3859xxxxxxxx“** – broj mobitela na koji se SMS poruka mora poslati
4. > dobiva se znak kao odgovor te nakon toga znaka se upisuje sadržaj poruke koja se treba poslati
5. Nakon upisivanja sadržaja poruke šalje se heksadekadski broj **1A** ili se pritisne *Ctrl+Z* kako bi se poruka poslala. Ukoliko je poruka pravilno poslana dobije se odgovor **OK** ili **ERROR** ukoliko je došlo do pogreške.

4.4 Izrada vlastitog programskog koda

Kako bi se krenulo s izradom programskog koda prvo je potrebno napraviti blok dijagram da bi se znao redoslijed izvedbe događaja u programskom kodu. Programski kod se vrti u krug (engl. *loop*) petlji. Zatim u slučaju ispunjenog uvjeta izvršavaju se mjerenja i prikazuje se rezultat na LCD ekran ili se šalje poruka ovisno o uvjetu koji je ispunjen.



Sl.4.3. Blok shema rada pčelarske SMS vage

Pčelarska SMS vaga prikazuje informacije koje prikuplja u dva slučaja. Arduino razvojna pločica mora istovremeno provjeravati oba uvjeta po kojima prikazuje informaciju. Arduino razvojna pločica nije u mogućnosti ispuniti oba uvjeta u isto vrijeme stoga je programski kod izveden na način da se provjerava prvo jedan, a zatim drugi uvjet. Prikupljene informacije dobiju se u sljedeća dva slučaja:

1. Pritiskom tipkala za LCD – pritiskom na tipkala šalje se signal na D3 priključak Arduino razvojne pločice. Zatim se provode mjerenja pozivanjem funkcije *mjerenje();* pomoću koje se mjere potrebne informacije (temperature zraka, vlage zraka i mase košnice). Zatim se izmjerene informacije prikazuju na LCD display.
2. Vremenski alarm – na Arduino razvojnu pločicu spojen je vanjski RTC pomoću kojega Arduino prati koliko je sati. Prije samog slanja SMS poruke poziva se funkcija

mjerenje(); Uvjet je ispunjen kada je trenutno vrijeme jednako zadanom vremenu. Zadano vrijeme je vrijeme u kojem će se SMS poruka poslati s potrebnim informacijama (temperature zraka, vlage zraka i mase košnice). Za slanje SMS poruke poziva se funkcija *SendTextMessage();* pomoću koje se šalje SMS poruka na željeni broj mobilnog uređaja.

U nastavku je prikazan cijeli programski kod koji je opisan

```
#include "DS3231.h"
#include <LiquidCrystal_I2C.h>
#include <SoftwareSerial.h>
#include <LowPower.h>
#include <String.h>
#include "DHT.h"
#define DHTPIN A0
#define DHTTYPE DHT22
#include "HX711.h"

#define DOUT 5 //pin na koji se spaja DT pin na pretvornik
#define CLK 6 //pin na koji se spaja SCK pin na pretvornik
LiquidCrystal_I2C lcd(0x27, 16, 2);
DHT dht(DHTPIN, DHTTYPE);
SoftwareSerial mySerial(7, 8);
DS3231 rtc(SDA, SCL);
Time  vrijeme;

HX711 scale(5, 6);

float calibration_factor = 19650; // Kalibraciski faktor koji je prije određen
float units=0;
float h=0;
float t=0;

int sat = 19, minut =00;      //alarm
int lcd_on;
```

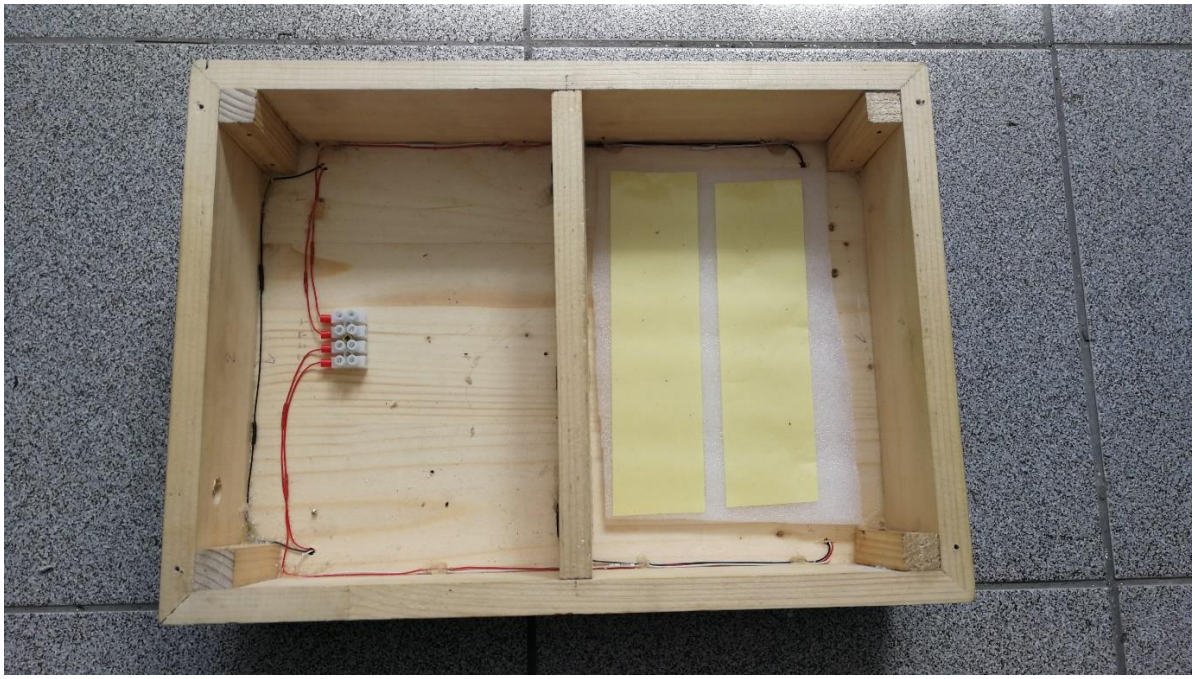
```
void setup()
{
  lcd.begin();
  dht.begin();
  rtc.begin();

  lcd.backlight();
  pinMode(3, INPUT_PULLUP);          // Prekidač za palenje LCD ekrana

  mySerial.begin(19200);

  lcd.setCursor(0, 0);
  lcd.print("Vaga je spremna");

  delay (2000);

  scale.set_scale(calibration_factor);
  scale.tare();                      //Tariranje, restatriranje na 0

  lcd.clear();
  lcd.noBacklight();

  scale.power_down();
}
void loop()
{
  vrijeme = rtc.getTime();
  lcd_on = digitalRead(3);

  if (lcd_on == LOW)
  { scale.power_up();
    lcd.backlight();

    mjerenje();

    lcd.setCursor(0, 0);             //ispis LCD
    lcd.print("V:");
    lcd.print(h);
    lcd.print("%");
    lcd.setCursor(8, 0);
    lcd.print("T:");
    lcd.print(t);
    lcd.print("C");
    lcd.setCursor(0, 1);
    lcd.print("M:");
    lcd.print(units);
    lcd.print("kg");
```

```
lcd.setCursor(9, 1);
    lcd.print("A:");
    lcd.print(sat);
    lcd.print(":");
    lcd.print(minut);

    delay (5000);

    lcd.noBacklight();
    lcd.clear() ;
    scale.power_down();
  }
//uvjet za sljanje SMS-a
  if (vrijeme.hour == sat && vrijeme.min == minut && vrijeme.sec == 0)
  { units=0;
    scale.power_up();
    mjerenje();
    powerUp();
    delay(5000);
    SendTextMessage();
    delay(10000);
    powerDown();
  }
}
void SendTextMessage()
{
  mySerial.println("AT+CMGF=1\r");   //SMS tekst način rada
  delay(100);
  mySerial.println("AT + CMGS = \"+385955203777\""); //broj mobitela na koji se SMS poruka
mora poslati
  delay(100);

  mySerial.print("Vlaga:");                 //Sardžaj poruke
  mySerial.print(h);
  mySerial.print("%\nTemp.:");
  mySerial.print(t);
  mySerial.print("C\nMasa:");
  mySerial.print(units);
  mySerial.print("kg");
  mySerial.print("\nAlarm:");
  mySerial.print(sat);
  mySerial.print(":");
  mySerial.print(minut);

  delay(100);
  mySerial.println((char)26); //ASCII kod za crtl+z je 26
  delay(100);
  mySerial.println();
  delay(1000);
}
```

```
void powerUp()
{
  pinMode(9, OUTPUT);
  digitalWrite(9, LOW);
  delay(1000);
  digitalWrite(9, HIGH);
  delay(2000);
  digitalWrite(9, LOW);
  delay(3000);
}
void powerDown()
{
  powerUp();
}

void mjerenje()
{
  // očitavanje vlage zraka, očitanje može potrajati do 250 milisekunde
  h = dht.readHumidity();
  // Očitavanje temeprature
  t = dht.readTemperature();

  scale.set_scale(calibration_factor);
  units = scale.get_units(10), 3;
  if (units < 0)
  {
    units = 0.00;
  }

}
```

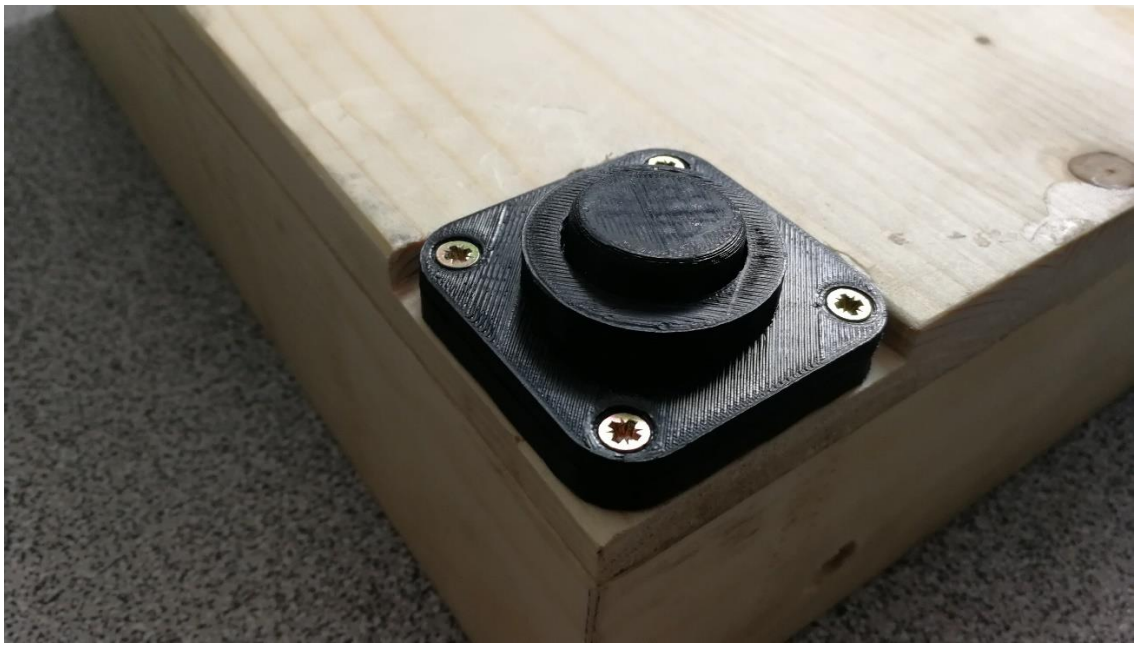
5. IZRADA PČELARSKE SMS VAGE

Nakon što je napravljena shema spajanja komponenti (slika 3.12.) potrebno je bilo napraviti kućište unutar kojega bi sve komponente stale. Također potrebno je bilo montirati Senzor mase (*load cell* senzor) na samo podnožje samog kućišta pčelarske SMS vage. Iz praktičnih razloga materijal iz kojega će se izraditi pčelarska SMS vaga je drvo zbog brzog oblikovanja i sastavljanja. Nakon toga odabrane su osnovne vanjske dimenzije samog kućišta (visina, širina i duljina). Prvo je izrađeno drveno kućište koje je prikazano na slici 5.1.



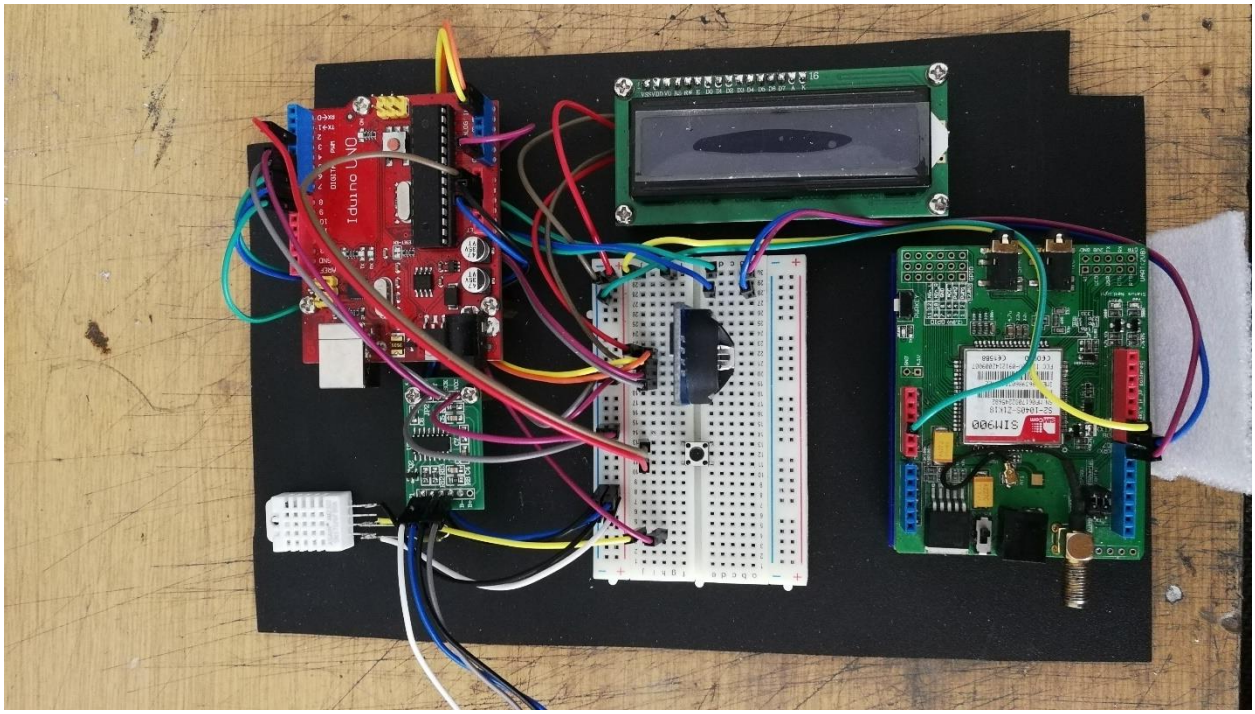
Sl.5.1. Drveno kućište

Kako bi se postavili senzore mase potrebno je bilo napraviti utore za njih. Kako bi se lakše postavili senzori mase na podnožje vage korišten je programski alat Autodask Fusion 360 kako bi se izradilo kućište za senzor mase. Korištenjem 3D printera isprintano je četiri kućišta za svaki senzor. Kućišta su napravljena tako da se pričvrste na podnožje vage pomoću vijaka. Na slici 5.2. prikazan je primjer kućišta za senzor mase te način na koji je pričvršćen.

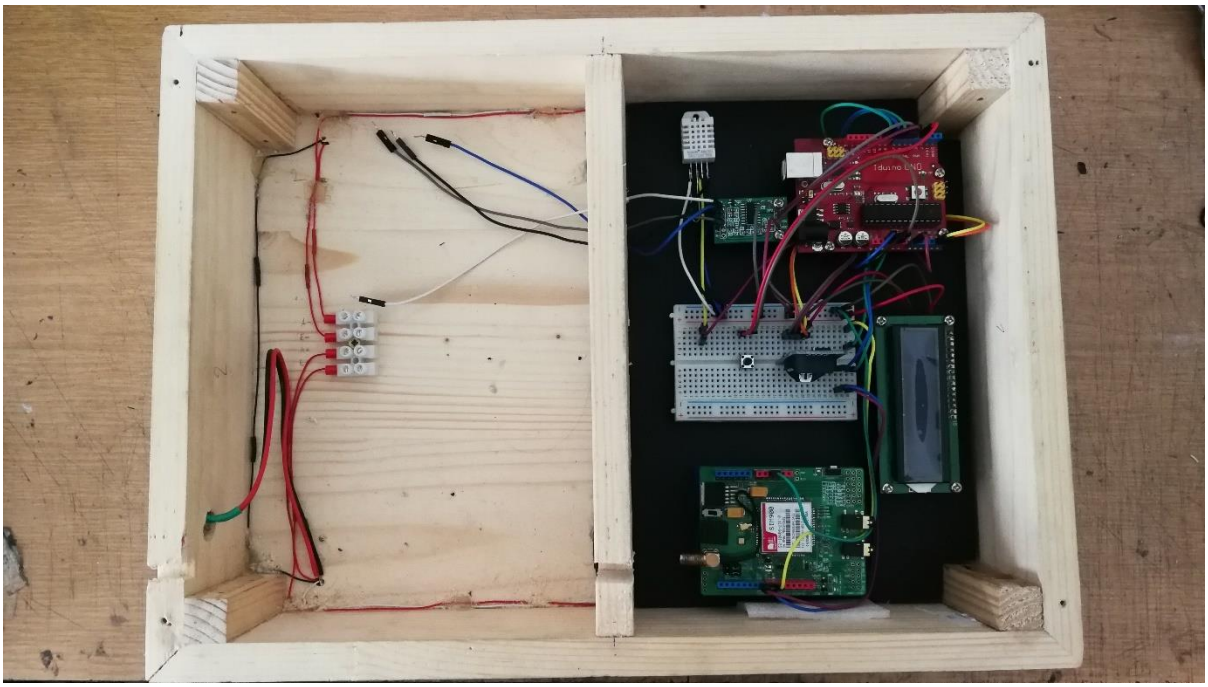


Sl.5.2. Kućište za senzore mase

Za pričvršćivanje elektroničkih komponenti napravljena je kvadratna plastična ploča na kojoj su pričvršćene sve korištene komponente. Glavne komponente su napravljene tako da se mogu izvaditi u jednom komadu iz vage. Na slici 5.3. prikazana je plastična ploča na kojoj se nalaze sve korištene komponente. Na slici 5.4. prikazana je ploča s glavnim komponentama unutar kućišta vage.

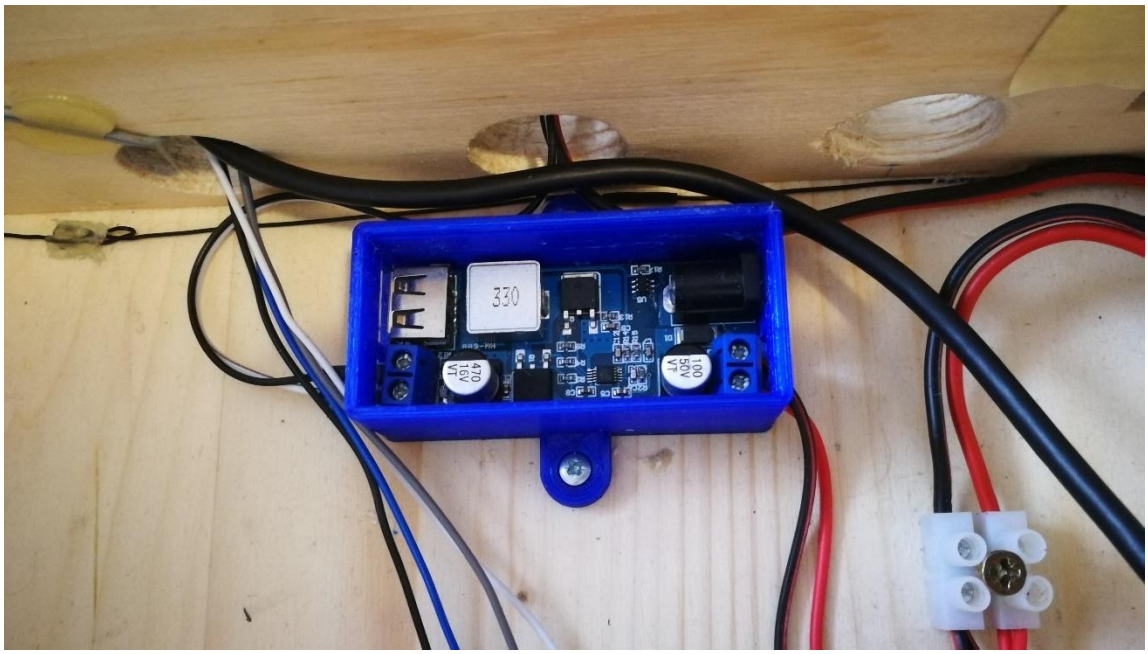


Sl.5.3. Ploča s glavnim komponentama



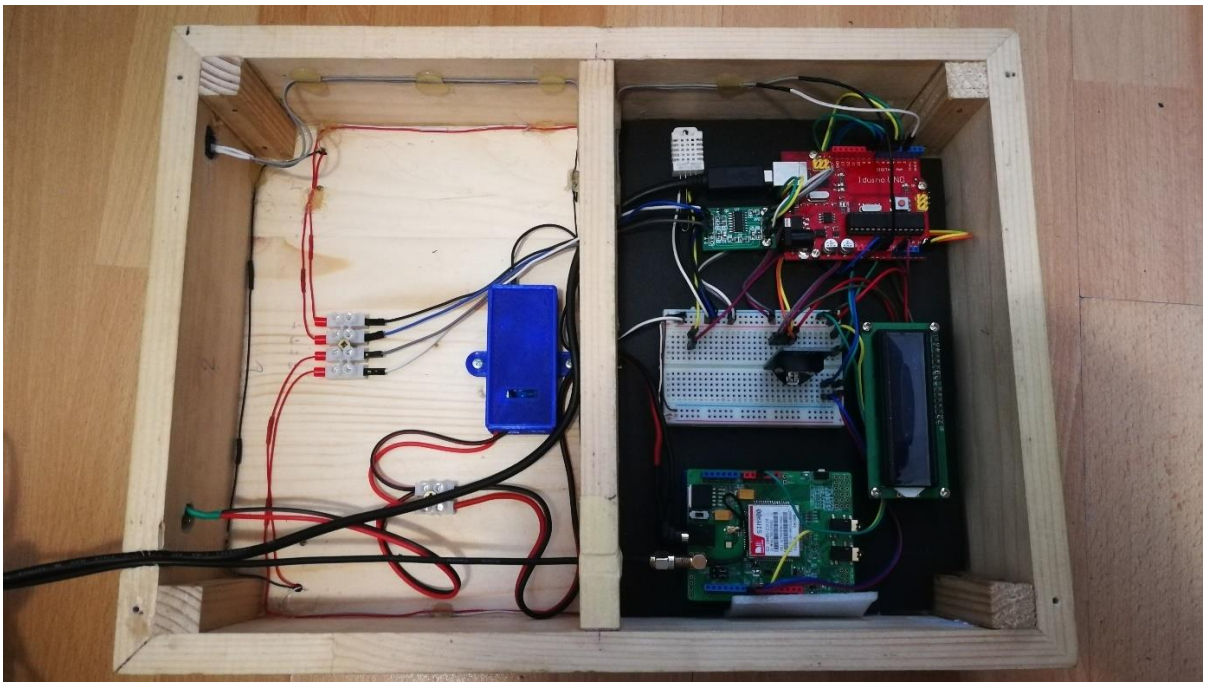
Sl.5.4. Kućište vage s glavnim komponentama

Na vagi je potrebno napraviti napajanje. Potrebno je 5V za Arduino razvojnu ploču i 12V za GPRS/GSM modul. Napajanje je izvedeno pomoću pretvarača napona za kojeg je napravljeno kućište pomoću 3D printera. Pretvarač koji je korišten ima mogućnost pretvaranja 24V-12V na 5V. Pretvarač je napajan pomoću baterije 12V. Na slici 5.5. prikazan je pretvarač napona u svome 3D printanom kućištu.



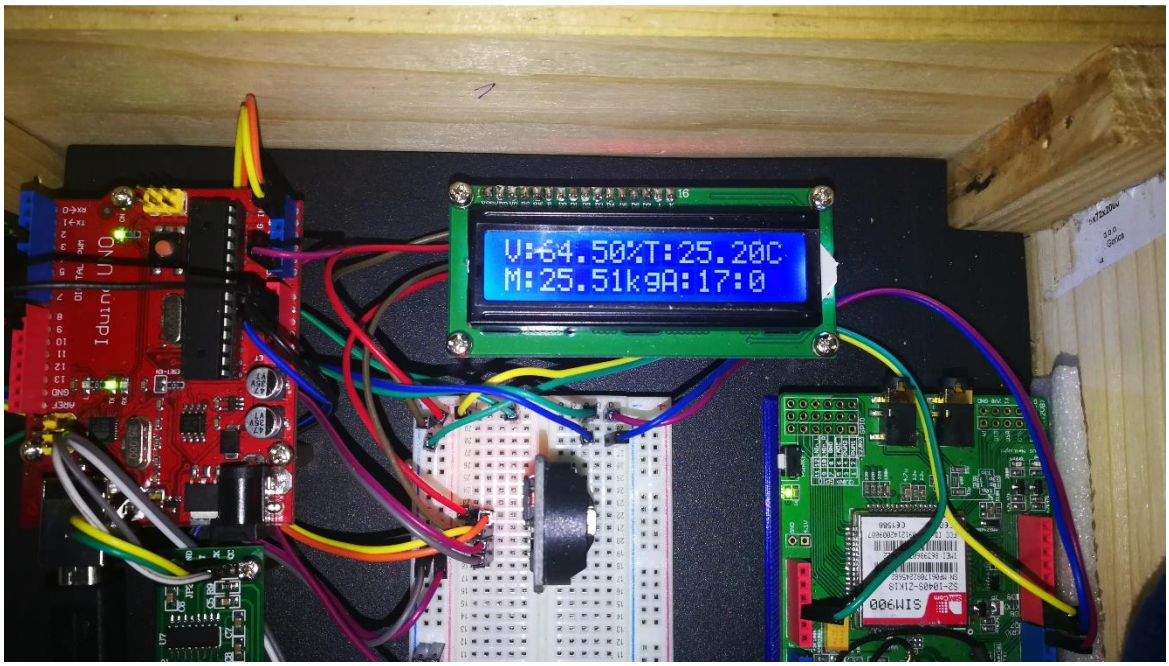
Sl.5.5. Pretvornik napona

Također je postavljen priključak za bateriju i tipkalo kojim se prikazuju mjerenja na LCD displeju. Probušene su rupe na bočnoj strani vage i u njih je postavljen priključak za bateriju i tipkalo. Posljednje sta je bilo potrebno napraviti je poklopac koji je izrađen od pleksiglasa kako bi se mogle pročitati izmjerene vrijednosti s LCD displeja. Na slici 5.6. prikazana je dovršena Pčelarska SMS vaga.

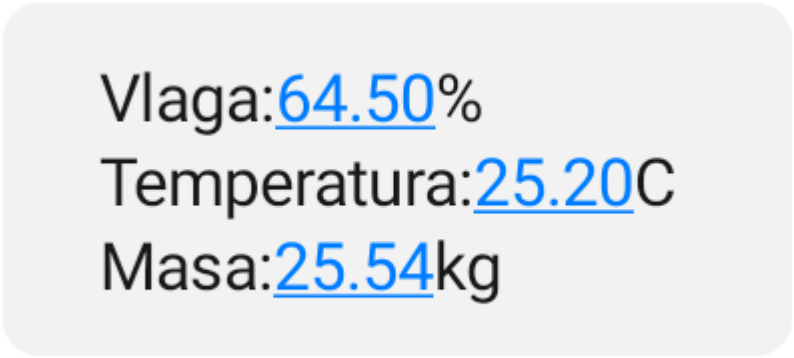


Sl.5.6. Pčelarska SMS vaga

Nakon izrade pčelarske SMS vage potrebno je bilo napraviti testiranje. Prvo je bilo potrebno postaviti programski kod u Arduino razvojnu pločicu i staviti otključanu SIM karticu unutar GPRS/GSM modula kako bi se moglo poslati SMS poruku. Na slici 5.7. vidljiv je prikaz mjerenih vrijednosti na LCD displeju. Na slici 5.8. je prikazan sadržaj SMS poruke koja je primljena na mobilni uređaj.



Sl.5.7. Prikaz rada LCD displeja



Sl.5.8. Sadržaj pristigle SMS poruke

6. ZAKLJUČAK

Cilj ovog diplomskog rada bio je izraditi pčelarsku SMS vagu koristeći Atmel studio ili Arduino IDE platformu. Korištena je Arduino IDE platforma radi već postojećih biblioteka za korištene komponente. Također programiranje u Arduino programskom okruženju je jednostavno i lako shvatljivo. Pčelarska SMS vaga ima mogućnost mjerenja težine košnice, temperature i vlage zraka okoline na mjestu gdje je postavljena. Komponente ovog projekta su osjetljive te je zbog toga došlo je do komplikacija. Najveće poteškoće su se pojavile tijekom kalibracije senzora mase. Senzori mase su tijekom dužeg vremena izbacivali sve veće i veće vrijednosti. Problem je bio u tome što se senzori mase trebaju preopteretiti s određenom masom na 24h. Senzori mase korišteni u ovom diplomskom radu su procijenjeni za rad do 50kg po senzoru stoga je uzeta masa od 30kg kako bi se senzori mase preopteretili. Prikupljanje informacija o temperaturi zraka i vlažnosti zraka uspješno je izvedena i zajedno s masom košnice se šalje u dva moguća slučaja. Prvi slučaj je prilikom pritiska tipkala koje se nalazi na samoj pčelarskoj SMS vagi. U tom slučaju informacije se prikazuju na LCD displeju koji se nalazi na samoj vagi. Drugi slučaj je kada se upali alarm pomoću kojega GPRS/GSM modul šalje izmjerene informacije na mobilni uređaj putem SMS poruke. Alarm je izveden tako da se okida svakih 24h. Također alarm nije ovisan o trenutku paljenja vage. Vrijeme okidanje alarma se uvijek može promijeniti. Pčelarske vage koje su trenutno na tržištu imaju puno veću cijenu od pojedinačnih komponenata. Iz tog razloga odlučena je izrada vlastite pčelarske SMS vage. Kako bi se vaga usavršila potrebno je mnogo više vremena provesti na testiranje i nadogradnju. Također ovu vagu je moguće uvijek nadograditi s dodatnim komponentama. Moguće nadogradnje su mjerenje brzine vjetra, promjena alarma putem SMS poruke ili putem tipkala ili uplaćivane bona na SIM karticu. U slučaju izrade još jedne vage potrebni programski kod se samo stavi na Arduino razvojnu pločicu i koriste se iste komponente te nema potreba trošiti puno vremena za testiranje.

LITERATURA

[1] https://vitamini.hr/hrana-i-zivot/hrana/med-710/

[2] https://pcelarstvo-veber.hr/propolis-zdravlje-imunitet-upala-grla/

[3] http://www.microel.hr/hr/products/gprs-sms-beehive-scale-xlog-bee#osnovno

[4] https://www.sparkfun.com/datasheets/Components/SMD/ATMega328.pdf

[5] https://www.geeetech.com/iduino-uno-328-with-atmega328ppu-compatible-with-arduino-ide-p-740.html

[6] https://www.indiamart.com/proddetail/dht22-digital-temperature-and-humidity-sensor-9936529097.html

[7] https://e-radionica.com/hr/blog/2019/04/08/kkm-hx711-load-cell/

[8] https://e-radionica.com/hr/adc-24-bit-s-pojacalom-hx711-made-by-e-radionica.html

[9] https://www.sparkfun.com/products/10245

[10] https://www.jsumo.com/hx711-module-load-cell-voltage-amplifier

[11] https://e-radionica.com/hr/lcd-ekran-16x2-hd44780.html

[12] https://learn.sparkfun.com/tutorials/i2c/all

[13] https://e-radionica.com/hr/iic-lcd-adapter.html

[14] https://e-radionica.com/hr/blog/2016/02/17/kkm-rtc-real-time-clock/

[15] http://www.geeetech.com/wiki/index.php/Arduino_GPRS_Shield

[16] http://fritzing.org/home/

[17] https://www.botshop.co.za/product/dht22-temperature-and-humidity-sensor-high-precision/

[18] https://e-radionica.com/hr/blog/2019/04/08/kkm-hx711-load-cell/

POPIS I OPIS UPOTREBLJENIH KRATICA

ADC – Analog to Digital Converter

AT KOMANDE - Attention naredbe

EEPROM - Electricaly Erasable Programmable Read Only Memory

GPRS - General Packet Radio Service

GSM - Global System for Mobile Communication

IDE - Integrated Development Environment

LCD - Liquid Crystal Display

RTC - Real Time Clock

SMS - Short Message Service

SRAM - Static Random Acces Memory

UART - Universal Asynchronous Receiver/Transmitter

SAŽETAK

U ovom radu objašnjena je izrada pčelarske SMS vage uz pomoć Arduino IDE platforme. Kroz ovaj rad upoznaju se osnove pčelarstva, proizvodi koji nastaju te osnovni alati i pribor koji se koristi. Pčelarska SMS vaga jedan je od osnovnih alata pčelara. SMS vaga osim mase košnice mjeri temperaturu i vlagu zraka okoline. Izmjerene podatke šalje putem SMS poruke na mobilni uređaj. Ovakva pčelarska SMS vaga omoguće nadzor košnice na velike udaljenosti. Prednost korištenja SMS vage je ušteda vremena i novaca.

KLJUČNE RIJEČI:

Arduino, pčelarska SMS vaga, mjerenje mase, mjerenje temperature i vlage zraka okoline

BEEKEEPER SMS SCALE

ABSTRACT

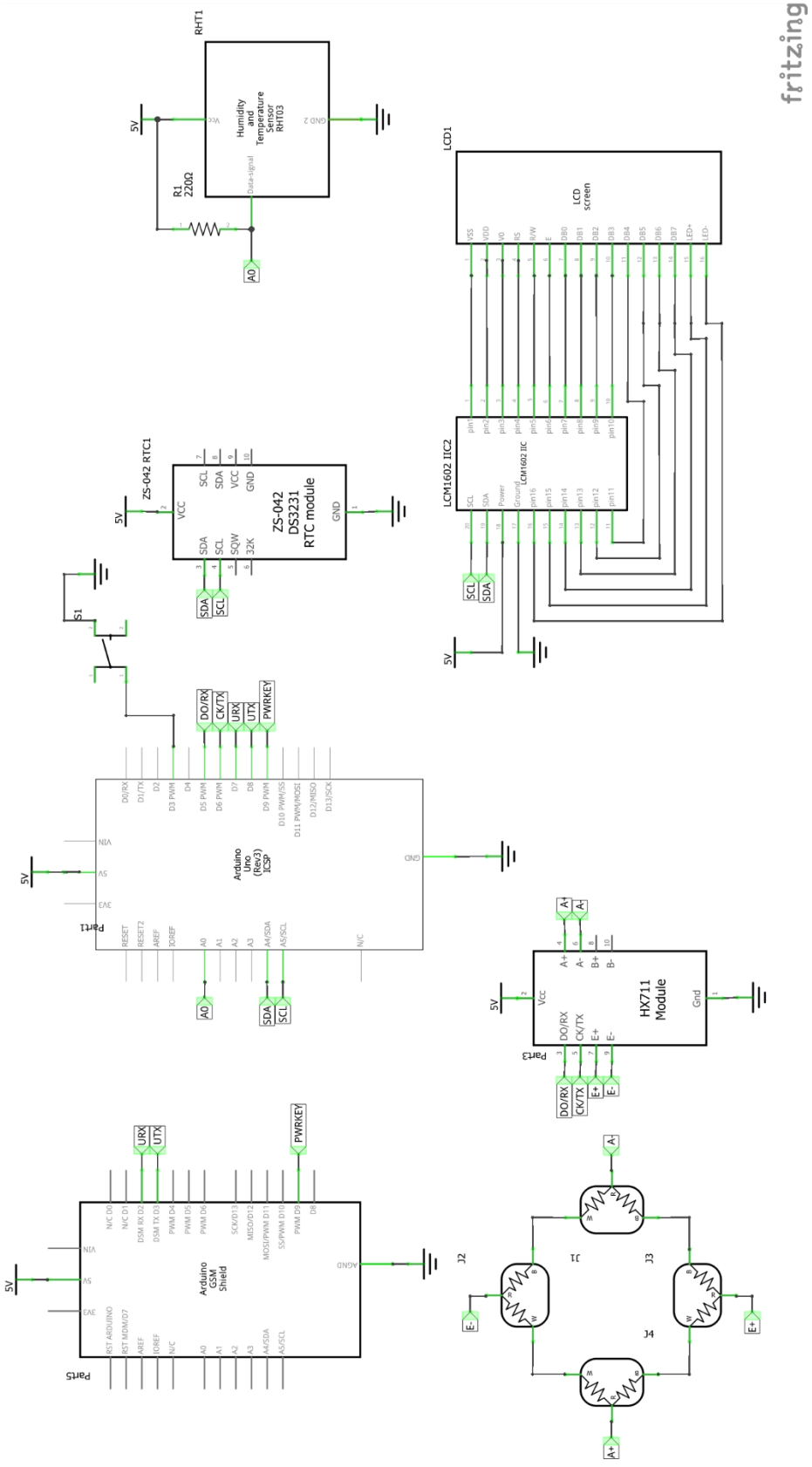
This paper explains how to create a beekeeper SMS scales using the Arduino IDE platform. Through this work, we are introduced to the basics of beekeeping, the products that are created, and the basic tools and accessories used. Beekeeper SMS scale is one of the basic tools of beekeepers. In addition to the mass of the hive, the SMS scale measures the temperature and humidity of the surrounding air. It sends measured data via SMS to your mobile device. Such beekeeper SMS scales allow the control of the hive over long distances. The advantage of using sms scales is to save time and money.

KEYWORDS:

Arduino, Beekeeper SMS scale, mass measurement, measuring ambient air temperature and humidity

ŽIVOTOPIS

Sebastian Schneider rođen je 23.studenog 1995. godine u Sisku. Završio je osnovnu školu Stjepana Kefelje u Kutini. Nakon završene osnovne škole upisao je Tehničku školu Kutina. Sudjelovao je u projektu SOELA tijekom školovanja u Tehničkoj školi Kutina. Maturirao 2014. godine i upisao preddiplomski sveučilišni studij elektrotehnike na Elektrotehničkom fakultetu u Osijeku. Nakon završetka preddiplomskog studija upisuje diplomski studij smjer komunikacijske tehnologije.

PRILOZI

P.2.1. Shema sklopa