

TEHNIČKA ŠKOLA RUĐERA BOŠKOVIĆA
GETALDIĆEVA 4, ZAGREB

ZAVRŠNI STRUČNI RAD
Uređaj za praćenje otkucaja srca

MENTOR: Zoran Dumančić

UČENIK: Borna Kovačević
RAZRED: 4.H

Zagreb, travanj 2021.

Sadržaj

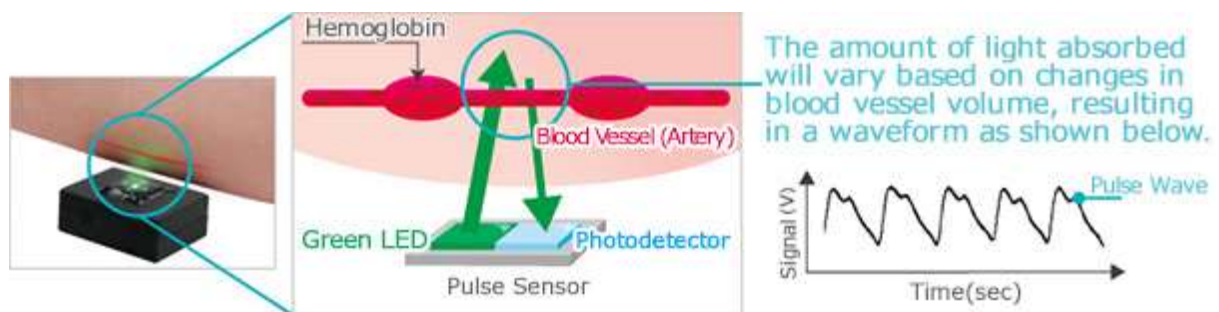
1. UVOD	3
2. Teorijski dio	4
2.1 Uređaji za mjerenje otkucaja srca	5
2.1.1 Princip mjerenja otkucaja srca pametnog sata.....	5
2.2 Visinomjer.....	6
2.3 Arduino Nano.....	7
2.3.1 Blok shema i pinout.....	8
3 Opis rada	9
3.1 Komponente	10
3.1.1 Osjetnik otkucaja srca	10
3.1.2 BMP280	11
3.1.3 Indikator	14
3.3 Izrada rada.....	15
3.3 Konačni izgled rada.....	19
4. Programski kod	21
5. Zaključak.....	25
6. Literatura	26
7. Prilog.....	27
7.1 Cjelokupni programski kod.....	27

1. UVOD

Mjerenje otkucaja srca se koristi kako bi se ustanovilo stanje čovjekovog fizičkog, ali i mentalnog zdravlja. Otkucaji srca se mijenjaju ovisno o stanju našeg uma ili tijela, ako izvodite neku fizičku aktivnost vaše srce vas mora opskrbiti svježom dozom kisika. Svaka osoba ima svoj jedinstveni uzorak otkucaja srca, ovisno o potrebi za kisikom srce određuje koliko će puta izvesti svoju radnju u jedinici vremena. Mi smo uzeli za mjeru jednu minutu i tako smo dobili mjernu jedinicu BPM (eng. Bates Per Minute) ili na hrvatskom broj otkucaja po minuti. Optički senzor funkcionira na principu da LE-dioda osvijetli kapilare i osjetnik osjeti promjenu smjera krvi odnosno njenu izmjenu. Cilj je bio omogućiti čovjeku da u svakom trenutku može izmjeriti broj otkucaja svoga srca te time ustanoviti ako što ne valja. Glavna motivacija mi je broj umrlih od posljedice srčanog udara, godišnje umire sedamnaest milijuna ljudi[1]. Što bi bilo da su imali uređaj na ruci koji bi im pokazao da nešto nije uredu s njima i sugerirao da potraže pomoć? Odabrao sam Arduino platformu radi široke dostupnosti, ali kao i vrlo lakim DIY alatom te njegovim pristupačnim cijenama. Willem Einthoven izumio je EKG 1901.g. i za to je dobio Nobelovu nagradu za fiziologiju ili medicinu 1924.g. Mjerenje nadmorske visine postalo je bitno sa prvim letovima. Određivanje visine je bitno, kako saznati koji na najviši vrh zemlje ili pak odrediti na kojoj si nadmorskoj visini ukoliko želiš izvesti neki pokus.

2. Teorijski dio

Princip rada pulsnog senzora vrlo je jednostavan. Ovaj senzor ima dvije površine, na prvu su površinu priključeni senzor za emitiranje svjetla i ambijentalne svjetlosti. Slično tome, na drugoj je površini spojen krug koji je odgovoran za poništavanje i pojačavanje buke. LED se nalazi iznad vene u ljudskom tijelu poput vrha uha ili vrha prsta, međutim, mora se nalaziti na vrhu sloja izravno. Jednom kada se LED nalazi na veni, tada LED počinje emitirati svjetlost. Jednom kad srce pumpa, tada će doći do protoka krvi unutar vena. Dakle, ako provjerimo protok krvi, onda možemo provjeriti i otkucaje srca. Ako se osjeti protok krvi, tada će senzor ambijentalne svjetlosti primiti više svjetla jer će ih reproducirati protok krvi. Ovu malu promjenu unutar dobivene svjetlosti možemo s vremenom ispitati kako bismo odredili brzinu pulsa. Na mjerenje otkucaja srca pomoću crvene ili infracrvene svjetlosti mogu utjecati infracrvene zrake sadržane u sunčevoj svjetlosti, sprječavajući stabilan rad. Iz tog razloga preporučuje se uporaba u zatvorenom ili polu-zatvorenom prostoru. Za mjerenje otkucaja srca na otvorenom poželjan je izvor zelenog svjetla koji ima visoku stopu apsorpcije hemoglobina i manje je osjetljiv na okolno svjetlo^[2].

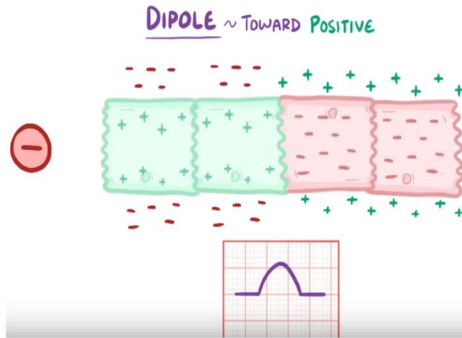


Slika 1. Osjetnik otkucaja srca

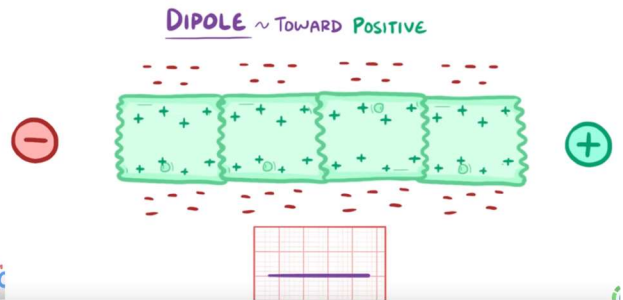
Visinomjer funkcionira na principu mjerenja tlaka oko sebe te kroz tablicu izračuna koja je visina. Postoji više načina mjerenja visine poput radarskog koji je suvremen i gotovo točan u milimetar. Postoji i relativni koji trenutno koristimo on se oslanja na tlak zraka i imamo vrstu koja je nastala početkom razvoja aviona, ta vrsta mjeri IAS, brzinu protoka zraka oko krila te tako izračunava brzinu, a ujedno mjeri i kut nosa aviona.

2.1 Uređaji za mjerenje otkucaja srca

EKG je uređaj koji vizualizira struju koja prolazi kroz srce. Kako EKG radi? Ljudske stanice su negativno polarizirane u unutrašnjosti, u odnosu na vanjski dio.



Slika 2. EKG

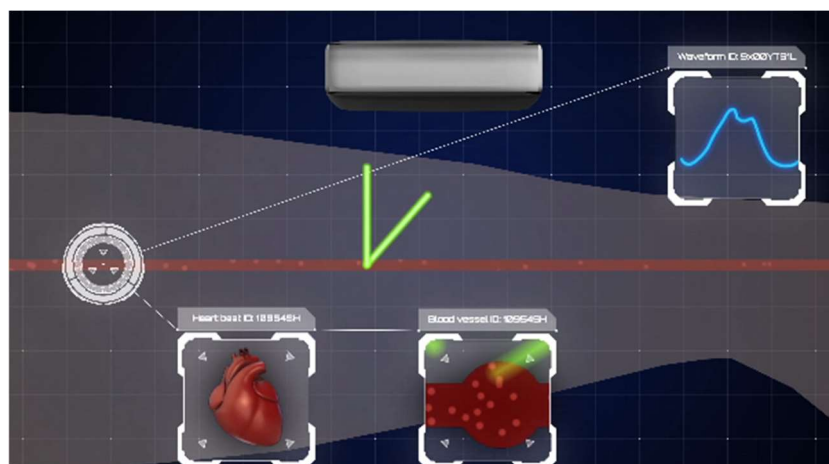


Slika 3. EKG

Brijeg odnosno impuls na dijagramu se dobije kada stanice izmjenjuju polaritete, odnosno između dvije točke (slika 2, EKG) negativne i pozitivne. Ravna crta se dobije dok se izmjenjuju svi polariteti (Slika 3, EKG)^[3].

2.1.1 Princip mjerenja otkucaja srca pametnog sata

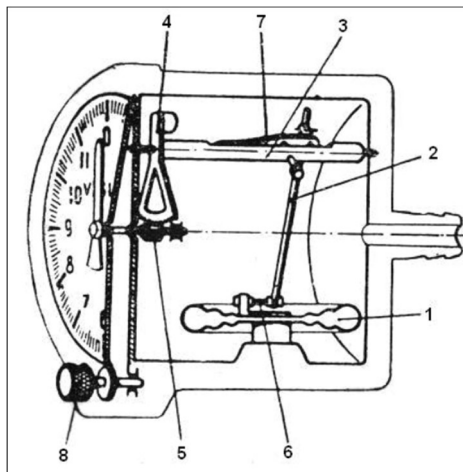
Pametni sat koristi Fotopletizmografiju kao način određivanja broja otkucaja srca. Fotopletizmografija radi na način da LE-dioda emitira svjetlost direktno u kapilare, fotosenzor prima zraku svjetlosti nazad odbijenu od kapilara (Slika 4, Senzor pametnog sata). Povratna zraka ovisi o stanju kapilara, ukoliko u njih dolazi krv povratna svjetlost će se pojačati i tako se zna razlika kod otkucaja srca^[4].



Slika 4. Senzor pametnog sata

2.2 Visinomjer

Kako visinomjer funkcionira? Koji su načini mjerenja visine? Visinomjer mjerni instrument za određivanje visine. Visinomjer ili altimeter je zapravo modificirani barometar koji očitava vrijednosti okolnog tlaka. Umjesto iznosa tlaka na određenim točkama ima visinu, pa tako nadmorskoj visini od 1219 m vrijednost tlaka je 875.105 mbar^[5].



Slika 5. Visinomjer

Popis komponenata (Slika 5, Visinomjer)

1. Aneroidna kapsula
2. Poluga
3. Osovina
4. Zupčanic
5. Zupčanic
6. Bimetalna poluga
7. Bimetalna poluga
8. Dugme

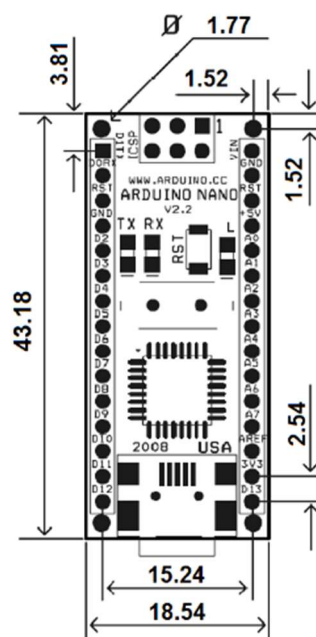
2.3 Arduino Nano

Arduino Nano je mala, kompletna pločica bazirana na Atmega328P mikroprocesoru. Napajanje dobiva preko Mini-B USB konektora. Moguće je napajanje preko Vin portova na koji se dovodi napon do 12 V i on sam regulira napon na 5 V^[6].

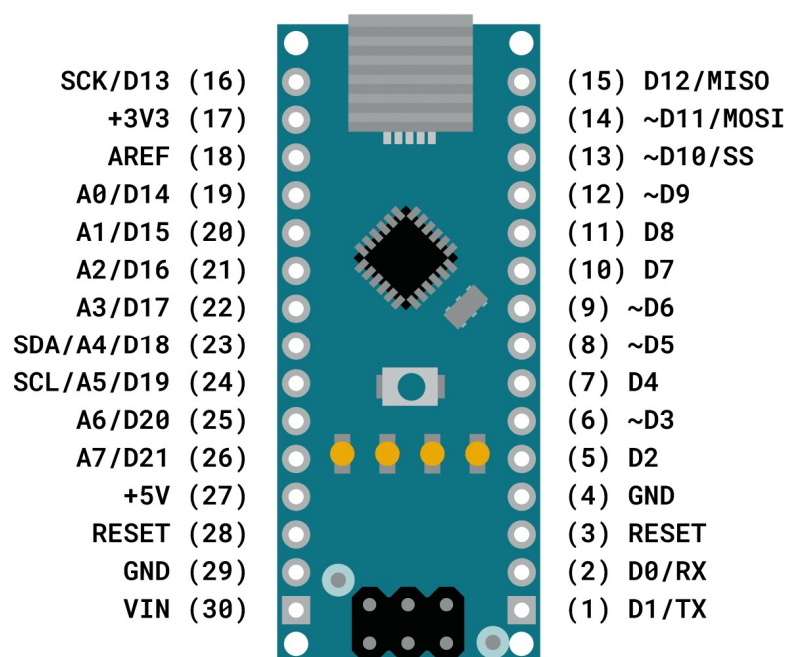
Specifikacije mikrokontrolera

- Mikrokontroler Atmega 328P
- Radni napon: 5 V
- Ulazni napon: 5-12 V
- Digitalni I/O pinovi: 14 (od kojih 6 podržava PWM izlaz)
- Analogni pinovi: 8
- DC struja po I/O pinu: 40mA
- Flash memorija: 32KB
- SRAM: 2KB
- EEPROM: 1 KB
- Brzina procesora: 16 MHz
- Dimenzije : 1.8542 x 4.318 cm
- Duljina: 43cm
- Širina 1.8cm

2.3.1 Blok shema i pinout



Slika 6. Nano dimenzije



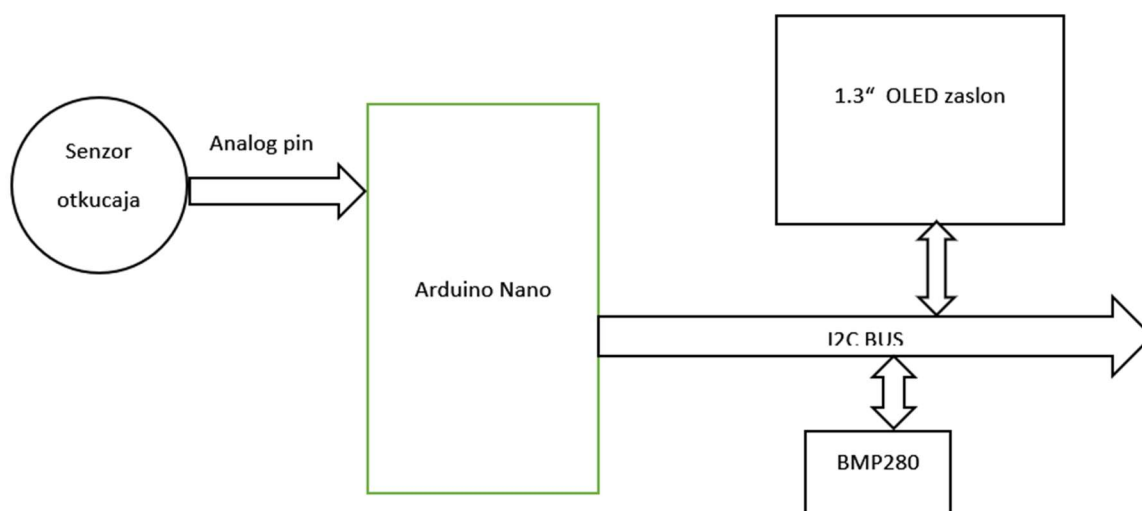
Slika 7. Arduino Nano pinovi

Pinovi (slika 7, Arduino Nano pinovi)

Broj pina	Ime	Tip	Opis
1,2, 5-16	D0-D13	I/O	Digitalni u/i portovi
3, 28	RESET	Input	Reset
4, 29	GND	PWR	Uzemljenje
17	3V3	Izlaz	+3.3 V
18	AREF	Ulaz	ADC
19-26	A7-A0	Ulaz	Analogni ulaz 0-7
27	+5 V	U/I	+5 V
30	VIN	PWR	Napon

3 Opis rada

Na Arduino Nano su spojena dva senzora, senzor otkucaja srca i visinomjer/termometar BMP280. Spojen je i 1.3 inch OLED prikazivač. Na I2C sabirnicu su spojeni OLED ekran i BMP280 pinovima SDA i SCL. OLED ekranu je potrebno 5 V, a BMP280 3.3 V. Senzor otkucaja srca je spojen svojim Signal pinom na analogni pin A0.

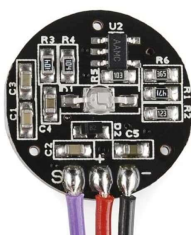


Blok shema rada

3.1 Komponente

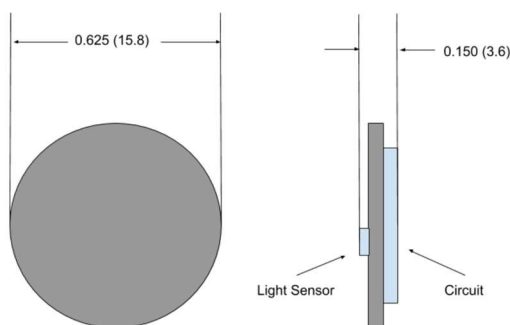
3.1.1 Osjetnik otkucaja srca

Broj Pina	Ime pina	Boja žice	Opis
1	Uzemljenje	Crna	Spojeno na uzemljenje
2	Vcc	Crvena	Spojeno na 5 V ili 3.3 V
3	Signal	Ljubičasta	Analogni izlaz



Slika 8. Senzor otkucaja srca

Senzor ima dvije strane, s jedne strane LED je smješten zajedno sa senzorom ambijentalnog svjetla, a s druge strane imamo neke sklopove. Ovaj je krug odgovoran za pojačavanje i uklanjanje šuma. LED na prednjoj strani senzora postavljen je preko vene u našem ljudskom tijelu. To mogu biti vrhovi prstiju ili vrhovi ušiju, ali ih treba staviti izravno na venu. Sada LED emitira svjetlost koja će padati izravno na venu. Vene će imati protok krvi u sebi samo kad srce pumpa, pa ako nadgledamo protok krvi, možemo pratiti i otkucaje srca. Ako se otkrije protok krvi, tada će senzor ambijentalne svjetlosti pokupiti više svjetlosti, jer će ih krv reflektirati, ta manja promjena u primljenoj svjetlosti analizira se tijekom vremena kako bi se utvrdilo otkucaji našeg srca.



Slika 9. Dimenzije senzora

3.1.2 BMP280

3.1.2.1 Opis rada BMP280

Suvremeni barometar koristi mems tehnologiju, što ga čini sposobnim za mjerenje atmosferskog tlaka u maloj i fleksibilnoj strukturi. Takvi su barometri primjenjivi na pametne telefone i Arduino ploče, kao što je BMP280 prikazan gore.

S takvom tehnologijom, barometrijski senzor tlaka MEMS nudi ne samo veću funkcionalnost, već je sposoban za dinamičko / statičko mjerenje tlaka zraka. To omogućuje da to bude popularna opcija za prijenosne meteorološke stanice i Arduino projekte^[7].

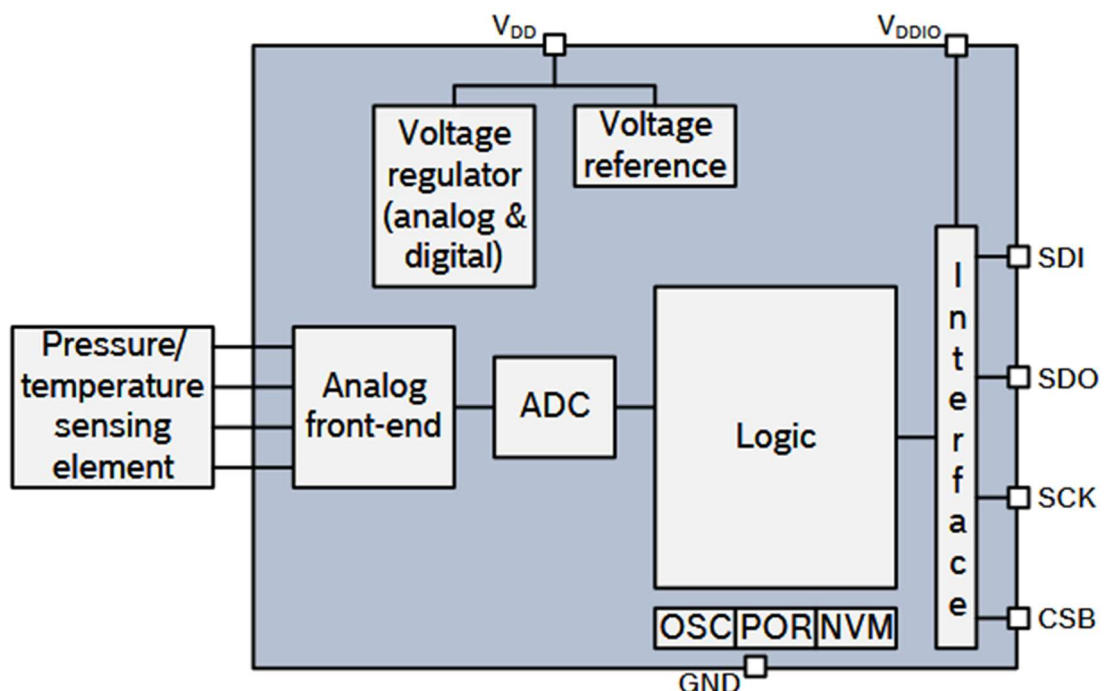
Princip rada:

Sadrži membranu koja je formirana kroz jednu kapacitivnu ploču koja je u kontaktu s atmosferom

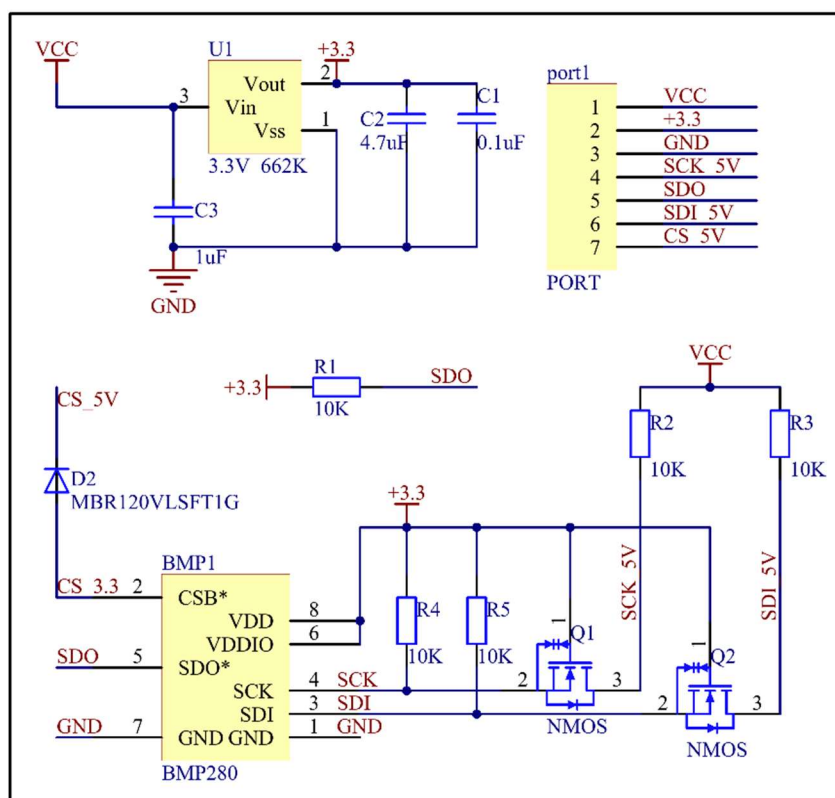
Atmosferski tlak se otkriva kroz to koliko je dijafragma deformirana uslijed rezultirajućeg tlaka

Što je pritisak veći, to se dijafragma više pomiče, što rezultira većim očitanjem barometra

3.1.2.2 Blok dijagram BMP280

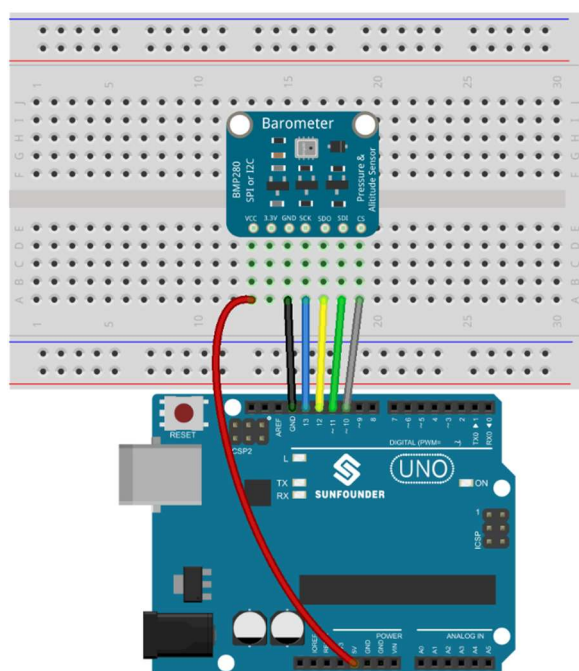


Slika 10. Blok shema BMP280



Slika 11. Električna shema BMP280

3.1.2.3 Način spajanja senzora



Slika 12. Način spjanja BMP280

3.1.2.4 Opis napajanja BMP280

Dva zasebna pina za napajanje

V_{DD} glavni pin za napajanje za integrirane analogne i digitalne blokove.

V_{DDIO} zasebni pin za napajanje, napaja digitalno sučelje.

3.1.2.5 Opis mjerenja BMP280

Mjerenje tlaka može biti i ugašeno, jer BMP280 mjeri temperaturu i tlak. Dok se mjeri postoji više načina mjerenja odnosno rezolucija i njihove točnosti^[8].

Preuzorkovanje - eng.(Oversampling)

Preuzorkovanje	Preuzorkovanje tlaka	Klasična rezolucija tlaka	Preporučena temperatura Preuzorkovanje
Preskočeno mjerenje	-	-	Ako je potrebna
Vrlo niski nivo snage	X 1	16 bit/ 2.62 Pa	X 1
Niski nivo snage	X 2	17 bit/ 1.31 Pa	X 1
Standardni nivo snage	X 4	18 bit/ 0.66 Pa	X 1
Visoki nivo snage	X 8	19 bit/ 0.33 Pa	X 1
Vrlo visoki nivo snage	X 16	20 bit/ 0.16 Pa	X 2

3.1.2.6 Potrošnja struje

Potrošnja struje ovisi o preuzorkovanju odnosno njegovom nivou.

Preuzorkovanje	Preuzorkovanje tlaka	Preuzorkovanje tlaka	Potrošnja u μA Na 1 Hz
Preskočeno mjerenje	-	-	0
Vrlo niski nivo snage	X 1	X 1	2,74
Niski nivo snage	X 2	X 1	4,17
Standardni nivo snage	X 4	X 1	7,02
Visoki nivo snage	X 8	X 1	12,7
Vrlo visoki nivo snage	X 16	X 2	24,8

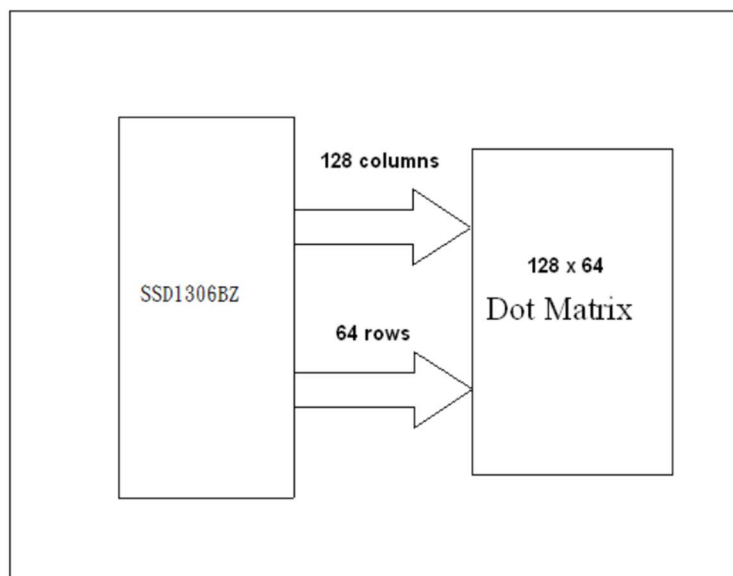
3.1.3 Indikator



Slika 13. 128*64 Indikator

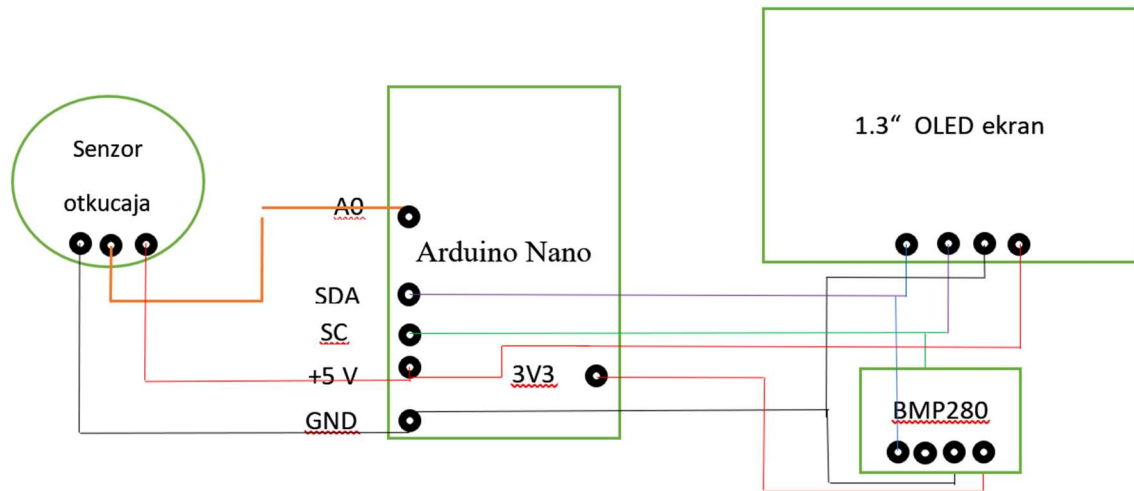
Ekran se sastoji od 128 segmenata i 64 redova. SSD1306 ugrađen je s kontrolom kontrasta, RAM-om zaslona i oscilatorom, što smanjuje broj vanjskih komponenti i potrošnju energije. Ima kontrolu svjetline u 256 koraka. Podaci / naredbe šalju se s općeg MCU-a putem paralelnog sučelja kompatibilnog sa serijom 6800/8000, I2C sučelja ili serijskog perifernog sučelja. Pogodan je za mnoge kompaktne prijenosne programe, kao što je pod-zaslon za mobitel, MP3 uređaj i kalkulator itd^[10].

3.2.3.2 Blok shema indikatora



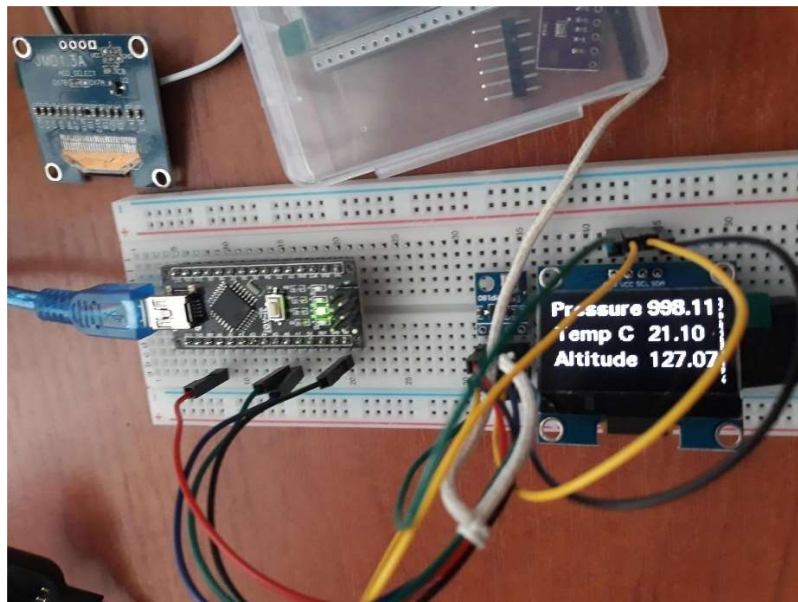
Slika 14. Blok shema indikatora

3.3 Izrada rada



Električna shema

Električna shema prikazuje način spajanja osjetnika na mikroupravljač. Prikazuje koje vrste komunikacije su korištene, te kako je uređeno napajanje pojedinih komponenata



Slika 15. Rad sa prvim kodom na eksperimentalnoj ploči

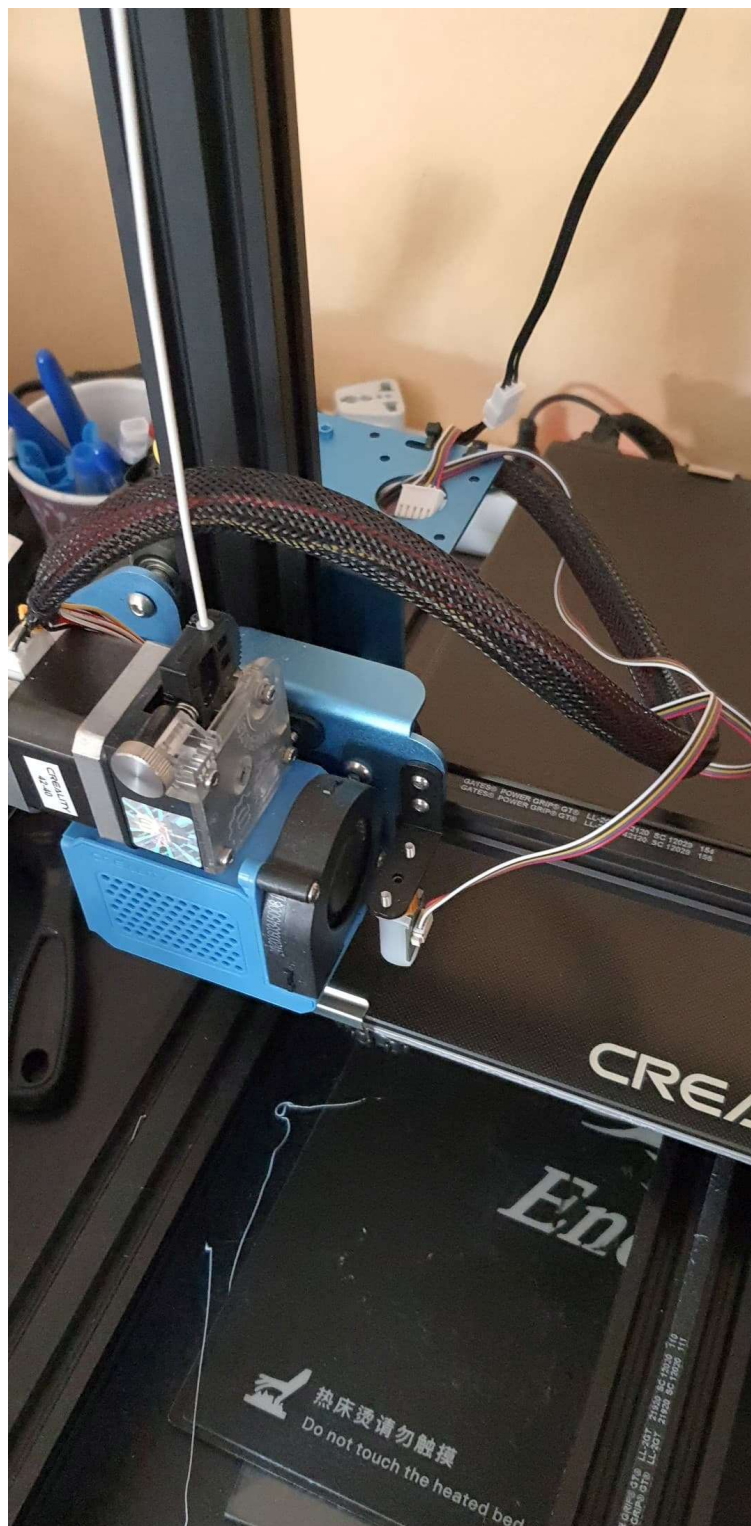
Upoznavanje sa tehnologijom i testiranje ispravnosti komponenata. Pisanje prve verzije koda za osnovne funkcije osjetnika (Slika 15.).



2

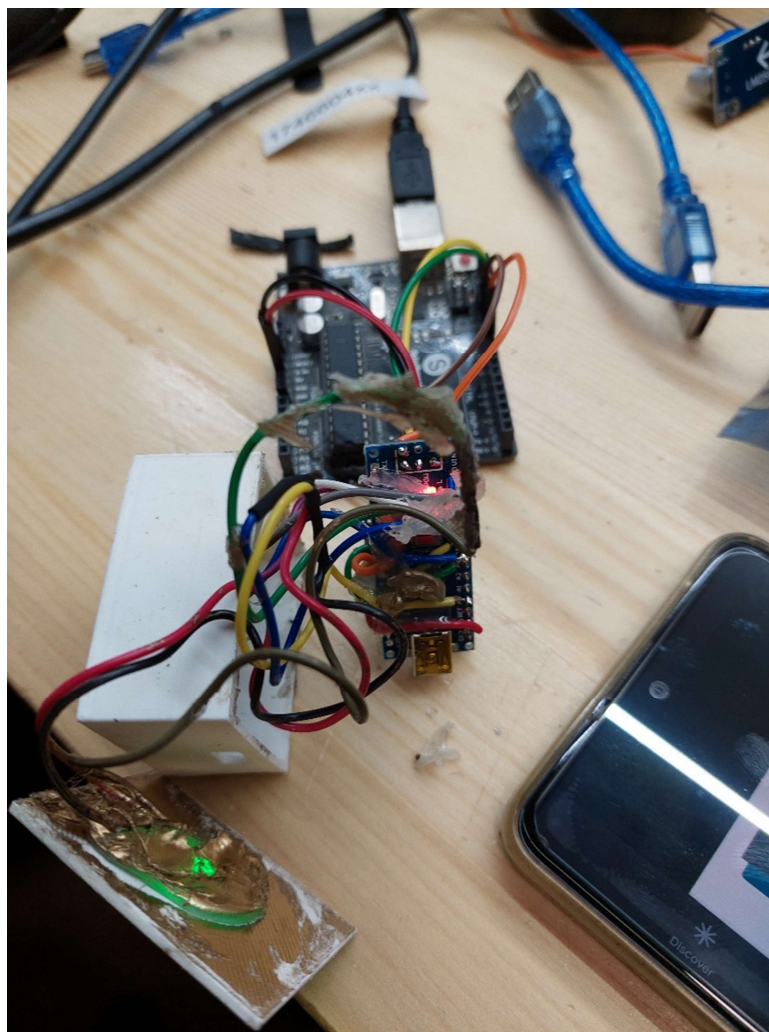
Slika 16. Zalemljene komponente

Nakon odabira komponenata, slijedilo je lemljenje i virtualno dizajniranje kako bi koja komponenta trebala biti smještena. Senzor je bilo potrebno zaštititi sa zadnje strane zbog smetnji (Slika 16.).



Slika 17. Početak 3D printanja

Početak 3D printanja, postupak je trajao 5 sati. Korištena je postavka visoke točnosti radi estetskog izgleda (Slika 17).



Slika 18. Sastavljanje

Proces sastavljanja sata, pokušaj smještanja svih komponenata u kućište. Kućište je dobro dizajnirano i sve mjere su točne. Otežavajuća stvar je bio izazov smjestiti sve u što manji prostor (Slika 18.).

3.3 Konačni izgled rada



Slika 19. Zaslون 1 od 3, Temperatura

Prvi zaslon koji se ispisuje prilikom paljenja uređaja (Slika 19.)



Slika 20. Zaslون 2 od 3, Nadmorska visina

Slika drugog zaslona koja prikazuje nadmorsku visinu (Slika 20.)



Slika 21. Zaslon 3 od 3, Broj otkucaja u minuti

Slika koja prikazuje broj otkucaja po minuti (Slika 21.)



Slika 22. Uređaj na ruci

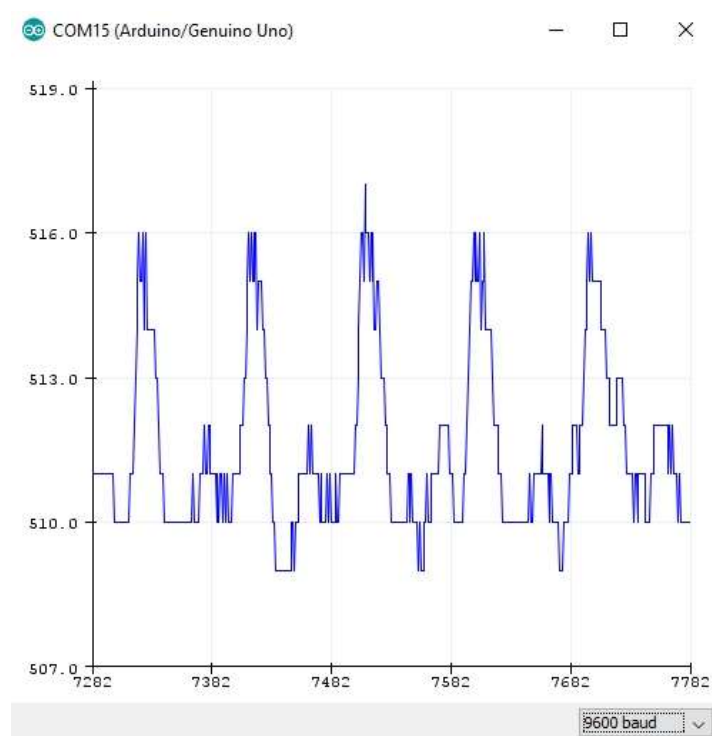
Konačni izgled uređaja na ruci narukvica je čičak traka (Slika 22.)

4. Programski kod

U ovom poglavlju su opisani najvažniji dijelovi programskog koda.

Naredba `pulseSensor.getBeatsPerMinute()` je sastavni dio datoteke `<PulseSensorPlayground.h>`

U normalnim okolnostima se koriste prekidi odnosno (eng. Interrupts). Umjesto toga koristim varijablu `threshold` koja određuje donju granicu i svi signali koji prelaze vrijednost `thresholda` su registrirani kao otkucaji (Slika 23. Serial plotter otkucaji).



Slika 23. Otkucaji srca prikazani na Seriall plotter-u

Dijelovi koda

```
void setup() {  
  Serial.begin(9600);  
  if (!bmp.begin()) {  
    Serial.println(F("Nema BMP280!!!")); //Postoji li BMP280  
    while (1);  
  }  
  display.begin(SH1106_SWITCHCAPVCC, 0x3C);  
  display.clearDisplay();  
  bmp.setSampling(Adafruit_BMP280::MODE_NORMAL, /* Operating Mode.  
  */  
    Adafruit_BMP280::SAMPLING_X2, /* Temp. preuzrokovanje */  
    Adafruit_BMP280::SAMPLING_X16, /* Pressure preuzrokovanje */  
    Adafruit_BMP280::FILTER_X16, /* Filtering. */  
    Adafruit_BMP280::STANDBY_MS_500); /* Standby time. */  
  
  bmp_temp->printSensorDetails();  
  pulseSensor.analogInput(PulseWire);  
  pulseSensor.setThreshold(Threshold);  
  pulseSensor.begin();  
}
```

U void funkciji pokrećemo glavne komponente, inicijaliziramo ih. Testiramo ispravnost konekcije komponenata jednostavnom if () naredbom npr. If(BMP neaktivan){radi to i to}. Isto tako određivanje preuzrokovanja za visinomjer.

```
void loop() {  
    info();  
    bpm();  
}
```

U `void loop()` svedeno je na minimum taj segment se ponavlja stalno i traži informacije od drugih funkcija.

```
void bpm() {  
    int otkucaji = pulseSensor.getBeatsPerMinute();  
    display.clearDisplay();  
    display.setTextSize(2);  
    display.setTextColor(WHITE);  
    display.setCursor(0, 0);  
    display.println("BPM:");  
    display.setTextSize(3);  
    display.setTextColor(WHITE);  
    display.setCursor(50, 25);  
    display.println(otkucaji);  
    display.display();  
    delay(2000);  
}
```

`Void bpm()` funkcija je zaslužna za dobivanje broja otkucaja srca i istovremeno `display.println(otkucaji)` ispisuje broj otkucaja na ekranu.

```

void info() {
    sensors_event_t temp_event, pressure_event;
    bmp_temp->getEvent(&temp_event);
    bmp_pressure->getEvent(&pressure_event);
    display.clearDisplay();
    display.setTextSize(1);
    display.setTextColor(WHITE);
    display.setCursor(0, 0);
    display.println("Temperature = ");
    display.setTextSize(2);
    display.setCursor(28, 16);
    display.print(temp_event.temperature);
    display.println("*C");
    display.display();
    delay(2500);
    display.clearDisplay();
    display.setTextSize(1);
    display.setTextColor(WHITE);
    display.setCursor(0, 0);
    display.println("Nadmorska visina: ");
    display.setTextSize(2);
    display.setCursor(28, 16);
    display.print(bmp.readAltitude(1013.25));
    display.print("m");
    display.display();
    delay(2000);
}

```

Void info() funkcija ispisuje temperaturu i nadmorsku visinu.

5. Zaključak

U radu sam koristio senzore BMP280, senzor otkucaja srca, Arduino Nano mikroupravljač. Cilj je bio odrediti broj otkucaja srca osobe, koristeći gore navedene komponente. Uređaj je treba mjeriti otkucaje srca osobe u realnom vremenu i ne pohranjivati podatke, uz to čitao bi ambijentalnu temperaturu oko osobe te nadmorsku visinu na kojoj se nalazi. Problemi na koje sam naišao su bili tehničke prirode. Odstupanja senzora su bila prevelika, sama platforma je dosta nepogodna za točno očitavanje vrijednosti. Slaba procesorska snaga uz kombinaciju malo programske memorije mi je zadalo dosta problema oko skraćivanja koda i smanjenja funkcionalnosti. Za ovakav projekt je puno bolja UNO platforma, koja doduše povećava fizičke dimenzije samog rada. Senzor otkucaja srca komponenta koja zahtijeva dosta struje. Ona također nije medicinska oprema i njezina odstupanja su $\pm 20\%$ što nikako nije dobro za medicinske svrhe. BMP280 senzor je stabilna komponenta koja čita atmosferski tlak pomoću MEMES senzora i očitava ambijentalnu temperaturu, temperaturna očitavanja su bila točna i precizna, dok nadmorska visina je varirala odnosno o temperaturu. Točnost BMP280 je bila $\pm 5\%$ što je zanemarivo. Takav rad bi se trebao raditi na drugačiji način. Integriranim krugovima i PCB pločicama. Zbog kompaktnosti, cijene, funkcionalnosti i naposljetku samog napajanja. Nakon ovog projekta sam naučio, kako funkcionira BMP280, senzor otkucaja srca, ali i Arduino Nano njegove mane i prednosti. Za rad je potrebno dosta vremena, za samu pripremu je potrebno dosta vremena. Morao sam naučiti lemiti, 3D dizajn u Fusion 360 programu, kao i arhitekturu samog Arduina Nana. Do sada nisam imao potrebu tražiti određene informacije o senzorima, mikroupravljačima sve smo imali na raspolaganju. Izazov je pronaći jednu informaciju u pdf datoteci od tristo stranica. U samom radu je bitna i kolegijalnost. Ovo je samostalan rad, ali ako sam proveoš sate bez napretka, potrebno je potražiti pomoć, ne da ti netko napravi završni rad, nego da ti objasni možda je imao isti problem. Jako je bitna komunikacija s mentorom, potrebno je uspostaviti profesionalni odnos i ponašati se tako. Mentor nije osoba koja tebi radi rad, u jednom trenutku o svom radu ti kao osoba koja ga izrađuje, znaš više od mentora. Kad bih morao napraviti istu stvar, sa stečenim znanjem pronašao bih milijun boljih načina od ovog.

6. Literatura

1. https://www.who.int/cardiovascular_diseases/resources/atlas/en/
(pristupljeno 15.4.2021.)
2. <https://www.elprocus.com/pulse-sensor-working-principle-and-its-applications/>
(pristupljeno 15.4.2021.)
3. <https://www.rohm.com/electronics-basics/sensor/pulse-sensor>
(pristupljeno 15.4.2021.)
4. <https://youtu.be/QTniLL0sQtA> (pristupljeno 15.4.2021.)
5. https://hr.wikipedia.org/wiki/Visinomjer#cite_note-1 (pristupljeno 15.4.2021.)
6. <https://www.makerguides.com/arduino-nano/> (pristupljeno 27.4.2021.)
7. <https://www.rohm.com/electronics-basics/sensor/barometric-pressure-sensor>
(pristupljeno 15.4.2021.)
8. BOCH službena stranica
<https://pdf1.alldatasheet.com/datasheet-pdf/view/1132069/BOSCH/BMP280.html>
(pristupljeno 17.4.2021.)
9. http://wiki.sunfounder.cc/index.php?title=BMP280_Pressure_Sensor_Module
(pristupljeno 23.4.2021)
10. Službena stranica indikatora, <https://cdn-shop.adafruit.com/datasheets/SSD1306.pdf>
(pristupljeno 28.4.2021)

7. Prilog

7.1 Cjelokupni programski kod

```
// ekran
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SH1106.h>
//Bosch
#include <Adafruit_BMP280.h>
Adafruit_BMP280 bmp; // use I2C interface
Adafruit_Sensor* bmp_temp = bmp.getTemperatureSensor();
Adafruit_Sensor* bmp_pressure = bmp.getPressureSensor();
//Pulse sensor
#define USE_ARDUINO_INTERRUPTS true
#include <PulseSensorPlayground.h>
const int PulseWire = 14; // analogni pin
int Threshold = 550; //osjetljivost
//ekran reset pin
#define OLED_RESET -1
Adafruit_SH1106 display(OLED_RESET);
PulseSensorPlayground pulseSensor;

void setup() {
  Serial.begin(9600);
  if (!bmp.begin()) {
    Serial.println(F("Nema BMP280!!!!")); //Postoji li BMP280
    while (1);
  }
  display.begin(SH1106_SWITCHCAPVCC, 0x3C);
  display.clearDisplay();
  bmp.setSampling(Adafruit_BMP280::MODE_NORMAL, /* Operating Mode.
  */
```

```

Adafruit_BMP280::SAMPLING_X2,    /* Temp. preuzrokovanje */
Adafruit_BMP280::SAMPLING_X16,   /* Pressure preuzrokovanje */
Adafruit_BMP280::FILTER_X16,     /* Filtering. */
Adafruit_BMP280::STANDBY_MS_500); /* Standby time. */

bmp_temp->printSensorDetails();
pulseSensor.analogInput(PulseWire);
pulseSensor.setThreshold(Threshold);
pulseSensor.begin();
}

void loop() {
    info();
    bpm();
}

void bpm() {
    int otkucaji = pulseSensor.getBeatsPerMinute();
    display.clearDisplay();
    display.setTextSize(2);
    display.setTextColor(WHITE);
    display.setCursor(0, 0);
    display.println("BPM:");
    display.setTextSize(3);
    display.setTextColor(WHITE);
    display.setCursor(50, 25);
    display.println(otkucaji);
    display.display();
    delay(2000);
}

void info() {
    sensors_event_t temp_event, pressure_event;
    bmp_temp->getEvent(&temp_event);

```

```
    bmp_pressure->getEvent(&pressure_event);
    display.clearDisplay();
    display.setTextSize(1);
    display.setTextColor(WHITE);
    display.setCursor(0, 0);
    display.println("Temperature = ");
    display.setTextSize(2);
    display.setCursor(28, 16);
    display.print(temp_event.temperature);
    display.println("*C");
    display.display();
    delay(2500);
    display.clearDisplay();
    display.setTextSize(1);
    display.setTextColor(WHITE);
    display.setCursor(0, 0);
    display.println("Nadmorska visina: ");
    display.setTextSize(2);
    display.setCursor(28, 16);
    display.print(bmp.readAltitude(1013.25));
    display.print("m");
    display.display();
    delay(2000);
}
```