

Odri Tomas 0036476538  
Maja Ovčarik 0036479064  
Andrea Matić 0036477689  
Nikola Vrebčević 0036479059  
Katarina Karužić 0036478345

SEMINARSKI RAD - SPVP



Fakultet elektrotehnike i računarstva, Sveučilišta u Zagrebu  
Zavod za elektroničke sustave i obradu informacija  
Sveučilište u Zagrebu

# Detektor mjesečarenja

- Δ Osobe s problemom mjesečarenja ili problemom demencije
- Δ Programiranje Arduina i Raspberry PI, Linux, Mrežna infrastruktura
- Δ Komunikacija s senzorima, obrada slike
- Δ Ultrazvučni senzor, ESP8266, Kamera

7. lipnja 2017



## Sažetak

Detektor mjesečarenja jedan je od *assisted living* projekata, razvijen je s ciljem da poveća sigurnost mjesečara ili dementne osobe pravovremenim obavješćavanjem ukućana o kretanju tijekom noći. Ako je riječ o nadzoru osobe koja mjesečari, sustav za alarmiranje skrbnika pali se nakon što je promatrana osoba ustala iz kreveta i izašla kroz vrata. Drugi način rada, predviđen za starije osobe, te alarmira ukućane po samom ustajanju ili padu iz kreveta.

## Sadržaj

1.	UVOD .....	3
2.	OPIS SUSTAVA .....	4
2.1.	Arduino Mega 2560 .....	4
2.2.	Arduino Uno .....	5
2.3.	ESP8266 .....	6
2.4.	Ultrazvučni modul .....	8
2.5.	Raspberry Pi 3 model B.....	9
2.6.	Kamera.....	10
2.7.	3.5 mm audio priključak i zvučnik .....	10
3.	PROGRAMSKA PODRŠKA.....	11
3.1.	Detekcija pokreta obradom slike .....	11
3.1.1.	Processing .....	11
3.1.2.	Detekcija pokreta .....	12
3.2.	Spajanje Raspberry Pi na kućnu bežičnu mrežu.....	15
3.3.	Detekcija ultrazvučnim senzorom .....	16
3.4.	Komunikacija ESP8266 s Raspberry PI 3 .....	17
3.4.1.	MQTT protokol .....	17
3.4.2.	Mosquitto .....	17
3.4.3.	Paho-mqtt .....	17
3.4.4.	Ostvarivanje komunikacije.....	18
3.4.5.	Programiranje ESP8266.....	19
3.4.6.	Zapisnik aktivnosti.....	20
3.5.	Dojava o pokretu na jedinicu za alarmiranje .....	20
3.6.	Cron .....	21
4.	ZAKLJUČAK.....	23
5.	LITERATURA.....	24
6.	POJMOVNIK .....	25
7.	RASPODJELA ZADATAKA.....	26

Ovaj seminarski rad je izrađen u okviru predmeta „Sustavi za praćenje i vođenje procesa“ na Zavodu za elektroničke sustave i obradbu informacija, Fakulteta elektrotehnike i računarstva, Sveučilišta u Zagrebu.

Sadržaj ovog rada može se slobodno koristiti, umnožavati i distribuirati djelomično ili u cijelosti, uz uvjet da je uvijek naveden izvor dokumenta i autor, te da se time ne ostvaruje materijalna korist, a rezultirajuće djelo daje na korištenje pod istim ili sličnim ovakvim uvjetima.

## 1. Uvod

Zdrave, kao i mnoge dementne osobe, nesvjesno ustaju iz kreveta i odlaze iz sobe, ponekad čak i iz kuće. Takvim događajima nesvjesno ugrožavaju sami sebe. Kad se uzme u obzir koliko su česta mjesečarenja kod male djece, te koliko često se događa da dementna osoba pri ustajanju iz kreveta padne, njihovim bližnjim je vrlo teško pratiti njihovo stanje. Najveći problem je pratiti njihovo stanje noću. Upravo zbog toga, unutar *assisted living* projekata našao se i ovaj sustav namijenjen detekciji mjesečarenja.

Sustav podržava dva načina rada, čiji se odabir vrši pri samom postavljanju sustava. Ako je riječ o prvom načinu rada, odnosno nadzoru osobe koja mjesečari, sustav za alarmiranje skrbnika pali se nakon što je promatrana osoba ustala iz kreveta i izašla kroz vrata. Drugi način rada koji je predviđen za starije osobe i osobe s demencijom, alarmira ukućane po samom ustajanju (ili padu) iz kreveta. Predstavljenim slijedom provjera osigurano je da sustav ne registrira stvari poput okretanja osobe na krevetu, prolaska kućnih ljubimaca, ulaska drugih osoba u sobu te sličnih lažnih uzbuna. Ovime je osiguran miran i bezbrižan san ukućanima i sigurnost promatranim osobama.

Prednost sustava je korištenje malog broja senzora, koji sudjeluju u prikupljanju informacija iz okoline o promatranoj osobi. Budući da je jedan od senzora kamera, negativna strana sustava je to što je snimanje osobe uvelike određeno prostorijom u kojoj se nalazi, te se video sustav mora dodatno kalibrirati.

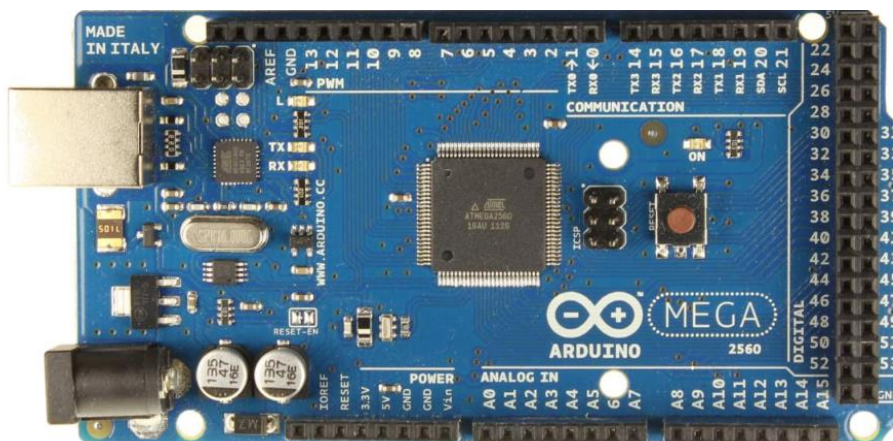
## 2. Opis sustava

Sustav za detekciju mjesečarenja se sastoji od:

- Arduino Mega 2560
- Arduino Uno
- Dva Wi-Fi modula ESP8266
- Ultrazvučni modul HC-SR04
- 3.5 mm Audio priključak
- Zvučnik
- Raspberry Pi 3
- Kamera

### 2.1. Arduino Mega 2560

Arduino Mega 2560 je razvojna platforma otvorenog tipa, koja sadrži 54 GPIO pina od kojih se 15 može koristiti kao PWM izlaz, a 16 kao analogni ulaz. Također sadrži 4 sučelja za UART komunikaciju, priključak za USB i ICSP kao i tipku za reset. Mikrokontroler koristi kristalni oscilator s frekvencijom 16 MHz. Napajanje je izvedeno preko AC/DC adaptera 9-12 V ili preko USB priključka. Napon koji koriste pinovi je 5 V. Na Slika 1 je prikazana spomenuta platforma.



Slika 1. Arduino Mega 2560

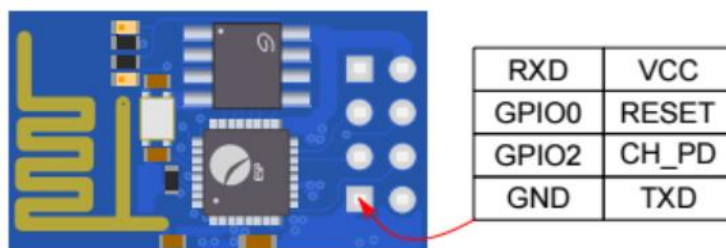
## 2.2. Arduino Uno

Arduino Uno je razvojna platforma otvorenog tipa koja se temelji na Atmega328P mikrokontroleru. Sadrži 14 digitalnih ulazni/izlaznih pinova od kojih se 6 može koristiti kao PWM izlaz, a 6 kao analogni ulaz. Sadrži tipku za reset, priključak za USB, priključak za napajanje i ICSP, kao i 13 ugrađenih led indikatorskih svjetala. Kao i prethodno opisan Arduino Mega 2560, također koristi kristalni oscilator s frekvencijom 16 MHz. Preporučuje se napajanje od 7-12 V ili preko USB priključka spojenog na računalo. Na Slika 2. prikazana je spomenuta razvojna platforma.



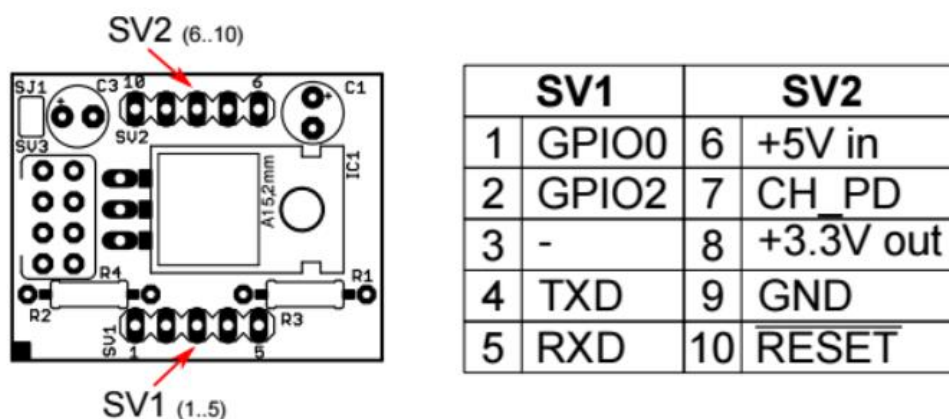
Slika 2. Arduino Uno

### 2.3. ESP8266



Slika 3. Modul ESP8266

Modul ESP8266 (Slika 3) tvrtke *Espressif Systems* je zamišljen kao jeftino, malo i brzo rješenje za sve projekte koji koriste 802.11b/g/n standard odnosno Wi-Fi komunikaciju. Korištena je jedna od nekolicine različitih izvedbi ovog modula ESP-01. Ova izvedba ima samo 2 GPIO pina. Radi na 3.3V zbog čega je potrebno koristiti regulator napajanja. Također potrebno je provesti prilagodbu naponskih razina za serijsku komunikaciju, odnosno pinove RXD i TXD. Prilagodba je provedena pomoću ESP8266 adaptera (Slika 4). Navedeni modul je kompatibilan s razvojnim okruženjem Arduino IDE uz korištenje dodatno skinutih biblioteka.



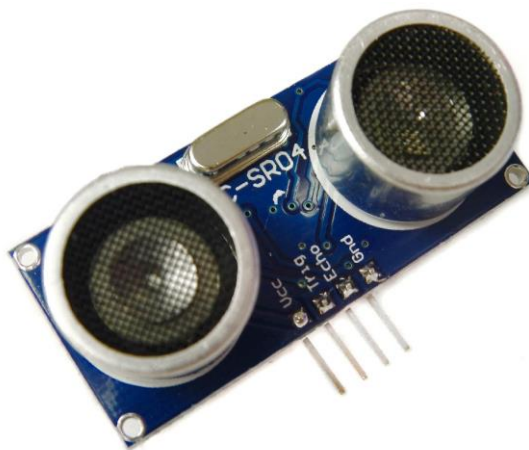
Slika 4. Adapter za ESP8266

Ako se izvršni kod na ESP8266 prenosi uz pomoć Arduina, potrebno je modul postaviti u način rada za prijenos koda putem UART komunikacije. Naime, potrebno je spojiti izvode 4 i 5 s ESP8266 adaptera na Rx i Tx Arduina, tako da je Rx s adaptera spojen na Rx Arduina isto tako i u slučaju Tx. Također je potrebno spojiti GPIO0 odnosno izvod 1 s adaptera na GND, te izvod 10 na izvod 8, odnosno na 3.3V. Izvod Rst na Arduinu, treba



biti spojen na GND. Priključivanjem Arduina na računalo, na priključeni ESP8266 potrebno je izvod 10 s 3.3 V prebaciti na GND, te ponovno vratiti na 3.3V. Sada je ESP8266, spreman za prijenos izvršnog koda putem UART komunikacije. Pri samom prijenosu programa na modulu se može vidjeti paljenje i gašenje plavog led indikatorskog svijeta.

## 2.4. Ultrazvučni modul



Slika 5. Ultrazvučni modul HC-SR04

Ultrazvučni senzor (engl. ultrasonic sensor) omogućava mjerenje udaljenosti između senzora i objekta. Korišteni ultrazvučni modul HC-SR04 (Slika 5) omogućava mjerenje u rasponu od 2 do 400 cm uz preciznost do 3 mm u idealnim uvjetima. Sastoji se od ultrazvučnog prijemnika i odašiljača te popratne kontrolne elektronike. Modul ima 4 pina. Prvi pin je VCC -napajanje modula (5 V), sljedeći je Trig – aktiviranje mjerenja, treći je Echo – signal odbijen od prepreke i zadnji je GND – uzemljenje. Princip rada senzora je sličan kao i kod radarskih sustava koji mjere udaljenost objekta uz pomoć vremena koje je proteklo od slanja vala i njegova povratka. U ovom slučaju prvi korak je aktivacija modula slanjem kontrolnog impulsa, modul zatim generira osam ultrazvučnih impulsa frekvencije 40 kHz. Detekcijom povratnih ultrazvučnih impulsa, modul generira izlazni signal čija je dužina proporcionalna udaljenosti. U samom izvršnom kodu, potrebno je preračunati trajanje impulsa u dužinu pomoću formule:

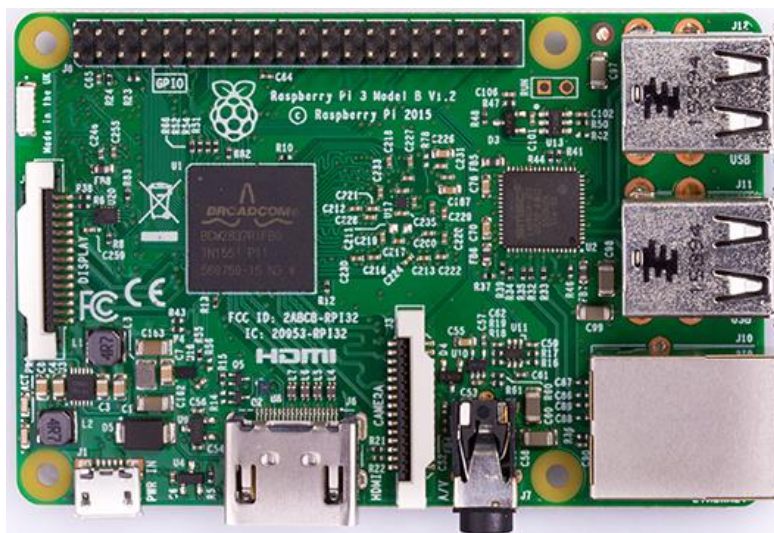
$$D_{cm} = \frac{duration(\mu s)}{brzina\ zvuka(\frac{340m}{s} = 29\mu s)/2}$$

Jednadžba 1. Izračun udaljenosti



## 2.5. Raspberry Pi 3 model B

Raspberry Pi 3 model B (Slika 6) je malo ugradbeno računalo koje u svojoj sastavnici sadrži sve komponente kao osobno ili prijenosno računalo: USB priključci, HDMI priključak, Ethernet priključak, RCA zajednički priključak za zvuk i sliku, DSI priključak za paralelnu vezu s prikaznikom i CSI priključak za paralelnu vezu s kamerom, izvodi opće namjene, integrirani 802.11b/g/n Wireless LAN modul, Bluetooth 4.1 i Bluetooth Low Energy. Kruna ovog malog računala jest centralna upravljačka jedinica koju čini 64-bitni četverojezgreni ARMv8 procesor koji radi na frekvenciji takta od 1.2 GHz. Ako se na Raspberry Pi 3 spajaju periferijske komponente poput kamere, ekrana ili slično, potrebno je istosmjerno napajanje iznosa 5 V i jakosti struje 2.5 A.



Slika 6. Raspberry Pi 3 model B

Kako bi posao upravljanja računalom bio jednostavniji te kako bismo mogli iskoristiti postojeće tehnologije u razvoju novih programskih podrški, Raspberry Pi 3 kao osnovni program pokreće operacijski sustav Raspbian Jessie (verzija: *April 2017*). Raspbian Jessie je operacijski sustav otvorenog koda baziran na Debian distribucijama Linuxa te je dodatno optimiziran za sklopovlje Raspberry Pi računala. Podržava brojne pakete standardnih Linux distribucija, što omogućava korištenje gotovo svih tehnologija.

## 2.6. Kamera

Za prosljeđivanje sličica videa u program za obradu slike uporabljena je web kamera TRACER Gamma Webcam (Slika 7). Kamera se na računalo spaja pomoću USB 2.0 protokola. Rezolucija kamere iznosi 640x480 piksela te omogućava snimanje 30 sličica u sekundi.



Slika 7. TRACER Gamma Webcam

## 2.7. 3.5 mm audio priključak i zvučnik

Za obavijest skrbnika o mjesečarenju ili padu promatrane osobe, korišten je 3.5 mm audio priključak spojen na Arduino u koji je priključen zvučnik. Audio priključak ima dva izvoda, jedan predstavlja uzemljenje a drugi se priključi na izvod Arduina. Na tom izvodu, pomoću programa na Arduinu stavi se melodija koja tada svira na zvučniku.

### 3. Programska podrška

Prethodno opisani sustav vrši naredbe zapisane u programskoj podršci koja se može podijeliti na:

- Detekcija pokreta obradom slike
- Spajanje Raspberry Pi na kućnu bežičnu mrežu
- Detekcija ultrazvučnim senzorom
- Komunikacija ESP8266 s Raspberry Pi 3
- Dojava o pokretu na jedinicu za alarmiranje

#### 3.1. Detekcija pokreta obradom slike

Svrha ovog dijela sustava je da pomoću kamere snima osobu koja ima epizode mjesečarenja te da iz njezinih pokreta odredi trenutak kada je osoba počela mjesečariti.

##### 3.1.1. Processing

Processing je programski jezik otvorenog koda koji implementira mnoštvo funkcionalnosti iz domene digitalne obrade slike. Svojom jednostavnošću i mogućnostima brzog modeliranja prvenstveno je namijenjen likovnim umjetnicima kako bi svoja djela ostvarili pomoću računalne tehnologije. Osnovni programski paket proširuju brojne biblioteke koje omogućavaju rad sa sličicama videa, animaciju, dizajniranje u trodimenzionalnom prostoru i mnogo drugih mogućnosti.

Iz perspektive ovog projekta, Processing predstavlja alat koji je omogućio brzo preusmjeravanje struje videa u program koji određenim tehnikama vrši analizu nad slikama videa. Budući da Processing ipak zahtjeva određenu količinu računalnih resursa, potrebno ga je instalirati na Raspberry Pi. Prije instalacije potrebno je osigurati zadnju verziju Java JDK-a. To je moguće izvršiti naredbom:

```
sudo apt-get update && sudo apt-get install oracle-java7-jdk
```

Processing je moguće instalirati naredbom:

```
curl https://processing.org/download/install-arm.sh | sudo sh
```

Kako bi se strujanje videa uspješno kanaliziralo u Processing, potrebno je instalirati Gstreamer biblioteku naredbom:

```
sudo apt-get install gstreamer0.10-plugins-good
```

Potrebno je proširiti Processing dodatnom Video bibliotekom kako se mogle koristiti funkcije potrebne za obradu videa. Biblioteka se može dodati kroz grafičko sučelje Processinga, međutim ako se ne radi s grafičkim sučeljem, datoteku je potrebno preuzeti te smjestiti u direktorij:

```
/home/<username>/sketch/libraries/
```

Ime biblioteke mora biti *processing.video*. Adresa na kojoj je moguće preuzeti datoteku je:

```
https://processing.org/reference/libraries/video/index.html
```

Ako se Processing koristi bez grafičkog sučelja Raspbiana, tada je potrebno pokrenuti Processing u virtualnom prozoru pomoću programa Xvfb koji se može instalirati naredbom:

```
sudo apt-get install xvfb libxrender1 libxtst6 libxi6
```

Pokretanje Processinga pomoću Xvfb izvodi se naredbom:

```
xvfb-run /home/<username>/processing/processing-java --  
sketch=/path/to/sketch/folder --run
```

## 3.1.2. Detekcija pokreta

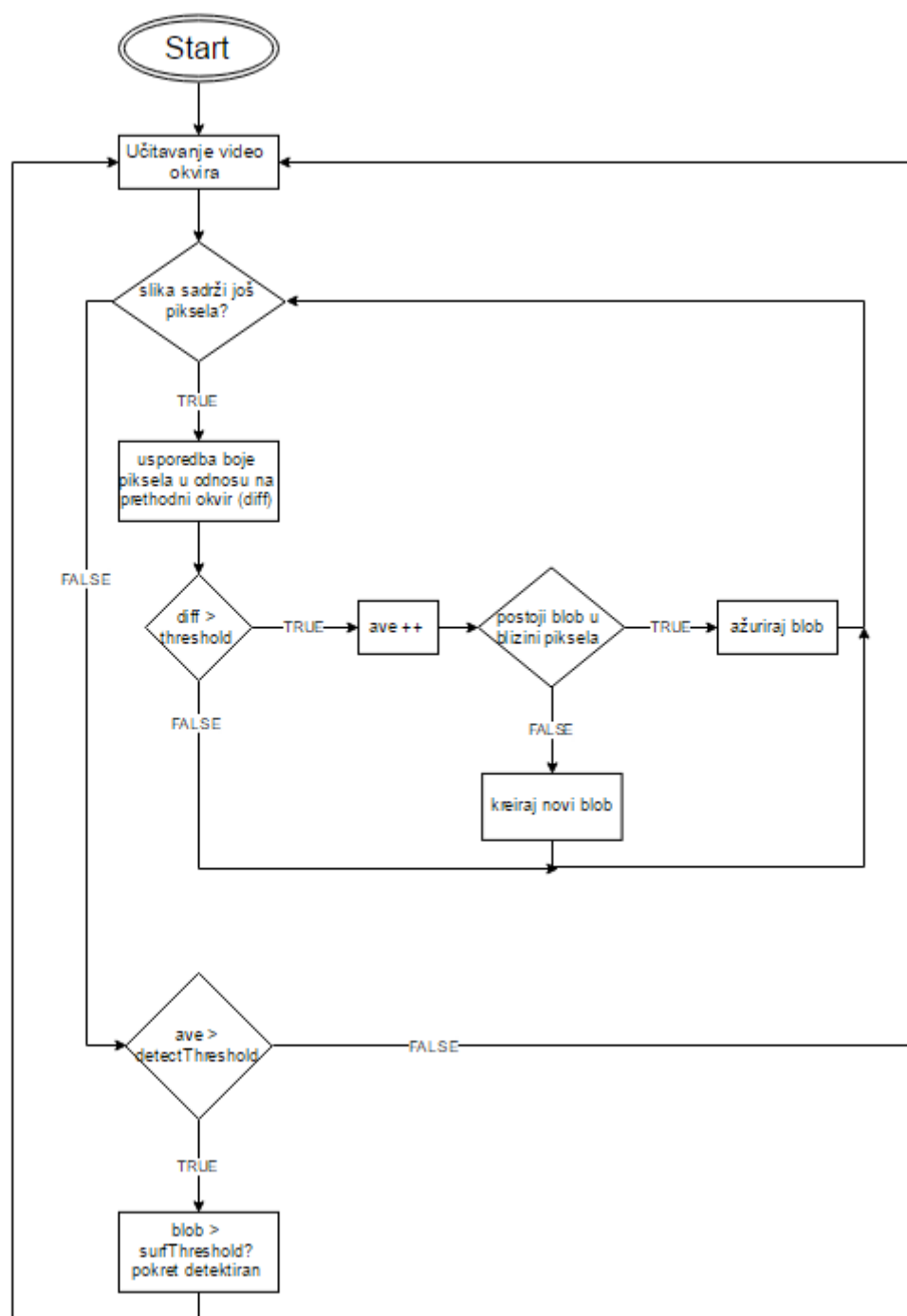
Algoritam koji detektira pokret može se podijeliti u dva dijela od kojih svaki dio predstavlja jednu razinu sigurnosti pri donošenju odluke o tome je li dogodio pokret ili ne. Potreba za više nezavisnih pokazatelja pokreta je posljedica zahtjeva da sustav detektira trenutak kada osoba koja mjesečari više nije na prostoru svojeg kreveta, nego je ona ustala ili pala s njega.

Prva razina detekcije pokreta izračunava razliku između susjednih sličica videa. Budući da se ovaj detektor modelirao pomoću kamere koja snima u boji, razliku između susjednih sličica videa potrebno je izračunavati za sva tri kanala boje (plavi, crveni i zeleni kanal). Razlika se određuje za svaki piksel trenutne sličice videa u odnosu na prošlu sličicu te se određuje broj piksela čija je razlika veća od predodređene vrijednosti. Konačno, prethodno određeni broj piksela čija je razlika za trenutnu i sadašnju sliku veća od predodređene vrijednosti, uspoređuje se s brojem piksela koji bi na slici određene rezolucije predstavljali relevantni pomak. Broj piksela koji predstavljaju relevantni pomak može se predodrediti eksperimentalno uz poznavanje rezolucije kamere.

Drugi dio detekcije pokreta izračunava površinu plohe koja se pomaknula između dva kadra u struji videa. Površina se određuje tako da se gleda predodređena okolina promatranog piksela te ako se nešto promijenilo u odnosu na prethodnu sliku, tada se

dogodio pomak. Iterativno, piksel po piksel, moguće je odrediti ukupnu površinu koja se pomaknula. Izračunata površina se tada uspoređuje s iznosom površine koji se eksperimentalno odredio. Eksperimentalno određena površina ugrubo odgovara veličini ljudskog torza snimanog s metar udaljenosti.

Algoritam ocjenjuje da se pokret dogodio ako broj različitih piksela između dvije sličice videa veći od broja piksela koji predstavljaju relevantan pokret (prva razina), a zatim, ako je površina koja se pomaknula određene veličine (druga razina). Ovim slijedom provjera osigurano je da sustav ne registrira male pokrete i pokrete koji nisu relevantni (primjerice pokreti ruke, posteljine ili nekog manjeg kućnog ljubimca). Također, bitno je napomenuti da je ovaj sustav neosjetljiv na promjene u intenzitetu svjetla. Opisani algoritam prikazan je na Dijagramu 1.



Dijagram 1. Dijagram toka za prepoznavanje pokreta



### 3.2. Spajanje Raspberry Pi na kućnu bežičnu mrežu

Raspberry Pi sklopovlje ima više različitih sučelja te mu je moguće pristupiti na različite načine. Osnovni način je spajanje tipkovnice i miša na USB sučelje te direktno upravljati sustavom. Također, moguće mu je postaviti statičku IP adresu za određeno Ethernet ili WLAN sučelje (*eth0*, *wlan0*). Tako koristeći vlastito osobno računalo i poznatu IP adresu možemo upravljati Raspberry Pi računalom preko terminala koristeći SSH protokol.

Statička IP adresa se postavlja uređivanjem datoteke `/etc/network/interfaces`. Primjer programskog koda za postavljanje statičke adrese Ethernet sučelja *eth0*:

```
iface eth0 inet static
address 192.168.1.2
netmask 255.255.255.0
network 192.168.1.0
```

Na sličan način postavljamo statičku IP adresu i za Wireless LAN sučelje *wlan0*. Za spajanje na bežičnu mrežu zaštićenu lozinkom i imenom, nakon definiranja statičke adrese (npr. 192.168.0.99) pozivamo

```
wpa-roam /etc/wpa_supplicant/wpa_supplicant.conf
```

te u samu `wpa_supplicant.conf` datoteku dodajemo lozinku i ime željene mreže.

```
network={
    ssid = 'Ime_mreze'
    psk = 'zaporka'
}
```

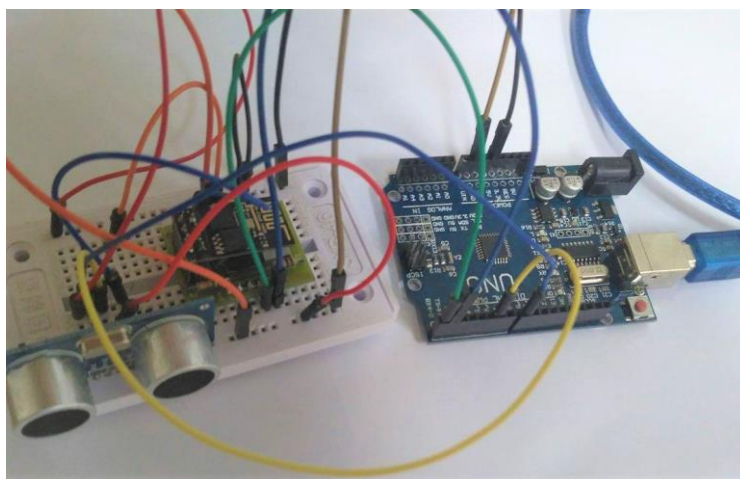
Statička IP adresa Ethernet sučelja nam je bila od velikog značaja pri spajanju osobnog računala na Raspberry Pi, a statička IP adresa WLAN sučelja je bila bitna za modul ESP8266-01 koji se spaja na Raspberry Pi kao poslužitelj (engl. server) za protokol razmjene informacije.

## 3.3. Detekcija ultrazvučnim senzorom

Ultrazvučni senzor namijenjen je za detekciju izlaska osobe iz sobe. Senzor je postavljen na vrata, te je programiran tako da šalje „1“ preko serije na ESP8266 u slučaju detekcije osobe na vratima. Potrebno je prvo inicijalizirati serijsku komunikaciju, zatim odabirom pina koji će biti spojen na izvod Trig ultrazvučnog senzora aktivirati mjerenje. Aktivira se njegovim konstantnim podizanjem u visoku razinu pa spuštanjem u nisku. Nadalje pomoću formule (Jednadžba 1) potrebno je izračunati udaljenost koju mjeri senzor, u ovom slučaju je to napravljeno funkcijom `microsecondsToCentimeters`.

```
long microsecondsToCentimeters(long microseconds) {
    return microseconds / 29 / 2;
}
```

Vrijednost varijable `duration` se dobije preko ugrađene funkcije `pulseIn` koja vraća duljinu pulsa u mikro sekundama. Ako je ta udaljenost manja od udaljenosti praga vrata, tada je osoba bila na vratima te se putem serije šalje „1“.



Slika 8. HC-SR04, ESP8266 i Arduino Uno

### 3.4. Komunikacija ESP8266 s Raspberry PI 3

#### 3.4.1. MQTT protokol

MQTT (engl. Message Queue Telemetry Transport) je protokol koji se koristi u M2M (engl. machine to machine) i IoT (engl. Internet of Things) sustavima. Dizajniran je za razmjenu jednostavnih poruka principom pretplate i objavljivanja na određenu temu (engl. topic). Protokol zahtjeva posrednika koji prevodi poruke pošiljalca te prilagođava i prosljeđuje svim primateljima. Pošiljalci su svi uređaji koji se prijavljuju za objavljivanje na određenu temu dok su primatelji svi uređaji koji slušaju određenu temu, odnosno primaju poruke s nje.

#### 3.4.2. Mosquitto

Eclipse Mosquitto je posrednik poruka za MQTT protokol (verzija 3.1 i 3.1.1.). Programski alat je otvorenog tipa te je pogodan za sve sustave kod koji su korišteni mobilni uređaji te uređaji niske snage. Instalacija Mosquitto posrednika na Raspberry Pi uređaj potrebno je pokrenuti sljedeće naredbe:

```
wget http://repo.mosquitto.org/debian/mosquitto-repo.gpg.key
sudo apt-key add mosquitto-repo.gpg.key
cd /etc/apt/sources.list.d/
```

kojima se preuzima i postavlja ključ za pristup Mosquitto repozitorije te se s:

```
sudo wget http://repo.mosquitto.org/debian/mosquitto-wheezy.list
apt-get install mosquitto
```

preuzima sami instalacijski paket Mosquitto posrednika. Potrebno je preuzeti i MQTT klijent s:

```
apt-get install mosquitto-clients
```

Ispravnost rada se može provjeriti s naredbama `mosquitto_sub` i `mosquitto_pub`. One predstavljaju primatelja, odnosno pošiljalca poruka. Ovakav alat za naredbeni redak je pogodan kod izgradnje i testiranja sustava jer se jednostavnom naredbom može provjeriti stižu li poruke na određenu temu te se također mogu poslati same poruke.

#### 3.4.3. Paho-mqtt

Postoje programski alati i paketi koji stvaraju MQTT klijente koji se spaja na posrednika. U ovom radu korišten je paket `paho-mqtt` napisan za programski jezik Python. Instalacija se pokreće jednostavnom naredbom: `pip install paho-mqtt`, dok se paket ugrađuju jednostavnom `import paho.mqtt` naredbom.

Mosquitto programski alati se podiže pri pokretanju Raspberry Pi uređaja, a paho-mqtt alat omogućuje stvaranje klijenta koji osluškuju ili zapisuju na određene teme.

Prvo je potrebno definirati klijentski objekt te ga spojiti na IP adresu i port posrednika na koji se spaja. Potom se definiraju funkcije koje se izvršavaju pri konekciji i primanja poruke klijenta. Napisane funkcije moguće je postaviti da se pokreću u beskonačnoj petlji.

## 3.4.4. Ostvarivanje komunikacije

U našem sustavu, MQTT protokol se koristi pri komunikaciji ESP8266 i Raspberry Pi 3. Jedan ESP8266 se nalazi kod ultrazvučnog senzora, on predstavlja pošiljatelja. Drugi ESP8266 koji se nalazi kod zvučnika predstavlja primatelja, a Raspberry Pi 3 se koristi kod središnji posrednik poruka. Sva tri uređaja se spajaju na zajedničku bežičnu mrežu. Budući da se u programskom kodu ESP8266 modula definira IP adresa posrednika na kojeg se spaja, potrebno je Raspberry Pi 3 uređaju postaviti statičku adresu. Sustav je zamišljen da se pri detekciji pokreta na ultrazvučnom senzoru generira i pošalje poruka na outTopic temu MQTT servera te Raspberry Pi sluša i prosljeđuje informaciju na inTopic temu koju sluša drugi ESP8266. On pri primitku odgovarajuće poruke generira audio signal koji bi trebao probuditi skrbnika.

Sljedeći programski isječak opisuje stvaranje klijentskog objekta na Raspberry Pi uređaju koji pri konekciji se preplaćuje na outTopic temu te pri primitku poruke, šalje '1' na inTopic temu. IP adresa brokera je localhost(127.0.0.1) što označava da se broker nalazi na samom Raspberry Pi uređaju i to na vratima 1883.

```
client = mqtt.Client()
client.connect("localhost", 1883, 60)

client.on_connect = on_connect
client.on_message = on_message

client.loop_forever()
```

Funkcija on\_connect je ostvaruje slušanje informacije sa outTopic teme.

```
def on_connect(client, userdata, flags, rc):
    print ("Client connected...")
    client.subscribe('outTopic')
```

Pri korištenju sustava, primijećeno je da ultrazvučni senzor zbog smetnje ponekad detektira pokret i generira poruku, iako nitko nije prošao ispred njega. Poruka se generira samo jednom, dok se primjerice pri kretanju čovjek ispred senzora pošalje više poruka u manjem vremenskom intervalu. Zbog toga, funkcija koja šalje poruku za alarm prije slanja provjerava jesu li poslane dvije poruke u malom vremenskom intervalu (dvije sekunde).

```
import time

time1 = 0.0
sent = 0

def on_message(client, userdata, msg):
    info = msg.payload.decode()
    print ("Package received..")
    print (info)
    global sent
    global time1
    if info == "1":
        sent +=1
        if sent == 2:
            if float(time.time())-time1 < 2:
                client.publish('inTopic',info)
                client.disconnect()
                print ('Data sent to alarm')
            sent = 0

time1 = float(time.time())
```

### 3.4.5. Programiranje ESP8266

Sljedeći programski odjeljak prikazuje slanje poruke prvog modula ESP8266 koji se nalazi uz sam senzor na `outTopic` te ispis same aktivnosti na `logTopic` kao i inicijalizaciju samog sustava te spajanje na WiFi mrežu i postavljanje `mqtt_server`a na vrata 1883 Raspberry PI-a.

```
#include <ESP8266WiFi.h>
#include "PubSubClient.h"

...

WiFiClient espClient;
PubSubClient client(espClient);

...
client.publish("logTopic", " ESP8266-sensor1 Connected");
client.publish("outTopic", msg);

void setup() {
    Serial.begin(115200);
    setup_wifi();
    client.setServer(mqtt_server, 1883);
}

...
```

Na sličan, gotovo identičan način se modificira programska podrška za drugi ESP8266 koji se nalazi uz alarm. On se za razliku od prvoga spaja na `inTopic`, te sadrži

funkciju `callback` putem koje na seriju stavlja string „Pali“ ako je primio „1“ od servera.

```
#include <ESP8266WiFi.h>
#include "PubSubClient.h"
...
WiFiClient espClient;
PubSubClient client(espClient);

client.publish("logTopic", "ESP8266-alarm Connected");
client.subscribe("inTopic");
...
```

`PubSubClient.h` je biblioteka, kompatibilna s ESP8266, koja pruža niz ugrađenih funkcija namijenjenih za stvaranje klijenta te prijavljivanje/objavljivanje na temu na poslužitelj koji podržava MQTT protokol.

## 3.4.6. Zapisnik aktivnosti

Zbog jednostavnosti i lakšeg praćenja aktivnosti senzora, omogućen je zapisnik aktivnosti koji se pokreće pri uključivanju Raspberry Pi 3. Sustav je dizajniran tako da je u datoteci `/etc/rc.local` dodana naredba `python logTopic.py` koja pokreće skriptu koja se preplaćuje na `logTopic` temu. Oba ESP8266 modula prijavljuju uspješno spajanje na MQTT tako da na `logTopic` pošalju svoj identifikator. Python skripta `logTopic.py` kreira klijenta koju sluša poruke na `logTopic` temi te dobivene informacije zapisuje u datoteku `log.txt`. Svaka informacije se zapisuje u novi redak u formatu

```
Log recieved @ "+date+" with info: "+ info + "\n"
```

gdje je `date` trenutno vrijeme, a `info` podatak pročitao sa `logTopic` teme. Po potrebi moguće je proširiti raspon podataka koji se zapisuju u zapisnik te prilagoditi njihov format.

## 3.5. Dojava o pokretu na jedinicu za alarmiranje

Jedinicu za alarmiranje predstavlja zvučnik, koji je spojen putem 3.5 mm audio priključka na Arduino Mega 2560. Arduino Mega 2560 je preko serijskog sučelja spojen na ESP8266. U slučaju da EPS8266 na svoju seriju stavi kodnu riječ „Pali“, Arduino će tada uz ugrađenu funkciju `tone(pin,frequency,duration)` na izvod 3 generirati pravokutni signal točno određene frekvencije koja odgovara određenoj noti definiranoj u datoteci `pitches.h`. Na izvod 3 spojen je drugi izvod 3.5 mm audio priključka (Slika 9). Osim datoteke s definiranim notama, potrebno je imati još dvije varijable u kojima se nalazi popis nota i trajanja pojedinih nota.

```
#include "pitches.h"
```

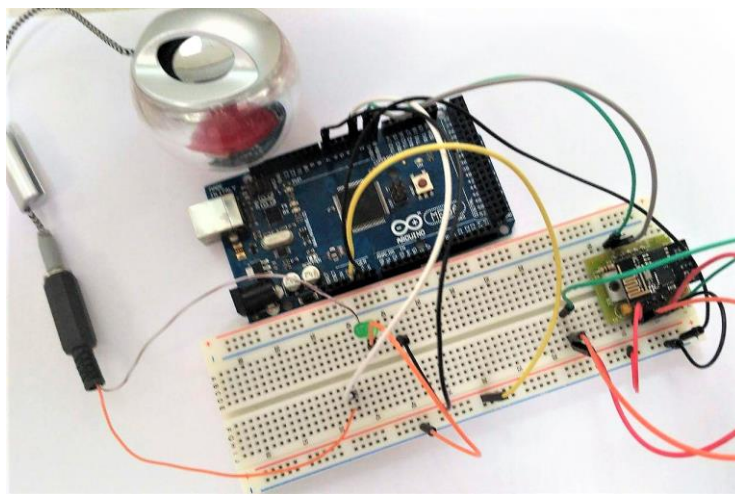


```
int melody[] ={
  NOTE_D2, NOTE_G4, NOTE_FS4, NOTE_A4,
  NOTE_G2, NOTE_C5, NOTE_AS4, NOTE_A4,

  NOTE_FS4, NOTE_G4, NOTE_A5, NOTE_FS4, NOTE_DS4, NOTE_D4,
  NOTE_C4, NOTE_D4, END };

int duration[]={
  8,4,8,4,
  4,4,4,12,
  4,4,4,4,4,4,
  4,16 };

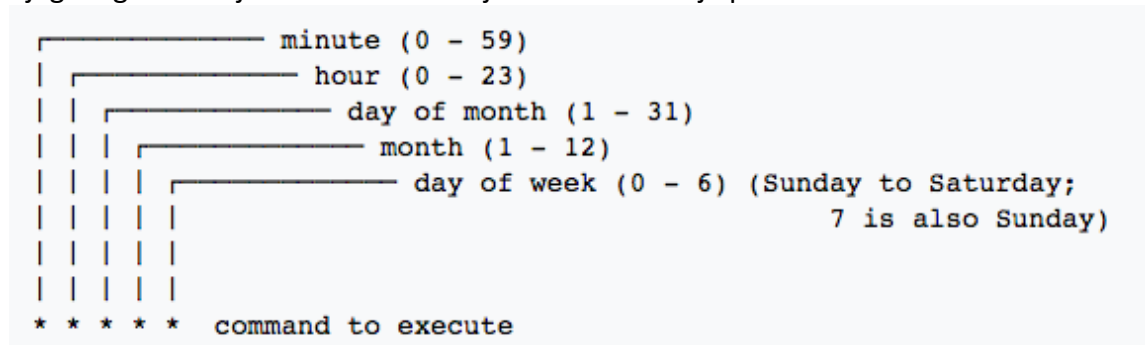
```



Slika 9. Zvučnik, 3.5 mm audio priključak, Arduino Mega 2560 i ESP8266

### 3.6. Cron

Cron je vremenski orijentiran raspoređivač poslova za operacijske sustave koji se ponašaju na sličan način kao Unix sustavi. Cron se koristi kako bi se u sustavima omogućilo pokretanje određenih poslova periodički, u određenom vremenskom trenutku ili intervalu. Korisnici ovim alatom mogu upravljati koristeći crontab –e naredbu. Svaka linija crontab datoteke predstavlja zadatak i vremenski trenutak njegovog izvođenja. Slika 10. Prikazuje način zadavanja poslova u crontab datoteci.



Slika 10 Crontab datoteka

Sustav je moguće prilagoditi navikama korisnike te se u crontabu može zadati vremenski trenutak kada se pokreće skripta koja pokreće sustav nadzora. Ako želimo da se sustav svaki dan pokreće u 20 navečer u crontab dodajemo redak:

```
20 * * * ./home/pi/start_processing.sh.
```

Skripta pokreće proces detekcije pokreta koji prestaje biti aktivan kada se osoba probudi, što se također može prilagoditi navikama pacijenta.

## 4. Zaključak

Razvijeni sustav za detekciju mjesečarenja osigurava alarmiranje ukućana o izlasku promatrane osobe iz sobe ako je riječ o mjesečaru, ili o samom ustajanju s kreveta ako se sustav koristi za sigurnost dementne osobe tijekom noći. Sustav time osigurava miran san skrbnicima. Ovim sustavom nije riješen problem otvaranja prozora kao i potencijalnih opasnosti koje to donosi. Također, pretpostavljeno je ako se sustav koristi za nadzor mjesečarenja, da je soba osigurana i da promatrana osoba ne može učiniti štetu sebi, drugima ili samom prostoru te objektima koji su joj dostupni u sobi. Sljedeći korak u razvoju ovog sustava mogao bi biti nadogradnja sustava određenom „budilicom“, čijim bi se navijanjem sustav palio, i gasio po isteku navedenog roka, kao i osiguranje od otvaranja prozora.

## 5. Literatura

- [1] Arduino Uno Overview,  
URL: <https://www.arduino.cc/en/main/arduinoBoardUno>
- [2] Arduino MEGA 2560 Overview,  
URL: <https://www.arduino.cc/en/main/arduinoBoardMega2560>
- [3] Ultrasonic Ranging Module HC – SR04,  
URL : <http://www.micropik.com/PDF/HCSR04.pdf>
- [4] ESP8266EX Datasheet, 2015  
URL: <http://download.arduino.org/products/UNOWIFI/0A-ESP8266-Datasheet-EN-v4.3.pdf>
- [5] Izrada jednostavnih digitalnih uređaja, WiFi, Davor Cihlar, svibanj 2016, Zagreb  
URL: <https://drive.google.com/file/d/0BwYaVckk1INYSy1GaXIEY1o4QzA/view>
- [6] MQTT Official Site  
URL: <http://mqtt.org/>
- [7] Mosquitto Official Site  
URL: <https://mosquitto.org/>
- [8] Cron, Wikipedia  
URL: <https://en.wikipedia.org/wiki/Cron>
- [9] Biblioteka PubSubClient, Knolleray, 2017  
URL: <https://github.com/knolleary/pubsubclient>
- [10] Raspberry Pi 3 model B Installing operating system images,  
URL: <https://www.raspberrypi.org/documentation/installation/installing-images/README.md>
- [11] Processing Overview,  
URL: <https://processing.org/overview/>
- [12] Processing Examples,  
URL: <https://processing.org/examples/>
- [13] Processing Tutorials,  
URL: <https://processing.org/tutorials/>
- [14] Running Without a Display, Jeff Thompson, 2017,  
URL: <https://github.com/processing/processing/wiki/Running-without-a-Display>
- [15] Using a Standard USB Webcam,  
URL: <https://www.raspberrypi.org/documentation/usage/webcams/>
- [16] Gstreamer Documentation,  
URL: <https://gstreamer.freedesktop.org/documentation/>

## 6. Pojmovnik

Pojam	Kratko objašnjenje	Više informacija potražite na
Arduino Uno	Razvojna platforma	<a href="#">Arduino Uno</a>
Arduino MEGA 2560	Razvojna platforma	<a href="#">Arduino MEGA 2560</a>
HC-SR04	Ultrazvučni senzor	<a href="#">Ultrasonic sensor</a>
ESP8266	WiFi modul	<a href="#">Datasheet ESP8266</a>
Raspberry Pi 3 model B	Ugradbeno računalo	<a href="#">Raspberry Pi 3 model B</a>
Raspbian	Operacijski sustav	<a href="#">Raspbian</a>
Processing	Programski jezik	<a href="#">Processing</a>
Gstreamer	Biblioteka za multimediju	<a href="#">GStreamer</a>
Xvfb	Virtualni spremnik grafičkog sučelja	<a href="#">Xvfb</a>
IP	Komunikacijski protokol	<a href="#">Internet Protocol</a>
SSH	kriptografski mrežni protokol za sigurno upravljanje mrežnim servisima preko nesigurne veze	<a href="#">Secure Shell</a>
PubSubClient	Javno dostupna biblioteka koja omogućava jednostavno korištenje MQTT protokola	<a href="#">PubSubClient</a>

## 7. Raspodjela zadataka

Članovi tima	Individualni zadaci
Odri Tomas	Komunikacija unutar sustava
Maja Ovčarik	Implementacija senzora, alarma i wifi modula
Andrea Matić	Detekcija pokreta na slici
Nikola Vrebčević	Inicijalizacija kamere, dojavljivanje o pomaku
Katarina Karužić	Detekcija oblika objekata