

PROJEKTOVANJE ELEKTRONSKIH SISTEMA

PROJEKTNA DOKUMENTACIJA

NAZIV PROJEKTA:

Sistem za detekciju i fotografisanje za digitalnu oglasnu tablu

TEKST ZADATKA:

Napraviti sistem za fotografisanje koji detektuje prisustvo čoveka na odgovarajućem rastojanju od digitalne oglasne table koristeći Raspberry Pi 3, Raspberry Pi Camera Board i WolkAbout platformu.

MENTOR PROJEKTA:

Vladimir Rajs

PROJEKAT IZRADIO:

Predrag Kovačević

DATUM ODBRANE PROJEKTA:

10.30.2018

1. Opis projekta i korišćenih komponenti

Sistem za detekciju i fotografisanje za digitalnu oglasnu tablu treba da ispunjava sledeće funkcije:

1. Proveru rastojanja korisnika od oglasne table
2. Fotografisanje na zahtev korisnika (pomoću tastera)
3. Kontrolu rada kamere putem WolkAbout platforme

Funkcije iznad su realizovane pomoću Raspberry Pi 3 model B (slika 1), Raspberry Pi Camera Board (slika 2), WolkAbout platform i ultrazvučnog senzora HC SR04 (slika 3).



Slika 1



Slika 2

“Raspberry Pi 3 model B je najnoviji model single-board kompjutera veličine kreditne kartice razvijen u Ujedinjenom Kraljevstvu(Engleska) od strane Fondacije Raspberry Pi sa namerom da promoviše i nauči osnove kompjuterske nauke u školama i razvojnim zemljama. Kao operativni sistem na raspolaganju su: Raspbian,Ubuntu MATE,Snappy Ubuntu Core,Windows 10 IoT Core,RISC OS,Debian,Arch Linux ARM.

Koristi 1200 MHz quad-core ARM Cortex-A53 procesor.Poseduje memorija od 1GB. Za skladištenje koristi MicroSDHC slot kao i njegovi prethodnici.

Grafiku predstavlja Broadcom VideoCore IV ali na većim clock frekvencijama nego njegovi prethodnici koji su radili na 250 MHz.

Snaga za napajanje ovog modela iznosi 4.0W” (https://sr.wikipedia.org/wiki/Raspberry_Pi)

U daljem tekstu umesto Raspberry Pi biće korišćena skraćenica RPi.

U ovom projektu student se odlučio za operativni sistem Raspbian. Pored operativnog sistema na RPi su instalirane sve potrebne biblioteke, drajveri i ostali propratni softveri da omoguće rad sa RPi kamere i WolkAbout platformom.

Osnovni podaci o RPi kamera modulu:

- 5MP sensor
- Wider image, capable of 2592x1944 stills, 1080p30 video
- 1080p video supported
- CSI

- Size: 25 x 20 x 9 mm



Slika 3

Osnovni podaci o ultrazvučnom senzoru HC SR04:

- Operating Voltage: 5V DC
- Operating Current: 15mA
- Measure Angle: 15°
- Ranging Distance: 2cm - 4m

Datasheet: https://cdn.sparkfun.com/assets/b/3/0/b/a/DGCH-RED_datasheet.pdf

2. Program

Za realizaciju softverskog dela funkcionalnosti sistema korišćen je programski jezik Python pošto je Linux njegovo prirodno okruženje. Program omogućava: konektovanje na WolkAbout platformu, prijem podataka sa WolkAbout platforme (da li sistem ima dozvolu za rad), inicijalizaciju kamere i komandu za fotografisanje preko tastera kao i referenciranje odgovarajućih GPIO pinova za komunikaciju i rad sa ultrazvučnim senzorom (merenje udaljenosti).

Kod koji sledi je objašnjen pomoću komentara koji se nalaze pre svake funkcionalne celine.

```
# učitavanje odgovarajućih biblioteka
```

```
import RPi.GPIO as GPIO
```

```
import time
```

```
import picamera
```

```

import math
import os
import random
import sys
import wolk

# Ovom komandom se za anotaciju pinova uzima "Broadcom SOC channel" druga opcija je
#GPIO.Board gde bi anotacija bila fizički redni broj

GPIO.setmode(GPIO.BCM)

TRIG = 4
ECHO = 18
BUTTON = 25

# definišu se ulazni i izlazni ponovi
GPIO.setup(TRIG, GPIO.OUT)
GPIO.setup(ECHO, GPIO.IN)
GPIO.setup(BUTTON, GPIO.IN)

#kreiranje objekta kamera
camera = picamera.PiCamera()
#okretanje slike za 180 stepeni
camera.vflip = True

time.sleep(1)

# Deo za Wolkabout platformu
#####
#definisnje device-a koji je kreiran na WolkAbout platformi
device = wolk.Device(
    key="7dxavm1c0r2xu2jc",
    password="4b8f5d9d-9bb4-45d9-8f39-e4e23591f2f3",
    actuator_references=["SW", "SL"],
)
class ActuatorSimulator:
    def __init__(self, initial_value):
        self.value = initial_value

```

```

switch = ActuatorSimulator(False)
slider = ActuatorSimulator(0)

# Omogućava način za čitanje podataka sa aktuatora ako device ima aktuatore
class ActuatorStatusProviderImpl(wolk.ActuatorStatusProvider):
    def get_actuator_status(self, reference):
        if reference == "SW":
            return wolk.ACTUATOR_STATE_READY, switch.value
        elif reference == "SL":
            return wolk.ACTUATOR_STATE_READY, slider.value

# Omogućava aktuator handler ako device ima aktuatora
class ActuationHandlerImpl(wolk.ActuationHandler):
    def handle_actuation(self, reference, value):
        print("Setting actuator " + reference + " to value: " + str(value))
        if reference == "SW":
            switch.value = value

        elif reference == "SL":
            slider.value = value

# Pass your device, actuation handler and actuator status provider
# Pass configuration handler and provider
# Pass custom outbound message queue implementation
# Enable firmware update by passing a firmware handler
try:
    wolk_device = wolk.WolkConnect(
        host="iot-elektronika.ftn.uns.ac.rs",
        port=1883,
        device=device,
        actuation_handler=ActuationHandlerImpl(),
        actuator_status_provider=ActuatorStatusProviderImpl(),
    )
except RuntimeError as e:
    print(str(e))
    sys.exit(1)

```

```

# Upostavlja konekciju sa WolkAbout platformom
print("Connecting to WolkAbout IoT Platform")
try:
    wolk_device.connect()
except RuntimeError as e:
    print(str(e))
    sys.exit(1)

wolk_device.publish_actuator_status("SW")
wolk_device.publish_actuator_status("SL")

publish_period_seconds = 5

wolk_device.publish()

#####
# KRAJ za Wolkabout platformu
print('Connected to WolkAbout Platform')
print('Pritisnite taster kako biste bili uslikani.')

# while petlja ce se prekinuti kada se stisne bilo koji taster na tastaturi
try:
    while True:
# proverava da li dovođen rad sistemu na WolkAbout platformi i da li je pritisnut fizicki taster
        if GPIO.input(BUTTON) == True and switch.value == True:

            time.sleep(2)
            GPIO.output(TRIG, True)
            time.sleep(0.0001)
            GPIO.output(TRIG, False)
# merenje udaljenosti
            while GPIO.input(ECHO) == False:
                start = time.time()

            while GPIO.input(ECHO) == True:
                end = time.time()

```

```

sig_time = end - start

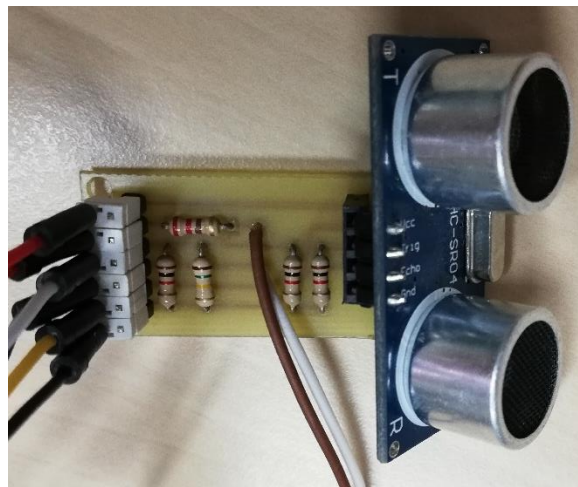
# cm:
distance = sig_time / 0.000058

print('Distance: {} cm'.format(distance))
# ako je korisnik preblizu ponoviti postupak
if distance > 20:
    print('Za 2 sekunde bicete uslikani.')
    time.sleep(2)
    camera.capture('example.jpg')
else:
    print('Udaljite se na vise od 20cm od senzora i Pritisnite taster ponovo.')
except KeyboardInterrupt:
# dereferencirati sve pinove
GPIO.cleanup()

```

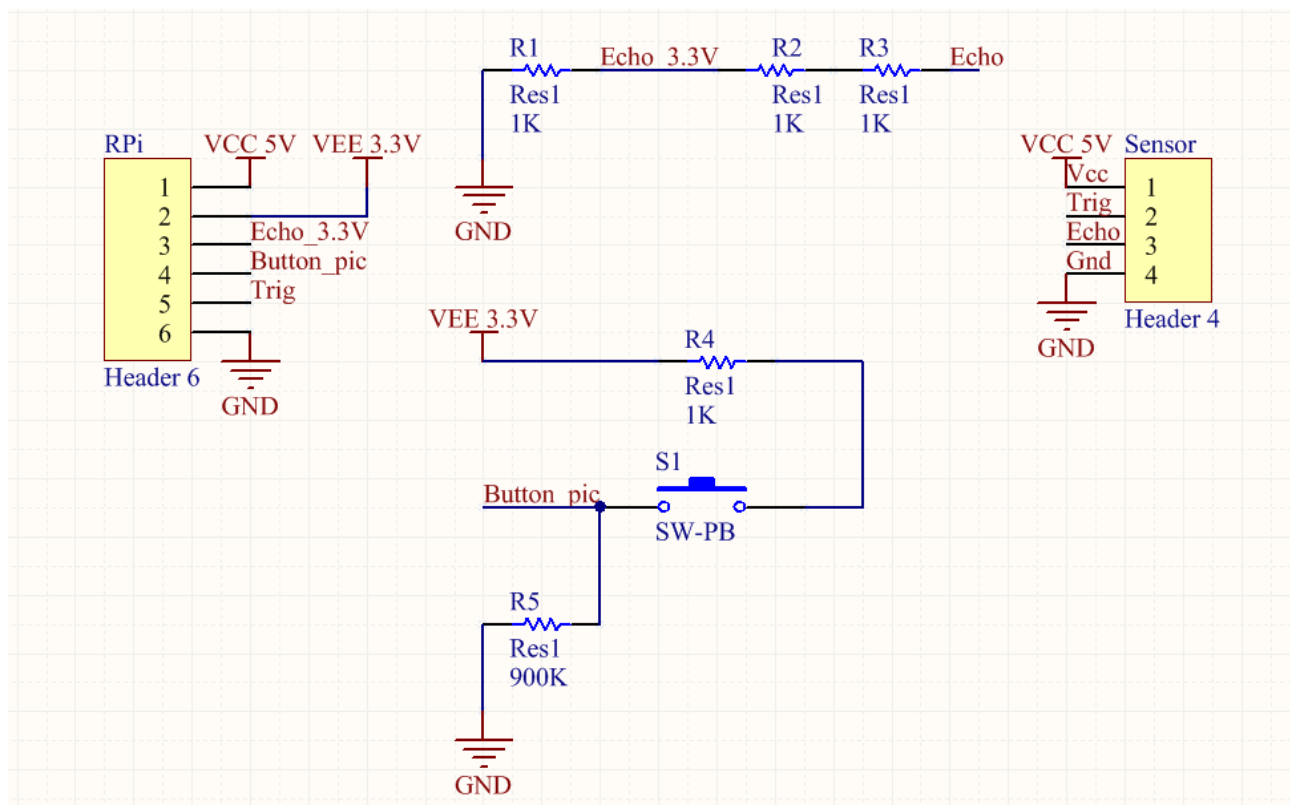
3. Hardver

Da bi se osigurao ispravan rad ultrazvučnog senzora napravljena je prilagodna PCB pločica (slika 4).



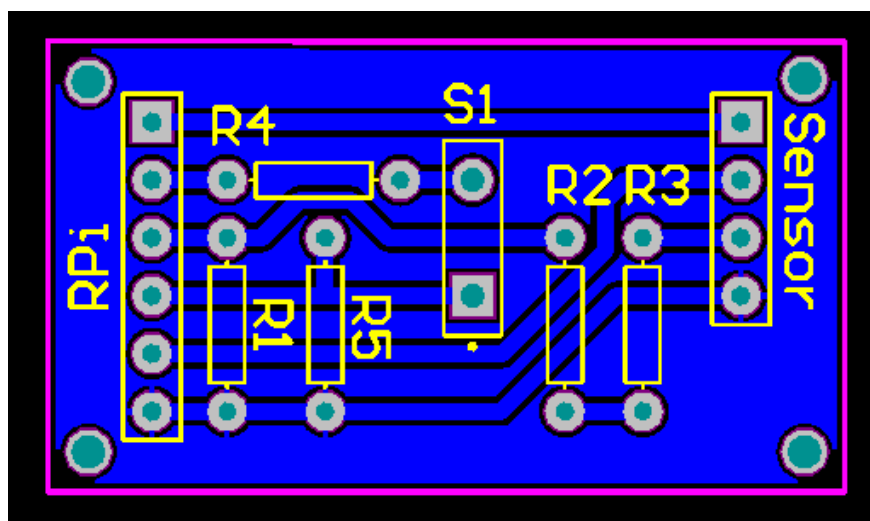
Slika 4

Šema PCB se može videti na slici 5.



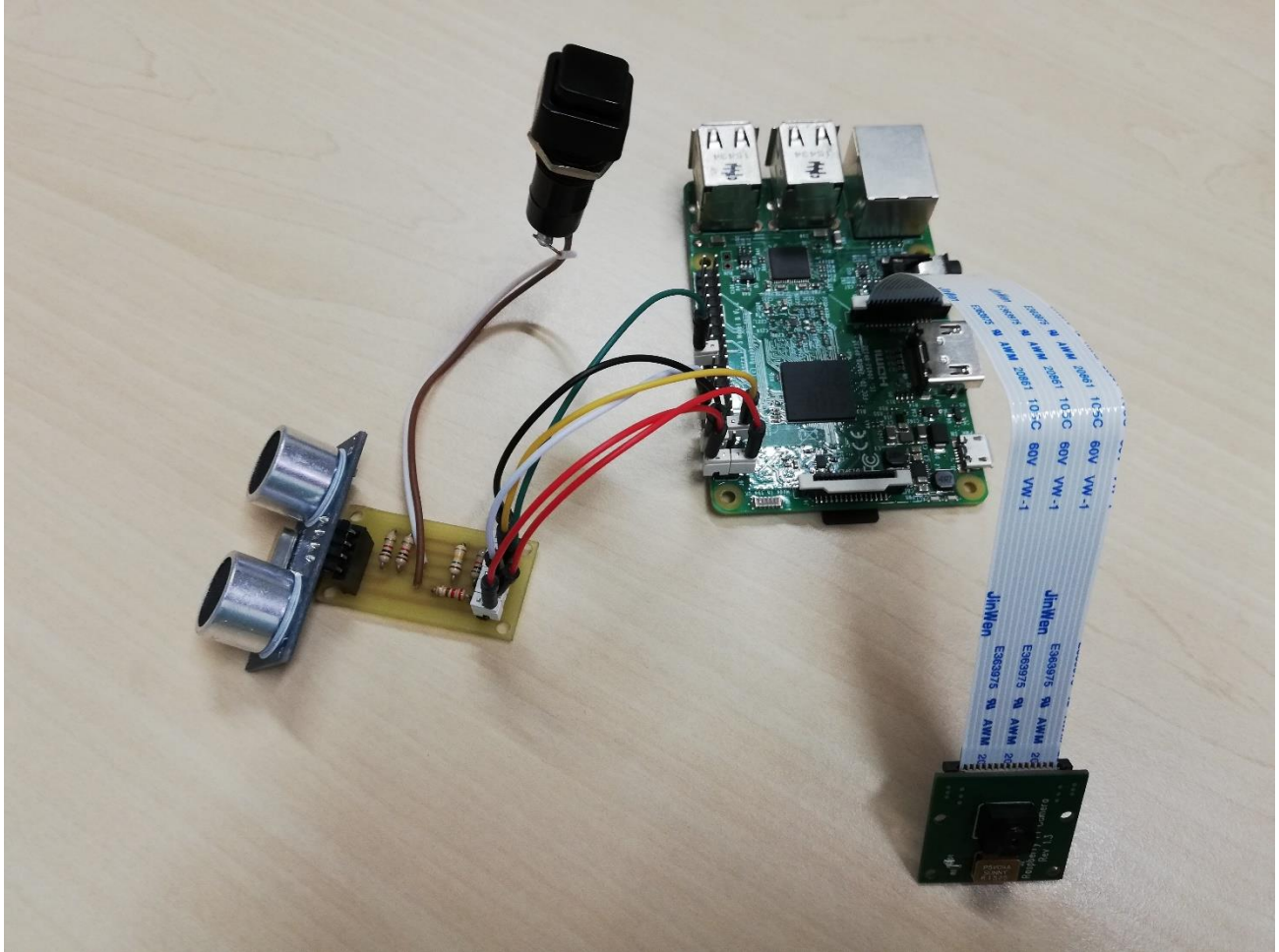
Slika 5

RPi header služi za spajanje prilagodne pločice sa RPi 3, dok je Sensor header mesto za spajanje ultrazvučnog senzora. R1, R2 i R3 se koriste kao naponski razdelnik pošto je maksimalni ulazni napon na RPi 3.3V dok je napon na izlazu ultrazvučnog senzora 5V. R5 je pulldown otpornik, a R4 sigurnosni otpornik u slučaju softverske greške prilikom referenciranja tipa pina. Na slici 6 se može videti PCB layout.



Slika 6

Na slici 7 se mogu videti svi povezani elementi, bez napajanja i ekrana.



Slika 7

4. Machine to machine

Machine to machine (skraćeno M2M) odnosi se na komunikaciju između uređaja koji koriste bilo koji vid komunikacionih kanala, uključujući žičani i bežični prenos. M2M komunikacija može da sadrži industrijsku instrumentaciju, omogućavajući senzora ili meračima da razmenjuju podatke koje mere (kao što su temperatura, stanje na lageru itd.) sa aplikativnim softverom koji ih koristi (primer, podešavanje industrijskih procesa na osnovu temperature ili poručivanja robe da se dopuni lager). Takva komunikacija je prvobitno ostvarena time što je ne remote network mašina prenosila informacije nazad u centralno čvorište (central hub) za analizu, koje bi potom bila preusmerena u sistem kao što je lični računar (PC).



Slika 8

Glavne komponente M2M sistema uključuju senzore, RFID, Wi-Fi ili cellular data network i softver za autonomic computing software koji je programiran da pomogne mrežnom uređaju da tumači podatke i donosi odluke. Ove M2M aplikacije prenose podatke, koje mogu pokrenuti unapred programirane, automatske akcije.

Jedan od najpoznatijih tipova komunikacije mašina-mašina je telemetrija, koja se od početka prošlog veka koristi za prenos operativnih podataka. Pioniri u telemetriji prvi put su koristili telefonske linije, a kasnije i radio talase, kako bi preneli merenja performansi prikupljenih od nadzornih instrumenata na udaljenim lokacijama.

Internet i poboljšani standardi za bežičnu tehnologiju proširili su ulogu telemetrije od čistih nauka, inženjeringa i proizvodnje do svakodnevne upotrebe u proizvodima kao što su grejne jedinice, električni uređaji i uređaji povezani putem interneta, kao što su kućni uređaji.

5. Internet of Things

Internet stvari (engleski: Internet of Things, skraćeno IoT) predstavlja međumrežavanje fizičkih objekata, vozila (što se odnosi i na „povezane uređaje” i „pametne uređaje”), zgrada i drugih stvari sa ugrađenom elektronikom, softverom, sensorima i konektivnošću koji omogućavaju objektima da razmenjuju podatke sa proizvođačem, operaterom i/ili drugim povezanim uređajima. Godine 2013. Global Standards Initiative on Internet of Things (IoT-GSI) definisala je IoT kao „globalnu infrastrukturu informatičkog društva koja omogućava napredne usluge (fizičkim i virtualnim) umrežavanjem stvari, pritom se zasnivajući na postojećim i interoperabilnim informacionim i komunikacionim tehnologijama u razvoju”. U tu svrhu, termin „stvar” predstavlja „predmet fizičkog sveta (fizičkih stvari) informacija ili reč (virtualne stvari), koji je moguće identifikovati i koji može da bude integrisan u komunikacionim mrežama”.

IoT omogućava da objekti budu opaženi i kontrolisani daljinski putem postojeće mrežne infrastrukture, stvarajući tako šanse za direktniju integraciju fizičkog sveta i računarskih sistema, što rezultuje povećanjem efikasnosti, tačnosti i ekonomske koristi, uz smanjenje ljudske intervencije. Kada je IoT povezan sensorima i pogonima, tehnologija postaje instanca opštije klase sajber-fizičkog sistema, što takođe obuhvata tehnologije poput pametnih mreža, virtualnih elektrana, pametnih kuća, inteligentnog transporta i pametnih gradova. Svaku stvar je moguće jedinstveno identifikovati kroz ugrađen kompjuterski sistem i svaka stvar je interoperabilna u okviru postojeće internet infrastrukture. Stručnjaci procenjuju da će IoT do 2020. godine sadržati između 26 i 30 milijardi objekata.

Termin „The Internet of Things” je predložio Kevin Ešton 1999. godine. Pojam Internet stvari prvi put je postao popularan preko centra za Auto-ID Center na MIT u vezi sa tržištem, iako je o konceptu razgovarano još od 1991. godine. Identifikacije radio-frekvencija (RFID) i analiza publikacije bile su viđene kao preduslov za “Internet of Things” u ranim danima. Kada bi svi objekti i ljudi u svakodnevnom životu bili opremljeni identifikatorima, oni bi bili memorisani u kompjuteru. Pored korišćenja RFID-a, označavanje se može postići kroz takve tehnologije kao što su približna polja komunikacije, barkod, QR kod i digitalni vodeni žig.

Prema kompaniji Gartner, biće skoro 26 milijardi uređaja na “Internet of Things” do 2020. Za razliku od njih, ABI istraživanje pretpostavlja da će ta brojka dostići više 30 milijardi uređaja koji će biti bežično povezani na “Internet of Things” do 2020. Cisco je kreirao dinamički brojač konekcija za praćenje i izračunavanje broja konekcija od jula 2013. do jula 2020.[18] Ovaj koncept, gde uređaji povezivanjem na internet/veb preko niskonaponskih radija, predstavlja polje na kome se najviše istražuje “Internet of Things”. Na male potrošnje radio ne treba da koristi Wi-Fi ili Bluetooth. Jeftinije alternative istražuju se od strane kompanije Chirp Networks.[19]

Svi senzori i mašinski čitljivi identifikatori potrebni da "Internet of Things" će morati da koriste internet protokol verzije IPv6 da bi primile izuzetno veliki adresni prostor koji je potreban. U svom originalnom tumačenju, jedna od prvih posledica sprovođenja "Internet of Things" opremanjem svih objekata u svetu sa minijturnim uređajima za identifikaciju ili mašinski čitljivim identifikatorima bi se mogao transformisati svakodnevni život u nekoliko pozitivnih načina. Na primer, takva tehnologija mogla bi omogućiti mnogo moćniju kontrolu kreatorima sadržaja i vlasnika preko svoje kreacije, boljom primenom ograničenja autorskih prava, tako da kad kupac kupuje Blu-ray disc sa filmom koji želi da gleda, može da bira da plati visoku cenu i bude u mogućnosti da gleda film cele godine, plati cenu umereno i ima pravo da gleda film nedelju dana, ili plati nisku cenu kada želi da gleda film.

Danas, međutim, termin "Internet of Things" se koristi da označi napredna povezivanja uređaja, sistema i usluga, koji dosežu van mašina ka mašini komunikacije machine-to-machine i pokriva različite protokole, domena i aplikacija.

6. Literatura

1. https://en.wikipedia.org/wiki/Machine_to_machine
2. <https://internetofthingsagenda.techtarget.com/definition/machine-to-machine-M2M>
3. https://en.wikipedia.org/wiki/Internet_of_things