

**Fakultet Tehničkih Nauka**

**Novi Sad**

**Predmetni projekat:**  
**Linux rukovalac za Raspberry Pi 2**  
**(Ultrazvučni senzor HC–SR04)**

**Predmet: Sistemska programska podrška u realnom vremenu 2**

**Projekat radili: Repac Marina**

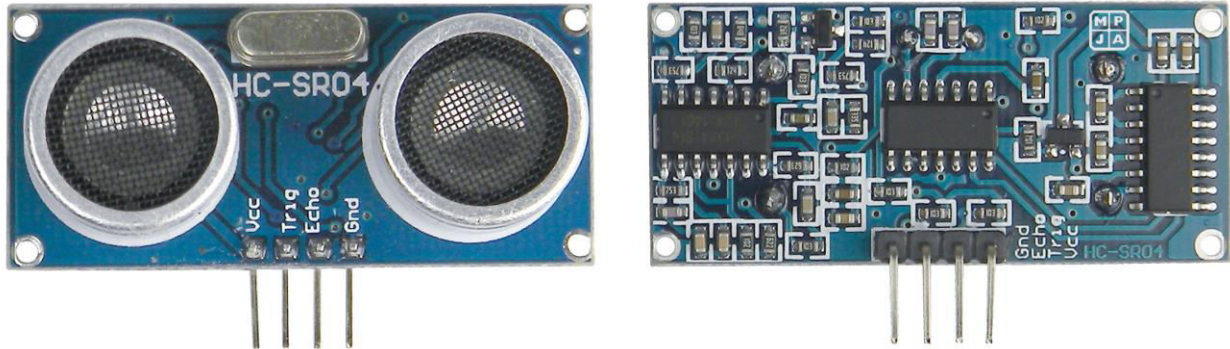
**Zagorac Aleksandar**

## Sadržaj:

Uvod .....	2
Zadatak .....	4
Koncept .....	5
Analiza problema .....	5
Komunikacija Korisnička aplikacija – Rukovalac .....	5
Komunikacija Rukovalac – Korisnička aplikacija .....	5
Opis rešenja .....	6
Funkcije .....	6
Aplikacija za testiranje modula .....	6
Zaključak .....	8

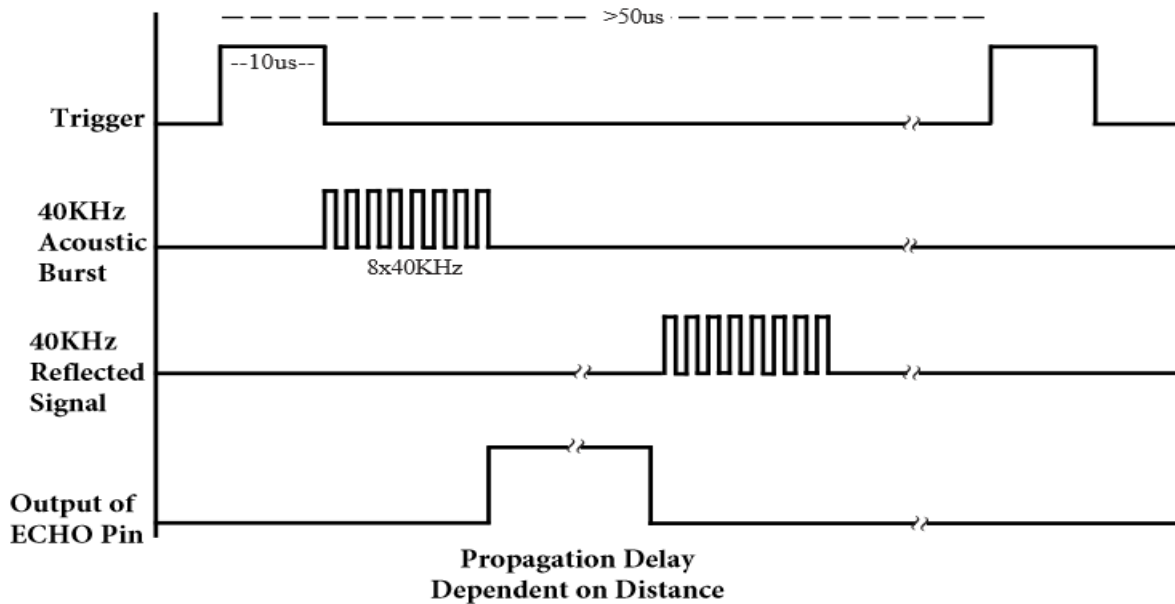
## Uvod

### Ultrasonic Sensor HC-SR04



Emituje ultrazvuk na frekvenciji 40 kHz koji putuje kroz vazduh i ako postoji prepreka na toj putanji zvuk će se odbiti i vratiće se nazad do senzora. Senzor će detektovati odbijeni zvuk. Kada uzmemo u obzir vreme za koje je ovaj ultrazvuk prešao putanju do prepreke i brzinu zvuka, možemo vrlo jednostavno izračunati rastojanje od senzora do prepreke.

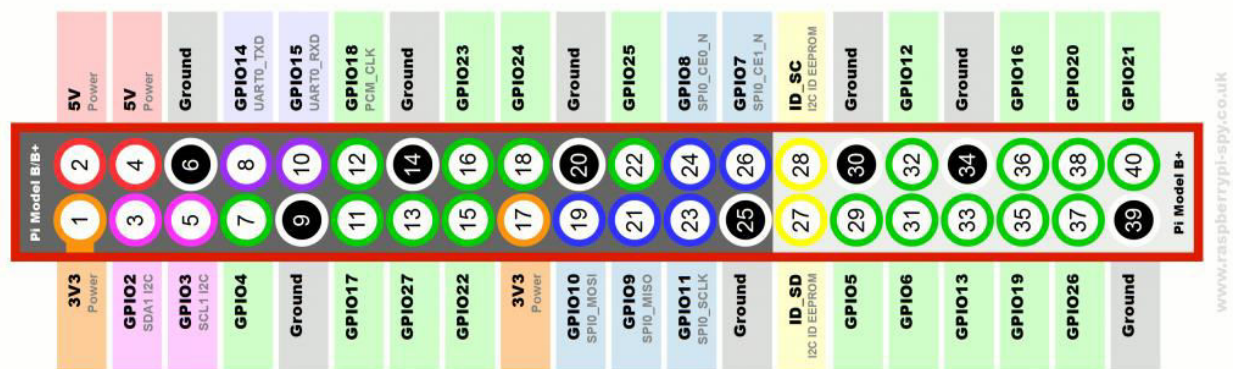
### HC-SR04 ULTRASONIC MODULE



Ultrazvučni modul ima 4 pina:

1. Vcc – napajanje od 5V
2. Trig – Trigger Input pin
3. Echo – prijemni Output pin
4. GND – masa

## Konekcija



Vcc → 5V (2)  
Ground → GROUND (30)  
Trigger → GPIO17 (11)  
Echo → GPIO27 (13)

## Zadatak

Potrebno je realizovati Linux rukovalac za Raspberry Pi 2 uređaj koji treba da omogućiti komunikaciju korisničkog programa i ultrazvučnog senzora HC-SR04 koji je povezan na GPIO prolaze dostupne na uređaju. Rukovalac je potrebno realizovati korišćenjem prekida.

Ultrazvučni senzor će biti korišćen za detekciju udaljenosti prepreka.

Rukovalac treba da omogućiti:

- generisanje Trigger (Ping) signala na zahtev korisničke aplikacije, u skladu sa datim vremenskim dijagramom
- određivanje širine impulsnog Echo signala u prekidnoj rutini (vreme proteklo između rastuće i opadajuće ivice)
- preuzimanje izmerenog vremena koje odgovara širini impulsa Echo signala, na zahtev korisničke aplikacije

U svrhu provere funkcionalnosti razvijenog rukovaoca potrebno je razviti korisničku aplikaciju za detekciju prepreka. Aplikacija treba da obrađuje i ispisuje informacije pristigle sa senzora. Obrada podrazumeva da aplikacija preko uređaja tj. rukovaoca preuzima podatke sa senzora, računa udaljenost prepreke i filtrira dobijene rezultate visokopropusnim IIR filtrom.

Uz zadatak je priloženo:

- datoteka [hc-sr04 ultrasonic module user guide.pdf](#) sa opisom senzora i načina rada istog
- datoteka [Broadcom-BC2835-datasheet.pdf](#) u kojoj se mogu pronaći svi detalji vezani za periferije dostupne na Raspberry Pi 2 uređaju (adrese, registri, načini korišćenja i ostali detalji)
- datoteka (slika) [Raspberry-Pi-GPIO-Layout-Model-B-Plus.png](#) u kojoj se mogu videti dostupni prolazi izvedeni na konektoru Raspberry Pi 2 uređaja

## Koncept

### Analiza problema

Neophodno je obezbediti generisanje Trigger (Ping) signala na zahtev korisničke aplikacije, a potom i prosleđivanje rezultata dobijenih sa senzora korisničkoj aplikaciji. Podatke dobijene sa senzora je potrebno obraditi pre slanja korisniku.

### Komunikacija Korisnička aplikacija – Rukovalac

Putem korisničke aplikacije je omogućeno pokretanje rukovaoca. Na zahtev korisnika počinje generisanje Trigger signala.

### Komunikacija Rukovalac – Korisnička aplikacija

Nakon što je senzor otpočeo sa radom, tj. počeo je generisanje Trigger signala, potrebno je izmeriti vreme koje protekne između rastuće i opadajuće ivice Echo signala (širina signala) u prekidnoj rutini. Unutar prekidne rutine uzimaćemo vrednost Echo signala na rastućoj i na opadajućoj ivici (`ktime_get()`). Takođe ćemo vršiti računanje širine Echo signala oduzimanjem završnog i početnog vremena echo istog, što će takođe biti učinjeno u preidnoj rutini.

Informacije o vremenu koje je potrebno Echo signalu da se vrati do senzora ćemo vratiti korisničkoj aplikaciji unutar funkcije `gpio_driver_read()`.

Deljenjem širine Echo signala, u korisničkoj aplikaciji, sa 58.0 dobijamo rastojanje u centimetrima. Ovde treba voditi računa da je sve što se nalazi na udaljenosti većoj od 400cm van opsega, što će biti ograničeno u samom kodu rukovaoca.

## Opis rešenja

### Funkcija za inicijalizaciju modula ( *gpio\_driver\_init()* )

Ova funkcija se poziva prilikom uvezivanja modula u kernel. U njoj se registruje novi chardev. Vršimo inicijalizovanje odgovarajućih pinova. Zatim se inicijalizuje prekidna rutina namenjena za pin Echo signala.

### Funkcija za prebacivanje podataka korisničkoj aplikaciji ( *gpio\_driver\_read()* )

Funkcija se poziva unutar korisničke aplikacije i u njoj vršimo postavljanje Trigger signala na visoki nivo, koji se zadržava 10us. Nakon toga proveravam da li dolazi do prekoračenja maksimalne udaljenosti objekta. Ukoliko ne dođe do interrupt-a u određenom vremenskom periodu korisniku se šalje poruka da je objekat van dometa uređaja. U suprotnom se šalje širina Echo signala.

### Funkcija prekidne rutine ( *irqreturn\_t h\_irq\_echo()* )

Prekidna rutina je podešena tako da reaguje na rastuću i opadajuću ivicu Echo signala. Na rastućoj ivici Echo signala funkcija se poziva i korišćenjem *ctime\_get()* funkcije, uzimamo trenutno vreme. Na isti način uzimamo vreme i na opadajućoj ivici signala i nakon toga vršimo oduzimanje dobijenih vrednosti. Na ovaj način dobijamo širinu Echo signala i signaliziramo funkciji *gpio\_driver\_read()* da je širina Echo signala uspešno izračunata.

### Funkcija za brisanje modula ( *gpio\_driver\_exit()* )

Ova funkcija se poziva kada se modul uklanja iz kernela. U njoj se oslobađa memorija koja je prethodno zauzeta i oslobađa se chardev.

### Ostale funkcije

Ostale funkcije preuzete su sa primera iz vežbi 6 i 7 i uglavnom služe za postavljanje stanja pinova na Raspberry Pi ploči.

### Aplikacija za testiranje modula

Aplikacija je napravljena sa ciljem testiranja ispravnosti modula. Od korisnika se očekuje da usmeri senzor ka objektu koji se nalazi unutar zadatog opsega. Ukoliko je to odrađeno na odgovarajući način, biće ispisana poruka sa vrednošću udaljenosti objekta. U suprotnom će biti ispisana poruka greške.

Prikaz rada test aplikacije:

```
pi@raspberrypi081331:~/Desktop/SPPURV2_PROJEKAT/gpio_driver_test_app/bin/Release $ ./gpio_driver_test_app
Place the object between 20 cm and 60 cm away from the sensor.
----- Press ENTER to begin -----
----- object is out of range (20cm - 60cm) -----
----- object is out of range (20cm - 60cm) -----
----- object is out of range (20cm - 60cm) -----
----- object is out of range (20cm - 60cm) -----
----- object is out of range (20cm - 60cm) -----
----- object is out of range (20cm - 60cm) -----
----- object is out of range (20cm - 60cm) -----
Distance: 24.81 cm
----- object is out of range (20cm - 60cm) -----
----- object is out of range (20cm - 60cm) -----
----- object is out of range (20cm - 60cm) -----
----- object is out of range (20cm - 60cm) -----
----- object is out of range (20cm - 60cm) -----
Distance: 29.26 cm
Distance: 30.07 cm
Distance: 29.67 cm
Distance: 29.57 cm
----- object is out of range (20cm - 60cm) -----
----- object is out of range (20cm - 60cm) -----
----- object is out of range (20cm - 60cm) -----
Distance: 33.84 cm
Distance: 34.47 cm
Distance: 35.14 cm
Distance: 40.84 cm
Distance: 43.40 cm
----- object is out of range (20cm - 60cm) -----
----- object is out of range (20cm - 60cm) -----
----- object is out of range (20cm - 60cm) -----
Distance: 43.03 cm
^C
pi@raspberrypi081331:~/Desktop/SPPURV2_PROJEKAT/gpio_driver_test_app/bin/Release $
```

Prikaz rada korisničke aplikacije:

```
----- Press ENTER to continue or enter q to quit -----
Distance: 115.57 cm
Distance: 115.48 cm
Distance: 115.17 cm
Distance: 115.05 cm
Distance: 14.95 cm
Distance: 13.88 cm
Distance: 13.67 cm
Distance: 13.62 cm
Distance: 13.16 cm
Distance: 13.57 cm
Distance: 13.57 cm
Distance: 13.50 cm
Distance: 13.14 cm
Distance: 13.60 cm
Distance: 14.07 cm
----- Press ENTER to continue or enter q to quit -----
q
pi@raspberrypi081331:~/Desktop/SPPURV2_PROJEKAT/gpio_driver_user_app/bin/Release $
```

## **Zaključak**

Zadatak projekta je realizacija Linux rukovalac za Raspberry Pi 2 uređaj. Rukovalac treba da omogući komunikaciju korisničkog programa i HC-SR04 senzora. Kako bi demonstrirali funkcionalnost naseg rešenja napravljena je aplikacija koja ispisuje trenutne vrednosti udaljenosti objekta u odnosu na senzor.