PROJEKAT IZ PREDMETA  
OPTOELEKTRONSKI I LASERSKI SISTEMI

NAZIV PROJEKTA:

LIGHTRANGER 4 Click

TEKST ZADATKA:

Potrebno je projektovati sistem koji će omogućiti mjerenje udaljenosti pomoću razvojnog okruzenja *mikromedia Plus for STM32* i klik pločice *Light Ranger 4 Click* kompanije *MikroElektronika*.

MENTOR PROJEKTA:

Bajić dr Jovan

Batinić Branislav

PROJEKAT IZRADILI:

Vladimir Marćiš, E1 83/2021

Milomir Spajić, E1 87/2021

DATUM ODBRANE PROJEKTA:

28. 02. 2022.

SADRŽAJ

[Uvod 3](#_Toc98012272)

[ToF (eng. Time of Flight) senzori 4](#_Toc98012273)

[Podela ToF senzora na osnovu vrste talasa koju koriste 5](#_Toc98012274)

[Prednosti i mane korišćenja ToF senzora 7](#_Toc98012275)

[Glavne komponente korišćene u realizaciji projekta i njihove karakteristike 8](#_Toc98012276)

[Mikromedia Plus for STM32 8](#_Toc98012277)

[Light Ranger 4 Click 10](#_Toc98012278)

[Blok šema sistema 13](#_Toc98012279)

[Algoritam rada sistema 14](#_Toc98012280)

[Biblioteka (drajver) za senzor VL53L1X (klik pločicu) 15](#_Toc98012281)

[Biblioteka (drajver) za TFT displej 17](#_Toc98012282)

[Glavni dio koda (eng. *main*) 20](#_Toc98012283)

[Aplikacija – korisničko uputstvo 23](#_Toc98012284)

[Zaključak 25](#_Toc98012285)

[Literatura 26](#_Toc98012286)

# Uvod

U okviru projekta za predmet *Laserski i optoelektronski sistemi*, realizovali smo sistem koji omogućava mjerenje udaljenosti objekata. Napravili smo aplikaciju i uvezli biblioteku za senzor udaljenosti, te nakon toga izvršili testiranje i napisali dokumentaciju za projekat.

Aplikacija je realizovana na razvojnom okruženju *mikromedia Plus for STM32* i u njoj je moguće birati mod mjerenja (kraće, srednje i duže distance). Na osnovu izabranog moda i parametara mjerenja (vrijeme potrebno za izvršenje mjerenja i vrijeme izmedju dva uzastopna mjerenja), pritiskom na opciju START na TFT displeju osjetljivom na dodir započinje mjerenje.

# ToF (eng. Time of Flight) senzori

Postoji mnoštvo senzora koji služe za beskontaktno merenje udaljenosti objekata, ali gotovi svi ovakvi senzori se mogu podeliti u tri velike grupe:

1. Senzori koji za merenje udaljenosti koriste talase čija je brzina prostiranja poznata (brzina svetlosti, brzina zvuka)

2. Senzori koji za merenje udaljenosti koriste talase koji se pravolinijski prostiru kroz homogenu sredinu

3. Senzori koji za merenje udaljenosti koriste činjenicu da intenzitet bilo koje energije u prirodi kontinualno i monotono opada kako se ta energija udaljava od svog izvora

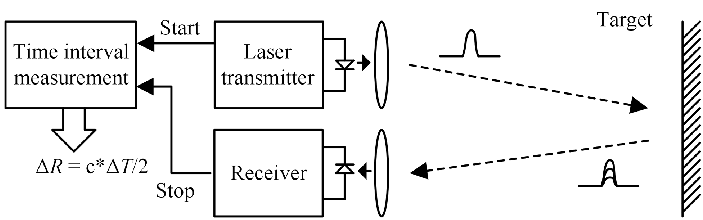
Na osnovu ove tri podele, može se zaključiti da postoje tri različite metode za merenje udaljenosti objekata:

1. ToF metoda

2. Metoda trijangulacije

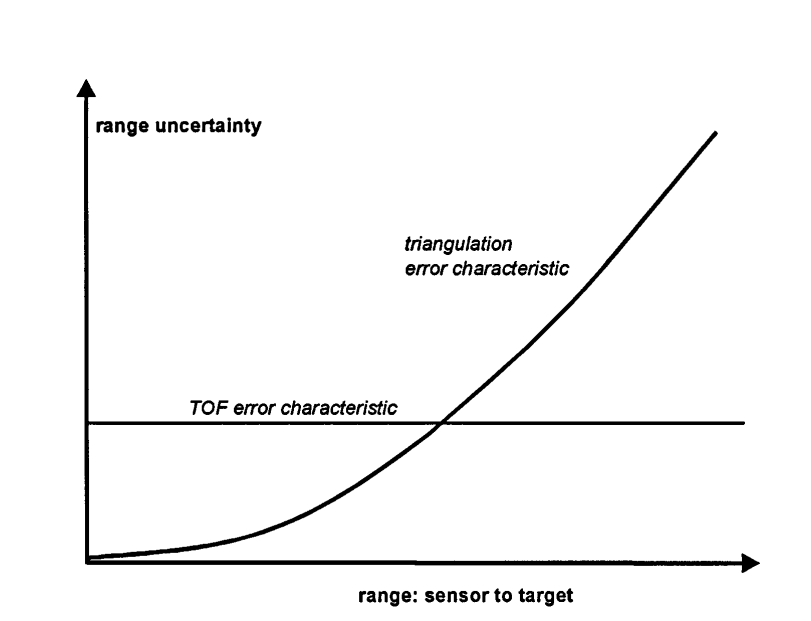
3. Metoda bazirana na principu polja

Dakle, ToF senzor radi na vrlo jednostavnom principu a to je da senzor emituje određeni talas čija je brzina prostiranja poznata. Taj talas dolazi do prepreke odnosno objekta čija se udaljenost meri. Talas se zatim odbija od prepreke i vraća se nazad do senzora. Senzor na osnovu vremena proteklog od trenutka kada je emitovao talas i do trenutka kada je registrovao reflektovani talas preračunava udaljenost datog objekta. Na sledećoj slici će biti prikazan princip rada jednog ToF senzora koji koristi svetlosni talas za proračunavanje udaljenosti.



Slika 1. Princip rada ToF senzora

Niti jedna od tri gorenavedene metode ne može na idealan način da izmeri udaljenost objekta. Na sledećoj slici će biti prikazana karakteristika koja zapravo predstavlja grešku merenja u zavisnosti od udaljenosti objekta za ToF senzor i senzor koji za merenje udaljenosti koristi metodu trijangulacije.



Slika 2. Greška merenja kod ToF metode i metode trijangulacije

Na slici se može videti da je greška u ToF tehnici nezavisna od udaljenosti nekog objekta. Ta greška najviše potiče od činjenice da postoji neki minimalni interval koji senzor koji radi na ToF principu može da detektuje. Ovaj problem je direktno vezan za problem oko detekcije i nezavistan je od udaljenosti objekta. Sa druge strane, senzor koji radi na metodi trijangulacije će praviti grešku koja će rasti kako raste i udaljenost objekta od datog senzora.

## Podela ToF senzora na osnovu vrste talasa koju koriste

Na osnovu tipa talasa koji koriste, ToF senzori se mogu podeliti u nekoliko grupa:

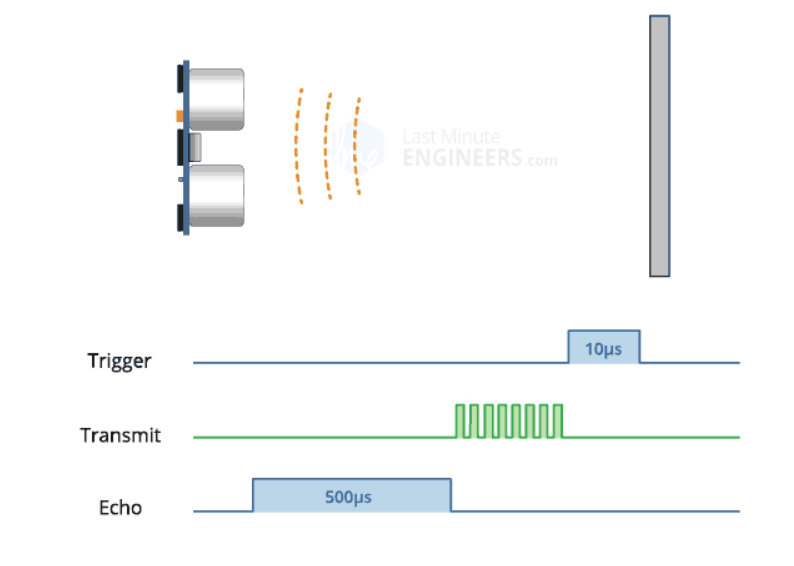
1. ToF senzori koji koriste zvučni talas

2. ToF senzori koji koriste radio talase

3. ToF senzori koji koriste svetlosni talas

1. Senzori koji koriste zvučni talas za preračunavanje udaljenosti nekog objekta su najčešće eho - impulsnog tipa (eng. Pulsed - Echo type). Ovi senzori za svoj rad koriste zvuk koji izlazi iz opsega zvuka koji čuje čovek odnosno takav zvuk spade u ultrazvuk.

Na sledećoj slici, će biti prikazan princip rada jednog ultrazvučnog senzora za merenje udaljenosti. Konkretno, u pitanju je senzor HC - SR04.



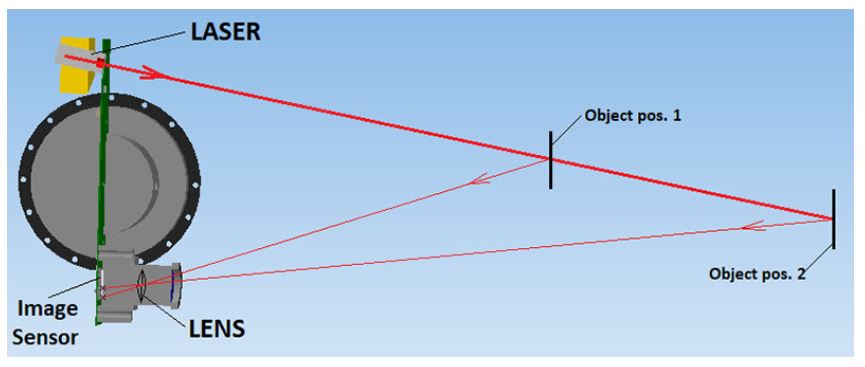
Slika 3. Princip rada ultrazvučnog senzora

Dakle, na odgovarajući pin ovog senzora (označen kao Trigger) se dovodi impuls trajanja 10µs. To je znak senzoru da treba da generiše 8 kratkih impulsa zvuka frekvencije 40kHz. Ovih 8 impulsa zapravo predstavlja određeni jedinstveni signal koji će senzor prepoznati kada se zvuk odbije od prepreke i na taj način će razlikovati odbijeni zvuk od drugih ultrazvučnih ambijentalnih zvukova. Kada senzor izgeneriše tih osam impulsa, pokreće određeni tajmer. Tajmer zaustavlja kada detektuje reflektovani signal. Na osnovu proteklog vremena i brzine zvuka (340 m/s) se može proračunati udaljenost predmeta na osnovu formule:

Dakle, pređeni put odnosno udaljenost nekog objekta se izračunava tako što se pomnoži brzina zvuka sa polovinom vremena koje je tajmer izmerio. To vreme se deli na pola jer je zvuku potrebno da dođe do prepreke i da se vrati nazad do senzora.Odnosno, zvuk dva puta pređe distancu koja se meri.

2. Senzori koji koriste radio talase za preračunavanje udaljenosti zapravo predstavljaju senzore koji su bazirani na korišćenju elektromagnetnih talasa u opsegu od 1m do 1mm talasne dužine. Ovakvi senzori su poznatiji pod imenom RADAR (eng. Radio Detection And Ranging). Radio talasi se mogu koristiti za preračunavanje velikih udaljenosti i maksimalna udaljenost najviše zavisi od trenutnih atmosferskih uslova. Zanimljiva primena radara kratkog dometa je radar koji prodire u zemlju i koristi se za detekciju i lociranje podzemnih objekata.

3. Senzori koji koriste svetlosni talas za preračunavanje udaljenosti su senzori koji koriste infracrveni, vidljivi i ultraljubičasti spektar svetlosti za preračunavanje udaljenosti. Ove frekvencije svetlosti mogu biti proizvedene od strane lasera i jako lako se mogu detektovati nekim fotodetektorom. ToF senzori koji koriste svetlosni talas spadaju u tzv. eho tip senzora (eng. Echo type). Posebna vrsta ToF senzora koji koriste svetlosne talase za merenje udaljenosti objekta naziva se LIDAR senzorima (eng. Light Detection And Ranging). Odlika ovakvih senzora jeste ta što ispaljuju jako veliki broj kratkih laserskih impulsa u sekundi i na osnovu reflektovanja tih impulsa senzor gradi mapu prostora. Princip rada LIDAR senzora prikazan je na sledećoj slici:



Slika 4. Princip rada LIDAR senzora

Prednost ovakvog tipa senzora u odnosu na senzore koje koriste radio talase za preračunavanje udaljenosti jeste ta što svetlosni zrak može imati veoma usku širinu talasnog snopa, što omogućuje veliku rezoluciju merenja. Sa druge strane, mana ovakvih senzora jeste ta što svetlosni snop slabi daleko više od radio talasa kada talas prolazi kroz oblake i maglu.

## Prednosti i mane korišćenja ToF senzora

Prednosti korišćenja ToF senzora su:

1. Beskontaktno merenje

2. Veoma brzo očitavanje i proračunavanje udaljenosti

3. Ukoliko je reč o senzorima koji koriste svetlosni talas onda je prednost ta što se laserski snop ne savija što povećava preciznost na većim udaljenostima

Mane korišćenja ToF senzora su:

1. Prilikom merenja udaljenosti uz pomoć ovakvih senzora, na rezultat merenja mogu negativno uticati razni talasi koji dolaze spolja (recimo drugi zvučni talas koji ne potiče od senzora može uticati na krajnji rezultat merenja itd.)

2. Ukoliko se merenje vrši uz pomoć senzora koji koristi svetlosni talas onda je mana to što rezultat zavisi od koeficijenta refleksije svetlosti datog predmeta čija se udaljenost meri. Ako je refleksija slabija onda to može dovesti do grešaka u merenju.

# Glavne komponente korišćene u realizaciji projekta i njihove karakteristike

Glavne komponente koje se koriste u ovom projektu su razvojno okruženje kompanije *MikroElektronika* pod nazivom *mikromedia Plus for STM32* kao i klik pločica *Light Ranger 4 Click* iste kompanije.

## Mikromedia Plus for STM32

Slika 5. mikromedia Plus for STM32 razvojno okruženje

Na slici 5. prikazano je razvojno okruženje *mikromedia Plus for STM32*.

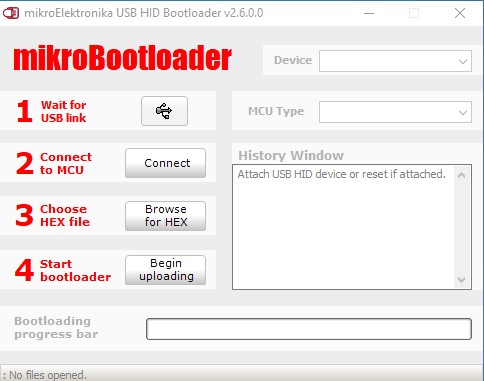
Razvojno okruženje *mikromedia Plus for STM32* je kompaktno dizajnirana razvojna ploča koja ima 4.3” TFT displej osjetljiv na dodir kao i mnoštvo periferija a sve to pokreće *STM32F746ZG* mikrokontroler iz ARM® Cortex™-M7 serije proizvođača *ST Microelectronics*.

Kao što je već navedeno, ovo razvojno okruženje ima mnoštvo periferija povezanih na 144-pinski mikrokontroler. Neke od tih periferija su sledeće:

* 4.3” TFT displej 480x272 osjetljiv na dodir,
* akcelerometar,
* slot za mikro SD karticu,
* bazer,
* IR prijemnik,
* RGB svetleće diode,
* PIN fotodioda,
* temperaturni senzor,
* 2.4GHz RF primopredajnik,
* Ethernet primopredajnik,
* 8Mbit fleš (eng. *flash*) memorija,
* RTC baterija,
* Li-Po baterijski punjač.

Osim ovih periferija, na ploči se nalazi i MINI-B USB konektor, 1.27mm terminal za napajanje, 2x5 JTAG konektor i prekidač za uključivanje i isključivanje.

Za ovo razvojno okruženje takođe postoji i ekspanziona ploča koja ima četiri mikroBUS konektora za povezivanje klik pločica.

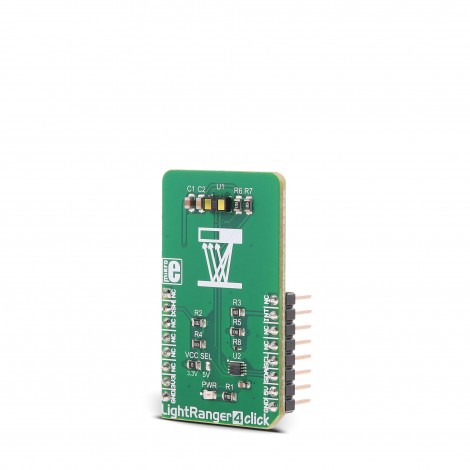
Programiranje mikrokontrolera na ovoj pločici je veoma jednostavno i može se izvršiti pomoću programatora *mikroElektronika USB HID Bootloader v2.6.0.0* kompanije *MikroElektronika*. Izgled programa koji se koristi za spuštanje koda (odnosno .hex fajla) na mikrokontroler, prikazano je na slici 6.

Korišćenje ovog programa je veoma jednostavno i vrši se u tri koraka. Prvo je potrebno resetovati samu ploču (pritiskom na *reset* taster koji se nalazi na ploči). Nakon toga, ikonica koja označava USB kabl označena kao broj 1. u programu postaje crvena. Dok je ikonica crvena, potrebno je kliknuti na taster *Connect*. Nakon toga, moguće je odabrati *.hex* fajl pritiskom na taster *Browse for HEX.* Na kraju je potrebno kliknuti na *Begin uploading* i na taj način započinje programiranje mikrokontrolera. Kada se programiranje završi, ploča se sama resetuje čime je postupak programiranja mikrokontrolera završen.

Slika 6. mikroElektronika USB HID Bootloader v2.6.0.0

Kod za potrebe ovog projekta je pisan u IDE (eng. *integrated development environment*) okruženju *STM32 Cube IDE* dok su biblioteke za periferije kao i podešavanja mikrokontrolera odrađena u programu *STM32 Cube MX*.

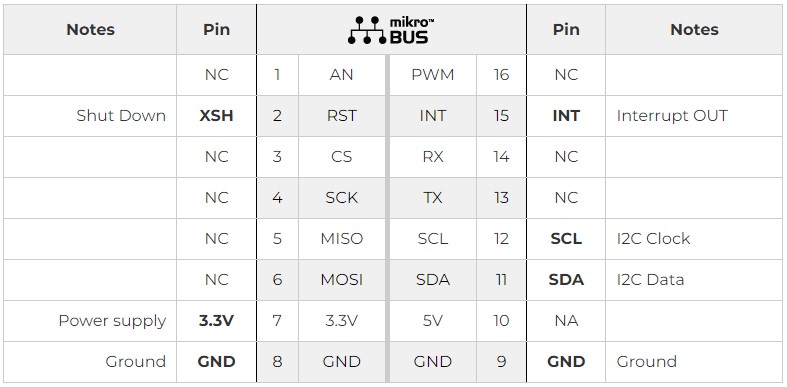
## Light Ranger 4 Click

Na slici 7. prikazana je klik pločica *Light Ranger 4* Click kompanije *MikroElektronika*.

Slika 7. Light Ranger 4 Click

Ova klik pločica se koristi za precizno mjerenje udaljenosti i princip rada se zasniva na ToF (eng. *Time of Flight*) principu mjerenja. Senzor *VL53L1X* kompanije *ST Microelectronics* je zapravo potpuni modul za mjerenje u samo jednom čipu. Kako u sebi sadrži mikrokontroler, veoma je jednostavno raditi sa njim. Ovaj modul ima samo priključke za I2C komunikaciju kao i XSH i INT priključke. Posjeduje ugrađen laser Klase 1 koji emituje laserski snop IR zračenja (940 nm) koji nije opasan po ljudski vid, fizičke infracrvene filtre kao i optiku za postizanje najboljeg opsega mjerenja u različitim uslovima ambijentalnog osvjetljenja. Može da mjeri udaljenosti do 4000mm frekvencijom do 50Hz.

Priključci ove klik pločice (eng. *pinout*) prikazani su na slici 8.



Slika 8. Pinout klik pločice Light Ranger 4

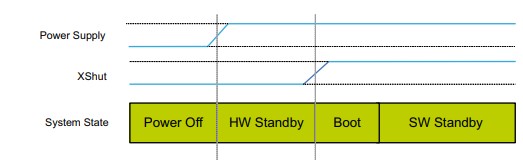
Ova pločica ima sledeće priključke:

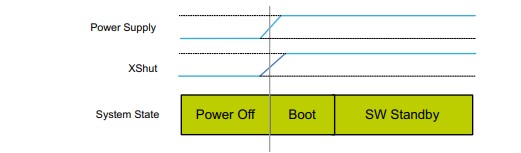
* Dva priključka za masu (GND),
* Priključak za napajanje od 3.3V,
* Priključke za I2C komunikaciju (SCL i SDA),
* XSH priključak i
* INT priključak.

Na pločici postoji i džamper kojim biramo da li će naponski nivo logičke jedinice biti 3.3V ili 5V. U ovom slučaju, izabrali smo da to bude 3.3V. Implementirana je i indikacija odnosno LED koja služi za indikaciju da je napajanje ispravno povezano.

Što se tiče priključka koji je označen sa XSH, to je skraćenica od *XSHUTDOWN* i koristi se za kontrolu moda senzora. Senzor može biti u hardverskom i softverskom modu pripravnosti (eng. *standby*). Ovaj priključak može biti povezan sa mikrokontrolerom na razvojnoj pločici i tada taj kontroler kontroliše mod senzora. Druga opcija je da se preko pulap (eng. *pull up*) otpornika ovaj priključak veže na napon napajanja i tada nije moguće da senzor bude u hardverskom modu pripravnsti već odmah nakon uključenja napajanja ulazi u softverski mod pripravnosti. Na slici 9. je prikazana sekvenca pri uključivanju senzora.

Slika 9. Uticaj priključka XSHUT na uključivanje senzora

 S obzirom da za našu realizaciju projekta nije bilo potrebno da koristimo hardverski mod pripravnosti, priključak XSHUT smo spojili na napon napajanja. Na taj način, dobili smo sekvencu uključivanja senzora koja je prikazana na slici 10.



Slika 10. Sekvenca prilikom uključivanja senzora u našoj realizaciji projekta

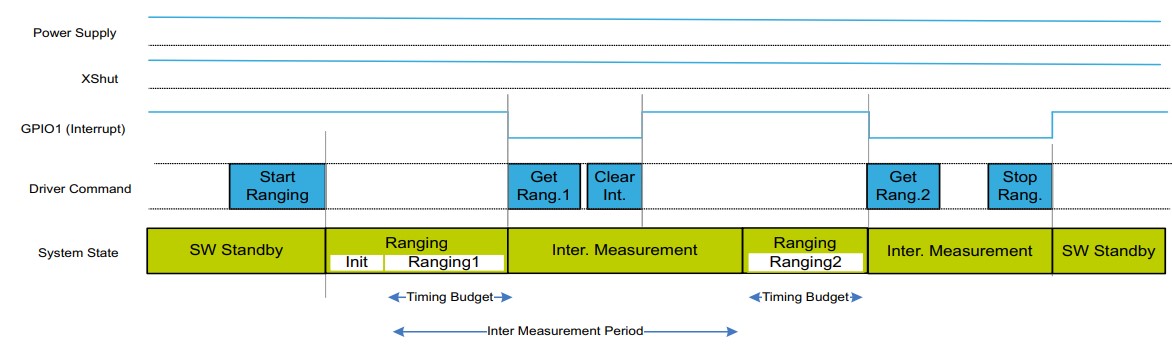
INT priključak služi da obavjesti mikrokontroler na razvojnoj pločici da su se desile određene bitne promjene. Ovaj priključak nismo koristili tokom realizacije projekta.

Kada govorimo o distanci koju je potrebno mjeriti, senzor *VL53L1X* ima tri moda rada:

* mod za mjerenje kratkih distanci (do 1300mm),
* mod za mjerenje srednjih distanci (od 1300mm do 3000mm) i
* mod za mjerenje dužih distanci (od 3000mm do 4000mm).

Naravno, sa svakim modom je moguće mjeriti bilo koji opseg od 0mm do 4000mm, međutim izborom pravog moda dobijamo preciznije i tačnije rezultate mjerenja te se zbog toga preporučuje korišćenje modova.

Dva veoma važna parametra kod senzora su i vrijeme mjerenja (eng. *timing budget*) kao i vrijeme koje protekne između dva uzastopna mjerenja (eng. *inter measurement period*). Na slici 11. je ilustrovano šta tačno ova dva parametra znače.



Komunikacija klik pločice sa razvojnim okruženjem se odvija preko I2C komunikacije. Maksimalna brzina I2C komunikacije ove klik pločice je 400kbit/s. Adresa uređaja je 0x52 ali se ona može mijenjati. Sa senzorom se komunicira tako što se prvo šalje registar u koji se upisuje ili iz kojeg se čita podatak. Ukoliko se podatak upisuje, nakon što se pošalje adresa registra, šalje se i vrijednost odnosno podatak koji se upisuje.

Slika 11. Vremenski parametri senzora

# Blok šema sistema

Na slici 12. je prikazana blok šema sistema.



Slika 12. Blok šema sistema

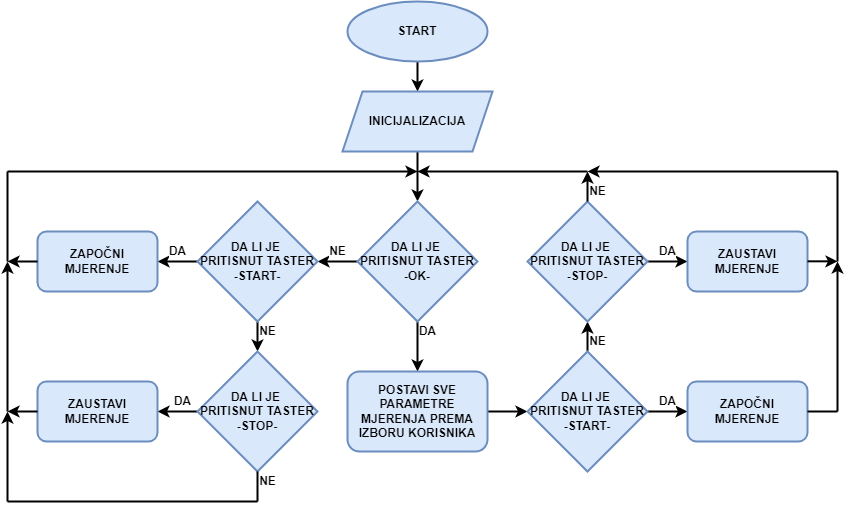
Sistem se sastoji od mikrokontrolera *STM32F746ZG* koji je centralni dio i upravlja ostatkom sistema, TFT displeja osjetljivog na dodir i senzora rastojanja VL53L1X. Takođe, postoje i dijelovi koji su neophodni za ispravno funkcionisanje mikrokontrolera i ostatka sistema poput napajanja, reset kola, priključaka za programiranje, butloudera (eng. *bootloader*) itd.

Mikrokontroler komunicira sa TFT displejom putem I2C komunikacije kanala 1, dok sa senzorom rastojanja komunicira preko drugog kanala I2C komunikacije. Takođe, za ispravan rad odnosno komunikaciju sa TFT displejom koriste se i digitalni ulazno izlazni priključci mikrokontrolera.

Na TFT displeju se prikazuje aplikacija kojom korisnik upravlja tako što unosi određene parametre za mjerenje, a pritiskom na START započinje se mjerenje i mogu se očitati rezultati mjerenja.

Senzor rastojanja dobija informacije sa mikrokontrolera koje služe za njegovu inicijalizaciju kao i podešavanje parametara i nakon toga vrši mjerenje. Rezultate mjerenja preko I2C komunikacije šalje mikrokontroleru koji to dalje preko drugog kanala I2C šalje na displej gdje korisnik može očitati te rezultate.

# Algoritam rada sistema

Na slici 13. je prikazan dijagram toka za algoritam rada sistema.

Slika 13. Dijagram toka za algoritam rada sistema

Na početku izvršavanja koda, potrebno je izvršiti inicijalizaciju svih periferija mikrokontrolera kao i samog senzora odnosno klik pločice.

Nakon inicijalizacije, čeka se na akciju od strane korisnika. Ukoliko korisnik klikne na taster OK, svi podešeni parametri mjerenja se povlače sa TFT displeja i šalju na senzor. Nakon toga čeka se na pritisak na taster START ili STOP. Ukoliko se pritisne taster START počinje mjerenje i rezultati mjerenja se ispisuju na TFT displeju. Pritiskom na taster STOP mjerenje se zaustavlja.

Važno je napomenuti i da postoje opcije za podešavanje parametara mjerenja koje se mogu odabrati a postavljaju se pritiskom na taster OK. Ovaj dio će detaljnije biti opisan u poglavlju *Aplikacija – korisničko uputstvo*.

# Biblioteka (drajver) za senzor VL53L1X (klik pločicu)

Za senzor *VL53L1X* postoji razvijena biblioteka odnosno drajver od strane kompanije *ST Microelectronics* koja sadrži API (eng. *Application programming interface*) funkcije pomoću kojih je moguće upravljati senzorom. Tokom izrade projekta, uključili smo biblioteku i iskoristili potrebne funkcije za realizovanu aplikaciju.

U nastavku će biti navedene funkcije koje su korišćene tokom realizacije projekta.

Prvo je potrebno izvršiti definisanje određenih struktura podataka, što je urađeno na sledeći način:

VL53L1\_RangingMeasurementData\_t RangingData;

VL53L1\_Dev\_t vl53l1\_c;

VL53L1\_DEV Dev = &vl53l1\_c;

Na ovaj način, definiše se podatak *RangingData* koji je neophodan za funkciju koja vraća trenutno rastojanje objekta od senzora. Nakon toga, definišemo strukturu Dev koja predstavlja senzor.

Nakon definisanja struktura, potrebno je povezati API funkcije sa hardverskim drajverom odnosno I2C komunikacijom. Odmah je izvršena i inicijalizacija adrese senzora. To je urađeno na sledeći način:

Dev->I2cHandle = &hi2c2;

Dev->I2cDevAddr = 0x52;

Za inicijalizaciju samog senzora, potrebno je pozvati sledeće funkcije:

VL53L1\_WaitDeviceBooted(Dev );

VL53L1\_DataInit(Dev );

VL53L1\_StaticInit(Dev );

Funkcija *VL53L1\_WaitDeviceBooted()* se koristi da bismo bili sigurni da je senzor ispravno uključen i spreman za rad. Funkcija *VL53L1\_DataInit()* se poziva samo jednom i ona vrši inicijalizaciju senzora. Treća funkcija koja je pozvana je *VL53L1\_StaticInit()* i njena uloga je da omogući učitavanje podešavanja senzor.

Sve ove funkcije izvršavaju se samo jednom nakon uključivanja sistema. U nastavku će biti objašnjene funkcije koje se pozivaju više puta u beskonačnoj petlji.

Funkcija *VL53L1\_StartMeasurement()* se poziva za početak mjerenja. Za kraj odnosno zaustavljanje mjerenja se poziva funkcija *VL53L1\_StopMeasurement()*. Kao parametar potrebno je proslediti strukturu senzora (u našem slučaju *Dev*).

Kao što je ranije navedeno, senzor ima različite modove rada za mjerenja različitih udaljenosti. Funkcija kojom se bira mod rada je *VL53L1\_SetDistanceMode()* i kao parametre je potrebno proslediti strukturu senzora i mod (kratka, srednja ili duga udaljenost) mjerenja,

Preostala dva parametra koja smo ranije objasnili i potrebno ih je postaviti su vrijeme mjerenja i vrijeme između dva uzastopna mjerenja. Vrijeme mjerenja (eng. *Timing budget*) se postavlja funkcijom *VL53L1\_SetMeasurementTimingBudgetMicroSeconds*() i potrebno joj je proslediti dva parametra, strukturu senzora i vrijeme u mikro sekundama. Vrijeme između dva uzastopna mjerenja se postavlja funkcijom *VL53L1\_SetInterMeasurementPeriodMilliSeconds*() kojoj se takođe prosleđuju dva parametra, u ovom slučaju strukturu senzora i vrijeme u mili sekundama.

Za rezultate mjerenja, prvo je potrebno pokrenuti mjerenje. Nakon toga, poziva se funkcija koja blokira izvršavanje koda dok mjerenje ne bude gotovo i ona je definisana kao *VL53L1\_WaitMeasurementDataReady()* i prosleđuje joj se struktura senzora. Nakon toga se poziva funkcija koja vraća izmjerenu vrijednost udaljenosti *VL53L1\_GetRangingMeasurementData(Dev, &RangingData )*. Prvi parametar je struktura senzora a drugi je referenca na strukturu koja se koristi za mjerenje udaljenosti. Polje strukture *RangingData* koje daje informaciju o udaljenosti objekta je *RangingData.RangeMilliMeter*.

# Biblioteka (drajver) za TFT displej

Slika 14. Izgled programa GUI builder

Biblioteka odnosno drajver za TFT displej, realizovali smo koršćenjem programa *GUI Builder*. Izgled ovog programa prikazan je na slici 14.

Kao što se može vidjeti na slici, u vrhu se mogu birati različiti objekti i dodavati u prozor koji će biti iscrtan na displeju. Za potrebe našeg projekta, koristili smo objekte za kreiranje teksta, tastere, prozor za ispis broja (eng. *edit*) i slajder. S obzirom da je bilo potrebno realizovati dva prozora u okviru ove aplikacije, koristili smo i objekat koji pravi više stranica (eng. *multipage*).

Nakon što se izaberu i rasporede svi objekti u aplikaciji *GUI Builder*, generiše se *.c* fajl sa kojim se dalje manipuliše uz pomoć nekih ugrađenih odnosno generisanih funkcija.

Osim ugrađenih funkcija u *.c* fajlu, definisali smo dodatne funkcije za lakše korišćenje odnosno manipulaciju podacima sa TFT displeja. Ove funkcije biće objašnjene u nastavku.

void WriteDistance(int distance) {

char distance\_tmp[5];

if(distance == -1) {

sprintf((char\*)distance\_tmp, "STOP\r");

EDIT\_SetText(WM\_GetDialogItem(hDialog\_app,ID\_EDIT\_0),distance\_tmp);

}

else {

sprintf((char\*)distance\_tmp, "%d\r",distance);

EDIT\_SetText(WM\_GetDialogItem(hDialog\_app,ID\_EDIT\_0),distance\_tmp);

}

}

Ova funkcija ispisuje vrijednost udaljenosti na displej. Potrebno je proslediti joj udaljenost koja je cjelobrojni tip, nema povratnu vrijednost.

int GetTimingBudget\_Percentage(void) {

return SLIDER\_GetValue(WM\_GetDialogItem(hDialog\_sett, ID\_SLIDER\_0));

}

Ova funkcija vraća trenutnu vrijednost slajdera za vrijeme mjerenja u procentima. Nije potrebno prosleđivati joj parametar.

int GetInterMeasPeriod\_Percentage(void) {

return SLIDER\_GetValue(WM\_GetDialogItem(hDialog\_sett, ID\_SLIDER\_1));

}

Ova funkcija vraća trenutnu vrijednost slajdera za vrijeme između dva uzastopna mjerenja u procentima. Nije potrebno prosleđivati joj parametar.

bool OK\_Button(void) {

if(BUTTON\_IsPressed(WM\_GetDialogItem(hDialog\_sett,ID\_BUTTON\_2))==0)

return false;

else

return true;

}

Ova funkcija vraća informaciju o trenutnom stanju tastera OK. Ukoliko je pritisnut vraća vrijednost *true*, u suprotnom vraća vrijednost *false*. Na isti način funkcionišu i ostale funkcije za očitavanje stanja tastera. Definisane su na sledeći način:

bool START\_Button(void)

bool STOP\_Button(void)

int DistanceMode(void) {

return RADIO\_GetValue(WM\_GetDialogItem(hDialog\_sett, ID\_RADIO\_0));

}

Ovo je poslednja funkcija koju smo definisali. Ona se koristi za dobijanje informacije sa TFT displeja o tome koji mod mjerenja je izabran.

# Glavni dio koda (eng. *main*)

U glavnom dijelu koda odnosno *main.c* prvo se izvršava inicijalizacija svih periferija mikrokontrolera kao i same klik pločice odnosno senzora udaljenosti. Ovaj dio koda nije potrebno detaljnije objašnjavati, te će akcenat biti na dijelu koda koji se izvršava u beskonačnoj petlji.

Glavni dio koda koji se izvršava u beskonačnoj petlji, prikazan je u sledećem dijelu teksta:

X\_koordinata=STMPE610\_GetX\_TOUCH();

Y\_koordinata=STMPE610\_GetY\_TOUCH();

STMPE610\_read\_xyz();

GUI\_TOUCH\_Exec();

keyPressed=GUI\_GetKey();

GUI\_Delay(1);

GUI\_SetColor(GUI\_BLACK);

GUI\_SetFont(&GUI\_Font20B\_ASCII);

U ovom dijelu koda vrši se očitavanje sa displeja odnosno rezistivne folije da li je i koji dio displeja pritisnut. Takođe, postavljaju se boja i font slova koji će biti ispisani na displeju.

if(OK\_Button()) {

start = false;

VL53L1\_StopMeasurement(Dev);

distanceMode = DistanceMode();

switch(distanceMode) {

case SHORT\_DISTANCE\_MODE:

VL53L1\_SetDistanceMode(Dev, VL53L1\_DISTANCEMODE\_SHORT);

timingBudget = ((TB\_HIGH-TB\_SHORT\_LOW)\*

((float)GetTimingBudget\_Percentage()/100))+TB\_SHORT\_LOW;

break;

case MEDIUM\_DISTANCE\_MODE:

VL53L1\_SetDistanceMode(Dev, VL53L1\_DISTANCEMODE\_MEDIUM);

timingBudget = ((TB\_HIGH-TB\_MEDIUM\_LOW)\*

((float)GetTimingBudget\_Percentage()/100))+TB\_MEDIUM\_LOW;

break;

case LONG\_DISTANCE\_MODE:

VL53L1\_SetDistanceMode(Dev, VL53L1\_DISTANCEMODE\_LONG);

timingBudget = ((TB\_HIGH-TB\_LONG\_LOW)\* (sledeća linija

((float)GetTimingBudget\_Percentage()/100))+TB\_LONG\_LOW;

break;

}

interMeasurementPeriod = ( (IMP\_HIGH - timingBudget+4) \*

((float)GetInterMeasPeriod\_Percentage()/100))+timingBudget+4;

timingBudget = timingBudget\*1000;

VL53L1\_SetDistanceMode(Dev, distanceMode);

VL53L1\_SetMeasurementTimingBudgetMicroSeconds(Dev, timingBudget);

VL53L1\_SetInterMeasurementPeriodMilliSeconds(Dev,interMeasurementPeriod);

}

U ovom dijelu koda se pritiskom na taster OK povlače sve vrijednosti odnosno parametri koji su podešeni od strane korisnika (mod mjerenja, vrijeme mjerenja i vrijeme između dva mjerenja). Nakon toga, ovi parametri se prosleđuju senzoru preko I2C komunikacije.

if(START\_Button())

start = true;

if(STOP\_Button())

start = false;

U ovom dijelu samo provjeravamo da li je pritisnut neki od taster START ili STOP i na osnovu toga se dodjeljuje vrijednost globalnoj promjenljivoj *start* koja može imati vrijednosti *true* ili *false*. Na osnovu ove promjenljive se dalje određuje da li mjerenje počinje ili se zaustavlja.

if(start) {

VL53L1\_StartMeasurement(Dev );

VL53L1\_WaitMeasurementDataReady(Dev );

VL53L1\_GetRangingMeasurementData(Dev, &RangingData);

WriteDistance(RangingData.RangeMilliMeter);

}

else {

VL53L1\_StopMeasurement(Dev);

WriteDistance(-1);

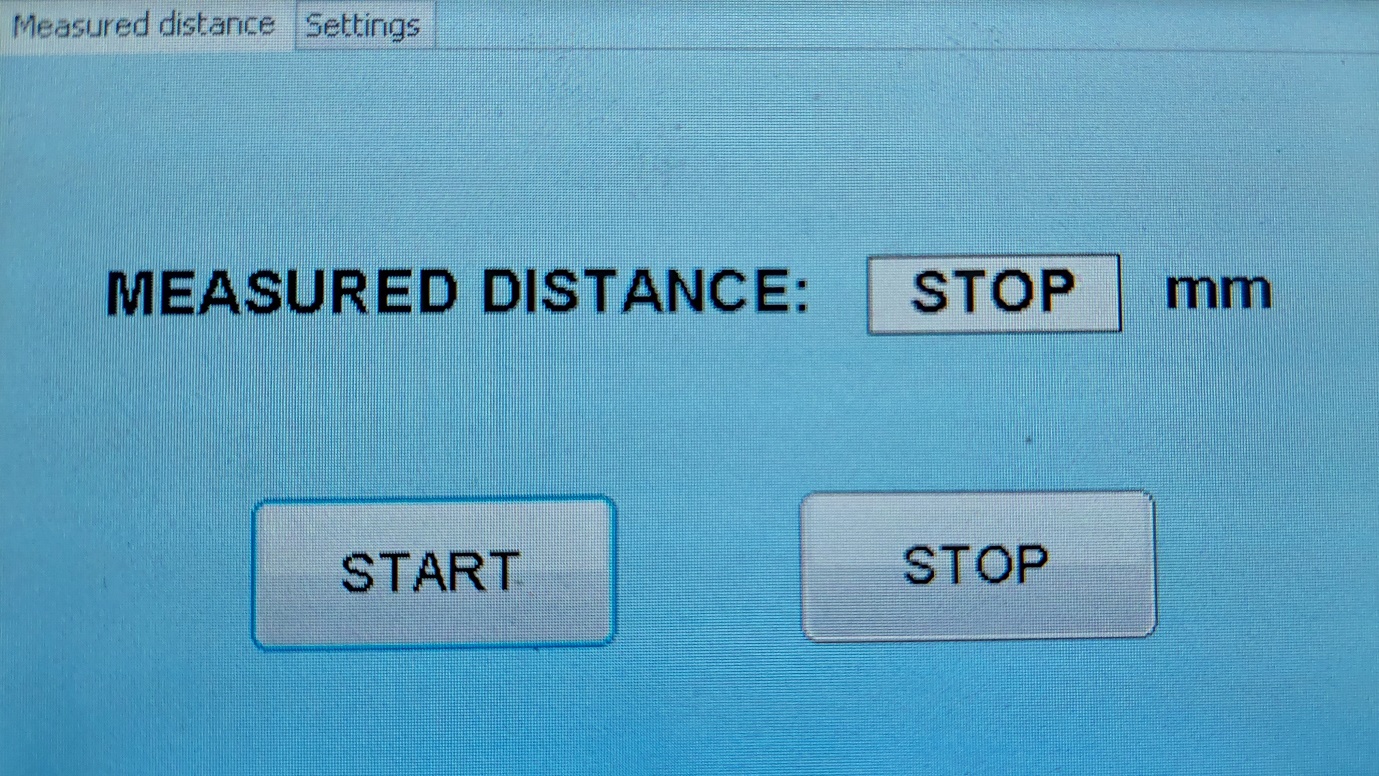
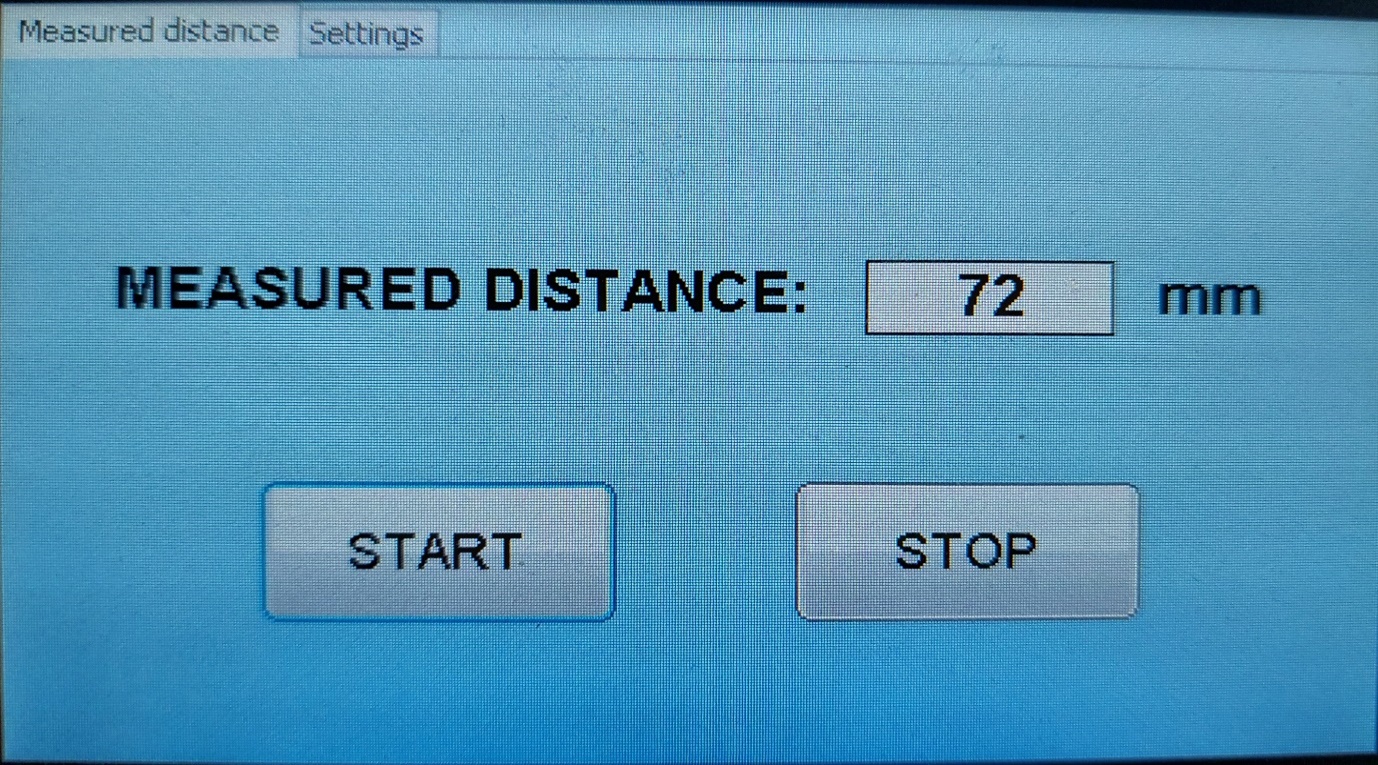
}

Ukoliko promjenljiva *start* ima vrijednost *true* započinje se mjerenje koje se vrši u svakoj iteraciji i ispisuje na TFT sve dok se ne pritisne taster STOP odnosno dok promjenljiva *start* ne dobije vrijednost *false*.

# Aplikacija – korisničko uputstvo

U ovom poglavlju biće opisan način na koji se koristi aplikacija odnosno realizovani projekat.

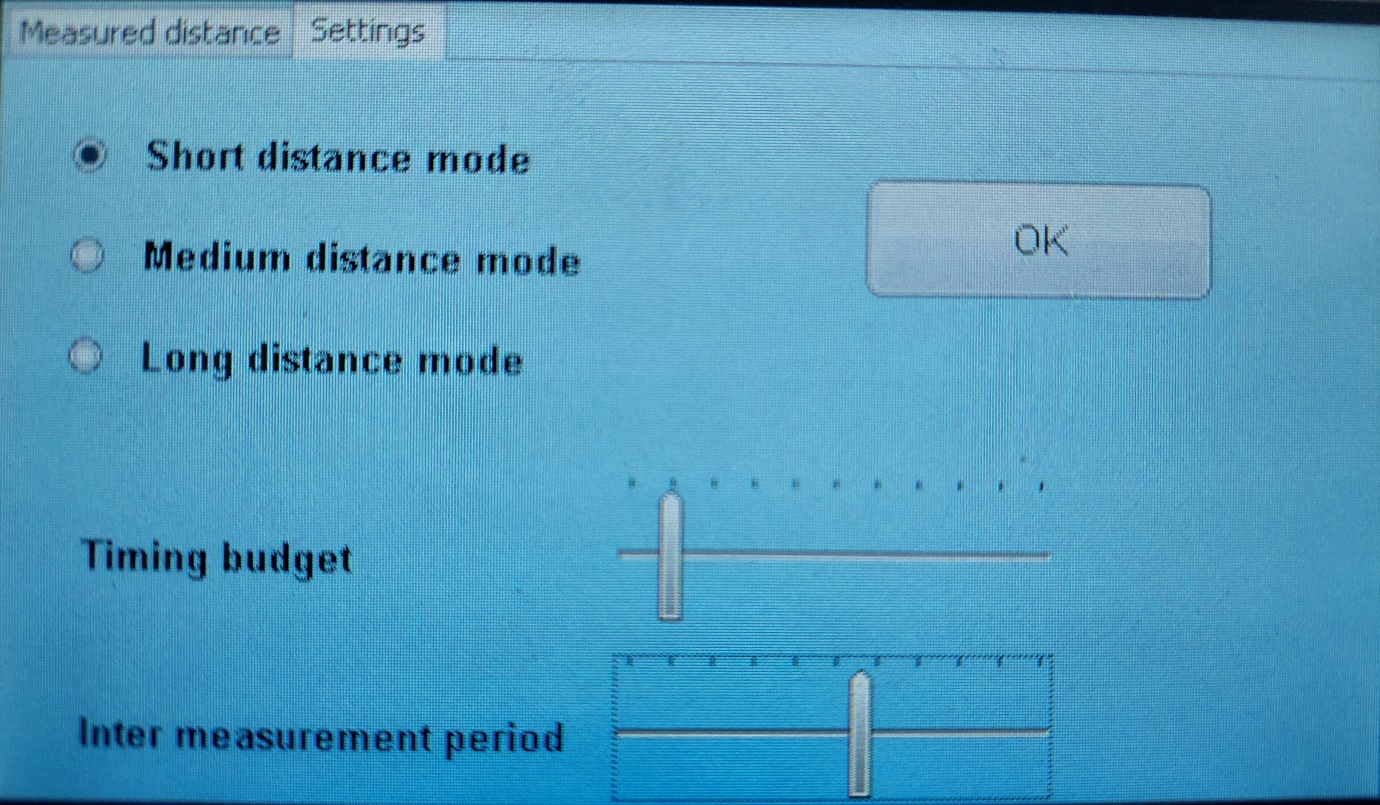
Na slici 15. i 16. prikazan je osnovni prozor aplikacije, odnosno prozor u kojem se prikazuju rezultati mjerenja.

Na slici 15. je prikazana aplikacija prije nego što je mjerenje započeto odnosno prije pritiska tastera START. Takođe, pritiskom na taster STOP, prozor ima isti izgled.

Slika 16. Aplikacija - mjerenje u toku

Slika 15. Aplikacija - zaustavljeno mjerenje

Na slici 16. je takođe prikazana aplikacija ali ovoga puta nakon što je mjerenje izvršeno. Možemo vidjeti da je ispisana trenutna udaljenost od objekta koja se mjeri.

Na slici 17. je prikazan drugi prozor aplikacije u kojem se podešavaju parametri mjerenja.

Slika 17. Aplikacija - podešavanja parametara mjerenja

Kao što se može vidjeti na slici, u gornjem lijevom uglu biramo mod mjerenja. Na donjoj polovini prozora nalaze se slajderi (eng. *slider*) kojim se bira vrijednost za vrijeme mjerenja i vrijeme između dva uzastopna mjerenja. Nakon što podesimo sve parametre, potrebno je pritisnuti taster OK kako bi se vrijednosti parametara prosledile senzoru.

Prelazak iz jednog prozora aplikacije u drugi vrši se izborom prozora (odnosno kartice) u gornjem lijevom uglu TFT displeja.

# Zaključak

Prilikom izrade projekta urađeno je sve što je bilo propisano projektnim zadatkom. Omogućeno je mjerenje udaljenosti određenih objekata na rastojanjima do 4m. Parametre mjerenja moguće je podešavati iz aplikacije koja je realizovana na razvojnom okruženju a grafički interfejs je realizovan na TFT displeju osjetljivom na dodir.

Na kraju je detaljno testirana aplikacija odnosno sve što je bilo zadato projektnim zadatkom. Izvršena su testiranja sva tri moda mjerenja kao i vremenskih parametara mjerenja. Rezultati testiranja se slažu sa zahtjevima projektnog zadatka.

Zaključak je da je projekat uspješno realizovan i da je implementirano i testirano sve što je bilo potrebno.

# Literatura

[ 1 ] https://www.optolab.ftn.uns.ac.rs/images/NASTAVA/OLS/files/ppt/0-stm32.pdf, mart 2022.

[ 2 ] https://www.mikroe.com/mikromedia-4-stm32f7, mart 2022.

[ 3 ] https://download.mikroe.com/documents/smart-displays/mikromedia/4/stm32f7/mikromedia-4-stm32f7-manual-v100.pdf, mart 2022.

[ 4 ] https://www.st.com/resource/en/datasheet/vl53l1x.pdf, mart 2022.

[ 5 ] https://www.st.com/resource/en/user\_manual/um2356-vl53l1x-api-user-manual-stmicroelectronics.pdf, mart 2022.

[ 6 ] <https://forum.digikey.com/t/adding-the-vl53l1x-driver-to-an-stm32cube-project/13276>, mart 2022.

[ 7 ] https://lastminuteengineers.com/arduino-sr04-ultrasonic-sensor-tutorial/

[ 8 ] John G. Webster , The measurement instrumentation and sensors