

ELEKTROTEHNIČKI FAKULTET U BEOGRADU

PROJEKAT IZ PREDMETA
32-BITNI MIKROKONTROLERI I PRIMENA

REALIZACIJA PAMETNOG SEMAFORA
POMOĆU SENZORA POKRETA



Profesor:

Dr. Dragan Vasiljevic

Student:

Branko Milosavljević 12/3174

Predmetni asistent:

Mr. Nenad Jovičić

Sadržaj

Projektni zadatak.....	2
Uvod	2
Hardverska realizacija.....	2
Softverska realizacija	7
Zaključak.....	9
Literatura	10

Projektni zadatak

Zadatak je projektovati sistem koji bi bio zadužen za simulaciju pametnog semafora koristeći senzor pokreta kao indikator postojanja saobraćaja na putu manjeg prioriteta.

Uvod

Projekat koji je predstavljen je rađen u okviru kursa „32-bitni mikrokontroleri i primena“ koji je deo master studija na odseku za elektroniku Elektrotehničkog fakulteta u Beogradu. Podrazumeva implementaciju sistema opisanog u projektnom zadatku koristeći se mikrokontrolerom iz familije STM32.

Razvoj sistema je realizovan na STM32VLDISCOVERY razvojnoj platformi, dok je za detekciju kretanja korišćen AMN11112 senzor pokreta. Semafor višeg i nižeg prioriteta predstavljen je sa po tri diode (crveno, žuto i zeleno svetlo).

Softverska realizacija sistema je realizovana u IAR Embedded Workbench for ARM Kickstart. Za potrebe testiranja korišćen je ST-link debugger, simulator i sama pločica.

U predstojećim poglavljima opisana je hardverska i softverska realizacija sistema, dok je u zaključku iznesen kritički osvrt na realizaciju sistema i predložen eventualan način za proširenje i poboljšanje performansi sistema.

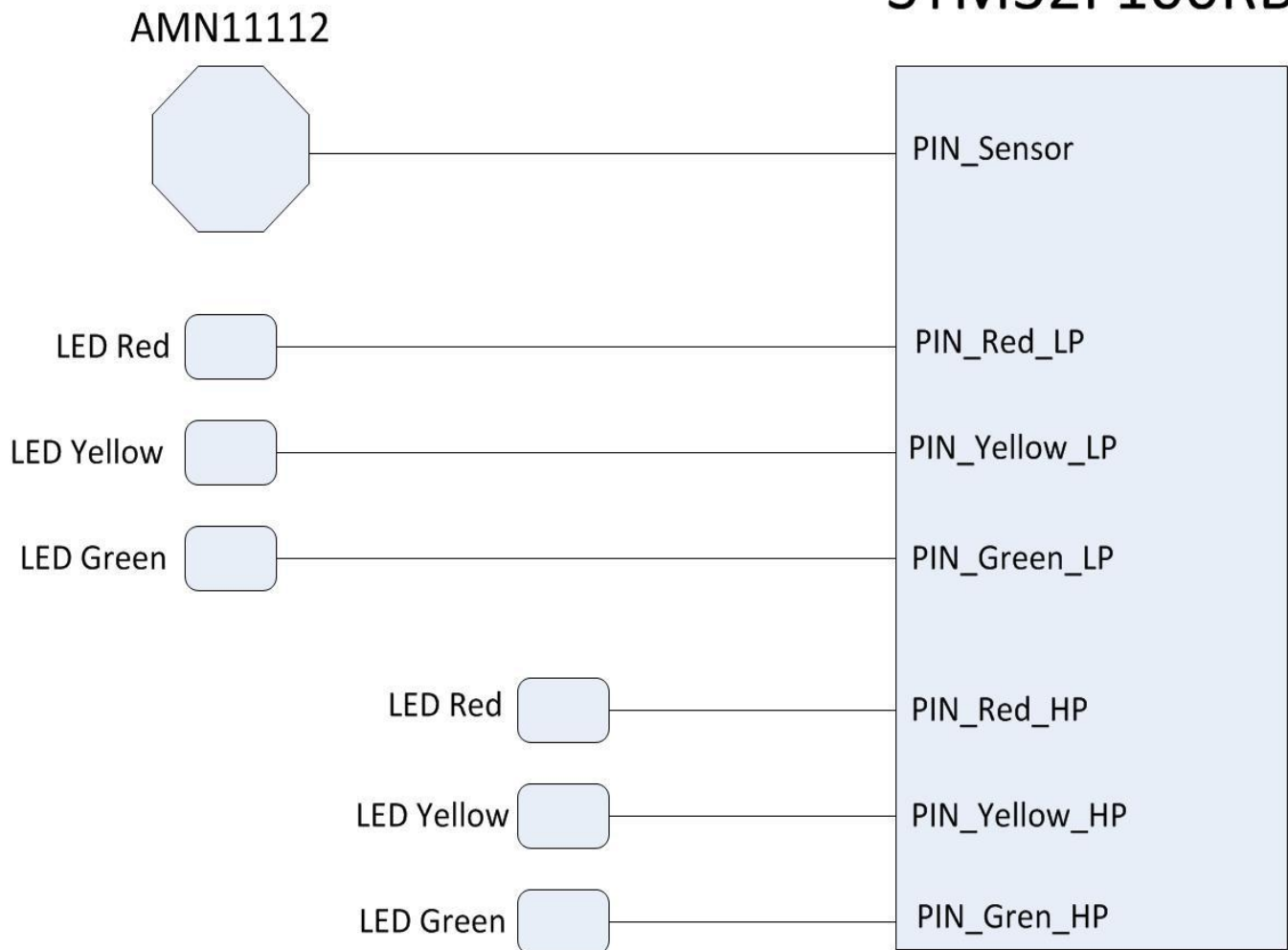
Hardverska realizacija

Sistem se sastoji iz dva semafora koje predstavljaju po tri diode, senzora pokreta i same razvojne platforme STM32VLDISCOVERY sa svojim periferijama.

Pinovi mikrokontrolera povezani su tako što je izlazni pin OUT senzora pokreta, povezan sa pinom PC13 mikrokontrolera. Pinovi PC1, PC2 i PC3 su rezervisani za diode koje predstavljaju prvi semafor, PA5, PA6 i PA7 su rezervisani za diode koje predstavljaju drugi semafor, diode Id3 i Id4 su povezane na pinove PC9 i PC8 koje signaliziraju aktivnost senzora i stanje u kome je senzor detektovao pokret. User taster je povezan na pin PA0 i služi za reset.

Na slici 1, može se videti uprošćena blok šema sistema.

STM32F100RB



Slika 1 – Blok šema sistema

Sam mikrokontroler je 32bitni i nosi punu oznaku STM32F100RB. Zasnovan je na ARM-Cortex M3 jezgru. ARM je 32-bitna RISK arhitektura i ravijena je od strane ARM Holdinga. Ranije je bila poznata kao Advanced RISK Machine, a pre toga kao Acorn RISC Machine. ARM arhitektura je najkorišćenija 32-bitna arhitektura seta instrukcija ako se posmatra broj proizvedenih jedinica. 2005. godine oko 98% od više od milijardu prodatih mobilnih telefona svake godine, je koristilo bar jedan ARM procesor. Od 2009 godine ARM procesori čine približno 90% svih namenskih 32-bitnih RISC procesora na tržištu, i imaju široku primenu u potrošačkoj elektronici, uključujući PDA (Personal digital assistant) uređaje, mobilne telefone, digitalne medija i muzičke plejere, portabl konzole, kalkulatore i kompjuterske periferije kao što su hard diskovi i ruteri.



Slika 2 - STM32VLDISCOVERY pločica

ARM2 ima 32-bitnu magistralu podataka , 26-bitnu adresnu magistralu i 27 32-bitnih registara. Programski kod je morao da bude smešten u prvih 64 MByte-a memorije, jer je program counter bio ograničen na 24 bita. ARM2 je bio najjednostavnij 32-bitni mikroprocesor u širokoj primeni na svetu sa samo 30, 000 tranzistora.

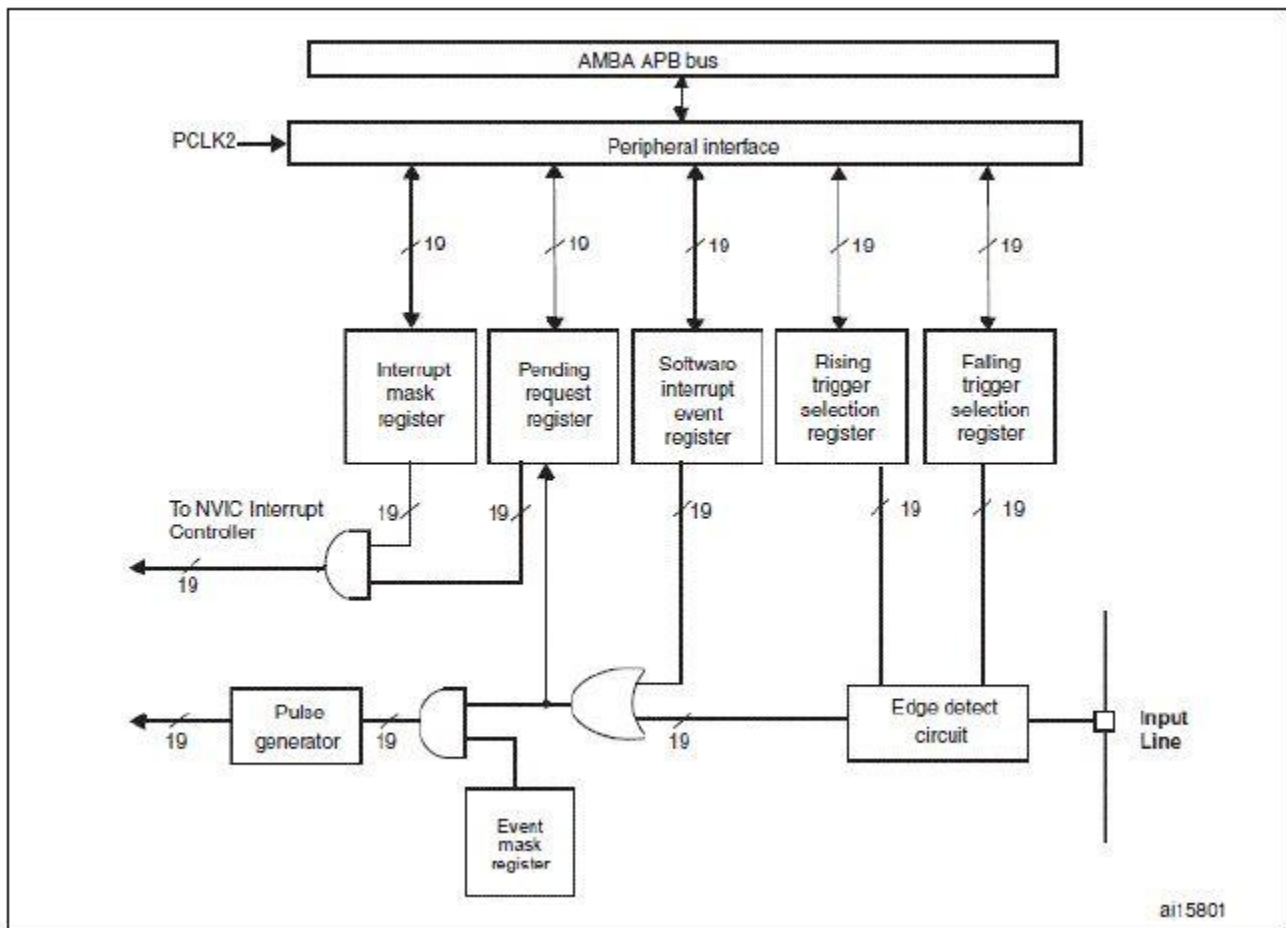
Mikrokontroler STM32F100RB sadrži 128Kb Flash memorije i 8Kb RAM-a. Odlikuje ga velika brzina i veliki broj periferija.

Kako je funkcionisanje sistema definisano prekidima, najvažnija periferija korišćena u projektu je kontrolor eksternih prekida.

Kontroler eksternih prekida/događaja se sastoji od devetnaest detektora ivica koji služe za generisanje zahteva za događajem/prekidom. Svakoju ulaznoj ivici se može nezavisno konfigurisati tip (npr.

reagovanje na ivicu) i odgovarajući triger (uzlazna i/ ili silazna ivica). Svaka linija se može nezavisno maskirati. Statusna linija zahteva za prekid se pamti u registru za čekanje.

Kako bi se obezbedilo generisanje prekida potrebno je prvo konfigurisati i omogućiti prekidnu liniju. To se postiže odabiranjem načina detektovanja ivica u odgovarajućim triger registrima i omogućavanjem prekida upisivanjem logičke jedinice u registar maskiranja prekida. Kada se dogodi odgovarajuća ivica generiše se zahtev za prekid. Bit koji označava da je zahtev za prekid na čekanju se generiše i zahtev se resetuje upisivanjem logičke jedinice u registar za čekanje. Prekid se, takođe, može generisati softverski, upisivanjem bita u softverski registar za prekide.

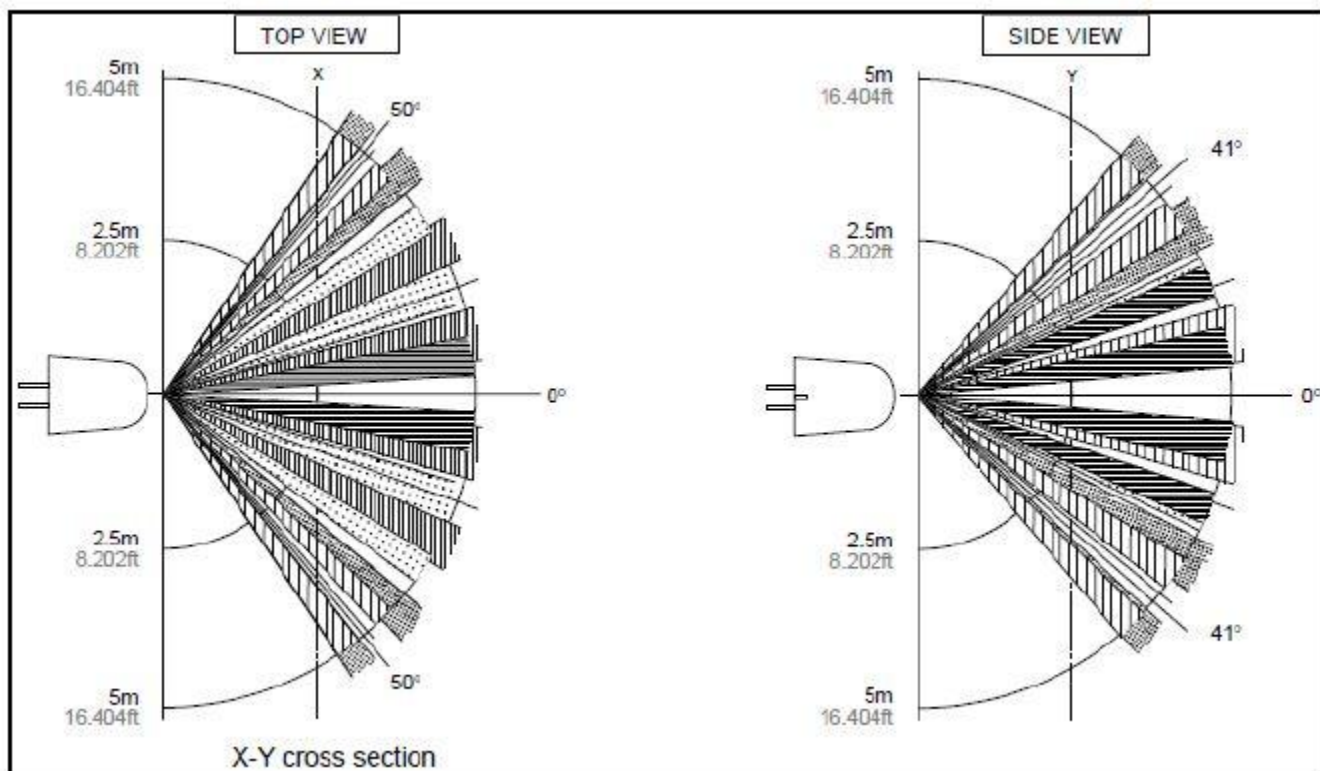


Slika 3 - Kontroler eksternih prekida

Ukoliko želimo da jednu od devetnaest linija iskoristimo za generisanje hardverskog prekida to se može postići tako što se konfiguriraju biti za maskiranje linija za prekid, Trigger Selection biti linija za prekid (EXTI_RTSR i EXTI_FTSR) i enable i mask biti koji mapiraju Nested Vector Interrupt Controller (NVIC) na

External Interrupt Controller. Softverski prekid generise se tako sto se konfigurišu biti za maskiranje linija za prekid u odgovarajućim registrima (EXTI_IMR, EXTI_EMR) i setovanjem potrebnih bita registra za generisanje softverskih prekida (EXTI_SWIER). EXTI linija 16 povezana je sa PVD izlazom, a EXTI linija 17 sa RTC Alarm događajem.

Što se tiče senzora pokreta koji je korišćen u projektu, on registruje promene infracrvenog zračenja do koga dolazi prilikom kretanja osobe ili pomeranja objekta, čija se temperatura razlikuje od temperature okruženja. Senzor AMN11112 je pasivni IR sensor koji registruje promene u intenzitetu infracrvenog svetla. Kada se pokret registruje, na izlazu senzora se postavlja logička jedinica (1). Kod senzora AMN11112 moze doci do greške u detekciji ukoliko izvor toplote nije čovek, ukoliko se ne menja temperatura ili ukoliko je izvor toplote statičan. Na pločici je ugrađen pojačavač koji znatno olakšava korišćenje senzora.



Slika 4 - Prostor koji detektuje senzor

Napajanje senzora mora biti između 3 i 6 V DC, dok je minimalni napon na izlazu Vdd-0.5V kada je pokret detektovan. Prostor u kojem se detektuje pokret ide od 82 stepena vertikalno do 100 stepeni u horizontalnom položaju. Maksimalan domet senzora iznosi oko 5 metara.

Softverska realizacija

Softverska realizacija sistema podrazumevala je usvajanje posebnog slučaja (u odnosu na običan semafor kao mašinu stanja), s' obzirom na to da je semafor specifičan, tj predstavlja semafor koji se nalazi na raskrsnici ulica visokog i niskog prioriteta.

Semafor je savršen primer sistema koji može da se realizuje preko mašine stanja. U sintaksnom smislu, upotrebila bi se switch-case struktura, međutim kako je drugi semafor niskog prioriteta i pali se samo u slučaju da je detektovan pešak(vozilo), rešeno je da se cela sekvenca promena svetla na semaforu radi u prekidnoj rutini, dok su u glavnom programu postavljena samo svetla koja odgovaraju situaciji u kojoj je otvorena ulica visokog prioriteta.

Što se tiče samog koda, realizovan je tako da se u prvom delu vrši inicijalizacija struktura koje će se koristiti kasnije u programu. Inicijalizuju se strukture za pinove opšte namene, kontrolor eksternog prekida, dovodi se clk signal na pinove koji će se koristiti, inicijalizuju se izlazni pinovi za svetla dioda, ulazni pin za senzor, kao i prekidi na pinu za senzor i tasteru za uključivanje, isključivanje sistema.

Posle inicijalizacije, program je realizovan u vidu beskonačne petlje pre koje su postavljena svetla koja odgovaraju otvaranju ulice visokog prioriteta.

U slučaju da je detektovan pokret (saobraćaj na ulici nižeg prioriteta), program ulazi u prekidnu rutinu u kojoj se debaunsira senzor i menjaju svetla po ranije utvrdjenim vremenima. Svetla se menjaju u vidu delay-a koje je realizovan u posebnom fajlu Counter.h. Postoji dva delay-a. Jedan koji odbrojava do kraja zadatog vremena, i drugi (delay ulice glavnog prioriteta) koji može biti prekinut, i čije se vreme pamti, pa se kao takvo upamćeno prosleđuje običnoj funkciji za delay u prekidnoj rutini. Na ovaj način se vrši sinhronizacija, i onemogućava da se semafor upali istog trenutka kada je detektovan pešak.

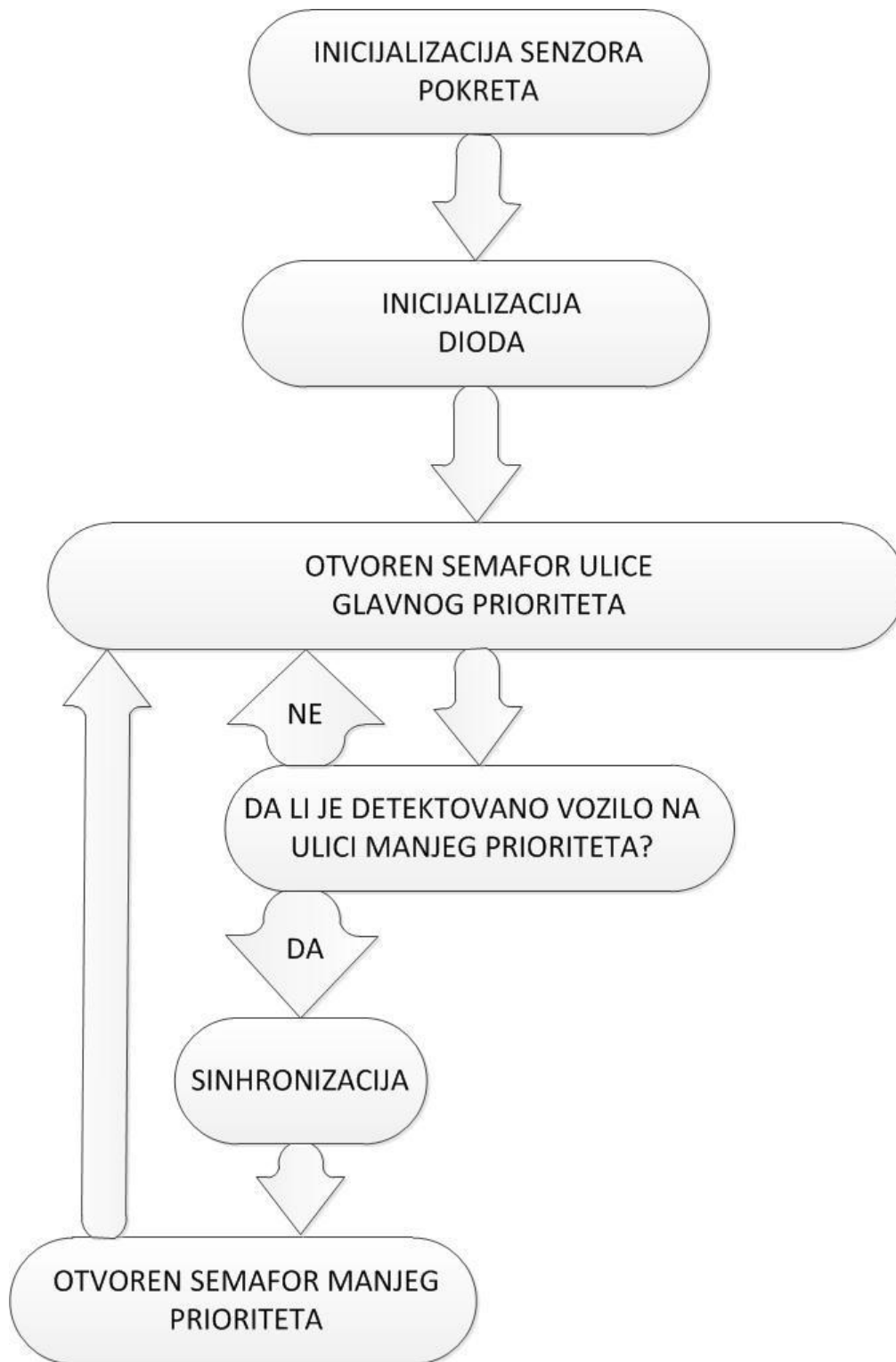
U slučaju da je senzor pokreta detektovao saobraćaj, pali se plava lampica kao indikator detekcije.

```
//diode prvog "semafora":
GPIO_InitStructure.GPIO_Pin = GPIO_Pin_1 | GPIO_Pin_2 | GPIO_Pin_3; // respektivno: crveno, zuto i zeleno
GPIO_InitStructure.GPIO_Speed = GPIO_Speed_10MHz;
GPIO_InitStructure.GPIO_Mode = GPIO_Mode_Out_PP;
GPIO_Init(GPIOC, &GPIO_InitStructure);

//diode drugog "semafora":
GPIO_InitStructure.GPIO_Pin = GPIO_Pin_5 | GPIO_Pin_6 | GPIO_Pin_7; //respektivno: crveno, zuto i zeleno
GPIO_InitStructure.GPIO_Speed = GPIO_Speed_10MHz;
GPIO_InitStructure.GPIO_Mode = GPIO_Mode_Out_PP;
GPIO_Init(GPIOA, &GPIO_InitStructure);
```

Slika 5 - Primer inicijalizacije dioda semafora

Na slici ispod, može se videti blok dijagram izvršavanja programa.



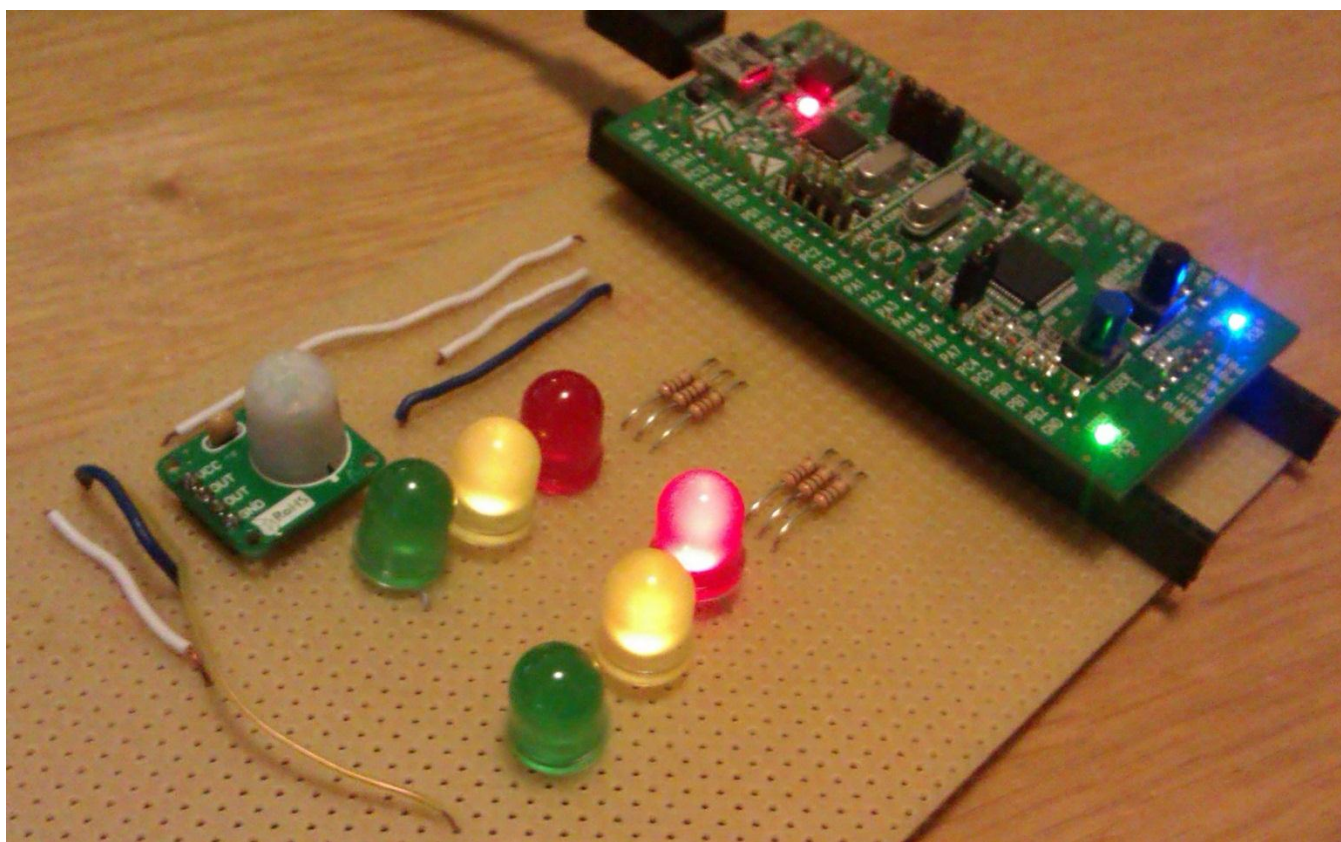
Slika 6 - Blok dijagram izvršavanja programa

Zaključak

Sistem koji je realizovan zahteva vrlo malo resursa, te je mogao biti realizovan i na mnogo slabijim mikrokontrolerima od korišćenog. Glavna ideja projekta je bila upoznavanje sa osnovnom komunikacijom sa mikrokontrolerima iz familije STM32 i sticanja nekog praktičnog iskustva u realizaciji hardvera (rada sa lemilicom, naručivanja komponenti, pisanje izveštaja i slično).

Sistem ima mnogo prostora za napredak u smislu softvera. Korišćenje mašine stanja i tajmera, u mnogo tome bi ostavili mogućnost da se obavi još neka akcija dok se čeka na istek vremena. Međutim, kako predloženi projekat nije zahtevao racionalnu upotrebu resursa, od takve realizacije se odustalo.

Sistem je testiran i radi po očekivanjima. Na slici ispod, možete videti kako sistem izgleda.



Slika 7 - Izgled sistema za simulaciju pametnog semafora

Literatura

- [1] RM0041 Reference manual for STM32F100xx, ST Microelectronics
- [2] UM0919 User Manual for STM32 value line Discovery, ST Microelectronics
- [3] Standard Firmware Library User Manual Rev2.pdf
- [4] UM0919 User Manual STM32VLDIDISCOVERY