

Univerzitet u Kragujevcu
Fakultet inženjerskih nauka



Seminarski rad
Arhitektura računarskih sistema

Tema:
144- pinski LPC2214 ARM mikrokontroler

Student:
Kosta Erić 561/2015

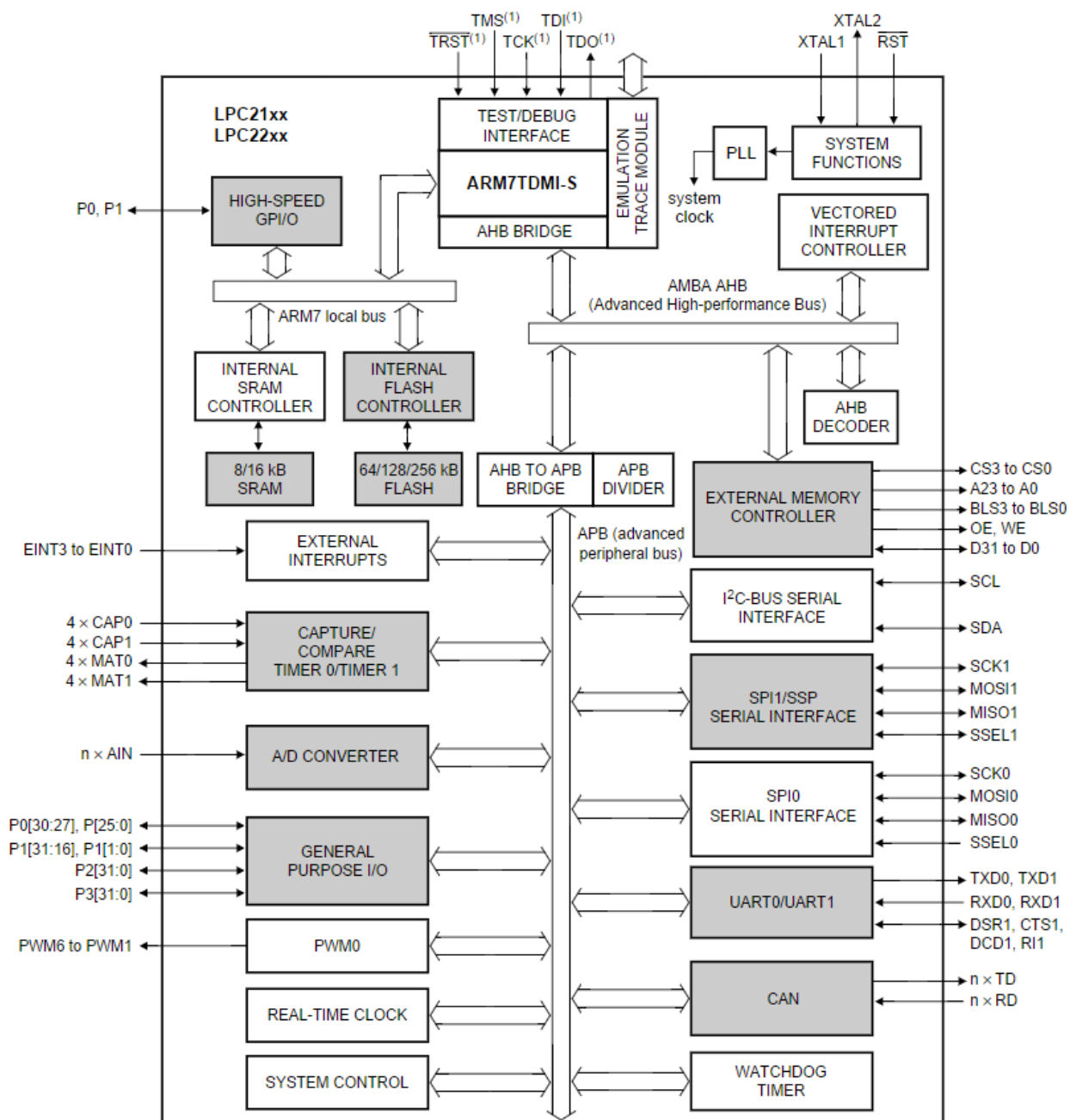
Predmetni profesor:
Aleksandar Peulić

SADRŽAJ

1. UVOD.....	2
2.ARHITEKTURA.....	4
2.1. ARM7TDMI-S procesor.....	5
2.2. Statička RAM memorija (SRAM)	6
2.3. Regulator napona.....	6
2.4. Kristalni oscilator.....	7
2.5. Ulazno/izlazni pinovi opšte namene (GPIO)	8
2.6. I2C Interfejs.....	9
2.7. Serijski periferni interfejs (SPI).....	10
2.8. WatchDog tajmer (WDT).....	11
2.9. Kovertor analognog signala u digitalni (ADC)	12
2.10. Clock u realnom vremenu (RTC – Real-time clock).....	12
2.11. Tajmeri opšte namene	12
2.12. Kontroler Flash memorije	13
2.13. MicroSD konektor	14
2.14. Programiranje mikrokontrolera	14
2.15. Kontroler prekida (interrupt)	16
2.16. UART	16
3. PROJEKTNİ ZADATAK.....	17
4. REALIZACIJA PROJEKTOG ZADATKA.....	17
5. ZAKLJUČAK	19
6. LITERATURA	20

1. UVOD

LPC 22xx familija mikrokontrolera (Slika 1) je zasnovana na 16/32 bitnom ARM7TDMI-S CPU procesoru sa emulacijom u realnom vremenu, kao i 64/128/256 kilobajtnom (kB) ugrađenom veoma brzom flash memorijom. Interfejs unutrašnje memorije širine 128-bitova i jedinstvena arhitektura akceleratora omogućavaju pokretanje 32-bitnih kodova pri maksimalnoj brzini clocka.



Slika 1: Blok šema 2214 mikrokontrolera

Sa svojim 64/144- pinskim paketima, minimalnom upotrebom napajanja, različitim 32- bitnim tajmerima, sa do 12 eksternih prekidnih (interrupt) pinova, i četiri 10- bitna ADC kanala kao i 46 ulazno/izlaznih (64- pinski paket) pinova za uređaje opšte namene, odnosno 8 10- bitnih kanala i 112 ulazno/izlaznih pinova (144- pinski paket), mikrokontroleri LPC 22xx porodice uređaja su praktično namenjeni za industrijsku kontrolu i kontrolu pristupa, a veliku primenu imaju i u medicinskim sistemima. Sa širokim spektrom interfejsa serijskih komunikacija, oni su takođe veoma pogodni za komunikacione brane (gateway), konvertere protokola, i ugrađene modeme kao i za veliki broj drugih primena opšte namene.

2.ARHITEKTURA

Osnovne karakteristike LPC 2214 144- pinskog ARM mikrokontrolera (Slika 2) su:

- LPC 2214 MCU procesor
- mini USB programabilan konektor
- JTAG konektor za programiranje i debugovanje
- Posebno reset kolo koje se aktivira pritiskom na dugme
- 1.8V i 3.3V regulatori napajanja
- 8 Mbit Flash memorije za SPI
- postolje za microSD karticu
- LED dioda
- Kristalni oscilator na samom kontroleru
- 16 kB statičke ram memorije kao i 256 kB programabilne flash memorije
- Interfejs širine do 128 bitova/ kao i akcelerator koji omogućava brzinu rada do 60 MHz
- Eksterna magistrala širine (8/16/32 bita)
- Programiranje mikrokontrolera je moguće korišćenjem Keil i Flash Magic softvera



Slika 2: LPC 2214 ARM 144-pin mikrokontroler

2.1. ARM7TDMI-S procesor

ARM7TDMI-S je 32-bitni mikroprocesor opšte namene, koji nudi visoke performanse uz veoma nisku potrošnju napajanja. ARM arhitektura je bazirana na principu kompjutera sa redukovanim setom instrukcija (RISC arhitektura), tako da je set instrukcija i mehanizam dekodiranja znatno jednostavniji nego kod uređaja baziranih na CISC arhitekturi (kompjuter sa kompleksnim setom instrukcija). Ovakva jednostavnost rezultira visokim protokom instrukcija i impresivnim odgovorom na prekide u realnom vremenu vrlo malog i efektivnog procesorskog jezgra.

Mnoge tehnike su implementirane kako bi svi delovi procesora i celokupnog sistema memorije mogli da funkcionišu neprekidno. Tipično, dok se jedna instrukcija izvršava, sledeća funkcija se dekodira, dok se treća po redu učitava iz memorije.

ARM7TDMI-S procesor takođe koristi jedinstvenu strategiju arhitekture poznatu kao THUMB, koja ga čini idealnim za procesovanje zahtevnih aplikacija koje imaju memorijsko ograničenje ili za aplikacije gde je gustina koda problem.

Ključna ideja koja stoji iza THUMB-a, je super-redukovani set instrukcija. Suštinski, ARM7TDMI-S procesor ima dva seta instrukcija:

- Standardni 32-bitni ARM set instrukcija, i
- 16-bitni THUMB set instrukcija

THUMB-ov 16-bitni set instrukcija omogućava duplo gušći kod u odnosu na standardni ARM kod zadržavajući većinu naprednih performansi ARM-a. Ovo je moguće jer THUMB kod izvršava operacije na istom 32-bitnom registru kao i sam ARM-ov kod.

THUMB kod može da proizvede do 65% veličine ARM koda, i 160% performansi ekvivalentnog ARM procesora koji je povezan na 16-bitni sistem memorije.

2.2. Statička RAM memorija (SRAM)

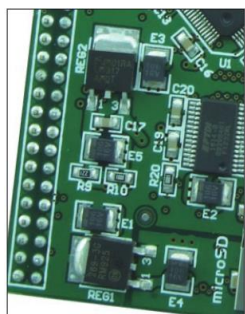
Statička RAM memorija na samom čipu može da si koristi i za kodiranje i za skladištenje podataka. Može biti veličine od 8 bitova, 16 bitova ili 32 bita.

LPC 2214 SRAM je dizajniran tako da može da mu se pristupi kao bajt-adresnoj memoriji. U njemu se nalazi i write-back (piši u nazad) bafer koji sprečava usporavanje procesora prilikom back-to-back ispisivanja.

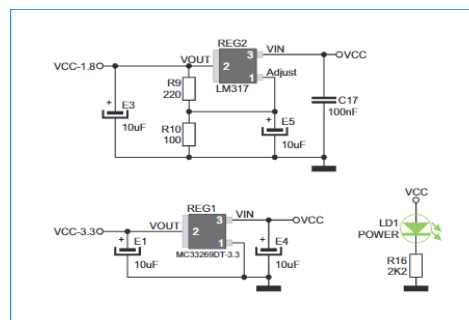
Write-back bafer uvek zadržava poslednji podatak koji je poslat iz softvera statičkoj memoriji. Taj podatak je upisan u SRAM jedino kada softver zahteva neki drugi upis (podatak je upisan samo kada softver vrši neko drugo upisivanje). Ukoliko se čip resetuje, u sadržaju statičke memorije neće postojati samo poslednji zahtev za upis. Bilo koji softver koji proverava sadržaj statičke ram memorije nakon resetovanja čipa, mora to da uzme u obzir. Dva identična ispisivanja podataka na neku lokaciju garantuju da taj zahtev za upis neće biti izgubljen nakon resetovanja.

2.3. Regulator napona

Mikrokontroler zahtevam dva izvora napajanja: 1.8V za procesor i 3.3V za ulazno/izlazne uređaje. Ploča dobija napajanje od 5V preko CN3 USB konektora koji se nalazi na samoj ploči.



Ukoliko se ploča napaja putem razvojnog sistema (EasyARM v6), funkcija regulatora napona ostaje ista. U tom slučaju, neophodno je pomeriti prekidač J2 (STANDALONE).



2.4. Kristalni oscilator

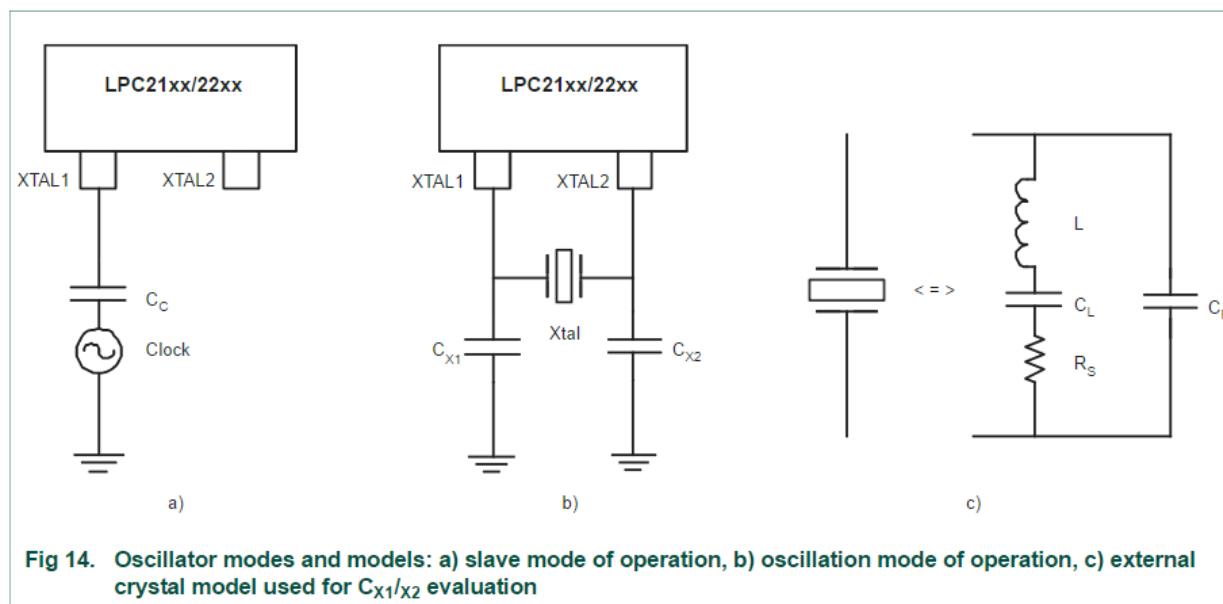
Ulazni signal od 50-50 cilusa sa rasponom frekvencije od 1 MHz do 25 MHz bi trebalo da se koristi na LPC 2214 čipu ako se napaja preko XTAL1 pina. Oscilator koji se nalazi na čipu mikrokontrolera sadrži spoljašnji kristal koji ima raspon frekvencije isključivo od 1 MHz do 25 MHz.

Izlazna frekvencija oscilatora se naziva Fosc, dok se frekvencija clocka ARM procesora označava kao CCLK. Fosc i CCLK imaju iste vrednosti dok je PLL (fazno zaključana petlja) sistem aktiviran i povezan.

Oscilator na čipu može da se izvršava kroz dva nivoa: slave mod i oscilatorni mod.

U slave modu ulazni signal clocka bi trebalo da se sprovede kroz kondenzator od 100 pF, sa amplitudom od najmanje 200 mVrms. XTAL 2 pin u ovakvoj konfiguraciji može da bude nepovezan. Ako je slave mod izabran, Fosc signal oscilatora može da se emituje na frakvenciji od 1MHz do 25 MHz.

Eksterne komponente i moduli koji se koriste i oscilatornom modu su prikazani na Slici 3. Pošto je otpor integrisan na čipu, jedino kristal i kondenzatori Cx1 i Cx2 moraju da budu povezani eksterno u slučaju osnovnog oscilatornog moda (osnovna frekvencija je predstavljena preko L, Cl i Rs). Ukoliko se na čipu izabere oscilatorni mod izvršavanja, ograničenja Fosc signala su od 1MHz do 25 MHz.

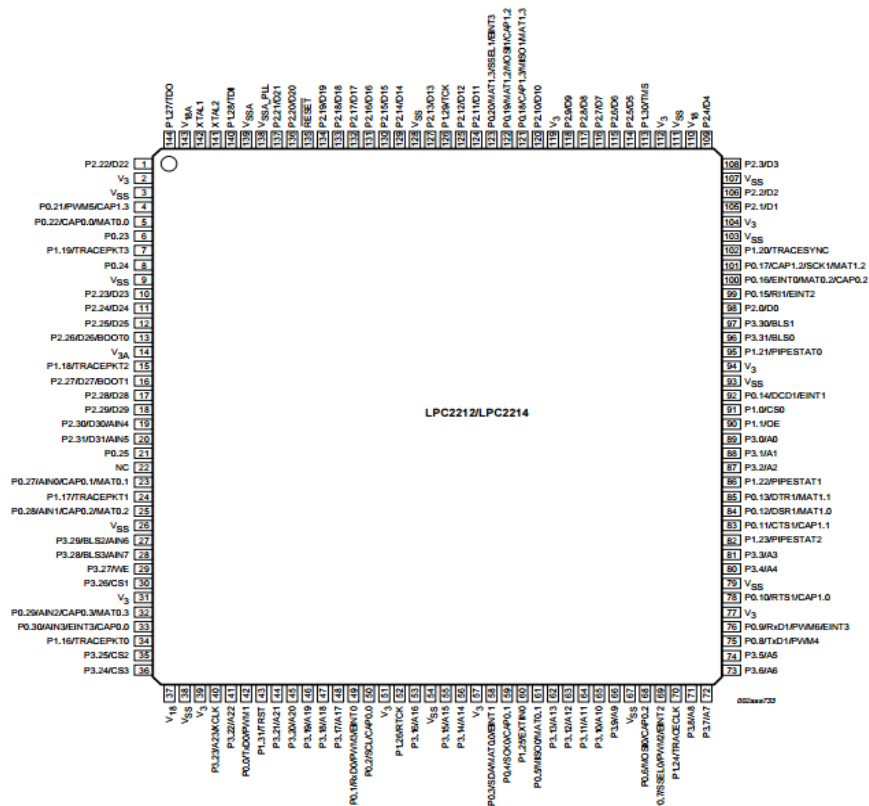


Slika 3

2.5. Ulazno/izlazni pinovi opšte namene (GPIO)

Glavne karakteristike GPIO (Slika 4) su:

- Svakom fizičkom GPIO portu može da se pristupi ili preko registara koji omogućavaju poboljšane karakteristike i ubrzavaju sam port, ili preko registara koji omogućavaju kompatibilnost sa ranijim LPC2000 uređajima
- Unapređene (ubrzane) GPIO funkcije su:
 - GPIO registri su locirani na ARM lokalnoj magistali tako da najbrže moguće I/O vreme pristupa može biti omogućeno.
 - Registri za maskiranje dozvoljavaju da se skupovi portova sagledavaju kao grupe, ostavljajući ostale bitove nepromenjene.
 - Svi registri mogu da se adresiraju kao bajtovi, reči ili polureči.
 - Celokupna vrednost porta može da se ispiše u jednoj instrukciji.
- Set i clear registri dozvoljavaju postavljanje jedne instrukcije ili brisanje bilo kog broja bitova na jednom portu.
- Smer svakog pina može individualno da se kontroliše.
- Svi I/O su postavljeni kao ulazni nakon resetovanja.

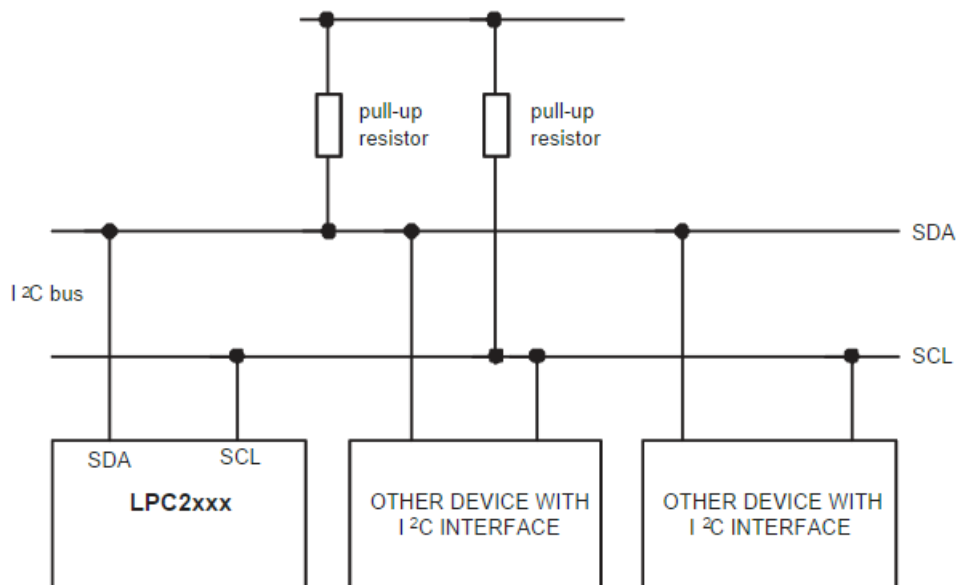


Slika 4: Blok šema pinova

2.6. I2C Interfejs

Tipična konfiguracija interfejsa I2C prikazana je na Slici 5. U zavisnosti od stanja bita usmerenja, dva tipa transfera podataka su moguća preko I2C- magistrale:

- Transfer podataka preko master predajnika do slave prijemnika. Prvi bajt koji je prenet preko predajnika predstavlja adresu prijemnika. Sledeći prati broj bitova podataka. Slave vraća bit prepoznavanja nakog svakog primljenog bita.
- Transfer podataka od slave predajnika do master prijemnika. Prvi bajt (slave adresa) je prenet putem mastera. Nakon toga slave vraća bit prepoznavanja. Sledi transfer bajtova podataka od slave predajnika ka master prijemniku. Master vraća bit prepoznavanja nakon pristiglih svih bajtova sem poslednjeg bajta. Na kraju transfera poslednjeg bajta, vraća se bit "ne prepoznavanja". Master uređaj generiše sve serijske impulse clocka kao i START i STOP stanja. Transfer podataka je završen sa STOP stanjem (bitom) ili sa ponovljenim START bitom. Kako ponovljeni START bit predstavlja početak sledećeg podatka (serijskog transfera), I2C- magistrala neće biti oslobođena.



Slika 5: Blok šema I2C magistrale

2.7. Serijski periferni interfejs (SPI)

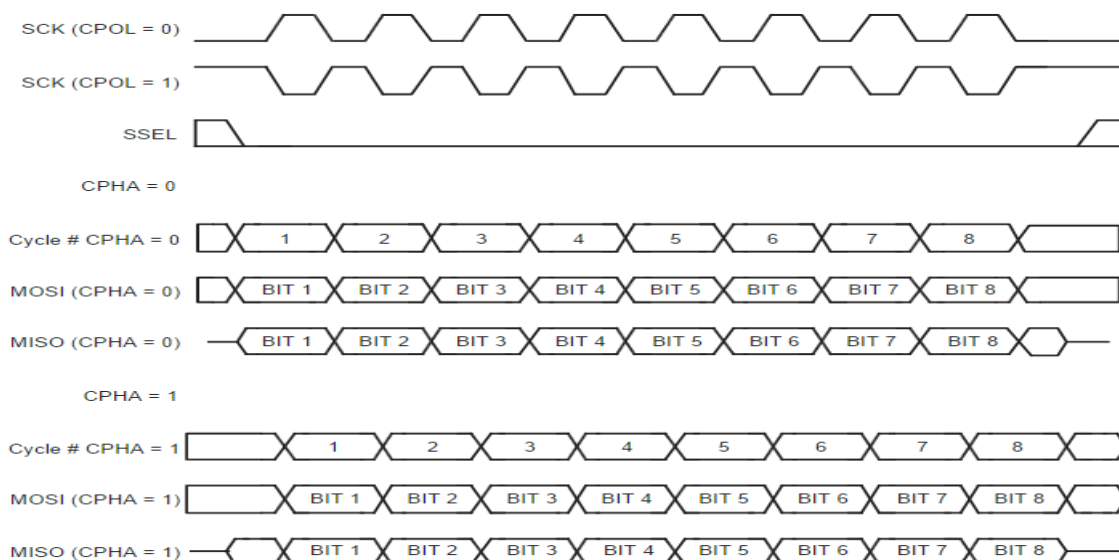
Osnovne karakteristike:

- Dva kompletna i nezavisna SPI kontrolera
- Sinhronizovana, serijska i potpuna dvostruka komunikacija
- Kombinovan master i slave Serijski Periferni Interfejs (SPI)
- Maksimalna brzina protoka podataka
- Samo 8 bitova ili od 8 do 16 bitova po transferu

SPI transfer podataka:

Slika 6 predstavlja dijagram koji pokazuje četiri različita formata transferovanja podataka koja su omogućena korišćenjem SPI. Dijagram pokazuje jedan 8-bitni transfer podataka. Prvo što treba primetiti na dijagrafu je to da je podeljen na tri horizontalna podeoka. Prvi deo opisuje SCK (serijski clock) i SSEL (Slave Select za SPI) signale. Drugi deo pokazuje MOSI (Master out Slave in) i MISO (Master in Slave out) signale kada promenljiva CPHA (Clock out Phase) ima vrednosti logicke nule. Treći deo dijagrama pokazuje MOSI i MISO signale kada CPHA ima vrednost jedinice.

U prvom delu dijagrama treba primetiti dve stvari. Prva, SPI je prikazan tako da CPOL ima vrednost i nule i jedinice. Druga, je aktiviranje i deaktiviranje SSEL signala. Kada je CPHA = 0, SSEL signal će uvek biti neaktivan prilikom transfera. To ne može da se kaže i za slučaj kada je CPHA = 1. (signal može da ostane u aktivnom stanju). Prilikom transfera podataka, master uvek šalje bajt podataka ka slave- u, i slave uvek šalje bajt podataka ka master- u.



Slika 6: Dijagram SPI transfera podataka

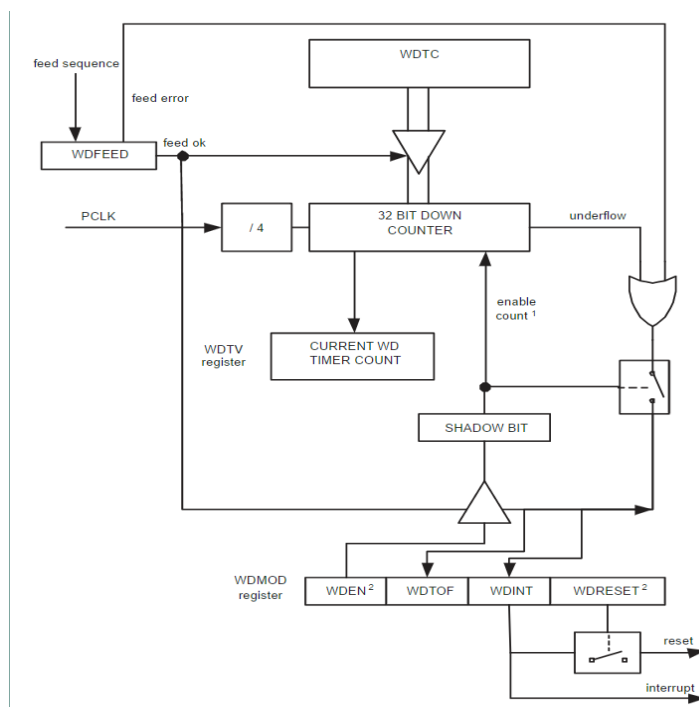
2.8. WatchDog tajmer (WDT)

Svrha Watchdog tajmera (Slika 7) je da resetuje mikrokontroler u razumnom vremenskom periodu ako uđe u pogrešno stanje. Kada je uključen, Watchdog će generisati resetovanje sistema ako korisnički program padne na 'feed' (ili reload) Watchdog u okviru unapred određenog vremena.

WDT se sastoji od 32-bitnog brojača. Clock je povezan sa tajmerom preko pre-scaler-a. Vreme se dekrementuje prilikom svakom otkucaja clocka. Minimalna vrednost sa koje se clock dekrementuje je 0xFF. Ukoliko se postavi da početna vrednost bude manja od 0xFF, tada se vrednost 0xFF učitava u brojač. WatchDog tajmer treba da se koristi na sledeći način:

- Treba postaviti tajmer WDT-a na konstantu vrednost u WDTC registru.
- Postaviti WDMOD registar.
- Pokrenuti WDT upisivanjem 0xAA, a odmah zatim 0x55 na WDTFEED registru.
- WatchDog bi trebalo ponovo biti podešen pre nego što brojač tajmera dodje do kraja, kako bi sprečila pojava neželjenih prekida ili resetovanja.

Kada vrednost brojača padne na minimum, brojač će krenuti da broji od 0x0000 0000 kao u slučaju resetovanja. WatchDog Timer-Out Flag (WDTOF) može da se postavi, kako bi proveravao da li došlo do resetovanja unutar samog WDT-a. WDTOF mora biti očišćen putem softvera.



Slika 7: Blok šema WDT

2.9. Kovertor analognog signala u digitalni (ADC)

Karakteristike ADC-a:

- 10- bitni ADC konvertor
- Moguće multipleksiranje ulaznih signala između 4 i 8 pinova
- Power-down mod
- Opseg merenja od V do Vdda
- 10- bitna konverzija $\geq 2,44$ mikro-sekunde (400.000 koverzija po sekundi)
- Mod brze konverzije za jedan ili više ulaza.
- Opcionalna konverzija na ulaznom pinu ili na Timer Match signalu

Osnovni signal za ADC konvertor je omogućen putem APB clocka. Programabilni delilac frekvencije clocka je uključen u svakoj konverziji kako bi ovaj clock sveo na rad od 4.5 MHz (maksimum). Za najpreciznije konverzije signala neophodno je koristiti 11 ovakih clockova istovremeno.

2.10. Clock u realnom vremenu (RTC – Real-time clock)

RTC je dizajniran da obezbedi skup brojača za merenje vremena kada je izabran normal-an ili idle (neaktivan) način rada. RTC je male potrošnje, što ga čini pogodnim za sisteme koji koriste baterije kada CPU ne radi konstantno (neaktivan mod).

2.11. Tajmeri opšte namene

Tajmer/Brojač je dizajniran da broji cikluse perifernog clocka (PCLK) ili clocka obezbeđenog spolja i opciono generiše prekide ili izvodi druge radnje u određenim vrednostima tajmera, u zavisnosti od četiri match registra. Takođe sadrži četiri ulaza za hvatanje kako bi prepoznao vrednost tajmera kada se menja ulazni signal i opciono generise prekid. Više pinova se može odabrati da izvode jedno hvatanje ili funkciju uparivanja, obezbeđujući aplikaciju sa 'ili' i 'i', kao i funkcije 'emitovanja' među njima.

2.12. Kontroler Flash memorije

Karakteristike:

- ISP (in-system programming) - predstavlja mogućnost programiranja i reprogramiranja čipa flash memorije koji se nalazi na mikrokontroleru korišćenjem boot loader softvera i serijskog porta. Tako nešto se može postići kada se promeni položaj na ploči.
- IAP (in-application programming) - predstavlja mogućnost korišćenja operacija brisanja i ispisivanja na čipu flash memorije.

Flash modul (Slika 8) omogućava 8 Mbit flash memorije koju kontroler može da koristi putem serijskog perifernog interfejsa (SPI).



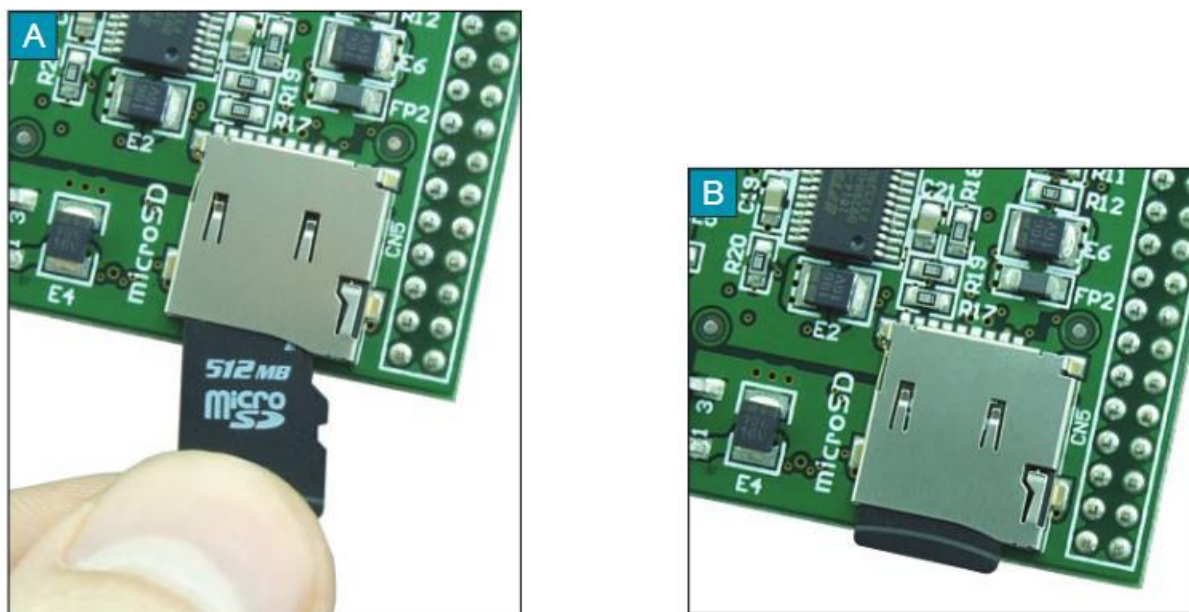
Slika 8: Flash modul

Flash boot loader kod se izvršava svaki put kada je uključen ili prilikom resetovanja. Loader može da izvrši ISP komandu ili aplikacioni kod korisnika. Niska vrednost (low) nakon reseta na pinu P0.14 znači da je poslat zahtev od strane eksternog hardvera da se pokrene ISP komanda. Pod pretpostavkom da se signal nalazi na XTAL 1 pinu, kada se generiše uzlazna ivica reset signala, treba da prođe do 3ms pre nego što je P0.14 pin preuzeo taj signal. Ako je P0.14 prihvatio nizak signal i ako je WDTOF (flag prekoračenja watchdog tajmera) aktiviran, zahtev eksternog hardvera za pokretanje ISP komande će biti ignorisan.

Pin P0.14, koji se koristi kao zahtev hardvera za pokretanje ISP komande zahteva posebnu pažnju. Kako je P0.14 pin u stanju visoke impedanse nakon resetovanja, neophodno je da korisnik omogući eksterni hardver kako bi se pin stavio u drugačije stanje. U suprotnom neočekivano pokretanje ISP komande može da se dogodi.

2.13. MicroSD konektor

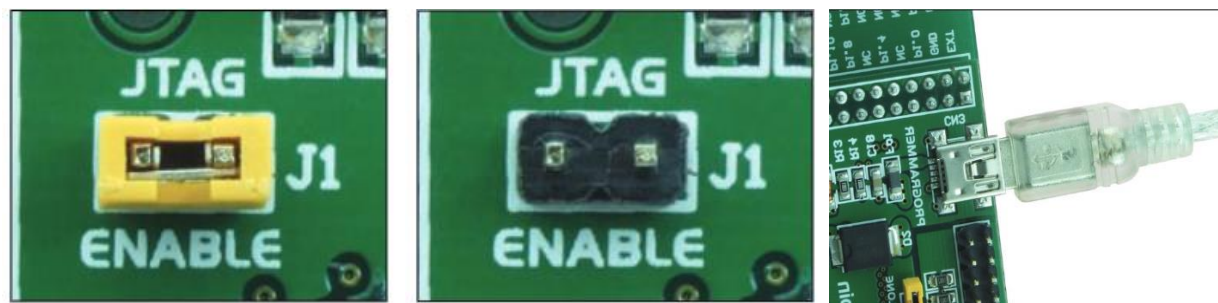
Na kontroleru se takođe nalazi i konektor CN5 (Slika 9) koji omogućava korišćenje microSD kartice. Kada je ubačena, microSD kartica omogućava dodatni memorijski prostor na kome mogu da se skladište podaci. Komunikacija između microSD kartice i mikrokontrolera se vrši preko SPI - serijskog perifernog interfejsa.



Slika 9: Slot za microSD

2.14. Programiranje mikrokontrolera

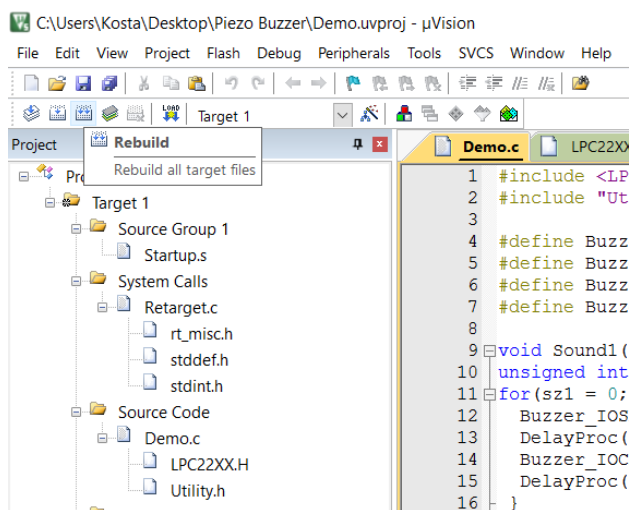
Mikrokontroler može da se programira na dva načina. Ili preko boot loadera, ili preko JTAG-a. Ukoliko se programira preko boot loadera, neophodno je povezati mikrokontroler sa PC računarom putem USB porta i to preko CN3 konektora (Slika). Ukoliko se programiranje vrši preko JTAG-a neophodno je jumper J1 skloniti sa kontrolera (Slika).



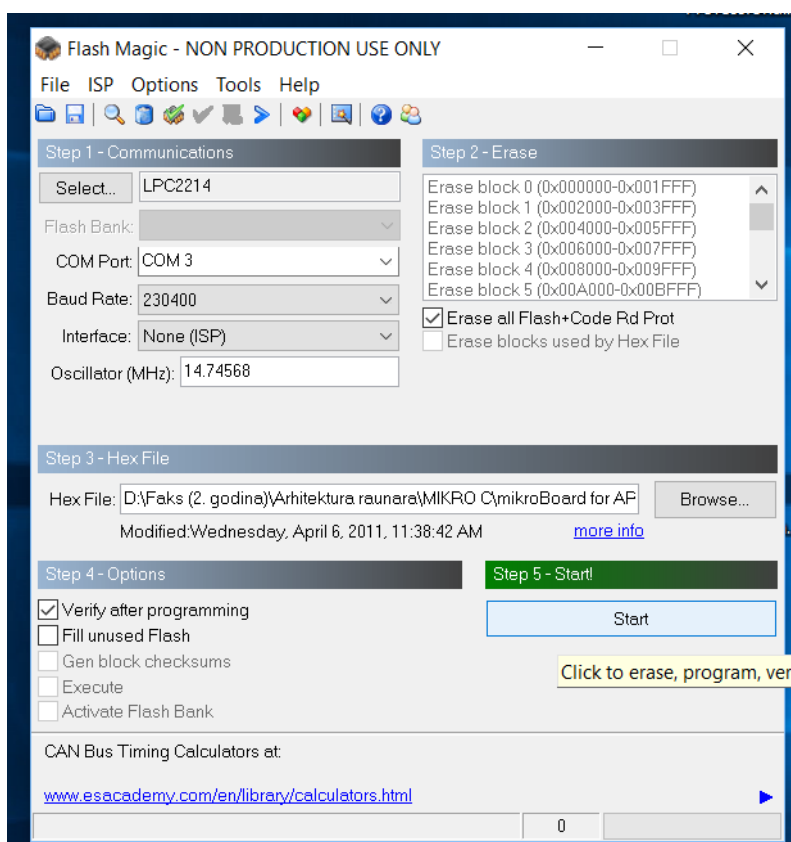
Slika 10: Jumper J1 na i van ploče/USB konektor

Ukoliko se mikrokontroler programira preko boot loadera, neophodno je koistiti dva softverska alata, Keil uVision 5 i FlashMagic.

U Keil uVision 5 programu se ispisuje odgovarajući kod u c/c++-u, a nakon toga se kompajlira i time se dobija odgovarajući .hex fajl.



Zatim se u FlashMagic-u taj .hex fajl programira na sam mikrokontroler.



2.15. Kontroler prekida (interrupt)

Vektorski kontroler prekida (VIC) prihvata sve zahteve prekida i kategorizuje ih kao VIC, vektorske IRQ. I ne-vektorske IRQ definisane prema programskim podešavanjima. Programirana/zadata šema znaci da se prioriteti prekida sa razlicitih mesta mogu dinamički zadati i odrediti.

Brzi zahtev prekida (FIQ- Fast Interrupt reQuest) ima najviši prioritet. Ako je više od jednog zahteva dodeljeno FIQ, VIC kombinuje zahteve kako bi prosledio FIQ ARM procesoru. Najbrži mogući FIQ odziv je postignut samo kada je jedan zahtev prekidan označen kao FIQ, jer tada FIQ servis rutina može jednostavno da počne da rešava zadatak na samom uređaju. Ali ako se više od jednog zahteva dodeli FIQ klasi FIQ šema može da pročita iz VIC-a koji će da identifikuje koji FIQ traži prekid.

Vektorski IRQ ima srednji prioritet. Šesnaest presretnutih signala mogu imati tu oznaku tj biti dodeljeni ovoj kategoriji. Bilo koji od signala prekida mogu biti dodeljeni bilo kojoj od 16 vektorskih IRQ slotova, od kojih priključak 0 ima najviši prioritet a priključak 15 najniži.

Ne-vektorski IRQ imaju najniži prioritet.

VIC kombinuje zahteve od svih vektorskih I nevektorskih IRQ-ova kako bi reprodukovao IRQ signal ARM procesoru. IRQ rutina može početi čitanjem registra VIC-a i prelaza na to mesto. Ako su bilo koji vektorski IRQ traženi, VIC obezbeđuje adresu najviše tražene IRQ rutine, u suprotnom obezbeđuje adresu normalne rutine koja se deli među svim ne vektorskim IRQ-ovima. Standardna rutina može biti pročitana od strane drugog VIC registra da bi se videlo koji IRQ- ovi su aktivni.

2.16. UART

LPC 2214 mikrokontroler sadrži ukupno dva UART- a. Jedan UART omogućava potpunu kontrolu interfejsa, dok drugi omogućava samo emitovanje i primanje linija podataka.

3. PROJEKTNİ ZADATAK

Ideja projektnog zadatka je realizacija alarmnog sistema, koji se aktivira usled prekoračenja određene granične vrednosti temperature. U osnovi alarmnog sistema se nalazi piezo buzzer koji treba da bude podešen tako da može da proizvede dva različita tona, jedan vrlo kratak, a drugi znatno duži i uznemirujući ton. Alarm bi mogao da se poveže sa senzerom temperature i da na određeno vreme očitava vrednosti temperature. Ukoliko bi temperatura bila manja od određene granice, sistem bi proizveo kratak zvuk, što bi značilo da je očitavanje temperature sa senzora uspešno izvršeno i da je vrednost temperature u okviru dozvoljenih graničnih vrednosti. Ukoliko bi prilikom čitanja vrednosti, temperatura bila veća od postavljenih granica, sistem bi trebalo da proizvede dugačak i uznemirujući zvuk, što bi značilo da temperatura mora da se reguliše.

4. REALIZACIJA PROJEKTOG ZADATKA

Projekat je realizovan programiranjem mikrokontrolera LPC 2214 koji je postavljen na razvojnu ploču UNI DS-6, na kojoj se nalazi piezo buzzer. Programiranje kontrolera je izvršeno preko softverskog alata Keil uvison 5, u kome je napisan odgovarajući programski kod i programa FlashMagic koji je zadužen za programiranje samog kontrolera.

Na početku samog koda je potrebno definisati koji se registri koriste i povezati ih na pin na kome se nalazi buzzer.

```
#include <LPC22XX.H>
#include "Utility.h"

#define Buzzer_Pin 4
#define Buzzer_IOSET IOSET0
#define Buzzer_IOCLR IOCLR0
#define Buzzer_IODIR IODIR0
```

Sledi definisanje prvog tona

```
void Sound1(){
    unsigned int sz1;
    for(sz1 = 0; sz1 < 100; sz1++){
        Buzzer_IOSET = (unsigned long)(1) << Buzzer_Pin; //generisanje visoke vrednosti (high)
        DelayProc(0.00001 * CCLOCK);
        Buzzer_IOCLR = (unsigned long)(1) << Buzzer_Pin; //generisanje niske vrednosti (low) i ciscenje IOSET registra
        DelayProc(0.00001 * CCLOCK);
    }
}
```

i drugog.

```
void Sound2() {  
    unsigned int sz2;  
    for(sz2 = 0; sz2 < 100; sz2++){  
        Buzzer_IOSET = (unsigned long) (1) << Buzzer_Pin;  
        DelayProc(0.00001 * CCLOCK);  
        Buzzer_IOCLR = (unsigned long) (1) << Buzzer_Pin;  
        DelayProc(0.0009 * CCLOCK);  
    }  
}
```

Registar IOSET0 je zadužen za postavljanje stanja visokog napona (high) na PORTU 0, kada ima vrednost logičke jedinice.

Registar IOCLR0 je zadužen za postavljanje stanja niskog napona (low) na PORTU 0, kada ima vrednost logičke jedinice, i za čišćenje bitova koji se nalaze na registru IOSET0.

Procedura DelayProc je ugrađena procedura koja se koristi za definisanje vremenskog razmaka (kašnjenja) između emitovanja dva tona, kao i za definisanje frekvencija rada buzzera (generisanje tona), uz frekvenciju CCLOCK-a od 60 MHz.

Za simulaciju vrednosti senzora za merenje toplote se koristi niz sa zadatim vrednostima koje treba da se pročitaju.

```
unsigned int sz;  
unsigned int sz1;  
unsigned int sz2;  
unsigned int i;  
char a[8] = {2,1,5,1,1,7,2,3};
```

Kako bi registar koji je zadužen za definisanje pravca nekog porta mogao da se koristi, potrebno je prvo definisati funkciju tog porta, ukoliko funkcija nije definisana tada nije moguće zadati pravac pinovima.

```
PINSEL0 = 0;  
Buzzer_IODIR |= (unsigned long) (1) << Buzzer_Pin;  
DelayProc(0.3 * CCLOCK);
```

Registar PINSEL0 kada mu je dodeljena vrednost 0, označava da je za funkciju pina izabrana uobičajena funkcija. IODIR0 u stanju logičke jedinice označava da je prvi pin nultog porta označen kao izlazni (pin na kome se nalazi buzzer).

Na kraju se za određene vrednosti temperature, emituju odgovarajući tonovi. U ovom slučaju, ako je temperatura manja od 5, emitovaće se ton 1 (kratak ton), u suprotnom će se emitovati ton 2 (označava da je vrednost temperature iznad dozvoljenih granica).

19

```
for(i = 0;i<8;i++){  
  
    if(a[i] < 5){  
        for(sz1 = 0; sz1 < 10; sz1++){  
            Sound1();  
  
        }  
        sz1 = 0;  
        DelayProc(0.3 * CCLOCK);  
    }  
    else{  
        for(sz2 = 0; sz2 < 10; sz2++){  
  
            Sound2();  
  
        }  
        sz2 = 0;  
        DelayProc(0.3 * CCLOCK);  
    }  
}
```

5. ZAKLJUČAK

Projekat koji je prikazan u ovom radu ima široku primenu. Ne samo da može da se koristi kao alarmni sistem gde je potrebno signalizirati kada je prekoračena temperatura, već i u bilo kom slučaju u kome se koriste senzori za bilo kakva merenja gde je potrebno znati tačnu vrednost merenja neke veličine unutar određenih granica, i signalizirati kada se vrednost nalazi van njih. (alarm u slučaju požara, u meteorologiji prilikom merenja brzine vetra, prilikom merenja koncentracije različitih supstanci u određenom jedinjenju itd..)

6. LITERATURA

- http://microcontrollershop.com/product_info.php?products_id=4128
- <https://en.wikipedia.org/wiki/Piezoelectricity>
- http://www.nxp.com/documents/user_manual/UM10114.pdf
- <https://download.mikroe.com/documents/full-featured-boards/easy/easyarm-v6/mikroboard-arm144-manual-v100.pdf>