

Univerzitet u Kragujevcu
Fakultet inženjerskih nauka



Seminarski rad iz predmeta Arhitektura računarskih sistema

Tema:
Mikrokontroler PIC18F4520

Student:
Sofija Anđelović 565/2015

Predmetni profesor:
Aleksandar Peulić

Kragujevac 2017.

Sadržaj:

1. UVOD	2
2. ARHITEKTURA	3
2.1. MikroBoard za PIC-40 pin	3
2.2. Opšte karakteristike	3
2.3. Oscilator	5
2.4. Režim napajanja	6
2.5. Resetovanje(Reset)	6
2.6. Memorija	6
2.7. 8x8 hardverski množać	8
2.8. Prekidi(Interrupts)	8
2.9. Ulazno/Izlazni portovi (I/O)	8
2.10. CCP registri	11
2.11. Master sinhroni serijski port	11
2.12. A/D konverotor	11
2.13. CPU	12
3. PROJEKTNII ZADATAK	13
4. REALIZACIJA PROJEKTOG ZADATKA	15
5. ZAKLJUČAK	15
6.LITERATURA	16
7. PRILOG KODOVI	17

1. UVOD

Mikrokontroler je jedno od najvećih tehničkih dostignuća koje je obeležilo dvadeseti vek. Sa napretkom tehnologije rastao je i stepen integracije integrisanih kola, što je dovelo do pojave prvih mikrokontrolera. Prvenstveno zbog svoje cene, mikrokontroleri nalaze sve veću primenu u svim oblastima izrade softverskih kontrolisanih uređaja i sistema. Druga velika prednost mikrokontrolera, zbog koje su danas neizbežni, jeste činjenica da se mogu programirati u višim programskim jezicima, kao što su tipa C, Paskal... Ovo omogućava njihovo programiranje, pa čak i bez znanja asemblerskog jezika.

Mikrokontroler je mali računar, realizovan u obliku integrisanog kola koji objedinjuje sve potrebne komponente, kako bi mogao samostalno da funkcioniše. Tu spadaju integrisani mikroprocesor, memorija, digitalni i analogni ulazi/izlazi, tajmeri, brojači, oscilatori i drugi sklopovi (u zavisnosti od vrste i namene mikrokontrolera) koji su ranije bili svaki u svom integrisanom čipu. Mikrokontroler se programira da normalno radi u beskonačnoj petlji i za to vreme očitava ulaze i podešava izlaze u zavisnosti od programa koji mu je zadat.

PIC mikrokontroler (Peripheral Interface Controller) je originalan proizvod kompanije Microchip Technology, vrlo je popularan među inženjerima i elektronskim početnicima, a dolazi u puno različitih verzija, svaka sa posebnim komponentama i mogućnostima. Mnogi elektronski projekti mogu se vrlo lako izvesti pomoću mikrokontrolera iz PIC porodice, poput digitalnih satova, vrlo jednostavnih video igrica, robota... Njihova primena je vrlo široka, a njihove mogućnosti i varijante brojne, a mogu se nabaviti po vrlo pristupačnim cenama što ih čini još popularnijima. Postoji više vrsta 8-bitnog PIC mikrokontrolera: 10xxx, 12xxx, 14xxx, 16xxx, 17xxx, 18xxx. U ovom projektu je reč o PIC18F4520.

2. ARHITEKTURA

2.1. MikroBoard za PIC-40 pin

MikroBoard je minijaturni razvojni alat koji uključuje mikrokontroler, programer, mali protoboard i dva 2x40 konektora za povezivanje sa UNI-DS6 razvojnim sistemom. Glavna svrha MikroBoarda je da poveže mikrokontroler sa UN-DS6 razvojnim sistemom. Integrirani deo MikroBoard-a je programer koji se koristi za programiranje mikrokontrolera. Da bi se ova ploča povezala sa PC računarom i uopšte dobila napajanje koristi se USB napajanje. USB napajanje može biti povezano putem CN3 priključka, kao i putem 2x40 konektora. Za napajanje putem 2x40 konektora, neophodan je regularni napon od 5V. Za programiranje je neophodno instalirati adekvatan softver, u ovom slučaju to je mikroProg Suite for PIC.



2.2. Opšte karakteristike

Cela PIC18 porodica zadržava neke opšte karakteristike PIC kontrolera uz dodatna poboljšanja. Odlikuje ih visoke računarske performanse, visoka izdržljivost, kao i niska cena. U ovoj porodici je poboljšan sam dizajn, tako da PIC18 postaje logičan izbor za mnoge aplikacije visokih performansi. Jedna od prednosti ove porodice je što ima niz dodatnih funkcija koju su smanjili značajno potrošnju energije tokom rada.

CPU

- C kompajler optimizovan RISK arhitekturom
- Do 10 MIPS operacija kada ga priključimo za napajanje od 3V
- 8x8 jednociklusni hardverski množač

Sistem

- Interni oscilator podržava frekvencije od 31kHz do 8MHz sa 4xPLL(Phase Lock Loop)
- Fail-Safe Clock Monitor - omogućava bezbedno isključivanje ako klok zakaže
- Watchdog Timer sa odvojenim RC oscilatorom
- Širok opseg napona 2.0V do 5.5V

Funkcije za upravljanje sa vrlo malo energije

- Run, Idle i SLEEP režimi
- Režim spavanja: 5.8uA
- Idle režim: 0.1uA

Analogne karakteristike

- 10-bitni ADC(Analogno-digitalni konverotr), 13 kanala, 100k uzorka po sekundi
- Modul za detekciju niskog napona koji se može programirati
- Brown-out-Reset modul koji se može programirati
- Dva analogna komparatora

Periferijalne karakteristike

- Glavni sinhroni serijski port (MSSP) koji podržava SPI i I2C master i slave režim
- Četiri Timer modula
- EUSART modul
- Do 5 PWM izlaza

Parameter Name	Value
Program Memory Type	Flash
Program Memory (KB)	32
CPU Speed (MIPS)	10
RAM Bytes	1,536
Data EEPROM (bytes)	256
Digital Communication Peripherals	1-UART, 1-SPI, 1-I2C1 - MSSP(SPI/I2C)
Capture/Compare/PWM Peripherals	1 CCP, 1 ECCP
Timers	1 x 8-bit, 3 x 16-bit
ADC	13 ch, 10-bit
Comparators	2
Temperature Range (C)	-40 to 125
Operating Voltage Range (V)	2 to 5.5
Pin Count	40

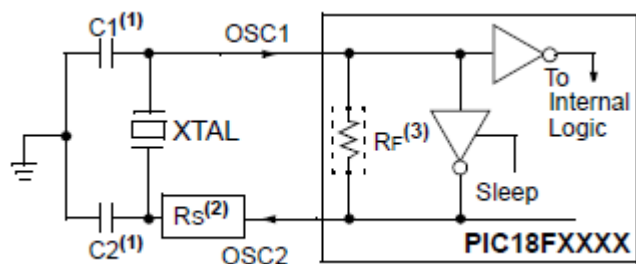
2.3. Oscillator

Uređaj PIC18F4520 može koristiti 10 različitih modova oscilatora:

1. LP *Low-Power* kristal
2. XT kristal/rezonator
3. HS Kristal visoke brzine/rezonator
4. HSPLL Kristal visoke brzine /rezonator sa odobrenim PLL
5. RC eksterni rezonator/kondenzator sa FOSC/4 izlazom na RA6
6. RCIO eksterni rezonator/kondenzator sa I/O na RA6
7. INTIO1 interni oscilator sa FOSC/4 izlazom na RA6 i I/O na RA7
8. INTIO2 interni oscilator sa I/O na RA6 i RA7
9. EC eksterni klok sa FOSC/4 izlazom
10. ECIO eksterni klok sa I/O na RA6

Kristalni oscilator / keramički resonator

U LP, XT, HS ili HSPLL režimima oscilatora, keramički resonator je konektovan na pinove OSC1 i OSC2 da bi se uspostavila oscilacija. Dizajn oscilatora zahteva paralelni rez kristala.



Eksterni klok

EC i ECIO oscilatorski režim zahteva eksterni clock kao izvor koji je povezan sa OSC1 pinom. Ne zahteva početno vreme prilikom paljenja, resetovanja ili izlaska iz Sleep mode. U EC oscilatorskom režimu, frekvencija oscilatora podeljena sa 4 dostupna je na OSC2 pinu.

RC oscilator

Za aplikacije koje služe za merenje vremena, RC i RCIO oscilatorski režimi nude veliku uštedu. Stvarna oscilatorska frekvencija je u funkciji nekoliko faktora, kao što su mrežni napon, vrednosti eksternog otpornika i kondenzatora i temperature operacije.

PLL množač frekvencija i HSPLL oscilatorski režim

PLL kolo sadrži opcije za korisnike koji žele da koriste niže frekvencije oscilatorskog kola ili clock uređaj kao njegovu najvišu frekvenciju kristalnog oscilatora. HSPLL režim koristi režim rada HS oscilatora do frekvencije od 10MHz.

PIC18 uređaji interni blok oscilatora koji generišu dva clock signala. Oba mogu biti korišćena kao clock mikrokontrolera. To može da eliminiše potrebu za kola eksternog oscilatora na pinovima OSC1 i OSC2. Što se tiče izvornog clocka i promene oscilatora veoma bitna karakteristika je da može da prebaci iz stanja sa glavnim oscilatorom u stanje clocka alternativne niske frekvencije. U osnovi, postoje tri vrste izvornog clocka za ove uređaje, a to su primarni oscilator, sekundarni oscilator i unitrašnji blok oscilator.

2.4. Režim napajanja

Uređaji PIC18 porodice imaju niz od sedam opcija koji omogućavaju efikasnije korišćenje energije. Ova opcija je veoma bitna kod aplikacija koji se napajaju sa ograničenim vidom energijem, tipa kao što je baterija. Postoje tri kategorije režima upravljanja:

1. Režim rada (Run mode) – rade i periferije i tajmer i jezgro
2. Režim mirovanja (Idle mode) – procesor kontrolera se zaustavi, ali periferije nastavljaju da rade
3. Režim spavanja (Sleep mode) - vrlo smanjena potrošnja energija

Iz sleep mode-a se izlazi na 3 načina: prekidom (Interrupt), resetovanjem (Reset) i Watchdog Time-out

2.5. Resetovanje (Reset)

PIC 18F4520 ima više različitih vrsta resetovanja:

- Power-on Reset – uključeno resetovanje
- MCLR Reset – resetovanje u toku normalnih operacija
- MCLR Reset – resetovanje tokom napajanja
- WatchDog Reset+ - tokom izvršenja
- Brown-out Reset koji može da se programira
- Reset instrukcija
- Stack Full Reset
- Stack Underflow Rest

2.6. Memorija

Postoje tri vrste memorije u mikrokontroleru PIC18, a to su :

- ROM
- RAM
- EEPROM

Uloga **ROM**-a je da trajno čuva program koji se izvršava, zato se često naziva i “programska memorija”, a pošto je izrađena u FLASH tehnologiji, njen sadržaj može da se promeni dovođenjem posebnog napona za programiranje.

RAM (Random Access Memory) dozvoljava da se podaci uzimaju direktno u nasumičnom redosledu.

Slično kao kod programske memorije, sadržaj **EEPROM**-a je trajno sačuvan i ne gubi se prestankom napajanja. Međutim, za razliku od ROM-a, sadržaj **EEPROM**-a može i da se menja u toku rada mikrokontrolera. Zbog toga je ova memorija (256 lokacija) idealna za trajno čuvanje vrednosti do kojih se dolazi u toku rada.

PIC18 mikrokontroleri implementiraju 21-o bitni **programski brojač**, koji je u stanju da adresira 2-megabajtni programski memorijski prostor. Sadrži 32Kbytes fleš memorije i mogu da skladište do 16,384 jednorecnih instrukcija.

Programski brojač se sastoji od tri odvojena 8-bitna registra. Niži bajt, poznat kao PCL registar, sa njega mogu i da se čitaju podaci, ali i da se upisuju. Viši bajt, iliti PCH registar [15:8], sa njega direktno ne mogu ni da se upisuju ni da se čitaju podaci, ali unapređen je pomoću PCLATH registra. Gornji bajt, PCU registar [20:16], sa njega takođe ne mogu direktno da se čitaju podaci, kao ni da se upisuju, ali je unapređen pomoću PCLATU registra.

Stek funkcioniše kao 31-word od 21 bit RAM-a i 5-bitni Steck Pointer. Prostor steka nije deo nijednog programa ili prostora podataka. Steck Pointer služi za čitanje i za pisanje.

Organizacija memorijskih podataka

Memorijski podaci u PIC18 uređajima su implementirani kao statički RAM. Svaki registar u memoriji ima 12-bitnu adresu, dozvoljavajući 4096 bajtova memorijskih podataka. Memorijski proctor je podeljen u 16 banka od po 256 bajtova. Memorijska podataka sadrži dva registra:

- SFRs (Special Function Registers) - koristi se za kontrolu, status kontrolera i perifernu funkciju
- GRPs (General Purpose Registers) - koristi se za skladištenje podataka

Instrukcije i arhitektura dozvoljavaju operacije nad svim bankama. Cela memorija može biti adresirana direktnim, indirektnim i indeksnim adresiranjem.

Podaci adresiranja

Informacije u memorijskom prostoru podataka mogu se adresirati na više različitih način. Za najveći broj instrukcija, adresni režim je fiksiran. Ostale instrukcije mogu da koriste do tri režima, u zavisnosti koji operand je korišćen. Načini adresiranja su: neprekidno, doslovno, direktno i indirektno.

Flash programska memorija

Flash programska memorija ima mogućnost čitanja, pisanja i brisanja u toku izvršavanja normalnog rada u VDD opsega. Što se tiče čitanja programske memorije, ono se izvršava jedan po jedan bajt. Pisanje u programsku memoriju se izvršava u blokovima od 32 bajta, dok se brisanje iz programske memorije izvršava u blokovima od 64 bajta. Kada dođe do pisanja ili brisanja podataka, sve ostale instrukcije se obustavljaju dok se pomenute operacije ne izvrše. Prilikom ovih operacija se ne može pristupiti memoriji i dok traju, kod se neće izvršiti.

EEPROM memorija

Podaci EEPROM-a su trajni memorijski nizovi, odvojeni od ROM-a i RAM-a, koji se koriste za dugoročno skladištenje podataka programa. Ova memorija nije direktno mapirana ni u jedan registar ili memorijski proctor, ali je indirektno adresirana kroz SFRs. EEPROM ima mogućnost i čitanja sa nje, kao i mogućnost pisanja u nju.

2.7. 8x8 hardverski množač

Svi PIC uređaji iz porodice 18 sadrže 8x8 hardverski množač kao deo Aritmetičko logičke jedinice(ALU). Množač obavlja neoznačenu operaciju i dobija kao rezultat 16bit-ni par koji smešta u registar proizvoda, PRODH:PRODL. Operacija množača ne utiče na bilo koji „fleg“ u statusnom registru.

2.8. Prekidi(Interrupts)

PIC18F4520 ima više izvora prekida, i oni imaju mogućnost da budu na nivou višeg i na nivou nižeg prioriteta, da bi sam mikrokontroler znao koju opciju pre da izvršava i koji je prekid bitniji. Vektor visokog prioriteta prekida je na 0008h, dok je vektor niskog prioriteta prekida na 0018h. Postoji 10 registra koji se koriste za kontrolu prekida, a to su:

- RCON
- INTCON
- INTCON2
- INTCON3
- PIR1, PIR2
- PIE1, PIE2
- IPR1, IPR2

Generalno, prekidi imaju tri bita koji kontrolišu njihove operacije:

- Flag bit – pokazuje da se prekid dogodio
- Enable bit – dozvoljava izvršavanje programa u vektorskoj adresi prekidačkog stable kada je flag bit podešen
- Priority bit – selektuje najviši i najniži prioritet

2.9. Ulazno/Izlazni portovi (I/O)

U zavisnosti koji uređaj izaberemo i koja je funkcija dozvoljena možemo imati na raspolaganju do pet portova. Kada se neki od pinova koristi za tipa funkcije periferija, ne može da se koristi i u opšte svrhe. Svaki port ima tri registra za ove operacije, a ti registry su:

- TRIS registar (upravljački registar podataka)
- PORT registar (za čitanje nivoa pinova uređaja)
- LAT registar (koristi se za read-modify-write operacije na vrednostima koje donose ulazno/izlazne jedinice)
- **PORTA** je širine 8 bitova, dvosmerni port. Njegov odgovarajući upravljački registar podataka je TRISA. Ako TRISA bit postavimo na logičku jedinicu, tada je PORTA ulazni port, a kada ga postavimo na logičku nulu, onda je izlazni port. LATA registar ima memorijsko svojstvo.

TABLE 10-2: SUMMARY OF REGISTERS ASSOCIATED WITH PORTA

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on page
PORTA	RA7 ⁽¹⁾	RA6 ⁽¹⁾	RA5	RA4	RA3	RA2	RA1	RA0	52
LATA	LATA7 ⁽¹⁾	LATA6 ⁽¹⁾	PORTA Data Latch Register (Read and Write to Data Latch)						52
TRISA	TRISA7 ⁽¹⁾	TRISA6 ⁽¹⁾	PORTA Data Direction Register						52
ADCON1	—	—	VCFG1	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0	51
CMCON	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	51
CVRCON	CVREN	CVROE	CVRR	CVRSS	CVR3	CVR2	CVR1	CVR0	51

Legend: — = unimplemented, read as '0'. Shaded cells are not used by PORTA.

Note 1: RA<7:6> and their associated latch and data direction bits are enabled as I/O pins based on oscillator configuration; otherwise, they are read as '0'.

- **PORTB** je širine 8 bitova, dvosmerni port. Njegov odgovarajući upravljački registar podataka je TRISB. Ako TRISB bit postavimo na logičku jedinicu, tada je PORTB ulazni port, a kada ga postavimo na logičku nulu, onda je izlazni port. LATB registar ima memorijsko svojstvo.

TABLE 10-4: SUMMARY OF REGISTERS ASSOCIATED WITH PORTB

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on page
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	52
LATB	PORTB Data Latch Register (Read and Write to Data Latch)								52
TRISB	PORTB Data Direction Register								52
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	49
INTCON2	RBPUP	INTEDG0	INTEDG1	INTEDG2	—	TMR0IP	—	RBIP	49
INTCON3	INT2IP	INT1IP	—	INT2IE	INT1IE	—	INT2IF	INT1IF	49
ADCON1	—	—	VCFG1	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0	51

Legend: — = unimplemented, read as '0'. Shaded cells are not used by PORTB.

- **PORTC** je širine 8 bitova, dvosmerni port. Njegov odgovarajući upravljački registar podataka je TRISC. Ako TRISC bit postavimo na logičku jedinicu, tada je PORTC ulazni port, a kada ga postavimo na logičku nulu, onda je izlazni port. LATC registar ima memorijsko svojstvo.

TABLE 10-6: SUMMARY OF REGISTERS ASSOCIATED WITH PORTC

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on page
PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	52
LATC	PORTC Data Latch Register (Read and Write to Data Latch)								52
TRISC	PORTC Data Direction Register								52

- **PORTD** je širine 8 bitova, dvosmerni port. Njegov odgovarajući upravljački registar podataka je TRISD. Ako TRISD bit postavimo na logičku jedinicu, tada je PORTD ulazni port, a kada ga postavimo na logičku nulu, onda je izlazni port. LATD registar ima memorijsko svojstvo.

TABLE 10-8: SUMMARY OF REGISTERS ASSOCIATED WITH PORTD

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on page
PORTD	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	52
LATD	PORTD Data Latch Register (Read and Write to Data Latch)								52
TRISD	PORTD Data Direction Register								52
TRISE ⁽¹⁾	IBF	OBF	IBOV	PSPMODE	—	TRISE2	TRISE1	TRISE0	52
CCP1CON	P1M1 ⁽¹⁾	P1M0 ⁽¹⁾	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	51

Legend: — = unimplemented, read as '0'. Shaded cells are not used by PORTD.

Note 1: These registers and/or bits are unimplemented on 28-pin devices.

- **PORTE** je širine 4 bita, tri bita su samostalno podesiva da budu ulazni ili izlazni port. Njegov odgovarajući upravljački registar je TRISE. Ako TRISE bit postavimo na logičku jedinicu, tada je PORTE ulazni port, a kada ga postavimo na logičku nulu, onda je izlazni port. LATE registar ima memorijsko svojstvo.

REGISTER 10-1: TRISE REGISTER (40/44-PIN DEVICES ONLY)

R-0	R-0	R/W-0	R/W-0	U-0	R/W-1	R/W-1	R/W-1
IBF	OBF	IBOV	PSPMODE	—	TRISE2	TRISE1	TRISE0
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

- **Paralelni slave port** – PORTD može takođe da radi kao 8-bitni paralelni slave port ili kao mikroprocesor port. PSP operacije kontrolišu gornja 4 bita TRISE registra

TABLE 10-11: REGISTERS ASSOCIATED WITH PARALLEL SLAVE PORT

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on page
PORTD	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	52
LATD	PORTD Data Latch Register (Read and Write to Data Latch)								52
TRISD	PORTD Data Direction Register								52
PORTE	—	—	—	—	RE3	RE2	RE1	RE0	52
LATE	—	—	—	—	—	LATE Data Latch Register			52
TRISE	IBF	OBF	IBOV	PSPMODE	—	TRISE2	TRISE1	TRISE0	52
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IF	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	49
PIR1	PSPIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	52
PIE1	PSPIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	52
IPR1	PSPIP	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	52
ADCON1	—	—	VCFG1	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0	51

Legend: — = unimplemented, read as '0'. Shaded cells are not used by the Parallel Slave Port.

2.10. CCP registri

Svi PIC18 uređaji imaju dva CPP modula (Capture/Compare/PWM – Čuvanje/Poređenje/PWM). Svaki modul sadrži 16-bitni registar koji radi kao 16-bitni registar za čuvanje ili 16-bitni registar za poređenje ili 16-bitni registar za PWM (Puls-Width Modulation).

U režimu Capture (čuvanja), periferni modul omogućava vreme trajanja događaja. Režim Compare (poređenja) omogućava korisniku da pokrene spoljni događaj unapred određen vremenski period. PWM režim može da generiše impulsnoširinske modulacione signale varirajuće frekvencije i radnog ciklusa.

2.11. Master sinhroni serijski port

Master sinhroni serijski port (MSSP) modul je serijski interfejs koristan kod komunikacije sa drugim perifernim uređajima ili uređajima mikrokontrolera. Što se tiče perifernih uređaja, to mogu da budu serijski EEPROMs, pomerački registar, A/D konvertor, displej itd. MSSP može da radi u dva režima:

- Serial Peripheral Interface (SPI) – serijsko-periferni pristup
- -Integrated Circuit (I^2C) – unutrašnje integrisano kolo

I^2C pristup podržava i prateće režime hardvera, a to su Master režim, Multi-Master režim i Slave režim.

SPI pristup podržava četiri režima rada, a to su Serial Data Out (SDO) – RC5/SDO, Serial Data In (SDI) – RC4/SDI/SDA, Serial Clock (SCK) – RC3/SCK/SCL, Slave Select (SS) – RA5/SS.

2.12. A/D konvertor

Analogno-digitalni konvertor ima 13 pinova. A/D modul dozvoljava konverziju analognog ulaza u odgovarajući 10-bitni digitalni broj. Ovaj modul ima 5 registra:

- A/D Result High Register (ADRESH)
- A/D Result Low Register (ADRESL)
- A/D Control Register 0 (ADCON0)
- A/D Control Register 1 (ADCON1)
- A/D Control Register 2 (ADCON2)

2.13. CPU

PIC18F4520 sadrži nekoliko karakteristika namenjenih za maksimalno povećanje sposobnosti obavljanja funkcija i minimizacije cena pomoću eliminacije eksternih komponenata, a to su:

- Selekcija oscilatora
- Resetovanje
 - Power-on Resetovanje (POR)
 - Power-up Tajmer (PWRT)
 - Start-up Tajmer oscilator(OST)
 - Brown-out Resetovanje (BOR)
- Interapti
- WatchDog tajmer
- Fail-Safe Clock Monitor
- Two-Speed Start-up
- Zaštita koda
- ID lokacija
- Serijsko programiranje unutar kola

Oscilator se može konfigurisati za aplikacije koje zavise od frekvencije, napajanja, preciznošću i troškova. Pored Power-Up i Start-Up tajmera predviđenim za Reset, PIC18 uređaji imaju Watchdog Timer (WDT), koji je ili trajno omogućen preko konfiguracionih bitova ili kontrolisan putem softvera. Sve ove karakteristike su omogućene i podešene postavljanjem odgovarajućeg konfiguracionog registra.

3. PROJEKTNI ZADATAK

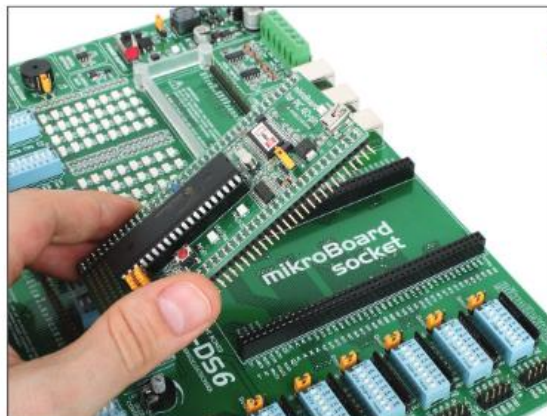
Projektni zadatak čini Mocartova melodija – Mala noćna muzika. Za realizaciju ovog projekta potrebna nam je ploča UNI-DS6, posebno njen deo Piezo Buzzer. Pa da nešto više kažemo o UNI-DS6.

UNI-DS6

UNI-DS6 razvojni sistem obezbeđuje razvojno okruženje za programiranje i eksperimentisanje sa raznovrsnim mikrokontrolerima od različitih proizvođača. Brojni moduli, LCD displej, piezo buzzer, USB-UART itd...koji čine ploču daju brojne mogućnosti koje je lako koristiti.



Ploča je dizajnirana tako da postoji mesto za mikrokontroler na njenom razvojnom sistemu. Svaka ploča ima integrisani programer koji se koristi za programiranje mikrokontrolera. Za konekciju sa razvojnim sistemom, ploča koristi dva 2x40 konektora. Iako je UNI-DS6 razvojni sistem za različite mikroBoard-e, svi oni imaju univerzalno mesto na ploči.



Da bismo programirali mikrokontroler priključimo USB kablom mikroBoard sa PC-om. Instaliramo odgovarajući softver. U ovom slučaju su to **mikroC Pro for PIC** i programator **mikroProg Suite for PIC**.

Pošto nam je za realizaciju projekta potreban Piezo buzzer, reći ćemo nešto više o njemu.

Piezo buzzer

Pomoću Piezo buzzer-a omogućeno je emitovanje audio signala. Da bi Piezo buzzer radio prikladno, potreban je napon signala određene frekvencije. Rezonantna frekvencija Piezo buzzer-a je 3.8 kHz. Druge frekvencije u rasponu između 20Hz i 20kHz takođe se mogu koristiti, ali najbolja izvedba je sa frekvencijama u rasponu između 2kHz i 4kHz. Za uspostavljanje veze između Piezo buzzer-a i mikrokontrolera, potrebno je staviti jumper J14 u odgovarajuću poziciju.



4. REALIZACIJA PROJEKTA

Projekat je realizovan tako što sam napisala kod u programskom jeziku C pomoću softvera mikroC for PIC. Kod se sastoji iz tri dela. U prvom delu su definisane funkcije svih nota koji se pojavljuju u melodiji. Svaka funkcija sadrži dva parametra, jedan je frekvencija, a drugi trajanje note. Cela melodija je urađena sa frekvencijama koje odgovaraju četvrtoj oktavi. Drugi deo ovog projekta čini funkcija melodije. Tu su u odgovarajućem redosledu pozivane funkcije odgovarajućih nota. Treći deo koda je glavna (main) funkcija. U tom delu je definisan RC6 kao input i kao odgovarajući start taster za melodiju. Isto tako u ovom delu su isključeni komparatori i A/D konvertor. Da bismo pustili melodiju, priključimo mikroBoard sa PC-om pomoću USB kabla, pomoću microProg Suite for PIC softvera programiramo ploču, pritisnemo RC6 taster na UNI-DS6 razvojnom sistemu i melodija će početi. Ovako neki projekat može se koristiti za neki vid alarma za buđenje.



5. ZAKLJUČAK

Dvadeseti vek doneo je velike inovacije na polju elektronike. Stvaranjem mikrokontrolera, kao uređaja koji u sebi ima kako mikroprocesor tako i ostale delove kao što su memorija, portovi, konvertori, komparatori itd..dosta je olakšalo projektovanje. Osim što je sve na jednom mestu, dodata je i mogućnost programiranja u višim programskim jezicima, tako da i bez poznavanja mašinskog jezika može se projektovati. U ovom projektu bila je reč o PIC18F4520 miktokontroleru. Opisana je njegova arhitektura i pokazana jedna od mnogobrojnih opcija. Najveće prednosti PIC18 mikrokontrolera su niska cena, velika izdržljivost i lako projektovanje.

6. LITERATURA

1. PIC18F2420/2520/4420/4520 Data Sheet
2. Verle, M.(2008.), *PIC mikrokontroler*, Beograd: Mikroelektronika
3. UNI-DS6 , User Manual, Mikroelektronika
4. mikroBoard for PIC 40-pin, Manual, Mikroelektronika

7. PRILOG KODOVI

```

void ToneG() {
    Sound_Play(392, 400);    // Frequency = 659Hz, duration = 250ms
}

void ToneD() {
    Sound_Play(294, 400);    // Frequency = 294Hz, duration = 400ms
}

void ToneB() {
    Sound_Play(494, 400);    // Frequency = 494Hz, duration = 400ms
}

void ToneC() {
    Sound_Play(262, 400);    // Frequency = 262Hz, duration = 400ms
}

void ToneA() {
    Sound_Play(440, 400);    // Frequency = 440Hz, duration = 400ms
}

void ToneF() {
    Sound_Play(349, 400);    // Frequency = 349Hz, duration = 400ms
}

void ToneE() {
    Sound_Play(330, 400);    // Frequency = 330Hz, duration = 400ms
}

```

```

void malanocnamuzika() {    // Mala nocna muzika - Mozart
    ToneE(); ToneG(); ToneA(); ToneE();
    ToneB(); ToneD(); ToneB(); ToneE();
    ToneF(); ToneB(); ToneE(); ToneF();
    ToneE(); ToneB(); ToneG(); ToneF();
    ToneB(); ToneF(); ToneE(); ToneB();
    ToneG(); ToneF(); ToneB(); ToneE();
    ToneG(); ToneA(); ToneE(); ToneB();
    ToneD(); ToneB(); ToneE(); ToneF();
    ToneB(); ToneE(); ToneF(); ToneG();
    ToneF(); ToneB(); ToneB(); ToneG();
    ToneF(); ToneB(); ToneE(); ToneB();
}

```

```
void main() {  
  
    ADCON1 = 0x0F;           // All AN pins as digital  
    ADON_bit = 0;            // Turn off AD convertor  
    CMCON = 0x07;            // Turn off comparators  
  
    TRISC = 0x40;             // Configure RC6 as input  
    Sound_Init(&PORTC, 1);  
  
    while (1) {  
        if (Button(&PORTC, 6, 1, 1)) // RC6 plays  
            Malanocnamuzika();  
        while (RC6_bit) ;             // Wait for button to be released  
    }  
}
```