



# Mikroračunarski sistemi

## 20ER6004

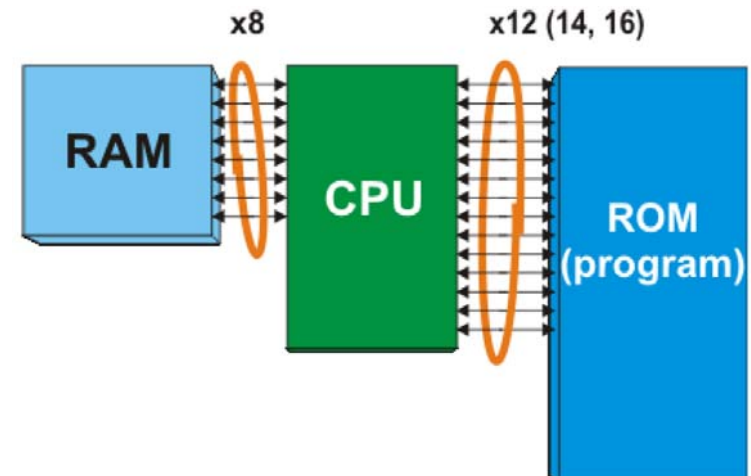
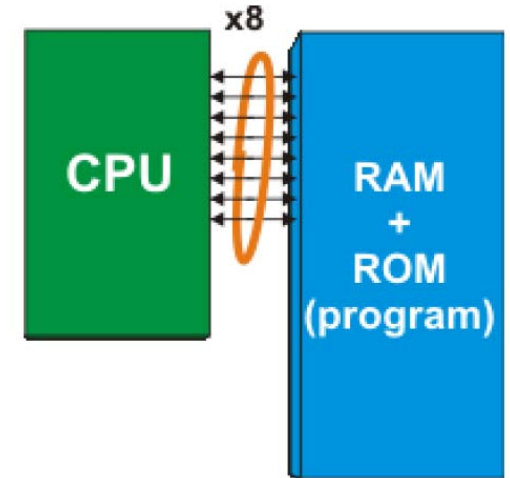


# Mikrokontroleri

- Sadrže sve bitne komponente jednog računarskog sistema u jednom čipu (procesor, memoriju, U/I portove, ...)
- I samo ime govori da su namenjeni da upravljaju (uređajima, procesima, ...)
- Zato poseduju dodatne komponente (A/D i D/A konvertore, UART, USB, ...)

# Podela prema arhitekturi

- **Von Neumann** – zajednička memorija za program i podatke, izvršenje u 2 ciklusa (jer se instrukcije i podaci prenose preko iste magistrale) *Princeton*
- **Harvard** – odvojene instrukcije i podaci, izvršenje u 1 ciklusu (jer se simultano mogu preuzimati instrukcije i podaci, uz *pipeline*)





# Podela prema skupu instrukcija

- **CISC** – (*Complex Instruction Set Computer*) smanjuje broj instrukcija po programu, ali povećava broj ciklusa po instrukciji; uglavnom zastupljena kod CPU, mada postoje i MCU sa ovom arhitekturom (npr. 68HC11 familija Motorolinih MCU)
- **RISC** – (*Reduced Instruction Set Computer*) smanjuje broj ciklusa (samo 1) po instrukciji, ali povećava program; većina MCU je sa ovom arhitekturom



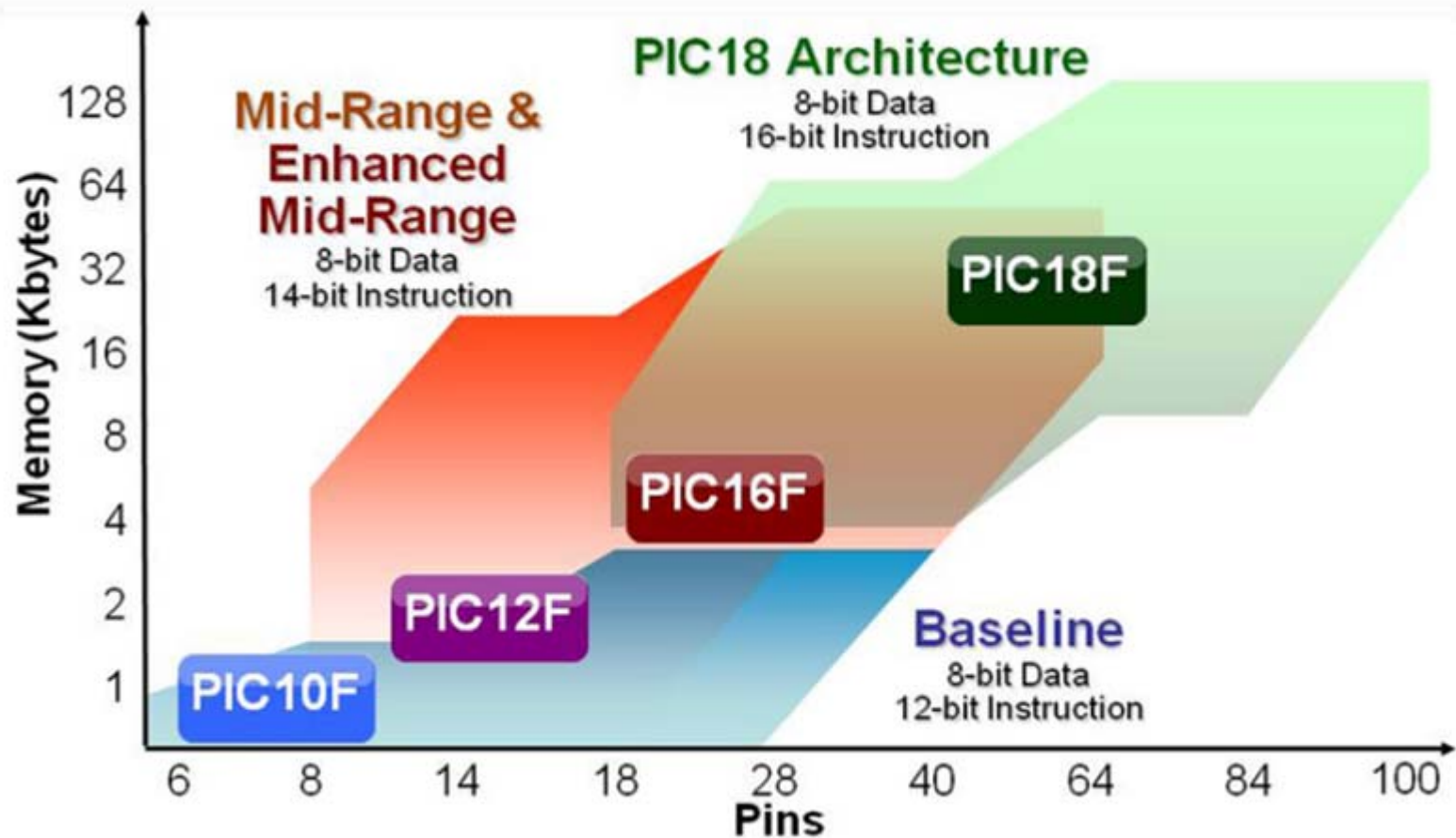


# Prema proizvođaču

- **8051** familija (**Intel**) – 8-bit MCU sa RISC Harvard arh. (od 1985.)
- **68HC11** familija (**Motorola**) – 8-bit MCU sa CISC Von Noimann arh. (od 1985.)
- **AVR** familija (**Atmel**) – 8-bit MCU sa RISC Harvard arh. (od 1996.)
- **PIC** familija (**Microchip**) – 8, 16 i 32-bit MCU sa RISC Harvard arh. (od 1976.)
- **ARM Cortex-M** familija (**Arm Holdings**) – 32-bit sa Von Neumann, Harvard, ili modifikovanom Harvard arh. (odvojene magistrale za instrukcije i podatke, ali pristupaju jedinstvenom memorijskom prostoru) (od 2004.)

# 8-bitni PIC mikrokontroleri

## PIC10, PIC12, PIC16 i PIC18



	Baseline Architecture	Mid-Range Architecture	Enhanced Mid-Range Architecture	PIC18 Architecture
Pin Count	6-40	8-64	8-64	18-100
Interrupts	No	Single interrupt capability	Single interrupt capability with hardware context save	Multiple interrupt capability with hardware context save
Performance	5 MIPS	5 MIPS	8 MIPS	Up to 16 MIPS
Instructions	33, 12-bit	35, 14-bit	49, 14-bit	83, 16-bit
Program Memory	Up to 3 KB	Up to 14 KB	Up to 28 KB	Up to 128 KB
Data Memory	Up to 138 Bytes	Up to 368 Bytes	Up to 1,5 KB	Up to 4 KB
Hardware Stack	2 level	8 level	16 level	32 level
Features	<ul style="list-style-type: none"> <li>■ Comparator</li> <li>■ 8-bit ADC</li> <li>■ Data Memory</li> <li>■ Internal Oscillator</li> </ul>	In addition to Baseline: <ul style="list-style-type: none"> <li>■ SPI/I<sup>2</sup>C™</li> <li>■ UART</li> <li>■ PWMs</li> <li>■ LCD</li> <li>■ 10-bit ADC</li> <li>■ Op Amp</li> </ul>	In addition to Mid-Range: <ul style="list-style-type: none"> <li>■ Multiple Communication Peripherals</li> <li>■ Linear Programming Space</li> <li>■ PWMs with Independent Time Base</li> </ul>	In addition to Enhanced Mid-Range: <ul style="list-style-type: none"> <li>■ 8x8 Hardware Multiplier</li> <li>■ CAN</li> <li>■ CTMU</li> <li>■ USB</li> <li>■ Ethernet</li> <li>■ 12-bit ADC</li> </ul>
Highlights	Lowest cost in the smallest form factor	Optimal cost to performance ratio	Cost effective with more performance and memory	High performance, optimized for C programming, advanced peripherals

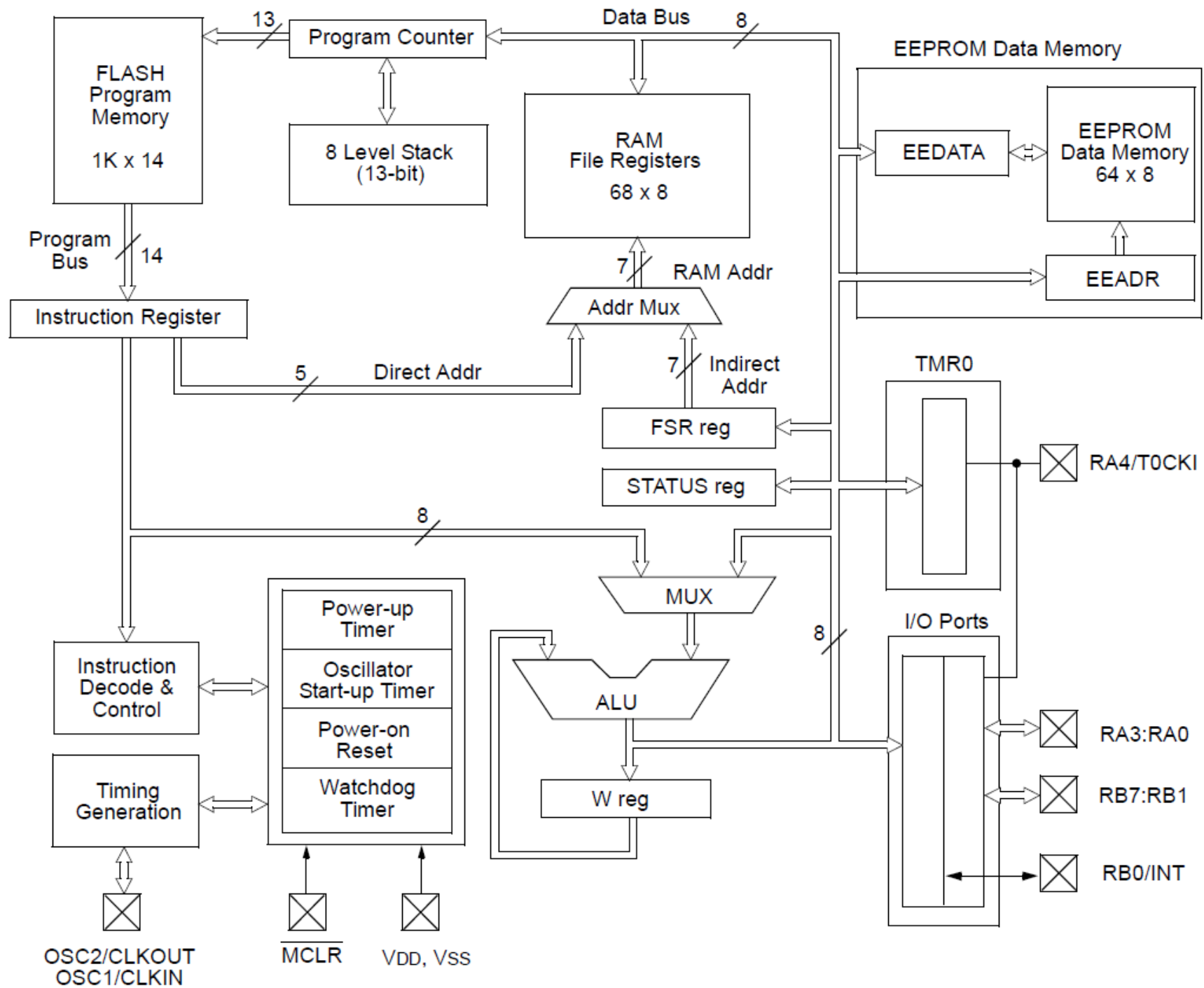




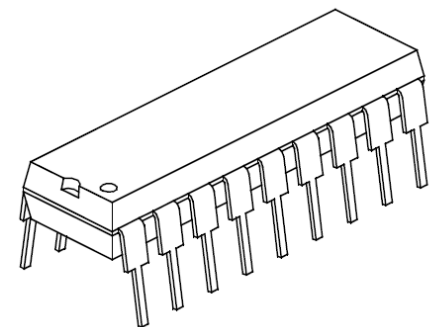
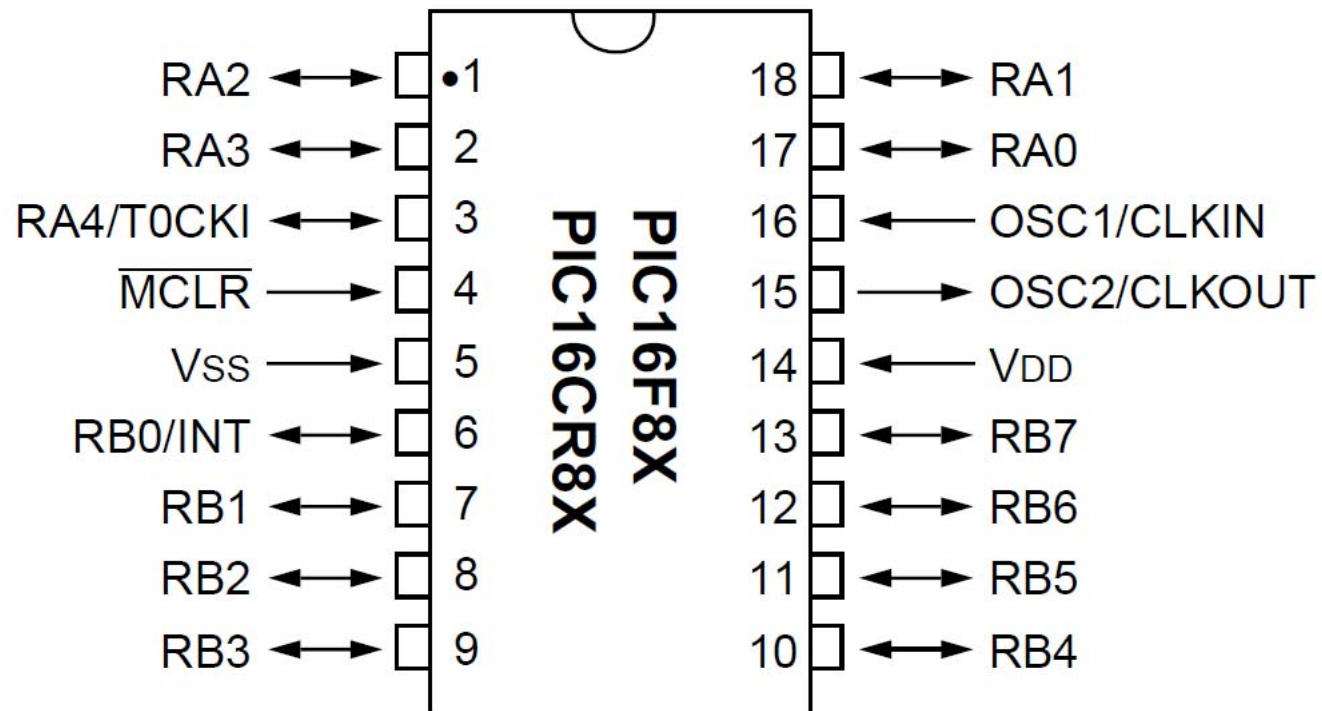
# PIC16F84A

- RISC arhitektura – samo 35 instrukcija
- Sve instrukcije se izvršavaju 1 ciklus, sem skokova (2 ciklusa)
- 1K reči *flash* memorija za smeštanje programa
- 68B data RAM
- 64B data EEPROM
- 10 MHz maksimalna frekvencija
- 14-bitne instrukcije
- 8-bitni podaci
- 15 SFR (*special function registers*)
- 8 – dubina hardverskog steka
- 4 izvora prekida (eksterni RB0/INT, prekoračenje TMR0 tajmera, PORTB<7:4> prekid na promenu vrednosti, završen upis podatka u EEPROM )
- 13 U/I pina



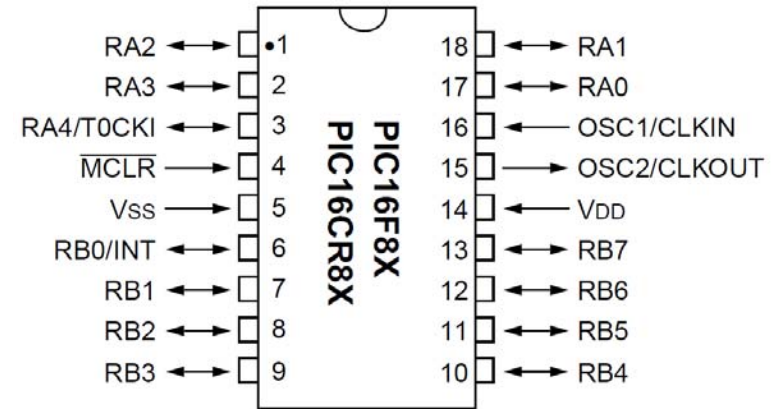


# PIC16F84A



# PIC16F84A

## funkcija pinova



- **OSC1/CLKIN** – ulaz eksternog takta
- **OSC2/CLKOUT** – izlaz kristalnog oscilatora ili izlazni takt koji je  $\frac{1}{4}$  takta OSC1 (tj. instrukcijski takt)
- **MCLR** – (*Master Clear*) niska vrednost resetuje komponentu
- **RA0-RA3** – port A (bidirekcionni U/I port)
- **RA4/T0CKI** – 5. pin porta A ili ulazni takt za TMR0 tajmer/brojač
- **RB0/INT** – 0. pin porta B, eksterni prekidni ulaz
- **RB1-RB3** – 1.-3. pin porta B
- **RB4-RB5** – 4. i 5. pin porta B, prekidni ulaz
- **RB6** – 6. pin porta B, prekidni ulaz, takt pri serijskom programiranju
- **RB7** – 7. pin porta B, prekidni ulaz, podaci pri serijskom programiranju
- **V<sub>SS</sub>** – uzemljenje
- **V<sub>DD</sub>** – (pozitivno) napajanje

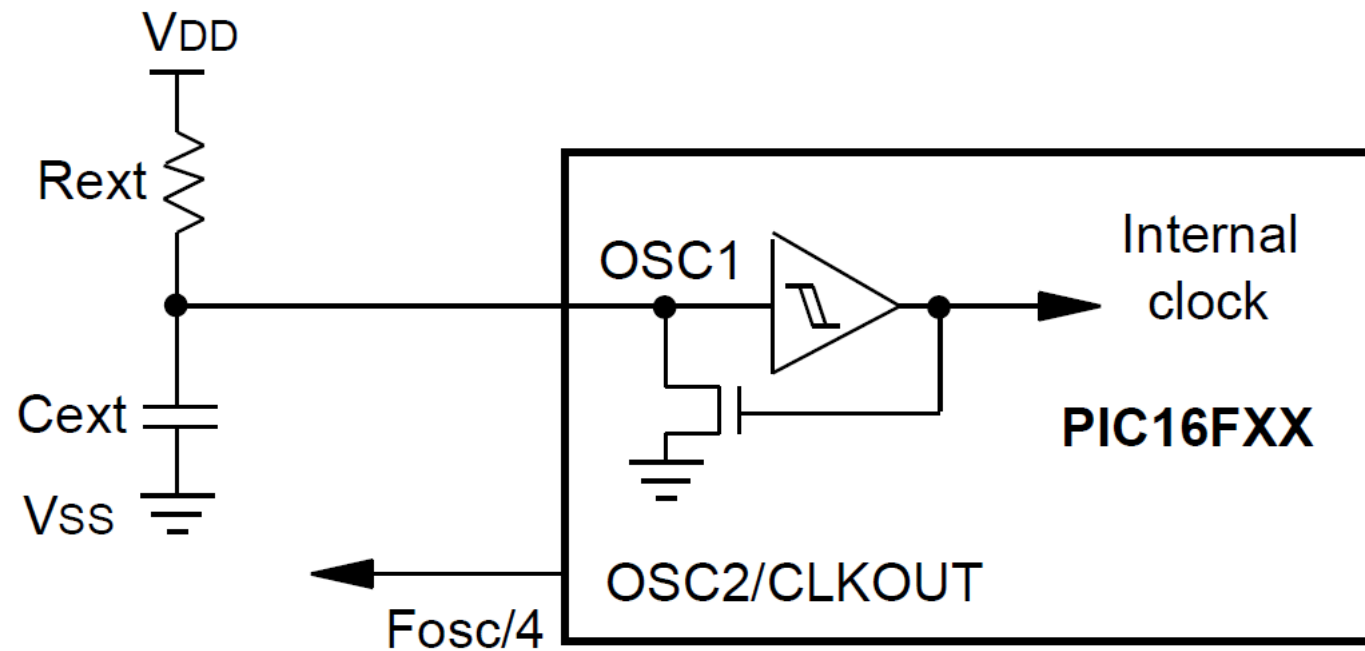


# Oscillator

- Oscilator je srce MCU; njegovi otkucaji predstavljaju sistemski takt koji pokreće čitav uređaj
- Ovaj modul može da se pobuđuje iz različitih izvora, čime se utiče na performanse i potrošnju
- PIC16F84A može da radi u jednom od 4 oscilatorska režima (u skladu sa stanjem konfiguracionih bitova FOSC1 i FOSC0):
  - LP – *Low-Power Crystal* mod (32 kHz do 200kHz)
  - XT – *Medium Gain Crystal* ili *Ceramic Resonator Oscillator* mod (100kHz do 4 MHz)
  - HS – *High Gain Crystal* ili *Ceramic Resonator* mod (4 MHz do 20 MHz)
  - RC – *External Resistor-Capacitor* mod (200kHz do 4MHz)
- Sistemski takt označava se sa  $F_{OSC}$  i uvek je 4x viši od instrukcijskog takta ( $F_{OSC}/4$ ); da bi se ostvarilo izvršenje jedne mašinske instrukcije u jednom taktu (sem grananja i skoka, koje zahtevaju 2 takta).



# Eksterni RC oscilator



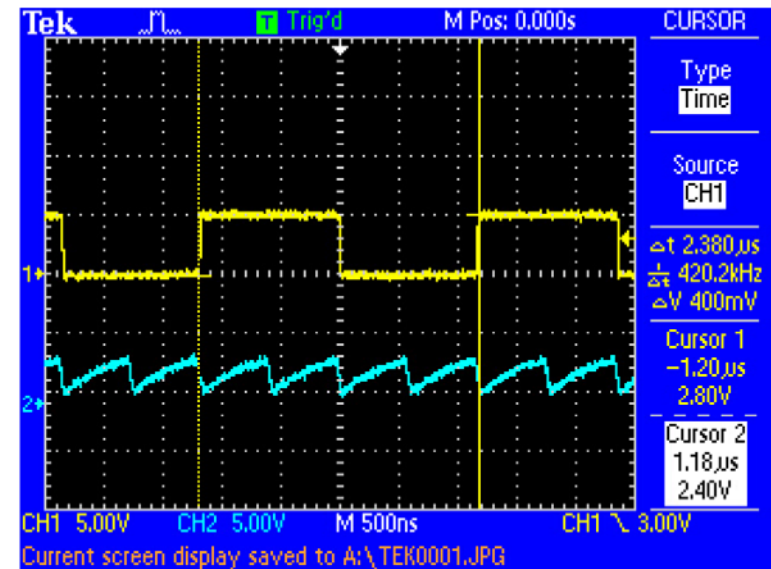
$$5\text{ k}\Omega \leq R_{ext} \leq 100\text{ k}\Omega$$

$$C_{ext} > 20\text{pF}$$

# Izračunavanje frekvencije

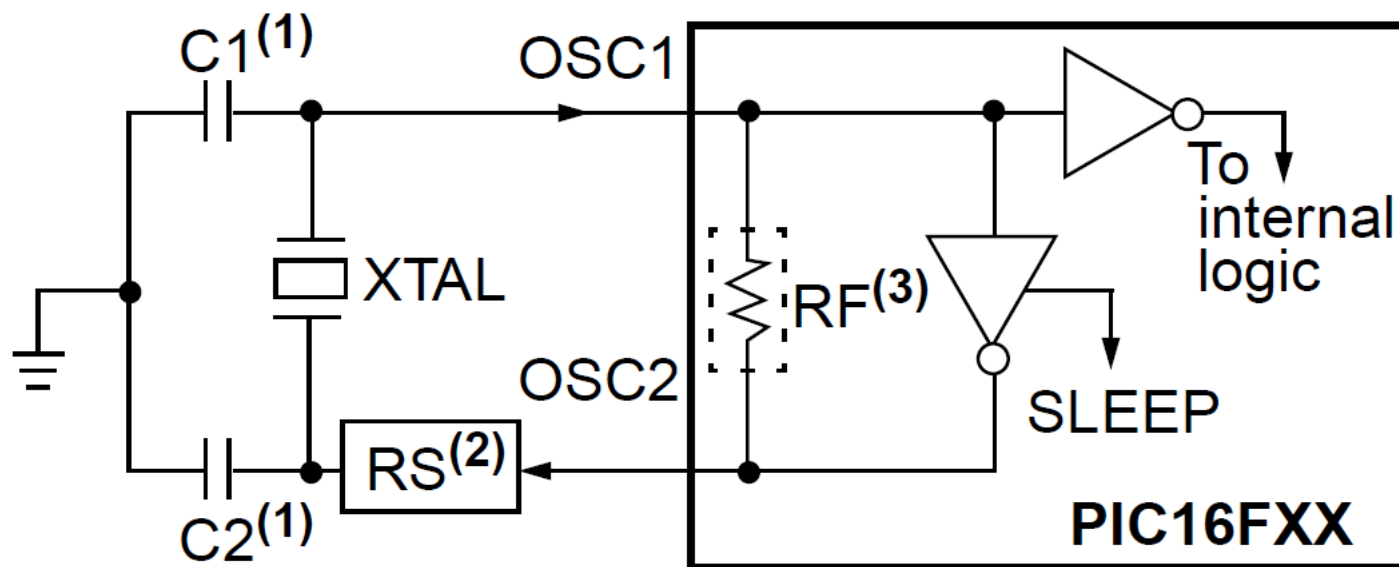
- $T_{OSC} = (R_{EXT} * C_{EXT}) * \ln[V_{DD}/(V_{DD}-V_{IH})]$
- $F_{OSC} = 1/T$
- $InstructionClock = F_{OSC}/4$

- Primer:
  - $V_{DD} = 5V$
  - $V_{IH} = 0.9 * V_{DD} = 4.5V$
  - $R_{EXT} = 10K\Omega$
  - $C_{EXT} = 22pF$



$T_{OSC} = 506ns$ ,  $F_{OSC} = 1.976MHz$ ,  $InstructionClock = 494kHz$

# HS, XT i LP konfiguracija oscilatora



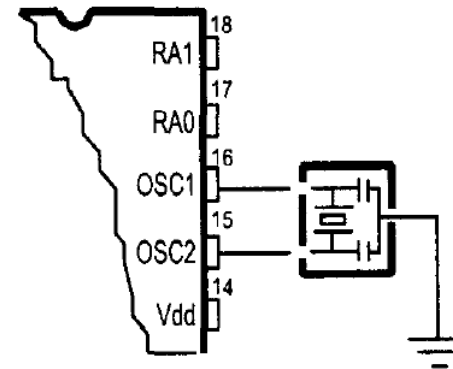
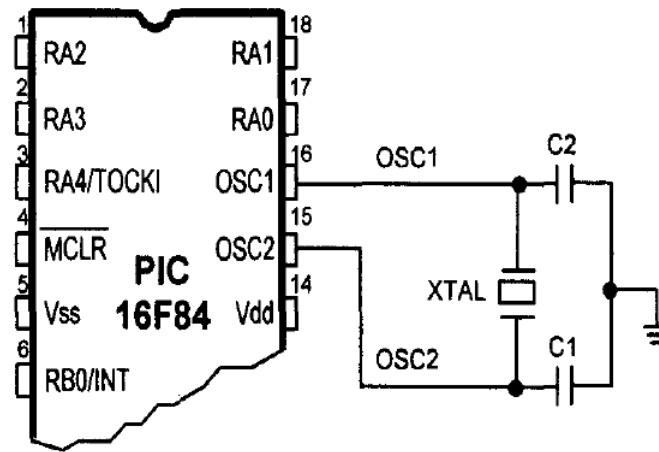
Izbor kondenzatora za kristalne oscilatore

Mode	Freq	OSC1/C1	OSC2/C2
LP	32 kHz	68 - 100 pF	68 - 100 pF
	200 kHz	15 - 33 pF	15 - 33 pF
XT	100 kHz	100 - 150 pF	100 - 150 pF
	2 MHz	15 - 33 pF	15 - 33 pF
	4 MHz	15 - 33 pF	15 - 33 pF
HS	4 MHz	15 - 33 pF	15 - 33 pF
	20 MHz	15 - 33 pF	15 - 33 pF

Izbor kondenzatora za keramičke rezonatore

Mode	Freq	OSC1/C1	OSC2/C2
XT	455 kHz	47 - 100 pF	47 - 100 pF
	2.0 MHz	15 - 33 pF	15 - 33 pF
	4.0 MHz	15 - 33 pF	15 - 33 pF
HS	8.0 MHz	15 - 33 pF	15 - 33 pF
	10.0 MHz	15 - 33 pF	15 - 33 pF

# Oscilatori i rezonatori



## Configuration Word (2007h)

CP	CP	CP	CP	CP	CP	CP	CP	CP	CP	PWRTE	WDTE	FOSC1	FOSC0
bit13													bit0

bit 1-0

**FOSC1:FOSC0:** Oscillator Selection bits

11 = RC oscillator

10 = HS oscillator

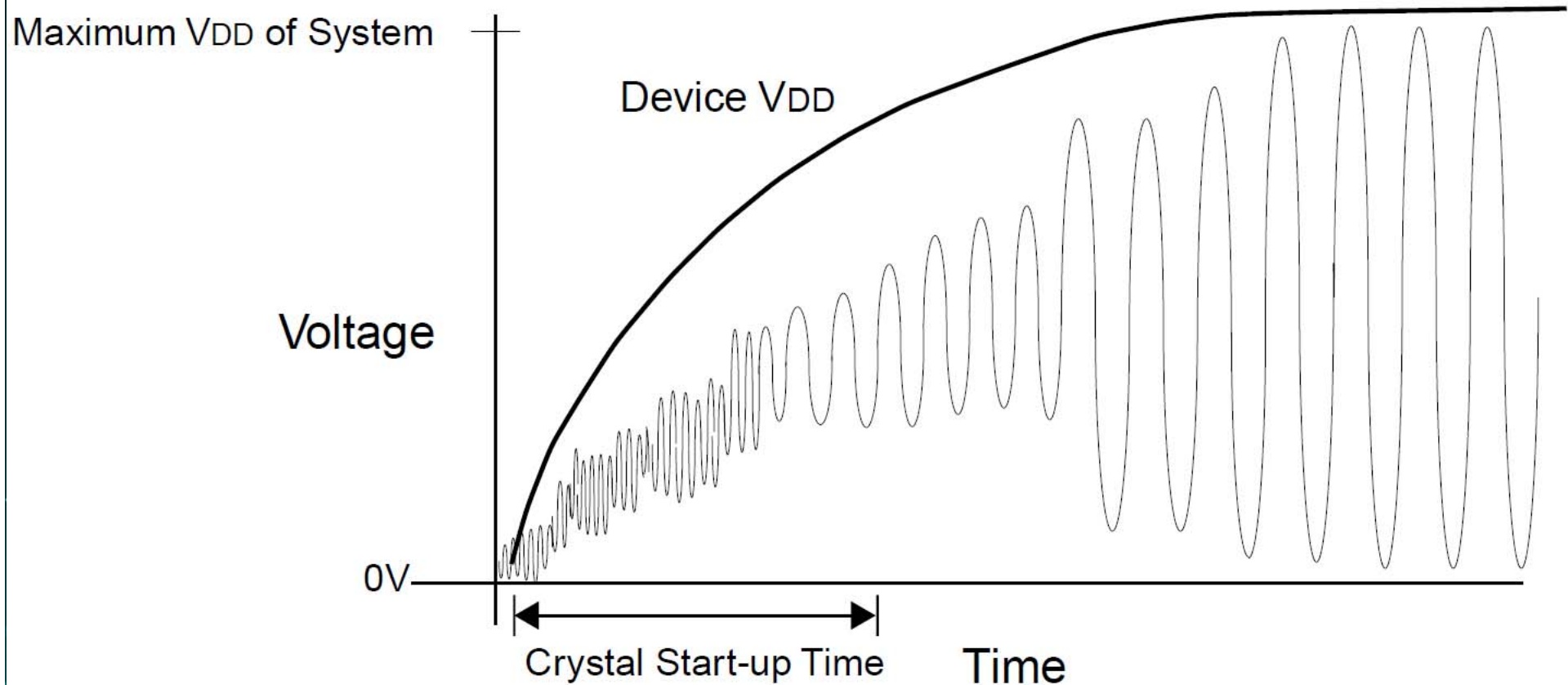
01 = XT oscillator

00 = LP oscillator





# Stabilizacija oscilatora





# PIC16F84A tipovi reset-a

- Resetovanje pri dovođenju napajanja, **POR** (Power-On Reset)
- Resetovanje tokom regularnog rada, dovođenjem logičke „0“ na pin **MCLR**
- Resetovanje za vreme SLEEP režima, dovođenjem logičke „0“ na pin **MCLR**
- Resetovanje na prekoračenje sigurnosnog brojača (**WDT**) tokom **regularnog rada**
- Resetovanje na prekoračenje sigurnosnog brojača (**WDT**) tokom **SLEEP režima** (*Wake-up*)
- Brown-out Reset (BOR)
- Parity Error Reset (PER)

# Power On Reset (POR)

- Kada se uključi PIC MCU, on interno izvršava neke definisane hardverske operacije, kojima se priprema za normalan rad.
- To obuhvata pokretanje oscilatora i stabilizaciju napajanja.
- POR drži MCU u resetovanom stanju dok se te operacije ne završe.
- Time se sprečava pokretanje softvera dok se ne dostigne minimalni napon napajanja i stabilna frekvencija.
- POR može držati MCU u resetovanom stanju duže, ako se zna da je sredina takva da unosi dosta šuma.
- To omogućuje *Power-up Timer (PWRT)*.
- PWRT obezbeđuje fiksno kašnjenje od **72ms** nakon uključivanja.

## Configuration Word (2007h)

CP	CP	CP	CP	CP	CP	CP	CP	CP	CP	PWRTÉ	WDTE	F0SC1	F0SC0
bit13													bit0

bit 3

**PWRTÉ**: Power-up Timer Enable bit  
1 = Power-up Timer is disabled  
0 = Power-up Timer is enabled

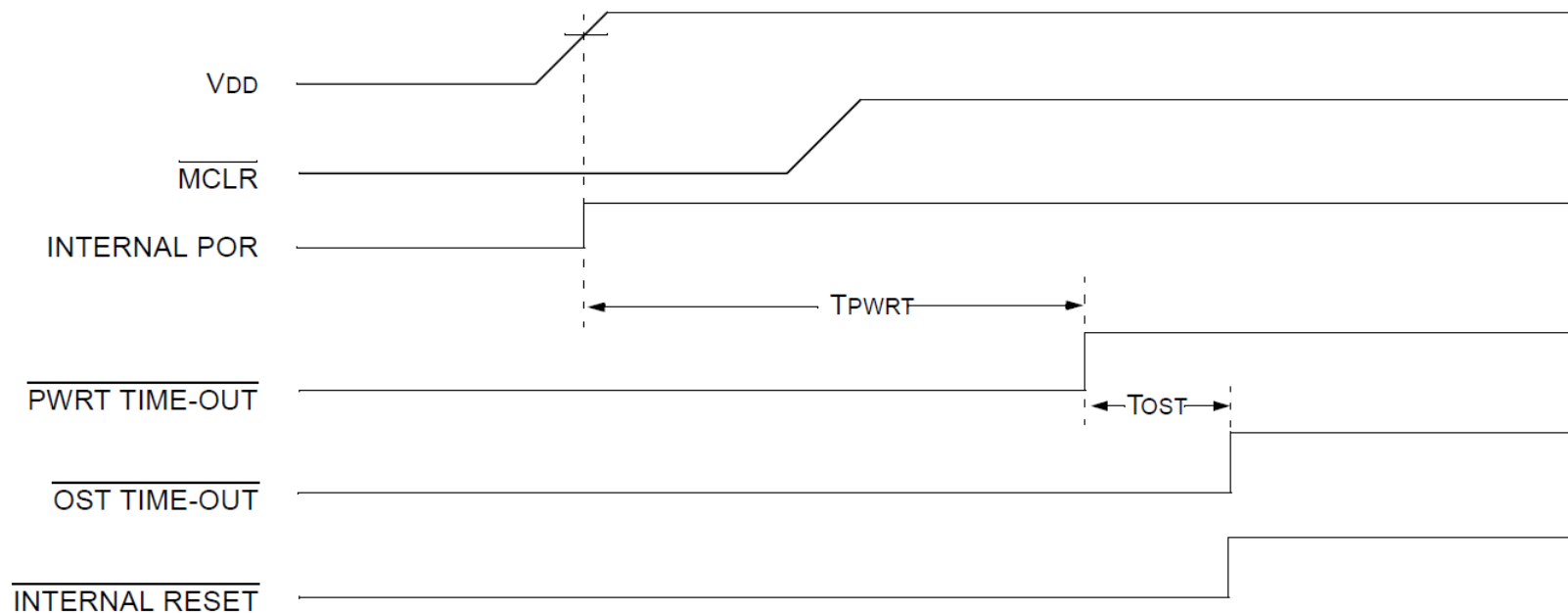


# Power On Reset (POR)

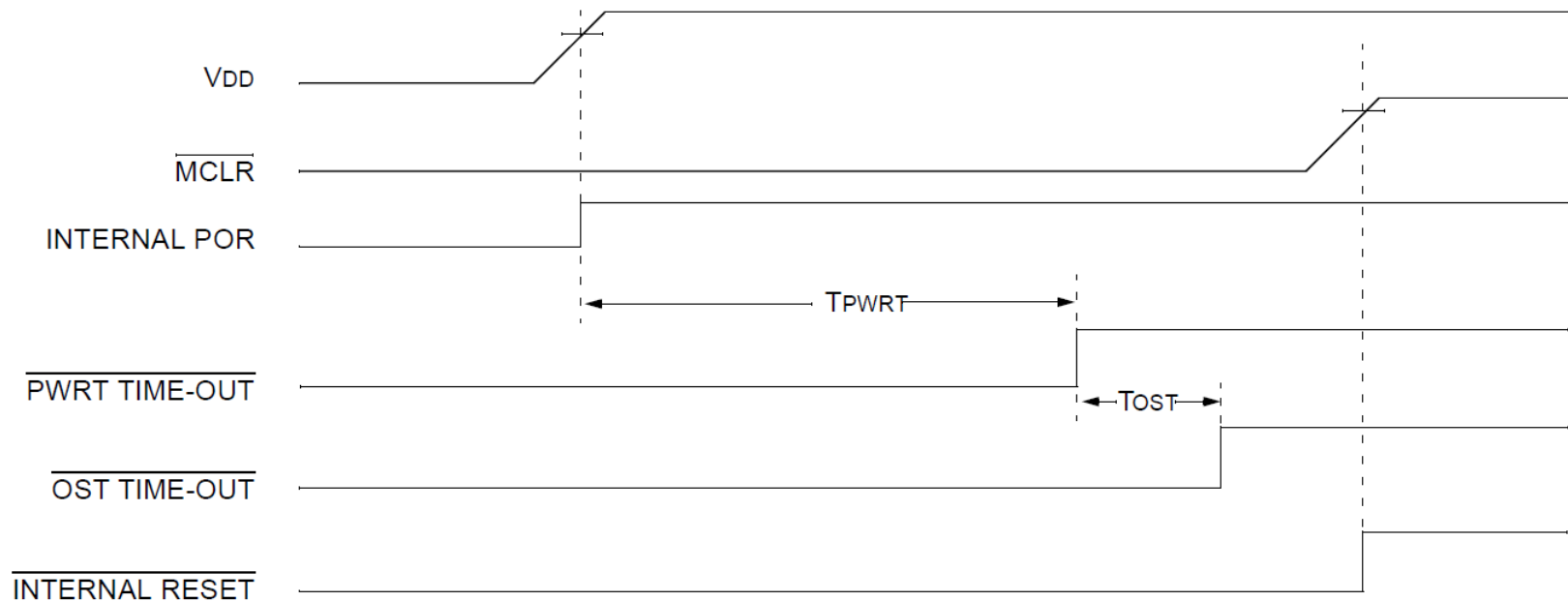
- POR impuls se generiše na samom čipu, kada se detektuje porast  $V_{DD}$  (u opsegu 1.2V – 1.7V).
- Treba povezati MCLR pin na  $V_{DD}$  (direktno ili preko otpornika), pri čemu treba obezbediti minimalnu brzinu porasta  $V_{DD}$ , da bi tako jednostavna konfiguracija funkcionisala ( $> 0.05V/ms$ ).
- $1.2V + 72ms * 0.05V/ms = 4.8V$
- POR kolo ne izaziva interni reset kada  $V_{DD}$  opada.
- Drugi tajmer koji obezbeđuje potrebno kašnjenje je *Oscillator Start-up Timer (OST)*.
- OST period iznosi **1024**  $T_{osc}$  ( $T_{osc} = OSC1$  period) i aktivira se nakon PWRT.
- OST se poziva samo za LP, XT i HS modove i samo za POR i Wake-up (povratak iz SLEEP režima).
- Ako PWRT i OST nisu dovoljni, neophodna je dodatna eksterna elektronika da produži vreme za koje MCU ostaje u resetovanom stanju.



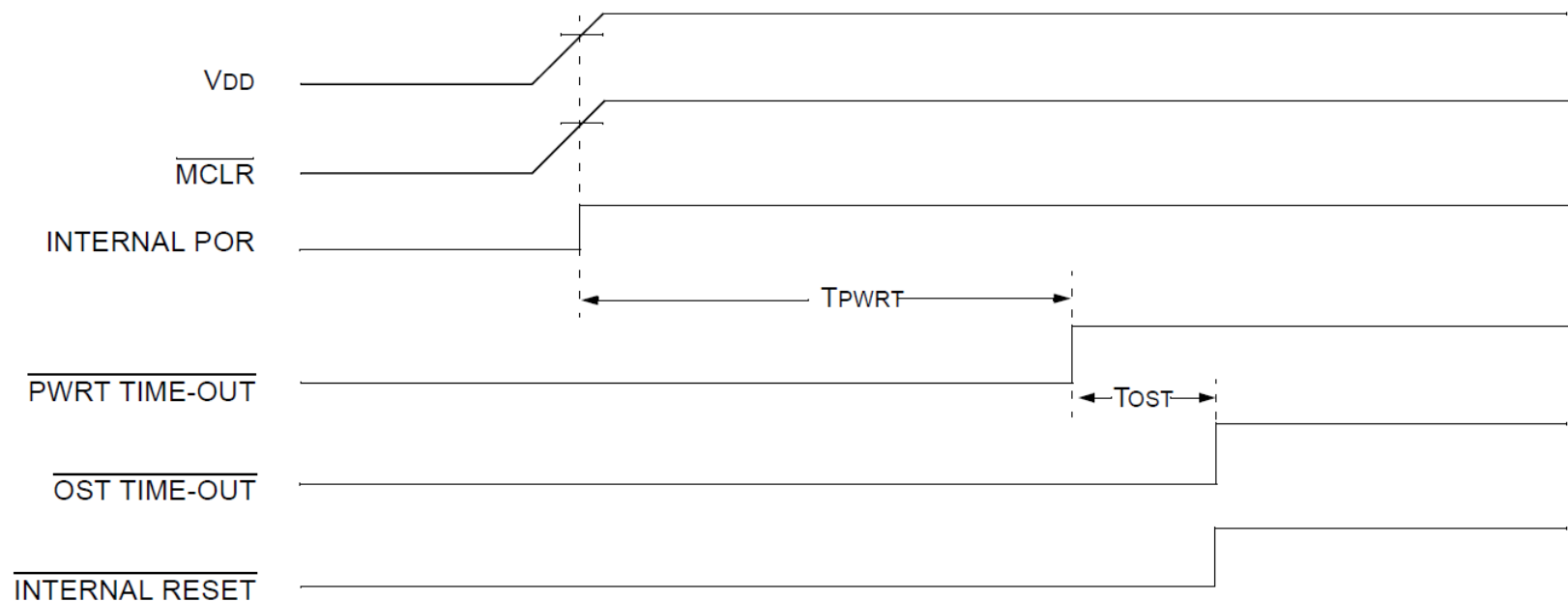
# Time-out sekvenca pri uključivanju kada MCLR nije vezan na $V_{DD}$



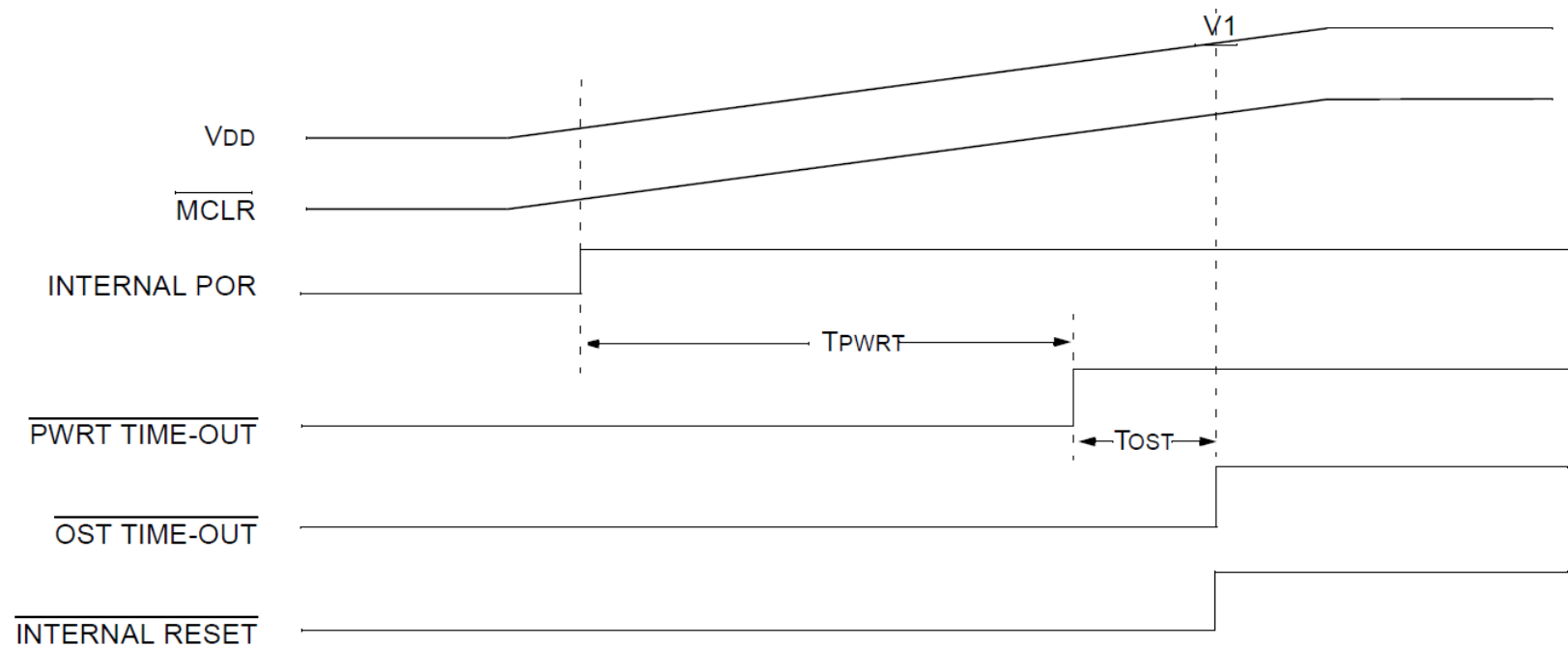
# Time-out sekvenca pri uključivanju kada MCLR nije vezan na $V_{DD}$ (duže aktivan MCLR)



# Time-out sekvenca pri uključivanju kada je MCLR vezan na $V_{DD}$ (brzi porast $V_{DD}$ )



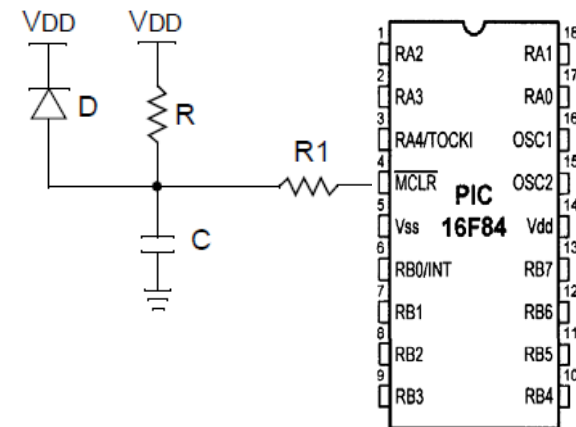
# Time-out sekvenca pri uključivanju kada je MCLR vezan na $V_{DD}$ (spor porast $V_{DD}$ )





# Spoljašnje reset-kolo (za spororastući $V_{DD}$ )

- Spoljašnje reset-kolo koristi se samo kada  $V_{DD}$  raste vrlo sporo (sporije od 0.05 V/ms).
- Dioda pomaže da se kondenzator brzo isprazni pri isključivanju napajanja.
- $R < 40\text{k}\Omega$  obezbeđuje da pad napona na R ne prelazi 0.2V (maksimalna struja curenja na MCLR je  $5\mu\text{A}$ ). Veći pad napona degradira  $V_{IH}$  nivo na MCLR.
- $R1 = 100\ \Omega$  (do  $1\text{ k}\Omega$ ) limitira struju koja utiče u MCLR iz kondenzatora u slučaju elektrostatičkog pražnjenja (ESD) ili električnog preopterećenja (EOS)
- Kondenzator C prilagoditi vremenu trajanja reseta.

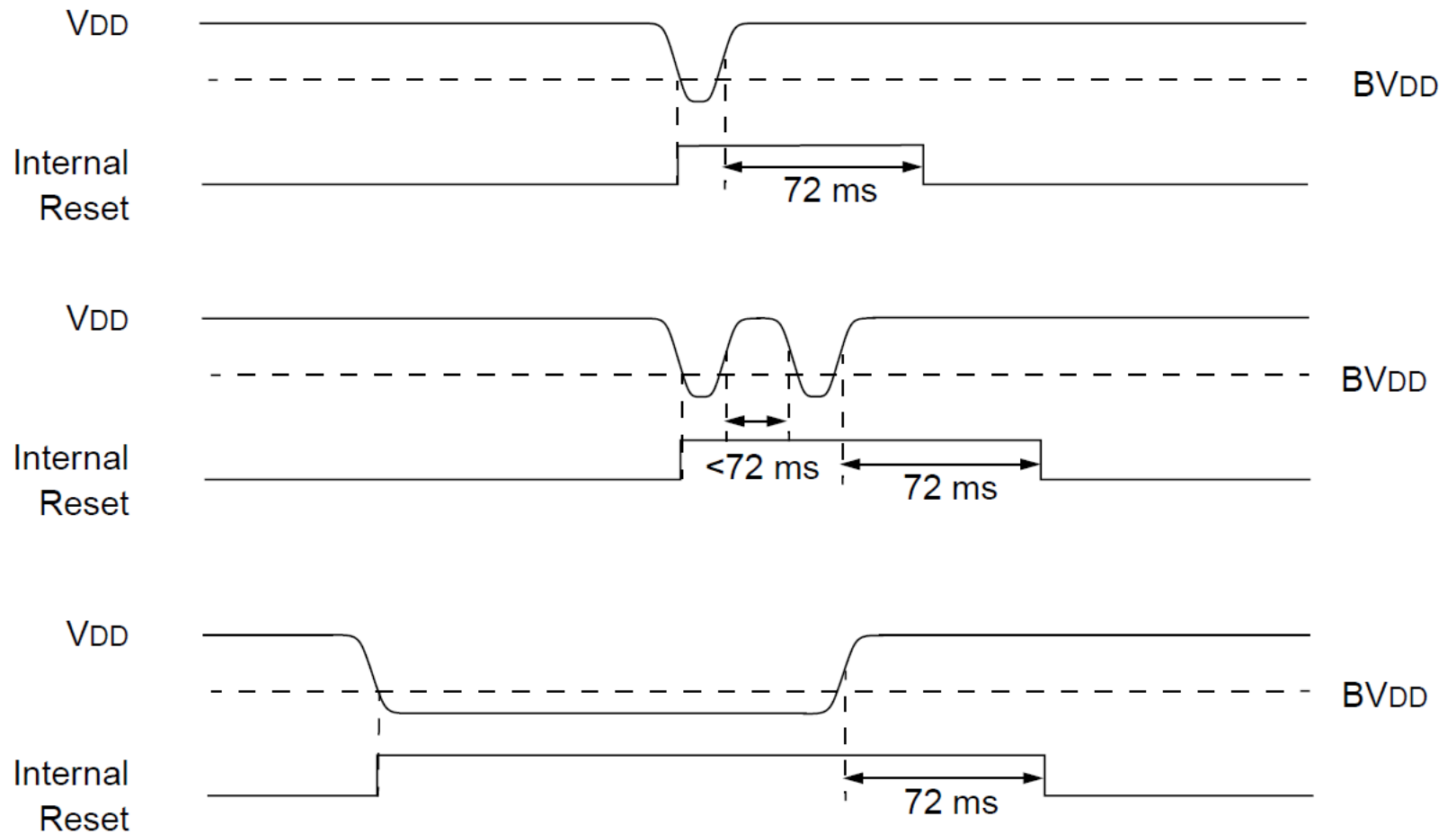




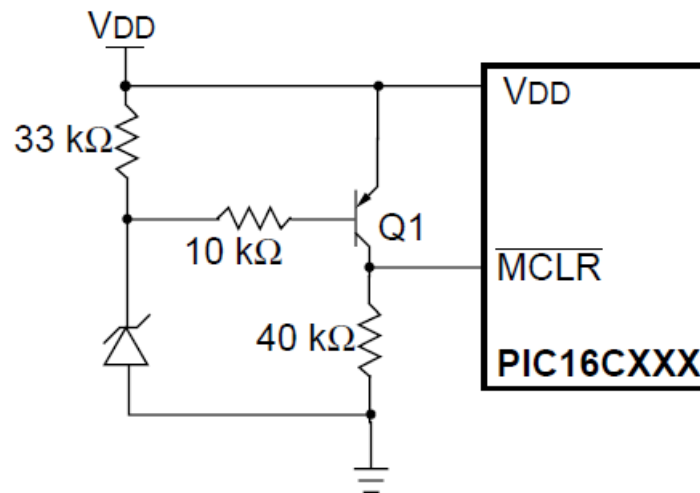
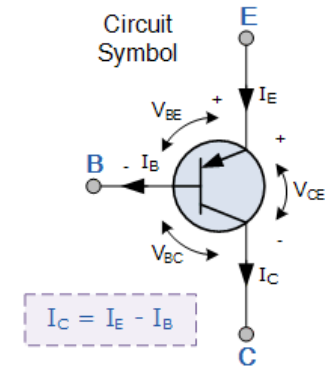
# Brown-out Reset (BOR)

- Omogućuje da MCU bude u resetovanom stanju kada  $V_{DD}$  padne ispod *brown-out* praga.
- Preporuka je da se uključi Power-up Timer (PWRT), kako bi se uključilo dodatno kašnjenje pri povratku iz BOR.
- Kod novih (unapređenih) MCU, BOR ima 4 režima (uvek uključen, isključen u SLEEP modu, upravljani softverski i uvek isključen), koji se podešavaju bitovima u konfiguracionoj reči.
- Ne postoje kod PIC16F84A

# Reset na pad napona (Brown-out Reset)



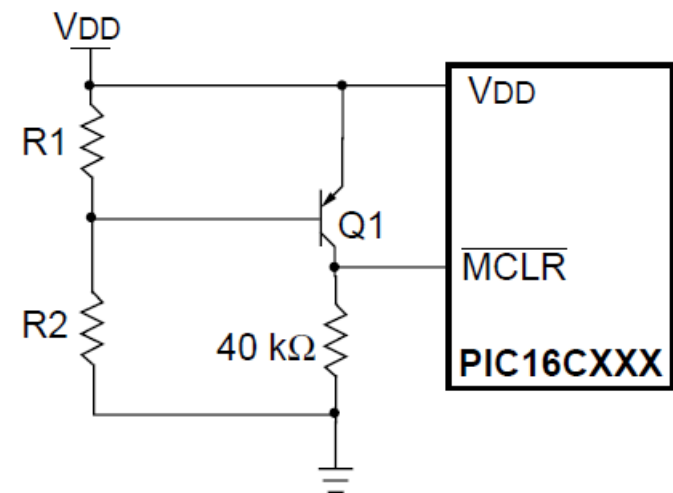
# Spoljašnja kola za Brown-out zaštitu



Reset se aktivira kada je:

$$V_{DD} < V_Z + 0.7V,$$

gde je  $V_Z$  Zenerov napon (napon kada Zener dioda provede struju od katode ka anodi)

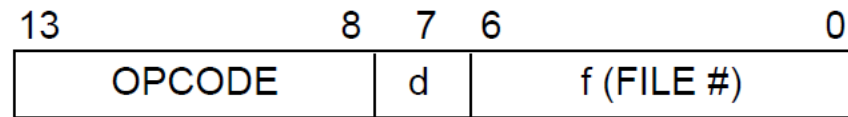


Jeftinije, ali manje precizno rešenje. Reset se aktivira (tj. tranzistor Q1 prestaje da provodi struju) kada je:

$$V_{DD} < 0.7V * (R1+R2)/R1$$

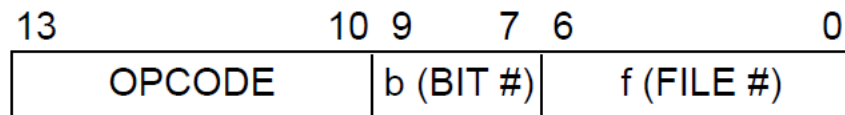
# Format instrukcije

## Byte-oriented file register operations



d = 0 for destination W  
d = 1 for destination f  
f = 7-bit file register address

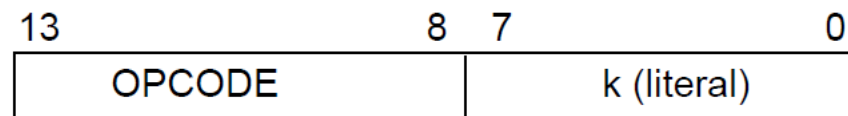
## Bit-oriented file register operations



b = 3-bit bit address  
f = 7-bit file register address

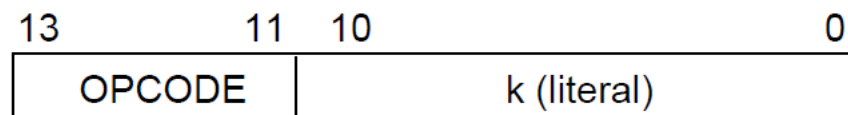
## Literal and control operations

### General



k = 8-bit immediate value

### CALL and GOTO instructions only

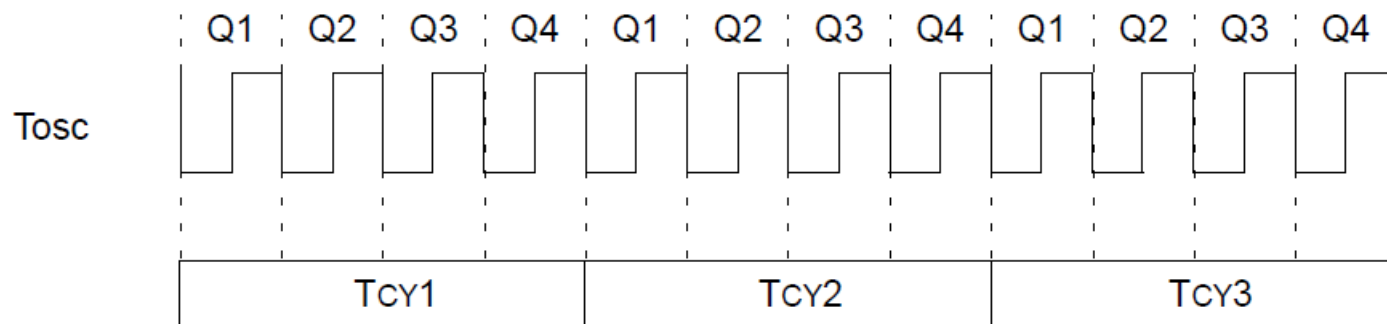


k = 11-bit immediate value

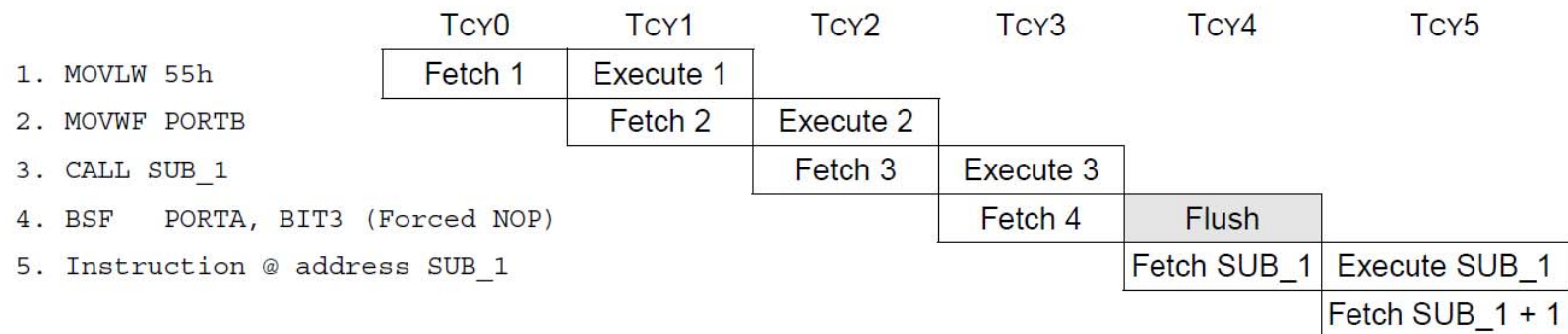
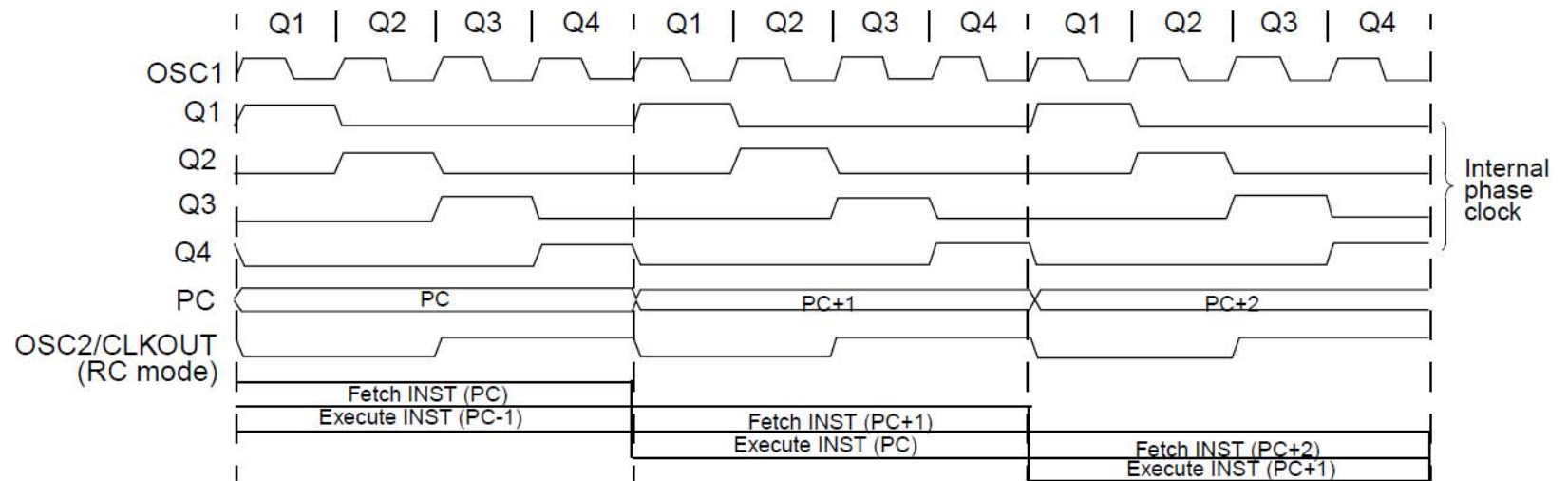


# Instrukcioni takt

- Svaki ciklus izvršenja instrukcije ( $T_{CY}$ ) se sastoji od 4 Q ciklusa (Q1-Q4)
- Q ciklus odgovara osnovnom taktu oscilatora ( $T_{OSC}$ )
- Q1 – dekodiranje instrukcije (ili NOP)
- Q2 – čitanje podataka (ili NOP)
- Q3 – obrada podataka
- Q4 – upis rezultata (ili NOP)

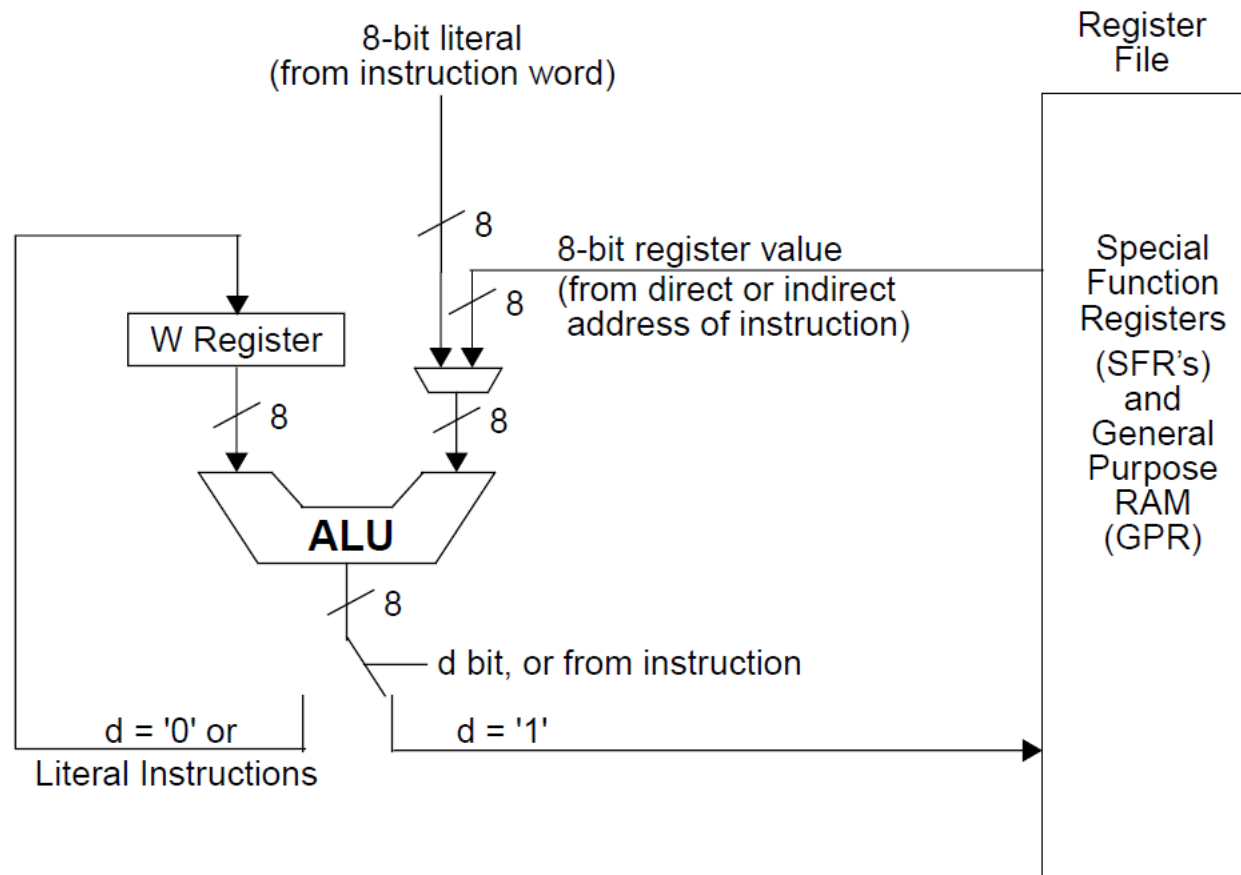


# Instrukcioni ciklus





# Aritmetičko-logička jedinica



- **Aritmetičke operacije**
  - ADD sabiranje
  - SUB oduzimanje
- **Logičke operacije**
  - AND logičko I
  - OR logičko ILI
  - XOR logičko eksl. ILI
  - COM komplement
- **Ostale operacije**
  - RL rotacija ulevo
  - RR rotacija udesno
  - INC inkrementiranje
  - DEC dekrmentiranje
  - SWAP zamena polubajtova
  - Postavljanje, brisanje i testiranje pojedinačnih bittova

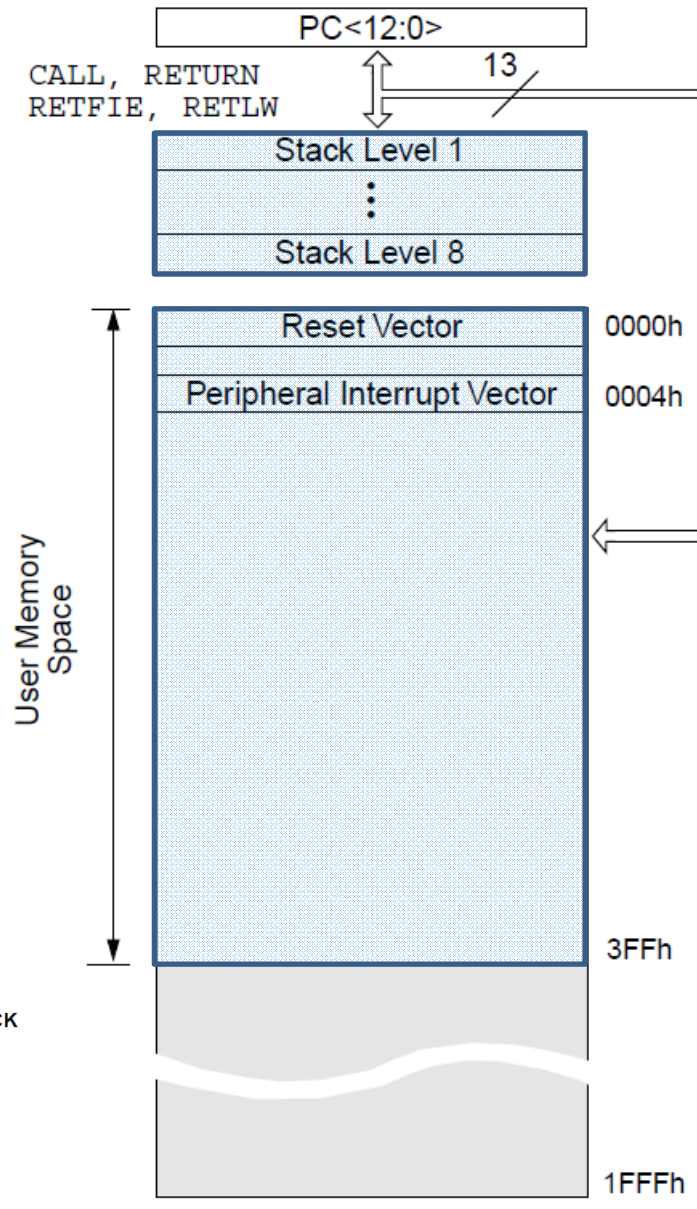




# Organizacija memorije

- Postoje dva bloka memorije
  - Programska memorija
  - Memorija sa podacima
- Svaki blok ima svoju magistralu (Harvard arhitektura)
- Memorija sa podacima deli se na
  - RAM opšte namene
  - Registre specijalnih funkcija (SFR)
  - EEPROM (indirektno mapirana u memoriju podataka – 64B)

# Programska memorija, stek i registri

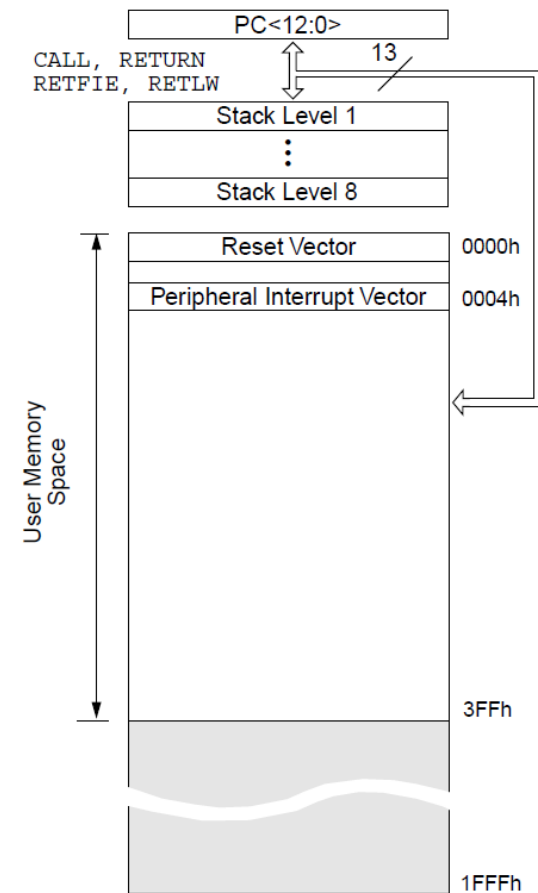


File Address	Indirect addr. <sup>(1)</sup>	Indirect addr. <sup>(1)</sup>	File Address
00h			80h
01h	TMR0	OPTION	81h
02h	PCL	PCL	82h
03h	STATUS	STATUS	83h
04h	FSR	FSR	84h
05h	PORTA	TRISA	85h
06h	PORTB	TRISB	86h
07h			87h
08h	EEDATA	EECON1	88h
09h	EEADR	EECON2 <sup>(1)</sup>	89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch			8Ch
	68 General Purpose registers (SRAM) / Mapped (accesses) in Bank 0		
4Fh			CFh
50h			D0h
	Bank 0 / Bank 1		
7Fh			FFh



# Programska memorija

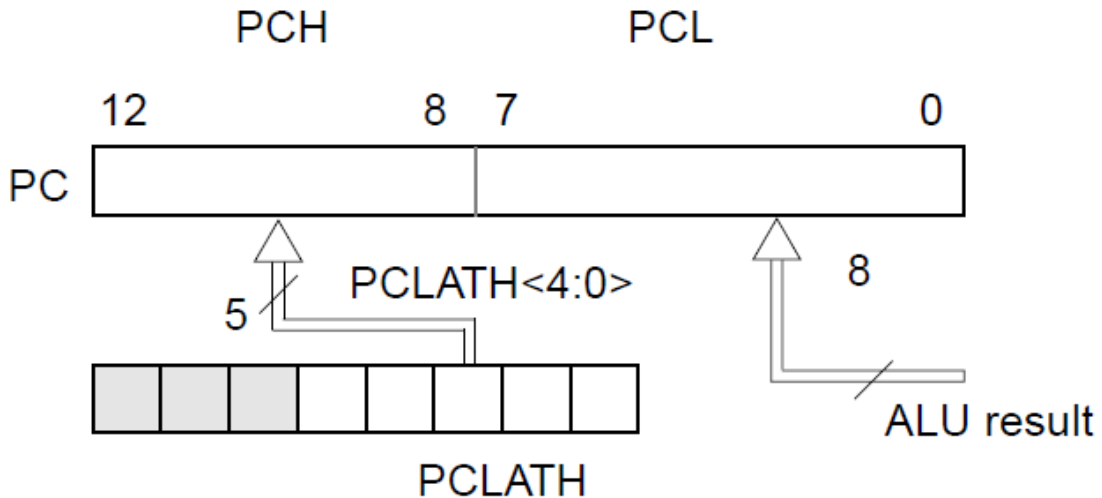
- $8k \times 14b$  organizacija, ali PIC16F84A „vidi“ samo  $1k \times 14b$  (0000h-03FFh)
- Adrese preko 1k se „premotavaju“ (npr. 20h, 420h, 820h itd. pristupiće istoj lokaciji – 20h)
- Lokacije 0000h i 0004h imaju posebnu namenu (reset i prekid)
- Reset postavlja programski brojač (PC) na adresu 0h (*Reset Vector Address*)
- Kada se potvrdi prekid, PC se postavlja na 04h (*Interrupt Vector Address*)
- Kalibracione informacije postoje samo kod nekih čipova i njih proizvođač upisuje na kraju programske memorije (implemetira se kao RETLW instrukcija sa vrednošću koja odgovara kalibracionoj informaciji)



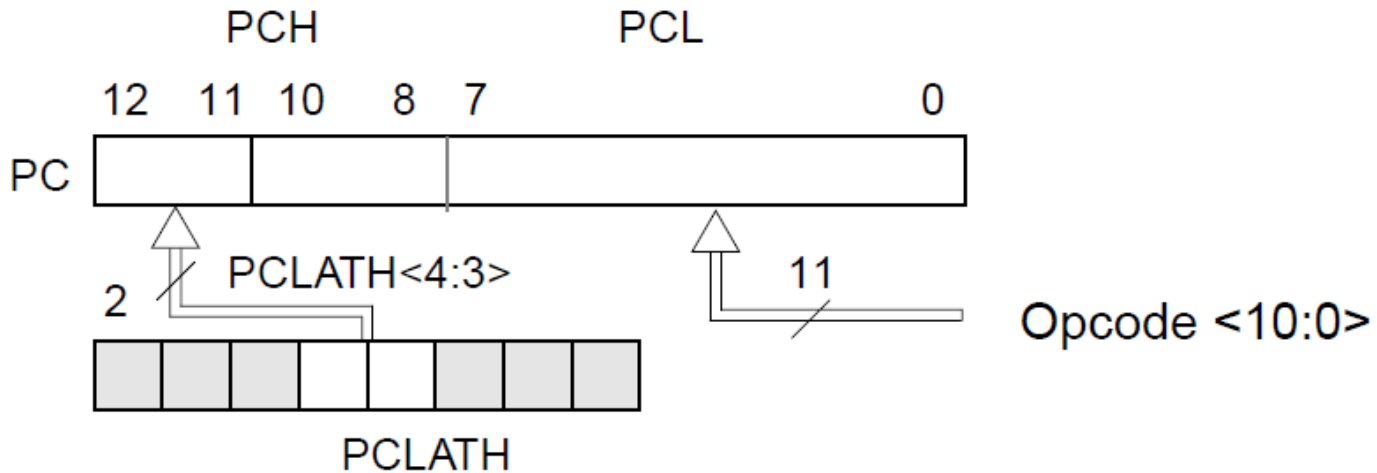
# Programski brojač

- Programski brojač (*Program Counter* - PC) definiše adresu instrukcije koju treba pribaviti radi izvršenja i dužine je 13b
- Niži bajt se naziva **PCL** registar i on se može i čitati i upisivati
- Viši bajt se naziva **PCH** registar, sadrži PC[12:8] bitove i ne može se direktno čitati ili upisivati
- Ažuriranje PCH registra ide preko **PCLATH** registra (**P**rogram **C**ounter **L**atch **H**igh)
- Memorija je kod PIC MCU srednje klase podeljena na stranice od po 2K (16F84A ima samo 1K); prelazak fizičke granice vrši premotavanje (stranice se menjaju pomoću PCLATH)

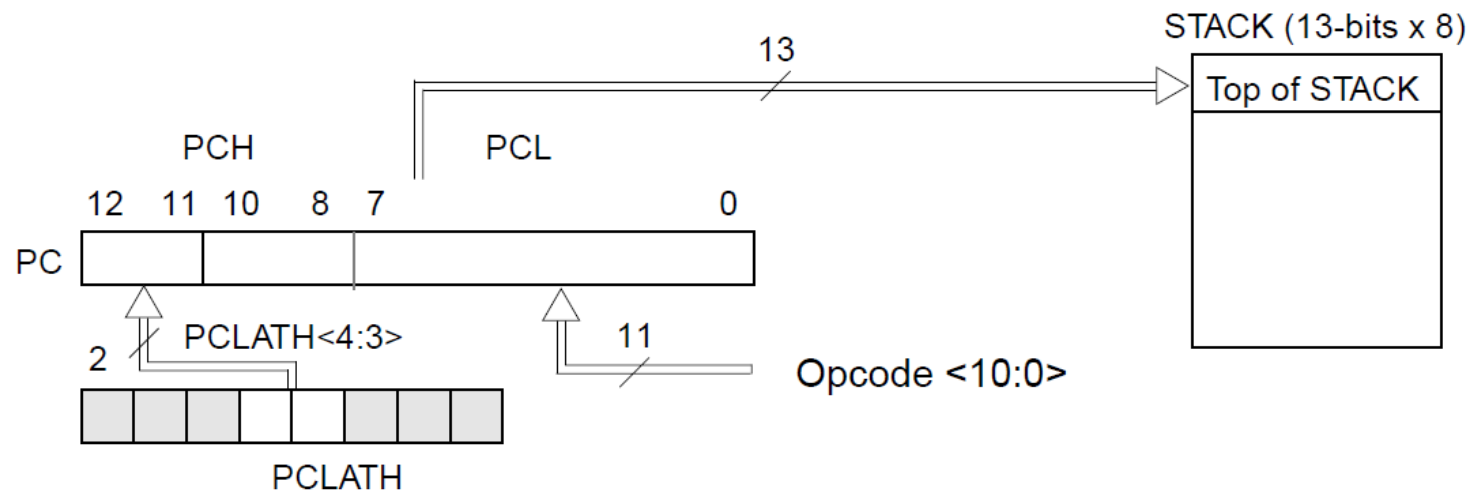
File Address		File Address	
00h	Indirect addr. <sup>(1)</sup>	Indirect addr. <sup>(1)</sup>	80h
01h	TMR0	OPTION_REG	81h
02h	PCL	PCL	82h
03h	STATUS	STATUS	83h
04h	FSR	FSR	84h
05h	PORTA	TRISA	85h
06h	PORTB	TRISB	86h
07h	—	—	87h
08h	EEDATA	EECON1	88h
09h	EEADR	EECON2 <sup>(1)</sup>	89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch			8Ch



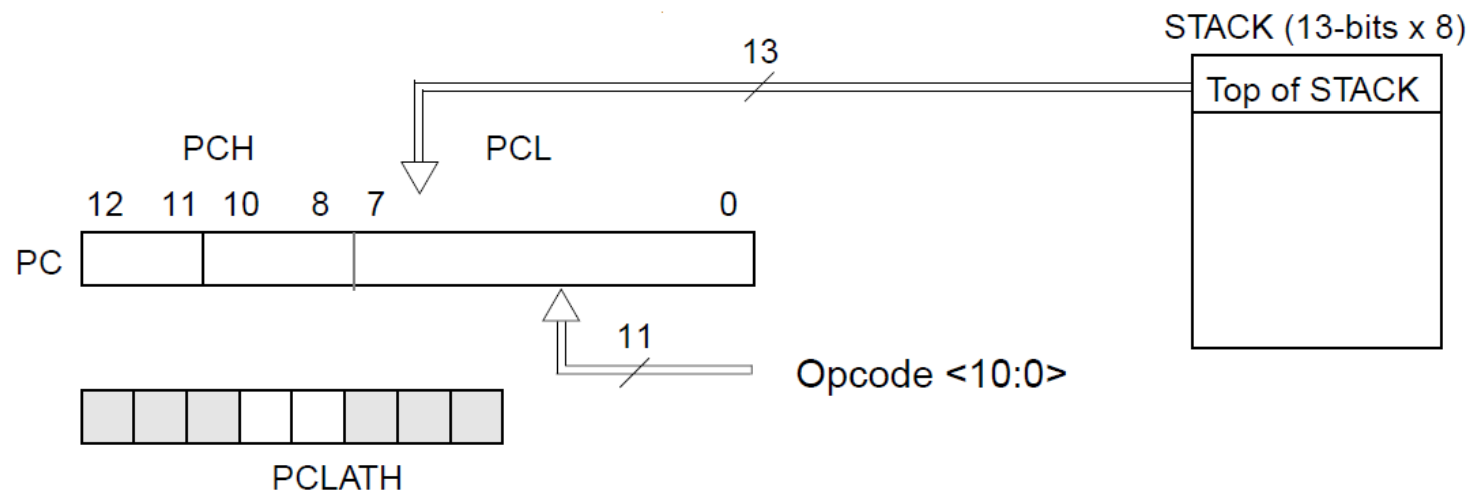
# GOTO instrukcija



## CALL instrukcija



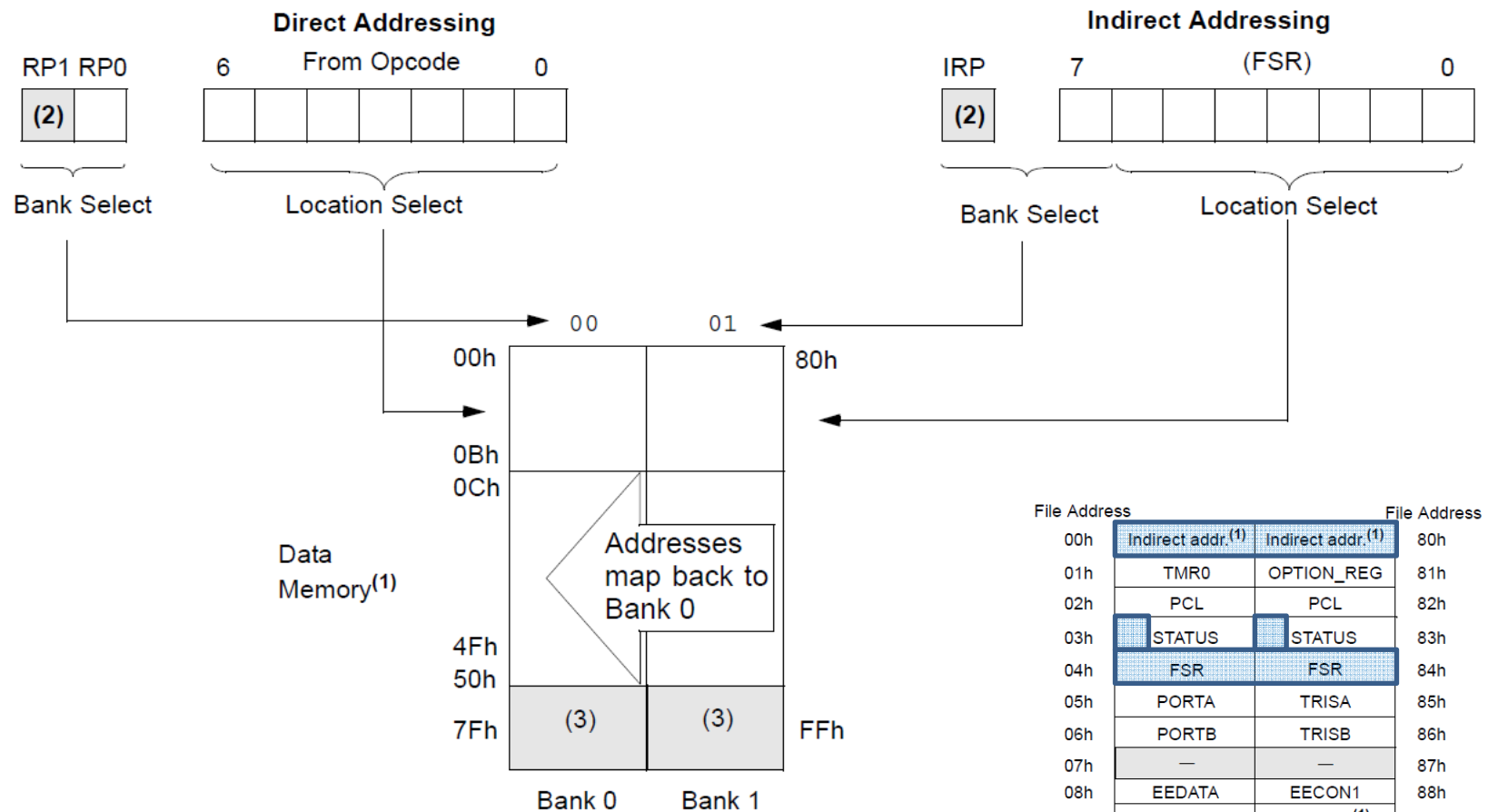
## RETURN, RETFIE ili RETLW instrukcija



Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-on Reset	Value on all other resets (Note3)
Bank 0											
00h	INDF	Uses contents of FSR to address data memory (not a physical register)								---- --	---- --
01h	TMR0	8-bit real-time clock/counter								xxxx xxxx	uuuu uuuu
02h	PCL	Low order 8 bits of the Program Counter (PC)								0000 0000	0000 0000
03h	STATUS <sup>(2)</sup>	IRP	RP1	RP0	T0	PD	Z	DC	C	0001 1xxx	000q quuu
04h	FSR	Indirect data memory address pointer 0								xxxx xxxx	uuuu uuuu
05h	PORTA	—	—	—	RA4/T0CKI	RA3	RA2	RA1	RA0	---x xxxx	---u uuuu
06h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0/INT	xxxx xxxx	uuuu uuuu
07h		Unimplemented location, read as '0'								---- --	---- --
08h	EEDATA	EEPROM data register								xxxx xxxx	uuuu uuuu
09h	EEADR	EEPROM address register								xxxx xxxx	uuuu uuuu
0Ah	PCLATH	—	—	—	Write buffer for upper 5 bits of the PC <sup>(1)</sup>					---0 0000	---0 0000
0Bh	INTCON	GIE	EEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
Bank 1											
80h	INDF	Uses contents of FSR to address data memory (not a physical register)								---- --	---- --
81h	OPTION_REG	RBP	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h	PCL	Low order 8 bits of Program Counter (PC)								0000 0000	0000 0000
83h	STATUS <sup>(2)</sup>	IRP	RP1	RP0	T0	PD	Z	DC	C	0001 1xxx	000q quuu
84h	FSR	Indirect data memory address pointer 0								xxxx xxxx	uuuu uuuu
85h	TRISA	—	—	—	PORTA data direction register					---1 1111	---1 1111
86h	TRISB	PORTB data direction register								1111 1111	1111 1111
87h		Unimplemented location, read as '0'								---- --	---- --
88h	EECON1	—	—	—	EEIF	WRERR	WREN	WR	RD	---0 x000	---0 q000
89h	EECON2	EEPROM control register 2 (not a physical register)								---- --	---- --
0Ah	PCLATH	—	—	—	Write buffer for upper 5 bits of the PC <sup>(1)</sup>					---0 0000	---0 0000
0Bh	INTCON	GIE	EEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u



# Direktno i indirektno adresiranje



- Note** 1: For memory map detail, see Figure 2-2.  
 2: Maintain as clear for upward compatibility with future products.  
 3: Not implemented.



# Indirektno adresiranje

- **INDF** registar (00h) nije fizički registar.
- Adresiranjem ovog registra se zapravo adresira ono na šta ukazuje **FSR** registar (04h).
- FSR je pokazivač; njegov sadržaj ukazuje na lokaciju sa podatkom.
- Kada se podaci upisuju ili čitaju iz **INDF** registra, oni se zapravo upisuju ili čitaju iz registra na koji ukazuje FSR (+ najviši bit STATUS registra – IRP).
- Na primer:
  - Na lokaciji 05 je upisano 10h
  - Na lokaciji 06 je upisano 0Ah
  - Učitati vrednost 05 u **FSR** registar
  - Čitanje **INDF** registra će zapravo vratiti 10h
  - Inkrementiranjem, vrednost u **FSR** postaje 06
  - Čitanjem **INDF** registra dobija je 0Ah

# STATUS register (03h, 83h)

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C

bit 7

bit 0

bit 7-6

**Unimplemented:** Maintain as '0'

bit 5

**RP0:** Register Bank Select bits (used for direct addressing)

01 = Bank 1 (80h - FFh)

00 = Bank 0 (00h - 7Fh)

bit 4

**$\overline{\text{TO}}$ :** Time-out bit

1 = After power-up, CLRWD $\overline{\text{T}}$  instruction, or SLEEP instruction

0 = A WDT time-out occurred

bit 3

**$\overline{\text{PD}}$ :** Power-down bit

1 = After power-up or by the CLRWD $\overline{\text{T}}$  instruction

0 = By execution of the SLEEP instruction

bit 2

**Z:** Zero bit

1 = The result of an arithmetic or logic operation is zero

0 = The result of an arithmetic or logic operation is not zero

bit 1

**DC:** Digit carry/borrow bit (ADDWF, ADDLW, SUBLW, SUBWF instructions) (for borrow, the polarity is reversed)

1 = A carry-out from the 4th low order bit of the result occurred

0 = No carry-out from the 4th low order bit of the result

bit 0

**C:** Carry/borrow bit (ADDWF, ADDLW, SUBLW, SUBWF instructions) (for borrow, the polarity is reversed)

1 = A carry-out from the Most Significant bit of the result occurred

0 = No carry-out from the Most Significant bit of the result occurred

**Note:** A subtraction is executed by adding the two's complement of the second operand. For rotate (RRF, RLF) instructions, this bit is loaded with either the high or low order bit of the source register.

File Address		File Address
00h	Indirect addr. <sup>(1)</sup>	80h
01h	TMR0	81h
02h	PCL	82h
03h	STATUS	83h
04h	FSR	84h
05h	PORTA	85h
06h	PORTB	86h
07h	—	87h
08h	EEDATA	88h
09h	EEADR	89h
0Ah	PCLATH	8Ah
0Bh	INTCON	8Bh
0Ch		8Ch

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

# OPTION registrar (81h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0

bit 7

bit 0

bit 7

**RBPU:** PORTB Pull-up Enable bit

1 = PORTB pull-ups are disabled

0 = PORTB pull-ups are enabled by individual port latch values

bit 6

**INTEDG:** Interrupt Edge Select bit

1 = Interrupt on rising edge of RB0/INT pin

0 = Interrupt on falling edge of RB0/INT pin

bit 5

**T0CS:** TMR0 Clock Source Select bit

1 = Transition on RA4/T0CKI pin

0 = Internal instruction cycle clock (CLKOUT)

bit 4

**T0SE:** TMR0 Source Edge Select bit

1 = Increment on high-to-low transition on RA4/T0CKI pin

0 = Increment on low-to-high transition on RA4/T0CKI pin

bit 3

**PSA:** Prescaler Assignment bit

1 = Prescaler is assigned to the WDT

0 = Prescaler is assigned to the Timer0 module

bit 2-0

**PS2:PS0:** Prescaler Rate Select bits

Bit Value    TMR0 Rate    WDT Rate

000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

File Address	Indirect addr. <sup>(1)</sup>	Indirect addr. <sup>(1)</sup>	File Address
00h			80h
01h	TMR0	OPTION_REG	81h
02h	PCL	PCL	82h
03h	STATUS	STATUS	83h
04h	FSR	FSR	84h
05h	PORTA	TRISA	85h
06h	PORTB	TRISB	86h
07h	—	—	87h
08h	EEDATA	EECON1	88h
09h	EEADR	EECON2 <sup>(1)</sup>	89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch			8Ch



# Prekidi

- PIC16F84A ima 4 izvora prekida
  - **Eksterni prekid** na pinu **RB0/INT** (okida se ivicom; prednjom (OPTION\_REG / INTEDG = 1) ili zadnjom (INTEDG = 0); može probuditi PIC iz SLEEP režima, ako je **INTE** fleg setovan pre „uspavljivanja“)
  - **Prekoračenje tajmera TMR0** (prelazak FFh→00h)
  - **Promena na portu B** (pinovi RB7-RB4)
  - **Završen upis u EEPROM**
- **INTerrupt CONtrol** (INTCON) registar omogućuje pojedinačne prekide (ili sve zajedno) i beleži pojedinačne zahteve za prekidom u odgovarajućim bitovima (flegovi)



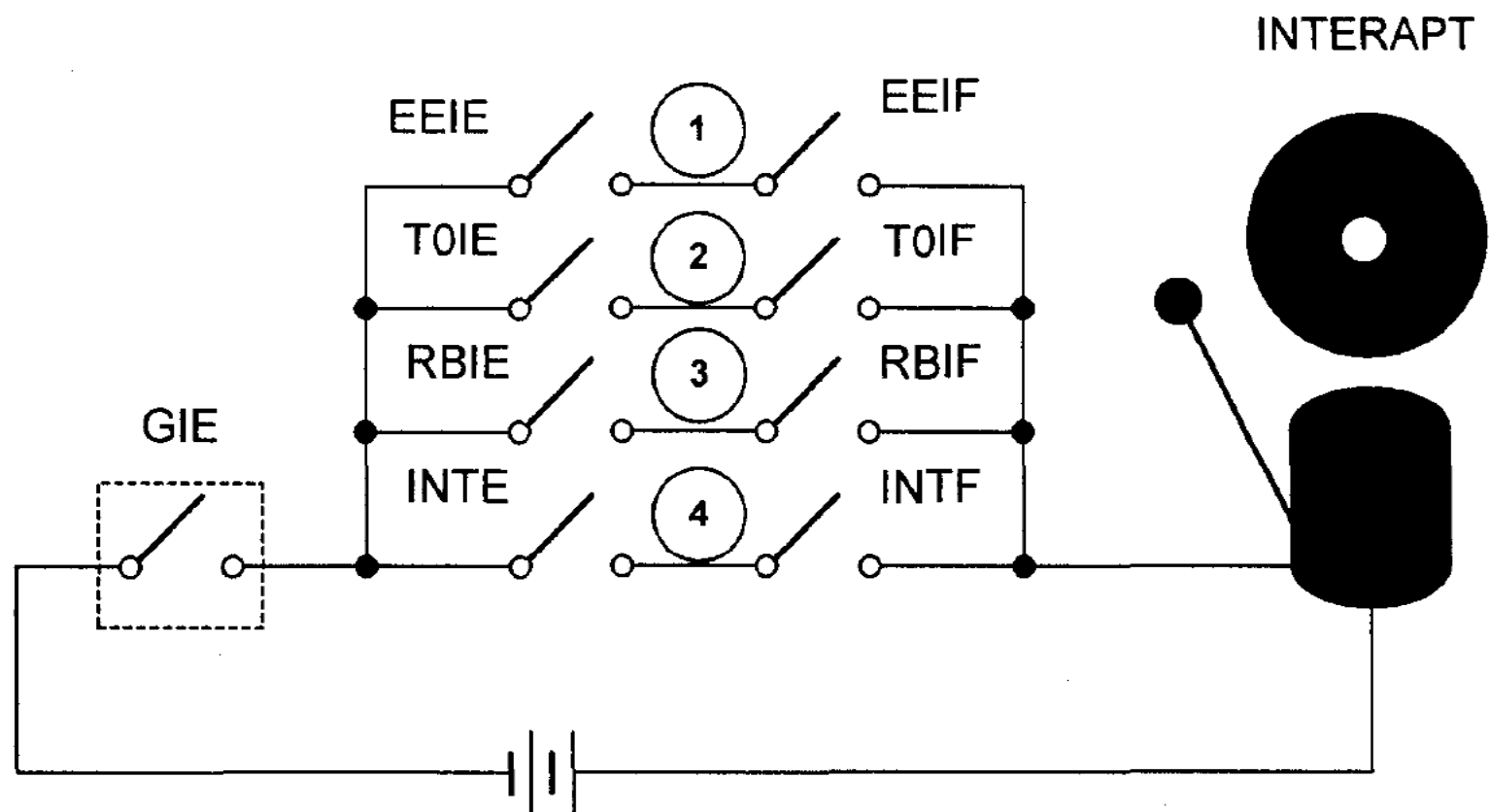
# INTCON registrar (0Bh, 8Bh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	EEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF
bit 7							bit 0

- bit 7 **GIE:** Global Interrupt Enable bit  
1 = Enables all unmasked interrupts  
0 = Disables all interrupts
- bit 6 **EEIE:** EE Write Complete Interrupt Enable bit  
1 = Enables the EE Write Complete interrupts  
0 = Disables the EE Write Complete interrupt
- bit 5 **T0IE:** TMR0 Overflow Interrupt Enable bit  
1 = Enables the TMR0 interrupt  
0 = Disables the TMR0 interrupt
- bit 4 **INTE:** RB0/INT External Interrupt Enable bit  
1 = Enables the RB0/INT external interrupt  
0 = Disables the RB0/INT external interrupt
- bit 3 **RBIE:** RB Port Change Interrupt Enable bit  
1 = Enables the RB port change interrupt  
0 = Disables the RB port change interrupt
- bit 2 **T0IF:** TMR0 Overflow Interrupt Flag bit  
1 = TMR0 register has overflowed (must be cleared in software)  
0 = TMR0 register did not overflow
- bit 1 **INTF:** RB0/INT External Interrupt Flag bit  
1 = The RB0/INT external interrupt occurred (must be cleared in software)  
0 = The RB0/INT external interrupt did not occur
- bit 0 **RBIF:** RB Port Change Interrupt Flag bit  
1 = At least one of the RB7:RB4 pins changed state (must be cleared in software)  
0 = None of the RB7:RB4 pins have changed state

File Address		File Address	
00h	Indirect addr. <sup>(1)</sup>	Indirect addr. <sup>(1)</sup>	80h
01h	TMR0	OPTION_REG	81h
02h	PCL	PCL	82h
03h	STATUS	STATUS	83h
04h	FSR	FSR	84h
05h	PORTA	TRISA	85h
06h	PORTB	TRISB	86h
07h	—	—	87h
08h	EEDATA	EECON1	88h
09h	EEADR	EECON2 <sup>(1)</sup>	89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch			8Ch

# Šema prekida



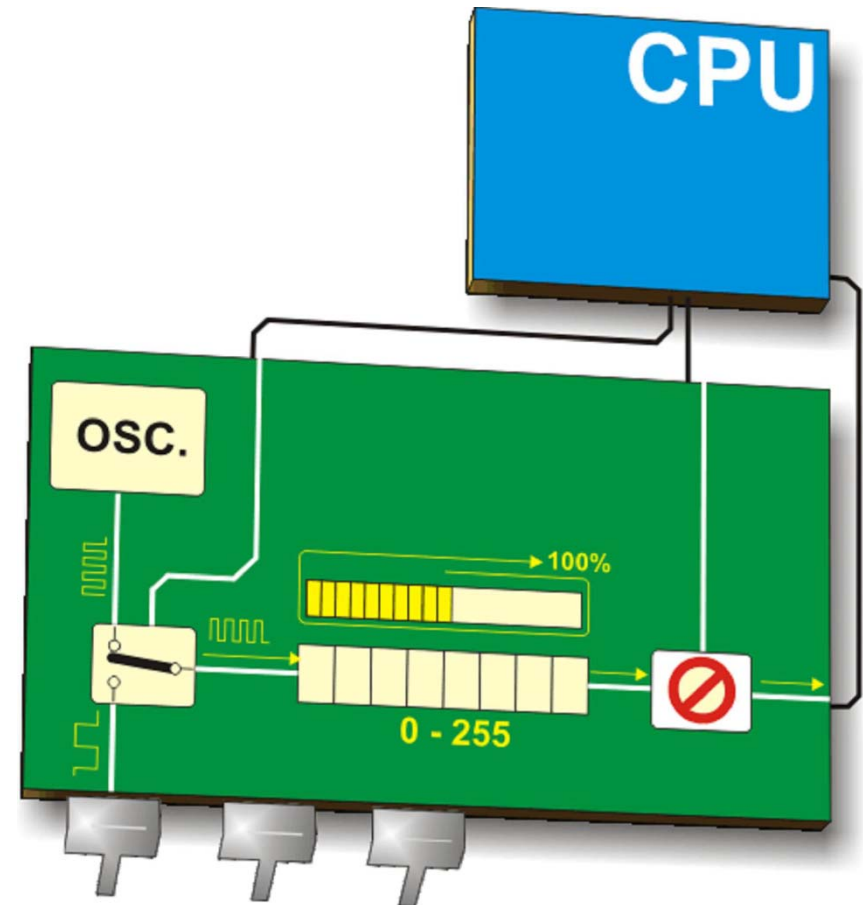


# Obrada prekida

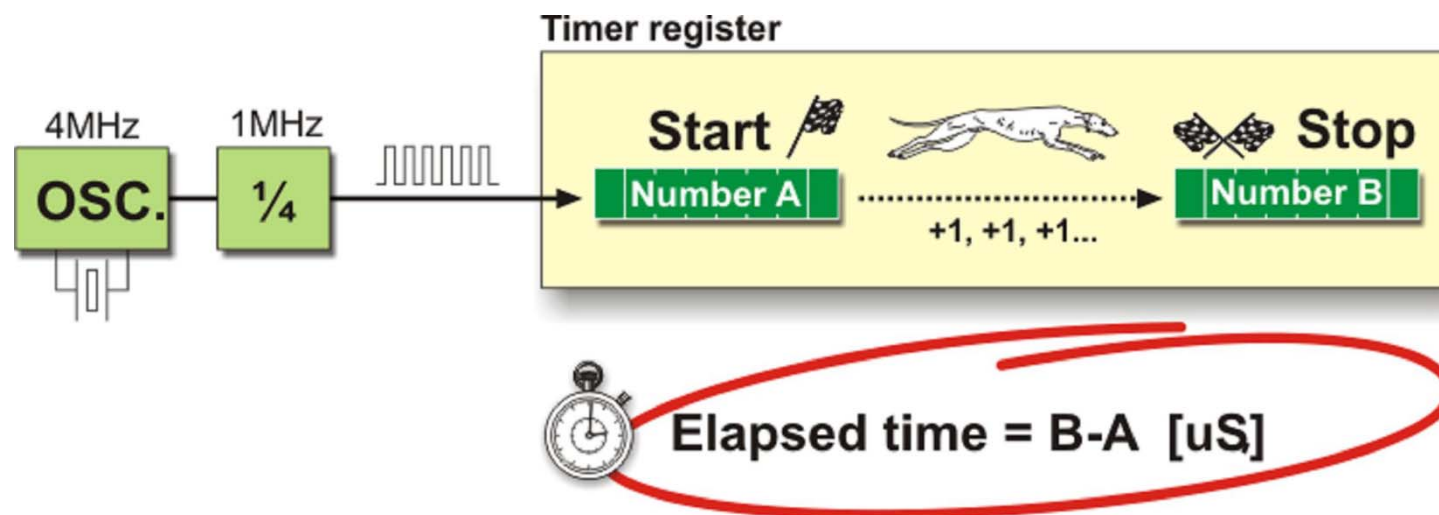
- Postoji više izvora prekida, ali svi oni rezultuju **skokom na adresu 004h**
- Zato se mora detektovati izvor prekida (**proverom INTCON flegova**) i na osnovu toga izabrati ISR
- Kada se odgovori na prekid, **GIE se resetuje**, da bi se sprečili dalji prekidi, povratna adresa (stanje PC) se stavlja na stek, a u PC se upisuje 004h
- Pre izvršenja ISR moraju se sačuvati sadržaji bitnih registara za dalji tok izvršenja (pre svih **W** i **STATUS**)
- Fleg koji označava **tekući prekid** mora se **softverski resetovati** u ISR, da se ne bi dobila beskonačna petlja
- Po završetki ISR restaurirati sadržaje zapamćenih registara

# Tajmeri i brojači

- **Brojač** je uređaj koji beleži broj pojavljivanja nekog događaja (obično broji prelaske sa 0 na 1 ili obrnuta na nekom pinu)
- **Tajmer** meri vremenske intervale (broji unapred) ili odbrojava unazad i definiše kašnjenje (obično broji mašinske cikluse)



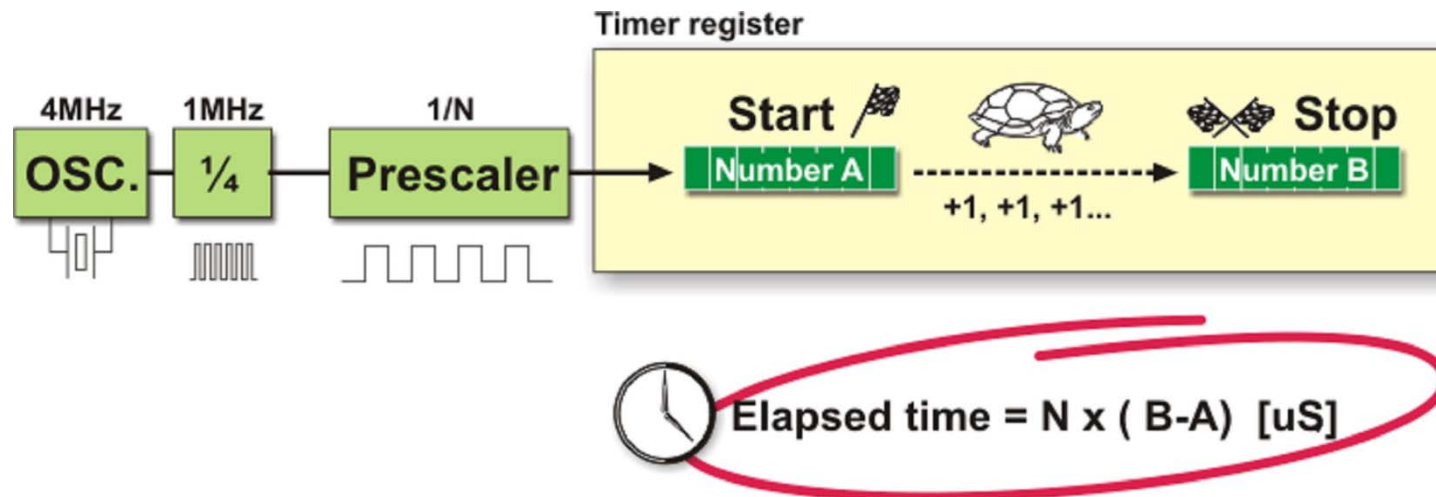
# Osnovno korišćenje tajmera



Zgodno za merenje kratkih vremenskih intervala (do 256 otkucaja takta, ako se koristi 8-bitni registar), ali je merenje vrlo precizno. U datom primeru može se meriti vreme do 256 $\mu$ s, sa preciznošću od 1 $\mu$ s.



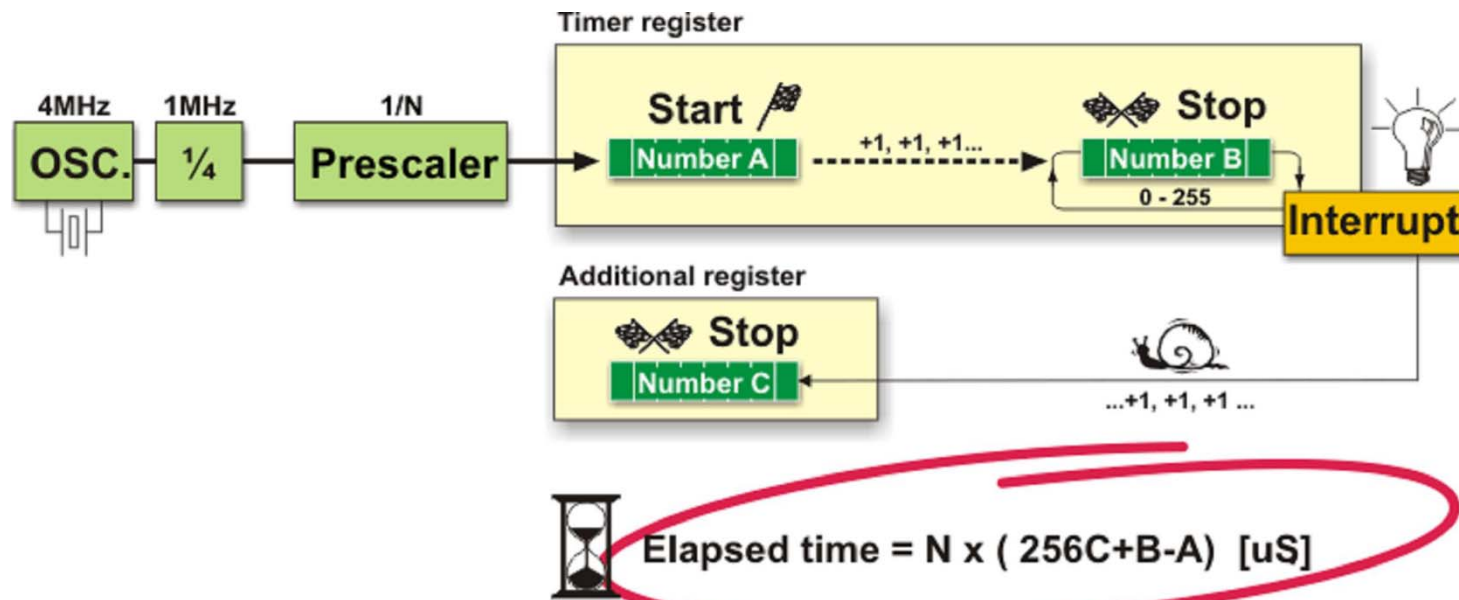
# Korišćenje preskalera za produžavanje perioda



Preskaler je elektronski uređaj koji smanjuje frekvenciju deljenjem nekim predefinisanim faktorom. MCU imaju ugrađene preskalere kojima se može programski upravljati.

Produžava se vreme koje se može meriti, ali se smanjuje preciznost. Npr. za  $N = 256$ , može se meriti period od  $65535\mu\text{s}$  ( $\sim 65.5\text{ms}$ ), ali sa rezolucijom od  $256\mu\text{s}$

# Korišćenje prekida za povećanje i vremena i preciznosti



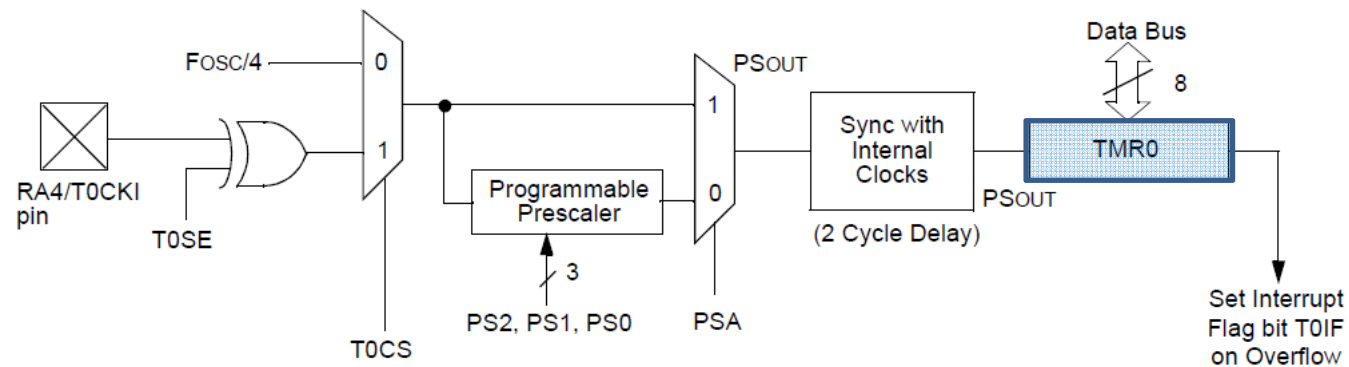
Kada tajmer dostigne maksimalnu vrednost, on se resetuje na 0. To je događaj koji se može signalizirati prekidom. U ISR možemo inkrementirati neki drugi registar, koji će nam služiti kao bajt veće težine brojača.

U našem primeru, za  $N = 1$ , može se meriti period od  $65535\mu s$  ( $\sim 65.5ms$ ), ali sa rezolucijom od  $1\mu s$ , do maksimalno  $16777216\mu s$  ( $\sim 16.77s$ ) sa rezolucijom od  $256\mu s$ , za  $N = 256$ .

# TIMER0

- 8-bitni tajmer/brojač
- Može se čitati i upisivati
- Interni ili eksterni izvor takta
- 8-bitni preskaler koji se softverski postavlja
- Generiše prekid kod prekoračenja (prelaz FFh→00h)

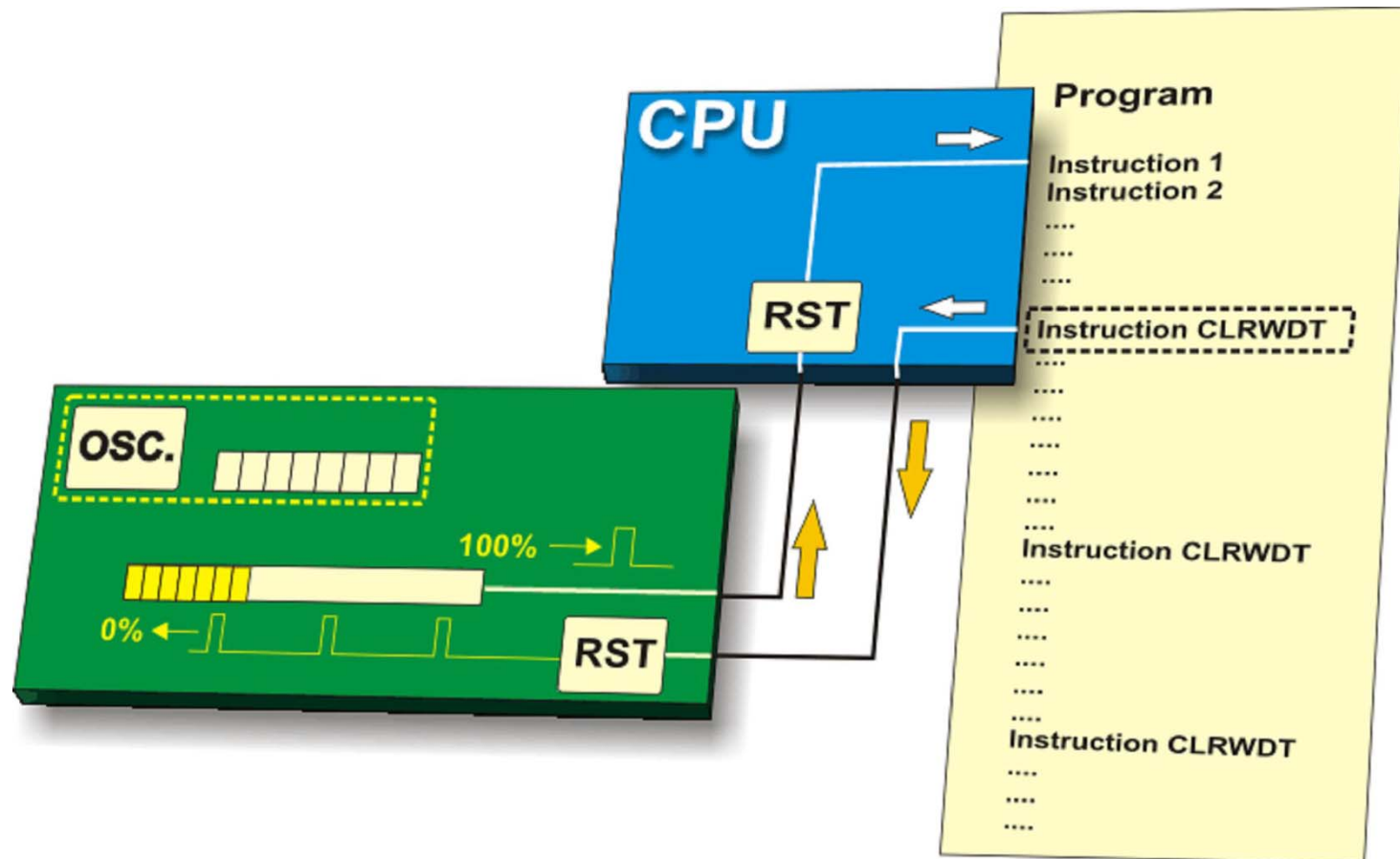
File Address	Indirect addr. <sup>(1)</sup>	Indirect addr. <sup>(1)</sup>	File Address
00h	TMR0	OPTION_REG	80h
01h	PCL	PCL	81h
02h	STATUS	STATUS	82h
03h	FSR	FSR	83h
04h	PORTA	TRISA	84h
05h	PORTB	TRISB	85h
06h	—	—	86h
07h	—	—	87h
08h	EEDATA	EECON1	88h
09h	EEADR	EECON2 <sup>(1)</sup>	89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch			8Ch



**Note** 1: T0CS, T0SE, PSA, PS2:PS0 (OPTION\_REG<5:0>).

2: The prescaler is shared with Watchdog Timer (refer to Figure 5-2 for detailed block diagram).

# Watchdog Timer (WDT)



- ## Configuration Word (2007h)

bit13



# WDT period

- WDT ima nominalni period od **18ms** (varira zavisno od  $V_{DD}$  i temperature)
- Ako je potrebno duže vreme, koristi se preskaler sa faktorom deljena do **128** (WDT do **2.3 s**), koji se postavlja preko OPTION registra
- **CLRWDT** i **SLEEP** instrukcije brišu WDT
- **TO** bit se resetuje (postaje aktivan) kada istekne WDT

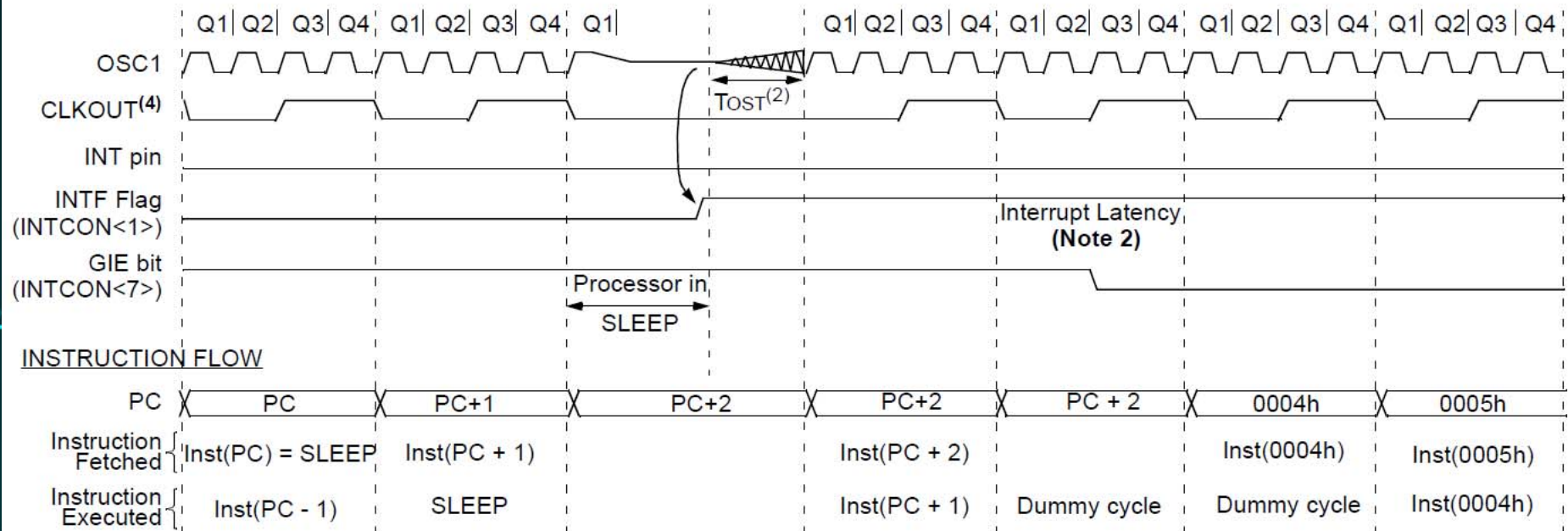
R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0



# SLEEP režim

- U ovaj režim ulazi izvršenjem SLEEP instrukcije
- Ako je omogućen, WDT se resetuje ali nastavlja da radi
- Bitovi u STATUS registru **PD** (Power-down) se resetuje (aktivan) a **TO** (Time-out) setuje (neaktivan)
- U/I portovi zadržavaju stanje koje su imali
- Izlazak iz ovog režima na jedan od događaja
  - Eksterni reset preko MCLR pina
  - Istek WDT
  - Spoljašnji prekid, promena na portu B ili završen upis u EEPROM
- Pri buđenju, ako je GIE=0, izvršenje se nastavlja od instrukcije iza SLEEP, a ako je GIE=1, skače se na adresu 0004h

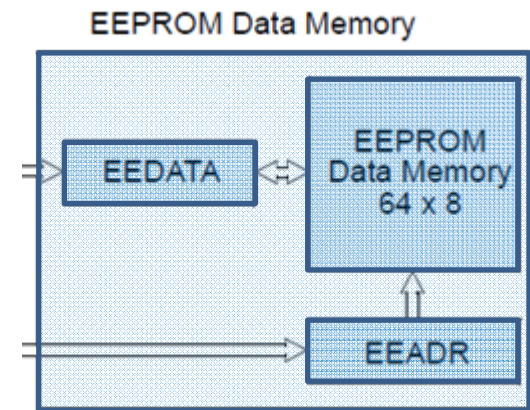
# Izlazak iz SLEEP režima



- Note**
- 1: XT, HS, or LP oscillator mode assumed.
  - 2:  $T_{OST} = 1024T_{OSC}$  (drawing not to scale). This delay will not be there for RC osc mode.
  - 3: GIE = '1' assumed. In this case after wake-up, the processor jumps to the interrupt routine. If GIE = '0', execution will continue in-line.
  - 4: CLKOUT is not available in these osc modes, but shown here for timing reference.

# EEPROM

- EEDATA – sadrži podatak koji treba da se upiše ili pročitani podatak
- EEADR – sadrži adresu lokacije kojoj se pristupa
- EECON1 – sadrži kontrolne bitove
- EECON2 – inicijalizuje upis (mada ne postoji kao fizički registar)
- Veličina – 64B (adrese: 00h-3Fh)
- Upis je upravljn tajmerom na čipu (vreme varira zavisno od napona napajanja i temperature, za 16F84A tipično  $T_{DEW} = 4ms$ , max.  $T_{DEW} = 8ms$ )



File Address		File Address	
00h	Indirect addr. <sup>(1)</sup>	Indirect addr. <sup>(1)</sup>	80h
01h	TMR0	OPTION_REG	81h
02h	PCL	PCL	82h
03h	STATUS	STATUS	83h
04h	FSR	FSR	84h
05h	PORTA	TRISA	85h
06h	PORTB	TRISB	86h
07h	—	—	87h
08h	EEDATA	EECON1	88h
09h	EEADR	EECON2 <sup>(1)</sup>	89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch			8Ch



# EECON1 register (88h)

U-0	U-0	U-0	R/W-0	R/W-x	R/W-0	R/S-0	R/S-0
—	—	—	EEIF	WRERR	WREN	WR	RD
bit 7			bit 0				

bit 7-5 **Unimplemented:** Read as '0'

bit 4 **EEIF:** EEPROM Write Operation Interrupt Flag bit

1 = The write operation completed (must be cleared in software)

0 = The write operation is not complete or has not been started

bit 3 **WRERR:** EEPROM Error Flag bit

1 = A write operation is prematurely terminated

(any MCLR Reset or any WDT Reset during normal operation)

0 = The write operation completed

bit 2 **WREN:** EEPROM Write Enable bit

1 = Allows write cycles

0 = Inhibits write to the EEPROM

bit 1 **WR:** Write Control bit

1 = Initiates a write cycle. The bit is cleared by hardware once write is complete. The WR bit can only be set (not cleared) in software.

0 = Write cycle to the EEPROM is complete

bit 0 **RD:** Read Control bit

1 = Initiates an EEPROM read RD is cleared in hardware. The RD bit can only be set (not cleared) in software.

0 = Does not initiate an EEPROM read

File Address		File Address
00h	Indirect addr. <sup>(1)</sup>	80h
01h	TMR0	OPTION_REG
02h	PCL	PCL
03h	STATUS	STATUS
04h	FSR	FSR
05h	PORTA	TRISA
06h	PORTB	TRISB
07h	—	—
08h	EEDATA	EECON1
09h	EEADR	EECON2 <sup>(1)</sup>
0Ah	PCLATH	PCLATH
0Bh	INTCON	INTCON
0Ch		





# Pristup EEPROM-u

- **Čitanje**
  - Upisati adresu u **EEADR**
  - Postaviti **RD** bit u **EECON1**
  - Pročitati **EEDATA** (već u prvoj sledećoj instrukciji)
- **Upis**
  - Upisati adresu u **EEADR**
  - Upisati podatak u **EEDATA**
  - Zabraniti prekide (zbog sekvence „otključavanja“)
  - Postaviti **WREN** bit u **EECON1** (omogućiti upis)
  - Upisati **55h**, pa **AAh** u **EECON2**, sukcesivno, da bi se „otključao“ upis (ovim se sprečava slučajan upis)
  - Postaviti **WR** bit u **EECON1** (inicirati upis) odmah nakon sekvence „otključavanja“
  - Omogućiti prekide
  - Završetak upisa se signalizira prekidom, tj. postavljanjem bita **EEIF** (*EE Write Complete Interrupt Flag* )
- Dobra programerska praksa nalaže da se nakon upisa proverí da li je upisana vrednost korektna

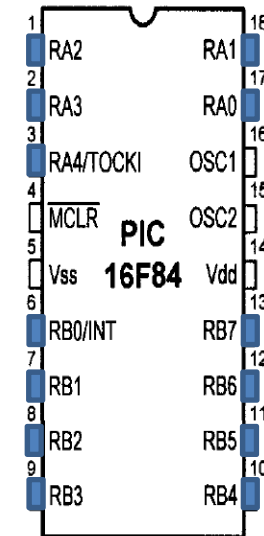


# Zaštita od lažnog upisa

- Da bi se sprečio „lažni“ upis podataka u EEPROM zbog variranja i naglih skokova napona ili lošeg rada softvera, koriste se brojni mehanizmi:
  - Pri uključivanju briše se **WREN** bit
  - Tokom trajanja **Power-up** tajmera (72ms) ne može se izvršiti upis u EEPROM
  - **WREN** mora biti postavljen pre početka sekvence otključavanja
  - **Sekvenca „otključavanja“** mora da prethodi upisu svakog bajta (upis 55h u EECON2, UPIS AAh u EECON2, postavljanje WR u EECON1)

# U/I registri

- PIC16F84A ima dva U/I porta (A i B)
- **Port A**
  - 5-bitni
  - Smer pinova se određuje registrom **TRISA** (1-ulazni smer, 0-izlazni smer)
  - Vrednost se čita/upisuje preko registra **PORTA**
- **Port B**
  - 8-bitni
  - Smer pinova se određuje registrom **TRISB** (1-ulazni smer, 0-izlazni smer)
  - Vrednost se čita/upisuje preko registra **PORTB**



File Address		File Address
00h	Indirect addr. <sup>(1)</sup>	Indirect addr. <sup>(1)</sup> 80h
01h	TMR0	OPTION_REG 81h
02h	PCL	PCL 82h
03h	STATUS	STATUS 83h
04h	FSR	FSR 84h
05h	PORTA	TRISA 85h
06h	PORTB	TRISB 86h
07h	—	— 87h
08h	EEDATA	EECON1 88h
09h	EEADR	EECON2 <sup>(1)</sup> 89h
0Ah	PCLATH	PCLATH 8Ah
0Bh	INTCON	INTCON 8Bh
0Ch		8Ch

## Registri vezani za port A

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-on Reset	Value on all other RESETS
05h	PORTA	—	—	—	RA4/T0CKI	RA3	RA2	RA1	RA0	---x xxxx	---u uuuu
85h	TRISA	—	—	—	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	---1 1111	---1 1111

## Registri vezani za port B

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-on Reset	Value on all other RESETS
06h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0/INT	xxxx xxxx	uuuu uuuu
86h	TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
81h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
0Bh,8Bh	INTCON	GIE	EEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u



# Port B dodatne funkcije

- Svaki pin porta B ima slabi interni pull-up
- Za sve pinove se pull-up uključuje jednim bitom (RBPU) u OPTION registru
- Pull-up se automatski isključuje kada se pin prebaci u izlazni mod
- Pull-up se onemogućuje pri power-on resetu
- Prekidi su omogućeni samo ako su pinovi (RB7:RB4) u ulaznom modu
- Stanje ulaznih pinova (iz opsega RB7:RB4) se poredi sa prethodno „lečovanom“ vrednošću, razlike se vode na logičko ILI kolo koje generiše *RB Port Change Interrupt* i postavlja fleg RBIF (u INTCON registru)
- Prekid može da probudi uređaj iz SLEEP režima
- Prekid se poništava čitanjem sa porta B ili brisanjem RBIF flega
- Prekid u slučaju promene na portu B, uz softersko upravljiv pull-up na 4 pina omogućuje jednostavan interfejs prema tastaturi i mogućnost „buđenja“ na pritisak tastera



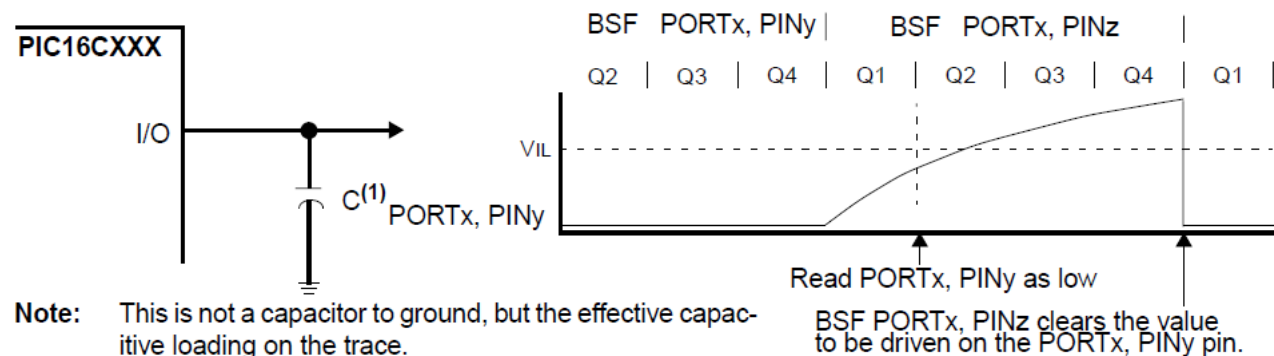


# Problem postavljanja U/I pinova

- Treba biti obazriv sa postavljanjem i brisanjem pojedinih bitova porta, ako se istovremeno neki pinovi koriste kao ulazni, a neki kao izlazni
- Instrukcije za postavljanje/brisanje jednog bita (BCF, BSF) zapravo prvo preuzimaju kompletan sadržaj porta (kopiraju u registar), zatim menjaju dati bit, pa vraćaju kompletan registar na port (*read-modify-write*)
- To može potpuno izbrisati (ulazne) bitove koji su u međuvremenu postavljeni (na osnovu ulaznog signala)

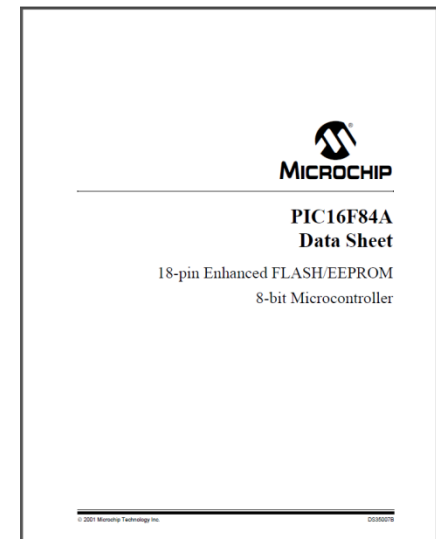
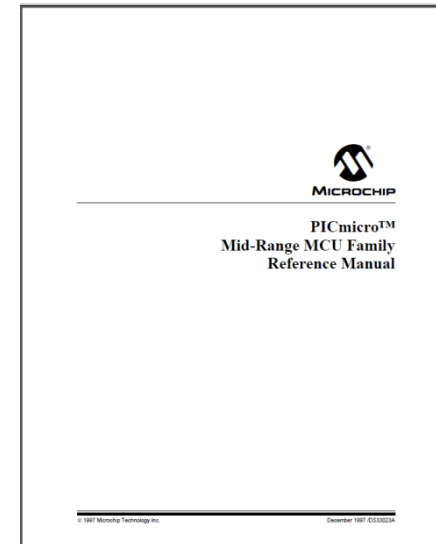
# Problem sukcesivnih operacija

- Upis na port se dešava u poslednjoj fazi ciklusa instrukcije (Q4), dok se čitanje sa porta obavlja u prvoj (Q1)
- Šta se dešava ako čitanje sledi odmah iza upisa?
- Mora se dozvoliti da se napon stabilizuje, inače se može očitati prethodna vrednost
- Zbog spoljašnje kapacitivnosti, pogotovu na višim frekvencijama, javlja se dodatno kašnjenje
- Zato se u takvim situacijama preporučuje razdvajanje ovih instrukcija jednom (ili više) NOP naredbom, ili nekom drugom naredbom koja ne pristupa datom portu



# Literatura

- PICmicro™ Mid-Range MCU Family Reference Manual, DS33023A, 1997,  
<http://ww1.microchip.com/downloads/en/devicedoc/33023a.pdf>
- PIC16F84A Data Sheet, 18-pin Enhanced FLASH/EEPROM 8-bit Microcontroller, 35007b, 2001,  
<https://ww1.microchip.com/downloads/en/devicedoc/35007b.pdf>



# Literatura

- Nebojša Matic, Dragan Andrić, **PIC mikrokontroleri**, Agencija APC, 2000.
- Milan Verle, **PIC mikrokontroleri**, Mikroknjiga, 2007, ISBN: 978-86-84417-14-7





# Literatura

- 8-Bit Microcontroller Summary,  
<https://microchipdeveloper.com/8bit:summary>
- External RC Oscillator,  
<https://microchipdeveloper.com/8bit:extrc>