

3. Organizacija pomnilnika

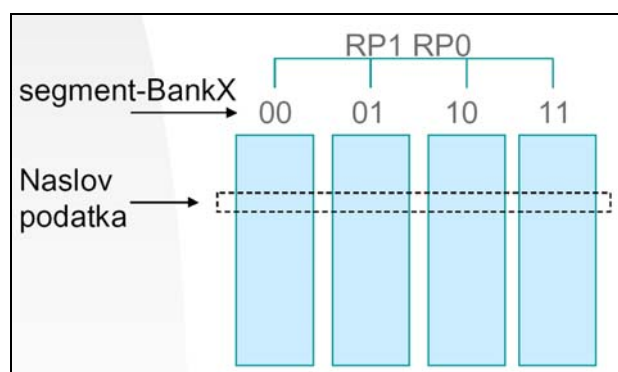
*Spoznali boste model in organizacijo podatkovnega pomnilnika (segmenti) in model programskega pomnilnika ter v zvezi s tem najpomembnejši datotečni register **STATUS**. Naučili se boste ločevati med podatkovnim pomnilnikom (RAM) in programskim pomnilnikom (FLASH ROM).*

3.1. Podatkovni pomnilnik (namenski in splošni registri)

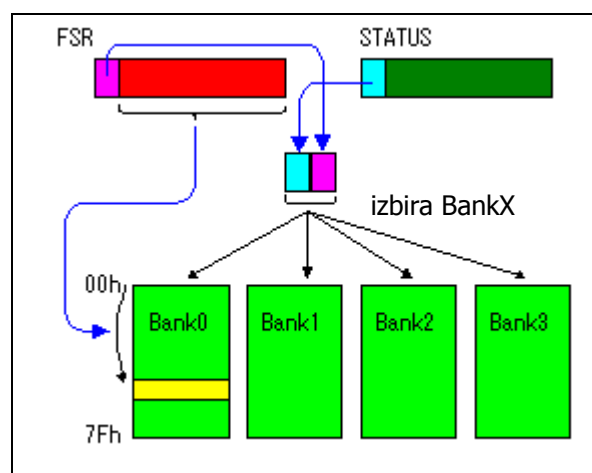
Podatkovni pomnilnik (Slika 3-1) mikrokrmilnikov PIC16F87x [6][8][16] je **razdeljen na štiri segmente** (Bank0, Bank1, Bank2, Bank3). Vsak od segmentov obsega do **128 zlogov** (8-bitnih besed) ali lokacij – datotečni registri (**File registers**) z naslovi od 0 do 7Fh (Slika 3-2).

Datotečne registre SFR (Slika 3-3), ki zasedajo začetni del (pri prvih dveh segmentih na naslovih: 0 do 1Fh), imenujemo tudi **namenski funkcijski registri** (angl.: **Special Function Registers**), ker so **namenjeni posebnim funkcijam**: sistemski registri in registri vhodno/izhodnih vmesnikov. Na voljo je nekaj **nad 50 različnih SFR** registrov, vsak pa ima specifično (rezervirano) ime (npr.: **PORTB**), katero lahko uporabimo kot **operand pri programiranju**. Nekateri (najpomembnejši) registri (npr.: **STATUS**, **FSR**) se pojavljajo v vseh štirih segmentih.

Datotečne registre GPR (Slika 3-3), ki zasedajo drugi del (pri prvih dveh segmentih na naslovih: 20h do 7Fh) imenujemo tudi **splošno uporabni registri** (angl.: **General Purpose Registers**), ker so **namenjeni splošni funkciji**, kot je shranjevanje spremenljivk. V vseh štirih segmentih je pri PIC16F876/877 na voljo **368 RAM pomnilniških lokacij**, pri čemer se zadnjih 16 lokacij (naslovi od 70h do 7Fh) pojavi v vseh štirih segmentih.



Slika 3-1: Segmentacija pomnilnika



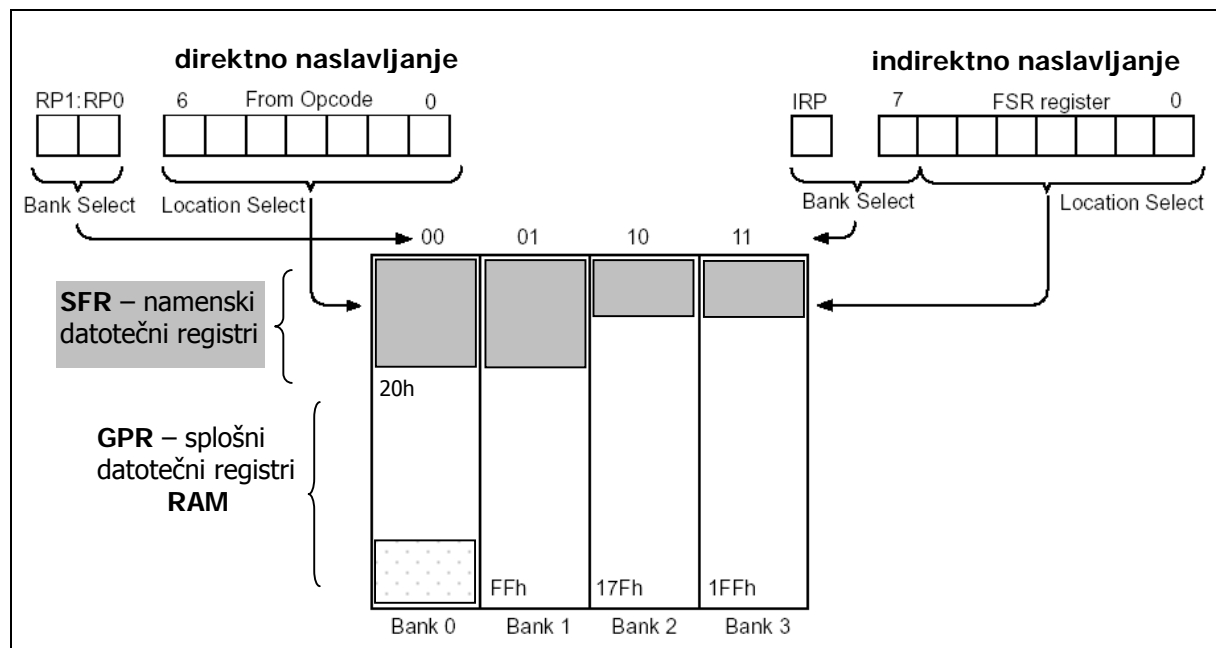
Slika 3-2: Izbira registrov v segmentih

Pri »**direktnem**« **načinu naslavljanja** (Slika 3-1, Slika 3-2, Slika 3-3) imamo programski dostop do posameznega **SFR** registra preko imena **f** registra, dostop do **GPR** registra pa preko izbire naslova (med 1Fh in 7Fh). **Pred tem moramo izbrati**

ustrezni segment, kar dosežemo z vpisom 2-bitne informacije v register **STATUS** (bita RP1, RP0).

Pri »indirektnem« načinu naslavljanja (Slika 3-2, Slika 3-3) imamo enoten programski dostop do **SFR** in **GPR** registrov preko izbire naslova (med 0Fh in FFh) s pomočjo »indeksnega registra« **FSR**. **Pred tem moramo izbrati ustrezni par segmentov**, kar dosežemo z vpisom 1-bitne informacije v register **STATUS** (bit z oznako IRP).

Za (manj večše) uporabnike je na začetku vsekakor **priporočljivo uporabljati »direktni« način naslavljanja**.



Slika 3-3: Nazornejši prikaz izbire segmentov in naslavljanja registrov

3.1.1. Podroben prikaz segmentne organizacije podatkovnega pomnilnika

naslov registra		naslov registra		naslov registra		naslov registra	
Indirect addr. ^(*)	00h	Indirect addr. ^(*)	80h	Indirect addr. ^(*)	100h	Indirect addr. ^(*)	180h
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTA	05h	TRISA	85h		105h		185h
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h
PORTC	07h	TRISC	87h		107h		187h
PORTD ⁽¹⁾	08h	TRISD ⁽¹⁾	88h		108h		188h
PORTE ⁽¹⁾	09h	TRISE ⁽¹⁾	89h		109h		189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
PIR1	0Ch	PIE1	8Ch	EEDATA	10Ch	EECON1	18Ch
PIR2	0Dh	PIE2	8Dh	EEADR	10Dh	EECON2	18Dh
TMR1L	0Eh	PCON	8Eh	EEDATH	10Eh	Reserved ⁽²⁾	18Eh
TMR1H	0Fh		8Fh	EEADRH	10Fh	Reserved ⁽²⁾	18Fh
T1CON	10h		90h		110h		190h
TMR2	11h	SSPCON2	91h		111h		191h
T2CON	12h	PR2	92h		112h		192h
SSPBUF	13h	SSPADD	93h		113h		193h
SSPCON	14h	SSPSTAT	94h		114h		194h
CCPR1L	15h		95h		115h		195h
CCPR1H	16h		96h		116h		196h
CCP1CON	17h		97h	General Purpose Register 16 Bytes	117h	General Purpose Register 16 Bytes	197h
RCSTA	18h	TXSTA	98h		118h		198h
TXREG	19h	SPBRG	99h		119h		199h
RCREG	1Ah		9Ah		11Ah		19Ah
CCPR2L	1Bh		9Bh		11Bh		19Bh
CCPR2H	1Ch		9Ch		11Ch		19Ch
CCP2CON	1Dh		9Dh		11Dh		19Dh
ADRESH	1Eh	ADRESL	9Eh		11Eh		19Eh
ADCON0	1Fh	ADCON1	9Fh		11Fh		19Fh
	20h		A0h		120h		1A0h
General Purpose Register 96 Bytes		General Purpose Register 80 Bytes		General Purpose Register 80 Bytes		General Purpose Register 80 Bytes	
	7Fh	accesses 70h-7Fh	EFh F0h	accesses 70h-7Fh	16Fh 170h	accesses 70h - 7Fh	1EFh 1F0h
Bank 0		Bank 1	FFh	Bank 2	17Fh	Bank 3	1FFh

neuporabljene lokacije, čitajo se kot '0'
 * ni fizični register
 Opomba 1: ti registri se ne uporabljajo v PIC16F876/3 ampak v PIC16F877/4
 Opomba 2: ti registri so rezervirani, vrednost je 0

Tabela 3-1: Razporeditev registrov po pomnilniških segmentih

3.1.2. Register STATUS

Register **STATUS** vsebuje bitni status aritmetične logične enote (ALU), resetni status in kontrolne **bite za izbiro segmenta (banke)** v pomnilniku. Statusni register lahko uporabimo kot cilj v katerem koli ukazu, če je ukaz takšen, da vpliva na bite Z, C ali DC; le-teh ni mogoče spreminjati, bitov \overline{TO} in \overline{PD} pa ni mogoče prepisati.

Naslov	Ime	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Vrednost pri: POR,BOR	Vrednost pri vseh drugih resetih
03h	STATUS	IRP	RP1	RP0	\overline{TO}	\overline{PD}	Z	DC	C	0001 1xxx	000q quuu

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	TO	PD	Z	DC	C
bit 7						bit 0	

Legenda: R = omogočeno branje bita, W = omogočeno pisanje v bit, U = neuporabljn bit, beri kot '0', -n = vrednost ob POR resetu

bit 7: **IRP**: Bit za izbiro bank registrov (Register Bank Select bit, uporabljen za posredno naslavljanje)

1 = Bank 2, 3 (100h - 1FFh)

0 = Bank 0, 1 (00h - FFh)

bit 6-5: **RP1:RP0**: Bit za izbiro banke registrov (Register Bank Select bits, uporabljen za neposredno naslavljanje)

00 = **Bank 0 (00h - 7Fh)**

01 = **Bank 1 (80h - FFh)**

10 = Bank 2 (100h - 17Fh)

11 = Bank 3 (180h - 1FFh)

Vsaka segment (Bank) ima maksimalno 128 zlogov (bytov).

bit 4: \overline{TO} : Bit za prekoračitev časa (Time-out bit)

1 = Po vklopu napajanja, ukazu CLRWDT ali SLEEP

0 = Pojavila se je prekoračitev časa WDT

bit 3: \overline{PD} : Bit za izklop napajanja (Power-down bit)

1 = Po vklopu ali z ukazom CLRWDT

0 = Z izvedbo ukaza SLEEP

bit 2: **Z**: Ničelni bit (Zero bit)

1 = rezultat aritmetične ali logične operacije je nič

0 = rezultat aritmetične ali logične operacije ni nič

bit 1: **DC**: Bit za prenos med nibbloma (Digit Carry/borrow bit, ukazi ADDWF, ADDLW, SUBLW, SUBWF)

(za borrow je polariteta obrnjena)

1 = Pojavil se je prenos s četrtega bita.

0 = Prenos s četrtega bita se ni pojavil.

bit 0: **C**: Bit za prenos/izposajo (Carry/borrow bit, ukazi ADDWF, ADDLW, SUBLW, SUBWF)

1 = Prišlo je do prenosa iz najvišjega bita (MSB)

0 = Ni prišlo do prenosa iz najvišjega bita (MSB)

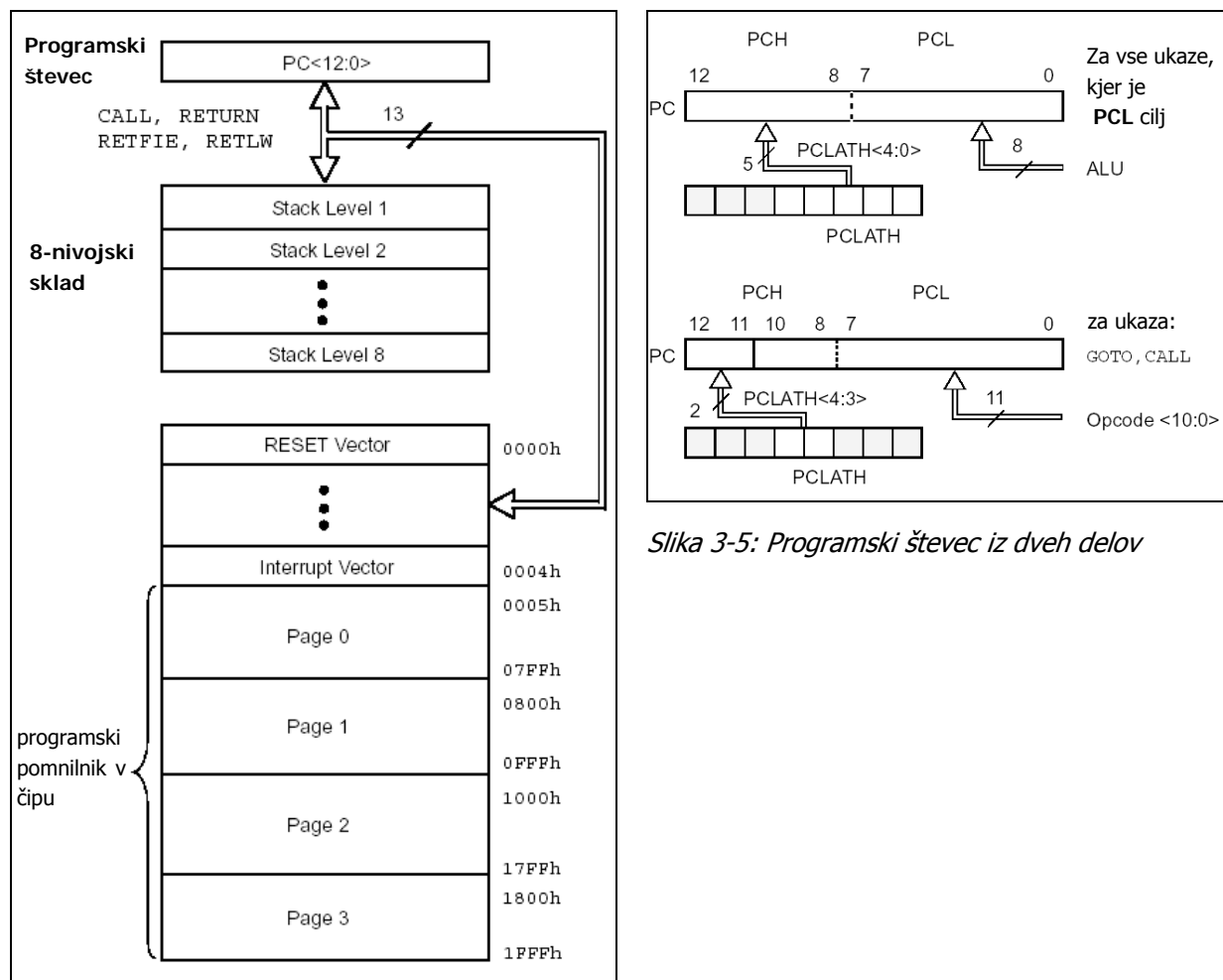
Opomba: Za borrow je polariteta obrnjena. Odštevanje je izvedeno s prištevanjem dvojiškega komplementa drugega operanda. Za ukaze za rotacije (RRF, RLF) se ta bit naloži z bodisi najvišjim ali najnižjim bitom registra, ki ga rotiramo.

3.2. Programski pomnilnik

Programski pomnilnik (Slika 3-4) v mikrokrmilniku PIC16F876/877 [6][8][16] vsebuje **8k besed** (14-bitne celice FLASH ROM), kamor se **naloži** (programira, »zapeče«) zaporedje **strojnih ukazov - program**. Pomnilnik je sicer razdeljen na segmente (strani, angl.: Page) po 2k, vendar to ne predstavlja večjih težav, razen v primeru, ko se izvajajo skoki izven segmenta.

Za vsak program je pomembno, da se **prvi ukaz nahaja na začetnem naslovu 0 (RESET Vector)**, kajti procesor začne ob vklopu napajanja (Reset) izvajati program prav iz začetne lokacije.

Pomemben je še **prekinitveni vektor (Interrupt Vector)**, ki se nahaja na naslovu 4. Če so prekinitve omogočene, se mora na navedenem naslovu nahajati **1. ukaz prekinitvenega strežnega programa** (angl.: Interrupt Service Routine).



Slika 3-5: Programski števec iz dveh delov

Slika 3-4: Mapa programskega pomnilnika in sklada

Sklad (angl.: Stack) procesne enote PIC mikrokontrolerov je posebnost, kajti na voljo je le 8 nivojev (angl.: Stack Level) za shranjevanje trenutne vrednosti (13 bitov) programskega števec **PC** (Program Counter) ob **skokih v podprograme** ali ob **prekinitvenih zahtevah**.

Programski števec (PC) je sicer 13-bitni, vendar se v 8-bitnih **SFR** registrih podatkovnega pomnilnika hrani ločeno višjih 5 bitov (**PCH**) in nižjih 8 bitov (**PCL**). Kot pomožni register za polnjenje **PCH** (Slika 3-5) služi »zadrževalni register« **PCLATH**.

Za razumevanje tematike je potrebno tudi predznanje iz osnov digitalne elektronike in gradnikov mikroprocesorskih sistemov (pomnilniški elementi), kar je na voljo med drugim v [3]. Dobri opisi zgradbe in opisa pomnilniškega modela PIC mikrokontrolerov so tudi v [2] in [6] ter na spletnem portalu: http://www.interq.or.jp/japan/seinou/e_pic.htm.

Vprašanja za utrjevanje:

Napišite zaporedje ukazov za izbiro segmenta Bank1

Rešitev:

```
BCF STATUS,RP1 ; 0 → RP1 (brisi bit 6 v reg. STATUS)
BSF STATUS,RP0 ; 1 → RP0 (postavi bit 5 v reg. STATUS)
```

1. *Napišite zaporedje ukazov za izbiro segmenta Bank2*
2. *Kakšen pomen imajo zastavice v registru STATUS ?*
3. *Kateri pomnilniški segment (Bank0, 1, 2, 3) se največ uporablja ?*
4. *Koliko bitne so celice v programskem pomnilniku ?*
5. *Koliko nivojev ima sklad (stack) ?*
6. *S katerega naslova v programskem pomnilniku se začne izvajati program ?*
7. *Kateri datotečni registri se pojavijo v vseh štirih segmentih ?*