

Jednočipový mikropočítač

PIC16F87/88

Překlad originální dokumentace



Ing. Vladimír Čebiš

Materiál vznikl v rámci projektu MŠMT – Podpora projektování elektronických systémů s mikroprocesory v českém jazyce na středních školách, který je řešen na Střední odborné škole a Středním odborném učilišti – COP Sezimovo Ústí. Je možné ho využít pro výuku odborných předmětů na školách v ČR.

Zvláštní poděkování patří firmě Microchip, Arizona, Chandler, USA a přímému mezinárodnímu zastoupení firmy Microchip pro ČR a SR – firmě MICRORISC s.r.o. Jičín za podporu uvedeného projektu a podporu výuky mikroprocesorové techniky na škole.

Symbolika textu

V textu je pomocí různého typu písma odlišen charakter terminologie.

Slova psaná neproporcionálním písmem, zde **Courier**, značí že jde o instrukci Assembleru (**MOVWF**).

Slova, která jsou vyznačena VELKÝM písmem označují názvy registrů nebo jde o zkratkové slovo (paměť RAM).

Pojmenování, která se vztahují ke konkrétnímu předmětu, jsou psána s Velkým počátečním písmenem (čítač Watchdog, režim Sleep).

Adresování X <3:0> odkazuje na bity 3 až 0 registru X.

Jednoduchou uvozovkou je označena konkrétní hodnota, např. „0“ – nulování bitu.

Ochrana duševního vlastnictví

Tento dokument je určen k nekomerčnímu využití. Veškeré komerční šíření, pořizování kopií nebo přebírání jakékoli části textu je bez předchozího souhlasu zakázáno.

Informace jsou zveřejněny bez ohledu na patenty třetích osob. Nároky na odškodnění na základě změn, chyb nebo vynechání jsou vyloučeny.

Veškeré použité registrované, obchodní a ochranné známky jsou majetkem jejich autorů nebo vlastníků a uvedením těchto známek nejsou zpochybněna autorská práva vyplývající z vlastnictví uvedených známek.

Závaznost informací

Aktuálnost a správnost informací vychází z publikovaného katalogu společnosti Microchip DS30487C, PIC16F87/88 Data Sheet, vydané v roce 2005 společností Microchip.

V případě nesrovnalostí uvedených dat vyhledejte aktuální katalog společnosti Microchip nebo změnové listy – errata (informaci najdete na konci dokumentu).

Veškeré uvedené informace mají charakter doporučení a mohou být pozdějšími revizemi uvedeného katalogu změněny nebo doplněny.

Text neprošel jazykovou korekturou.

PIC16F87/88

8-bitový jednočipový mikropočítač s 18/20/28 vývody s technologií nanoWatt

Nízká spotřeba:

- režimy s nízkou spotřebou
 - normální režim: takt RC, 76 μ A, 1 MHz, 2 V
 - RC_RUN: 7 μ A, 31,25 kHz, 2 V
 - SEC_RUN: 9 μ A, 32 kHz, 2 V
 - Sleep: 0,1 μ A, 2 V
- oscilátor Timer1: 1,8 μ A, 32 kHz, 2 V
- Časovač Watchdog: 2,2 μ A, 2 V
- Dvou rychlostní taktovací oscilátor pro rozběh

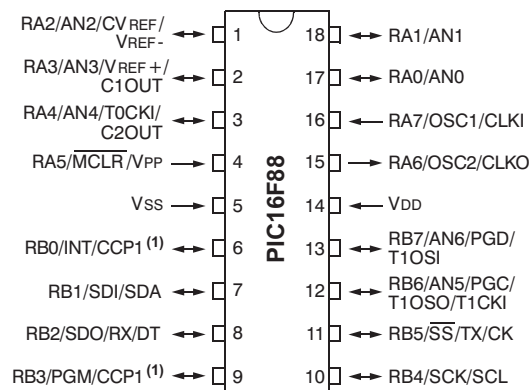
Taktovací oscilátor:

- Tři typy krystalových oscilátorů:
 - LP, XT, HS: do 20 MHz
- Dva režimy externího RC oscilátoru
- Externí zdroj taktování:
 - ECIO: do 20 MHz
- Modul interního oscilátoru
 - 8 uživatelsky volitelných frekvencí:
31 kHz, 125 kHz, 250 kHz, 500 kHz, 1 MHz,
- 2 MHz, 4 MHz, 8 MHz

Možnosti periférií mikropočítače:

- modul PWM s volbou záchytného nebo komparačního režimu
 - rozdílení v záchytném režimu je 16 bitů, nejvýše 12,5 ns
 - rozdílení v komparačním režimu je 16 bitů, nejvýše 200 ns
 - nejvyšší rozdíl pulsní šířkové modulace PWM je 10 bitů
- 10-bitový A/D převodník s nejvýše sedmi vstupy
- Synchronní sériový port s možností SPI™ (Master/Slave) nebo I²C™ (Slave)
- Univerzální asynchronní port AUSART/SCI s možností 9-bitové komunikace
 - možnost použití rozhraní RS232C s vnitřním taktovacím oscilátorem
- Dvojice analogových komparátorů:
 - nastavitelná vnitřní napěťová reference
 - vstupy komparátorů lze přepínat mezi vývody a vnitřní napěťovou referencí
 - výstupy komparátorů lze připojit na vývody součástky

Zapojení vývodů:



Poznámka 1: Vývod CCP1 je definován pomocí bitu CCPMX v konfiguračním registru č. 1

Speciální vlastnosti mikropočítače:

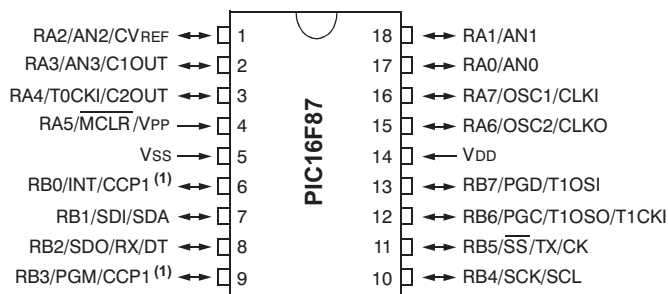
- Počet zápisů do paměti programu typu Flash: 100.000
- Počet zápisů do paměti dat typu EEPROM: 1.000.000
- Doba uchování dat v paměti EEPROM: > 40 let
- Programování v aplikaci ICSP™ (s využitím dvou vývodů)
- Možnost programového čtení a zápisu v paměti programu
- Programování nízkým napětím (LVP)
- Podpora ladění v aplikaci ICD
- Časovač Watchdog (WDT)
 - nastavitelná doba přetečení od 1 ms do 268 s
- Rozsah pracovního napětí: 2,0 V – 5,5 V

Mikropočítač	Paměť programu		Paměť dat		Vstupně výstupních vývodů (I/O)	Vstupů AD převodníku	Modulů CCP (PWM)	AUSART	Komparátorů	SSP	Čítače 8/16 bitů
	Bajtů	Instrukcí	RWM (bajtů)	EEPROM (bajtů)							
PIC16F87	7168	4096	368	256	16	N/A	1	ANO	2	ANO	2/1
PIC16F88	7168	4096	368	256	16	1	1	ANO	2	ANO	2/1

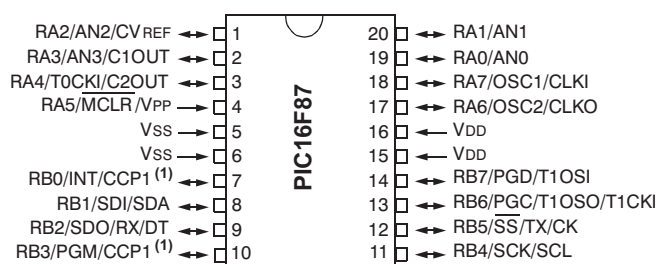
PIC16F87/88

Zapojení vývodů

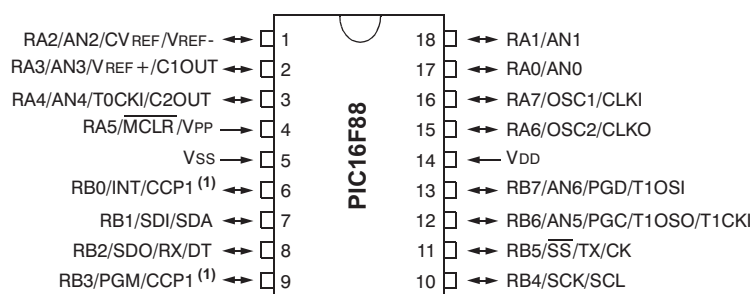
18-Pin PDIP, SOIC



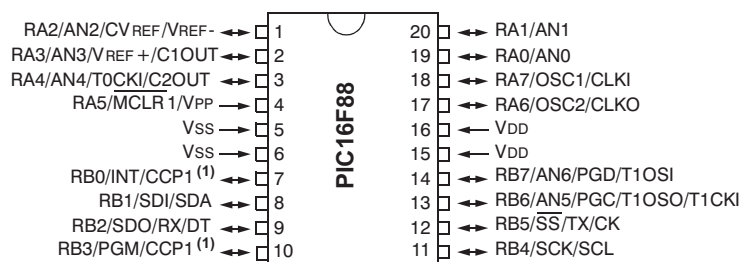
20-Pin SSOP



18-Pin PDIP, SOIC



20-Pin SSOP

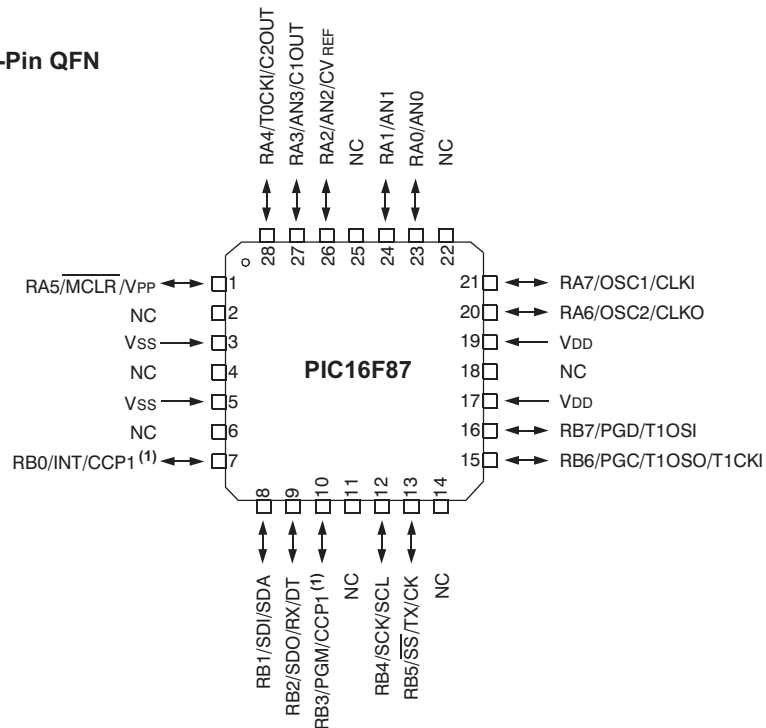


Poznámka 1: Vývod CCP1 je definován pomocí bitu CCPMX v konfiguračním registru č. 1

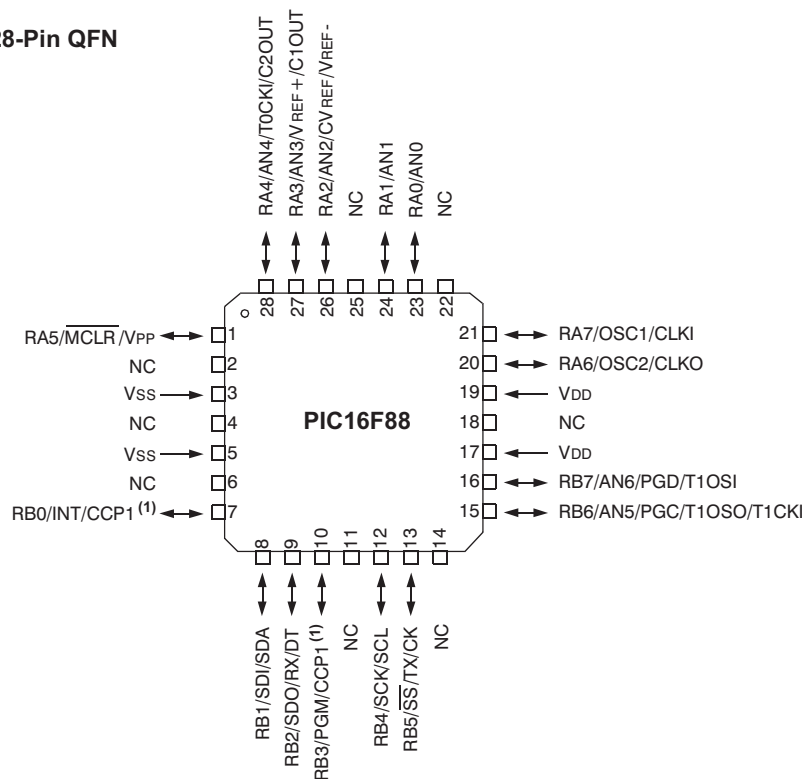
PIC16F87/88

Zapojení vývodů (pokračování)

28-Pin QFN



28-Pin QFN



Poznámka 1: Vývod CCP1 je definován pomocí bitu CCPMX v konfiguračním registru č. 1

PIC16F87/88

Obsah

1	Celkový popis	7
2	Organizace paměťového prostoru	13
3	Paměť dat typu EEPROM a paměť programu typu Flash	29
4	Konfigurace taktovacího oscilátoru	39
5	Vstupně výstupní brány	53
6	Modul čítače/časovače Timer0	69
7	Modul čítače/časovače Timer1	73
8	Modul čítače/časovače Timer2	81
9	Modul CCP (Capture/Compare/PWM)	83
10	Modul synchronního sériového portu (SSP)	89
11	Modul AUSART	99
12	Modul AD převodníku	115
13	Modul komparátoru	123
14	Modul napěťové reference	129
15	Speciální vlastnosti mikropočítače	131
16	Instrukční sada	153
17	Vývojové prostředky	163
18	Elektrické charakteristiky	169
19	Grafy	199
20	Pouzdra	213
Značení na pouzdře		219

PIC16F87/88

Poznámky:

1 CELKOVÝ POPIS

Tato dokumentace obsahuje popis následujících mikropočítačů: PIC16F87, PIC16F88

Další informace o uvedených mikropočítačích lze získat v dokumentaci s označením DS33023 s názvem *PICmicro® Mid-Range Reference Manual*, kterou lze získat na internetových stránkách výrobce.

Mikropočítače PIC16F87 a PIC16F88 navazují na řadu 8-bitových mikropočítačů PIC16Fxxx. Blokové schéma mikropočítačů je uvedeno na obrázku 1-1 a 1-2. Ve srovnání s dosavadními produkty mají uvedené mikropočítače následující vylepšení:

- Režimy s nízkou proudovou spotřebou: Konfigurace RC_RUN umožňuje taktování mikropočítače vnitřním oscilátorem, zatímco konfigurace SEC_RUN umožňuje taktování pomocí úsporného oscilátoru Timer1, viz **kapitulu 4.7**.
- Vnitřní oscilátor lze programově nastavit na 8 různých hodnot od 31,25 kHz do 8 MHz. Vnitřní oscilátor může být konfigurován jako primární nebo sekundární zdroj taktovací frekvence, viz **kapitulu 4.5**.
- Proudová spotřeba modulu čítače/časovače Timer1 byla výrazně zredukována z obvyklých 20 μ A na 1,8 μ A (při 32 kHz a 2 V), což je mimořádně vhodné pro aplikace v reálném čase, viz **kapitulu 7.0**.
- Ochranný časovač Watchdog byl vybaven vlastní 16. bitovou předděličkou, což umožňuje nastavení doby přetečení od 1 ms do 268 s, viz **kapitulu 15.12**.
- Dvourychlostní taktování při rozběhu oscilátorů: Při konfiguraci taktovacího oscilátoru LP, XT nebo HS je mikropočítač po dobu náběhu těchto oscilátorů taktován pomocí vnitřního RC oscilátoru. To umožňuje velice rychlou reakci na zapnutí nebo probuzení mikropočítače, viz **kapitulu 15.12.3**.
- Nouzový režim taktovacího oscilátoru: Při výpadku primárního nebo sekundárního taktovacího oscilátoru je možné automaticky použít vnitřní RC oscilátor a zajistit tak bezproblémové pokračování ve výkonu programu.
- Modul AD převodníku obsahuje nový registr ANSEL pro pohodlnější konfiguraci vývodů mikropočítače.

Tabulka 1-1: Dostupný paměťový prostor v mikropočítačích PIC16F87 a PIC16F88

Mikropočítač	Paměť programu typu Flash	Paměť dat typu RWM	Paměť dat typu EEPROM
PIC16F87/88	4K x 14	368 x 8	256 x 8

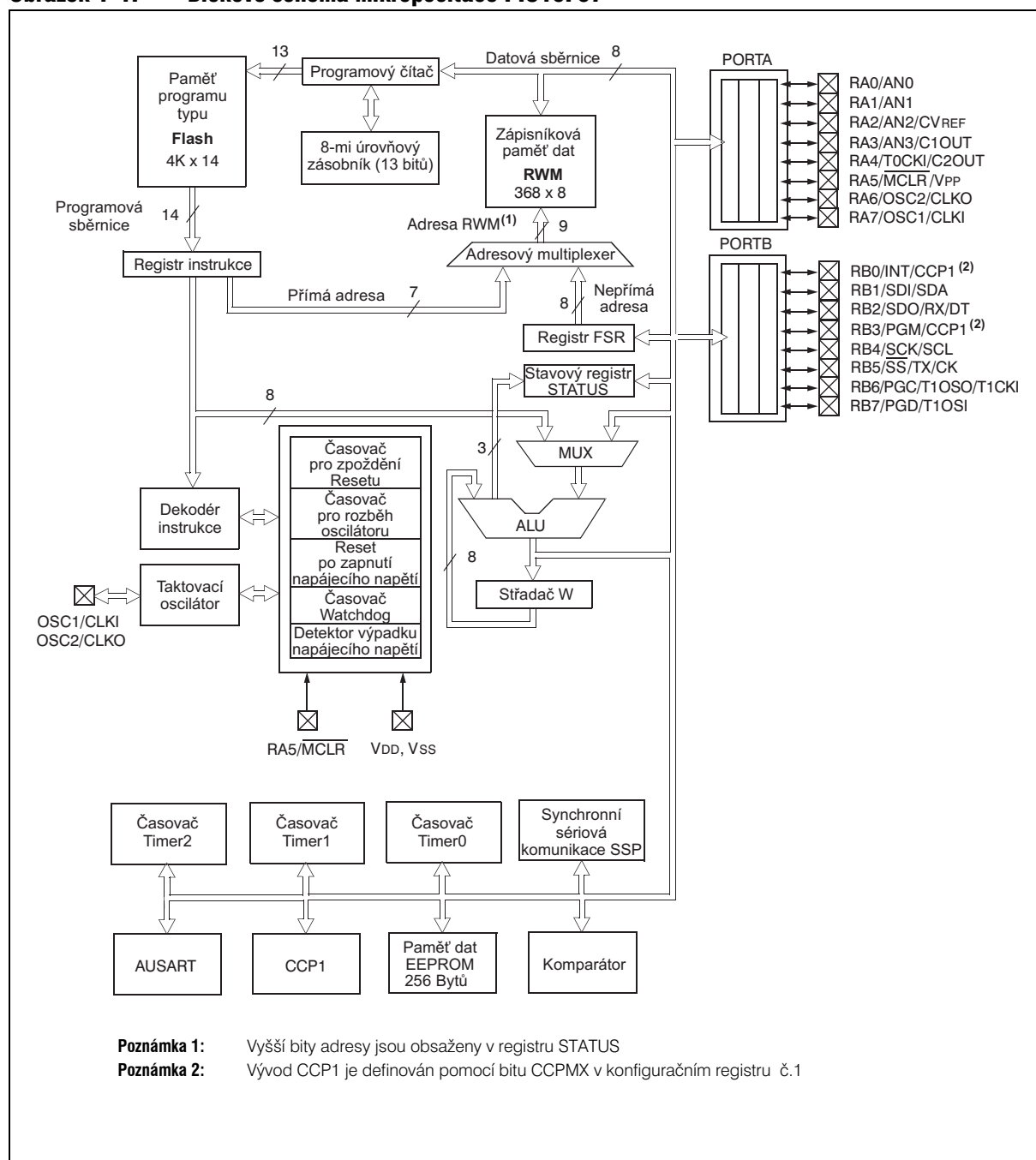
Uživatel má k dispozici celkem 16 vývodů., které může konfigurovat podle potřeby. Řada vývodů je použita pro několik různých periférií. Jedná se o následující periferní funkce:

- Vnější přerušení
- Přerušení změnou na bráně PotrB
- Vstup čítače/časovače Timer0
- Oscilátor čítače/časovače Timer1
- Režim CCP
- 7 vstupů pro 10. bitový AD převodník (jen PIC16F88)
- SPI™/I²C™
- Dva analogové komparátory
- AUSART
- Vstup $\overline{\text{MCLR}}$ (RA5) může být konfigurován jako univerzální vstup

Popis vývodů mikropočítače je uveden v tabulce 1-2.

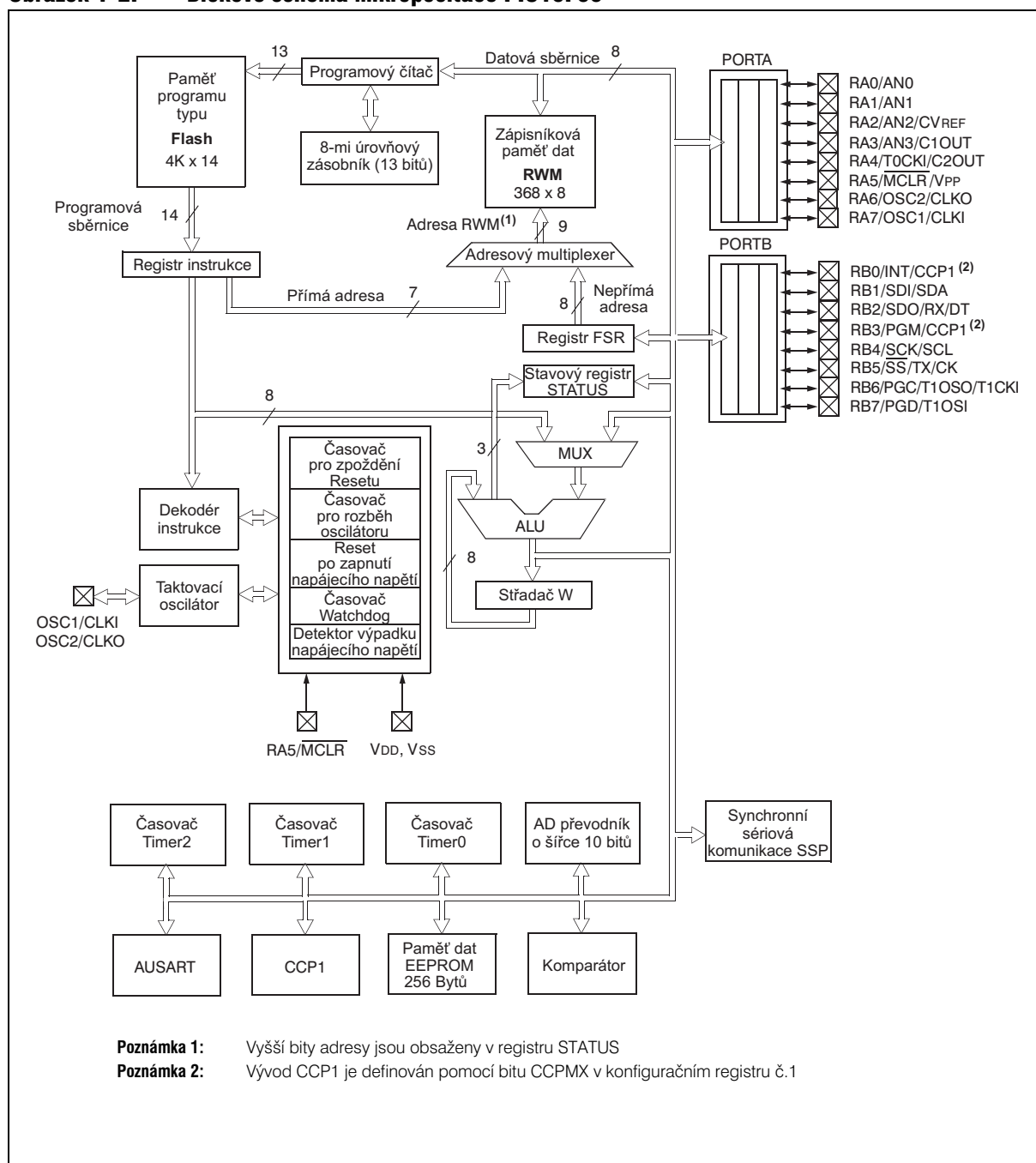
PIC16F87/88

Obrázek 1-1: Blokové schéma mikropočítače PIC16F87



PIC16F87/88

Obrázek 1-2: Blokové schéma mikropočítače PIC16F88



PIC16F87/88

Tabulka 1-2: Popis vývodů mikropočítačů PIC16F87/88

Název vývodu	Číslo vývodu pouzdra PDIP a SOIC	Číslo vývodu pouzdra SSOP	Číslo vývodu pouzdra QFN	Směr vstupně výstupního vývodu	Druh vstupu	Popis (PORTA je obousměrná vstupně výstupní brána)
RA0/AN0 RA0 AN0	17	19	23	I/O I	TTL Analog	Digitální vstup/výstup Analogový vstup č.0
RA1/AN1 RA1 AN1	18	20	24	I/O I	TTL Analog	Digitální vstup/výstup Analogový vstup č.1
RA2/AN2/CVREF/VREF- RA2 AN2 CVREF VREF- (4)	1	1	26	I/O I O I	TTL Analog Analog	Digitální vstup/výstup Analogový vstup č.2 Výstup napěťové reference komparátoru Vstup dolního referenčního napětí pro AD převodník
RA3/AN3/VREF+/C1OUT RA3 AN3 VREF+ (4) C1OUT	2	2	27	I/O I I O	TTL Analog Analog	Digitální vstup/výstup Analogový vstup č.3 Vstup horního referenčního napětí pro AD převodník Výstup komparátoru č. 1
RA4/AN4/T0CKI/C2OUT RA4 AN4 T0CKI C2OUT	3	3	28	I/O I I O	ST Analog ST	Digitální vstup/výstup Analogový vstup č.4 Vnější vstup pro čítač/časovač Timer0 Výstup komparátoru č. 2
RA/MCLR/VPP RA MCLR VPP	4	4	1	I O P	ST ST —	Digitální vstup Vstup signálu reset nebo vstup programovacího napětí. Signál reset je aktivní v log.0. Vstup pro programovací napětí
RA6/OSC2/CLKO RA6 OSC2 CLKO	15	17	20	I/O O O	ST — —	Digitální vstup/výstup Výstup taktovacího oscilátoru. Připojuje se ke krystalu nebo rezonátoru. V konfiguraci RC je na vývodu přítomna taktovací frekvence dělená čtyřmi, jejíž perioda odpovídá jednomu instrukčnímu cyklu
RA7/OSC1/CLKI RA7 OSC1 CLKI	16	18	21	I/O I I	ST ST/CMOS (3) —	Digitální vstup/výstup Vstup krystalového oscilátoru Vstup vnějšího taktovacího signálu

Vysvětlivky: I = vstup O = výstup I/O = vstup/výstup P = napájení
— = nepoužito TTL = vstup s úrovněmi TTL ST = vstup je ošetřen Schmittovým klopným obvodem

- Poznámka 1:** Pokud je tento vstup konfigurován jako vstup pro vnější přerušení, je ošetřen Schmittovým klopným obvodem
- 2:** V režimu sériového programování je tento vstup ošetřen Schmittovým klopným obvodem.
- 3:** Vstup je ošetřen Schmittovým klopným obvodem jen v konfiguraci oscilátoru RC.
Při ostatních druzích konfigurace se jedná o vstup s úrovněmi CMOS.
- 4:** Pouze u mikropočítače PIC16F88
- 5:** Vývod CCP1 je definován pomocí bitu CCPMX v konfiguračním registru č.1

PIC16F87/88

Tabulka 1-2: Popis vývodů mikropočítačů PIC16F87/88 (pokračování)

Název vývodu	Číslo vývodu pouzdra PDIP a SOIC	Číslo vývodu pouzdra SSOP	Číslo vývodu pouzdra QFN	Směr vstupně výstupního vývodu	Druh vstupu	Popis (PORTB je obousměrná vstupně výstupní brána. Na vstupech je možné softwarově zapojit odpory pull-up pro přidržení nezapojeného vývodu ve stavu log.1)
RB0/INT/CCP1 ⁽⁵⁾ RB0 INT CCP1	6	7	7	I/O I I/O	TTL ST ⁽¹⁾ ST	Digitální vstup/výstup Vnější přerušení Vstup Capture, výstup Compare, výstup PWM
RB1/SDI/SDA RB1 SDI SDA	7	8	8	I/O I I/O	TTL ST ST	Digitální vstup/výstup Vstup dat v režimu SPI Vstup dat v režimu I ² C
RB2/SDO/RX/DT RB2 SDO RX DT	8	9	9	I/O O I I/O	TTL ST	Digitální vstup/výstup Výstup dat v režimu SPI Příjem dat v režimu asynchronní AUSART detekce synchronního AUSART
RB3/PGM/CCP1 ⁽⁵⁾ RB2 PGM CCPI	9	10	10	I/O I/O I	TTL ST ST	Digitální vstup/výstup Povolení programování v režimu s nízkým napájením Vstup Capture, výstup Compare, výstup PWM
RB4/SCK/SCL RB4 SCK SCL	10	11	12	I/O I/O I	TTL ST ST	Digitální vstup/výstup Hodinový signál v režimu SPI (vstup/výstup) Hodinový signál v režimu I ² C (vstup)
RB5/SS/TX/CK RB5 SS TX CK	11	12	13	I/O I O I/O	TTL TTL	Digitální vstup/výstup Vstup Select v režimu SPI Slave Vysílání dat v režimu asynchronní AUSART Hodinový signál v režimu synchronní AUSART
RB6/AN5/PGC/T1OSO/T1CKI RB6 AN5 ⁽⁴⁾ PGC T1OSO T1CKI	12	13	15	I/O I I/O O I	TTL ST ⁽²⁾ ST ST	Digitální vstup/výstup Analogový vstup č.5 Ladění v aplikaci a vstup hodinového signálu ICSP Výstup čítače/časovače Timer1 Vnější vstup čítače/časovače Timer1
RB7/AN6/PGD/T1OSI RB7 AN6 ⁽⁴⁾ PGD T1OSI	13	14	16	I/O □ I I I	TTL ST ⁽²⁾ ST	Digitální vstup/výstup Analogový vstup č.6 Ladění v aplikaci a vstup datového signálu ICSP Vstup oscilátoru čítače/časovače Timer1
Vss	5	5, 6	3, 5	P	—	Napájecí a signálová zem
VDD	14	15, 16	17, 19	P	—	Kladné napájecí napětí

Vysvětlivky: I = vstup O = výstup I/O = vstup/výstup P = napájení
— = nepoužito TTL = vstup s úrovněmi TTL ST = vstup je ošetřen Schmittovým klopným obvodem

- Poznámka 1:** Pokud je tento vstup konfigurován jako vstup pro vnější přerušení, je ošetřen Schmittovým klopným obvodem
- 2:** V režimu sériového programování je tento vstup ošetřen Schmittovým klopným obvodem.
- 3:** Vstup je ošetřen Schmittovým klopným obvodem jen v konfiguraci oscilátoru RC.
Při ostatních druzích konfigurace se jedná o vstup s úrovněmi CMOS.
- 4:** Pouze u mikropočítače PIC16F88
- 5:** Vývod CCP1 je definován pomocí bitu CCPMX v konfiguračním registru č.1

PIC16F87/88

Poznámky:

2 ORGANIZACE PAMĚŤOVÉHO PROSTORU

Mikropočítače PIC16F87/88 mají tři oddělené bloky paměti, paměť programu, paměť dat RWM a paměť dat EEPROM. Paměť programu a paměť dat RWM používají oddělené sběrnice. To umožňuje současný nezávislý přístup do obou těchto pamětí. Tyto dva paměťové prostory jsou popsány v této kapitole. **Paměť dat typu EEPROM** je popsána v **kapitole 3.0**.

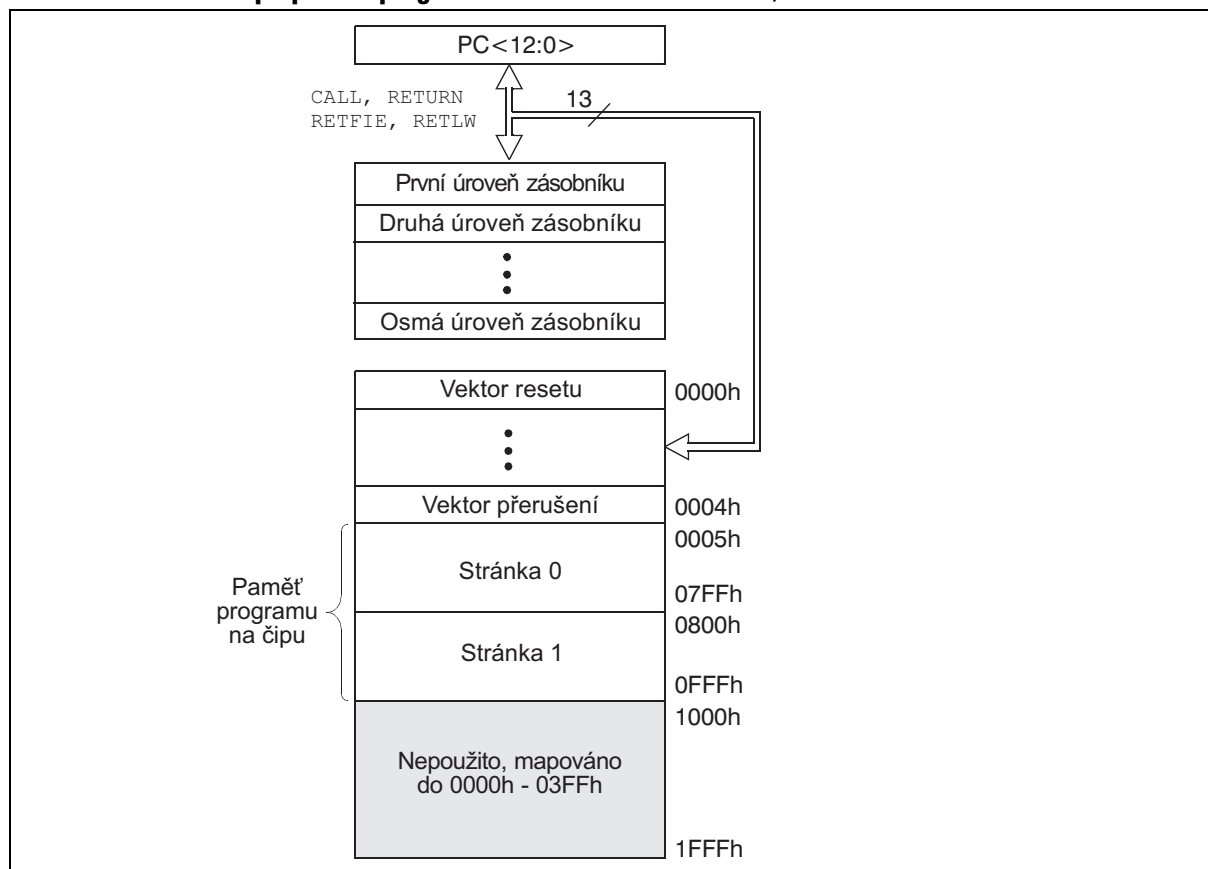
Další informace lze nalézt v dokumentaci DS33023 s názvem *PIC®micro Mid-Range MCU Reference Manual*.

2.1 Organizace paměti programu

Mikropočítače PIC16F87/88 mají 13-bitový programový čítač PC, který umožňuje adresování celkově 8192 buněk, když každá buňka obsahuje instrukční kód o délce 14 bitů. Mikropočítače mají fyzicky implementovanou paměť v rozsahu 0000h – 0FFFh, tedy pouze 4096 buněk. Adresování buněk na vyšších než fyzicky implementovaných adresách způsobí přetečení programového čítače PC. V praxi to znamená, že na adrese 020h a 1020h nalezneme tutéž paměťovou buňku.

Zahájení programu po signálu reset je na adrese 0000h (vektor resetu). Obsluhy přerušení začíná na adrese 0004h (vektor přerušení).

Obrázek 2-1: Mapa paměti programu a zásobníku PIC16F630/676



2.2 Organizace paměti dat

Paměť dat je rozdělena na dvě oblasti. První tvoří speciální funkční registry (SFR), druhou paměť pro všeobecné použití (RWM). Pomocí SFR lze řídit veškerou činnost mikropočítače. Paměť je fyzicky rozdělena do čtyř bank, z nichž každá má celkem 128 adresovatelných buněk. Každá buňka představuje jeden bajt (8 bitů). Volba banky se provádí nastavením nebo nulováním bitů RP1 a RP0 v registru STATUS<6,5>.

RP1:RP0	Registr
00	0
01	1
10	2
11	3

V každé bance lze použít 7-bitovou adresu v rozsahu 00h – 7Fh. Na nižších adresách se nalézají speciální funkční registry. Tyto registry jsou umístěny ve všech čtyřech bankách, přičemž některé významné registry jsou přístupné ve více bankách současně.

Poznámka: Popis paměti dat EEPROM je uveden v **kapitole 3.0 Paměť dat typu EEPROM a paměť programu typu Flash**.

PIC16F87/88

2.2.1 Registry pro všeobecné použití (zápisníková paměť RWM)

Registry pro všeobecné použití lze adresovat buď přímo nebo nepřímou pomocí dvojice registrů FSR a INDF.

Obrázek 2-2: Mapa paměti dat PIC16F87

[illegible]

PIC16F87/88

Obrázek 2-3: Mapa paměti dat PIC16F88

Adresa registru		Adresa registru		Adresa registru		Adresa registru	
Nepřímá adr. (*)	00h	Nepřímá adr. (*)	80h	Nepřímá adr. (*)	100h	Nepřímá adr. (*)	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	WDTCON	105h		185h
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h
	07h		87h		107h		187h
	08h		88h		108h		188h
	09h		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 ⁽¹⁾	18Eh
TMR1H	0Fh	OSCCON	8Fh	EEADRH	10Fh	Reserved ⁽¹⁾	18Fh
T1CON	10h	OSCTUNE	90h	Registry pro všeobecné použití (Zápisníková paměť RWM) 16 Bajtů	110h	Registry pro všeobecné použití (Zápisníková paměť RWM) 16 Bajtů	190h
TMR2	11h		91h				
T2CON	12h	PR2	92h				
SSPBUF	13h	SSPADD	93h				
SSPCON	14h	SSPSTAT	94h				
CCPR1L	15h		95h				
CCPR1H	16h		96h				
CCP1CON	17h		97h				
RCSTA	18h	TXSTA	98h				
TXREG	19h	SPBRG	99h				
RCREG	1Ah		9Ah				
	1Bh	ANSEL	9Bh				
	1Ch	CMCON	9Ch				
	1Dh	CVRCON	9Dh				
ADRESH	1Eh	ADRESL	9Eh				
ADCON0	1Fh	ADCON1	9Fh		11Fh		19Fh
Registry pro všeobecné použití (Zápisníková paměť RWM) 96 Bajtů	20h	Registry pro všeobecné použití (Zápisníková paměť RWM) 80 Bajtů	A0h	Registry pro všeobecné použití (Zápisníková paměť RWM) 80 Bajtů	120h	Registry pro všeobecné použití (Zápisníková paměť RWM) 80 Bajtů	1A0h
			EFh		16Fh		1EFh
			F0h		170h		1F0h
			Mapováno do 70h-7Fh		Mapováno do 70h-7Fh		Mapováno do 70h-7Fh
	7Fh		FFh		17Fh		1FFh
Bank 0		Bank 1		Bank 2		Bank 3	
 Nepoužité buňky, při čtení "0"							
* Nejedná se o fyzický registr							
Pozn1: Tyto registry jsou rezervované, ponechte jejich obsah roven 00h.							

PIC16F87/88

2.2.2 Speciální funkční registry

Pomocí speciálních funkčních registrů lze ovládat činnost mikropočítače a jeho periférií. Tyto registry mohou být rozděleny do dvou skupin. První skupinu tvoří registry ovládající jádro mikropočítače. Tyto registry jsou popsány v této kapitole. Druhou část tvoří registry pro práci s perifériemi, které jsou popsány v příslušných kapitolách popisujících tyto periferie.

Tabulka 2-1: Celkový přehled speciálních funkčních registrů

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Popsán na str.
Banka 0											
00h ⁽²⁾	INDF	Adresováním tohoto registru je přístupováno do paměti RWM na adresu ve FSR (nepřímá adresa)								0000 0000	26, 135
01h	TMR0	8-bitový čítač/časovač Timer0								xxxx xxxx	69
02h ⁽²⁾	PCL	Nižších osm bitů programového čítače (PC)								0000 0000	
03h ⁽²⁾	STATUS	IRP	RP1	RP0	T0	PD	Z	DC	C	0001 1xxx	17
04h ⁽²⁾	FSR	Ukazatel pro nepřímé adresování pro přístup do RWM prostřednictvím INDF								xxxx xxxx	135
05h	PORTA	Záchytný registr brány PortA (PIC16F87) Záchytný registr brány PortA (PIC16F88)								xxxx 0000 xxxx 0000	52
06h	PORTA	Záchytný registr brány PortB (PIC16F87) Záchytný registr brány PortB (PIC16F88)								xxxx xxxx 00xx xxxx	58
07h	—	Nepoužito								—	—
08h	—	Nepoužito								—	—
09h	—	Nepoužito								—	—
0Ah ^(1,2)	PCLATH	—	—	—	Záchytný registr pro horních 5 bitů programového čítače PC				— — — 0	0000	135
0Bh ⁽²⁾	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	0000 000x	16, 69, 77
0Ch ⁽²⁾	PIR1	—	ADIF ⁽⁴⁾	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	21, 77
0Dh	PIR2	—	—	—	—	—	—	—	—	00-0 ----	23, 34
0Eh	TMR1L	Záchytný registr pro dolní polovinu 16-bitového čítače/časovače Timer1								xxxx xxxx	77, 83
0Fh	TMR1H	Záchytný registr pro horní polovinu 16-bitového čítače/časovače Timer1								xxxx xxxx	77, 83
10h	T1CON	—	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	-000 0000	72, 83
11h	TMR2	Pracovní registr modulu Timer2								0000 0000	80, 85
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	80, 85
13h	SSPBUF	Zásobníkový registr synchronního sériového portu								xxxx xxxx	90, 95
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	89, 95
15h	CCPR1L	Registr pro nižší bajt Capture/Compare/PWM								xxxx xxxx	83, 85
16h	CCPR1H	Registr pro vyšší bajt Capture/Compare/PWM								xxxx xxxx	83, 85
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	81, 83
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	98, 99
19h	TXREG	Vysílací registr AUSART								0000 0000	103
1Ah	RCREG	Přijímací registr AUSART								0000 0000	105
1Bh	—	Nepoužito								—	—
1Ch	—	Nepoužito								—	—
1Dh	—	Nepoužito								—	—
1Eh	ADRESH ⁽⁴⁾	Vyšší bajt výsledku AD převodu								xxxx xxxx	120
1Fh	ADCON0 ⁽⁴⁾	ADCS1	ADCS0	CHS2	CHS1	GO/DONE	—	—	ADON	0000 00-0	114, 120

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, q = hodnota závisí na podmínkách, stínované buňky nejsou použity, při čtení "0"

Poznámka 1: Horní část programového čítače PC není programově přístupná. Obsah PCLATH je zapsán do horní části programového čítače automaticky vždy, pokud dojde k zápisu do PCL nebo při doplnění horních bitů adresy, která není obsažena v instrukčním kódu (stránky 0-3 paměti programu)

- 2:** Tyto registry jsou přístupné ve všech bankách.
- 3:** Vývod RA5 je pouze vstupní, stav bitu TRIS5 nemá žádný vliv a tento bit je vždy čten jako "1"
- 4:** Pouze u mikropočítače PIC16F88

PIC16F87/88

Tabulka 2-1: Celkový přehled speciálních funkčních registrů (pokračování)

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Popsán na str.
Banka 1											
80h	INDF	Adresováním tohoto registru je přístupováno do paměti RWM na adresu ve FSR (nepřímá adresa)								0000 0000	26,135
81h	OPTION_REG	RBPŮ	INTEDG	TOCS	T0SE	PSA	PS2	PS1	PS0	1111 1111	18,69
82h	PCL	Nižších osm bitů programového čítače (PC)								0000 0000	135
83h	STATUS	IRP	RP1	RP0	T0	PD	Z	DC	C	0001 1XXX	17
84h	FSR	Ukazatel pro nepřímé adresování pro přístup do RWM prostřednictvím INDF								XXXX XXXX	135
85h	TRISA	TRISA7	TRISA6	TRISA5	Konfigurační registr brány PortA					1111 1111	52,126
86h	TRISB	Konfigurační registr brány PortB								1111 1111	58,85
87h	—	Nepoužito								—	—
88h	—	Nepoužito								—	—
89h	—	Nepoužito								—	—
8Ah	PCLATH	—	—	—	Záchytný registr pro horních 5 bitů programového čítače PC					-- -0 000	135
8Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	0000 000X	19,69,77
8Ch	PIE1	—	ADIE (4)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	20,80
8Dh	PIE2	OSFIE	CMIE	—	EEIE	—	—	—	—	00 -0 - - - -	22,34
8Eh	PCON	—	—	—	—	—	—	POR	BOR	- - - - - - 0q	24
8Fh	OSCCON	—	IRCF2	IRCF1	IRCF0	OSTS	IOFS	SCS1	SCS0	-000 0000	40
90h	OSCTUNE	—	—	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0	- - 00 0000	38
91h	—	Nepoužito								—	—
92h	PR2	Registr modulu Timer2 (perioda v režimu PWM)								1111 1111	80,85
93h	SSPADD	Adresový registr synchronního portu I ² C								0000 0000	95
94h	SSPSTAT	SMP	CKE	D/A	P	S	R/W	UA	BF	0000 0000	88,95
95h	—	Nepoužito								—	—
96h	—	Nepoužito								—	—
97h	—	Nepoužito								—	—
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 - 010	97,99
99h	SPBRG	Registr rychlosti sériového přenosu								0000 0000	99,103
9Ah	—	Nepoužito								—	—
9Bh	ANSEL (4)	—	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	-111 1111	120
9Ch	CMCON	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	0000 0111	121,126, 128
9Dh	CVRCON	CVREN	CVROE	CVRR	—	CVR3	CVR2	CVR1	CVR0	000 - 0000	126,128
9Eh	ADRESL (4)	Nižší bajt výsledku AD převodu								xxxx xxxx	120
9Fh	ADCON1 (4)	ADFM	ADCS2	VCFG1	VCFG0	—	—	—	—	0000 - - - -	52,115, 120

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, q = hodnota závisí na podmínkách, stínované buňky nejsou použity, při čtení "0"

Poznámka 1: Horní část programového čítače PC není programově přístupná. Obsah PCLATH je zapsán do horní části programového čítače automaticky vždy, pokud dojde k zápisu do PCL nebo při doplnění horních bitů adresy která není obsažena v instrukčním kódu (stránky 0-3 paměti programu)

- 2:** Tyto registry jsou přístupné ve všech bankách
- 3:** Vývod RA5 je pouze vstupní, stav bitu TRIS5 nemá žádný vliv a tento bit je vždy čten jako "1"
- 4:** Pouze u mikropočítače PIC16F88

PIC16F87/88

Tabulka 2-1: Celkový přehled speciálních funkčních registrů (pokračování)

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Popsán na str.
Banka 2											
100h ⁽²⁾	INDF	Adresováním tohoto registru je přístupováno do paměti RWM na adresu ve FSR (nepřímá adresa)								0000 0000	26, 135
101h	TMR0	8-bitový čítač/časovač Timer0								xxxx xxxx	69
102h ⁽²⁾	PCL	Nižších osm bitů programového čítače (PC)								0000 0000	135
103h ⁽²⁾	STATUS	IRP	RP1	RP0	T0	PD	Z	DC	C	0001 1xxx	17
104h ⁽²⁾	FSR	Ukazatel pro nepřímé adresování pro přístup do RWM prostřednictvím INDF								xxxx xxxx	135
105h	WDTCN	—	—	—	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN	---0 1000	142
106h	PORTB	Záchytný registr brány PortB (PIC16F87) Záchytný registr brány PortB (PIC16F88)								xxxx xxxx 00xx xxxx	58
107h	—	Nepoužito								—	—
108h	—	Nepoužito								—	—
109h	—	Nepoužito								—	—
10Ah ^(1,2)	PCLATH	—	—	—	Záchytný registr pro horních 5 bitů programového čítače PC				---0 0000	135	
10Bh ⁽²⁾	INTCON	GIE	PEIE	TMROIE	INTOIE	RBIE	TMROIF	INTOIF	RBIF	0000 000x	19, 69, 77
10Ch	EEDATA	Datový registr pro zápis do EEPROM (nižších 8 bitů)								xxxx xxxx	34
10Dh	EEADR	Adresový registr pro zápis do EEPROM (nižších 8 bitů)								xxxx xxxx	34
10Eh	EEDATH	—	—	Datový registr pro zápis do EEPROM (vyšších 6 bitů)						--xx xxxx	34
10Fh	EEADRH	—	—	—	—	Adresový registr pro zápis do EEPROM (vyšší 4 bitů)				---- xxxx	34
Banka 3											
180h ⁽²⁾	INDF	Adresováním tohoto registru je přístupováno do paměti RWM na adresu ve FSR (nepřímá adresa)								0000 0000	135
181h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	18, 69
182h ⁽²⁾	PCL	Nižších osm bitů programového čítače (PC)								0000 0000	135
183h ⁽²⁾	STATUS	IRP	RP1	RP0	T0	PD	Z	DC	C	0001 1xxx	17
184h ⁽²⁾	INDF	Ukazatel pro nepřímé adresování pro přístup do RWM prostřednictvím INDF								xxxx xxxx	135
185h	—	Nepoužito								—	—
186h	TRISB	Konfigurační registr brány PortB								1111 1111	58, 83
187h	—	Nepoužito								—	—
188h	—	Nepoužito								—	—
189h	—	Nepoužito								—	—
18Ah ^(1,2)	PCLATH	—	—	—	Záchytný registr pro horních 5 bitů programového čítače PC					---0 0000	135
18Bh ⁽²⁾	INTCON	GIE	PEIE	TMROIE	INTOIE	RBIE	TMROIF	INTOIF	RBIF	0000 000x	19, 69, 77
18Ch	EECON1	EEPGD	—	—	FREE	WRERR	WREN	WR	RD	x--x x000	28, 34
18Dh	EECON2	Řídicí registr č.2 pro práci s EEPROM (nejedná se o fyzický registr)								---- ----	34
18Eh	—	Rezervováno, udržujte obsah v hodnotě 00h								0000 0000	—
18Fh	—	Rezervováno, udržujte obsah v hodnotě 00h								0000 0000	—

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, q = hodnota závisí na podmínkách, stínované buňky nejsou použity, při čtení "0"

Poznámka 1: Horní část programového čítače PC není programově přístupná. Obsah PCLATH je zapsán do horní části programového čítače automaticky vždy, pokud dojde k zápisu do PCL nebo při doplnění horních bitů adresy která není obsažena v instrukčním kódu (stránky 0-3 paměti programu)

- 2: Tyto registry jsou přístupné ve všech bankách
- 3: Vývod RA5 je pouze vstupní, stav bitu TRIS5 nemá žádný vliv a tento bit je vždy čten jako "1"
- 4: Pouze u mikropočítače PIC16F88

2.2.2.1 Registr STATUS

Registr STATUS obsahuje aritmetické příznaky, příznaky resetu a bity výběru banky paměti RWM. Registr STATUS může být cílovým registrem jakékoli operace stejně jako ostatní registry. Pokud je registr STATUS cílovým registrem pro uložení výsledku instrukce ovlivňující příznaky Z, DC nebo C, je zápis do těchto bitů zakázán a bity jsou nastaveny v závislosti na výsledku instrukce. Příznaky resetu TO a PD není možné programově ovlivnit. Je tedy vhodné ukládat výsledek instrukce, jejímž operandem je registr STATUS do střadače.

Například instrukce **CLRF STATUS** vynuluje horní tři bity a nastaví bit Z do log.1. Po této operaci vypadá obsah STATUS následovně: **000u u1uu** (kde u označuje stav beze změny).

Je tedy vhodné používat pro změny obsahu registru STATUS instrukce **BCF**, **BSF** a **MOVWF**, které neovlivňují aritmetické příznaky. Důsledek použití dalších instrukcí je popsán v **kapitole 16.0 "Instrukční sada"**.

Poznámka 1: Bity pro indikaci přetečení C a DC jsou použity rovněž jako bity pro indikaci výpůjčky při odčítání $\overline{\text{BOROW}}$ a $\overline{\text{DIGITBORROW}}$. Oba tyto bity jsou negovány. Viz podrobněji popis instrukcí

Registr 2-1: Registr STATUS (adresa: 03h, 83h, 103h, 183h)

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 **IRP:** bit výběru banky registrů RWM (používá se jen při nepřímém adresování)

- 1 = banky 2,3 (adresy 100h-1FFh)
- 0 = banky 0,1 (adresy 00h-FFh)

bit 6-5 **RP1:RP0:** bity výběru banky registrů RWM (používá se při přímém adresování)

- 11 = banka 3 (adresy 180h-1FFh)
- 10 = banka 2 (adresy 100h-17Fh)
- 01 = banka 1 (adresy 80h-FFh)
- 00 = banka 0 (adresy 00h-7Fh)
- každá banka má 128 bajtů

bit 4 **$\overline{\text{TO}}$:** příznak přetečení časovače Watchdog

- 1 = po zapnutí napájení, vykonání instrukce **CLRWDT** nebo **SLEEP**
- 0 = po resetu způsobeném přetečením časovače Watchdog

bit 3 **$\overline{\text{PD}}$:** příznak režimu Sleep

- 1 = po zapnutí napájení nebo vykonání instrukce **CLRWDT**
- 0 = po vykonání instrukce **SLEEP**

bit 2 **Z:** příznak nuly

- 1 = výsledek aritmetické nebo logické operace je nula
- 0 = výsledek aritmetické nebo logické operace není nula

bit 1 **DC:** příznak polovičního přenosu/výpůjčky (pro instrukce **ADDWF**, **ADDLW**, **SUBWF**, **SUBLW**)

- ADDWF(ADDLW):** 1 = nastal přenos z bitu b3 do bitu b4 výsledku aritmetické operace
- 0 = nenastal přenos z bitu b3 do bitu b4 výsledku aritmetické operace
- SUBWF(SUBLW):** 1 = nenastala výpůjčka z bitu b3 do bitu b4 výsledku aritmetické operace
- 0 = nastala výpůjčka z bitu b4 do bitu b3 výsledku aritmetické operace

bit 0 **C:** příznak přenosu/výpůjčky (pro instrukce **ADDWF**, **ADDLW**, **SUBWF**, **SUBLW**)

- ADDWF(ADDLW):** 1 = nastal přenos (přetečení přes hodnotu 255)
- 0 = nenastal přenos z bitu b7 do bitu b8 výsledku aritmetické operace
- SUBWF(SUBLW):** 1 = nenastala výpůjčka (podtečení pod hodnotu 0)
- 0 = nastala výpůjčka z bitu b8 do bitu b7 výsledku aritmetické operace

Poznámka 1: Pro bity výpůjček je hodnota negovaná. Odečítání je realizováno metodou dvojkového doplňku.

2: Při instrukcích rotace **RRF** a **RLF** je tento bit načten do nejvýznamnějšího, resp. nejméně významného bitu výsledku.

Význam:

R = bit pro čtení	W = bit pro zápis	U = nepoužitý bit, čteno jako '0'
-n = hodnota po resetu POR	'1' = bit je nastaven	'0' = bit je v nule
		x = hodnota bitu je neznámá

PIC16F87/88

2.2.2.2 Registr OPTION_REG

Registr obsahuje řídicí bity pro ovládání a konfiguraci programovatelné předděličky, vnějšího přerušovacího vstupu, čítače/časovače Timer0 a pull-up rezistorů brány PORTB. Všechny bity lze číst a rovněž všechny bity je možné libovolně nastavit.

Poznámka: Pro dosažení dělicího poměru při použití čítače/časovače Timer0 1:1 je nutné připojit předděličku k časovači Watchdog. Předdělička může být připojena pouze k jednomu ze dvou možných obvodů, tedy k Timer0 nebo Watchdog. Časovač Watchdog je vybaven vlastní další předděličkou. To umožňuje používat u obou obvodů různé hodnoty předděličky. Blíže je pojednáno v **kapitole 15.12**.

Registr 2-2: Registr OPTION_REG (adresa: 91h, 181h)

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

- bit 7 **RAPU:** Povolení připojení Pull-up rezistorů na všechny vývody PortB, konfigurované jako vstupní
1 = pull-up na PORTB jsou zakázány
0 = pull-up na PORTB jsou povoleny pro vývody konfigurované jako vstupní v registru TRISB
- bit 6 **INTEDG:** Volba aktivní hrany signálu pro vnější přerušení
1 = přerušení nastane s náběžnou hranou na vývodu RB0/INT
0 = přerušení nastane se sestupnou hranou na vývodu RB0/INT
- bit 5 **T0CS:** Volba zdroje hodinového signálu pro čítač/časovač Timer0
1 = vnější zdroj na vývodu RA4/T0CKI/C2OUT
0 = vnitřní zdroj v závislosti na vnitřním taktovacím oscilátoru (CLKO)
- bit 4 **T0SE:** Volba aktivní hrany při vnějším zdroji hodinového signálu pro Timer0
1 = inkrementace se sestupnou hranou na vývodu RA4/T0CKI/C2OUT
0 = inkrementace s náběžnou hranou na vývodu RA4/T0CKI/C2OUT
- bit 3 **PSA:** Volba použití předděličky
1 = předdělička je předřazena časovači Watchdog (WDT)
0 = předdělička je předřazena čítači/časovači Timer0
- bit 2-0 **PS2:PS0:** Volba dělicího poměru předděličky v závislosti na použití předděličky

kombinace	poměr pro Timer0	poměr pro Watchdog
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

Význam:

R = bit pro čtení

W = bit pro zápis

U = nepoužitý bit, čteno jako '0'

–n = hodnota po resetu POR

'1' = bit je nastaven

'0' = bit je v nule

x = hodnota bitu je neznámá

2.2.2.3 Registr INTCON

Registr INTCON obsahuje bity pro práci s přerušovacím systémem mikropočítače. Pomocí registru INTCON lze maskovat 4 a indikovat 3 přerušovací zdroje. Všechny bity lze číst a všechny bity lze také libovolně nastavit.

Poznámka: Příznaky přerušení se nastaví vždy, když jsou splněny podmínky, které mohou vyvolat přerušení a to i tehdy, pokud žádost o přerušení není přijata, např. přerušení nejsou povolena (GIE=0) nebo jsou přerušení maskována. Uživatel musí před povolením přerušení ošetřit příslušný příznak jejich vynulováním.

Registr 2-3: Registr INTCON (adresa: 0Bh, 8Bh, 10Bh, 18Bh)

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	PEIE	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
bit 7							bit 0

- bit 7 **GIE:** Globální povolení přerušení
1 = povolena všechna nemaskovaná přerušení
0 = zakázána všechna přerušení
- bit 6 **PEIE:** Povolení přerušení od periférií (maska)
1 = povolena všechna nemaskovaná přerušení od periférií
0 = zakázána všechna přerušení od periférií
- bit 5 **TMR0IE:** Povolení přerušení od čítače/časovače Timer0 (maska)
1 = povoleno přerušení přetečením Timer0
0 = zakázáno (maskováno) přerušení přetečením Timer0
- bit 4 **INT0IE:** Povolení přerušení vnějším signálem na vývodu RB0/INT (maska)
1 = povoleno přerušení od signálu na vývodu RB0/INT
0 = zakázáno (maskováno) přerušení od signálu na vývodu RB0/INT
- bit 3 **RBIE:** Povolení přerušení změnou hodnoty na jednom ze čtyř vyšších bitů brány PortB (maska)
1 = povoleno přerušení změnou hodnoty na bráně PortB
0 = zakázáno (maskováno) přerušení změnou hodnoty na bráně PortB
- bit 2 **TMR0IF:** Příznak přerušení od čítače/časovače Timer0
1 = došlo k přetečení registru TMR0 (nutno nulovat programově)
0 = nedošlo k přetečení registru TMR0
- bit 1 **INT0IF:** Příznak přerušení vnějším signálem na vývodu RB0/INT
1 = žádost o přerušení z vnějšího zdroje RB0/INT (nutno nulovat programově)
0 = nedošlo k žádosti o přerušení z vnějšího zdroje RB0/INT
- bit 0 **RBIF:** Příznak přerušení změnou hodnoty na jednom ze čtyř vyšších bitů brány PortB (maska)
 Událost trvá, dokud nedojde k novému čtení z brány PortB.
 Teprve poté je hardwarově umožněno nulovat příznak RBIF.
1 = nejméně jeden ze čtyř bitů RB7:RB4 změnil stav (nutno nulovat programově)
0 = žádný ze čtyř bitů RB7:RB4 nezměnil stav

Význam:

R = bit pro čtení	W = bit pro zápis	U = nepoužitý bit, čteno jako '0'
–n = hodnota po resetu POR	'1' = bit je nastaven	'0' = bit je v nule x = hodnota bitu je neznámá

PIC16F87/88

2.2.2.4 Registr PIE1

Registr PIE1 obsahuje bity pro individuální povolení jednotlivých druhů přerušení od periférií. Pomocí tohoto registru lze tato přerušení maskovat.

Poznámka: Pro povolení obsluhy jakéhokoli přerušení od periférie musí být současně povolena všechna přerušení od periférií nastavením bitu PEIE v registru INTCON<6>.

Registr 2-4: Registr PIE1 (adresa 8Ch)

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	ADIE ⁽¹⁾	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
bit 7							bit 0

- bit 7 **Nepoužito:** při čtení "0"
- bit 6 **ADIE:** Povolení přerušení dokončením AD převodu
1 = povoleno přerušení dokončením AD převodu
0 = zakázáno přerušení dokončením AD převodu
Poznámka 1: tento bit je použit jen u PIC16F88, u PIC16F87 je vždy čten jako "0"
- bit 5 **RCIE:** Povolení přerušení od přijetí bajtu pomocí AUSART
1 = povoleno přerušení od přijetí bajtu pomocí AUSART
0 = zakázáno přerušení od přijetí bajtu pomocí AUSART
- bit 4 **TXIE:** Povolení přerušení od dokončení vysílání bajtu pomocí AUSART
1 = povoleno přerušení od dokončení vysílání bajtu pomocí AUSART
0 = zakázáno přerušení od dokončení vysílání bajtu pomocí AUSART
- bit 3 **SSPIE:** Povolení přerušení od modulu SSP
1 = povoleno přerušení od modulu SSP
0 = zakázáno přerušení od modulu SSP
- bit 2 **CCP1IE:** Povolení přerušení od modulu CCP1
1 = povoleno přerušení od modulu CCP1
0 = zakázáno přerušení od modulu CCP1
- bit 1 **TMR2IE:** Povolení přerušení při rovnosti registrů TMR2 a PR2
1 = povoleno přerušení při rovnosti registrů TMR2 a PR2
0 = zakázáno přerušení při rovnosti registrů TMR2 a PR2
- bit 0 **TMR1IE:** Povolení přerušení přetečením čítače/časovače Timer1
1 = povoleno přerušení přetečením čítače/časovače Timer1
0 = zakázáno přerušení přetečením čítače/časovače Timer1

Význam:

R = bit pro čtení

W = bit pro zápis

U = nepoužitý bit, čteno jako '0'

–n = hodnota po resetu POR

'1' = bit je nastaven

'0' = bit je v nule

x = hodnota bitu je neznámá

2.2.2.5 Registr PIR1

Registr PIR1 obsahuje příznakové bity pro přerušení od periférií.

Poznámka: Příznaky přerušení se nastaví vždy, když jsou splněny podmínky, které mohou vyvolat přerušení a to i tehdy, pokud žádost o přerušení není přijata, např. přerušení nejsou povolena (GIE=0) nebo jsou přerušení maskována. Uživatel musí zajistit programově jejich nulování před povolením přerušení.

Registr 2-5: Registr PIR1 (adresa 0Ch)

U-0	R/W-0	R-0	R-0	R-0	R/W-0	R/W-0	R/W-0
—	ADIF ⁽¹⁾	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
bit 7							bit 0

- bit7 **Nepoužito**, při čtení "0"
- bit6 **ADIF:** Příznak přerušení od dokončení AD převodu
1 = došlo k dokončení AD převodu
0 = nedošlo k dokončení AD převodu
Poznámka 1: tento bit je použit jen u PIC16F88, u PIC16F87 je vždy čten jako "0"
- bit5 **RCIF:** Příznak přerušení od přijetí bajtu pomocí AUSART
1 = došlo k příjmu bajtu pomocí AUSART
0 = nedošlo k příjmu bajtu pomocí AUSART
- bit4 **TXIF:** Příznak přerušení od dokončení vysílání bajtu pomocí AUSART
1 = došlo k dokončení vysílání bajtu pomocí AUSART
0 = nedošlo k dokončení vysílání bajtu pomocí AUSART
- bit3 **SSPIF:** Příznak přerušení od modulu synchronního sériového přenosu SSP
1 = došlo k žádosti přerušení od modulu SSP
0 = nedošlo k žádosti přerušení od modulu SSP
- bit2 **CCP1IF:** Příznak přerušení od modulu CCP1
1 = došlo k žádosti o přerušení od modulu CCP1
0 = nedošlo k žádosti o přerušení od modulu CP1
- bit1 **TMR2IF:** Příznak přerušení při rovnosti registrů TMR2 a PR2
1 = došlo ke shodě obsahu registrů TMR2 a PR2
0 = nedošlo ke shodě obsahu registrů TMR2 a PR2
- bit0 **TMR1IF:** Příznak přerušení přetečením čítače/časovače Timer1
1 = došlo k přetečení čítače/časovače Timer1
0 = došlo k přetečení čítače/časovače Timer1

Význam:

R = bit pro čtení

–n = hodnota po resetu POR

W = bit pro zápis

'1' = bit je nastaven

U = nepoužitý bit, čteno jako '0'

'0' = bit je v nule

x = hodnota bitu je neznámá

PIC16F87/88

2.2.2.6 Registr PIE2

Registr PIE2 obsahuje bity pro individuální povolení přerušení od modulu oscilátoru, dokončení zápisu do EEPROM a komparátoru. Pomocí tohoto registru lze tato přerušení maskovat.

Registr 2-6: Registr PIE2 (adresa 8Dh)

R/W-0	R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0
OSFIE	CMIE	—	EEIE	—	—	—	—
bit 7				bit 0			

- bit7 **OSFIE:** Povolení přerušení při výpadku oscilátoru
1 = povoleno
0 = zakázáno
- bit6 **CMIE:** Povolení přerušení od komparátoru
1 = povoleno přerušení od komparátoru
0 = zakázáno přerušení od komparátoru
- bit5 **nepoužit:** při čtení "0"
- bit4 **EEIE:** Povolení přerušení od dokončení zápisu bajtu do paměti EEPROM
1 = povoleno přerušení od dokončení zápisu bajtu do paměti EEPROM
0 = zakázáno přerušení od dokončení zápisu bajtu do paměti EEPROM
- bit3-1 **nepoužity:** při čtení "0"

Význam:

R = bit pro čtení

W = bit pro zápis

U = nepoužitý bit, čteno jako '0'

–n = hodnota po resetu POR

'1' = bit je nastaven

'0' = bit je v nule

x = hodnota bitu je neznámá

2.2.2.7 Registr PIR2

Registr PIR2 obsahuje příznakové bity pro přerušení od modulu oscilátoru, dokončení zápisu do EEPROM a komparátoru.

Poznámka: Příznaky přerušení se nastaví vždy, když jsou splněny podmínky, které mohou vyvolat přerušení a to i tehdy, pokud žádost o přerušení není přijata, např. přerušení nejsou povolena (GIE=0) nebo jsou přerušení maskována. Uživatel musí zajistit programově jejich nulování před povolením přerušení.

Registr 2-7: Registr PIR2 (adresa 0Dh)

R/W-0	R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0
OSFIF	CMIF	—	EEIF	—	—	—	—
bit 7							bit 0

- bit7 **OSFIF:** příznak výpadku taktovacího oscilátoru
1 = došlo k výpadku taktovacího oscilátoru
0 = nedošlo k výpadku taktovacího oscilátoru
- bit6 **CMIF:** Příznak přerušení od změny komparátoru
1 = došlo ke změně na výstupu komparátoru (nutno nulovat softwarově)
0 = nedošlo ke změně na výstupu komparátoru
- bit5 **nepoužit:** při čtení "0"
- bit4 **EEIF:** Příznak přerušení od dokončení zápisu bajtu do paměti EPROM
1 = došlo k dokončení zápisu do EEPROM (nutno nulovat softwarově)
0 = nedošlo k dokončení zápisu do EEPROM nebo nebyl zápis zahájen
- bit3-1 **nepoužity:** při čtení "0"

Význam:

R = bit pro čtení	W = bit pro zápis	U = nepoužitý bit, čteno jako '0'
–n = hodnota po resetu POR	'1' = bit je nastaven	'0' = bit je v nule x = hodnota bitu je neznámá

2.2.2.8 Registr PCON

Poznámka: Příznaky přerušení se nastaví vždy, když jsou splněny podmínky, které mohou vyvolat přerušení a to i tehdy, pokud žádost o přerušení není přijata, např. přerušení nejsou povolena (GIE=0) nebo jsou přerušení maskována. Uživatel musí zajistit programově jejich nulování před povolením přerušení.

Registr PCON obsahuje příznakové bity určující druh resetu. Podle nastavení těchto bitů lze určit, zda poslední reset byl reset po zapnutí napájecího napětí (POR), reset vnějším signálem na vývodu MCLR, reset aktivací ochranného obvodu detekujícím výpadek napájecího napětí (BOD) nebo reset způsobený přetečením ochranného časovače Watchdog.

Poznámka: Hodnota bitu $\overline{BOR}$ není po resetu $\overline{POR}$ definovaná. Uživatel musí tento bit nastavit a dále detekovat aby zjistil, zda došlo k resetu způsobeném $\overline{BOD}$. Bit $\overline{BOR}$ je nastaven nezávisle na povolení nebo zákazu obvodu BOD v konfiguračním slově bitem BODEN.

Registr 2-8: Registr PCON (adresa 8Eh)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-x
—	—	—	—	—	—	$\overline{POR}$	$\overline{BOR}$
bit 7							bit 0

bit7-2 **nepoužity:** při čtení "0"

bit1 **POR:** příznak resetu po zapnutí napájecího napětí

- 1 = nedošlo k resetu způsobeném zapnutím napájecího napětí
- 0 = došlo k resetu způsobeném zapnutím napájecího napětí (nutno nastavit softwarově)

bit0 **BOR:** Příznak resetu způsobeného výpadkem napájecího napětí

- 1 = nedošlo k poklesu napájecího napětí
- 0 = došlo k poklesu napájecího napětí

Význam:

R = bit pro čtení

W = bit pro zápis

U = nepoužitý bit, čteno jako '0'

-n = hodnota po resetu POR

'1' = bit je nastaven

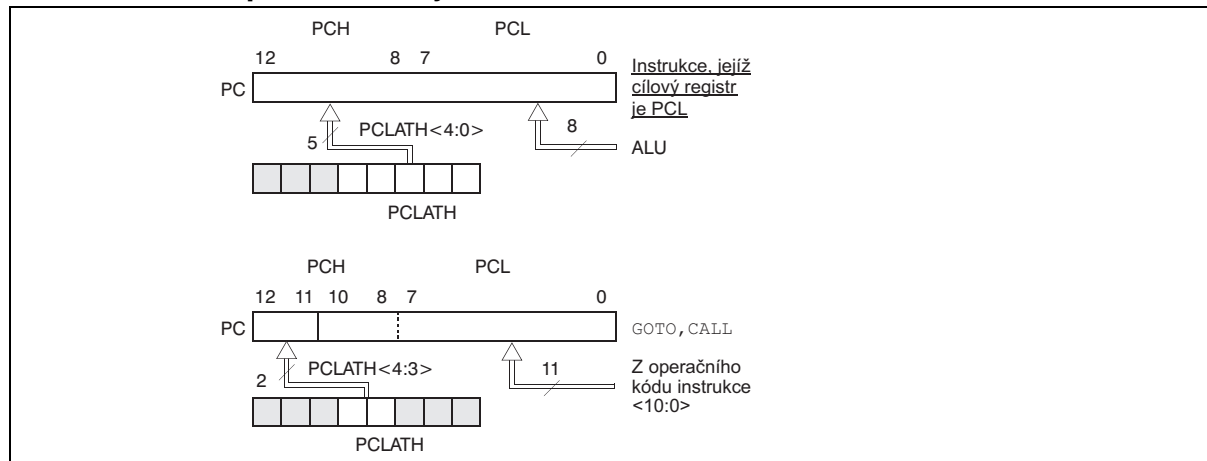
'0' = bit je v nule

x = hodnota bitu je neznámá

2.3 Registry PCL a PCLATH

Programový čítač (PC) má šířku 13 bitů. Nižší bajt je registr PCL, z něhož lze číst i do něj zapisovat. Vyšší bajt ($PC < 12:8 >$) nelze číst vůbec, zapisovat do něj lze nepřímo pomocí PCLATH. Při resetu jakéhokoli druhu dojde k vynulování celého programového čítače. Na obrázku 2-5 jsou uvedeny dva příklady přesunu dat do programového čítače. V horní ukázce jde o přesun dat při zápisu do PCL, kdy automaticky dojde i k přesunu dat z PCLATH do horní části PC. Spodní ukázka zachycuje situaci, při které je obsah PC načítán během vykonání instrukcí CALL nebo GOTO.

Obrázek 2-4: Zápis do PC v různých situacích



2.3.1 Skok modifikací PCL

V rámci jedné stránky paměti programu (256 instrukcí) lze provést skok modifikací obsahu PCL. Každý zápis do PCL způsobí přepsání horních bitů PC ze záchytného registru PCLATH. Je nezbytné před každým takovým skokem nastavit do PCLATH adresu požadované stránky paměti programu do které chceme skočit. K samotnému skoku lze pak využít např. instrukci **ADDWF PCL**. Podrobněji je pojednáno v aplikační příručce s označením AN556.

2.3.2 Zásobník (Stack)

Mikropočítače PIC16F87/88 jsou vybaveny 8-úrovňovým 13-bitovým hardwarovým zásobníkem. Tento zásobník ani jeho ukazatel nejsou součástí žádného paměťového prostoru. Obsah programového čítače PC se ukládá do zásobníku vždy, pokud je voláno přerušení nebo instrukce **CALL**. Dochází tím k uchování návratové adresy. Obsah zásobníku je naopak vyzvednut při vykonání jedné z návratových instrukcí **RETURN**, **RETLW** nebo **RETFIE**. Obsah registru PCLATH není operacemi uložení (PUSH) nebo vyzvednutí (POP) nikterak ovlivněn.

Zásobník pracuje jako kruhový posuvný registr. Po osmém uložení dat do zásobníku, tedy při pokusu o deváté uložení dojde k přepsání první uložené návratové adresy. Desáté uložení přepíše druhou návratovou adresu atd.

Poznámka 1: Přetečení zásobníku nelze zjistit pomocí žádného příznakového bitu

2: Do zásobníku nelze uložit nic jiného než návratovou adresu. Nejsou k dispozici žádné instrukce typu PUSH nebo POP.

2.4 Stránkování paměti programu

Mikropočítače PIC16F87/88 jsou schopny adresovat paměť programu v rozsahu 8K. Adresa pro takový adresový prostor má celkem 13 bitů. V operačním kódu instrukcí **CALL** nebo **GOTO** je pouze 11 bitů. Tyto instrukce jsou tedy schopny adresovat pouze v rámci 11-bitové adresy. Z toho vyplývá rozdělení paměti programu na čtyři stránky, z nichž každá má právě 2048 buňek. Při každém vykonání instrukce **GOTO** nebo **CALL** dojde k doplnění dvou nejvyšších bitů PC z PCLATH. Nastavením těchto bitů lze tedy provést skok do libovolného místa v paměti programu v rozsahu 13-bitové adresy. Do zásobníku se ukládá celá návratová adresa, proto nemá obsah PCLATH vliv na vykonání návratových instrukcí **RETURN**, **RETLW** a **RETFIE**.

Poznámka: Před každou instrukcí CALL nebo GOTO musí uživatel nastavit správnou hodnotu do registru PCLATH.

Na příkladu 2-1 je uvedeno volání podprogramu na stránce 1 paměti programu ze stránky 0. Pokud by bylo povoleno přerušení, je nutné uložit vždy obsah PCLATH v rutině přerušení.

Příklad 2-1: Volání podprogramu na stránce 1 ze stránky 0

	ORG	0x500	
	BCF	PCLATH, 4	
	BCF	PCLATH, 3	;Výběr stránky 1
			; (adresy 800h-FFFh)
	CALL	SUB1_P1	;Volání podprogramu
	:		;na stránce 1(adresy 800h-FFFh)
	:		
	ORG	0x900	;Podprogram na stránce 1
SUB1_P1	:		;Volaný podprogram
			;na stránce 1
	:		
	RETURN		;Návrat
			;na stránku 0
			; (adresy 000h-7FFh)

2.5 Nepřímé adresování, registry INDF a FSR

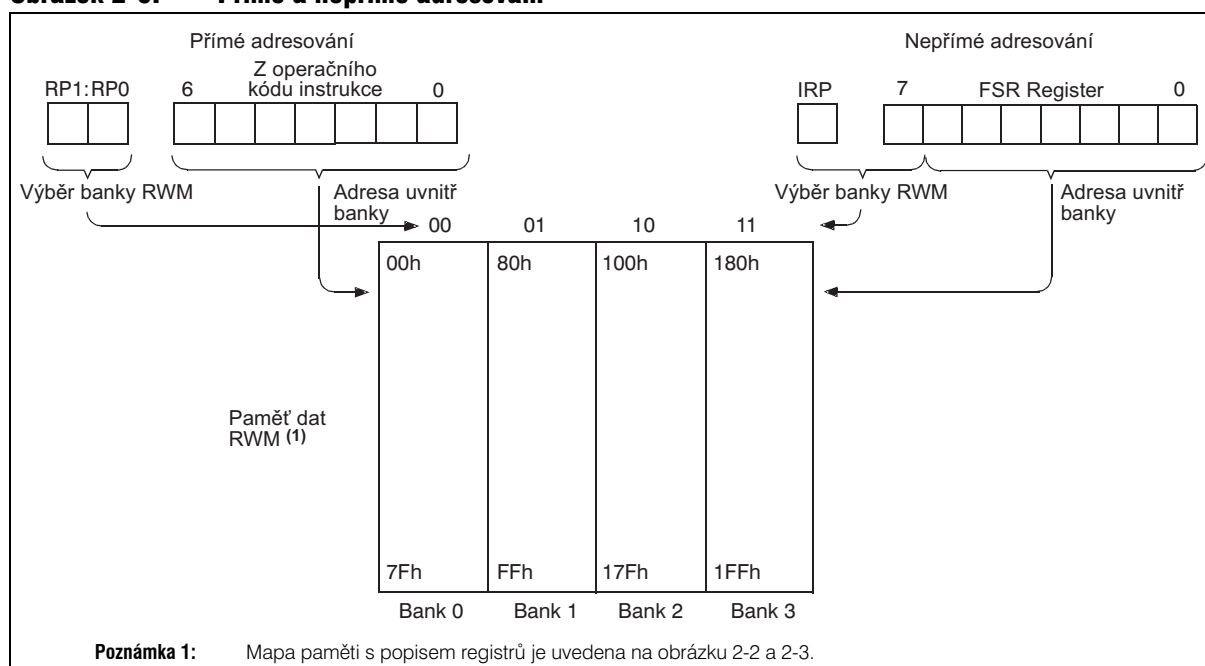
Registr FSR je ukazatel paměti pro nepřímé adresování. Obsahuje adresu buňky v paměti RWM, se kterou chceme pracovat. Registr INDF není skutečný registr, při práci s ním pracujeme ve skutečnosti s registrem, jehož adresa je uložena v ukazateli FSR. Čtením registru INDF (při FSR = 0) získáme hodnotu 00h. Pomocí nepřímého adresování lze adresovat celý prostor paměti RWM. Jedná se celkem o 4 banky po 128 adresách. Celý prostor má velikost 512 buněk a k jeho adresování je zapotřebí celkem 9 bitů. 8 nižších bitů je uloženo v registru FSR. Nejvyšší bit je doplněn z bitu IRP v registru STATUS<7>.

Na příkladu 2-2 je uveden jednoduchý program, který pomocí nepřímého adresování vynuluje paměť RWM v rozsahu 20h-2Fh.

Příklad 2-2: Nepřímé adresování

	MOVLW	0x20	;nastavení ukazatele
	MOVWF	FSR	;do FSR
NEXT	CLRF	INDF	;nuluj RWM podle FSR
	INCF	FSR, F	;další adresa
	BTFSS	FSR, 4	;už je konec?
	GOTO	NEXT	;ne, nuluj dál
CONTINUE	:		;ano, pokračuj

Obrázek 2-5: Přímé a nepřímé adresování



3 PAMĚŤ DAT TYPU EEPROM A PAMĚŤ PROGRAMU TYPU Flash

Paměť dat typu EEPROM a paměť programu typu Flash je přístupná pro čtení i zápis během výkonu programu. Paměť není přístupná přímo, ale pomocí speciálních funkčních registrů. Pro zápis nebo čtení z těchto pamětí slouží celkem 6 speciálních funkčních registrů.

- EECON1
- EECON2
- EEDATA
- EEDATH
- EEADR
- EEADRH

V této kapitole je popsán přístup do paměti dat a programu během výkonu programu.

Přístup do paměti pomocí sériového programování pomocí externího zařízení je popsáno v příslušné kapitole.

Při přístupu do paměti dat typu EEPROM se používá registr EEADR jako ukazatel paměti. Obsahuje adresu buňky, se kterou budeme pracovat v rozsahu 00h – FFh.

Registr EEDATA je pracovním registrem. Obsahuje 8-bitová data která jsou určena k zápisu nebo jsou vyčtena z určené buňky. Tímto způsobem lze adresovat celkem 256 buněk paměti dat EEPROM. Pokud má mikropočítač fyzicky implementováno pouze 128 buněk, jsou adresy mimo rozsah 00h–7Fh mapovány do tohoto prostoru. Při pokusu o zápis do nepoužitých oblastí paměti EEPROM je zapisovací obvod vyřazen z činnosti.

Při přístupu do paměti programu typu Flash se používají registry EEADR a EEADRH jako ukazatele paměti v rozsahu 0000h – 1FFFh. Obsahuje adresu buňky, se kterou budeme pracovat. Registry EEDATA a EEDATH jsou pracovní registry. Obsahují 14-bitová data která jsou určena k zápisu nebo jsou vyčtena z určené buňky. Tímto způsobem lze adresovat celkem 8192 buněk paměti programu typu Flash. Pokud má mikropočítač fyzicky implementováno pouze 4096 buněk, jsou adresy mimo rozsah 000h–FFFh mapovány do tohoto prostoru.

Do paměti dat typu EEPROM lze zapisovat a číst jen po jednotlivých bajtech. Před zápisem nových dat dojde automaticky k vymazání předchozího obsahu. Z paměti programu lze číst po jednotlivých slovech, ale zápis musí být proveden současně pro čtyři po sobě následující slova. Nejprve musí dojít k současnému vymazání čtyř buněk v paměti programu, následuje postupné zapsání čtyř slov. V paměti dat typu EEPROM dochází k mazání automaticky.

Zápis je řízen vnitřním časovačem. Programovací napětí je generováno vnitřní nábojovou pumpou, která pracuje v celém rozsahu napájecího napětí.

Do paměti dat EEPROM lze libovolně přistupovat i v případě, že je aktivní ochrana kódu proti vyčtení. Z paměti programu lze číst kdykoli, zápis je možný v závislosti na nastavení konfiguračních bitů. Nastavení ochrany proti vyčtení dat znemožní čtení paměti vnějším zařízením, uvnitř mikropočítače je čtení povoleno vždy.

3.1 Registry EEADR a EEADRH

Pomocí registrů EEADR a EEADRH lze adresovat prostor do 256 nebo do 8192 adres.

Při práci s pamětí dat EEPROM je použit pouze registr EEADR. Při práci s pamětí programu typu Flash je dolní část adresy umístěná v registru EEADR a horní v EEADRH.

Pokud má mikropočítač menší paměť než odpovídá plnému rozsahu adres, je nejvýznamnější bit (MSb) ignorován.

3.2 Registry EECON1 a EECON2

Registr EECON1 obsahuje řídicí bity pro práci s pamětí EEPROM dat a pamětí programu typu Flash.

Řídicí bit EEPGD určuje, zda bude pracováno s pamětí dat typu EEPROM nebo s pamětí programu typu Flash. Po resetu je tento bit nulován. To odpovídá práci s pamětí dat. Jeho nastavením budeme přistupovat do paměti programu.

Řídicí bity RD a WR zahajují čtení resp. zápis z určené paměti. Tyto bity nelze nulovat programem. Nulují se automaticky po dokončení čtecí resp. zápisové operace.

Řídicí bit WREN povoluje zápisové resp. mazací operace v paměti. Po resetu je tento bit nulován. Pokud nechcete zapisovat do paměti, je vhodné udržovat tento bit v nule. Příznakový bit WRERR indikuje přerušení operace zápisu nebo mazání signálem $\overline{\text{MCLR}}$ nebo resetem způsobeným časovačem Watchdog během vykonání programu.

Po zjištění nastavení tohoto bitu po resetu je nutné zopakovat zápis. Obsah registrů EEDATA a EEADR zůstává beze změny.

Příznakový bit EEIF v registru PIR2 je nastaven vždy po dokončení zápisové operace. Jeho nulování je nutné provést softwarově.

Registr EECON2 není fyzický registr. Jeho čtením získáme hodnotu 00h. Tento registr slouží při zápisu do paměti jako ochranný registr, do kterého je nutné zapsat požadovanou sekvenci dat pro inicializaci zápisu.

PIC16F87/88

Registr 3-1: Registr EECON1 (adresa 18Ch)

R/W-x	U-0	U-0	R/W-x	RW-x	R/W-0	R/S-0	R/S-0
EEPGD	—	—	FREE	WRERR	WREN	WR	RD
bit 7							bit 0

- bit7 **EEPGD:** Volba mezi pamětí dat typu EEPROM a pamětí programu typu Flash
1 = přistupuje se do paměti programu
0 = přistupuje se do paměti dat
- bit6-5 **nepoužito:** při čtení "0"
- bit4 **FREE:** Povolení mazání bloku paměti programu
1 = mazání povoleno před zahájením další zápisové operace
0 = mazání je zakázáno
- bit3 **WRERR:** Příznak nedokončení zápisu do paměti EEPROM
1 = proces zápisu do paměti byl předčasně přerušen (resetem na vývodu MCLR nebo přetečením časovače Watchdog)
0 = proces zápisu proběhl bez přerušení
- bit2 **WREN:** Povolení zápisu do paměti EEPROM
1 = zápis je povolen
0 = zápis je zakázán
- bit1 **WR:** Řídící bit pro zápis dat
1 = zahájení zápisu do EEPROM, bit je hardwarově vynulován okamžitě po dokončení zápisu (bit může být hardwarově pouze nastaven, nikoli nulován)
0 = zápis dat do paměti EEPROM byl dokončen nebo nebyl zahájen.
- bit0 **RD:** Řídící bit pro čtení dat
1 = zahájení čtení z EEPROM, bit je hardwarově vynulován okamžitě po dokončení čtení (bit může být hardwarově pouze nastaven, nikoli nulován)
0 = čtení dat z paměti EEPROM bylo dokončeno nebo nebylo zahájeno

Význam:

R = bit pro čtení	W = bit pro zápis	U = nepoužitý bit, čteno jako '0'
–n = hodnota po resetu POR	'1' = bit je nastaven	'0' = bit je v nule x = hodnota bitu je neznámá

3.3 Čtení z paměti dat typu EEPROM

Pro čtení dat z paměti EEPROM je nutné použít ukazatele dat v registru EEADR. Do tohoto registru je nutné umístit adresu buňky, kterou chceme přečíst. Dále je nutné vynulovat bit EEPGD v registru EECON1 <7> a nastavit bit RD v registru EECON1 <0>. Již v následujícím instrukčním cyklu jsou v registru EEDATA připravena data vyčtená z paměti EEPROM. V registru EEDATA zůstane vyčtená hodnota do dalšího čtení z paměti EEPROM nebo do přepsání registru jinou hodnotou.

Programová rutina obsluhy je uvedena na příkladu 3-1.

Při čtení z paměti EEPROM je nutné provést následující kroky:

1. Zapsat adresu do ukazatele EEADR. V případě 7-bitové adresy ponechte nejvyšší bit v "0".
2. Nulovat bit EEPGD pro aktivaci paměti dat typu EEPROM.
3. Nastavit bit RD pro zahájení procesu čtení z paměti.
4. Přečíst data z registru EEDATA.

Příklad 3-1: Rutina čtení z paměti EEPROM

BANKSEL	EEADR	;přepnutí do banky 2
MOVF	ADDR, W	;
MOVWF	EEADR	;nastavení adresy
		;do ukazatele
BANKSEL	EECON1	;přepnutí do banky 3
BCF	EECON1, EEPGD	;aktivace paměti dat EEPROM
BSF	EECON1, RD	;zahájení čtení
BANKSEL	EEDATA	;přepnutí do banky 2
MOVF	EEDATA, W	;načtení dat z EEPROM

3.4 Zápis do paměti dat typu EEPROM

Pro zápis do paměti dat typu EEPROM je nutné zapsat adresu buňky do které se bude zapisovat do registru EEADR. Do registru EEDATA je nutné zapsat 8-bitové slovo určené k zápisu. Následně musí uživatel zajistit provedení požadované sekvence instrukcí, která inicializuje proces zápisu.

Proces zápisu nebude zahájen, pokud nebude zcela přesně dodržena požadovaná sekvence instrukcí. Jedná se o zápis dat 55h a AAh postupně do registru EECON2 s následnou aktivací zápisového bitu WR pro každý zapisovaný bajt. Během této sekvence instrukcí je zcela nezbytné zakázat přerušení. Programová rutina pro zápis je uvedena na příkladu 3-2.

Před provedením této požadované sekvence je nutné povolit zápis nastavením bitu WREN v registru EECON1 <2>. Tento postup zabraňuje nežádoucímu zápisu do paměti EEPROM, ke kterému by mohlo dojít např. při zabloudění programu. Bit WREN by měl být udržován v nule po celou dobu běhu programu a nastaven vždy jen po dobu zápisu do EEPROM. Tento bit je nutné nulovat softwarově.

Pokud již došlo k zahájení zápisu pomocí požadované sekvence, nulování bitu WREN nemá vliv na dokončení zápisu. Bit WR naopak nelze nastavit do té doby, pokud není nastaven i bit WREN. Po dokončení zápisu je bit WR automaticky nulován a je nastaven příznak EEIF. Nastavení tohoto příznaku může mít za následek přerušení od dokončení zápisu do paměti dat typu EEPROM, pokud je povoleno. Příznak EIF je nutné nulovat softwarově.

Pro zápis do paměti dat typu EEPROM je nutné provést následující kroky:

1. Pokud není použit bod 10, otestujte stav bitu WR pro zjištění, zda neprobíhá dřívější zápis.
2. Zapsat adresu do ukazatele EEADR. V případě 7-bitové adresy ponechte nejvyšší bit v "0".
3. Zapsat data pro zápis do registru EEDATA
4. Nulovat bit EEPGD pro aktivaci paměti dat typu EEPROM
5. Nastavit bit WREN pro povolení zápisové operace
6. Zakázat přerušení
7. Provést požadovanou sekvenci instrukcí:
 - zapsat 55h do EECON2 přes střadač
 - zapsat AAh do EECON2 přes střadač
 - nastavit bit WR
8. Povolit přerušení (pokud jsou používána)
9. Nulovat bit WREN
10. Po dokončení zápisu bude bit WR automaticky nulován a příznak EEIF bude nastaven. Příznak EEIF musí být nulován programově. Pokud není použit bod 1. je možné testovat stav EEIF nebo WR pro zjištění, zda byl zápisový cyklus dokončen.

Příklad 3-2: Rutina zápisu do paměti dat typu EEPROM

požadovaná sekvence	BANKSEL	EECON1	;přepnutí do banky 3
	BTFSC	EECON1, WR	;čekání na dokončení
	GOTO	\$-1	;předchozího zápisu
	BANKSEL	EEADR	přepnutí do banky 2
	MOVF	ADDR, W	;
	MOVWF	EEADR	;nastavení adresy
			;do ukazatele
	MOVF	VALUE,	;
	MOVWF	EEADR	;příprava dat
	BANKSEL	EECON1	;přepnutí do banky 3
	BCF	EECON1, EEPGD	;aktivace paměti dat EEPROM
	BSF	EECON1, WREN	;povolení zápisu do EEPROM
	BCF	INTCON, GIE	;zákaz přerušení
	MOVLW	55h	;
	MOVWF	EECON2	;zápis 55h
	MOVLW	AAh	;
	MOVWF	EECON2	;zápis AAh
	BSF	EECON1, WR	;zahájení zápisu
	BSF	INTCON, GIE	;povolení přerušení
	BCF	EECON1, WREN	;zákaz dalších zápisů

3.5 Čtení z paměti programu typu Flash

Z paměti programu typu Flash lze přečíst operační kód libovolné instrukce v délce 14 bitů. Pro přečtení těchto dat je nezbytné nejprve uložit adresu buňky v paměti programu do registrů EEADR a EEADRH. Dále je nutné nastavit bit EEPGD pro aktivaci paměti programu a nastavit bit RD pro zahájení čtení. Čtení z paměti programu zabere 2 instrukční cykly. Po dobu těchto dvou instrukčních cyklů se nevykonává program, resp. 2 instrukce následující za instrukcí, která aktivuje čtení jsou ignorovány. Tyto dvě instrukce je nutné zapsat do zdrojového programu jako prázdné instrukce NOP. Přečtený operační kód je k dispozici ve dvojici registrů EEDATA a EEDATH již během instrukce, následující za uvedenými ignorovanými instrukcemi. Hodnoty v těchto registrech zůstávají až do přepsání jinou hodnotou nebo do dalšího čtení.

Příklad 3-3: Rutina čtení z paměti programu typu Flash

BANKSEL	EEADRH	;přepnutí do banky 2
MOVF	ADDRH, W	;
MOVWF	EEADRH	;nastavení vyšší
		;adresy do ukazatele
MOVF	ADDRL, W	;
MOVWF	EEADR	;nastavení nižší
		;adresy do ukazatele
BANKSEL	EECON1	;přepnutí do banky 3
BCF	EECON1, EEPGD	;aktivace paměti programu Flash
BSF	EECON1, RD	;zahájení čtení
		;
NOP		;jakékoli
		;dvě instrukce
NOP		;v tomto místě
		;budou ignorovány
BANKSEL	EEDATA	;přepnutí do banky 2
MOVF	EEDATA, W	;načtení nižšího bajtu
MOVWF	DATAL	;
MOVF	EEDATH, W	;načtení vyššího bajtu
MOVWF	DATAH	;

3.6 Mazání paměti programu typu Flash

V paměti programu není možné programově mazat jednotlivá adresová místa. Je možné vymazat pouze celý blok paměti sestávající z 32 po sobě následujících buněk. Větší část paměti lze vymazat pouze pomocí externího programátoru v režimu ICSP.

Blok paměti k vymazání je určen obsahem registrů EEADRH a EEADRL, když určujících je pouze nejvyšších 11 bitů adresy. Nejnižších 5 bitů, odpovídajících 32 adresám uvnitř vybraného bloku je ignorováno.

Mazání je řízeno pomocí registru EECON1. Je nutné nastavit bit EEPGD pro aktivaci paměti programu. Dále je nutné nastavit bit WREN, čímž dojde k povolení zápisu do paměti programu. Nastavením bitu FREE zvolíme mazání paměti.

Před zahájením mazání je nutné dodržet bezpečnostní sekvenci zápisu do registru EECON2. Mazání je zahájeno nastavením bitu WR. Následující dvě instrukce procesor nevykonává žádnou užitečnou činnost a je vhodné na toto místo umístit instrukce **NOP**. Následuje pozastavení činnosti mikropočítače na dobu typicky 2 ms, po kterou probíhá mazání. V této době se nevykonávají žádné instrukce. Nejedná se však o režim Sleep, neboť taktovací oscilátor běží a periferie jsou dále napájeny. Po dokončení mazání pokračuje mikropočítač třetí instrukcí následující po nastavení bitu WR v registru EECON1.

3.6.1 Postup při mazání paměti programu

Při mazání bloku paměti programu je nutné dodržet následující postup:

1. Nastavit adresu bloku paměti do adresových registrů EEADRH a EEADRL
2. Nastavit bity EEPGD, WREN a FREE v registru EECON1
3. Zakázat přerušení
4. Zapsat hodnotu 55h do registru EECON2
5. Zapsat hodnotu AAh do registru EECON2
6. Nastavit bit WR
7. Mikropočítač zahájí mazací cyklus, pokračuje po jeho skončení třetí instrukcí po nastavení WR.

Příklad 3-4: Rutina mazání bloku paměti programu typu Flash

ERASE_ROW	BANKSEL	EEADRH	;přepnutí do banky 2
	MOVF	ADDRH, W	;
	MOVWF	EEADRH	;nastavení vyšší adresy do ukazatele
	MOVF	ADDRL, W	;
	MOVWF	EEADR	;nastavení nižší adresy do ukazatele
	BANKSEL	EECON1	;přepnutí do banky 3
	BSF	EECON1, EEPGD	;aktivace paměti programu Flash
	BSF	EECON1, WREN	;povolení zápisu
	BSF	EECON1, FREE	;povolení mazání
	;		
	BCF	INTCON, GIE	;zákaz přerušení
	MOVLW	55h	;
	MOVWF	EECON2	;zápis 55h
	MOVLW	AAh	;
	MOVWF	EECON2	;zápis AAh
	BSF	EECON1, WR	;zahájení mazání
	NPO		;jakékoli dvě instrukce v tomto místě budou ignorovány,
	NOP		;mikropočítač dokončí mazací cyklus a bude pokračovat v programu
			;třetí následující instrukcí
	BCF	EECON1, FREE	;zákaz mazání
	BCF	EECON1, WREN	;zákaz zápisu
	BSF	INTCON, GIE	;povolení přerušení podle potřeby

3.7 Zápis do paměti programu typu Flash

Do paměti programu lze zapisovat jen v případě, že příslušná část paměti programu není chráněna proti přepsání nastavením konfiguračních bitů WRT1 a WRT2. Zápis do paměti programu lze realizovat jen jako současný zápis bloku čtyř paměťových buněk. Tento blok je tvořen čtyřmi po sobě následujícími adresami, když první adresa má nejméně významné bity rovny 0 ($EEADR<1:0> = 00$). Tyto bity také odpovídají adrese zapisovaných dat ve vyrovnávací paměti.

Před zahájením zápisu je nutné nejprve zapsat obsah celého bloku (čtyři buňky paměti programu) do vyrovnávací paměti (buffer). Z této paměti, která není uživatelsky přístupná, budou data zapsána do čtyř buněk paměti programu najednou poté, co bude zapsán obsah poslední buňky (odpovídá adrese $EEADR<1:0> = 11$).

Situace je znázorněna na obrázku 3-1. Pro zápis je nutné nejprve zapsat adresu do dvojice registrů EEADR a EEADRH a následně data pro zápis do dvojice registrů EEDATA a EEDATH. Toto je nutné čtyřikrát opakovat. Pro zahájení každého zápisu musí být provedena následující požadovaná sekvence:

1. Nastavit bit EEPGD pro aktivaci paměti programu typu Flash
2. Provést požadovanou sekvenci instrukcí:
 - zapsat 55h do EECON2 přes střadač
 - zapsat AAh do EECON2 přes střadač
3. nastavit bit WR

Zápis do paměti programu automaticky vymaže předchozí obsah dotčených čtyř buněk.

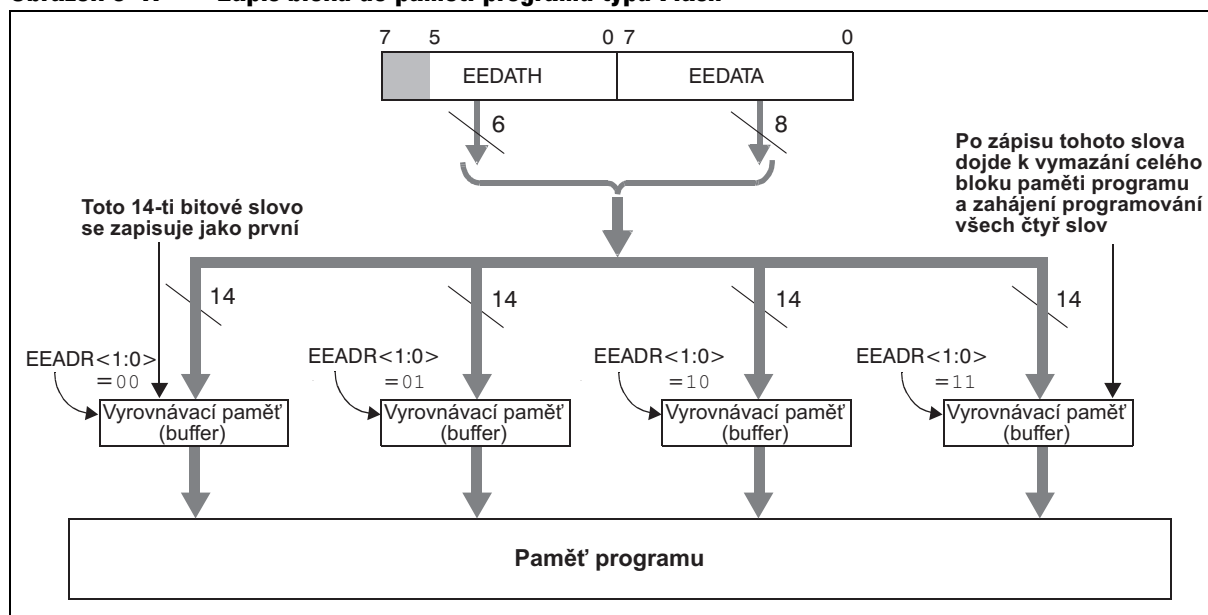
Ve všech registrech vyrovnávací paměti **musí** být uložena správná data. Tato data musí být zapsána do bloku paměti programu. Pokud by došlo k zápisu jenom části bloku, je nezbytné vyčistit ostatní (nezapsané) pozice paměti programu a zapsat je pomocí požadované sekvence. V opačném případě nebude zápis proveden správně.

Zápis do vyrovnávací paměti musí být proveden se vzestupnými adresami počínaje dvojcíslím 00 na konci (00,01,10,11). První tři zápisy budou trvat velice krátkou dobu, protože ve skutečnosti k žádnému fyzickému zápisu do paměti nedochází.

Projeví se to tím, že bit WR bude mít hodnotu "0" již ve druhém instrukčním cyklu následujícím za jeho nastavením. Teprve čtvrtý v pořadí zápis způsobí zahájení fyzického programování paměti. Toto programování zabere přibližně 2 ms. Po tuto dobu se program nevykonává. Dvě následující instrukce za instrukcí nastavující bit WR se ignorují. Tento stav nelze zaměňovat s režimem Sleep. Program se sice nevykonává, ale je v chodu taktovací oscilátor, periferie a výstupní obvody.

Po každém fyzickém zápisu bloku do paměti programu dojde automaticky k nastavení všech bitů všech registrů vyrovnávací paměti (hodnota 3FFFh).

Obrázek 3-1: Zápis bloku do paměti programu typu Flash



Programová obsluha zápisu do paměti programu typu Flash je uvedena na příkladu 3-5. Počáteční adresa pro zápis je uložena do dvojice registrů EEADRH a EEADR.

Čtyři 14-bitová slova pro zápis jsou vyčítána pomocí nepřímého adresování z paměti RWM.

Příklad 3-5: Rutina zápisu do paměti programu typu Flash

;Tato programová rutina předpokládá následující úkony:

- ;1. Blok paměti programu o velikosti 32 instrukcí je vždy vymazán
- ;2. Počáteční adresa pro zápis (s koncovkou 00) je uložena v registrech ADDRH a ADDR L
- ;3. Tento příklad začíná na adrese 0x100
- ;4. 8 bajtů pro zápis (4 instrukce) jsou v RWM na adrese nazvané ARRAY
- ;5. Jedná se pouze o příklad
- ;6. Proměnná word_block je definována v paměti dat

	BANKSEL	EECON1	;přepnutí do banky 2
	BSF	EECON1, EEPGD	;aktivace paměti programu
	BSF	EECON1, WREN	;povolení zápisu
	BCF	EECON1, FREE	;zákaz mazání
	BANKSEL	word_block	
	MOVLW	.4	
	MOVWF	word_block	;příprava 4 instrukcí pro zápis
	BANKSEL	EEADRH	;první zápis bude na adrese 0x100
	MOVLW	0x01	
	MOVWF	EEADRH	;načtení vyššího bajtu adresy
	MOVLW	0x00	
	MOVWF	EEADR	;načtení nižšího bajtu adresy
	BANKSEL	ARRAY	
	MOVLW	ARRAY	;nastavení FSR na začátek dat
	MOVWF	FSR	
LOOP	BANKSEL	EEDATA	
	MOVF	INDF, W	;načtení EEDATA
			;pomocí nepřímého adresování
	MOVWF	EEDATA	
	INCF	FSR, F	;inkrementace ukazatele dat
	MOVF	INDF, W	;načtení EEDATH
			;pomocí nepřímého adresování
	MOVWF	EEDATH	
	INCF	FSR, F	;inkrementace ukazatele dat
	BANKSEL	EECON1	
	MOVLW	0x55	;požadovaná bezpečnostní sekvence
	MOVLW	EECON2	
	MOVLW	0xAA	
	MOVWF	EECON2	
	BSF	EECON1, WR	;nastavením WR dojde k zahájení zápisu
	NOP		;tyto dvě instrukce budou ignorovány
	NOP		
	BANKSEL	EEADR	
	INCF	EEADR, f	;načtení dalších dat
	BANKSEL	word_block	
	DECFSZ	word_block, f	;už jsou zapsány všechny 4 instrukce?
	GOTO	loop	;ještě ne, pokračuj v zapisování
	BANKSEL	EECON1	
	BCF	EECON1, WREN	;ano, zákaz dalších zápisů
	BSF	INTCON, GIE	;povolení přerušení

požadovaná
sekvence

3.8 Ochrana paměti proti náhodnému zápisu

Ochrana proti nežádoucímu zápisu je zajištěna několika různými mechanismy.

Základní ochrana spočívá v nutnosti spustit požadovanou programovou sekvenci.

Současně musí být povolen zápis bitem WREN. Tento bit je po resetu ve stavu "0".

Rovněž během aktivního časovače pro zpoždění resetu po zapnutí napájecího napětí (PWRT) je zápis blokován.

3.9 Práce s pamětí v režimu ochrany kódu proti vyčtení

Pokud je aktivní ochrana kódu proti vyčtení u paměti dat typu EEPROM, do paměti nelze přistupovat z vnějšího prostředí pomocí žádného zařízení. Paměť nelze externě číst ani do ní zapisovat. To neplatí pro procesor mikropočítače, který může pomocí vlastního programu do paměti přistupovat libovolně včetně čtení i zápisu.

Pokud je aktivní ochrana kódu proti vyčtení u paměti programu typu Flash, procesor mikropočítače může tuto paměť číst nebo do ní zapisovat pomocí vlastního programu. Možnosti zápisu lze omezit pomocí konfiguračních bitů WR1 a WRO, kterými lze vymezit oblasti paměti programu které nelze přepsat. Externě však do této paměti přistupovat nelze. Podrobněji viz **kapitulu 15.1** s názvem **Konfigurační bity**.

Tabulka 3-1: Registry, které mají vztah k paměti dat typu EEPROM a paměti programu typu Flash

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR	Hodnota po ostatních druhých resetu
10Ch	EEDATA	Nižší bajt pracovního registru pro čtení/zápis dat								xxxx xxxx	uuuu uuuu
10Dh	EEADR	Nižší bajt ukazatele adresy v paměti dat nebo programu								xxxx xxxx	uuuu uuuu
10Eh	EEDATH	—	—	Vyšší bajt pracovního registru pro čtení/zápis dat						--xx xxxx	--uu uuuu
10Fh	EEADRH	—	—	—	—	Vyšší bajt ukazatele adresy v paměti dat nebo programu				---- xxxx	---- uuuu
18Ch	EECON1	EEPGD	—	—	FREE	WRERR	WREN	WR	RD	x--x x000	x--x q000
18Dh	EECON2	Ochranný registr pro zápis požadované sekvence (nejedná se o skutečný registr)								---- ----	---- ----
0Dh	PIR2	OSFIF	CMIF	—	EEIF	—	—	—	—	00-0 ----	00-0 ----
8Dh	PIE2	OSFIE	CMIE	—	EEIE	—	—	—	—	00-0 ----	00-0 ----

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, q = hodnota závisí na podmínkách, stínované buňky nejsou použity pro práci s pamětí dat nebo pamětí programu.

PIC16F87/88

Poznámky:

4 KONFIGURACE TAKTOVACÍHO OSCILÁTORU

4.1 Typy taktovacího oscilátoru

Mikropočítač může být taktován pomocí osmi různých taktovacích oscilátorů.

Uživatel volí druh oscilátoru programově pomocí bitů FOSC2, FOSC1 a FOSC0.

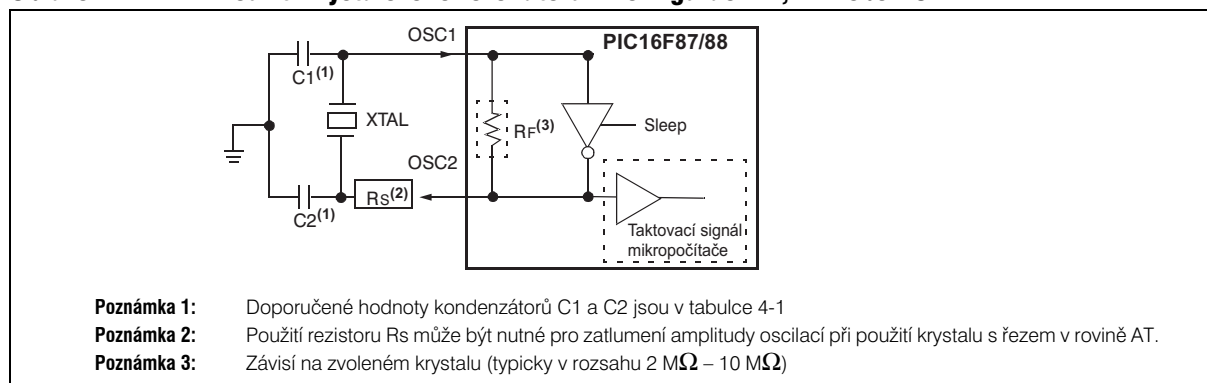
Taktovací oscilátory uvedené pod číslem 5 – 8 jsou u tohoto mikropočítače novinkou v řadě PIC 16xxx.

1. LP krystalový oscilátor s nízkou spotřebou
2. XT krystalový/keramický rezonátor
3. HS rychlý krystalový/keramický rezonátor
4. RC externí oscilátor s výstupem Fosc/4 na vývod RA6
5. RCIO externí oscilátor bez výstupu, vývod RA6 je univerzální vstup/výstup
6. INTIO1 interní oscilátor s výstupem na RA6, vývod RA7 je univerzální vstup/výstup
7. INTIO1 interní oscilátor bez výstupu, vývody RA6 a RA7 jsou univerzální vstupy/výstupy
8. ECIO vnější taktovací zdroj, vývod RA6 je univerzální vstup/výstup

4.2 Použití krystalového oscilátoru nebo keramického rezonátoru

Podle zvolené frekvence je konfigurován typ LP, XT nebo HS. Ve všech případech se oscilátor připojuje k součástce pomocí dvou vývodů OSC1/CLKI a OSC2/CLKO, viz schéma na obr. 14-1 a 4-2. Vývod OSC1 plní funkci vstupu, OSC2 plní funkci výstupu vnitřního obvodu oscilátoru. Taktovací oscilátor je uzpůsoben pro krystaly s paralelní rezonancí.

Obrázek 4-1: Použití krystalového rezonátoru v konfiguraci LP, XT nebo HS



Tabulka 4-1: Doporučené hodnoty kondenzátorů pro krystalové oscilátory (jsou uvedeny pouze vybrané hodnoty kmitočtů)

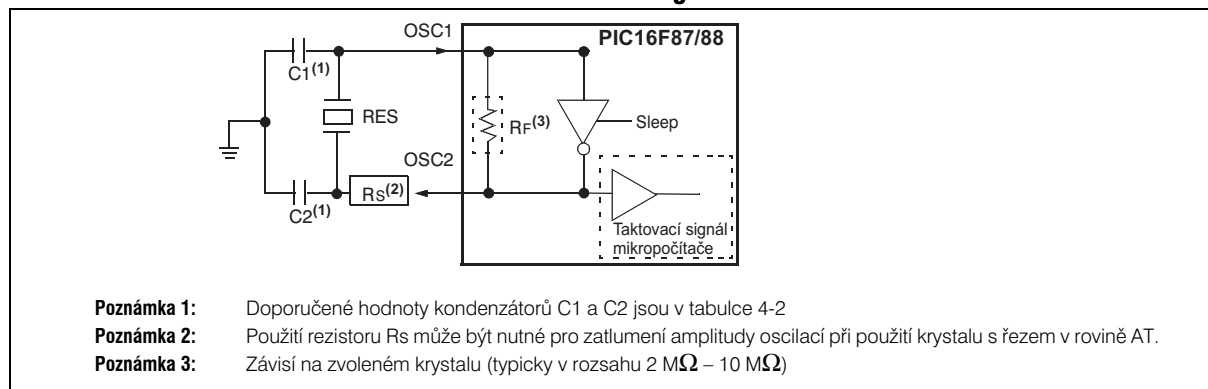
Typ	Kmitočet	Doporučené hodnoty	
		C1	C2
LP	32 kHz	33 pF	33 pF
XT	200 kHz	56 pF	56 pF
	1 MHz	15 pF	15 pF
	4 MHz	15 pF	15 pF
HS	4 MHz	15 pF	15 pF
	8 MHz	15 pF	15 pF
	20 MHz	15 pF	15 pF

Uvedené hodnoty kondenzátorů jsou testovány s ohledem na rychlé ustálení oscilací.

Hodnoty nejsou zaručené, jen doporučené. Skutečné hodnoty mohou záviset na napájecím napětí, teplotě a dalších vlivech.

- Poznámka 1:** Při použití vyšších hodnot kondenzátorů bývá sice oscilátor stabilnější, zároveň se však prodlužuje doba ustálení oscilací po zapnutí.
- 2:** Doporučené hodnoty kondenzátorů leží v uvedených rozsazích. Každý druh rezonátoru má své vlastní charakteristiky, udávané svým výrobcem.
- 3:** Použití rezistoru Rs může být nutné pro zatlumení amplitudy oscilací v režimu XT nebo HS
- 4:** Ověřujte použitelnost konkrétního oscilátoru v celém rozsahu napájecího napětí a provozních teplot.

Obrázek 4-1: Použití keramického rezonátoru v konfiguraci XT nebo HS



Tabulka 4-2: Doporučené hodnoty kondenzátorů pro keramické rezonátory (jsou uvedeny pouze vybrané hodnoty kmitočtů)

Doporučené hodnoty			
Typ	Kmitočet	OSC1	OSC2
XT	455 kHz	56 pF	56 pF
	2.0 MHz	47 pF	47 pF
	4.0 MHz	33 pF	33 pF
HS	8.0 MHz	27 pF	27 pF
	16.0 MHz	22 pF	22 pF

Uvedené hodnoty kondenzátorů jsou testovány s ohledem na rychlé ustálení oscilací. Hodnoty nejsou zaručené, jen doporučené. Skutečné hodnoty mohou záviset na napájecím napětí, teplotě a dalších vlivech.

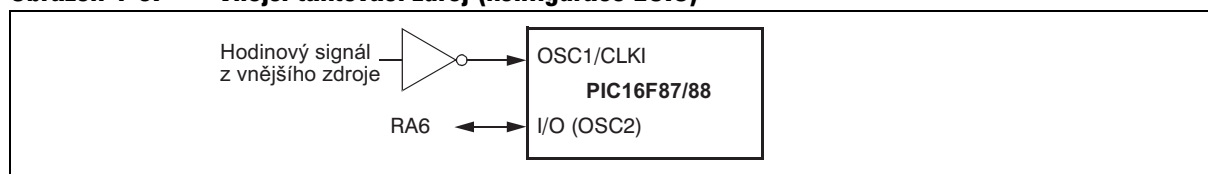
Poznámka: Při použití taktovacího kmitočtu vyššího než 3,5 MHz používejte raději konfiguraci HS. V konfiguraci HS je oscilátor funkční v celém rozsahu napájecího napětí mikropočítače. V některých případech není vyloučené přebuzení vstupu oscilátoru. Toto lze ošetřit rezistorem o doporučené hodnotě 330 Ω .

4.3 Vnější taktovací zdroj

Tento druh taktovacího oscilátoru předpokládá přivedení vnějšího taktovacího signálu na vývod OSC1. Tento oscilátor nevyžaduje použití časovače pro zpoždění resetu při zapnutí napájecího napětí nebo po probuzení ze stavu Sleep.

Při této konfiguraci je možné využít vývod RA6 (OSC2) jako univerzální vstup/výstup. Na obrázku 4-3 je uvedeno zapojení vnějšího taktovacího zdroje.

Obrázek 4-3: Vnější taktovací zdroj (konfigurace ECIO)



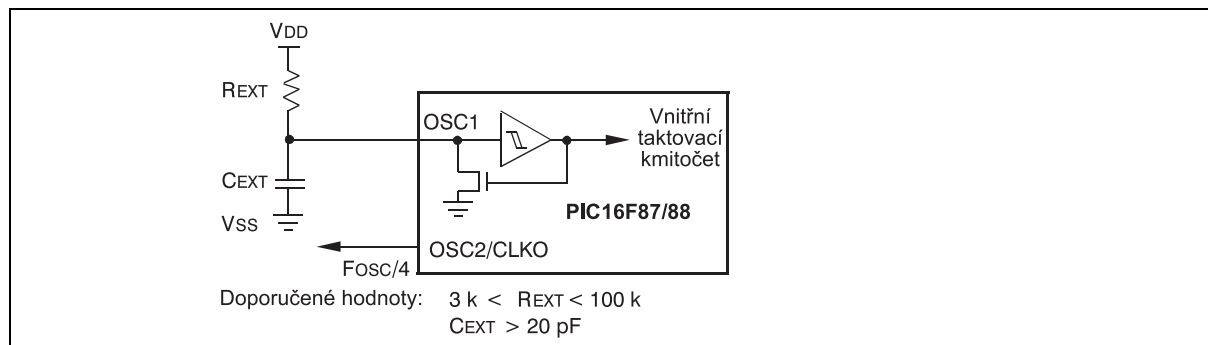
4.4 RC oscilátor

V aplikacích, které nejsou náročné na přesnost taktovacího oscilátoru je možné použít levný oscilátor sestavený z vnějšího rezistoru a kondenzátoru. Výsledný kmitočet je závislý na odporu rezistoru (R_{EXT}) a kapacitě kondenzátoru (C_{EXT}).

Taktovací kmitočet je však rovněž závislý na velikosti napájecího napětí a na teplotě. Kmitočet oscilátoru je navíc proměnný u každého mikroprocesoru vlivem výrobních tolerancí. Návrhář musí počítat i s rozptylem parametrů u vnějších součástek (R_{EXT} a C_{EXT}). Zejména při použití malých kondenzátorů (C_{EXT}) se může významně uplatňovat parazitní kapacita pouzdra, případně kapacita plošného spoje.

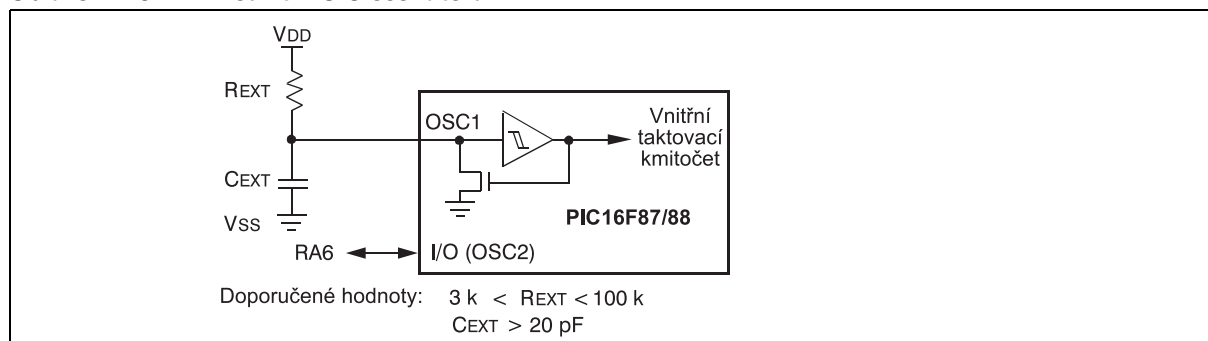
Schéma zapojení vnějšího RC oscilátoru je uvedeno na obrázku 4-4.

Obrázek 4-4: Použití RC oscilátoru



Pokud potřebujeme použít vývod RA6 pro jiné účely než je výstup taktovacího oscilátoru, můžeme použít konfiguraci RCIO. Jedná se o shodný oscilátor s RC až na to, že vývod RA6 plní funkci univerzálního vstupu/výstupu. Zapojení je uvedeno na obrázku 4-5.

Obrázek 4-5: Použití RCIO oscilátoru



4.5 Modul vnitřního taktovacího oscilátoru

Mikropočítač je vybaven modulem vnitřního taktovacího oscilátoru, který obsahuje dva rozdílné oscilátory, z nichž každý může být použit jako zdroj taktovací frekvence. Vnitřní oscilátor umožňuje činnost mikropočítače bez jakýchkoli dalších součástek připojených na vývody OSC1 a OSC2.

Hlavní oscilátor (INTOSC) pracuje na frekvenci 8 MHz a je možné jej použít přímo jako zdroj taktovací frekvence. K dispozici je rovněž programovatelná předdělička k tomuto oscilátoru, která umožňuje snížení taktovací frekvence v šesti krocích v rozsahu od 125 kHz do 4 MHz.

Druhý oscilátor (INTRC) pracuje na frekvenci 31,25 kHz (odpovídající perioda typicky 32 μs). Tento oscilátor je použit jako zdroj taktovací frekvence při volbě INTRC nebo pokud je povolen jeden z následujících systémů:

- Časovač pro zpoždění resetu při zapnutí napájecího napětí
- Ochranný časovač Watchdog
- Dvou rychlostní rozběh taktovacího oscilátoru
- Ochrana průběhu taktovacího kmitočtu

Tyto vlastnosti jsou podrobně rozebrány v kapitole 15.0.

Všechny uvedené druhy oscilátoru jsou ovládány pomocí řídicích bitů IRCF v registru OSCCON.

Poznámka: Pokud je v této dokumentaci vnitřní oscilátor označen jako konfigurace INTRC, může se jednat obecně o použití modulu vnitřního taktovacího oscilátoru nezávisle na konkrétní volbě INTOSC (8 MHz), INTOSC s předděličkou nebo INTRC (32 kHz).

4.5.1 Režimy vnitřního taktovacího oscilátoru INTRC

Použití vnitřního oscilátoru INTRC umožňuje ušetřit dva vývody mikropočítače pro jiné využití. Jsou k dispozici dvě různé konfigurace tohoto oscilátoru:

- režim INTIO1 umožňuje použití vývodu OSC1/RA7 jako univerzální číslicový vstup/výstup.
Na vývodu OSC2/RA6 je přítomna taktovací frekvence dělená čtyřmi – $F_{osc}/4$
- režim INTIO2 umožňuje použití obou vývodů OSC1/RA7 a OSC2/RA6 jako univerzální číslicové vstupy.

4.5.2 Registr OSCTUNE

Frekvence vnitřního taktovacího oscilátoru je nastavena výrobcem, uživatel může taktovací frekvenci pouze doladit pomocí registru OSCTUNE. Přeladitelnost oscilátoru je $\pm 12\%$ s lineárním průběhem.

Při zápisu do registru OSCTUNE dojde ke změně taktovací frekvence. Při použití vnitřního oscilátoru INTRC je dosaženo nové hodnoty nejpozději do 8 cyklů oscilátoru (přibližně $8 \cdot 32 \mu s = 256 \mu s$). Při použití oscilátoru INTOSC dojde ke změně frekvence do 1 ms. Program se vykonává v obou případech průběžně dál. Nelze zjistit pomocí žádného příznaku, že došlo k dokončení změny frekvence. Systémy mikropočítače, které jsou závislé na oscilátoru 31,25 kHz jako jsou Watchdog a podobně jsou rovněž přeladěním příslušně ovlivněny.

Registr 4-1: Registr OSCTUNE – doladovací registr vnitřního taktovacího oscilátoru (adresa 90h)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0
bit 7		bit 0					

bit 7-6 **nepoužito:** při čtení '0'

bit 5-0 **TUN5 – TUN0:** bity pro doladění taktovací frekvence
011111 = nejvyšší frekvence
011110 =
 •
 •
 •
000001 =
000000 = střed frekvence
111111 =
 •
 •
 •
100000 = nejnižší frekvence

Význam:

R = bit pro čtení	W = bit pro zápis	U = nepoužitý bit, čteno jako '0'
–n = hodnota po resetu POR	'1' = bit je nastaven	'0' = bit je v nule x = hodnota bitu je neznámá

4.6 Zdroje taktu a přepínání taktovacích oscilátorů

Mikropočítač je vybaven možností přepínat zdroj taktovacího signálu během výkonu programu. To má za následek především možnost řídit spotřebu mikropočítače v různých situacích.

Jsou celkem tři různé zdroje taktovacího signálu.

- Primární taktovací oscilátor
- Sekundární taktovací oscilátor
- Vnitřní taktovací oscilátor (INTRC)

Primární taktovací oscilátor zahrnuje vnější krystalový nebo keramický oscilátor, všechny režimy vnějšího RC oscilátoru, vnější taktovací zdroj a dále blok vnitřního oscilátoru. Primární taktovací oscilátor je použitý vždy po resetu POR v závislosti na konfiguračním slově mikropočítače.

Sekundární taktovací oscilátor není připojen pomocí vývodů OSC1 a OSC2. Jedná se o takové typy externích oscilátorů, které jsou použity jako taktovací oscilátor jen v režimu s řízenou spotřebou.

Mikropočítač umožňuje použít čítač/časovač Timer1 jako zdroj taktovacího kmitočtu. V takovém případě hovoříme o použití sekundárního taktovacího oscilátoru. Tento taktovací oscilátor běží i při výkonu instrukce SLEEP.

Hodinový krystal o frekvenci 32,768 kHz se připojuje v konfiguraci LP na vývody OSC1 a OSC2 pomocí vazebních kondenzátorů proti zemi. Tento typ oscilátoru je popsán v **kapitole 7.6**.

Taktovací oscilátor může být použit jako primární taktovací oscilátor s řízenou spotřebou. Vnitřní oscilátor INTRC o kmitočtu 31,25 kHz je vždy použit jako zdroj hodinového signálu pro časovač Watchdog, časovač pro zpoždění resetu, hlídací obvod taktovacího oscilátoru a systém dvou rychlostního rozběhu taktovacího oscilátoru.

Situace je znázorněna na obrázku 4-6. Podrobnosti o použití Timer1 jsou uvedeny v **kapitole 7.0**. Konfigurační bity jsou popsány v **kapitole 15.1**.

4.6.1 Registr OSCCON

Pomocí registru OSCCON lze nastavit a ovládat systém taktovacího oscilátoru mikropočítače při normálním běhu programu i v režimu s řízenou spotřebou.

Bity SCS1 a SCS0 určují druh oscilátoru v režimu s řízenou spotřebou. Pokud jsou tyto bity v nule, je kmitočet odvozen z taktovacího kmitočtu primárního oscilátoru, který je určen konfiguračními bity FOSC2 – FOSC0 v konfiguračním slově mikropočítače. Pokud jsou tyto bity nastaveny, je takt odvozen od Timer1 v případě, že SCS0=1 nebo od modulu vnitřního oscilátoru v případě, že SCS1=1. Po resetu jsou tyto bity vždy nulovány.

Poznámka: Instrukce bezprostředně následující za instrukcí měnící bity SCS0 nebo SCS1 je vykonána rychlostí odpovídající předcházejícímu nastavení. Toto je nutné brát v úvahu při ladění časově kritických částí programového kódu.

Bity IRCF2:IRCF0 určují taktovací frekvenci bloku vnitřního oscilátoru. Jako zdroj taktovací frekvence je možné určit buď vnitřní oscilátor INTRC pracující na frekvenci 31,25 kHz, vnitřní taktovací oscilátor INTOSC pracující a frekvenci 8 MHz nebo jeden ze šesti výstupů předděličky zařazené za oscilátor INTOSC v rozsahu 125 kHz – 4 MHz. Změnou těchto bitů dojde okamžitě ke změně taktovací frekvence.

Bity OSTS a IOFS jsou příznakové bity taktovacího oscilátoru. Bit IOFS indikuje, zda je frekvence dodávána z příslušného zdroje stabilní, bit OSTS indikuje přetečení rozběhového časovače taktovacího oscilátoru.

4.6.2 Přepínání zdroje taktu

Zdroj taktu mikropočítače může být změněn z následujících důvodů:

- Pokud došlo k nastavení bitu FCMEN v konfiguračním slově mikropočítače, mikropočítač je taktován z primárního zdroje taktovacího kmitočtu a tento zdroj vypadl, automaticky dojde k přepnutí na vnitřní taktovací RC oscilátor.
- Pokud došlo k nastavení bitu FCMEN v konfiguračním slově mikropočítače, mikropočítač je taktován pomocí T1OSC a tento zdroj vypadl, automaticky dojde k přepnutí na vnitřní taktovací RC oscilátor.
- Následně po signálu reset, pokud je taktovací oscilátor nastaven v režimu dvou rychlostního rozběhu, dojde k přepnutí z vnitřního RC taktovacího oscilátoru na primární taktovací oscilátor určený nastavením příslušných bitů v konfiguračním slově mikropočítače.
- Při probuzení ze stavu Sleep pomocí přerušení nebo přetečením časovače Watchdog se současným povolením dvou rychlostního rozběhu taktovacího oscilátoru. Pokud je jako primární zdroj taktovacího oscilátoru zvolen režim XT, HS nebo LP, bude primární zdroj připojen teprve po 1024 periodách (OST) a dalších 8 periodách tohoto oscilátoru. Toto platí jen pokud jsou bity SCS v nule.
- Nastavení bitů SCS bylo změněno
- Nastavení bitů IRCF bylo změněno

Poznámka: Protože při každém resetu jsou nulovány bity SCS, nemůže dojít k přepnutí taktovacího oscilátoru dokud je povolen systém dvou rychlostního rozběhu taktovacího oscilátoru a primární taktovací oscilátor je XT, HS nebo LP. Mikropočítač bude čekat na dosažení stability primárního taktovacího oscilátoru.

4.6.3 Přepínání taktu a časovač Watchdog

Při přepínání taktovacího oscilátoru je zakázán časovač Watchdog. Důvodem je užití společného posuvného registru jak pro časovač Watchdog tak pro časovač pro zpoždění resetu, používaný při přepínání taktovacího oscilátoru.

Poznámka: Časovač pro rozběh taktovacího oscilátoru OST je použit jen při přepnutí na taktovací oscilátor XT, HS nebo LP.

Jakmile je přepnutí taktovacího oscilátoru dokončeno a takt mikropočítače je zabezpečen novým oscilátorem, je činnost obvodu Watchdog opět povolena s tím, že obsah časovače Watchdog je automaticky vynulován. To umožňuje uživateli synchronizovat nový pracovní cyklus časovače Watchdog s přechodem na novou taktovací frekvenci.

Registr 4-2: Registr OSCCON – řídicí registr vnitřního taktovacího oscilátoru (adresa 8Fh)

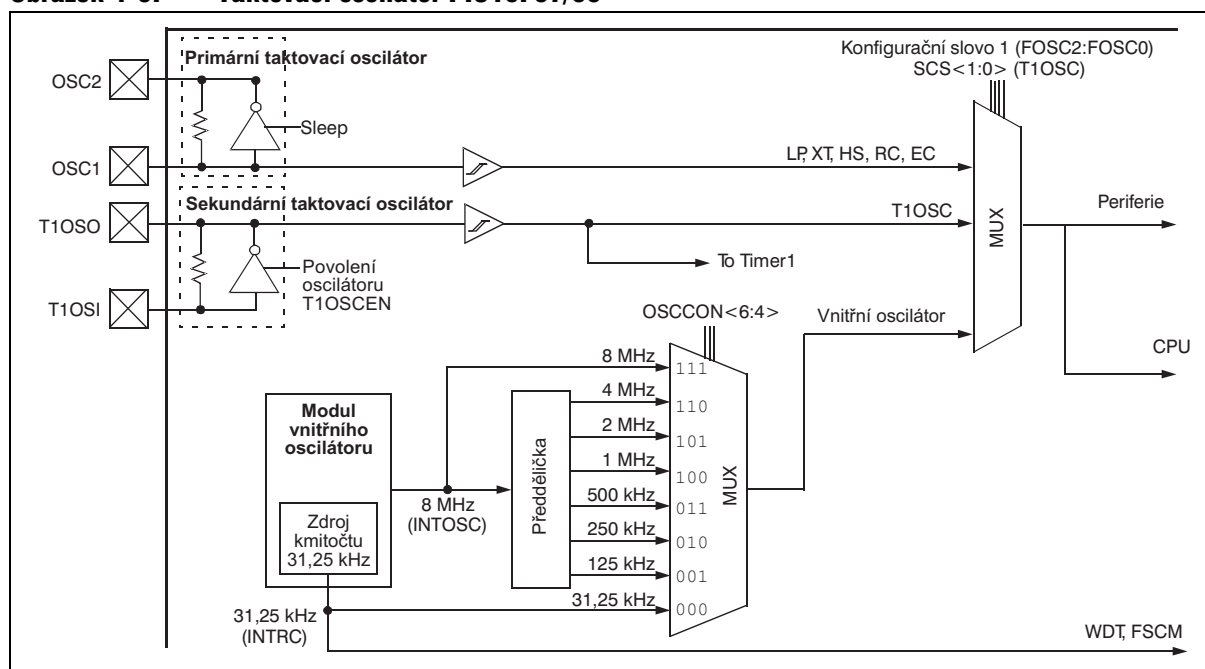
U-0	R/W-0	R/W-0	R/W-0	R-0	R/W-0	R/W-0	R/W-0
—	IRCF2	IRCF1	IRCF0	OSTS ^(*)	IOFS	SCS1	SCS0
bit 7							bit 0

- bit 7 **nepoužito:** při čtení '0'
- bit 6-4 **IRCF2 – IRCF0:** bity pro volbu frekvence vnitřního taktovacího oscilátoru
- bit 3 **OSTS:** Příznak přetečení rozběhového časovače
1 = mikropočítač zahájil výkon programu s primárním taktovacím oscilátorem
0 = mikropočítač zahájil výkon programu se sekundárním taktovacím oscilátorem T1OSC nebo INTRC
- bit 2 **IOFS:** Příznak stability taktovací frekvence
1 = taktovací frekvence je stabilní
0 = taktovací frekvence není stabilní
- bit 1-0 **SCS1-SCS0:** Bity výběru taktovacího oscilátoru
00 = druh oscilátoru je určen bity FOSC v konfiguračním slově mikropočítače
01 = T1OSC je použit jako taktovací oscilátor
10 = modul vnitřního oscilátoru je použit jako taktovací oscilátor
11 = nepoužito

Význam:

R = bit pro čtení	W = bit pro zápis	U = nepoužitý bit, čteno jako '0'
-n = hodnota po resetu POR	'1' = bit je nastaven	'0' = bit je v nule x = hodnota bitu je neznámá

Obrázek 4-6: Taktovací oscilátor PIC16F87/88



4.6.4 Modifikace bitů IRCF

Bity IRCF lze měnit kdykoli nezávisle na tom, jaký typ oscilátoru je zdrojem taktovacího kmitočtu mikropočítače. Uživatel může měnit programem taktovací frekvenci mikropočítače během výkonu programu modifikací bitů IRCF v registru OSCCON. Události které nastanou po změně bitů IRCF závisí na stavu těchto bitů před modifikací. Pokud je před modifikací v činnosti oscilátor INTRC (31,25 kHz, $IRCF<2:0> = 000$) a dojde k nastavení odlišnému od stavu "000", je aktivováno zpoždění typicky 4 ms. Program je vykonáván na vyšší frekvenci než je požadovaná až do doby, než se nová taktovací frekvence ustálí. Po tuto dobu je možné testovat hodnotu bitu IOFS, který indikuje dosažení ustáleného stavu taktovací frekvence. Časově kritické části programu je vhodné vykonávat pouze v době, kdy je taktovací frekvence stabilní.

Pokud jsou bity IRCF modifikovány z jakéhokoli jiného stavu (31,25 kHz, $IRCF<2:0> \neq 000$), není nutné žádné zpoždění (typicky 4 ms). Nová frekvence oscilátoru INTOSC je stabilní po **osmém** sestupné hraně nové taktovací frekvence. Bit IOFS je nastaven po dosažení stabilní frekvence.

Poznámka: Je nutná opatrnost při modifikaci bitů IRCF pomocí instrukcí BSF a BCF. V některých případech by tím mohlo být dosaženo taktovacích frekvencí ležících mimo rozsah povoleného napájecího napětí. Například při napájecím napětí 2,0 V není možné nastavit všechny bity IRCF, protože by to odpovídalo taktu 8 MHz, což je nepřipustná kombinace.

4.6.5 Postup při přepínání taktovací frekvence

Při přepínání vnitřního taktovacího oscilátoru je nutné dodržet následující postup.

- Pro nastavení před změnou: 31,25 kHz, $IRCF<2:0> = 000$
 1. Provedte modifikaci bitů IRCF na požadovanou kombinaci
 2. Vnitřní řídicí obvod čeká na sestupnou hranu taktovací frekvence, jakmile je tato hrana detekována, dojde k přidržení signálu CLK0 v log.0.
 3. Vnitřní řídicí obvod čeká 8 sestupných hran nové taktovací frekvence, po jejich detekci obnoví signál CLK0 již na nové taktovací frekvenci.
 4. Po dobu přibližně 4 ms je podržen v nule příznak stability taktovací frekvence IOFS. Tento příznak by měly testovat všechny časově kritické části programu.
 5. Přepnutí je dokončeno
- Pro nastavení před změnou: jedna z předdělíček oscilátoru INTOSC, $IRCF<2:0> \neq 000$
 1. Provedte modifikaci bitů IRCF na požadovanou kombinaci
 2. Vnitřní řídicí obvod čeká na sestupnou hranu taktovací frekvence, jakmile je tato hrana detekována, dojde k přidržení signálu CLK0 v log.0.
 3. Vnitřní řídicí obvod čeká 8 sestupných hran nové taktovací frekvence, po jejich detekci obnoví signál CLK0 již na nové taktovací frekvenci.
 4. Přepnutí je dokončeno

Pro nastavení před změnou: jedna z předděliček oscilátoru INTOSC, IRCF<2:0> ≠ 000

1. Provedte modifikaci bitů IRCF na požadovanou kombinaci
2. Vnitřní řídicí obvod čeká na sestupnou hranu taktovací frekvence, jakmile je tato hrana detekována, dojde k přidržení signálu CLK0 v log.0.
3. Vnitřní řídicí obvod čeká 8 sestupných hran nové taktovací frekvence, po jejich detekci obnoví signál CLK0 již na nové taktovací frekvenci.
4. Je nastaven bit IOFS
5. Přepnutí je dokončeno

4.6.6 Zpoždění taktovacího oscilátoru v různých situacích.

V tabulce 4-3 jsou uvedeny různé doby zpoždění pro různé situace, jako je zapnutí napájecího napětí, probuzení ze stavu Sleep nebo přepnutí taktovací frekvence.

Tabulka 4-3: Zpoždění taktovacího oscilátoru

Přepnutí taktovací frekvence		Taktovací kmitočet	Zpoždění taktovacího oscilátoru	Poznámky
Z	Na			
Sleep/POR	INTRC T1OSC	31,25 kHz 32,768 kHz	Náběh CPU	Po probuzení ze stavu Sleep nebo resetu po zapnutí napájecího napětí dojde k zapnutí CPU a výkonu programu.
	INTOSC/ INTOSC Předdělička	125 kHz – 8 MHz	Přibližně 4 ms a náběh CPU (1)	
INTRC/Sleep	EC, RC	DC – 20 MHz		
INTRC (31,25 kHz)	EC, RC	DC – 20 MHz		
Sleep	LP, XT, HS	32,768 kHz – 20 MHz	1024 Taktů (OSC)	Po změně nastavení INTRC musí následovat zpoždění vyvolané časovačem OST
INTRC (31,25 kHz)	INTOSC/ INTOSC Předdělička	125 kHz – 8 MHz	4 ms (přibližně)	Viz kapitolu 4.6.4

Poznámka: Při taktovací frekvenci 1 MHz je doba náběhu CPU v rozsahu 5-10 μs.

4.7 Režimy s řízenou spotřebou

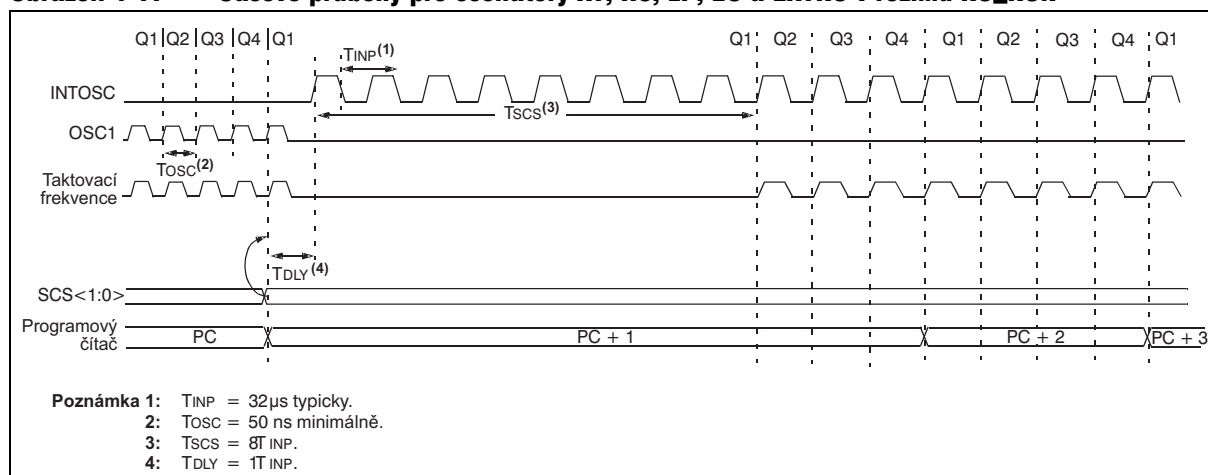
4.7.1 Režim RC_RUN

Pokud není zdrojem primární taktovací frekvence vnitřní oscilátor INTRC, je možné tento oscilátor použít jako sekundární a přepnout taktovací frekvenci z důvodu snížení spotřeby. Po nulování bitu OSTS dojde k přepnutí taktovací frekvence z primární na sekundární a současně k vypnutí primárního taktovacího oscilátoru z důvodu snížení spotřeby. Primárním zdrojem může být v případě konfigurace $SCS<1:0> = "00"$ jedno z nastavení dané konfigurací bitů IRCF nebo v případě konfigurace $SCS<1:0> = "01"$ oscilátor čítače/časovače Timer1. K uvedenému přepnutí nemůže dojít tehdy, pokud je primárním zdrojem taktovací frekvence přímo vnitřní oscilátor INTRC.

Pokud při změně bitů SCS není zdrojem taktovací frekvence INTRC (31,25 kHz) a konfigurační bity IRCF v registru OSCCON jsou nastaveny pro použití jiného zdroje než je INTRC, taktovací frekvence nemusí být stabilní okamžitě po změně. Dosažení stabilní taktovací frekvence je indikováno nastavením bitu IOFS po přibližně 4 ms.

Po přepnutí taktovací frekvence je nulován příznak OSTS indikující přechod do režimu s řízenou spotřebou a současně upozorňující na to, že zdrojem taktovací frekvence mikropočítače není primární taktovací oscilátor. Vnitřní taktovací signál mikropočítače je během taktu Q1 přidržen ve stavu log.0 do doby, než uplyne 8 sestupných hran oscilátoru INTRC. Po této době je taktovací signál uvolněn a mikropočítač pokračuje ve výkonu programu. Situace je znázorněna na obrázku 4-7.

Obrázek 4-7: Časové průběhy pro oscilátory XT, HS, LP, EC a EXTRC v režimu RC_RUN



4.7.2 Režim SEC_RUN

Mikropočítač může být konfigurován pro taktování ze zdroje T1OSC, což je oscilátor čítače/časovače Timer1 připojený k vývodům T1OSO a T1OSI s hodinovým krystalem 32,768 kHz. Jedná se o stejnou konfiguraci oscilátoru jako v případě časování Timer1 pomocí nízko-příkonového oscilátoru, viz **kapitulu 7.6**. Při konfigurování bitů SCS pro taktování ze zdroje T1OSC dojde k přepnutí taktovací frekvence. Je nulován příznak OSTS, je přepnuta taktovací frekvence z primárního zdroje podle konfigurace bitů SCS a FOSC na sekundární zdroj, tvořený externím nízko příkonovým oscilátorem čítače/časovače Timer1 a dojde k vypnutí primárního zdroje taktovací frekvence z důvodu snížení spotřeby.

Po přepnutí taktovací frekvence je vnitřní taktovací signál mikropočítače je během taktu Q1 přidržen ve stavu log.0 do doby, než uplyne 8 sestupných hran oscilátoru T1OSC. Po této době je taktovací signál uvolněn a mikropočítač pokračuje ve výkonu programu. Situace je znázorněna na obrázku 4-8. Současně dojde k nastavení příznaku T1RUN v registru T1CON indikujícího, že zdrojem taktovací frekvence mikropočítače je T1OSC.

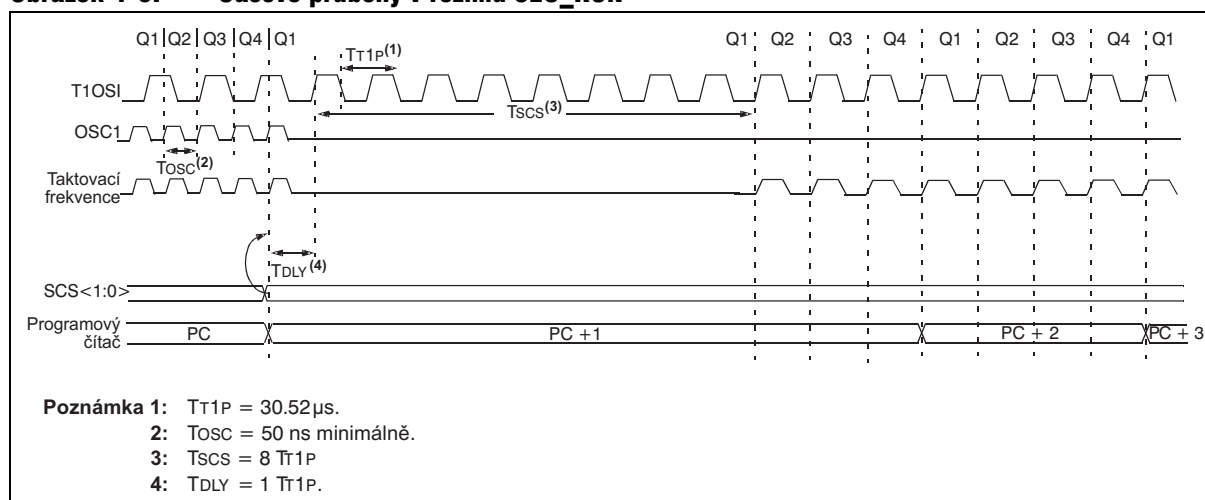
Poznámka 1: Před přepnutím taktovací frekvence musí být oscilátor T1OSC povolen nastavením bitu T1OSCEN a uživatel musí zajistit, aby v okamžiku přepnutí již bylo dosaženo stability tohoto oscilátoru.

2: Při $T1OSCEN=0$ mohou nastat následující stavy:

Původní nastavení $SCS<1:0>$	Nové nastavení $SCS<1:0>$	Výsledné nastavení $SCS<1:0>$
00	01	00 – beze změny
00	11	10 – INTRC
10	11	10 – beze změny
10	01	00 – oscilátor je definován pomocí bitů FOSC <2:0>

K přepnutí taktovací frekvence dojde v případě, že výsledné nastavení bitů SCS je odlišné od původního.

Obrázek 4-8: Časové průběhy v režimu SEC_RUN



4.7.3 Návrat k primárnímu zdroji taktovací frekvence

Návrat k primárnímu zdroji taktovací frekvence je možný nulováním bitů SCS.

Události, které následují po tomto přepnutí závisí na nastavení konfiguračních bitů FOSC v konfiguračním slově mikropočítače. Pokud je primární zdroj taktovacího kmitočtu konfigurován krystalový oscilátor (HS, XT nebo LP), dojde k přechodu na primární taktovací zdroj po 1024 taktech. Tato prodleva je nezbytná při každém zapnutí krystalového oscilátoru. Po tuto dobu mikropočítač vykonává program podle původního taktu.

K přepnutí na primární zdroj taktovací frekvence dojde až po ustálení krystalového oscilátoru. Tato skutečnost je indikována příznakem OSTs. Jakmile dojde k nastavení tohoto příznaku, uživatel si může být jistý, že mikropočítač je taktován z primárního zdroje taktovací frekvence se stabilním průběhem.

Během samotného přepnutí mezi sekundárním a primárním zdrojem taktovací frekvence je na dobu 8 sestupných hran taktovací frekvence podržen v taktu Q1v log.0 taktovací signál. Po této době je taktovací signál uvolněn a mikropočítač pokračuje ve výkonu programu. Situace je znázorněna na obrázku 4-10.

V režimu SEC_RUN způsobí nulování bitu T1OSCEN v registru T1CON nulování bitu SCS<0>, což způsobí změnu nastavení bitů SCS<1:0> do stavu "00" nebo "10" v závislosti na tom, v jakém stavu byl bit SCS<1>. Ačkoliv je bit T1OSCEN nulován, oscilátor T1OSC je dále povolen a program se dále vykonává v taktu tohoto oscilátoru dokud neuplyne doba přetečení časovače pro rozběh krystalového oscilátoru OST. Teprve v tuto chvíli dojde ke skutečnému návratu k primárnímu zdroji taktovacího kmitočtu a oscilátor T1OSC je následně vypnut.

Poznámka: Pokud je primárním zdrojem taktovací frekvence konfigurován oscilátor RC nebo EC, dojde po přepnutí ze sekundárního zdroje na primární k vnitřnímu zpoždění o délce 5 – 10 μs .

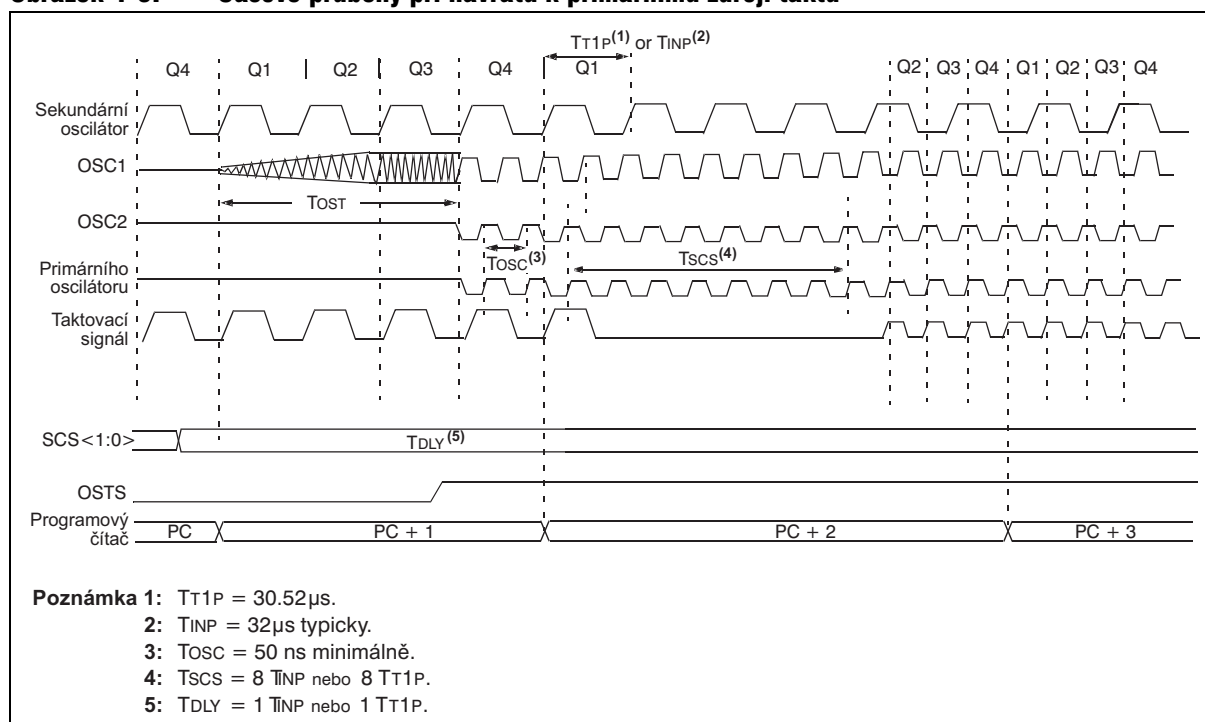
4.7.3.1 Postup při návratu k primárnímu zdroji taktovací frekvence

Návrat k primárnímu zdroji taktovací frekvence z režimů RC_RUN nebo SEC_RUN je možný přepnutím bitů SCS<1:0> do stavu "00" nebo nulováním bitu T1OSCEN v registru T1CONv případě, že sekundárním zdrojem taktovací frekvence byl oscilátor T1OSC.

V obou případech je postup následující:

1. Pokud je primární zdroj taktovací frekvence konfigurován jako EC, RC nebo INTRC, je prodleva způsobená časovačem pro ustálení krystalového oscilátoru vynechána. V tomto případě pokračujte bodem 3.
2. Pokud je primární zdroj taktovací frekvence konfigurován jako HS, XT nebo LP je aktivní časovač pro rozběh krystalového oscilátoru. Je nutné počkat po dobu 1024 taktů tohoto oscilátoru.
3. V následujícím taktu Q1 mikropočítač podrží taktovací signál ve stavu log.0.
4. Mikropočítač zůstane v tomto stavu po dobu osmi následujících sestupných hran taktovací frekvence.
5. Po uplynutí této doby mikropočítač pokračuje ve výkonu programu podle primárního taktovacího oscilátoru.
6. Pokud byl sekundárním zdrojem taktovací frekvence vnitřní oscilátor INTRC a primární oscilátor není INTRC, je tento oscilátor odpojen z důvodu snížení spotřeby. Časovač Watchdog a Obvod pro monitorování taktovací frekvence jsou vybaveny vlastním zdrojem hodinového signálu.
7. Pokud byl sekundárním zdrojem taktovacího kmitočtu oscilátor T1OSC, pokračuje tento oscilátor v činnosti pokud je dále povolen bitem T1OSCEN. V opačném případě je tento oscilátor odpojen.

Obrazek 4-9: Časové průběhy při návratu k primárnímu zdroji taktu



4.7.3.2 Návrat k primárnímu zdroji taktovací frekvence pomocí resetu

Jakýkoli signál reset způsobí nulování bitů $SCS<1:0>$ a tím spuštění programu s primárním zdrojem taktovací frekvence. Po aktivaci resetu dojde vždy ke spuštění primárního taktovacího oscilátoru podle nastavené konfigurace. Pokud je mikropočítač konfigurován pro vnější krystalový oscilátor (XT, HS nebo LP), zůstane po resetu podržen taktovací signál v taku Q1 ve stavu log.0 po dobu 1024 taktů. Tato doba je nezbytná pro dosažení stability krystalového oscilátoru.

Během rozběhu oscilátoru je výkon instrukcí stejně jako činnost periferních zařízení mikropočítače pozastavena.

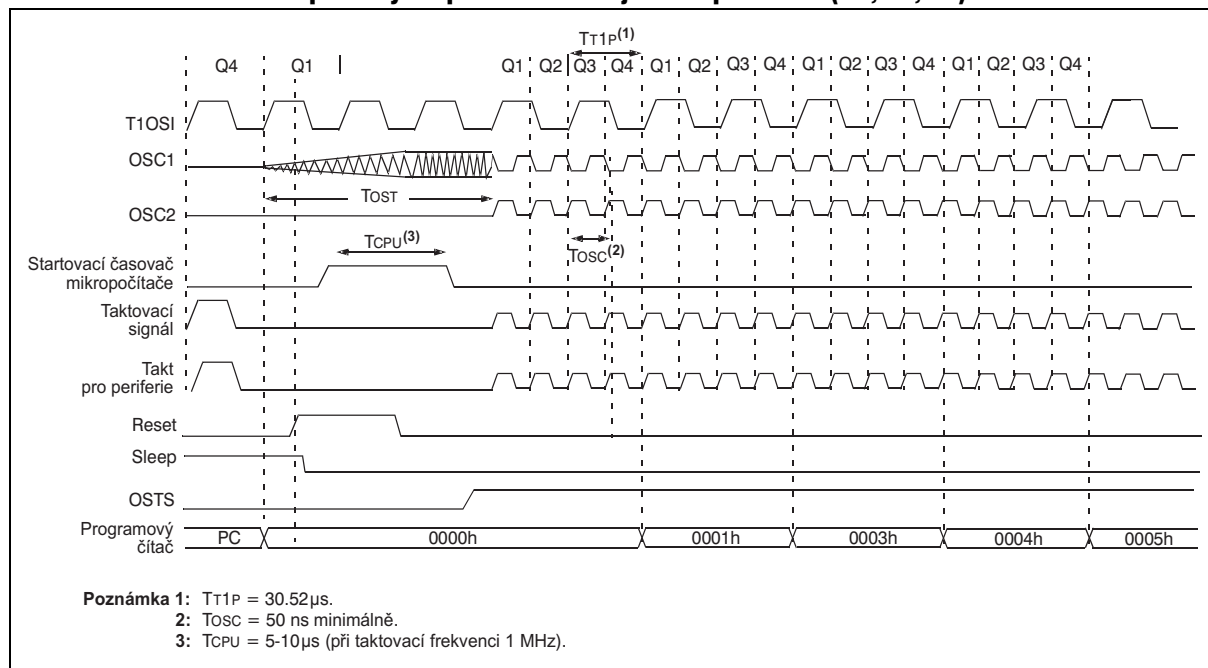
Poznámka: Při použití dvou rychlostního startu taktovacího oscilátoru je vnitřní oscilátor INTRC použit jako sekundární do přetečení časovače pro rozběh krystalového oscilátoru OST.

Pokud je primárním zdrojem taktovací frekvence oscilátor RC, EC nebo INTRC, je výkon programu zahájen v prvním taktu Q1 po události, která způsobila reset. V takovém případě není nutné čekat na ustálení zdroje taktovací frekvence, protože tento zdroj je stále v činnosti a je stabilní. Nicméně dojde ke zpoždění mezi taktem Q1 a taktem Q2 prvního instrukčního cyklu. Toto zpoždění je způsobené vnitřním časovačem a je rovno přibližně 5 – 10 μs .

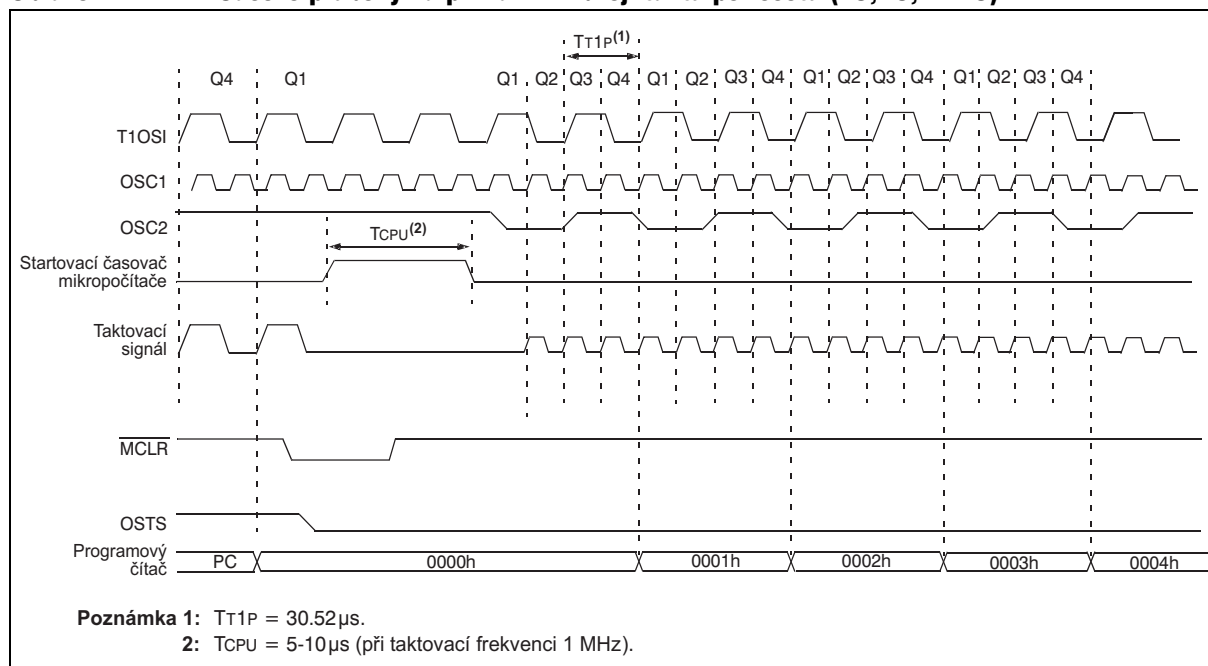
Při resetu mikropočítače dojde k následujícím událostem:

1. Reset mikropočítače může být způsoben jedním z mnoha zdrojů (WDT, BOR, $\overline{MCLR}$, atd)
2. Je generován signál reset a mikropočítač je po dobu určenou startovacím časovačem držen ve stavu resetu.
3. Pokud je primárním zdrojem taktovací frekvence konfigurován krystalový oscilátor XT, HS nebo LP, je následně generováno zpoždění 1024 taktů tohoto oscilátoru. Po tuto dobu je mikropočítač držen ve stavu reset. Časovač pro rozběh krystalového oscilátoru OST a startovací časovač mikropočítače jsou v činnosti současně.
4. Po přetečení obou uvedených časovačů je proveden jeden instrukční cyklus pro načtení první instrukce a následuje výkon první instrukce.

Obrázek 4-10: Časové průběhy na primárním zdroji taktu po resetu (HS, XT, LP)



Obrázek 4-11: Časové průběhy na primárním zdroji taktu po resetu (EC,RC,INTRC)



Tabulka 4-4: Režimy taktovacího oscilátoru

Stávající zdroj taktovací frekvence	Bity SCS <1:0> změněny na:	Zpoždění	OSTS Bit	IOFS Bit	T1RUN Bit	Nový zdroj taktovací frekvence	Poznámky
LP, XT, HS, T1OSC, EC, RC	10 (INTRC) FOSC <2:0> = LP, XT nebo HS	8 taktů oscilátoru INTRC	0	1 (1)	0	Oscilátor INTRC nebo INTOSC nebo INTOSC s předděličkou	Vnitřní RC oscilátor je ovládán bity IRCF
LP, XT, HS, INTRC, EC, RC	01 (T1OSC) FOSC <2:0> = LP, XT nebo HS	8 taktů oscilátoru T1OSC	0	N/A	1	T1OSC	Bit T1OSCEN musí být povolen
INTRC T1OSC	00 FOSC <2:0> = EC nebo FOSC <2:0> = RC	8 taktů oscilátoru EC nebo RC	1	N/A	0	EC nebo RC	
INTRC T1OSC	00 FOSC <2:0> = LP, XT, HS	1024 taktů časovače OST + 8 taktů oscilátoru LP, XT nebo HS	1	N/A	0	LP, XT, HS	Během 1024 taktů je program vykováván podle sekundárního taktovacího zdroje, primární oscilátor se zatím stabilizuje
LP, XT, HS	00 (po resetu) LP, XT, HS	1024 taktů časovače OST	1	N/A	0	LP, XT, HS	Po signálu reset nedojde k přepínání taktovacího zdroje. Výkon programu a činnost periférií je pozastavena do povolení dvourychlostního startu, kdy dojde k rozběhu programu z vnitřního taktovacího oscilátoru INTRC, který je po 1024 taktech přepnut na externí krystalový oscilátor.

Poznámka: Pokud je nový zdroj taktovací frekvence oscilátor INTOSC nebo INTOSC s předděličkou, je bit IOFS nastaven přibližně po 4 ms po vyvolání přepnutí taktu.

4.7.4 Probuzení ze stavu Sleep pomocí přerušení

Každé přerušení včetně přerušení od časovače Watchdog nebo na vývodu INTO může způsobit ukončení režimu Sleep.

Bity SCS nejsou instrukcí **SLEEP** nijak ovlivněny a jejich stav je stejný před zahájením i po ukončení režimu Sleep. Zdroj taktovací frekvence použitý po probuzení ze stavu Sleep je určen pomocí bitů SCS.

4.7.4.1 Popis událostí po probuzení ze stavu Sleep

při SCS<1:0> = "00":

1. Mikroprocesor je držen ve stavu Sleep dokud nedojde k dokončení startovací sekvence CPU.
2. Pokud je zdrojem taktovací frekvence konfigurován vnější oscilátor typu HS, XT nebo LP, je aktivován časovač OST a je generováno zpoždění 1024 taktů primárního taktovacího oscilátoru. Po tuto dobu je mikroprocesor držen ve stavu Sleep i když je povolen systém dvou rychlostního rozběhu taktovacího oscilátoru. Podrobnosti jsou uvedeny v **kapitole 15.12.3**. Časovač OST a časovač dvourychlostního rozběhu Start-up běží současně.
3. Po přetečení obou časovačů je zahájen výkon programu podle taktu, jehož zdroj je definován bity FOSC.

při SCS<1:0> = "01" nebo "10":

1. Mikroprocesor je držen ve stavu Sleep dokud nedojde k dokončení startovací sekvence CPU.
2. Po přetečení časovače Start-up dojde k ukončení režimu Sleep a je zahájen výkon programu podle zvoleného taktovacího oscilátoru.

Poznámka: Pokud uživatel změní bity SCS <1:0> těsně před uvedením do režimu Sleep, může být taktovací frekvence při zahájení režimu Sleep odlišná od taktovací frekvence použité při ukončení režimu Sleep. Například, pokud je SCS <1:0> = "01" a zdrojem taktovací frekvence je T1OSC a je vykonán následující sled instrukcí: je vyvoláno přepnutí taktovací frekvence. Pokud je primárním zdrojem taktu oscilátor XT, LP nebo HS, mikroprocesor pokračuje v činnosti s taktovacím oscilátorem T1OSC a vykoná instrukci **SLEEP**. Po ukončení režimu Sleep je program vykonáván podle nového krystalového taktovacího oscilátoru, po přetečení časovače OST.

PIC16F87/88

Ponámky:

5 VSTUPNÉ VÝSTUPNÍ BRÁNY

5.1 Registry PORTA a TRISA

Některé vývody mikropočítače jsou použity nejen jako univerzální digitální vstupně výstupní vývody, ale mohou sloužit pro činnost některé z periférií. Obecně platí, že pokud je vývod použit pro činnost periférie není možné ho využívat jako univerzální digitální vývod.

Podrobnější informace lze nalézt v dokumentaci DS 33023 s názvem *PICmicro® Mid-Range Reference Manual*.

Brána PortA je obousměrný, 8-bitový port. Registr PORTA je záchytný pracovní registr portu. Obsahuje 8 funkčních bitů. Registr TRISA je konfigurační registr brány PortA. Nastavením do stavu log.1 každého bitu konfiguruje odpovídající vývod brány PortA jako vstupní. Jedná se o stav vysoké impedance (třetí stav).

Nulováním bitu v registru TRISA konfiguruje příslušný vývod jako výstupní. V tom případě se objeví hodnota příslušného bitu z registru PORTA na vývodu mikropočítače.

Poznámka: Po resetu POR jsou vývody PORTA<4:0> konfigurovány jako analogové vstupy a při čtení dávají log. 0.

Čtením registru PORTA zjistíme stav na vývodech mikropočítače, který je určován vnějším prostředím. Zápisem do tohoto registru umístím data do výstupního záchytného registru a mikropočítač je bude udržovat na výstupních vývodech.

Všechny zápisové operace jsou typu čtení-modifikace-zápis. To se týká i zápisu jednotlivých bitů. Před každým zápisem je celý port načten, modifikován příslušný bit a teprve potom jsou změněná data umístěna na výstup. Pokud vnější prostředí způsobí na vývodech jinou úroveň než odpovídá hodnotě záchytného registru (např. zkrat proti zemi při log.1), je tato úroveň načtena a následně zapsána do výstupního registru.

Vývod RA4 je sdílen jako univerzální digitální vývod a jako vstup čítače/časovače Timer0. U mikropočítače PIC16F88 je rovněž sdílen s analogovým vstupem AN4. Vývod je vybaven Schmittovým klopným obvodem na vstupu a v režimu výstupu je zapojen jako dvojčinný výstup CMOS.

Vývod RA5 je sdílen se vstupem pro vnější reset. Vývod je ošetřen schmittovým klopným obvodem na vstupu.

Vývod RA6 je sdílen se vstupem modulu oscilátoru nebo výstupem externího taktovacího oscilátoru. Vývod RA7 je sdílen se vstupem modulu oscilátoru nebo vstupem externího taktovacího oscilátoru. Oba vývody jsou ošetřeny ve vstupním režimu Schmittovým klopným obvodem a v režimu výstupu se jedná o dvojčinné výstupy CMOS.

Vývody PORTA<1:0> jsou sdíleny jako analogové vstupy, vývody PORTA<3:2> jsou sdíleny jako analogové vstupy nebo výstupy pro komparátor. U mikropočítače PIC16F88 jsou vývody PORTA<3:2> sdíleny rovněž jako vstupy pro napěťovou referenci VREF. Vývody PORTA<3:0> mají ve vstupním režimu úroveň TTL a ve výstupním režimu se jedná o dvojčinné výstupy s úrovněmi CMOS.

Příklad 5-1: Inicializace brány PortA

BANKSEL	PORTA	;přepnutí do banky 0
CLRF	PORTA	;vynulování výstupního ;záchytného registru
BANKSEL	ANSEL	;přepnutí do banky 1
MOVLW	0x00	;konfigurování všech vývodů
MOVWF	ANSEL	;jako digitální vývod
MOVLW	0xFF	;hodnota pro konfiguraci
MOVWF	TRISA	;všechny vývodu jsou vstupní

Tabulka 5-1: Funkce brány PORTA

Název	Bit	Typ vstupu	Funkce vývodu
RA0/AN0	bit 0	TTL	Digitální vstup/výstup nebo analogový vstup
RA1/AN1	bit 1	TTL	Digitální vstup/výstup nebo analogový vstup
RA2/AN2/CVREF/VREF- (2)	bit 2	TTL	Digitální vstup/výstup nebo analogový vstup nebo vstup VREF- nebo výstup komparátoru VREF
RA3/AN3/VREF+ (2)/C1OUT	bit 3	TTL	Digitální vstup/výstup nebo analogový vstup nebo VREF+ nebo výstup komparátoru
RA4/AN4 (2)/TOCKI/C2OUT	bit 4	ST	Digitální vstup/výstup nebo vstup hodinového signálu pro Timer0 nebo výstup komparátoru nebo analogový vstup
RA5/MCLR/VPP	bit 5	ST	Digitální vstup nebo vstup pro reset nebo vstup programovacího napětí
RA6/OSC2/CLKO	bit 6	ST	Digitální vstup/výstup, připojení krystalu nebo rezonátoru, výstup oscilátoru nebo výstup taktovací frekvence dělené čtyřmi v konfiguraci vnější RC taktovací oscilátor.
RA7/OSC1/CLKI	bit 7	ST/CMOS (1)	Digitální vstup/výstup, připojení krystalu nebo rezonátoru nebo vstup oscilátoru.

Význam: TTL = vstup úroveň TTL, ST = vstup opatřen Schmittovým klopným obvodem

Poznámka 1: Tento vstup je ošetřen Schmittovým klopným obvodem jen při konfiguraci jako vstup RC oscilátoru

2: Pouze u mikropočítače PIC16F88

PIC16F87/88

Tabulka 5-2: Registry, které mají vztah k bráně PortA

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druzích resetu
05h	PORTA	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx 0000 ⁽¹⁾ xxx0 0000 ⁽²⁾	uuuu 0000 ⁽¹⁾ uuu0 0000 ⁽²⁾
85h	TRISA	TRISA7	TRISA6	TRISA5 ⁽³⁾	Konfigurační registr pro určení vstupu/výstupu						1111 1111
9Fh	ADCON1	ADFM	ADCS2	VCFG1	VCFG0	—	—	—	—	0000 ----	0000 ----
9Bh	ANSEL ⁽⁴⁾	—	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	-111 1111	-111 1111

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s PortA.

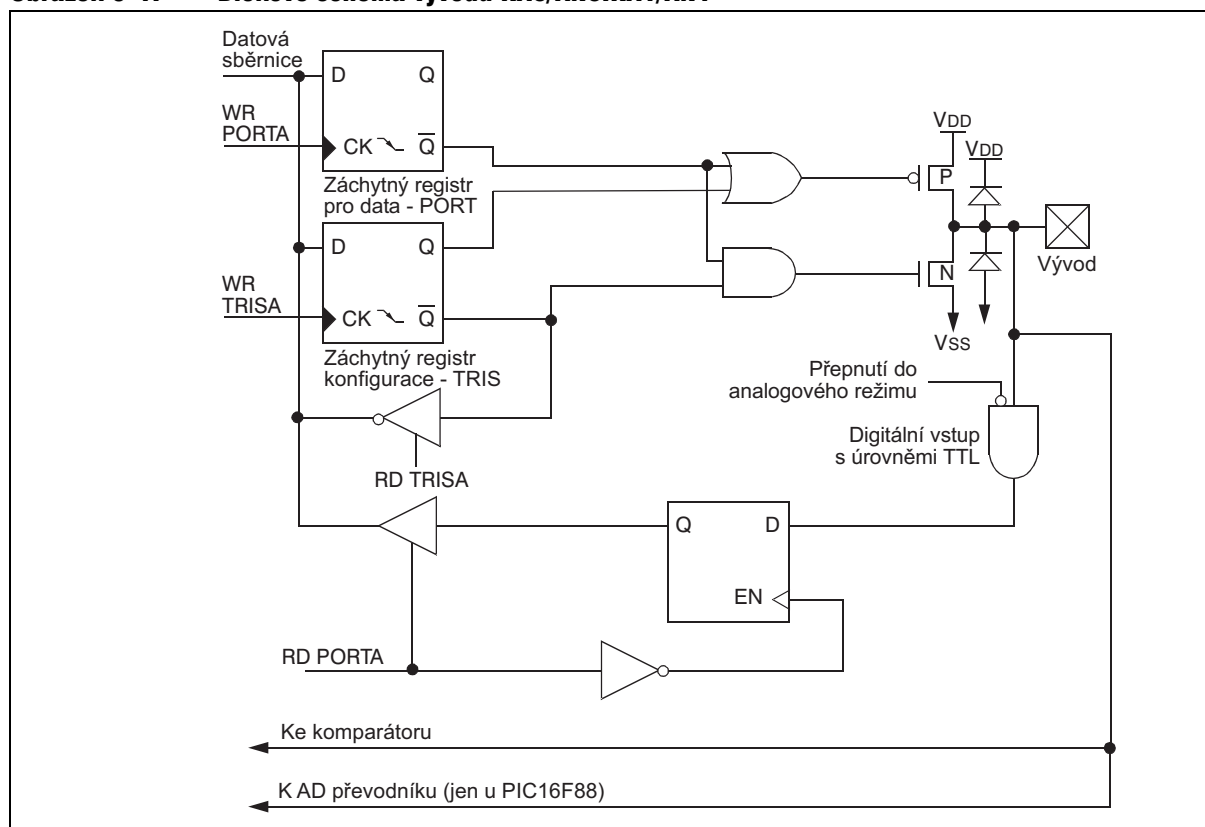
Poznámka 1: Tyto hodnoty platí jen pro mikročítač PIC16F87

2: Tyto hodnoty platí jen pro mikročítač PIC16F88

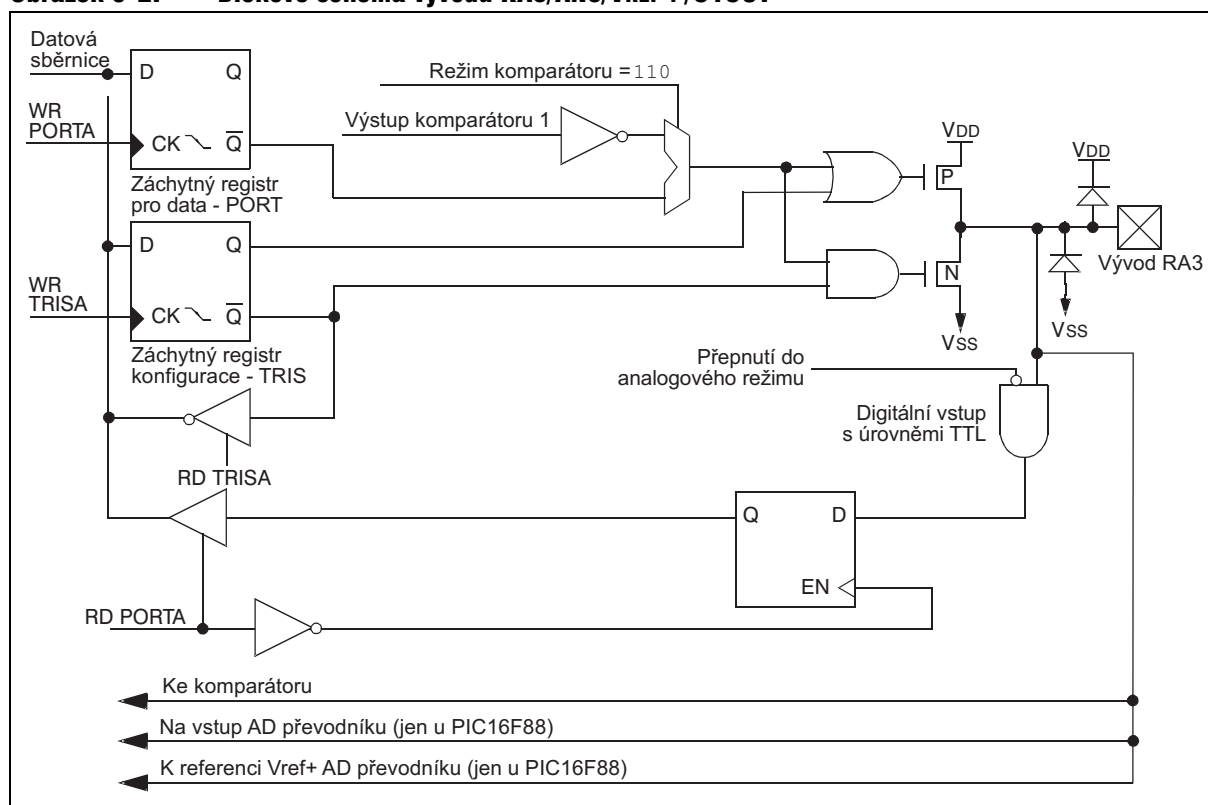
3: Vývod 5 je pouze vstupní; nulování bitu TRIS5 nemá žádný vliv a tento bit je vždy čten jako log.1

4: Pouze pro mikročítač PIC16F88

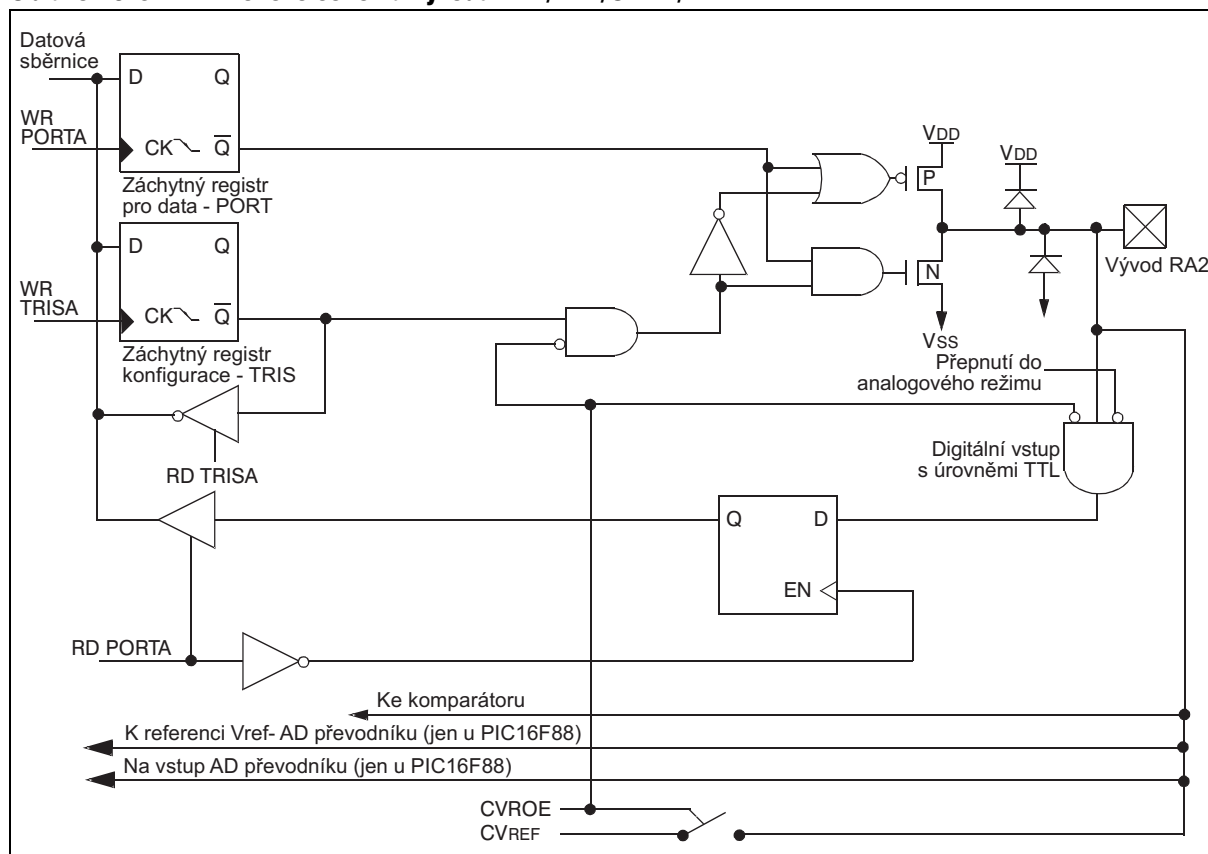
Obrázek 5-1: Blokové schéma vývodů RA0/AN0:RA1/AN1



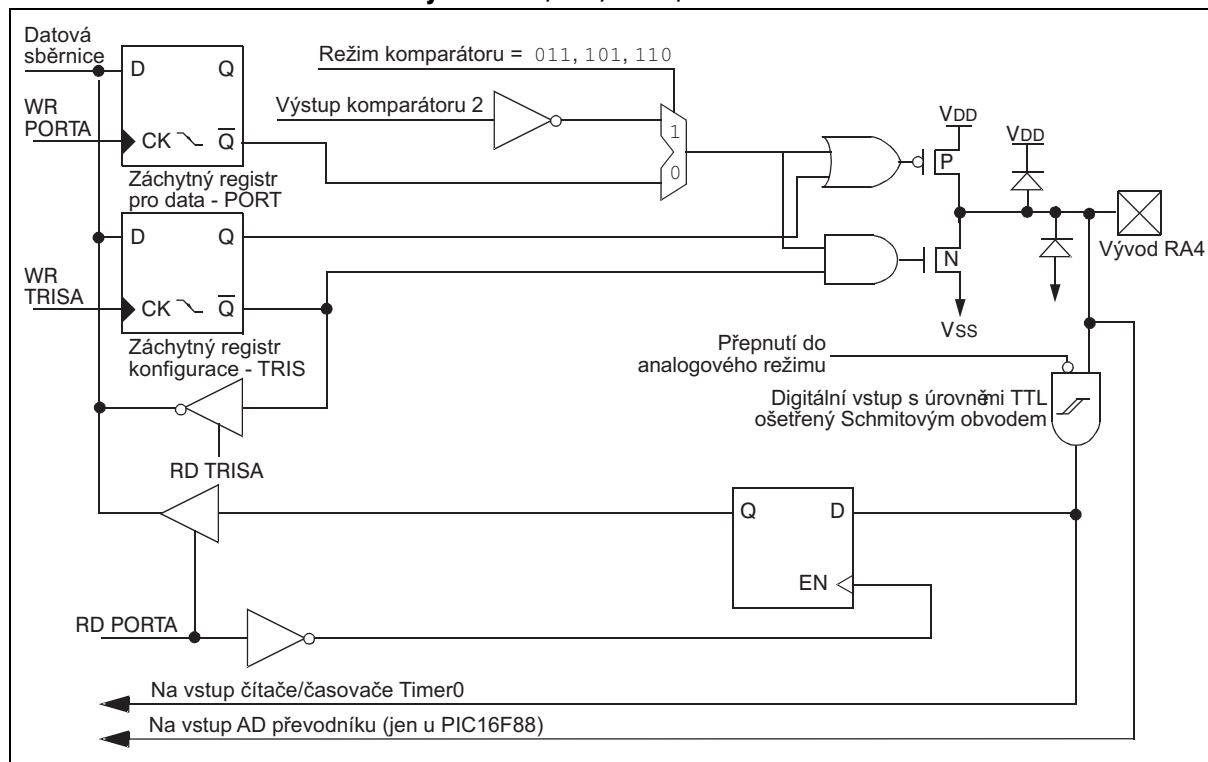
Obrázek 5-2: Blokové schéma vývodů RA3/AN3/VREF+/C10UT



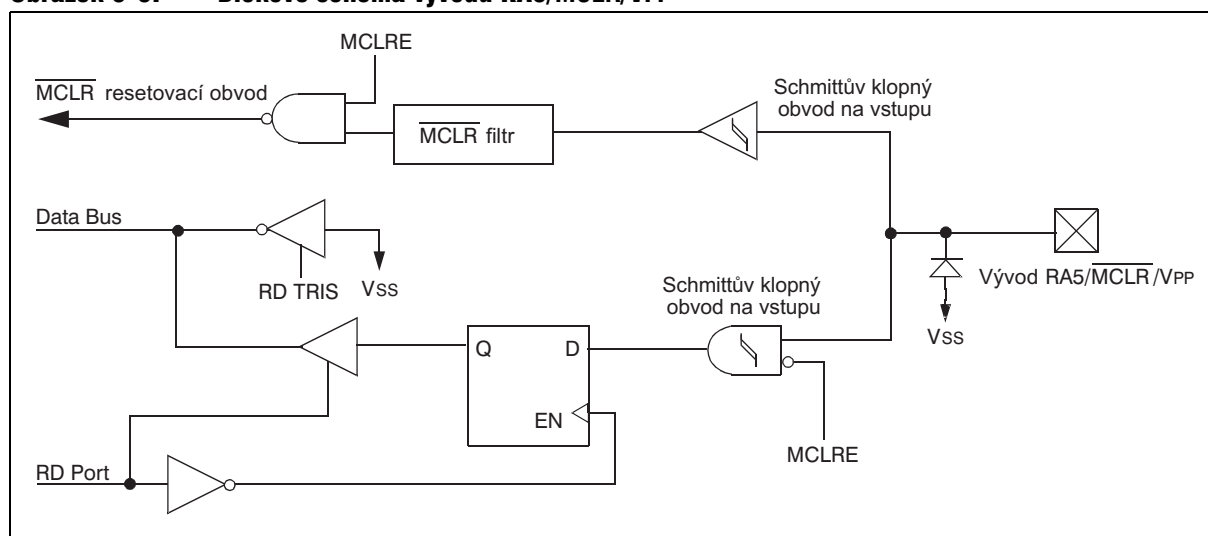
Obrázek 5-3: Blokové schéma vývodů RA2/AN2/CVREF/VREF



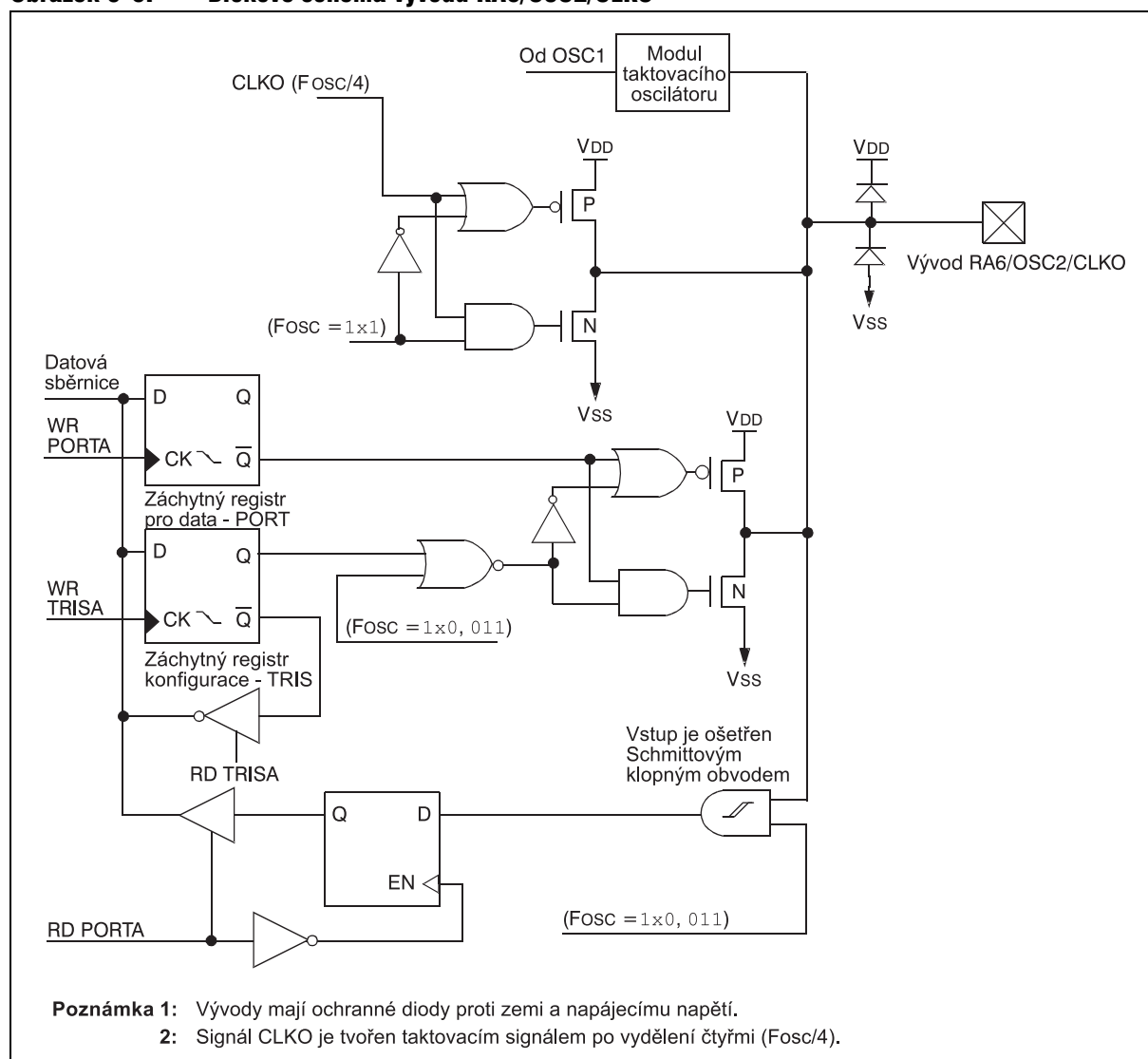
Obrázek 5-4: Blokové schéma vývodů RA4/AN4/TOCKI/C2OUT



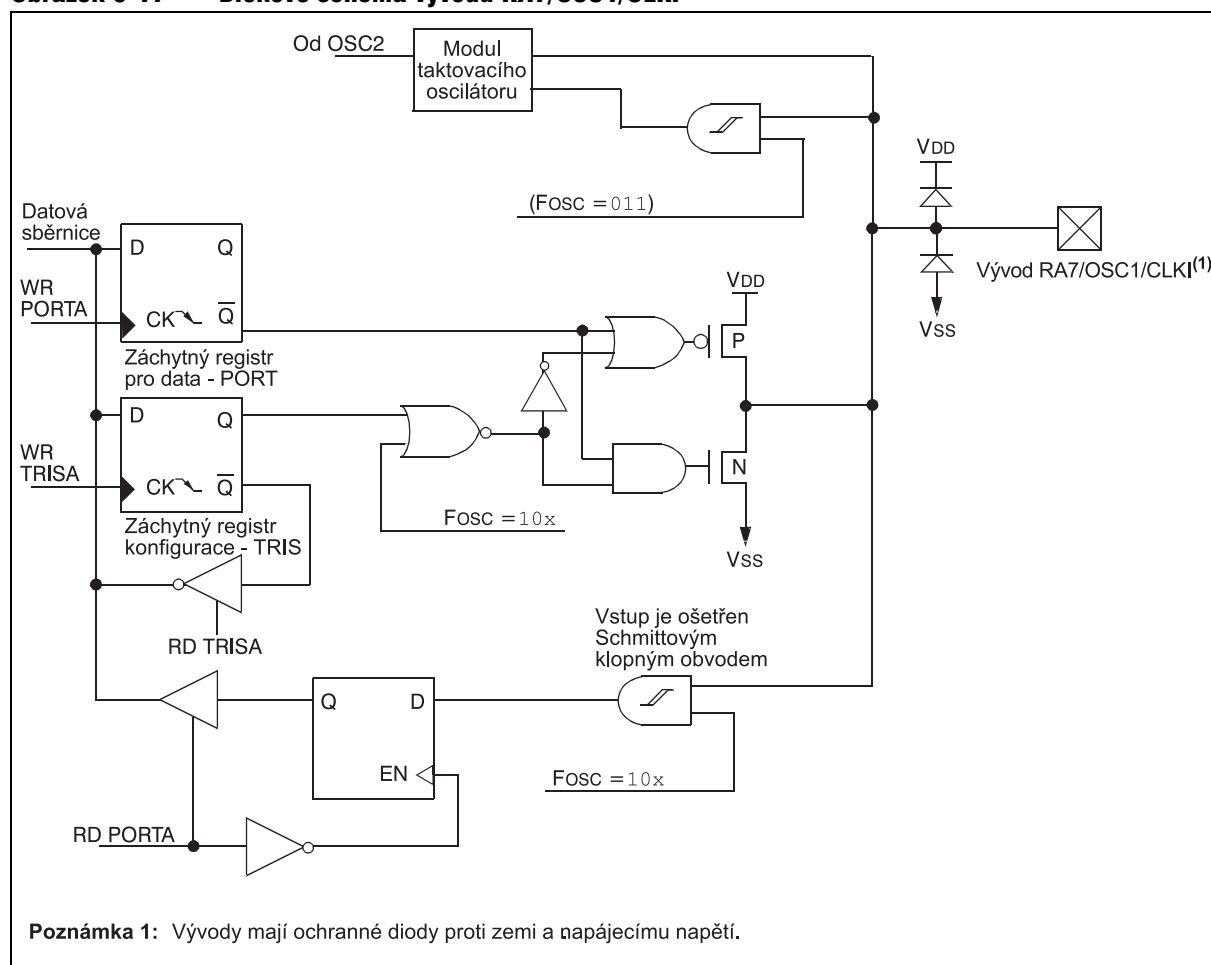
Obrázek 5-5: Blokové schéma vývodů RA5/MCLR/Vpp



Obrázek 5-6: Blokové schéma vývodů RA6/OSC2/CLKO



Obrázek 5-7: Blokové schéma vývodů RA7/OSC1/CLKI



5.2 Registry PORTB a TRISB.

Brána PortB je obousměrný, 8-bitový port. Registr PORTB je záchytný pracovní registr portu. Obsahuje 8 funkčních bitů. Registr TRISB je konfigurační registr brány PortB. Nastavením do stavu log.1 každého bitu konfiguruje odpovídající vývod brány PortB jako vstupní. Jedná se o stav vysoké impedance (třetí stav).

Nulováním bitu v registru TRISB konfiguruje příslušný vývod jako výstupní. V tom případě se objeví hodnota příslušného bitu z registru PORTB na vývodu mikropočítače.

Každý vývod brány PortB je vybaven možností připojit ve vstupním režimu vnitřní "pull-up" rezistor na vstup. Tím dojde k přidržení vstupního nezatíženého vývodu ve stavu log.1. Nulováním řídicího bitu $\overline{\text{RPU}}$ v registru $\text{OPTION_REG} \langle 7 \rangle$ lze připojit na všechny vstupní vývody brány PortB rezistory pull-up. Přepnutím do výstupního režimu dojde automaticky k odpojení vnitřního pull-up rezistoru. Po signálu reset jsou všechny pull-up rezistory zakázány pomocí čtyř vývodů brány PortB, RB7:RB4 lze vyvolat žádost o přerušení změnou na bráně PortB, konfigurovaných do vstupního stavu. Vývody konfigurované jako výstup nemohou vyvolat tento druh přerušení. Vstupní hodnota na uvedených bitech brány PortB je v každém instrukčním cyklu porovnávána s hodnotou posledního čtení brány PortB. Pokud dojde k jakékoli změně, generuje se žádost o přerušení a dojde k nastavení příznaku přerušení změnou na bráně PortB RBIF v registru $\text{INTCON} \langle 0 \rangle$.

Tento druh přerušení umožňuje probuzení mikropočítače z režimu Sleep. V obslužné rutině přerušení musí uživatel ukončit žádost o přerušení následujícím způsobem:

- a) Provést jakékoli čtení nebo zápis do registru PORTB. Tím dojde k ukončení změnového stavu, který generuje žádost o přerušení.
- b) Nulovat příznak přerušení RBIF.

Pokud nedojde po detekované změně na RB7:RB4 k uvedené proceduře, změnový stav neustále nastavuje příznak přerušení RBIF a jeho nulování nezpůsobí jeho trvalé uvedení do stavu log.0.

Použití uvedeného způsobu přerušení lze doporučit pro probuzení ze stavu Sleep např. po stisku tlačítka klávesnice nebo v případech, kdy je pro uvedený druh přerušení použita celá brána PortB. Používání brány pro další účely současně s využitím tohoto druhu přerušení se nedoporučuje.

Vývod RB0/INT je vstup pro vnější přerušení. Konfiguruje se pomocí bitu INTEDG v registru $\text{OPTION_REG} \langle 6 \rangle$.

Vývody brány PortB jsou sdíleny s řadou periferních zařízení mikropočítače, viz tabulku 5-3. Všechny vývody brány PortB jsou v režimu vstupu ošetřeny Schmittovým klopným obvodem.

Při použití periférií užívajících PortB buďte pozorní při práci s konfiguračními bity v registru TRIS. Některá periferní zařízení nulují či naopak nastavují příslušné bity a konfigurují si tak potřebné vývody automaticky. Pokud dojde k takovému automatickému nastavení registru TRIS a program následně provede operaci typu Čtení-modifikace-zápis (**BCF**, **BSF**, **XORWF**) jejímž cílovým registrem je TRIS, může dojít k nežádoucímu přepsání správné hodnoty nastavení TRIS. Uživatel musí vždy před použitím konkrétní periferie detailně pochopit způsob ovládání registru TRIS.

PIC16F87/88

Tabulka 5-3: Funkce brány PORTB

Název	Bit	Typ vstupu	Funkce vývodu
RB0/INT/CCP1 ⁽⁷⁾	bit 0	TTL/ST ⁽¹⁾	Digitální vstup/výstup nebo vstup pro vnější přerušení. Vstup pro Capture nebo výstup pro Compare nebo výstup pro PWM. Lze softwarově připojit vnitřní pull-up rezistor.
RB1/SDI/SDA	bit 1	TTL/ST ⁽⁵⁾	Digitální vstup/výstup. Vstup dat pro SPI™ nebo vstup/výstup pro I ² C™ Lze softwarově připojit vnitřní pull-up rezistor.
RB2/SDO/RX/DT	bit 2	TTL/ST ⁽⁴⁾	Digitální vstup/výstup, výstup dat SPI™ Asynchronní příjem dat nebo data pro synchronní režim modulu AUSART Lze softwarově připojit vnitřní pull-up rezistor.
RB3/PGM/CCP1 ^(3,7)	bit 3	TTL/ST ⁽²⁾	Digitální vstup/výstup, vstup pro programovací napětí v režimu LVP nebo vstup pro Capture nebo výstup pro Compare nebo výstup pro PWM. Lze softwarově připojit vnitřní pull-up rezistor.
RB4/SCK/SCL	bit 4	TTL/ST ⁽⁵⁾	Digitální vstup/výstup, hodiny pro SPI™ nebo I ² C™ (vstup pro přerušení změnou na bráně). Lze softwarově připojit vnitřní pull-up rezistor.
RB5/SS/TX/CK	bit 5	TTL	Digitální vstup/výstup nebo vstup pro výběr režimu Slave pro SPI™. Asynchronní vysílání dat nebo hodiny pro synchronní režim modulu AUSART Lze softwarově připojit vnitřní pull-up rezistor.
RB6/AN5 ⁽⁶⁾ /PGC/T1OSO/T1CKI	bit 6	TTL/ST ⁽²⁾	Digitální vstup/výstup, analogový vstup, hodiny pro sériové programování (vstup pro přerušení změnou na bráně). Výstup pro oscilátor Timer1 nebo vstup pro hodinový signál čítače/časovače Timer1. Lze softwarově připojit vnitřní pull-up rezistor.
RB7/AN6 ⁽⁶⁾ /PGD/T1OSI	bit 7	TTL/ST ⁽²⁾	Digitální vstup/výstup, analogový vstup, data pro sériové programování (vstup pro přerušení změnou na bráně). Vstup pro oscilátor Timer1. Lze softwarově připojit vnitřní pull-up rezistor.

Význam: TTL = vstup úrovně TTL, ST = vstup opatřen Schmittovým klopným obvodem

- Poznámka 1:** Vstup je ošetřen Schnittovým klopným obvodem jen v případě použití jako vstupu pro vnější přerušení.
- 2:** Vstup je ošetřen Schnittovým klopným obvodem jen v případě použití při programování ICSP.
- 3:** Programování nízkým napětím (LVP) je u mikropočítače implicitně povoleno a tím zakázáno použití vývodu RB3 jako univerzálního digitálního vývodu. Pro použití vývodu RB3 jako univerzálního vývodu je nutné zakázat LVP v konfiguračním slově. Tím je dosaženo maximální kompatibility s ostatními 18 vývodovými mikropočítači řady Mid-range.
- 4:** Vstup je ošetřen Schnittovým klopným obvodem jen v případě použití jako vstupu pro CCP nebo SSP.
- 5:** Vstup je ošetřen Schnittovým klopným obvodem jen v případě použití jako vstupu pro SPI nebo I²C.
- 6:** Jen pro mikropočítač PIC16F88
- 7:** Vývod CCP1 je konfigurován pomocí bitu CCPMX v konfiguračním slově mikropočítače č.1.

Tabulka 5-4: Registry, které mají vztah k bráně PortB

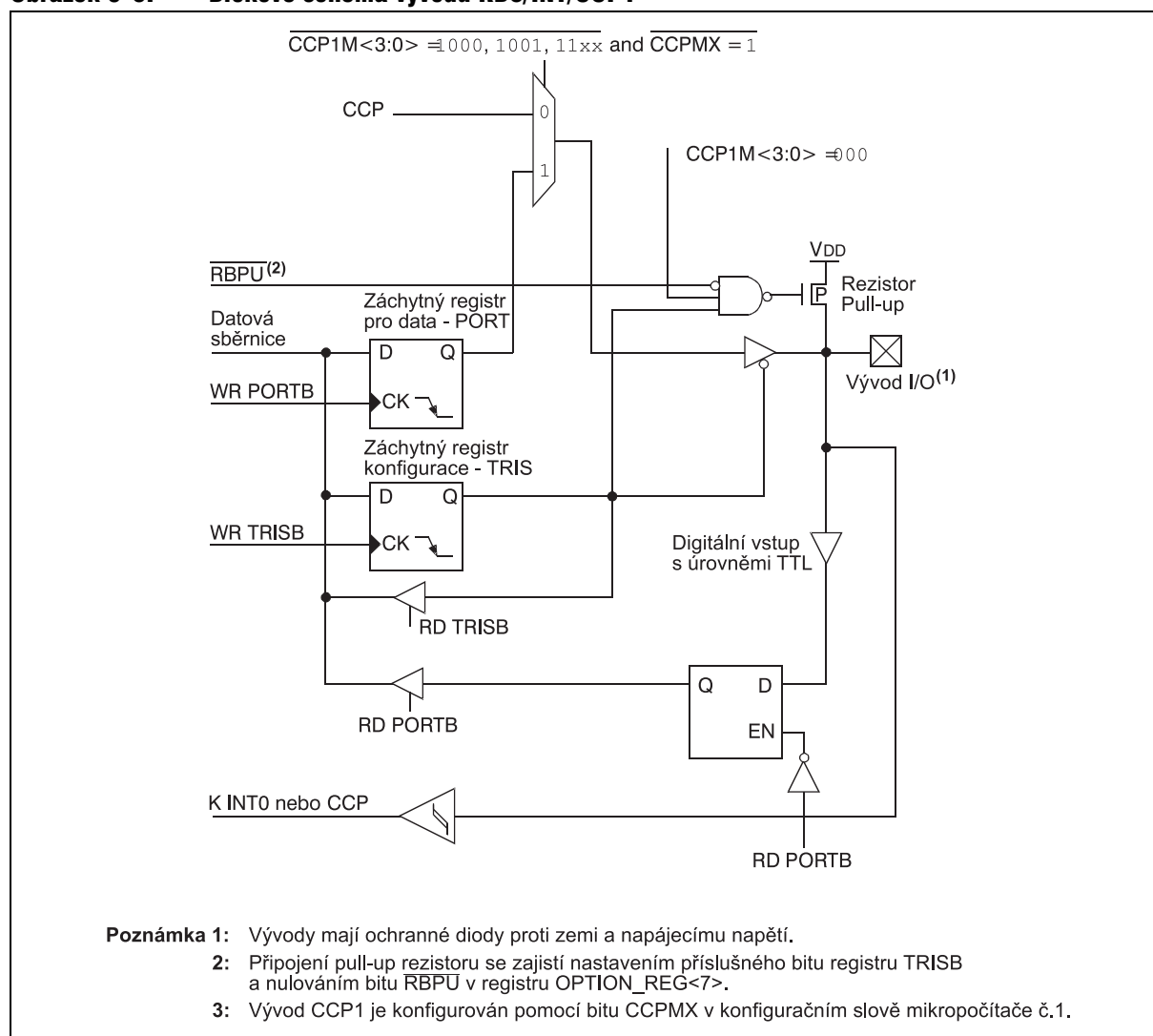
Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druhích resetu
06h, 106h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx ⁽¹⁾ 00xx xxxx ⁽²⁾	uuuu uuuu ⁽¹⁾ 00uu uuuu ⁽²⁾
86h, 186h	TRISB	Konfigurační registr pro určení vstupu/výstupu								1111 1111	1111 1111
81h, 181h	OPTION_REG	RBPU	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
9Bh	ANSEL ⁽²⁾	—	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	-111 1111	-111 1111

Vysvětlivky: x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s PortB.

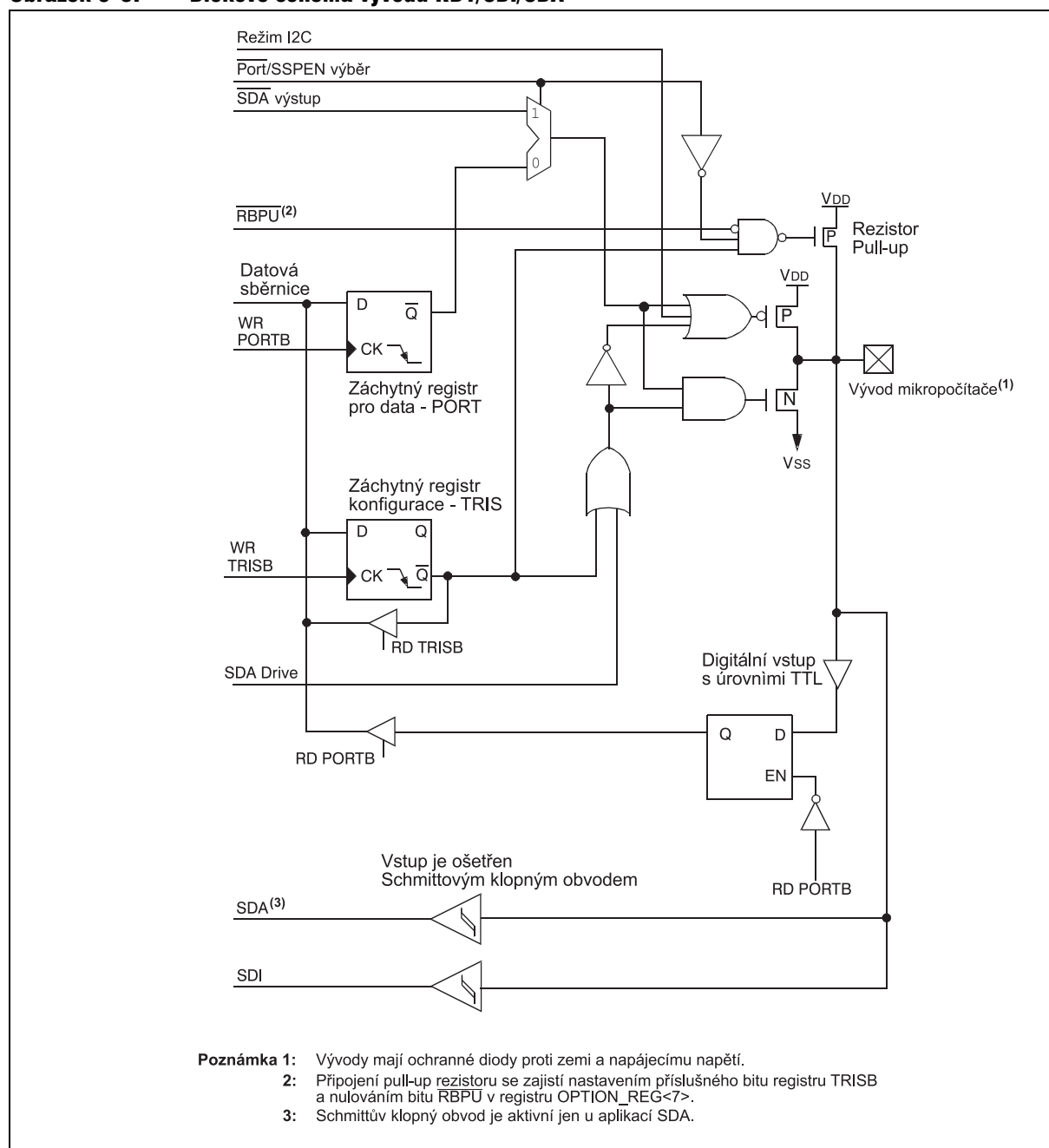
Poznámka 1: Hodnoty platí jen pro mikropočítač PIC16F87

2: Hodnoty platí jen pro mikropočítač PIC16F88

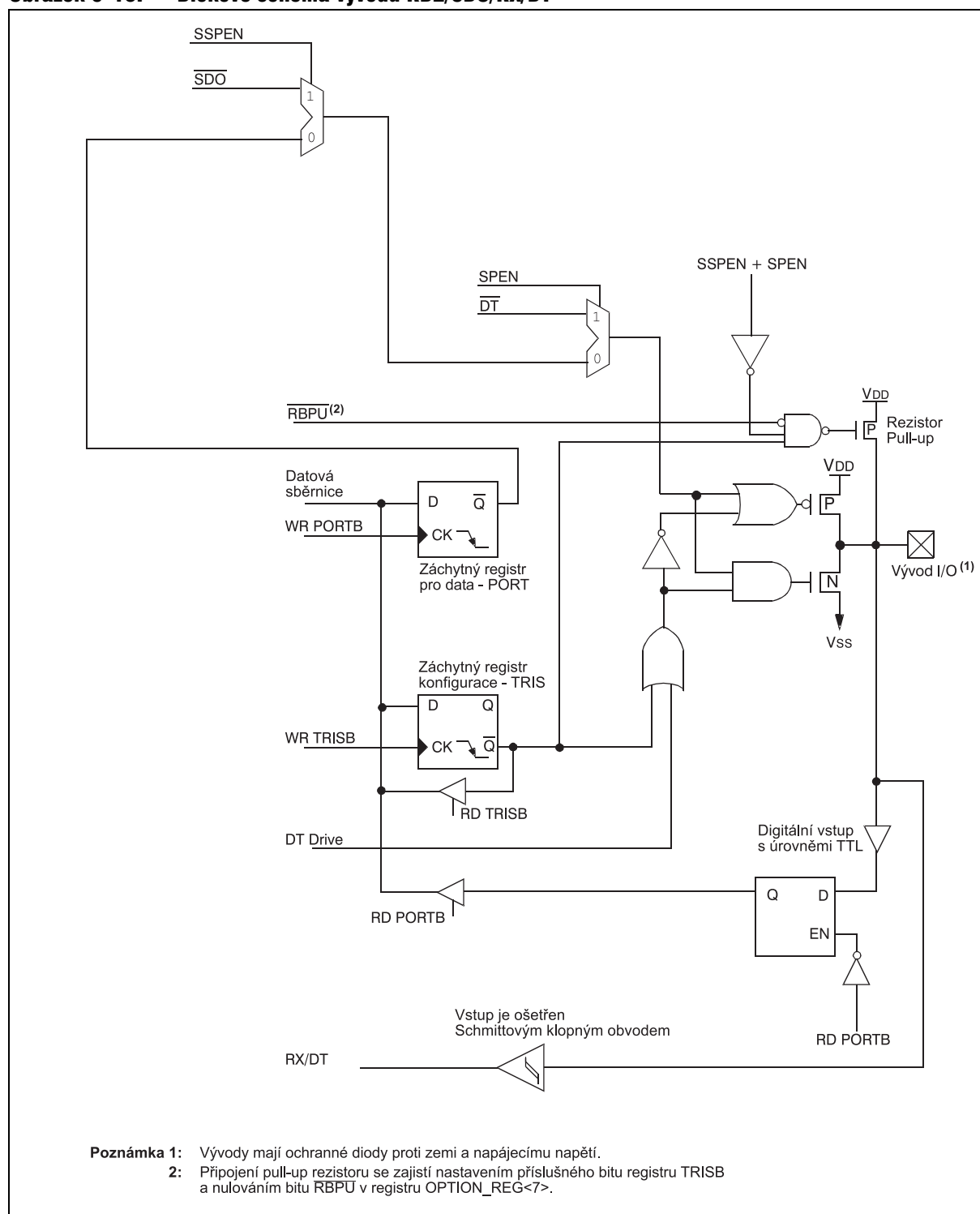
Obrázek 5-8: Blokové schéma vývodu RB0/INT/CCP1



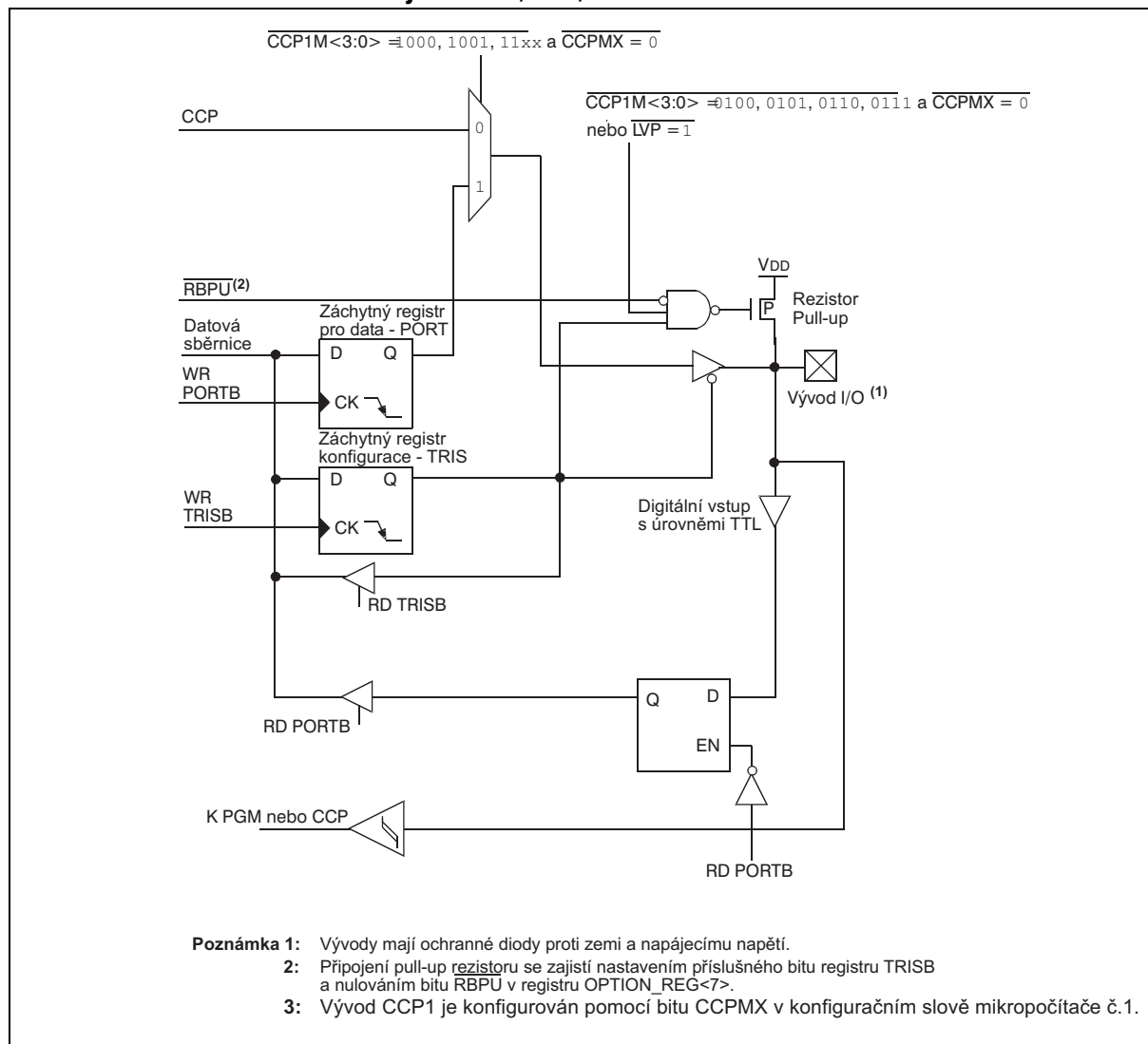
Obrázek 5-9: Blokové schéma vývodu RB1/SDI/SDA



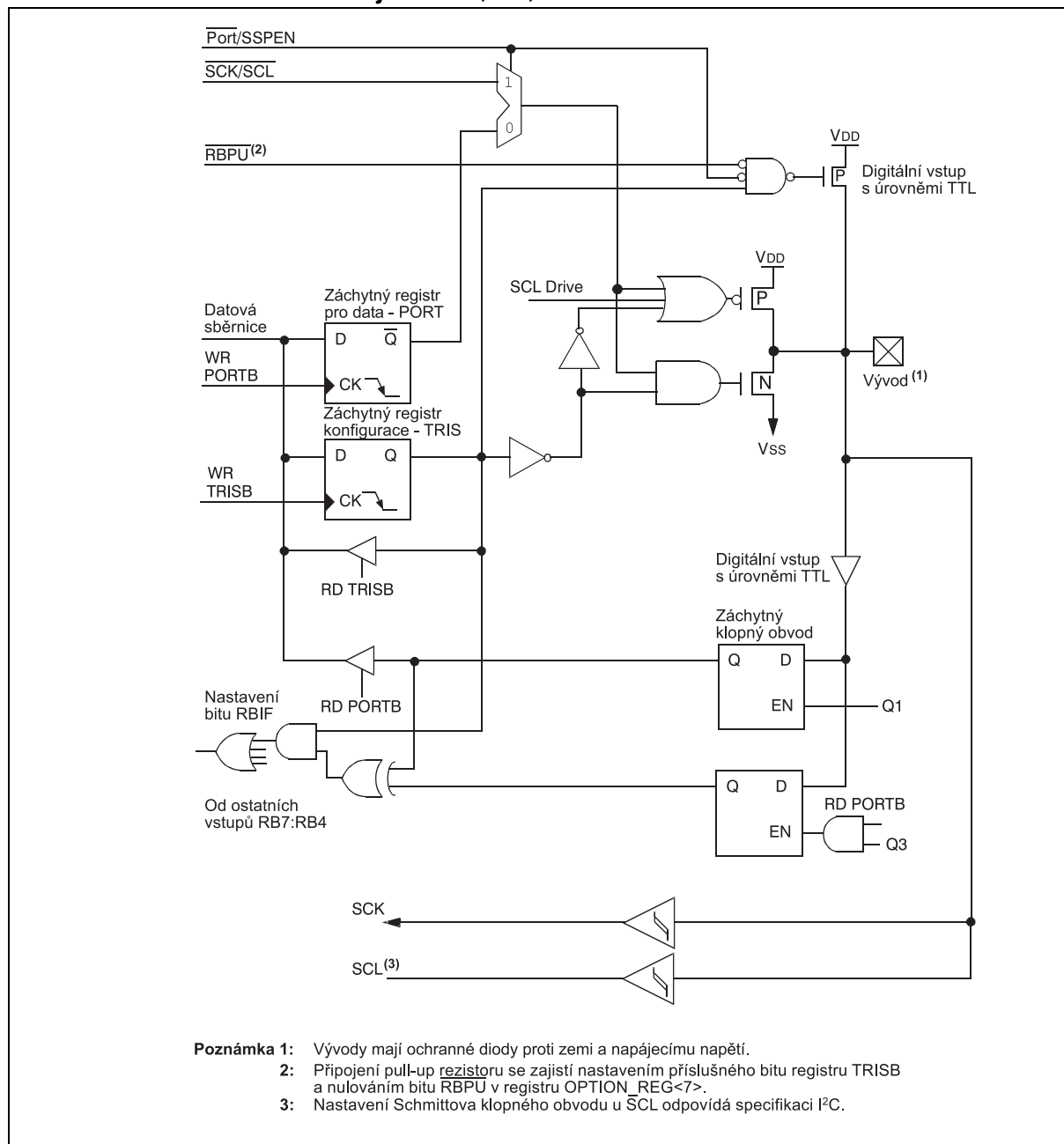
Obrázek 5-10: Blokové schéma vývodu RB2/SDO/RX/DT

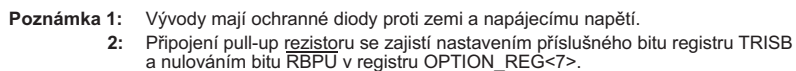


Obrázek 5-11: Blokové schéma vývodu RB3/PGM/CCP1

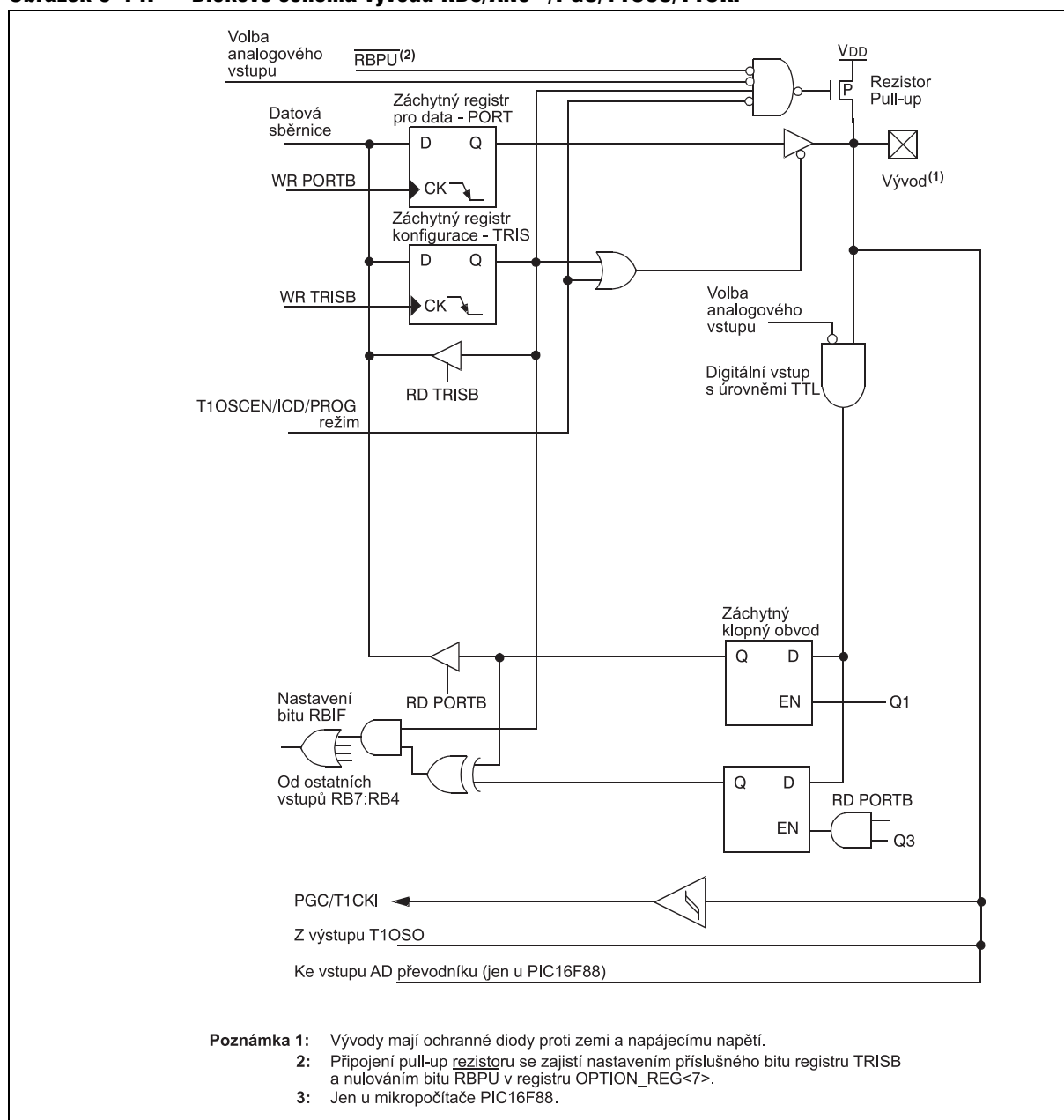


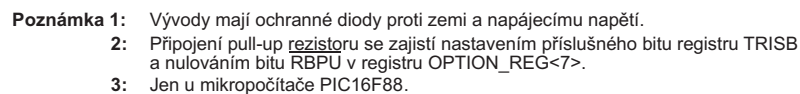
Obrázek 5-12: Blokové schéma vývodu RB4/SCK/SCL





Obrázek 5-14: Blokové schéma vývodu RB6/AN5⁽³⁾/PGC/T1OSO/T1CKI





- Modul čítače/časovače má následující základní vlastnosti:
- Šířka pracovního registru TMRO je 8 bitů
- Registr čítače/časovače TMRO lze číst i do něj zapisovat
- Volitelný zdroj hodinového signálu vnitřní nebo vnější
- Volitelná aktivní hrana vnějšího hodinového signálu
- 8-bitová programovatelná předdělička
- Prerušeni přetečením registru TMRO ze stavu FFh na 00h

6.1 Činnost čítače/časovače Timer0

Režim čítače vnějších událostí je zvolen nastavením řídicího bitu T0CS registru OPTION_REG<5>. V tomto režimu je registr TMR0 inkrementován s každou aktivní hranou vnějšího hodinového signálu přivedeného na vývod RA4/T0CKI. Sestupná hrana je aktivní nastavením bitu TOSE v registru OPTION_REG<4>. Nulováním tohoto bitu je zvolena aktivní náběžná hrana. Omezení pro použití čítače/časovače v režimu buzení vnějším signálem jsou podrobněji probírána v **kapitole 6.3**.

6.2 Přerušení od Timer0

Prerušení od čítače/časovače Timer0 nemůže probudit mikropočítač ze stavu Sleep, protože v tomto režimu je Timer0 odpojen.

The diagram illustrates the internal structure of the TMR0 module, divided into two main functional blocks: the Watchdog timer and the 8-bit prescaler.

Časovač Watchdog (Watchdog Timer):

- It receives a **Povolení časovače Watchdog** (Watchdog timer enable) signal.
- The timer is configured with a **31.25 kHz** frequency and a **16. bitová předdělička** (16-bit prescaler).
- The output of the timer is connected to a **MUX** (Multiplexer) controlled by the **PSA** bit.
- The output of this MUX is connected to the **Přetčení časovače Watchdog** (Watchdog timer overflow) signal.

Předdělička (8-bit prescaler):

- This block contains an **8. bitová předdělička** (8-bit prescaler) and a **Multiplexer 1 z 8** (1 of 8 multiplexer).
- The prescaler is controlled by the **PS2:PS0** bits.
- The output of the prescaler is connected to another **MUX** controlled by the **PSA** bit.
- The output of this MUX is connected to the **Přetčení časovače Watchdog** (Watchdog timer overflow) signal.

Other components and signals:

- The **CLKO (= Fosc/4)** signal is connected to the **MUX** and the **Synchronizace s vnitřním taktm** (Synchronization with internal clock) block.
- The **Vývod RA4/T0CKI/C2OUT** signal is connected to the **MUX** and the **Synchronizace s vnitřním taktm** block.
- The **T0SE** signal is connected to the **MUX** and the **Synchronizace s vnitřním taktm** block.
- The **T0CS** signal is connected to the **MUX** and the **Synchronizace s vnitřním taktm** block.
- The **PSA** signal is connected to the **MUX** and the **Synchronizace s vnitřním taktm** block.
- The **Synchronizace s vnitřním taktm** block is connected to the **Registr TMR0** (TMR0 register).
- The **Registr TMR0** is connected to the **Datová sběrnice** (Data bus) via an **8**-bit bus.
- The **Registr TMR0** is also connected to the **Nastavení příznaku TMR0IF při přetečení** (Setting the TMR0IF flag on overflow) signal.

6.3 Použití čítače/časovače Timer0 s vnějším hodinovým signálem

V případě, kdy není použita předdělička je vnější hodinový signál shodný s výstupním signálem s předděličky. Synchronizace taktovacího kmitočtu a kmitočtu na vstupu TOCKI je zajištěna tím, že ke vzorkování dochází ve druhém Q2 a čtvrtém Q4 strojovém taktu každého instrukčního cyklu. Z tohoto důvodu je důležité, aby stav log.1 tohoto signálu trval alespoň 2Tosc (strojové takt) a stav log.0 tohoto signálu rovněž 2Tosc. Náběžná hrana tohoto signálu nesmí být větší než 2Tt0H. Podrobnosti jsou uvedeny v elektrických charakteristikách.

6.4 Předdělička

Předdělička je tvořena 8-bitovým čítačem který může být zařazen před čítač/časovač Timer0 nebo za časovač Watchdog. Pro jednoduchost bude tento čítač nazýván vždy jako předdělička. Tato předdělička může být předřazena jen jednomu z uvedených zařízení, v žádném případě oběma současně. Obsah předděličky nelze zjišťovat čtením ani měnit zápisem, viz obrázek 6-1.

Poznámka: Předděličku lze přiřadit jen jednomu z uvedených zařízení. V tomto mikropočítači má Watchdog zařazenu další vlastní předděličku, což umožňuje dosáhnout různých dělicích poměrů pro obě zařízení. Podrobněji je pojednáno v kapitole 15.12.

Předděličku lze ovládat pomocí bitů PSA, PS2, PS1, PS0 v registru OPTION_REG <3:0>. Nastavením bitu PSA je předdělička přiřazena obvodu Watchdog, nulováním tohoto bitu pak čítači/časovači Timer0. Pomocí zbylých bitů se nastavuje dělicí poměr předděličky. Pokud je předdělička přiřazena čítači/časovači Timer0, veškeré instrukce zapisující do registru TMRO způsobí současně vynulování předděličky. Pokud je předdělička přiřazena časovači Watchdog, způsobí instrukce CLRWDT současně s nulováním obsahu Watchdog i nulování předděličky. Obsah předděličky nelze žádným způsobem číst. Po signálu reset je předdělička vždy vynulována.

Poznámka: Při zápisu do registru TMRO dojde i k vynulování předděličky za předpokladu, že je tato přiřazena čítači/časovači Timer0. Na obsah časovače Watchdog to nemá žádný vliv.

Registr 6-1: Registr OPTION_REG

	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	RBPƯ	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0
bit 7								bit 0
bit7	RBPƯ: Povolení připojení pull-up rezistorů na všechny vývody brány PortB konfigurované jako vstup 1 = zakázáno 0 = povoleno							
bit6	INTEDG: Volba aktivní hrany signálu pro vnější přerušení 1 = přerušení nastane s náběžnou hranou na vývodu RBO/INT 0 = přerušení nastane se sestupnou hranou na vývodu RBO/INT							
bit5	TOCS: Volba zdroje hodinového signálu pro čítač/časovač Timer0 1 = vnější zdroj na vývodu TOCKI 0 = vnitřní zdroj Fosc/4 (inkrementace jednou za instrukční cyklus, CLK0)							
bit4	TOSE: Volba aktivní hrany při vnějším zdroji hodinového signálu pro TMRO 1 = inkrementace se sestupnou hranou na vývodu TOCKI 0 = inkrementace s náběžnou hranou na vývodu TOCKI							
bit3	PSA: Volba použití předděličky 1 = předdělička je předřazena časovači Watchdog 0 = předdělička je předřazena čítači/časovači Timer0							
bit2-0	Volba dělicího poměru předděličky							
	kombinace	poměr pro Timer0		poměr pro Watchdog				
	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				

Význam:

R = bit pro čtení W = bit pro zápis U = nepoužitý bit, čteno jako '0'
-n = hodnota po resetu POR '1' = bit je nastaven '0' = bit je v nule x = hodnota bitu je neznámá

Poznámka: Při přepínání předděličky od čítače/časovače Timer0 k časovači Watchdog je nutné použít bezpečnostní sekvenci instrukcí, která je popsána v dokumentaci DS33023 s názvem *PICmicro® Mid-Range MCU Family Reference Manual*. Tato sekvence musí být použita i v případě, že je časovač Watchdog v konfiguračním slově zakázán.

PIC16F87/88

Příklad 6-1: Přepnutí předděličky od časovače Watchdog k Timer0

CLRWDT		;Nuluj WDT a předděličku
BANKSEL	OPTION_REG	;Výběr banky OPTION_REG
MOVLW	b'xxxx0xxx'	;Volba TMR0, nový dělicí poměr
MOVWF	OPTION_REG	;Uložení nastavení

Tabulka 6-1: Registry, které mají vztah k čítači/časovači Timer0

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druzích resetu
01h,101h	TMR0	Pracovní registr čítače/časovače Timer0								xxxx xxxx	uuuu uuuu
0Bh,8Bh,10Bh,18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IE	INT0IF	RBIF	0000 000x	0000 000u
81h,181h	OPTION_REG	RBPŮ	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s čítačem/časovačem Timer0.

PIC16F87/88

Poznámky:

7 MODUL ČÍTAČE/ČASOVAČE Timer1

- Čítač/časovač Timer1 má následující vlastnosti:
- 16-bitový čítač/časovač je tvořen realizovanými registry TMR1H a TMR1L
- Do obou registrů lze zapisovat nebo lze číst jejich obsah
- Zdroj hodinového signálu je možné zvolit buď vnější nebo vnitřní
- Čítač/časovač může pracovat buď v synchronním nebo v asynchronním režimu
- Přetečením čítače/časovače z hodnoty 0xFFFF na 0x0000 lze vyvolat žádost o přerušení

Příznak přerušení TMR1IF je v registru PIR1. Přerušení od čítače/časovače Timer1 lze maskovat pomocí bitu TMR1IE v registru PIE1.

Čítač/časovač Timer1 má vlastní oscilátor, který může být použit jako sekundární taktovací oscilátor mikropočítače. Toho dosáhneme nastavením bitu T1RUN se současným nastavením SCS<1:0> do stavu "01". Pokud je aktivní obvod monitorující taktovací oscilátor a dojde k výpadku tohoto druhu oscilátoru, ze stavu bitu T1RUN lze usuzovat, zda je zdrojem taktu oscilátor Timer1 nebo se jedná o jiný zdroj.

Čítač/časovač Timer1 je možné rovněž použít jako obvod reálného času nezávisle na systémovém taktu s minimem vnějších součástek.

7.1 Práce s čítačem/časovačem Timer1

Řídící registr čítače/časovače Timer1 je registr T1CON. Čítač/časovač Timer1 může pracovat ve třech různých režimech:

- 16-bitový časovač
- 16-bitový synchronní čítač
- 16-bitový asynchronní čítač

V režimu časovače je Timer1 inkrementován v každém instrukčním cyklu. V režimu čítače je Timer1 inkrementován s každou náběžnou hranou na vnějším hodinovém vstupu T1CKI, když tato inkrementace je v synchronním režimu synchronizována s instrukčními cykly mikropočítače a v asynchronním režimu probíhá nezávisle.

Čítač/časovač Timer1 lze zakázat/povolit nulováním/nastavením bitu TMR1ON v registru T1CON<0>. Bitem TMR1CS v registru T1CON<1> se provádí volba zdroje hodinového signálu a tím i volba mezi čítačem (counter) nebo časovačem (timer).

Čítač/časovač Timer1 může být nulován vnitřním signálem reset, který je odvozen od činnosti modulů CCP1. Podrobněji je pojednáno v kapitole 9.1.

Jako zdroj hodinového signálu pro Timer1 je možné použít vlastní oscilátor, který se připojí na vývody RB6/PGC/T1OSO/T1CKI a RB7/PGD/T1OSI. Povolení tohoto oscilátoru se provede nastavením bitu T1OSCEN v registru T1CON<3>. V tomto případě je ignorováno nastavení příslušných bitů registru TRIS pro vývody RB6 a RB7.

PIC16F87/88

Registr 7-1: Registr T1CON – řídicí registr čítače/časovače Timer1 (adresa: 10h)

U-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON
bit 7							bit 0

bit 7 **nepoužít:** při čtení '0'

bit 6 **T1RUN:** Příznakový bit taktovacího oscilátoru

- 1 = zdrojem taktovací frekvence je oscilátor Timer1
- 0 = zdrojem taktovací frekvence není oscilátor Timer1

bit 5-4 **T1CKPS1:T1CKPS0:** volba dělicího poměru předděličky v režimu časovače

- 11 = dělicí poměr 1:8
- 10 = dělicí poměr 1:4
- 01 = dělicí poměr 1:2
- 00 = dělicí poměr 1:1

bit 3 **T1OSCEN:** volba oscilátoru mikropočítače typu LP jako zdroje hodin pro Timer1

- 1 = LP oscilátor je použit pro hodinový signál čítače/časovače Timer1
- 0 = LP oscilátor je vypnutý (inverter oscilátoru je vypnutý z důvodu minimalizace odběru)

bit 2 **T1SYNC:** volba synchronizace s instrukčními cykly v případě, že TMRCS = 1, má tento bit následující význam:

- 1 = vnější hodinový signál není synchronizován s instrukčními cykly
- 0 = vnější hodinový signál je synchronizován s instrukčními cykly v případě, že TMRCS = 0 je tento bit ignorován, Timer1 je inkrementován z vnitřního taktovacího oscilátoru

bit 1 **TMR1CS:** Volba zdroje hodinového signálu pro Timer1

- 1 = vnější (náběžná hrana signálu na vývodu T1CKI)
- 0 = vnitřní (taktovací frekvence Tosc/4)

Poznámka: Pouze pro mikropočítač PIC16F88

bit 0 **TMR1ON:** Zapnutí Timer1

- 1 = Timer1 je zapnut
- 0 = Timer1 je zastaven

Význam:

R = bit pro čtení

W = bit pro zápis

U = nepoužitý bit, čteno jako '0'

–n = hodnota po resetu POR

'1' = bit je nastaven

'0' = bit je v nule

x = hodnota bitu je neznámá

7.2 Použití Timer1 v režimu časovače

V režimu časovače je Timer1 inkrementován v každém instrukčním cyklu ($F_{osc}/4$).

Tento režim je zvolen nulováním bitu TMR1CS v registru T1CON <1>. Bit $\overline{T1SYNC}$ je ignorován.

7.3 Použití Timer1 v režimu čítače

Timer1 může pracovat v synchronním nebo asynchronním režimu, v závislosti na nastavení bitu TMR1CS. V režimu čítače je Timer1 inkrementován s každou náběžnou hranou na vnějším hodinovém vstupu T1CKI, když tato inkrementace je v synchronním režimu synchronizována s instrukčními cykly mikropočítače a v asynchronním režimu probíhá nezávisle. V režimu čítače je za první aktivní náběžnou hranu hodinového signálu považována ta hrana, které předcházela sestupná hrana tohoto signálu.

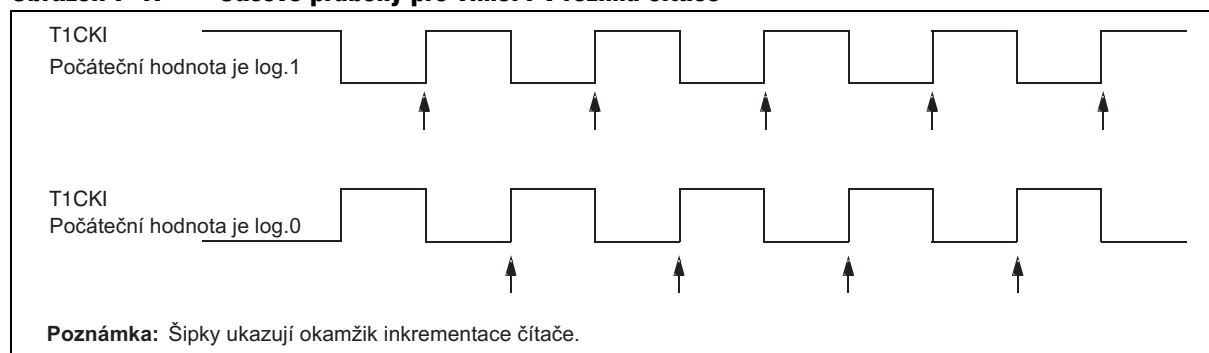
7.4 Použití Timer1 v režimu synchronního čítače.

Tento režim se zvolí nastavením bitu TMR1CS a nulováním bitu $\overline{T1SYNC}$ v registru T1CON. V tomto režimu čítač inkrementuje s každou náběžnou hranou na vývodu RB7 pokud je bit T10SCEN nastaven, nebo s každou náběžnou hranou na vývodu RB6 pokud je bit T10SCEN nulován.

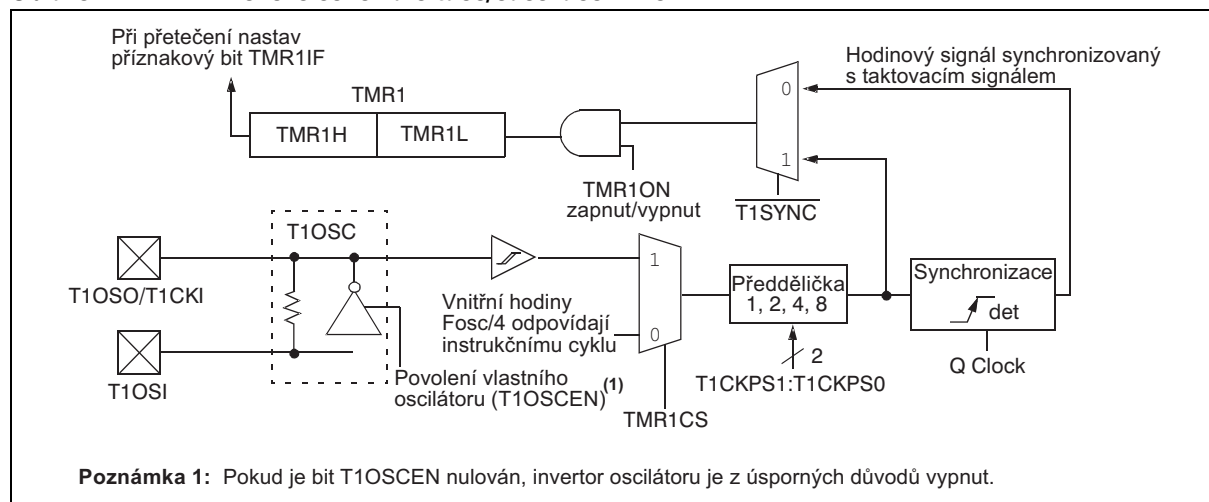
Pokud je bit $\overline{T1SYNC}$ nulován, inkrementace je synchronizována s instrukčními cykly. Synchronizace probíhá až za předděličkou, která může být případně předřazena čítači.

V režimu Sleep není funkční synchronizace hodinového signálu a tak nedochází k inkrementaci čítače. Předdělička může být v režimu Sleep inkrementována.

Obrázek 7-1: Časové průběhy pro Timer1 v režimu čítače



Obrázek 7-2: Blokové schéma čítače/časovače Timer1



7.5 Čítač/časovač Timer1 v režimu asynchronního čítače

V případě, že bit $\overline{T1SYNC}$ v registru $T1CON<2>$ je nastaven, vnější hodinový signál inkrementující čítač Timer1 není synchronizován s instrukčními cykly mikroprocesoru. Inkrementace probíhá nezávisle na taktovacím kmitočtu mikroprocesoru. Čtení může pokračovat i v režimu Sleep a přetečení čítače/časovače Timer1 může vyvolat přerušení, které způsobí probuzení procesoru.

Podrobněji je pojednáno v kapitole 7.5.1.

V tomto režimu není možné použít Timer1 pro operace Capture a Compare.

7.5.1. Čtení a zápis do registrů čítače/časovače Timer1 v režimu asynchronního čítače

Při čtení obsahu registrů TMR1H nebo TMR1L je hardwarově zajištěno bezpečné vyčtení jejich obsahu i během doby, kdy je čítač v chodu. Uživatel musí mít však na paměti skutečnost, že úplnou šestnáctibitovou hodnotu čítače získává postupným čtením obsahu dvou osmibitových registrů. Ke čtení nedochází pochopitelně ve stejný okamžik. Timer1 může tedy změnit svůj obsah i mezi těmito dvěma čteními, když v krajním případě může dojít i k přetečení čítače mezi čteními.

Při zápisu do registrů TMR1H a TMR1L je vhodné zastavit na dobu zápisu čtení. V opačném případě může snadno dojít ke konfliktu dat v registrech. Pokud by došlo k zápisu do některého z registrů Timer1 v době inkrementace čítače/časovače, může dojít k nepředvídatelnému nastavení obsahu registrů.

Čtení 16-bitového obsahu registrů vyžaduje určitou péči. Příklady správného postupu při čtení a zápisu do registrů Timer1 jsou uvedeny jako příklady 7-1 a 7-2.

Příklad 7-1: Zápis do 16-bitového volně běžícího čítače

```
; všechna přerušení jsou zakázána
CLRF      TMR1L      ;nuluj nižší bajt, aby neovlivnil TMR1H
MOVLW     HI_BYTE     ;hodnotu pro zápis do TMR1H
MOVWF     TMR1H, F    ;zápis horního bajtu
MOVLW     LO_BYTE     ;hodnotu pro zápis do TMR1L
MOVWF     TMR1H, F    ;zápis nižšího bajtu
; povolení přerušení (pokud jsou používána)
CONTINUE      ;pokračování programu
```

Příklad 7-2: Čtení 16-bitového volně běžícího čítače

```
; všechna přerušení jsou zakázána
MOVF      TMR1H, W     ;načtení vyššího bajtu
MOVWF     TMPH
MOVF      TMR1L, W     ;načtení nižšího bajtu
MOVWF     TMPL
MOVF      TMR1H, W     ;načtení vyššího bajtu
SUBWF     TMPH, W      ;odečti první načtený bajt od druhého načteného
BTFSC     STATUS, Z    ;je výsledek 0?
GOTO      CONTINUE     ;ano, jedná se o správně přečtenou hodnotu
; ne, došlo k inkrementaci čítače v době čtení
; nyní načteme hodnoty znova a dobře
MOVF      TMR1H, W     ;načtení vyššího bajtu
MOVWF     TMPH
MOVF      TMR1L, W     ;načtení nižšího bajtu
MOVWF     TMPL
;povolení přerušení (pokud jsou používána)
CONTINUE      ;pokračování programu
```

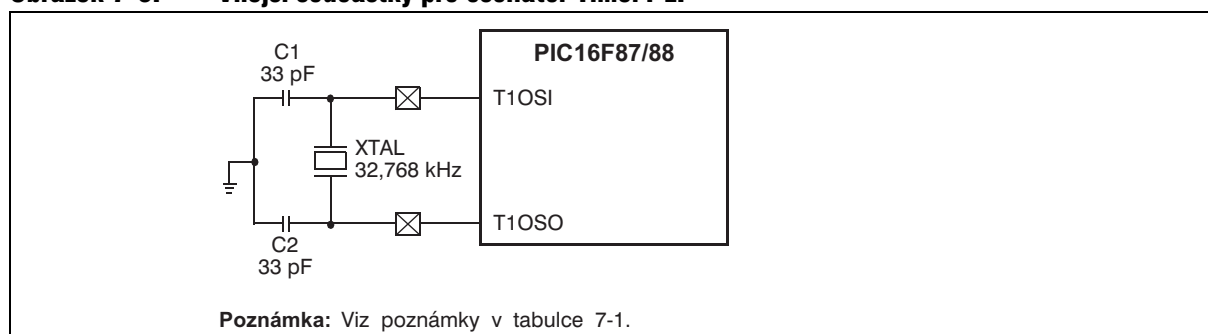
7.6 Použití vlastního oscilátoru pro Timer1

Vlastní vnější oscilátor lze připojit mezi vývody T1OSI (vstup) a T1OSO (výstup). Pro jeho použití je nutné nastavit řídicí bit T1OSCEN v registru T1CON<3>. Jedná se o nízkopříkonový typ oscilátoru LP, který pracuje do kmitočtu 32,768 kHz. Oscilátor je v činnosti i během režimu Sleep. a obrázku 7-3 je uvedeno schéma zapojení tohoto oscilátoru. V tabulce 7-1 jsou uvedeny velikosti kondenzátorů pro připojení k LP oscilátoru.

Vlastní vnější oscilátor pro Timer1 je identický s taktovacím oscilátorem typu LP. Stejně jako v případě použití LP oscilátoru pro taktování mikropočítače musí uživatel zajistit programem časovou prodlevu pro bezpečný rozběh oscilátoru.

Poznámka: Oscilátor Timer1 používá stejné vývody jako obvod pro programování ICSP a ladění v aplikaci ICD. Pokud je použit oscilátor Timer1, nemusí systém ICSP fungovat normálně nebo může docházet k výpadkům při komunikaci systému ICD. Výsledkem snah o používání režimu ICSP nebo systému ICD může být i poškození oscilátoru. Pokud používáte ICSP nebo ICD, je nutné odpojit krystal od oscilátoru nebo ho naopak připojit až po provedení programování. Vazební kondenzátory oscilátoru mohou být v zapojení ponechány.

Obrázek 7-3: Vnější součástky pro oscilátor Timer1 LP



Tabulka 7-1: Volba kondenzátorů pro vlastní oscilátor Timer1

Typ oscilátoru	Frekvence	C1	C2
LP	32 kHz	33 pF	33 pF

- Poznámka 1:** Tyto hodnoty jsou navrženy firmou Microchip jako výchozí
- 2:** Vyšší hodnoty kapacit zvyšují stabilitu oscilátoru ale prodlužují dobu ustálení po zapnutí
- 3:** Každý krystal nebo rezonátor má lehce odlišné parametry (viz údaje výrobce).
- 4:** Uvedené hodnoty slouží pouze pro potřeby návrhu

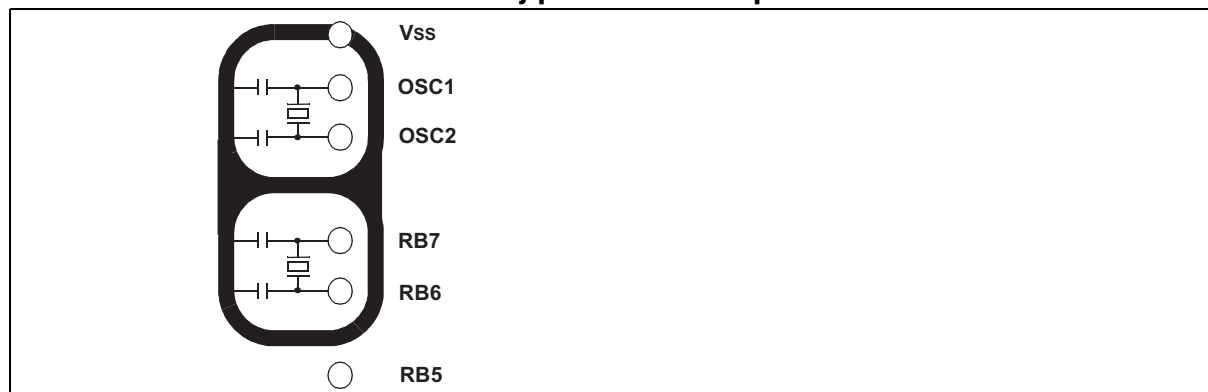
7.7 Návrhová pravidla pro plošný spoj oscilátoru Timer1

Oscilátor Timer1 odebírá během provozu velmi malý proud. Z tohoto důvodu je oscilátor relativně citlivý na vnější vlivy, zejména na rychlé změny proudů ve svém okolí.

Obvod oscilátoru, jak je naznačeno na obrázku 7-3, by měl být řešen co možná nejlépe pouzdra mikropočítače. Obvodem oscilátoru by neměl procházet žádný další vodič v okolí oscilátoru by se neměly vyskytovat jiné vodiče než měl napájecí, tedy napětí V_{DD} nebo V_{SS}.

Pokud musí být v blízkosti LP oscilátoru Timer1 umístěn jiný oscilátor pracující na vyšších frekvencích je velice vhodné odstínění oscilátoru po obou stranách plošného spoje. Řešení je znázorněné na obrázku 7-4.

Obrázek 7-4: Oscilátor Timer1 odstíněný pomocí zemnicího prstence



7.8 Nulování čítače/časovače Timer1 s použitím modulu CCP

Pokud je modul CCP1 nastaven do porovnávacího režimu Compare, mohou být aktivní některé další funkce Timer1. Pokud budou bity CCP1M3:CCP1M0 ve stavu **1011**, bude čítač Timer1 nulován a současně bude zahájen AD převod (pokud je AD převod povolen).

Poznámka: Tyto další funkce nenastavují příznakový bit přetečení Timer1 TMR1IF.

Aby byly aktivní další funkce, musí být Timer1 konfigurován jako časovač nebo synchronní čítač. Pokud je Timer1 konfigurován jako asynchronní čítač, není tato funkce nulování funkční.

V případě, že zápis do registrů Timer1 koliduje s další funkcí modulu CCP1, má přednost tento zápis.

V tomto režimu jsou registry CCPR1H a CCPR1L použity jako registry periody čítače/časovače Timer1.

7.9 Nulování čítače/časovače Timer1 (resp. dvojice registrů TMR1H a TMR1L)

K nulování dvojice registrů TMR1H a TMR1L které tvoří čítač/časovač Timer1 nedochází při žádném z druhů resetu. Výjimku tvoří nulování způsobené modulem CCP1.

Registr T1CON je nulován pouze resetem po zapnutí napájecího napětí (POR) a resetem při výpadku napájecího napětí (BOD). Ostatní druhy resetu tento registr neovlivňují.

7.10 Předdělička čítače/časovače Timer1

Předdělička je nulována při každém zápisu do registrů TMR1H a TMR1L.

7.11 Použití Timer1 jako obvodu reálného času

Použití vnějšího LP oscilátoru Timer1 (viz **kapitulu 7.6**) dává uživateli možnost využít v aplikaci obvod reálného času, nezávislého na systémovém taktovacím oscilátoru. Pro toto použití se předpokládá využití levného hodinového krystalu ve spojení s jednoduchou programovou rutinou. Pokud bude toto řešení kombinováno s využitím režimu Sleep ve spojení s vlastní baterií nebo velkokapacitním kondenzátorem, odpadá nutnost použití obvyklého externího zdroje informace o aktuálním čase.

Na obrázku 7-3 je uveden příklad programové rutiny, která generuje každou sekundu přerušení. Přetečení registrů Timer1 je v rutině obsluhy přerušení následně obslouženo inkrementací minutového a následně hodinového počítadla.

Čítač Timer1 tvořený dvěma registry je celkově 16. bitový. Při použití krystalu o frekvenci 32,768 kHz dojde k jeho přetečení každé 2 sekundy. Nejjednodušším způsobem jak vytvořit sekundové intervaly je přednastavení nejvyššího bitu čítače pomocí instrukce BSF. Pamatujte však, že registr TMR1L není tímto způsobem nikdy nastavován a dochází k chybám, které se projeví kumulovaně po delší době jako nepřesnost odečítání času.

Aby byla uvedená metoda přesná, je nutné použít Timer1 v režimu asynchronního čítače, jehož přetečení vyvolá přerušení, které musí být povoleno v registru PIE1, jak je uvedeno v programové rutině **RTCinit** v příkladu 7-3. Oscilátor čítače/časovače Timer1 musí být rovněž povolen a dále musí být v činnosti nepřetržitě po celou dobu udržování reálného času.

Příklad 7-3: Použití Timer1 a přerušení pro funkci reálného času

RTCinit	BANKSEL	TMR1H		
	MOVLW	0x80		; Nastavení dvojice registrů TMR1
	MOVWF	TMR1H		; pro interval 1 sekunda
	CLRF	TMR1L		
	MOVLW	b'00001111'		; konfigurace pro vnější hodinový signál
	MOVWF	T1CON		; asynchronní režim, externí oscilátor
	CLRF	secs		; nulování registrů pro uložení času
	CLRF	mins		
	MOVLW	.12		
	MOVWF	hours		
	BANKSEL	PIE1		
	BSF	PIE1, TMR1IE		; povolení přerušení Timer1
	RETURN			
RTCisr	BANKSEL	TMR1H		
	BSF	TMR1H, 7		; nastavení na přetečení každou 1 sekundu
	BCF	PIR1, TMR1IF		; nuluj příznak přerušení
	INCF	secs, F		; inkrementuj počítadlo sekund
	MOVF	secs, w		
	SUBLW	.60		
	BTFS	STATUS, Z		; už uplynulo 60 sekund?
	RETURN			; ne
	CLRF	seconds		; ano, nuluj počítadlo sekund
	INCF	mins, f		; inkrementuj počítadlo minut
	MOVF	mins, w		
	SUBLW	.60		
	BTFS	STATUS, Z		; už uplynulo 60 minut
	RETURN			; ne
	CLRF	mins		; ano, nuluj počítadlo minut
	INCF	hours, f		; inkrementuj počítadlo hodin
	MOVF	hours, w		
	SUBLW	.24		
	BTFS	STATUS, Z		; už uplynulo 24 hodin?
	RETURN			; ne
	CLRF	hours		; ano, nuluj počítadlo hodin
	RETURN			; návrat

Tabulka 7-2: Registry, které mají vztah k čítači/časovači Timer1

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druzích resetu
0Bh,8Bh,10Bh,18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IE	INT0IF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF (1)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
8Ch	PIE1	—	ADIE (1)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
0Eh	TMR1L	Pracovní registr pro dolních 8 bitů čítače/časovače Timer1								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Pracovní registr pro horních 8 bitů čítače/časovače Timer1								xxxx xxxx	uuuu uuuu
81h,181h	T1CON	—	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNCR	TMR1CS	TMR1ON	-000 0000	-uuu uuuu

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s čítačem/časovačem Timer1.

Poznámka 1: Tento bit je použit jen u mikropočítače PIC16F88. U mikropočítače PIC16F87 je čten jako nula.

PIC16F87/88

Poznámky:

8 MODUL ČÍTAČE/ČASOVAČE Timer2

Čítač/časovač Timer2 je osmibitový čítač, ke kterému je možné předřadit předděličku. Další předděličku je možné zařadit za čítač/časovač Timer2. Modul Timer2 je možné použít jako časovou základnu výstupu pulsně šířkové modulace (PWM). Pracovní registr TMR2 lze libovolně číst i do něj zapisovat. Po každém druhu resetu je Timer2 nulován.

Modul čítače/časovače Timer2 je vybaven registrem periody PR2. Registr TMR2 je inkrementován od hodnoty 00h do hodnoty uložené v registru periody PR2. Po resetu je hodnota PR2 rovna FFh. Během inkrementace registru TMR2 se hodnota registru PR2 nemění.

Vstupní hodinový signál je odvozen od taktovacího oscilátoru. K inkrementaci čítače Timer2 dochází v každém instrukčním cyklu, pokud není předřazena předdělička. Dělicí poměr předděličky je určen nastavením bitů T2CKPS1 a T2CKPS2 v registru T2CON. Dělicí poměr může být buď 1:1, 1:4 nebo 1:16, když poměr je vztažen k frekvenci instrukčních cyklů.

Výstupní signál z čítače/časovače Timer2 odpovídá přetečení registru TMR2 z hodnoty FFh na 00h. Tento výstupní signál může být dále vydělen pomocí další předděličky s dělicím poměrem od 1:1 do 1:16, když dělicí poměr je možné nastavit plynule v celém rozsahu hodnot pomocí bitů TOUTPS3 – TOUTPS0 v registru T2CON.

Výstupní signál může po případném vydělení druhou předděličkou způsobit přerušení od čítače/časovače Timer2. K tomu je nutné povolit tento druh přerušení nastavením bitu TMR2IE v registru PIE1. Výstupní signál rovněž nastavuje příznak přetečení TMR2IF v registru PIR1 bez ohledu na to, zda je přerušení povoleno či nikoliv.

Modul čítače/časovače může být vypnut nulováním řídicího bitu TMR2ON v registru T2CON. Ve vypnutém stavu je minimalizována spotřeba proudu.

V registru 8-1 je popsán řídicí registr čítače/časovače Timer2.

Další informace jsou uvedeny v dokumentaci DS33023 s názvem *PICmicro® Mid-Range MCU Family Reference Manual*.

8.1 Předděličky čítače/časovače Timer2

Obě předděličky čítače/časovače Timer0 jsou nulovány při jakékoli z následujících podmínek:

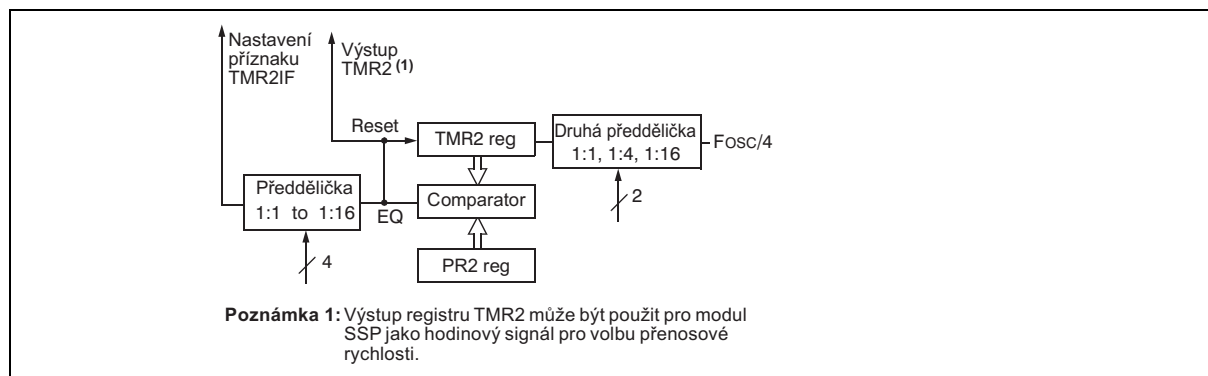
- zápis do registru TMR2
- zápis do registru T2CON
- jakýkoli druh resetu

Registr TMR2 není nulován při zápisu do registru T2CON.

7.2 Výstup z registru TMR2

Výstup z registru TMR2 (před druhou předděličkou) je přiveden do modulu SSP, kde může být využit pro hodinový signál při přenosu dat.

Obrázek 8-2: Blokové schéma Timer2



PIC16F87/88

Registř 8-1: Registr T2CON – řídicí registr čítače/časovače Timer2 (adresa: 12h)

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0
bit 7							bit 0

- bit 7 **nepoužit:** při čtení '0'
- bit 6-3 **TOUTPS3:TOUTPS0:** volba dělicího poměru druhé předděličky zařazené za TMR2
 0000 = dělicí poměr 1:1
 0001 = dělicí poměr 1:2
 0010 = dělicí poměr 1:3
 •
 •
 1111 = dělicí poměr 1:16
- bit 2 **TMR2ON:** zapnutí modulu čítače/časovače Timer2
 1 = Timer2 je zapnut
 0 = Timer2 je vypnutý
- bit 1-0 **T2CKPS1:T2CKPS0:** volba dělicího poměru předděličky zařazené před TMR2
 00 = dělicí poměr 1:1
 01 = dělicí poměr 1:4
 1x = dělicí poměr 1:16

Význam:

R = bit pro čtení

W = bit pro zápis

U = nepoužitý bit, čteno jako '0'

–n = hodnota po resetu POR

'1' = bit je nastaven

'0' = bit je v nule

x = hodnota bitu je neznámá

Tabulka 8-1: Registry, které mají vztah k čítači/časovači Timer2

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druzích resetu
0Bh, 8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IE	INT0IF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF (1)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
8Ch	PIE1	—	ADIE (1)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 000	-000 0000
11h	TMR2	Pracovní registr čítače/časovače Timer2								0000 0000	0000 0000
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
92h	PR2	Registr periody čítače/časovače Timer2								1111 1111	1111 1111

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s čítačem/časovačem Timer2.

Poznámka 1: Tento bit je použit jen u mikropočítače PIC16F88. U mikropočítače PIC16F87 je čten jako nula.

9 MODUL CCP (Capture/Compare/PWM)

Mikropočítač je vybaven modulem Capture/Compare/PWM (CCP). Modul obsahuje 16. bitový registr, který může pracovat v režimu:

- 16-bitový registr v záchytném systému
- 16-bitový registr v komparačním systému
- Pulsně šířkový modulátor PWM

V tabulce 9.1 jsou uvedeny možná nastavení modulu CCP.

Pracovní 16-bitový registr CCPR1 sestává ze dvou osmibitových registrů CCPR1L (nižší bajt) a CCPR1H (vyšší bajt). Řídící registr se nazývá CCP1CON. Modul může generovat událost, která způsobí nulování registru čítače/časovače Timer1 nebo zahájení AD převodu (pokud je AD převodník povolen).

Vstup nebo výstup modulu CCP1 může být konfigurován na vývodu RB0 nebo RB3. Volba se provádí pomocí bitu CCPMX, což je 12. bit konfiguračního slova mikropočítače.

Další informace jsou uvedeny v dokumentaci DS33023 s názvem "PICmicro® Mid-Range MCU Family Reference Manual" a dále v dokumentaci DS00594 s názvem "Using the CCP Module(s)" a v Aplikační příručce jako poznámce AN594.

Tabulka 9-1: Čítače používané v různých režimech CCP

Režim CCP	Čítač
Capture	Timer1
Compare	Timer1
PWM	Timer2

Registr 9-1: Registr CCP1CON – řídicí registr Capture/Compare/PWM (adresa: 17h)

	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0
	bit 7							bit 0
bit 7-6	nepoužity: při čtení '0'							
bit 5-4	CCP1X:CCP2Y: nejméně významné bity PWM režim Capture: bez významu režim Compare bez významu režim PWM: tyto bity jsou dva nejméně významné bity pro nastavení střídý PWM. Osm významnějších bitů je v registru CCPR1L							
bit 3-0	CCP1M3:CCP1M0: volba režimu modulů CCP 0000 = modul CCPx je vypnut 0100 = režim záchytného systému Capture, k záchytu dojde s každou sestupnou hranou 0101 = režim záchytného systému Capture, k záchytu dojde s každou náběžnou hranou 0110 = režim záchytného systému Capture, k záchytu dojde s každou 4. náběžnou hranou 0111 = režim záchytného systému Capture, k záchytu dojde s každou 16. náběžnou hranou 1000 = režim komparačního systému Compare, při shodě je nastaven výstupní vývod CCPx 1001 = režim komparačního systému Compare, při shodě je nulován výstupní vývod CCPx 1010 = režim komparačního systému Compare, při shodě je aktivováno přerušování 1011 = režim komparačního systému Compare – speciální funkce (nastavuje příznak přerušování) pro modul CCP1: při shodě je nulován obsah Timer1 a stav vývodu není ovlivněn pro modul CCP2: při shodě je nulován obsah Timer1 a spuštěn AD převod 11xx = režim PWM							

Význam:

R = bit pro čtení

–n = hodnota po resetu POR

W = bit pro zápis

'1' = bit je nastaven

U = nepoužitý bit, čteno jako '0'

'0' = bit je v nule

x = hodnota bitu je neznámá

V záhytném režimu Capture dochází k uložení 16-bitové hodnoty čítače Timer1 do pracovních registrů CCPR1H a CCPR1L současně s událostí na vnějším vývodu CCP1. Zachycení může nastat:

- s každou sestupnou hranou
- s každou náběžnou hranou
- s každou 4. náběžnou hranou
- s každou 16. náběžnou hranou

Druh události je určen nastavením řídicích bitů CCP1M3:CCP1M0. Při zachycení hodnoty dojde k nastavení příznaku přerušení CCP1IF. Tento příznak musí být nulován softwarově. Pokud dojde k další události dříve než jsou data z pracovních registrů přečtena, dojde ke ztrátě těchto dat.

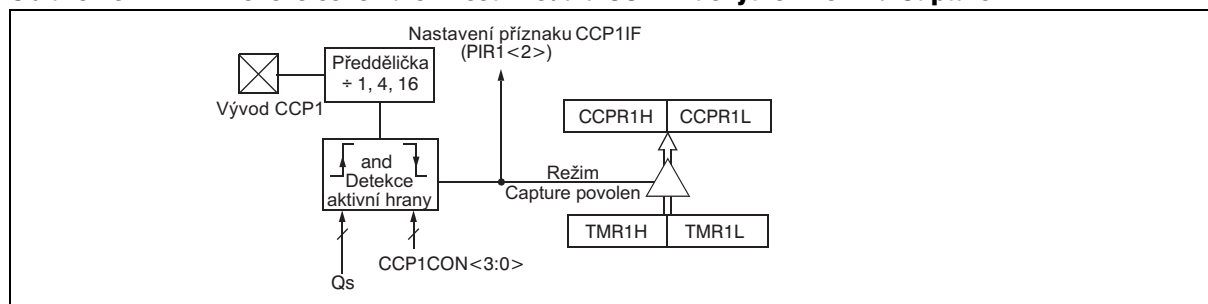
9.1.1 Konfigurace vývodů CCP

V režimu záchytného systému Capture musí být vývod CCP1 konfigurován jako vstup nastavením příslušného bitu v registru TRISB.

Poznámka 1: Pokud je vývod CCP1 konfigurován jako výstup, může zápis do výstupního záchytného registru způsobit událost CCP.

2: Nastavení bitů registru TRIS (TRISB0 nebo TRISB3) musí odpovídat nastavení 12. bitu konfiguračního registru mikropočítače (CCPMX).

Obrázek 9-1: Blokové schéma činnosti modulu CCP v záchytném režimu Capture



9.1.2 Nastavení čítače/časovače Timer1

Pro činnost zachytného systému v režimu Capture musí být čítač/časovač Timer1 spuštěn v režimu časovače nebo synchronního čítače. Pokud by byl čítač/časovač v režimu asynchronního čítače, není možné jej použít pro správnou činnost zachytného systému v režimu Capture.

9.1.3 Nežádoucí přerušení

Při změně nastavení zachytného systému v režimu Capture může dojít k nežádoucím přerušením. Uživatel musí toto ošetřit podržením bitu CCP1IE v registru PIE v nule a dále nulováním příznakového bitu CCP1IF při každé změně nastavení režimu Capture.

9.1.4 Předdělička CCP

Předdělička modulu CCP má celkem čtyři možná nastavení podle bitů CCP1M3 – CCP1M0. Každé vypnutí modulu CCP způsobí nulování předděličky. Předdělička je nulována dále vždy, pokud není modul CCP v režimu Capture. Každý reset způsobí rovněž nulování předděličky.

Změna dělicího poměru předděličky může způsobit přerušení. Tato změna nenuluje předděličku, proto první záchyt dat může proběhnout z náhodného stavu předděličky. Na příkladu 9-1 je uveden doporučený postup pro přepínání dělicích poměrů předděličky.

Příklad 9-1: Přepínání dělicích poměrů předděličky

CLRF	CCP1CON	;vypnutí modulu CCP
MOVLW	NEW_CAPT_PS	;načtení do střadače nové hodnoty
		;dělicího poměru a příprava zapnutí CCP
MOVWF	CCP1CON	;uložení do řídicího registru

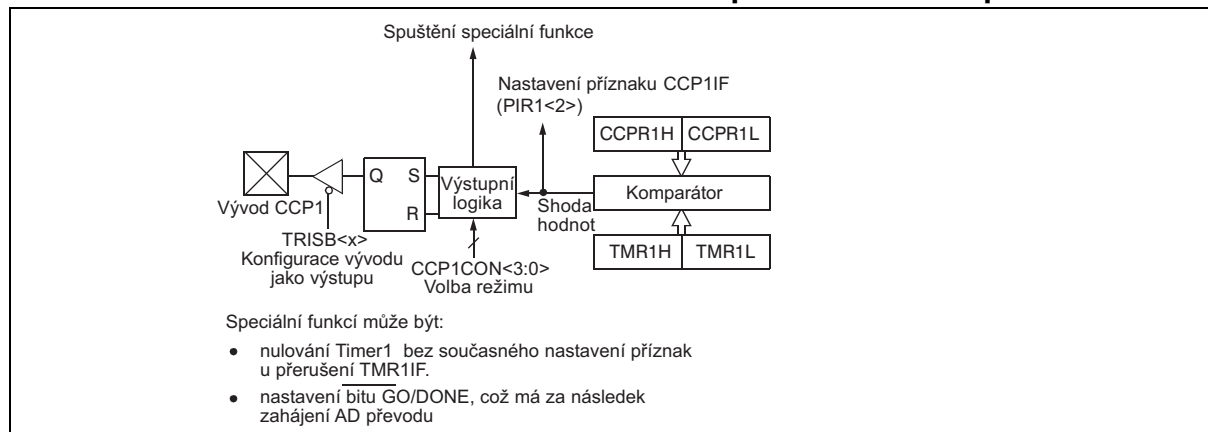
9.2 Režim komparačního systému Compare

V režimu komparačního systému Compare dochází k neustálému porovnávání hodnoty pracovních registrů CCPR1 s hodnotou pracovního registru čítače/časovače Timer1. Při shodě dojde k jedné z následujících událostí na výstupním vývodu RC2/CCP1:

- nastavení do log.1
- nulování do log.0
- výstup beze změny, je vyvoláno přerušení

Druh události je nastaven pomocí řídicích bitů CCP1M3-CCP1M0. Při každé události dojde k nastavení příznaku přerušení CCP1IF.

Obrázek 9-2: Blokové schéma činnosti modulu CCP v komparačním režimu Compare



9.2.1 Konfigurace vývodů CCP

V režimu záchytného systému Capture musí být vývod CCP1 konfigurován jako výstup nastavením příslušného bitu v registru TRISB.

Poznámka 1: Nulování registru CCP1CON způsobí nastavení výstupního vývodu CCP do stavu log. 0. Nemá to nic společného s obsahem pracovního registru PORTB.

- 2:** Nastavení bitů registru TRIS (TRISB0 nebo TRISB3) musí odpovídat nastavení 12. bitu konfiguračního registru mikropočítače (CCPMX).

9.2.2 Nastavení čítače/časovače Timer1

Pro činnost komparačního systému v režimu Compare musí být čítač/časovač Timer1 spuštěn v režimu časovače nebo synchronního čítače. Pokud by byl čítač/časovač v režimu asynchronního čítače, není možné jej použít pro správnou činnost komparačního systému v režimu Compare.

9.2.3 Využití přerušení

Jednou z možností komparačního systému v režimu Compare je aktivace přerušení bez změny na vývodu CCP. Při shodě dojde pouze k nastavení příznaku přerušení CCP1IF a vyvolání přerušení (pokud je povoleno).

9.2.4 Speciální funkce

Ve speciálním funkčním režimu je možné využít shodu hodnot pro aktivaci následující události.

Při shodě dojde k nulování 16 bitového čítače/časovače Timer1 a současně k zahájení AD převodu, pokud je AD převodník povolen. Registr CCPR1 tak může být použit jako registr periody čítače/časovače Timer1.

Poznámka: Speciální funkce modulu CCP1 nenastavuje příznak přerušení TMR1IF v registru PIR1 <0>.

Tabulka 9-2: Registry, které mají vztah k modulu CCP v záchytném režimu Capture a komparačním režimu Compare a k modulu Timer1

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druhých resetu
0Bh,8Bh,10Bh,18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IE	INT0IF	RBF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF (1)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
8Ch	PIE1	—	ADIE (1)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
86h	TRISB	Konfigurační registr brány PortB								1111 1111	1111 1111
0Eh	TMR1L	Pracovní registr dolní poloviny čítače/časovače Timer1								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Pracovní registr horní poloviny čítače/časovače Timer1								xxxx xxxx	uuuu uuuu
10h	T1CON	—	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	-000 0000	-uuu uuuu
15h	CCPR1L	Pracovní registr CCP1 (dolní polovina 16 bitové hodnoty)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Pracovní registr CCP1 (horní polovina 16 bitové hodnoty)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s režimem Capture, Compare nebo modulem Timer1.

Poznámka 1: Tento bit je použit jen u PIC16F88, u mikropočítače PIC16F87 je čten jako "0"

9.3 Režim Pulsně šířkového modulátoru PWM

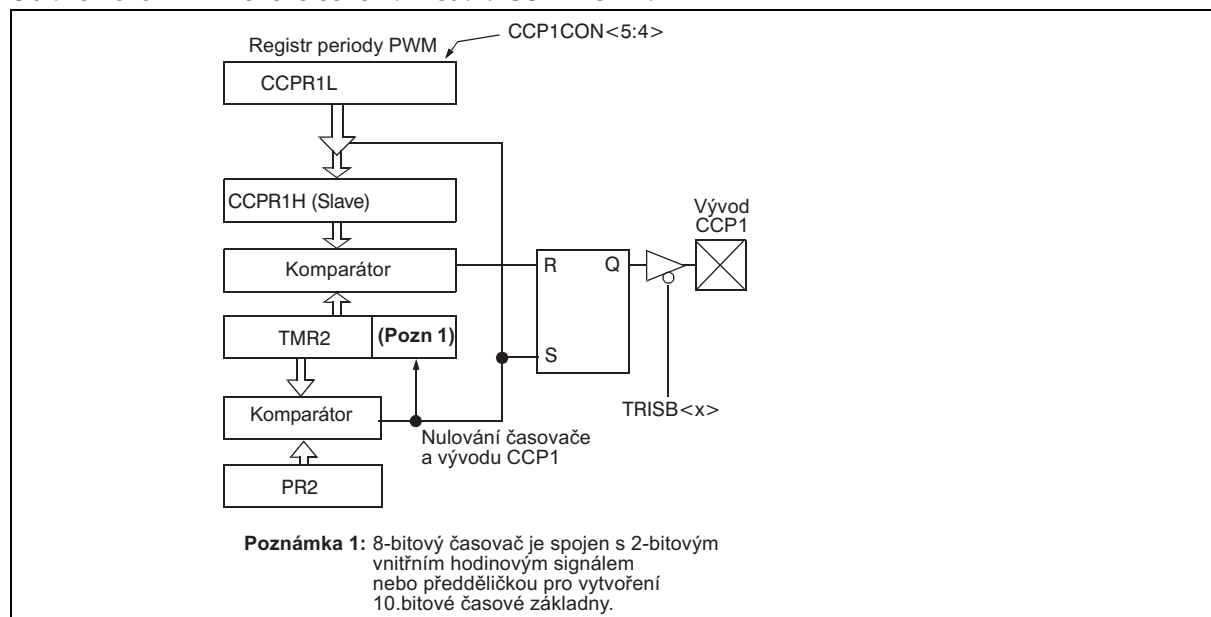
V režimu PWM je na výstupu CCP přítomen pulsně šířkově modulovaný signál s rozlišením až 10 bitů. Příslušný bit registru TRISB musí být vynulován, aby vývod CCP byl nastaven jako výstupní.

Poznámka: Vynulování registru CCP1CON způsobí nastavení výstupního vývodu CCP do stavu log.0. Nemá to nic společného se stavem pracovního registru PORTB.

Na obrázku 9-3 je znázorněno jednoduché blokové schéma modulu CCP v režimu PWM.

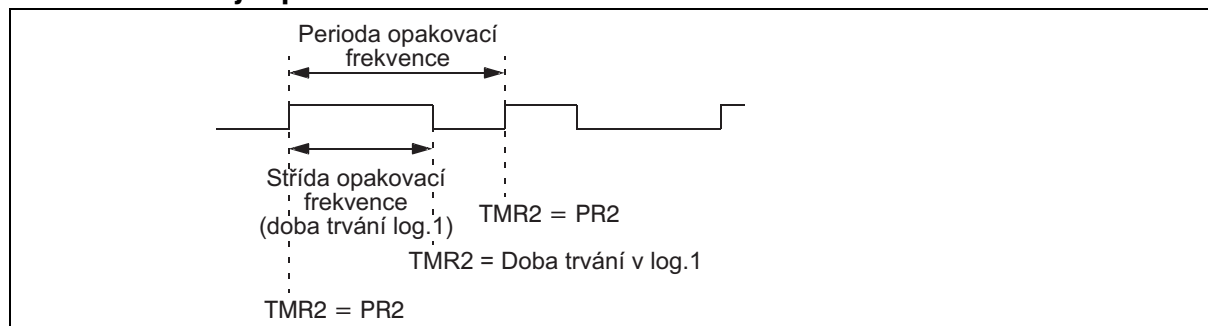
Podrobněji je pojednáno v kapitole 9.3.3.

Obrázek 9-3: Blokové schéma modulu CCP v režimu PWM



Výstup PWM je v pravidelných intervalech nastavován do stavu log.1. Perioda tohoto nastavování odpovídá základní opakovací frekvenci PWM. K nulování výstupu dochází podle hodnoty uložené v registru periody. Situace je znázorněna na obrázku 9-4.

Obrázek 9-4: Výstup PWM



9.3.1 Perioda opakovací frekvence PWM

Opakovací frekvence PWM se nastavuje v registru PR2. Perioda opakovací frekvence je dána vzorcem.

Vzorec 9-1:

$$\text{Perioda PWM} = [(PR2) + 1] \cdot 4 \cdot T_{osc} \cdot (\text{dělicí poměr předěličky Timer2})$$

Opakovací frekvence je rovna převrácené hodnotě periody.

Pokud je hodnota registru TMR2 rovna hodnotě registru PR2 dojde k následujícím třem událostem:

- registr TMR2 je vynulován
- výstupní vývod CCP je nastaven do log.1 (vyjma případu, kdy doba trvání v log 1 je 0 % periody)
- hodnota střidy je zapsána z registru CCPR1L do registru CCPR1H

Poznámka: Druhá předělička zařazená za čítač/časovač Timer2 nemá vliv na opakovací frekvenci PWM. Podrobněji viz kapitolu 8.0.

9.3.2 Střída opakovací frekvence PWM

Střidu opakovací frekvence lze nastavit v rozsahu 1024 hodnot nastavením registru CCPR1L a dvou bitů v registru CCP1CON. V registru CCPR1L je obsaženo osm významnějších bitů rozlišení střidy. Pro stanovení střidy PWM lze použít následující vzorec:

Vzorec 9-2:

$$\text{Střída PWM} = (CCPR1L:CCP1CON<5:4>) \cdot T_{osc} \cdot (\text{dělicí poměr předěličky Timer2})$$

Registry pro nastavení střidy CCPR1L a CCP1CON mohou být modifikovány kdykoli, ale nastavená změna střidy se projeví až po ukončení periody opakovací frekvence, kdy dojde k zápisu do registru CCPR1H. V režimu PWM je registr CCPR1H určen jen ke čtení.

Registr CCPR1H a dvou bitový vnitřní záchytný obvod pracují jako dvojitý buffer. Toto řešení umožňuje plynulé generování signálu PWM. Nejvyšší rozlišení PWM pro danou opakovací frekvenci lze zjistit ze vzorce 9-3:

Vzorec 9-3:

$$\text{Rozlišení} = \frac{\log\left(\frac{F_{osc}}{F_{PWM}}\right)}{\log(2)} \text{ bitů}$$

Poznámka: Pokud je nastavená doba pro určení střidy delší než celková perioda opakovací frekvence, nebude výstup PWM nulován.

9.3.3 Nastavení pro obsluhu PWM

Při práci s PWM je nutné provést následující kroky:

1. Nastavit periodu opakovací frekvence PWM v registru PR2
2. Nastavit střídu PWM zápisem do registru CCPR1L a dvou bitů registru CCP1CON
3. Konfigurovat vývod CCP1 jako výstup pomocí registru TRISC
4. Nastavit předděličku pro Timer2 a povolit Timer2 zápisem do registru T2CON
5. Konfigurovat modul CCP v režimu PWM

Poznámka 2: Nastavení bitů registru TRIS (TRISB0 nebo TRISB3) musí odpovídat nastavení 12. bitu konfiguračního registru mikropočítače (CCPMX).

Tabulka 9-3: Příklad možných opakovacích frekvencí PWM a rozlišení pro takt 20 MHz

Opakovací frekvence PWM	1,22 kHz	4,88 kHz	19,53 kHz	78,12 kHz	156,3 kHz	208,3 kHz
Dělicí poměr předděličky (1,4,16)	16	4	1	1	1	1
Hodnota v registru PR2	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
Maximální rozlišení v bitech	10	10	10	8	7	6,6

Tabulka 9-4: Registry, které mají vztah k modulu CCP v režimu PWM a k modulu Timer2

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druhích resetu
0Bh,8Bh,10Bh,18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IE	INT0IF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF (*)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
8Ch	PIE1	—	ADIE (*)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
86h	TRISB	Konfigurační registr brány PortB								1111 1111	1111 1111
11h	TMR2	Pracovní registr čítače/časovače Timer2								0000 0000	0000 0000
92h	PR2	Registr periody PWM čítače/časovače Timer2								1111 1111	1111 1111
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
15h	CCPR1L	Pracovní registr CCP1 (dolní polovina 16 bitové hodnoty)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Pracovní registr CCP1 (horní polovina 16 bitové hodnoty)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	- -00 0000	- -00 0000

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s režimem PWM nebo modulem Timer2.

Poznámka 1: Tento bit je použit jen u PIC16F88, u mikropočítače PIC16F87 je čten jako "0"

10 MODUL SYNCHRONNÍHO SÉRIOVÉHO PORTU (SSP)

10.1 Popis modulu SSP

Modul synchronního sériového portu (SSP – Synchronous Serial Port) je sériové rozhraní, používané pro komunikaci s dalšími periferiemi nebo jednočipovými mikropočítači. Tato periferní zařízení mohou být sériové EEPROM paměti, posuvné registry, řadiče displejů, A/D převodníky atd. Modul MSSP může pracovat v jednom ze dvou režimů:

- Serial Peripheral Interface (SPI™)
- Inter-Integrated Circuit (I²C™)

Další informace jsou uvedeny v dokumentaci DS33023 s názvem "*PICmicro® Mid-Range MCU Family Reference Manual*" a dále v dokumentaci DS00578 s názvem "*I²C Multi-Master Environment*" a v Aplikační příručce jako poznámce AN578.

10.2 SPI režim

SPI režim dovoluje souběžné vysílání a přijímání 8-bitových dat. Jsou podporované všechny 4 režimy SPI. Pro realizaci komunikace se obvykle používají tři vývody:

- sériový výstup (SDO) Serial Data Out – RB2/SDO/RX/DT
- sériový vstup (SDI) Serial Data In – RB1/SDI/SDA
- hodinový signál (SCK) Serial Clock – RB4/SCK/SCL

Navíc může být čtvrtý vývod použit v podřízeném režimu:

- výběr periferie – Slave Select ($\overline{SS}$) – RB5/ $\overline{SS}$ /TX/CK

Při použití režimu SPI je nutné specifikovat několik dalších parametrů. Toto se provádí pomocí odpovídajících bitů v registrech SSPCON a SSPSTAT. Kombinace těchto bitů umožňuje následující nastavení:

- režim Master (vývod SCK je výstup hodinového signálu)
- režim Slave (vývod SCK je vstup hodinového signálu)
- polarita hodinového signálu
- aktivní hrana (platnost dat může určovat buď náběžná nebo sestupná hrana)
- rychlost hodinového signálu (jen v režimu Master)
- výběr režimu Slave (jen v režimu Slave)

Poznámka: Před povolením modulu SPI v režimu Slave musí úroveň hodinového signálu na vývodu odpovídat nastavení polarity hodinového signálu. Logická úroveň může být zjištěna čtením vývodu SCK. Polarita je určena bitem CKP v registru SSPCON<4>.

PIC16F87/88

Registr 10-1: SSPSTAT: stavový registr synchronního portu SPI (adresa 94h)

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	D/Ā	P (1)	S (1)	R/W	UA	BF
bit 7							bit 0

- bit 7 **SMP:** bit určující vzorkování
 SPI v řídicí režimu (Master mode):
 1 = vstupní data jsou snímána po odeslání výstupních dat
 0 = vstupní data jsou snímána uprostřed odesílání výstupních dat
 SPI závislý režim (Slave mode):
 bit musí být v tomto režimu nulován
 I²C režim
 bit musí být v tomto režimu nulován
- bit 6 **CKE:** výběr hodinového signálu
 1 = platnost dat nastane při změně hodinového signálu z aktivního na klidový stav
 0 = platnost dat nastane při změně hodinového signálu z klidového na aktivní stav
Poznámka: Polarita hodinového signálu je nastavena bitem CKP v registru SSPCON1 <4>
- bit 5 **D/Ā:** bit pro výběr data/adresa
 používá se pouze v I²C režimu
- bit 4 **P:** Stop bit (ukončení komunikace)
 používá se pouze v I²C režimu
 1 = Stop bit byl detekován
 0 = Stop bit nebyl detekován
- bit 3 **S:** Start bit (zahájení komunikace)
 používá se pouze v I²C režimu
 1 = Start bit byl detekován
 0 = Stop bit nebyl detekován
- bit 2 **R/W:** informace o čtení/zápisu
 používá se pouze v I²C režimu
 uchovává informaci po poslední platné adrese nejpozději do dalšího Start bitu, Stop bitu nebo bitu ĀCK
 1 = čtení
 0 = zápis
- bit 1 **UA:** aktualizace adresy
 používá se pouze v 10. bitovém I²C režimu
 1 = je potřeba aktualizovat adresu v registru SSPADD
 0 = není třeba aktualizovat adresu
- bit 0 **BF:** Příznakový bit plného zásobníku
 Příjem v režimu SPI nebo I²C
 1 = příjem je kompletní, SSPBUF je plný
 0 = příjem není kompletní, SSPBUF je prázdný
 Vysílání v režimu I²C
 1 = probíhá vysílání, SSPBUF je plný (8 bitů)
 0 = vysílání je dokončeno, SSPBUF je prázdný
Poznámka 1: Tento bit je nulován pokud je modul SSP zakázán (tedy pokud je nulován bit SSPEN)

Význam:

R = bit pro čtení

W = bit pro zápis

U = nepoužitý bit, čteno jako '0'

–n = hodnota po resetu POR

'1' = bit je nastaven

'0' = bit je v nule

x = hodnota bitu je neznámá

Registř 10-2: SSPCON – řídicí registr synchronního portu (adresa 14h)

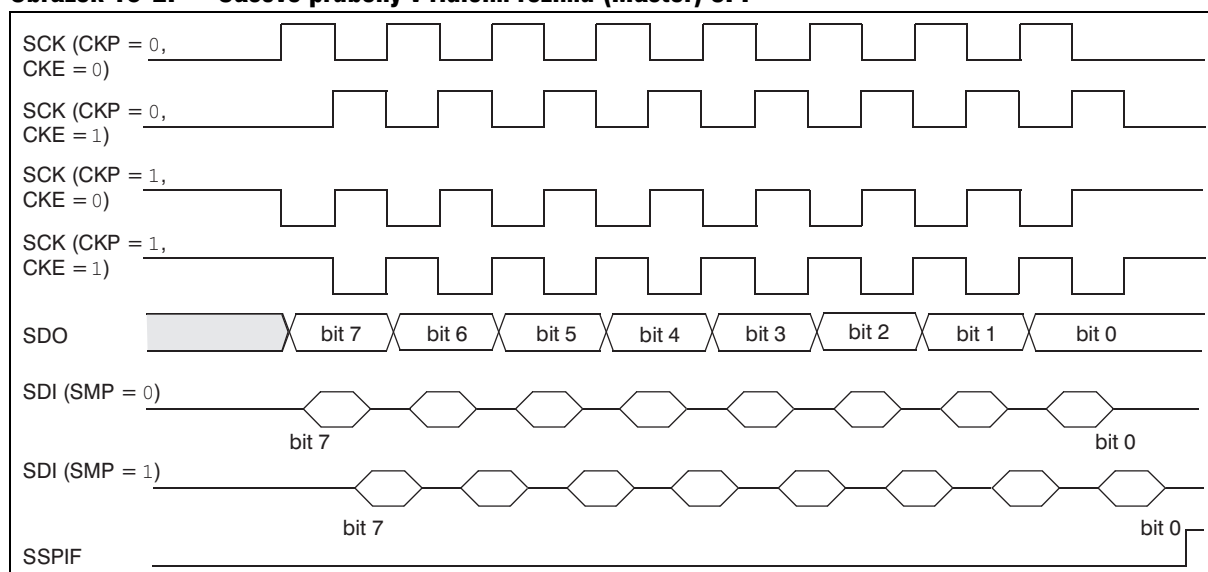
R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	D/A	P (1)	S (1)	R/W	UA	BF
bit 7							bit 0

- bit 7 **WCOL**: detekce kolize na sběrnici
1 = zápis do registru SSPBUF v době, kdy ještě probíhá vysílání předchozího bytu (nulování bitu musí být zajištěno programově).
0 = žádná kolize
- bit 6 **SSPOV**: příznak přetečení při příjmu
V režimu SPI:
1 = nový byte je přijat, zatímco registr SSPBUF ještě obsahuje předchozí data. V případě přetečení jsou data v registru SSPSR ztracena. Přetečení může nastat pouze v podřízeném režimu. Uživatel musí číst SSPBUF, dokonce i když pouze vysílá data (nulování bitu musí být zajištěno programově).
0 = nedošlo k přepsání dat
V režimu I²C:
1 = nový byte je přijat, zatímco registr SSPBUF ještě obsahuje předchozí data. Během vysílání nemá žádný význam (nulování bitu musí být zajištěno programově).
0 = nedošlo k přepsání dat
- Bit 5 **SSPEN**: povolení synchronního sériového portu (1)
V režimu SPI:
1 = povolení sériového synchronního portu a konfigurace vývodů SCK, SDO a SDI jako vývodů sériového portu.
0 = zakázání sériového portu a konfigurace příslušných vývodů jako univerzální vstupy/výstupy
V režimu I²C:
1 = povolení sériového synchronního portu a konfigurace vývodů SDA a SCL jako vývodů sériového portu.
0 = zakázání sériového portu a konfigurace příslušných vývodů jako univerzální vstupy/výstupy
Poznámka 1: V obou režimech musí být v případě povolení sériového portu všechny vývody správně konfigurovány jako vstupy nebo výstupy.
- Bit 4 **CKP**: výběr polaritý hodinového signálu
V režimu SPI:
1 = vysílání se sestupnou hranou, příjem s náběžnou a klidový stav je vysoká úroveň
0 = vysílání s náběžnou hranou, příjem se sestupnou a klidový stav je nízká úroveň
V režimu I²C:
řízení signálu SCK
1 = hodinový signál je povolen
0 = hodinový signál je přidržen v log. 0.
- bit 3-0 **SSPM3:SSPM0**: výběr režimu SSP
0000 = Řídicí režim (Master) SPI, hodinový signál je OSC/4
0001 = Řídicí režim (Master) SPI, hodinový signál je OSC/16
0010 = Řídicí režim (Master) SPI, hodinový signál je OSC/64
0011 = Řídicí režim (Master) SPI, hodinový signál je výstup TMR2/2
0100 = Řízený režim (Slave) SPI, hodiny na vývodu SCK, ovládání na SS povoleno
0101 = Řízený režim (Slave) SPI, hodiny na vývodu SCK, ovládání na SS zakázáno, SS je vstupně výstupní vývod
0110 = Řízený režim (Slave) I²C, 7. bitová adresa
0111 = Řízený režim (Slave) I²C, 10. bitová adresa
1011 = Řídicí režim (Master) I²C pomocí Firmware
1110 = Řízený režim (Slave) I²C, 7. bitová adresa s povolením přerušením od Start a Stop bitu
1111 = Řízený režim (Slave) I²C, 10. bitová adresa s povolením přerušením od Start a Stop bitu
1000, 1001, 1010, 1100, 1101 = rezervováno

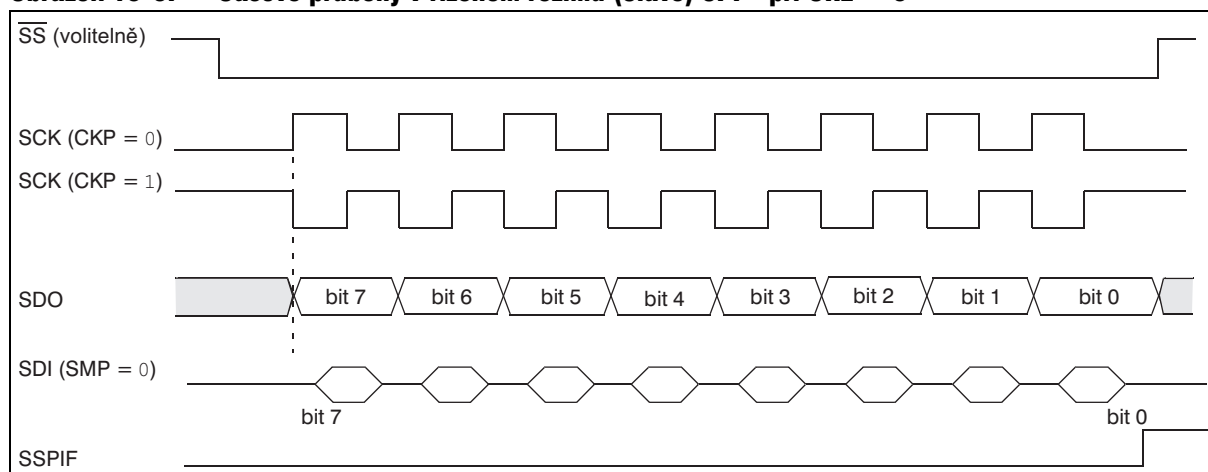
Význam:

R = bit pro čtení	W = bit pro zápis	U = nepoužitý bit, čteno jako '0'
-n = hodnota po resetu POR	'1' = bit je nastaven	'0' = bit je v nule
		x = hodnota bitu je neznámá

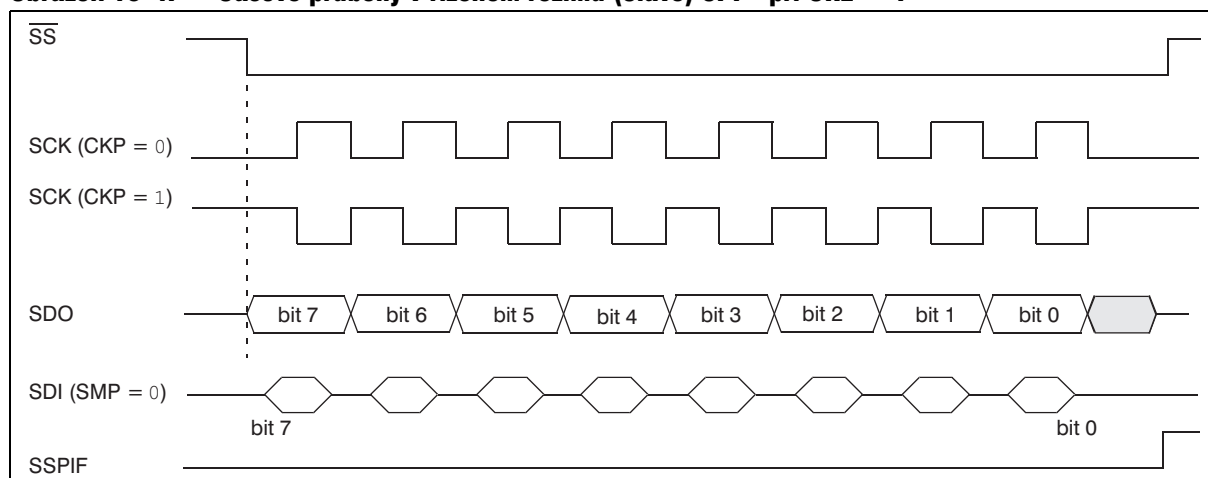
Obrázek 10-2: Časové průběhy v řídicím režimu (Master) SPI™



Obrázek 10-3: Časové průběhy v řízeném režimu (Slave) SPI™ při CKE = 0



Obrázek 10-4: Časové průběhy v řízeném režimu (Slave) SPI™ při CKE = 1



10.3 Režim I²C

Modul SSP má v režimu I²C realizované všechny podřízené (slave) funkce a poskytuje přerušení na Start a Stop bity pro určení volné sběrnice (multimaster funkce). Tento modul podporuje kromě standardní specifikaci také 7 a 10 bitové adresování.

Pro komunikaci jsou používány dva vývody. Vývod RB4/SCK/SCL je určen pro přenos hodinového signálu. Vývod SDI/SDA je určen pro přenos dat. Uživatel musí zajistit správnou konfiguraci těchto vývodů pomocí příslušných bitů registru TRISB.

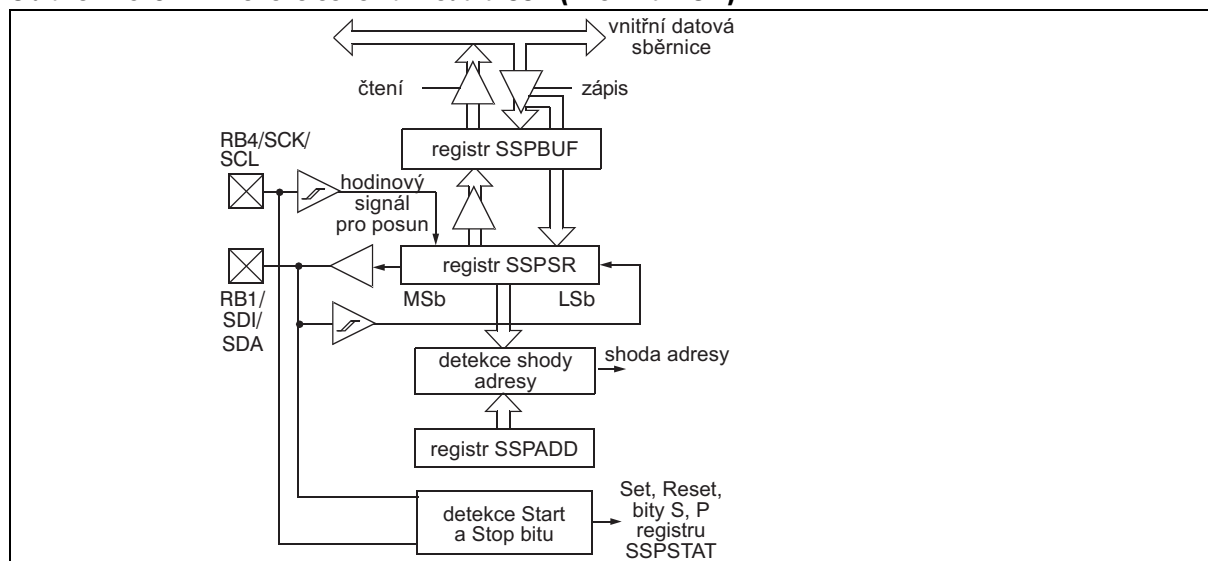
Pro zajištění správné činnosti modulu I²C je nutné všechny bity registru TRISB, odpovídající použití v rámci I²C nastavit na hodnotu log.1. Pokud je jakýkoli bit registru TRISB během činnosti modulu I²C modifikován pomocí instrukce typu čtení-modifikace-zápis (BSF, BCF), může dojít i ke změně nastavení bitů odpovídajících vývodům pro I²C a komunikace se může zhroutit. Proto během činnosti modulu I²C zásadně neměňte obsah registru TRISB tímto způsobem. Pokud je zcela nezbytné tak učinit, použijte následující programovou rutinu:

Příklad 10-1:

MOVWF	TRISC, W	; příklad pro 18. vývodový mikropočítač
		; jako je PIC16F818/819
IORLW	0x18	; zajištění bitů b4 a b3 ve stavu "11"
ANDLW	B'11111001'	; konfigurace bitů b2 a b1 jako výstupů
		; bez ovlivnění dalších bitů
		; uživatel může použít vlastní logiku,
		; např. IORLW, XORLW nebo ANDLW
MOVWF	TRISC	

Činnost celého modulu SSP je povolena nastavením bitu SSPEN v registru SSPCON.

Obrázek 10-5: Blokové schéma modulu SSP (v režimu I²C™)



Modul synchronní komunikace SSP má pět registrů pro obsluhu I²C. Jsou to:

- řídicí registr SSP SSPCON
- stavový registr SSP SSPSTAT
- sériová přijímací/vysílací vyrovnávací paměť – registr SSPBUF
- posuvný registr SSP SSPSR – uživatelsky nepřístupný
- adresový registr SSP SSPADD

Registr SSPCON umožňuje řídit operace při komunikaci pomocí I²C. Pomocí čtyř řídicích bitů SSPCON3:SSPCON0 tohoto registru lze zvolit jeden z následujících režimů I²C:

- Řízený režim (Slave) I²C s 7-bitovou adresou
- Řízený režim (Slave) I²C s 10-bitovou adresou
- Řízený režim (Slave) I²C s 7-bitovou adresou s povolením přerušení Start a Stop bitem povolující použití Master režimu ovládaného pomocí Firmware.
- Řízený režim (Slave) I²C s 10-bitovou adresou s povolením přerušení Start a Stop bitem povolující použití Master režimu ovládaného pomocí Firmware.
- Řídicí režim (Master) I²C ovládaný pomocí Firmware s povolením přerušení Start a Stop bitem. Slave je potlačen.

Při výběru jakéhokoli z uvedených režimů je nutné uvést vývody SCL a SDA do stavu, umožňujícího zapojení s otevřeným kolektorem. Odpovídající nastavení v registru TRISB je vstupní stav. Potřebné Pull-up rezistory pro zajištění stavu log.1 na lince je nutné zajistit externě.

Posuvný registr SSPSR slouží k přijímání nebo odesílání jednotlivých bitů komunikačních dat. SSPBUF je vyrovnávací paměť, do které nebo ze které jsou data přenášena do posuvného registru SSPSR.

V registru SSPADD je uložena v režimu I²C slave adresa jednočipového mikropočítače. Pokud je nastaven režim I²C master, pak dolních sedm bitů tohoto registru slouží ke konfiguraci komunikační rychlosti.

Během přijímání SSPSR a SSPBUF dva vyrovnávací paměťové bajty. Po dokončení přenosu dat je byte z SSPSR přenesen do SSPBUF a nastaven příznak přerušení SSPIF.

Při odesílání dat nemá komunikace dvojitou vyrovnávací paměť. Zápisem do SSPBUF jsou uložena data jak do SSPBUF tak do SSPSR.

Další informace jsou uvedeny v dokumentaci DS33023 s názvem "PICmicro[®] Mid-Range MCU Family Reference Manual"

10.3.1 Podřízený režim (Slave)

V podřízeném režimu (Slave) musí být vývody SDA a SCL konfigurovány jako vstupní (TRISC<4:3> = 11). Modul SSP bude přepisovat konfiguraci vývodu SDA na výstupní v případě nutnosti (vysílání podřízeného obvodu).

Pokud je zaznamenána shoda adresy nebo jsou přijata data (po předchozí schodě adres), modul automaticky vytvoří potvrzovací signál ($\overline{ACK}$) a uloží přijatá data z registru SSPSR do vyrovnávacího registru SSPBUF.

Jakákoliv kombinace následujících podmínek způsobí, že modul SSP nevytvoří potvrzovací signál $\overline{ACK}$:

- a) příznak zaplnění vyrovnávací paměti, bit BF (SSPSTAT<0>) byl nastaven před přijetím přenosu dat.
- b) příznak přetečení, SSPOV (SSPCON<6>) byl nastaven před přijetím přesunu dat.

V těchto případech nejsou data přenesena z registru SSPSR do SSPBUF, ale je nastaven příznak přerušení SSPIF (PIR1<3>). V tabulce 10-2 je situace znázorněna.

Pro správnou komunikaci musí mít vstupní hodinový signál odpovídající časové parametry. Doba trvání signálu ve vysoké a nízké úrovni podle specifikace I²C, také požadovaná od modulu SSP, jsou definovány jako časové parametry #100 a #101.

10.3.1.1 Adresování

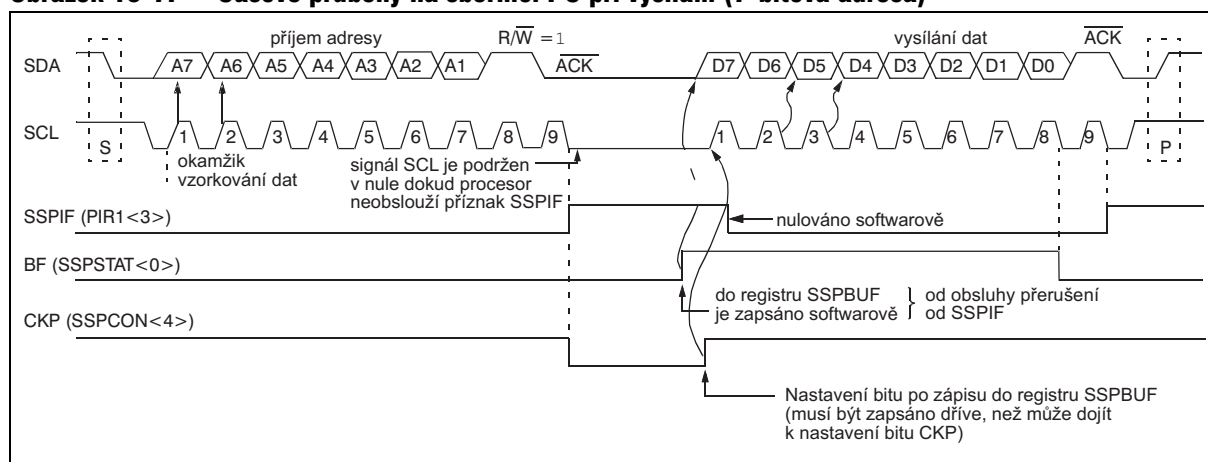
Jakmile je modul SSP povolen, čeká na příchod Startovací podmínky. Za touto kombinací signálů SDA a SCL jsou přijímány jednotlivé bity posouvané do registru SSPSR. Bity sériových dat jsou vzorkovány s náběžnou hranou hodinového signálu SCL. Hodnota v registru SSPSR<7:1> je porovnána s adresou (registr SSPADD) během sestupné hrany osmého taktu hodinového signálu (SCL). Při shodě adresy a nulových příznacích BF a SSPOV nastanou následující události:

- a) Hodnota registru SSPSR je uložena vyrovnávacího registru SSPBUF
- b) Příznak zaplnění vyrovnávacího registru SSPBUF je nastaven
- c) Vytvoří se potvrzovací signál $\overline{ACK}$
- d) Příznak přerušení modulu SSP SSPIF (PIR<3>) je nastaven se sestupnou hranou devátého pulsu hodinového signálu SCL (dojde k vyvolání přerušení pouze pokud je povoleno)

V případě 10-bitové adresového režimu musí být přijaty dva adresové bajty do podřízeného zařízení. Pět nejvýznamnějších bitů (MSb) prvního adresového bajtu stanoví, zda se jedná o rozšířenou adresu. Ovládací bit R/ $\overline{W}$ (SSPSTAT<2>) musí být nulován, aby podřízený obvod přijal druhou část adresy. První bajt 10-bitové adresy má tvar '11110 A9 A8 0', kde A9 a A8 jsou dva nejvýznamnější bity adresy. Posloupnost událostí při 10-bitové adresaci je následující (kroky 7-9 nastanou při odesílání dat z podřízeného obvodu):

1. Příjem prvního (vyššího) bytu adresy (příznaky SSPIF, BF a UA (Update Adress (SSPSTAT<1>)) jsou nastaveny)
2. Aktualizace registru SSPADD druhou (dolní) částí adresy (nuluje příznak UA a je uvolněn hodinový signál)
3. Čtení vyrovnávacího registru SSPBUF (nuluje příznak BF) a nulování příznaku SSPIF
4. Příjem druhého (dolního) bajtu (příznaky SSPIF, BF a UA jsou nastaveny)
5. Aktualizace registru SSPADD první (horní) částí adresy (pokud dojde ke shodě adresy a je uvolněn hodinový signál bude nulován příznak UA)
6. Čtení vyrovnávacího registru SSPBUF (nuluje příznak BF) a nulování příznaku SSPIF
7. Příjem opakované Startovací podmínky
8. Příjem prvního (vyššího) bajtu adresy (příznaky SSPIF a BF jsou nastaveny)
9. Čtení vyrovnávacího registru SSPBUF (nuluje příznak BF) a nulování příznaku SSPIF

Obrazek 10-7: Časové průběhy na sběrnici I²C při vysílání (7-bitová adresa)



10.3.2 Režim Master

Režim Master je umožněn pomocí firmware, který používá rutiny obsluhy přerušení od startovací a ukončovací podmínky. Bity start (S) a stop (P) jsou nulovány po resetu nebo v případě, že je modul SSP zakázán. Oba dva tyto bity jsou závislé na generování startovací nebo ukončovací podmínky. K ovládání sběrnice I²C může dojít v případě, že je bit P nastaven, nebo je sběrnice uvolněná nebo jsou oba bity S i P současně v nule.

V režimu Master jsou vodiče SCL a SDA ovládány pomocí firmware, který ovládá nastavení příslušných bitů registru TRISB. Výstupní úroveň je vždy nízká, nezávisle na obsahu registru PORTB. Stav log.1 je vyslán na sběrnici tak, že je příslušný bit v registru TRISB nastaven na log.1 a tím je zajištěn režim vstupu na vývodu SDA mikropočítače. Pro přenos log.0 je nutné přefigurovat vývod do režimu výstupu. Stejná filozofie je použita při ovládání vodiče SCL. V obou případech je nutné zajistit externě Pull-up rezistory.

Nastavení příznaku SSPIF může způsobit jedna z následujících událostí (pokud je přerušení od SSP povoleno)

- startovací podmínka
- ukončovací podmínka
- dokončení příjmu nebo vyslání bajtu

Režim Master může být realizován buď s potlačením režimu Slave, což odpovídá nastavení bitů SSPM3:SSPM0 do stavu "1011" nebo s aktivním režimem Slave. Pokud je současně aktivní režim Master i Slave, rutina obsluhy přerušení musí správně rozeznávat druh přerušení.

Podrobněji je pojednáno v Aplikační příručce v poznámce AN544 s názvem: "Software Implementation of I²C™ Bus Master"

10.3.3 Režim Multi-Master

V režimu Multi-Master je povolení přerušení od startovací nebo ukončovací podmínky závislé na tom, zda je sběrnice volná. Bity start (S) a stop (P) jsou nulovány po resetu nebo v případě, že je modul SSP zakázán. Oba dva tyto bity jsou závislé na generování startovací nebo ukončovací podmínky. K ovládání sběrnice I²C může dojít v případě, že je bit P nastaven, nebo je sběrnice uvolněná nebo jsou oba bity S i P současně v nule. V případě, že je sběrnice používána jiným zařízením, povolení přerušení od SSP způsobí přerušení jen při detekci ukončovací podmínky.

V režimu Multi-Master musí být monitorován vodič SDA pro zjištění očekávaného stavu tohoto signálu. Toto je nutné provádět jen pokud je očekávána vysoká úroveň signálu. Pokud je očekávána vysoká úroveň signálu a nastane nízká úroveň, zařízení musí uvolnit sběrnici a to oba vodiče SDA i SCL. (nastavením příslušných bitů registru TRISB). Jsou možné dva případy, kdy tento způsob řízení sběrnice může způsobit kolizi dat:

- přenos adresy
- přenos dat

Pokud je zařízení Slave povoleno, pokračuje v přijímání. Pokud dojde ke kolizi v době přenosu adresy, může dojít k pokračujícímu příjmu. Pokud je adresa přijmuta, dojde ke generování potvrzení ACK. Pokud dojde ke kolizi při přenosu dat, je nutné data poslat opakovaně později.

Podrobněji je pojednáno v Aplikační příručce v poznámce AN578 s názvem: "Use of the SSP Module in the I²C Multi-Master Environment"

PIC16F87/88

Tabulka 10-3: Registry, které mají vztah k režimu I²C

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druhích resetu
0Bh,8Bh,10Bh,18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IE	INT0IF	RBF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF (1)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
8Ch	PIE1	—	ADIE (1)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
13h	SSPBUF	Vyrovnávací sériová paměť pro příjem a vysílání								xxxx xxxx	uuuu uuuu
93h	SSPAD	Adresový registr v režimu I ² C								0000 0000	0000 0000
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
94h	SSPSTAT	SMP (2)	CKE (2)	D/Ā	P	S	R/W	UA	BF	0000 0000	0000 0000
86h	TRISB	Konfigurační registr brány PortB								1111 1111	1111 1111

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s režimem I²C modulu SSP.

Poznámka 1: Tento bit je použit jen u PIC16F88, u mikropočítače PIC16F87 je čten jako "0"

2: Tento bit ponechte v režimu I²C v nule

11 MODUL AUSART

Adresovatelný univerzální synchronní/asynchronní přijímač/vysílač AUSART je jedním ze dvou sériových komunikačních portů mikropočítače. AUSART bývá rovněž někdy označován zkratkou SCI (sériový komunikační interface). AUSART může být konfigurován jako plně obousměrný (duplexní) asynchronní komunikační port připojitelný například k portu COM osobního počítače. Druhá možnost konfigurace modulu AUSART je poloduplexní synchronní port pro připojení k periferním zařízením jako jsou externí AD převodníky, paměti nebo další integrované obvody.

Modul AUSART může být konfigurován v následujících režimech:

- Asynchronní (plně duplexní)
- Synchronní – Master (poloduplexní)
- Synchronní – Slave (poloduplexní)

Bity SPEN v registru RCSTA<7> a bity TRISB<5:2> musí být nastaveny podle konfigurace vývodů RB5/ $\overline{SS}$ /TX/CK a RB2/SDO/RX/DT pro použití modulem AUSART.

Modul AUSART může být rovněž konfigurován pro realizaci adresovatelného multi-procesorového protokolu pro síťové propojení až 256 účastníků.

Registr 11-1: Stavový a řídicí registr vysílání TXSTA (adresa 98h)

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R-1	R/W-0
CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D
bit 7							bit 0

- bit 7 **CSRC:** Volba zdroje hodinového signálu
Asynchronní režim:
 nemá význam
Synchronní režim:
 1 = režim Master (hodinový signál je generován vnitřně pomocí BRG)
 0 = režim Slave (hodinový signál je zajištěn vnějším zdrojem)
- bit 6 **TX9:** Volba druhu vysílání
 1 = je vysíláno 9 bitů
 0 = je vysíláno 8 bitů
- bit 5 **TXEN:** Povolení vysílání
 1 = vysílání povoleno
 0 = vysílání zakázáno
Poznámka: SREN/CREN přepisuje bit TXEN v synchronním režimu
- bit 4 **SYNC:** režim modulu AUSART
 1 = synchronní režim
 0 = asynchronní režim
- bit 3 **nepoužit:** při čtení 0
- bit 2 **BRGH:** Volba tabulky rychlosti generátoru hodinového signálu
Asynchronní režim:
 1 = vyšší rychlosti
 0 = nižší rychlosti
Synchronní režim:
 nemá význam
- bit 1 **TRMT:** příznak vysílacího posuvného registru
 1 = TSR je prázdný
 0 = TSR není prázdný
- bit 0 **TX9D:** devátý bit pro vysílání (může mít význam paritního bitu)

Význam:

R = bit pro čtení	W = bit pro zápis	U = nepoužitý bit, čteno jako '0'
–n = hodnota po resetu POR	'1' = bit je nastaven	'0' = bit je v nule x = hodnota bitu je neznámá

Registru 11-2: Stavový a řídicí registr příjmu RCSTA (adresa 18h)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0	R-0	R-x
SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
bit 7							bit 0

- bit 7 **SPEN:** Povolení modulu AUSART
1 = sériový port je zapnut (vývody RB2 a RB5 jsou ovládány výhradně pomocí AUSART)
0 = sériový port je vypnut
- bit 6 **RX9:** Volba druhu příjmu
1 = je přijímáno 9 bitů
0 = je přijímáno 8 bitů
- bit 5 **SREN:** Povolení jediného příjmu
Asynchronní režim:
nemá význam
Synchronní režim – Master:
1 = povoleno
0 = zakázáno
tento bit je nulován po dokončení celého příjmu
Synchronní režim – Slave:
nemá význam
- bit 4 **CREN:** Povolení průběžného příjmu
Asynchronní režim:
1 = povoleno
0 = zakázáno
Synchronní režim:
1 = povoleno dokud je nulován bit CREN (CREN může být nastaven pomocí SREN)
0 = zakázáno
- bit 3 **ADDEN:** povolení detekce adresového bitu
jen pro asynchronní režim s nastavenou délkou 9 bitů:
1 = povolení detekce adresy,
0 = zákaz detekce adresy, devátý bit může být použit jako paritní
- bit 2 **FERR:** příznak chyby rámce
1 = nastala chyba rámce (může být nulován čtením z RCREG a přijetím dalšího platného data)
0 = nenastala chyba rámce
- bit 1 **OERR:** příznak přetečení vstupního bufferu
1 = nastalo přetečení bufferu (je nulován nulováním bitu CREN)
0 = nenastalo přetečení bufferu
- bit 0 **RX9D:** devátý přijímaný bit (může mít význam paritního bitu ale detekce parity musí být zajištěna uživatelem)

Význam:

R = bit pro čtení	W = bit pro zápis	U = nepoužitý bit, čteno jako '0'
-n = hodnota po resetu POR	'1' = bit je nastaven	'0' = bit je v nule
		x = hodnota bitu je neznámá

11.1 Generátor hodinového signálu pro AUSART (BRG)

Generátor hodinového signálu se používá pro oba režimy modulu AUSART, pro asynchronní i synchronní. Od kmitočtu generátoru je odvozena přenosová rychlost. Generátor je tvořen 8 bitovým čítačem. Perioda tohoto čítače je určena obsahem řídicího registru SPBRG. V asynchronním režimu je přenosová rychlost dále určena bitem BRGH v registru TXSTA<2>. V synchronním režimu je tento bit ignorován. V tabulce 11-1 je uveden vzorec pro výpočet přenosové rychlosti pro různé nastavení registru SPBRG a bitu BRGH.

Přenosová rychlost závisí nejen na obsahu registru SPBRG případně bitu BRGH, ale i na taktovací frekvenci mikropočítače. Po vypočtení potřebné hodnoty SPBRG uložíme do tohoto registru nejbližší celočíselnou hodnotu v rozsahu 0 – 255. Tím dosáhneme požadované přenosové rychlosti s určitou chybou. Pokud je chyba příliš velká, je možné vypočítat obsah registru SPBRG pro jiný rozsah hodnot odpovídající jinému nastavení bitu BRGH.

Obvykle je výhodnější používat horní skupinu přenosových rychlostí. Vlivem většího poměru děličky vycházejí normované přenosové rychlosti zpravidla s menší chybou.

Při zápisu do registru SPBRG dojde k vynulování pracovního registru čítače a okamžité změně přenosové rychlosti. Ke změně nedojde tedy teprve po přetečení pracovního čítače rychlostí odpovídající dřívějšímu nastavení.

11.1.1 Modul AUSART a vnitřní oscilátor INTRC

Mikropočítač je vybaven vnitřním taktovacím oscilátorem, který umožňuje ušetřit obvykle používané externí součástky pro realizaci vnějšího taktovacího oscilátoru. Pokud je zdrojem taktovacího signálu vnitřní oscilátor INTRC je možné rovněž používat modul AUSART. V tabulce 11-1 jsou uvedeny některé hodnoty frekvence INTRC vhodné pro použití jako generátoru přenosové rychlosti.

11.1.2 Práce v režimu se sníženou spotřebou.

Přenosová rychlost je vždy odvozena z taktovacího oscilátoru mikropočítače. V režimu se sníženou spotřebou může být zdrojem taktovacího signálu sekundární oscilátor s odlišnou taktovací frekvencí. V takovém případě je nutné změnit nastavení v registru SPBRG. V režimu Sleep není taktovací oscilátor v činnosti vůbec.

11.1.3 Vzorkování

Pro zjištění úrovně na vývodu RB2/SDO/RX/DT při příjmu je toto napětí vzorkováno celkem třikrát.

Tabulka 11-1: Vzorec pro výpočet přenosové rychlosti

SYNC	Skupina nižších rychlostí	Skupina vyšších rychlostí
0	Přenosová rychlost (asynchronní) = $F_{osc}/(64(X + 1))$	Přenosová rychlost = $F_{osc}/(16(X + 1))$
1	Přenosová rychlost (synchronní) = $F_{osc}/(4(X + 1))$	N/A

Význam: X = obsah registru SPBRG v rozsahu 0 – 255

Tabulka 11-2: Registry, které mají vztah ke generátoru hodinového signálu pro AUSART.

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druzích resetu
98h	TXSTA	CSCR	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 - 010	0000 - 010
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
99h	SPBRG	Konfigurační registr pro stanovení přenosové rychlosti								0000 0000	0000 0000

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s AUSART.

Tabulka 11-3: Přenosové rychlosti pro asynchronní režim (skupina nižších rychlostí BRGH=0)

přenosová rychlost (kBd)	Fosc = 20 MHz			Fosc = 16 MHz			Fosc = 10 MHz		
	KBAUD	% chyba	hodnota SPBRG (dakadicky)	KBAUD	% chyba	hodnota SPBRG (dakadicky)	KBAUD	% chyba	hodnota SPBRG (dakadicky)
0,3	—	—	—	—	—	—	—	—	—
1,2	1,221	+1,75	255	1,202	+0,17	207	1,202	+0,17	129
2,4	2,404	+0,17	129	2,404	+0,17	103	2,404	+0,17	64
9,6	9,766	+1,73	31	9,615	+0,16	25	9,766	+1,73	15
19,2	19,531	+1,72	15	19,231	+0,16	12	19,531	+1,72	7
28,8	31,250	+8,51	9	27,778	-3,55	8	31,250	+8,51	4
33,6	34,722	+3,34	8	35,714	+6,29	6	31,250	-6,99	4
57,6	62,500	+8,51	4	62,500	+8,51	3	52,083	-9,58	2
nejvyšší rychlost	1,221	—	255	0,977	—	255	0,610	—	255
nejnižší rychlost	312,500	—	0	250,000	—	0	156,250	—	0

přenosová rychlost (kBd)	Fosc = 4 MHz			Fosc = 3,6864 MHz		
	KBAUD	% chyba	hodnota SPBRG (dakadicky)	KBAUD	% chyba	hodnota SPBRG (dakadicky)
0,3	0,300	0	207	0,3	0	191
1,2	1,202	+0,17	51	1,2	0	47
2,4	2,404	+0,17	25	2,4	0	23
9,6	8,929	+6,99	6	9,6	0	5
19,2	20,833	+8,51	2	19,2	0	2
28,8	31,250	+8,51	1	28,8	0	1
33,6	—	—	—	—	—	—
57,6	62,500	+8,51	0	57,6	0	0
nejvyšší rychlost	0,244	—	255	0,225	—	255
nejnižší rychlost	62,500	—	0	57,6	—	0

Tabulka 11-4: Přenosové rychlosti pro asynchronní režim (skupina vyšších rychlostí BRGH=1)

přenosová rychlost (kBd)	Fosc = 20 MHz			Fosc = 16 MHz			Fosc = 10 MHz		
	KBAUD	% chyba	hodnota SPBRG (dakadicky)	KBAUD	% chyba	hodnota SPBRG (dakadicky)	KBAUD	% chyba	hodnota SPBRG (dakadicky)
0,3	—	—	—	—	—	—	—	—	—
1,2	—	—	—	—	—	—	—	—	—
2,4	—	—	—	—	—	—	2,441	+1,71	255
9,6	9,615	+0,16	129	9,615	+0,16	103	9,615	+0,16	64
19,2	19,231	+0,16	64	19,231	+0,16	51	19,531	+1,72	31
28,8	29,070	+0,94	42	29,412	+2,13	33	28,409	-1,36	21
33,6	33,784	+0,55	36	33,333	-0,79	29	32,895	-2,10	18
57,6	59,524	+3,34	20	58,824	+2,31	16	56,818	-1,36	10
nejvyšší rychlost	4,883	—	255	3,906	—	255	2,441	—	255
nejnižší rychlost	1250,000	—	0	1000,000	—	0	625,000	—	0

přenosová rychlost (kBd)	Fosc = 4 MHz			Fosc = 3,6864 MHz		
	KBAUD	% chyba	hodnota SPBRG (dakadicky)	KBAUD	% chyba	hodnota SPBRG (dakadicky)
0,3	—	—	—	—	—	—
1,2	1,202	+0,17	207	1,2	0	191
2,4	2,404	+0,17	103	2,4	0	95
9,6	9,615	+0,16	25	9,6	0	23
19,2	19,231	+0,16	12	19,2	0	11
28,8	27,798	-3,55	8	28,8	0	7
33,6	35,714	+6,29	6	32,9	-2,04	6
57,6	62,500	+8,51	3	57,6	0	3
nejvyšší rychlost	0,977	—	255	0,9	—	255
nejnižší rychlost	250,000	—	0	230,4	—	0

Tabulka 11-5: Přenosové rychlosti pro asynchronní režim s INTRC (skupina nižších rychlostí BRGH=0)

přenosová rychlost (kBd)	Fosc = 8 MHz			Fosc = 4 MHz			Fosc = 2 MHz			Fosc = 1 MHz		
			hodnota			hodnota			hodnota			hodnota
	KBAUD	% chyba	SPBRG (dakadicky)	KBAUD	% chyba	SPBRG (dakadicky)	KBAUD	% chyba	SPBRG (dakadicky)	KBAUD	% chyba	SPBRG (dakadicky)
0,3	N/A	—	—	0,300	0	207	0,300	0	103	0,300	0	51
1,2	1,202	+0,16	103	1,202	+0,16	51	1,202	+0,16	25	1,202	+0,16	12
2,4	2,404	+0,16	51	2,404	+0,16	25	2,404	+0,16	12	2,232	-6,99	6
9,6	9,615	+0,16	12	8,929	-6,99	6	10,417	+8,51	2	N/A	—	—
19,2	17,857	-6,99	6	20,833	+8,51	2	N/A	—	—	N/A	—	—
28,8	31,250	+8,51	3	31,250	+8,51	1	31,250	+8,51	0	N/A	—	—
33,6	41,667	+8,51	2	N/A	—	—	N/A	—	—	N/A	—	—
57,6	62,500	+8,51	1	62,500	8,51	0	N/A	—	—	N/A	—	—

Tabulka 11-6: Přenosové rychlosti pro asynchronní režim s INTRC (skupina vyšších rychlostí BRGH=1)

přenosová rychlost (kBd)	Fosc = 8 MHz			Fosc = 4 MHz			Fosc = 2 MHz			Fosc = 1 MHz		
			hodnota			hodnota			hodnota			hodnota
	KBAUD	% chyba	SPBRG (dakadicky)	KBAUD	% chyba	SPBRG (dakadicky)	KBAUD	% chyba	SPBRG (dakadicky)	KBAUD	% chyba	SPBRG (dakadicky)
0,3	N/A	—	—	N/A	—	—	N/A	—	—	0,300	0	207
1,2	N/A	—	—	1,202	+0,16	207	1,202	+0,16	103	1,202	+0,16	51
2,4	2,404	+0,16	207	2,404	+0,16	103	2,404	+0,16	51	2,404	+0,16	25
9,6	9,615	+0,16	51	9,615	+0,16	25	9,615	+0,16	12	8,929	-6,99	6
19,2	19,231	+0,16	25	19,231	+0,16	12	17,857	-6,99	6	20,833	+8,51	2
28,8	29,412	+2,12	16	27,778	-3,55	8	31,250	+8,51	3	31,250	+8,51	1
33,6	38,462	+0,16	12	35,714	-6,99	6	41,667	+8,51	2	N/A	—	—
57,6	55,556	-3,55	8	62,500	+8,51	3	62,500	+8,51	1	62,500	+8,51	0

11.2 Asynchronní režim modulu AUSART

V tomto režimu používá modul AUSART standardní protokol (1 start bit, 8 nebo 9 datových bitů a 1 stop bit). Nejpoužívanější datový formát je 8. bitový. Pomocí vlastního generátoru hodinového signálu odvozeného od taktu mikropočítače lze dosáhnout všechny standardní přenosové rychlosti. První je vysílán a přijímán nejméně významný bit (LSb). Vysílání a příjem může probíhat nezávisle na sobě, je však nutné dodržet stejnou přenosovou rychlost a formát komunikace. Přenosová rychlost je odvozena od taktovací frekvence. Pokud je zapotřebí přenášet paritní bit (devátý), je nutné jej softwarově vypočítat pro vysílání i příjem. Asynchronní přenos není možný během režimu Sleep.

Asynchronní režim je zvolen nulováním bitu SYNC v registru TXSTA<4>.

Modul AUSART v režimu asynchronního přenosu sestává z následujících podstatných částí:

- Generátor hodinového signálu pro AUSART
- Vzorkovací obvod
- Asynchronní vysílač
- Asynchronní přijímač

11.2.1 Asynchronní vysílač modulu AUSART

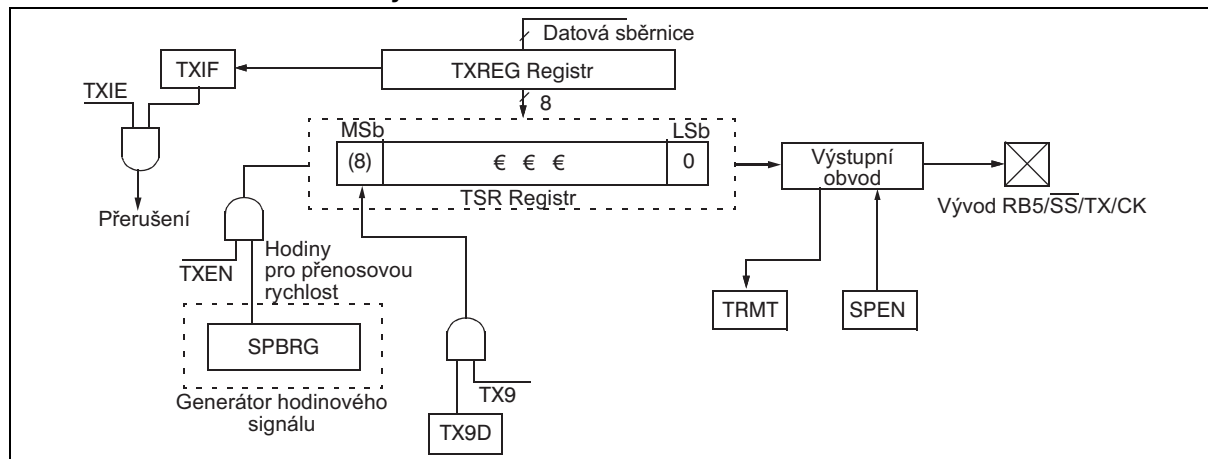
Blokové schéma vysílače AUSART je uvedeno na obrázku 11-1. Jádrem vysílače je tvořeno posuvným registrem TSR. Do tohoto registru jsou zapsána data z výstupního registru TXREG automaticky při zahájení vysílání. Do registru TXREG zapisuje data uživatel softwarově. Do posuvného registru TSR nemůže být zapsán další bajt, dokud nedojde k vyslání stop bitu předešlého bajtu. Bezprostředně po vyslání stop bitu je do posuvného registru zapsán další bajt z TXREG (pokud je tento bajt přítomen). Jakmile je vysílaný bajt zapsán do posuvného registru TSR, je nastaven příznak TXIF v registru PIR1<4> označující, že výstupní registr TXREG je již prázdný. Tato událost může vyvolat přerušení, pokud je toto povoleno bitem TXIE v registru PIE1<4>. K nastavení bitu TXIF dochází nezávisle na tom, zda je příslušné přerušení povoleno nebo zakázáno a není možné jej vynulovat softwarově. Je nulován automaticky jedině v případě zápisu nového bajtu do výstupního registru TXREG. Zatímco bit TXIF indikuje stav výstupního registru TXREG, stav posuvného registru TSR je indikován příznakem TRMT v registru TXSTA<1>. Tento příznak je určen pouze ke čtení a je nastaven vždy, pokud je posuvný registr TSR prázdný. Změnou bitu TRMT nelze vyvolat žádné přerušení, nicméně uživatel jej může testovat softwarově.

- Poznámka 1:** Posuvný registr TSR není uživatelsky přístupný na žádné adrese paměti RWM
2: Příznak TXIF je nastaven při povolení bitu TXEN a nulován při zápisu do registru TXREG.

Vysílání je povoleno nastavením bitu TXEN v registru TXSTA<5>. Vysílání nebude zahájeno, dokud nebude zapsán vysílaný bajt do registru TXREG. Časové průběhy vysílání jsou uvedeny na obrázku 11-2. Vysílání může být rovněž zahájeno tak, že nejprve dojde k zápisu do registru TXREG a teprve poté k povolení vysílání nastavením bitu TXEN. Obvykle je před zahájením vysílání posuvný registr TSR prázdný. Zápis do výstupního registru TXREG způsobí tedy okamžitý zápis do tohoto posuvného registru a uvolnění registru TXREG pro další zápis. Tato situace je znázorněna na obrázku 11-3. Nulování řídicího bitu TXEN během přenosu způsobí přerušení vysílání a vynulování posuvného registru s pochopitelnou ztrátou dat. Výsledkem zákazu vysílání je přechod vývodu RB5/SS/TX/CK do třetího stavu.

Pokud je zvolena 9-bitová komunikace nastavením bitu TX9 v registru TXSTA<6>, je nutné zapsat devátý bit před zahájením přenosu do bitu TX9D v registru TXSTA<0>. Tento devátý bit musí být zapsán dříve, než dojde k zápisu vysílaného bajtu do výstupního registru TXREG. Zápis do registru TXREG totiž může mít za následek okamžitý zápis do posuvného registru (pokud je tento prázdný) a devátý bit by mohl být vyslán špatně.

Obrázek 11-1: Blokové schéma vysílače AUSART

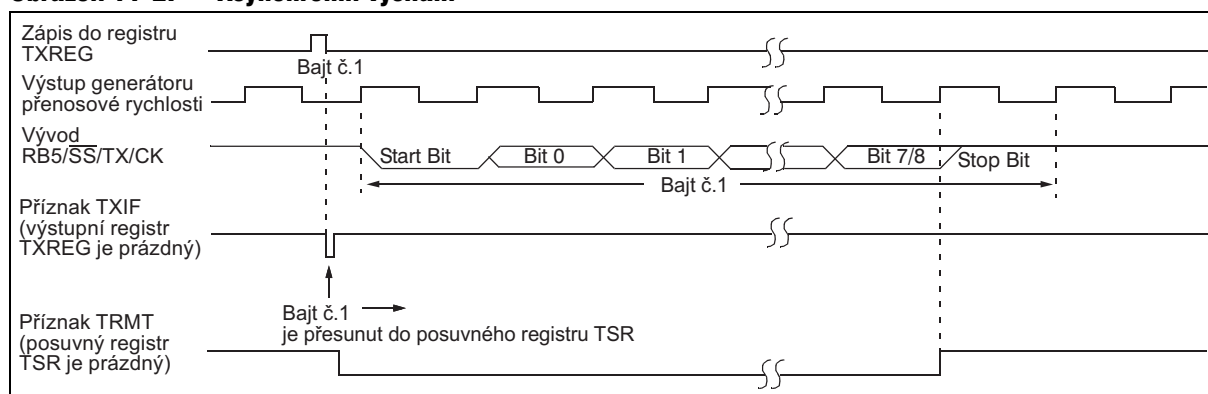


PIC16F87/88

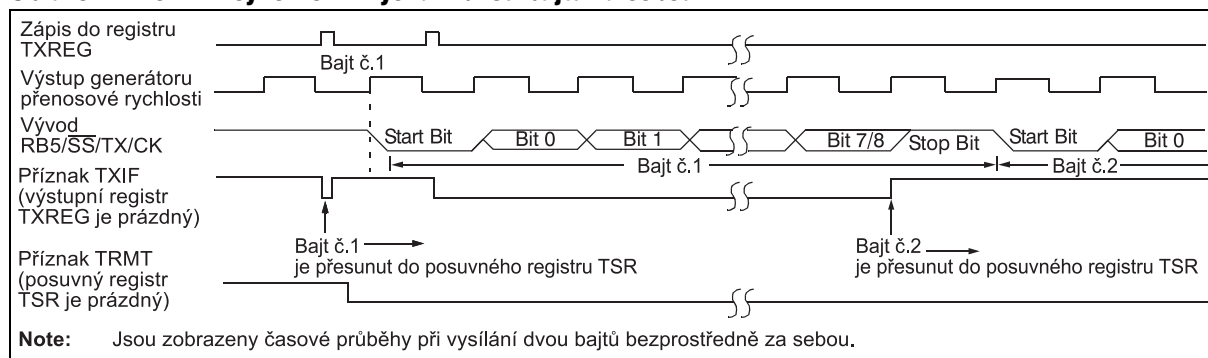
Při asynchronním vysílání je nutné dodržet následující kroky:

1. Nastavit správně obsah registru SPBRH a bit BRGH podle požadované přenosové rychlosti.
Podrobněji je pojednáno v **kapitole 11.1**.
2. Povolit asynchronní sériový přenos nulováním bitu SYNC a nastavením bitu SPEN.
3. Pokud je požadováno přerušení, povolit přerušení nastavením bitu TXIE.
4. Pokud je požadován 9.bitový přenos, nastavit bit TX9.
5. Povolit vysílání nastavením bitu TXEN což nastaví i bit TXIF.
6. Pokud je nastavena 9.bitová komunikace, zapsat devátý bit do bitu TX9D
7. Zapsat data do výstupního registru TXREG (tím dojde k faktickému zahájení vysílání)
8. Pokud používáte přerušení, ujistěte se, že jsou nastaveny bity GIE a PEIE v registru INTCON.

Obrázek 11-2: Asynchronní vysílání



Obrázek 11-3: Asynchronní vysílání dvou bajtů za sebou



Tabulka 11-7: Registry, které mají vztah k asynchronnímu vysílání modulem AUSART

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR	Hodnota po ostatních druzích resetu
0Bh,8Bh, 10Bh,18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IE	INT0IF	RBFIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF (1)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
19h	TXREG	Výstupní registr								0000 0000	0000 0000
8Ch	PIE1	—	ADIE (1)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Konfigurační registr pro stanovení přenosové rychlosti								0000 0000	0000 0000

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s AUSART.

Poznámka 1: Tento bit je použit jen u mikropočítače PIC16F88. U mikropočítače PIC16F87 je čten jako nula.

11.2.2 Asynchronní přijímač modulu AUSART

Blokové schéma přijímače AUSART je uvedeno na obrázku 11-4. Vstupní signál je snímán na vývodu RB2/SD0/RX/DT a dále veden do obvodu pro obnovení dat. Základem obvodu pro obnovení dat je rychlý posuvný registr, pracující na 16 krát vyšší frekvenci než je přenosová rychlost. Přijatý bajt je umístěn v přijímacím posuvném registru, pracujícím pochopitelně na frekvenci odpovídající přenosové rychlosti.

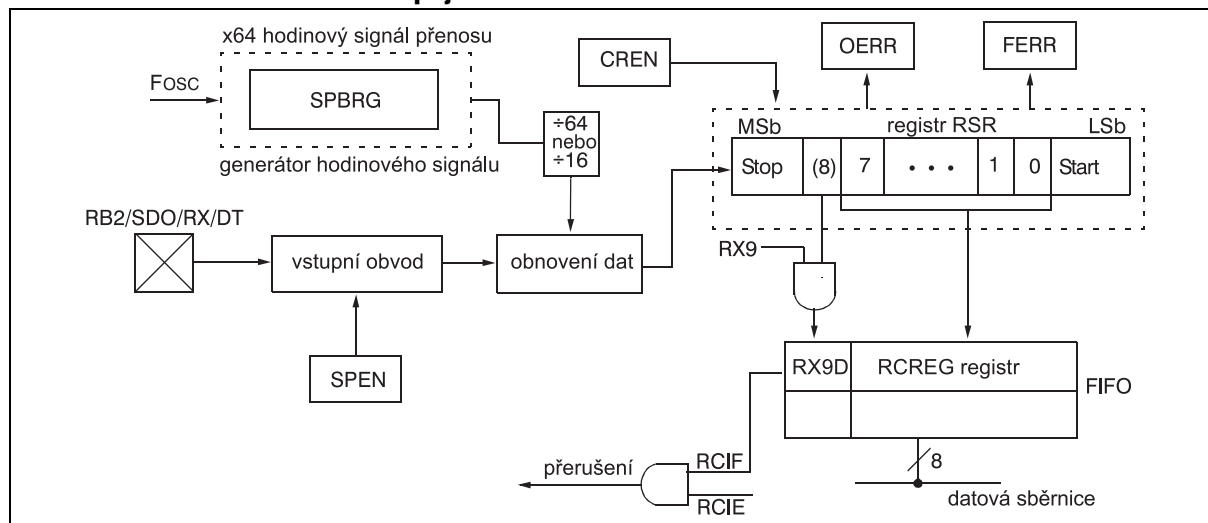
Jakmile je zvolen asynchronní režim, příjem dat je povolen nastavením bitu CREN v registru RCSTA<4>.

Jádro přijímače tvoří již zmíněný rychlý posuvný registr RSR. Po přijetí stop bitu je přijatý bajt přesunut z tohoto registru do přijímacího registru RCREG, pokud je tento registr volný. Pokud je přenos kompletní, je nastaven příznakový bit RCIF v registru PIR1<5>. Tato událost může vyvolat přerušení, pokud je povoleno nastavením bitu RCIE v registru PIE1<5>. Příznak CIF je určen pouze ke čtení, a je nulován hardwarově. Je nulován čtením přijímacího registru RCREG a to jen tehda, jestli je po čtení tento registr prázdný. Přijímací registr RCREG je totiž konstruován jako paměť FIFO o velikosti 2 bajty. V přijímacím registru RCREG tedy mohou být dva přijaté nepřečtené bajty, zatímco dochází ke čtení třetího bajtu. K přetečení vstupního bufferu dojde tedy teprve po přijetí Stop bitu třetího bajtu (pokud v té době není ještě přečtený RCREG se dvěma dříve přijatými bajty). Chyba přetečení je indikována nastavením příznaku OERR v registru RCSTA<1>. Bajt přijatý jako třetí je tím ztracen.

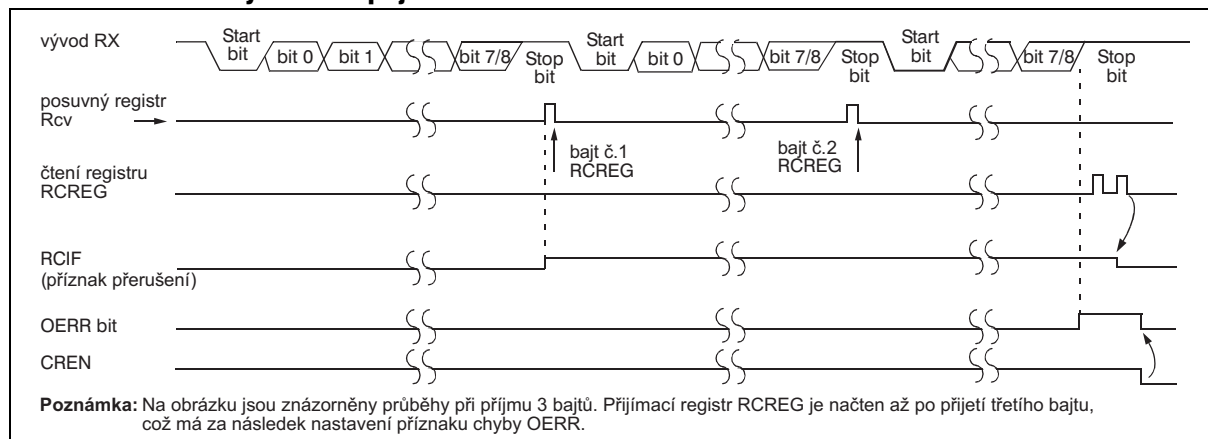
Pro získání přijatých dat je nutné číst registr RCREG. Pokud je tento registr čten dvakrát za sebou, jsou získány dva po sobě následující přijaté bajty (pokud jsou v paměti FIFO přítomny). Příznak chyby přetečení vstupního bufferu OERR je nulován softwarově. K jeho nulování je nutné nulovat a znovu nastavit bit CREN. Pokud je příznak OERR nastaven, je jakýkoli přenos z registru RSR do přijímacího registru RCREG zakázán. V takovém případě pochopitelně nejsou přijímána žádná další data.

Pokud je v při příjmu detekován stop bit roven 0, je nastaven příznak chyby rámce FERR v registru RCSTA<2>. Tento příznak je stejně jako případný devátý bit uložen v paměti stejným způsobem jako přijatý bajt, tedy ve struktuře FIFO. Čtení registru RCREG způsobí nastavení těchto dvou bitů do stavu, který odpovídá aktuálně načtenému bajtu. Tímto způsobem je zajištěno, že nemůže dojít ke ztrátě ani dřívějšího devátého bitu ani dřívějšího příznaku chyby rámce.

Obrázek 11-4: Blokové schéma přijímače AUSART



Obrázek 11-5: Asynchronní příjem



Poznámka: Na obrázku jsou znázorněny průběhy při příjmu 3 bajtů. Příjmací registr RCREG je načten až po přijetí třetího bajtu, což má za následek nastavení příznaku chyby OERR.

PIC16F87/88

Při asynchronním příjmu je nutné dodržet následující kroky:

1. Nastavit správně obsah registru SPBRH a bit BRGH podle požadované přenosové rychlosti.
Podrobněji je pojednáno v **kapitole 11.1**.
2. Povolit asynchronní sériový přenos nulováním bitu SYNC a nastavením bitu SPEN.
3. Pokud je požadováno přerušení, povolit přerušení nastavením bitu RCIE.
4. Pokud je požadován 9-bitový přenos, nastavit bit RX9.
5. Povolit příjem nastavením bitu CREN
6. Příznakový bit RCIF bude nastaven po příjmu 1. bajtu. Tato událost může způsobit přerušení, pokud je povoleno.
7. Přečíst obsah registru RCSTA a zjistit případné chyby při přenosu nebo určit případný devátý bit přenosu.
8. Přečíst přijatý bajt z registru RCREG.
9. Pokud byla během přenosu indikována chyba, nulovat bit CREN (a následně nastavit)
10. Při použití přerušení se ujistěte, že jsou správně nastaveny bity GIE a PEIE v registru INTCON.

Tabulka 11-8: Registry, které mají vztah k asynchronnímu příjmu modulem AUSART

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druhích resetu
0Bh, 8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IE	INT0IF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF (1)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
1Ah	RCREG	Příjmový registr								0000 0000	0000 0000
8Ch	PIE1	—	ADIE (1)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Konfigurační registr pro stanovení přenosové rychlosti								0000 0000	0000 0000

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s AUSART.

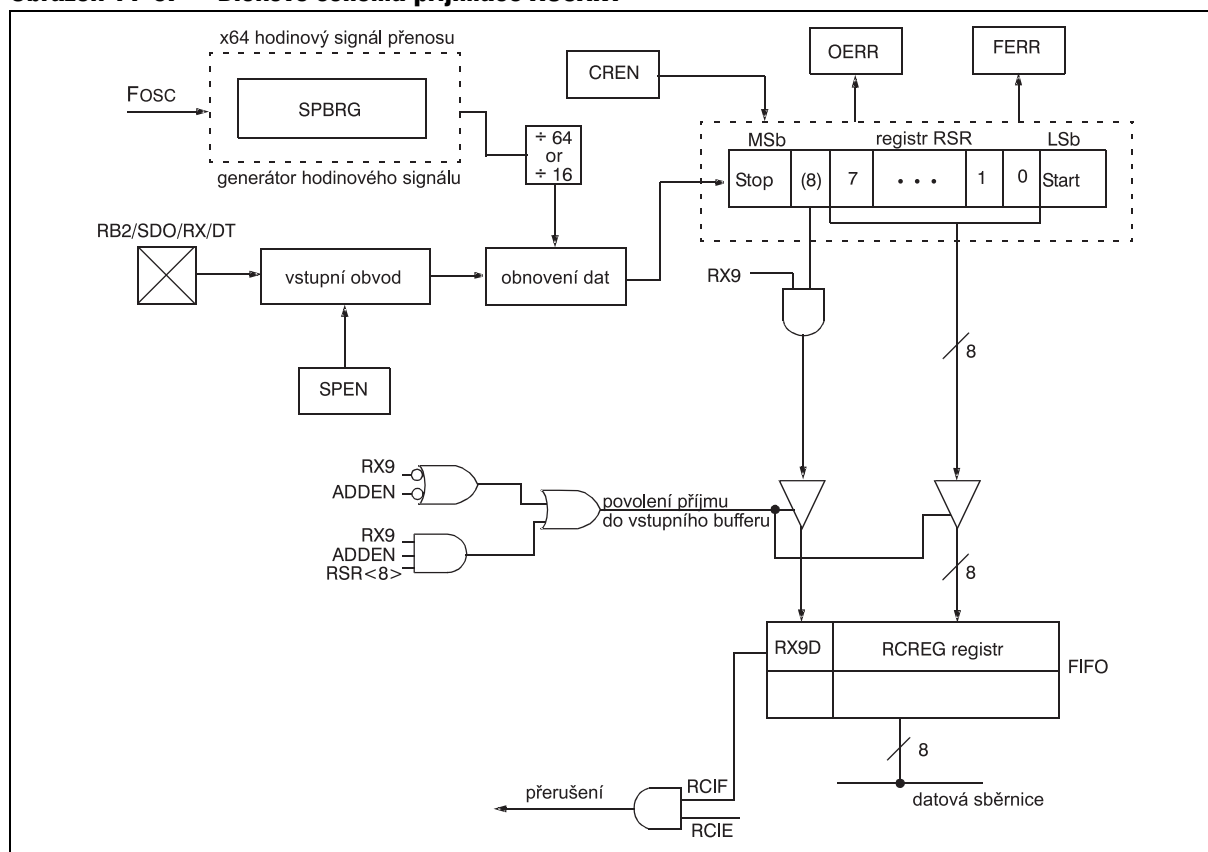
Poznámka 1: Tento bit je použit jen u mikropočítače PIC16F88. U mikropočítače PIC16F87 je čten jako nula.

10.2.3 Použití 9-bitového protokolu s detekcí adresy (síťový protokol)

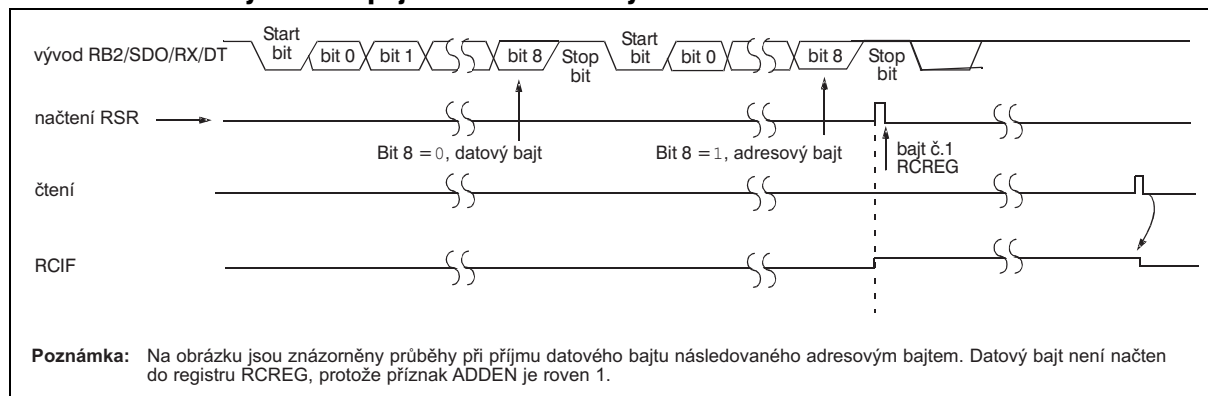
Při asynchronním příjmu s detekcí adresy je nutné provést následující kroky:

1. Nastavit správně obsah registru SPBRG a bit BRGH podle požadované přenosové rychlosti.
2. Povolit asynchronní sériový přenos nulováním bitu SYNC a nastavením bitu SPEN.
3. Pokud je požadováno přerušení, povolit přerušení nastavením bitu RCIE.
4. Nastavit bit RX9
5. Nastavit bit ADDEN pro povolení detekce adresy
6. Povolit příjem nastavením bitu CREN
7. Příznakový bit RCIF bude nastaven po příjmu 1. bajtu. Tato událost může způsobit přerušení, pokud je povoleno.
8. Přečíst obsah registru RCSTA a zjistit případné chyby při přenosu nebo určit případný devátý bit přenosu.
9. Přečíst přijatý bajt z registru RCREG a zjistit, zda je dané zařízení adresováno
10. Pokud byla během přenosu indikována chyba, nulovat bit CREN (a následně nastavit)
11. Pokud bylo zařízení adresováno (jeho adresa se shoduje s přijatou adresou), nulovat bit ADDEN.
Tím se umožní příjem dat do vstupního bufferu.

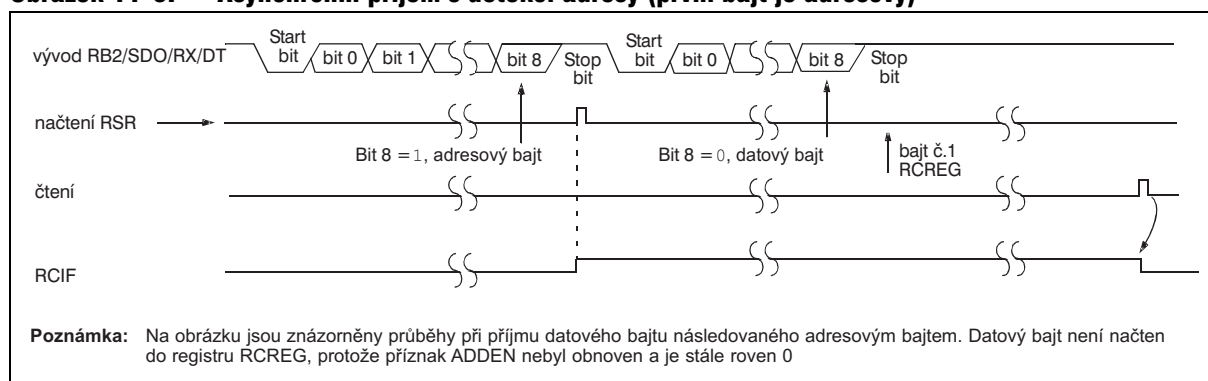
Obrázek 11-6: Blokové schéma přijímače AUSART



Obrázek 11-7: Asynchronní příjem s detekcí adresy



Obrázek 11-8: Asynchronní příjem s detekcí adresy (první bajt je adresový)



Tabulka 11-9: Registry, které mají vztah k asynchronnímu příjmu modulem AUSART

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druhých resetu
0Bh, 8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IE	INT0IF	RBFIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF (*)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
1Ah	RCREG	Příjmový registr								0000 0000	0000 0000
8Ch	PIE1	—	ADIE (*)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Konfigurační registr pro stanovení přenosové rychlosti								0000 0000	0000 0000

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s AUSART.

Poznámka 1: Tento bit je použit jen u mikropočítače PIC16F88. U mikropočítače PIC16F87 je čten jako nula.

11.3 Synchronní režim modulu AUSART – nastavení Master

V synchronní režimu jsou data přenášena v poloduplexním způsobem, ve stejnou dobu nemůžou být data současně vysílána a zároveň přijímána. Při vysílání dat je příjem zakázán. Synchronní režim je zvolen nastavením bitu SYNC v registru TXSTA<4>. Řídící bit SPEN v registru RCSTA<7> je nastaven v závislosti na konfiguraci vývodů RB5/SS/TX/CK a RB2/SDO/RX/DT pro přenos dat nebo hodinového signálu. V nastavení Master mikropočítač vysílá na sběrnici hodinový signál. Nastavení Master je zvoleno nastavením bitu CSRC v registru TXSTA<7>.

11.3.1 Vysílání v synchronním režimu modulem AUSART v nastavení Master

Blokové schéma vysílače v synchronním režimu je uvedeno na obrázku 11-6. Jádrem vysílače je tvořeno posuvným registrem TSR. Do tohoto registru je zapsán bajt pro vysílání z výstupního registru TXREG. Do registru TXREG je možno zapisovat softwarově. Zápis do registru TSR je prováděn automaticky. K tomuto zápisu nemůže dojít do doby, dokud není vyslán poslední bit z předešlého vysílaného bajtu. Jakmile dojde k zápisu do výstupního registru TSR, dojde k uvolnění výstupního registru TXREG a je nastaven příznak prázdnosti TXIF v registru PIR1<4>.

Přerušení může být povoleno nastavením bitu TXIE v registru PIE1<4>. Příznak prázdnosti TXIF je nastavován nezávisle na povolení přerušení a nelze jej nulovat softwarově. K jeho nulování dojde výhradně až dalším zápisem do výstupního registru TXREG. Zatímco příznak TXIF indikuje prázdnost registru TXREG, prázdnost registru TSR je indikována nastavením příznaku TRMT v registru TXSTA<1>. Tento příznak je možno pouze číst a je nastaven jen tehdy, pokud je posuvný registr TSR prázdný. Tento příznak nemůže vyvolat žádné přerušení. Registr TSR není mapován do žádné oblasti paměti RWM.

Vysílání je povoleno nastavením bitu TXEN v registru TXSTA<5>. Vysílání nebude zahájeno, dokud nebude zapsán vysílaný bajt do registru TXREG. První bit bude vyslán s první následující náběžnou hranou hodinového signálu na vývodu CK. Data jsou platná se sestupnou hranou tohoto signálu. Časové průběhy vysílání jsou uvedeny na obrázku 11-9. Vysílání může být rovněž zahájeno tak, že nejprve dojde k zápisu do registru TXREG a teprve poté k povolení vysílání nastavením bitu TXEN. Obvykle je před zahájením vysílání posuvný registr TSR prázdný. Zápis do výstupního registru TXRWG způsobí tedy okamžitý zápis do tohoto posuvného registru a uvolnění registru TXREG pro další zápis. Tato situace je znázorněna na obrázku 11-10.

Nulování řídicího bitu TXEN během přenosu způsobí přerušení vysílání a vynulování posuvného registru s pochopitelnou ztrátou dat. Výsledkem zákazu vysílání je přechod vývodů DT a CK do třetího stavu. Pokud jeden z bitů CREN nebo SREN přejde během vysílání do stavu log.1, vysílání je přerušeno a vývod DT přejde do stavu vysoké impedance (pro další příjem). Pokud je bit SCRC nastaven, zůstane vývod CK v režimu výstupu (vnitřní hodinový signál). Vysílací logika není resetována, ale je odpojena od výstupních vývodů. Aby došlo k resetu vysílací logiky, je nutné programově nulovat bit TXEN. Pokud je bit SREN v log.1, pak po přijetí jednoho bajtu je bit SREN nulován a modul sériového přenosu přechází do režimu vysílání, i když je TXEN stále v log.1. Vývod DT přechází ze stavu vysoké impedance při příjmu do výstupního režimu odpovídajícímu vysílání. Aby k tomuto nedocházelo, je nutné udržovat bit TXEN v log.0.

Pokud je zvolena 9.bitová komunikace nastavením bitu TX9 v registru TXSTA<6>, je nutné zapsat devátý bit před zahájením přenosu do bitu TX9D v registru TXSTA<0>. Tento devátý bit musí být zapsán dříve, než dojde k zápisu vysílaného bajtu do výstupního registru TXREG. Zápis do registru TXREG totiž může mít za následek okamžitý zápis do posuvného registru (pokud je tento prázdný) a devátý bit by mohl být vyslán špatně.

Při synchronním vysílání v nastavení Master je nutné provést následující kroky:

1. Nastavit správně obsah registru SPBRH a bit BRGH podle požadované přenosové rychlosti.
Podrobněji je pojednáno v **kapitole 11.1**.
2. Povolit synchronní sériový přenos s nastavením Master nastavením bitů SYNC, SPEN CSRC.
3. Pokud je požadováno přerušení, povolit přerušení nastavením bitu TXIE.
4. Pokud je požadován 9.bitový přenos, nastavit bit TX9.
5. Povolit vysílání nastavením bitu TXEN.
6. Pokud je nastavena 9.bitová komunikace, zapsat devátý bit do bitu TX9D
7. Zapsat data do výstupního registru TXREG (tím dojde k faktickému zahájení vysílání)
8. Pokud používáte přerušení, ujistěte se, že jsou nastaveny bity GIE a PEIE v registru INTCON.

PIC16F87/88

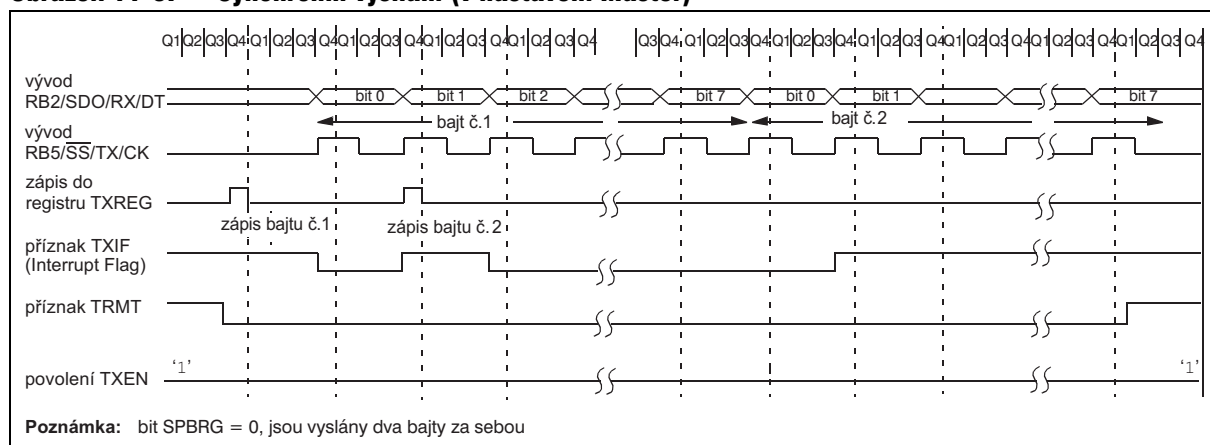
Tabulka 11-10: Registry, které mají vztah k synchronnímu vysílání modulem AUSART v nastavení Master

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druhích resetu
0Bh,8Bh,10Bh,18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IE	INT0IF	RBFIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF (*)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
19h	TXREG	Výstupní registr								0000 0000	0000 0000
8Ch	PIE1	—	ADIE (*)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Konfigurační registr pro stanovení přenosové rychlosti								0000 0000	0000 0000

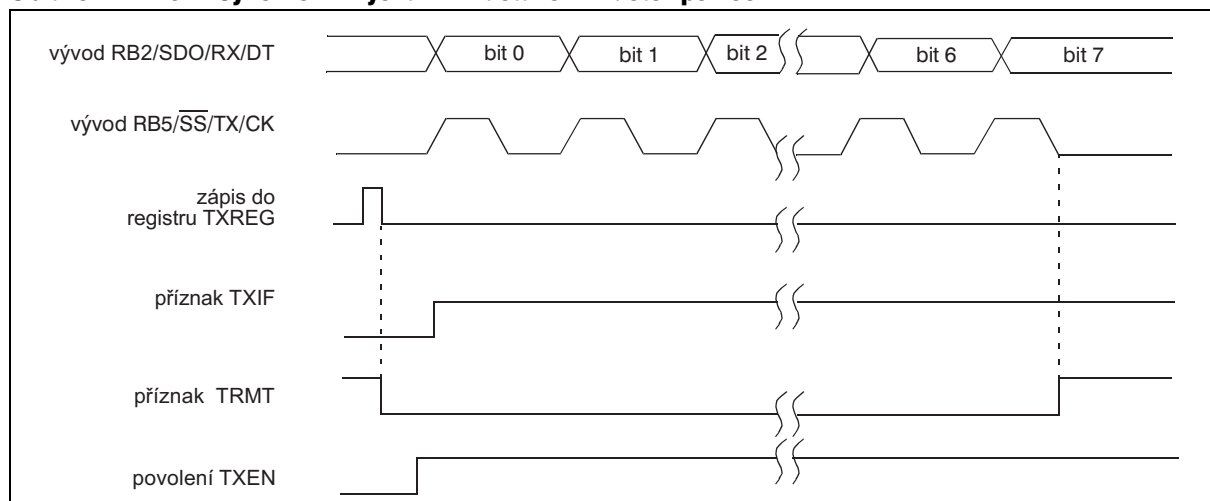
Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s AUSART.

Poznámka 1: Tento bit je použit jen u mikropočítače PIC16F88. U mikropočítače PIC16F87 je čten jako nula.

Obrázek 11-9: Synchronní vysílání (v nastavení Master)



Obrázek 11-10: Synchronní vysílání v nastavení Master pomocí TXEN



11.3.2 Synchronní přijímač modulu AUSART v nastavení Master

Jakmile je zvolen synchronní režim, příjem dat je povolen nastavením bitu SREN v registru RCSTA<5> nebo bitu CREN v registru RCSTA<4>. Data jsou sejmuta z vývodu RB2/SDO/RX/DT se sestupnou hranou hodinového signálu. Pokud je nastaven bit SREN, je možný příjem pouze jediného bajtu. Pokud je nastaven bit CREN, pokračuje příjem až do okamžiku, kdy je tento bit nulován. Pokud jsou nastaveny oba tyto bity, má přednost CREN.

Po přijmutí posledního bitu jsou data z posuvného registru přesunuta do příjmového registru RCREG, pokud je prázdný. Pokud je přenos kompletní, je nastaven příznakový bit RCIF v registru PIR1<5>. Tato událost může vyvolat přerušení, pokud je povoleno nastavením bitu RCIE v registru PIE1<5>.

Příznak CIF je určen pouze ke čtení, a je nulován hardwarově. Je nulován čtením příjmacího registru RCREG a to jen tehdy, jestli je po čtení tento registr prázdný. Příjmací registr RCREG je totiž konstruován jako paměť FIFO o velikosti 2 bajty. V příjmacím registru RCREG tedy mohou být dva přijaté nepřechtené bajty, zatímco dochází ke čtení třetího bajtu. K přetečení vstupního bufferu dojde tedy teprve po přijmutí stop bitu třetího bajtu (pokud v té době je stále nepřechtený RCREG se dvěma dříve přijatými bajty). Chyba přetečení je indikována nastavením příznaku OERR v registru RCSTA<1>. Bajt přijatý jako třetí je tím ztracen.

Pro získání přijatých dat je nutné číst registr RCREG. Pokud je tento registr čten dvakrát za sebou, jsou získány dva po sobě následující přijaté bajty (pokud jsou v paměti FIFO přítomny). Příznak chyby přetečení vstupního bufferu OERR je nulován softwarově. K jeho nulování je nutné nulovat a znovu nastavit bit CREN. Pokud je příznak OERR nastaven, je jakýkoli přenos z registru RSR do příjmacího registru RCREG zakázán. V takovém případě pochopitelně nejsou přijímána žádná další data.

Devátý bit je uložen v bitu RX9D a to stejným způsobem jako přijatý bajt, tedy ve struktuře FIFO. Čtení registru RCREG způsobí nastavení tohoto bitu do stavu, který odpovídá aktuálně načtenému bajtu. Tímto způsobem je zajištěno, že nemůže dojít ke ztrátě dřívějšího devátého bitu.

Při synchronním příjmu je nutné dodržet následující kroky:

1. Nastavit správně obsah registru SPBRH a bit BRGH podle požadované přenosové rychlosti.
Podrobněji je pojednáno v **kapitole 11.1**.
2. Povolit synchronní sériový přenos nastavením bitů SYNC, SPEN a CSRC.
3. Přesvědčete se, že jsou bity CREN a SREN v nule
4. Pokud je požadováno přerušení, povolte přerušení nastavením bitu RCIE.
5. Pokud je požadován 9-bitový přenos, nastavte bit RX9.
6. Povolte kontinuální příjem nastavením bitu CREN nebo jednorázový nastavením bitu SREN
7. Příznakový bit RCIF bude nastaven po příjmu 1. bajtu. Tato událost může způsobit přerušení, pokud je povoleno bitem RCIE.
8. Přečíst obsah registru RCSTA a zjistit případné chyby při přenosu nebo určit případný devátý bit přenosu.
9. Přečíst přijatý bajt z registru RCREG.
10. Pokud byla během přenosu indikována chyba, nulovat bit CREN (a následně nastavit)
11. Při použití přerušení se ujistěte, že jsou správně nastaveny bity GIE a PEIE v registru INTCON.

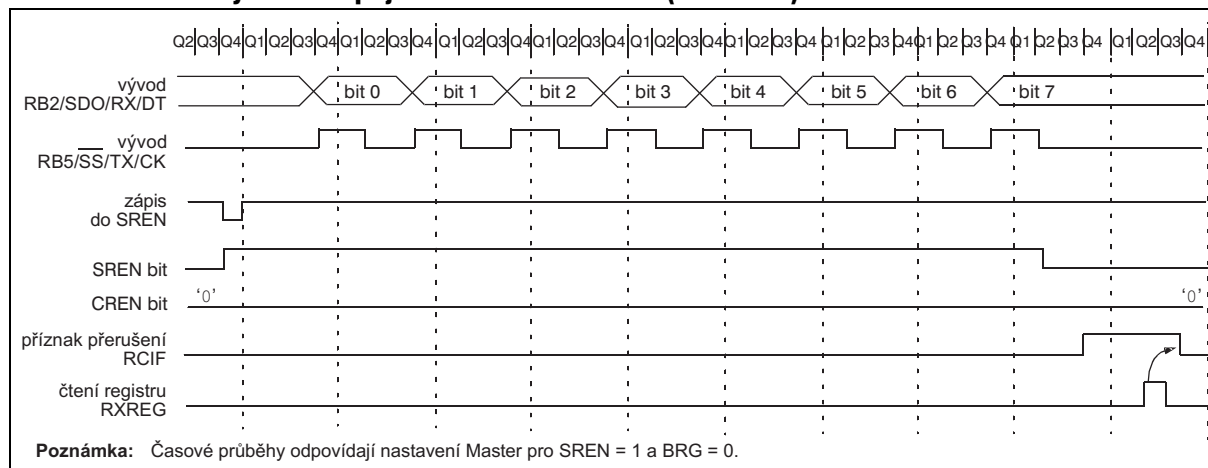
Tabulka 11-11: Registry, které mají vztah k synchronnímu příjmu modulem AUSART

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druhých resetu
0Bh,8Bh,10Bh,18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IE	INT0IF	RBFIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF (*)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
1Ah	RCREG	Příjmový registr								0000 0000	0000 0000
8Ch	PIE1	—	ADIE (*)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Konfigurační registr pro stanovení přenosové rychlosti								0000 0000	0000 0000

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s AUSART v synchronním režimu s nastavením Master.

Poznámka 1: Tento bit je použit jen u mikropočítače PIC16F88. U mikropočítače PIC16F87 je čten jako nula.

Obrázek 11-11: Synchronní příjem v nastavení Master (SREN = 1)



11.4 Synchronní přijímač modulu AUSART v nastavení Slave

Synchronní přenos v nastavení Slave se liší od nastavení Master tím, že zdroj hodinového signálu pro přenos je zajištěn externím zařízením a signál je přiveden na vývod RB5/SS/TX/CK. To umožňuje mikropočítači vysílat nebo přijímat data i v režimu Sleep. Nastavení Slave je zvoleno nulováním bitu CSRC v registru TXSTA<7>.

11.4.1 Vysílání v synchronním režimu modulem AUSART v nastavení Slave

Vysílání v synchronním režimu je stejné v nastavení Master i Slave kromě vysílání v režimu Sleep.

Pokud jsou do výstupního registru TXREG zapsány dva bajty po sobě a následně dojde k zahájení režimu Sleep, stane se následující:

- první bajt bude přepsán do posuvného registru TSR a vyslán
- druhý bajt zůstane v registru TXREG
- příznak přerušení TXIF nebude nastaven
- jakmile bude vyslán první bajt, dojde k načtení druhého bajtu z registru TXREG do posuvného registru TSR a bude nastaven příznak přerušení TXIF
- pokud je povoleno přerušení bitem TXIE, dojde k probuzení mikropočítače z režimu Sleep a pokud je zároveň povolen GIE, dojde k obsluze přerušení na adrese 0004h

Při synchronním příjmu v nastavení Slave je nutné dodržet následující kroky:

- Povolit synchronní sériový přenos nastavením bitů SYNC, SPEN a nulováním CSRC.
- Vynulovat bity CREN a SREN.
- Pokud je požadováno přerušení, povolit přerušení nastavením bitu TXIE.
- Pokud je požadován 9-bitový přenos, nastavit bit TX9.
- Povolit vysílání nastavením bitu TXEN.
- Pokud je nastavena 9-bitová komunikace, zapsat devátý bit do bitu TX9D
- Zapsat data do výstupního registru TXREG (tím dojde k faktickému zahájení vysílání)
- Pokud používáte přerušení, ujistěte se, že jsou nastaveny bity GIE a PEIE v registru INTCON.

Tabulka 11-12: Registry, které mají vztah k synchronnímu vysílání modulem AUSART v nastavení Slave

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druzích resetu
0Bh,8Bh,10Bh,18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IE	INT0IF	RBFIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF (*)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
19h	TXREG	Výstupní registr								0000 0000	0000 0000
8Ch	PIE1	—	ADIE (*)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Konfigurační registr pro stanovení přenosové rychlosti								0000 0000	0000 0000

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s USART v synchronním vysílání v nastavení Slave.

Poznámka 1: Tento bit je použit jen u mikropočítače PIC16F88. U mikropočítače PIC16F87 je čten jako nula.

11.4.2 Synchronní přijímač modulu AUSART v nastavení Slave

Příjem v synchronním režimu je stejný v nastavení Master i Slave kromě příjmu v režimu Sleep. Stav bitu SREN nemá v nastavení Slave žádný význam.

Pokud je příjem povolen nastavením bitu CREN před uvedením mikropočítače do režimu Sleep, je možné v tomto režimu přijímat data. Jakmile bude přijat první bajt, dojde k přesunu přijatého bajtu z posuvného registru RSR do registru RCREG a bude nastaven příznak přerušení RCIF.

Pokud je povoleno přerušení bitem RCIE, dojde k probuzení mikropočítače z režimu Sleep a pokud je zároveň povolen GIE, dojde k obsluze přerušení na adrese 0004h.

Při synchronním příjmu v nastavení Slave je nutné dodržet následující kroky:

1. Povolit synchronní sériový přenos nastavením bitů SYNC, SPEN a nulováním bitu CSRC.
2. Pokud je požadováno přerušení, povolit přerušení nastavením bitu RCIE.
3. Pokud je požadován 9.bitový přenos, nastavit bit RX9.
4. Povolit kontinuální příjem nastavením bitu CREN
5. Příznakový bit RCIF bude nastaven po příjmu 1. bajtu. Tato událost může způsobit přerušení, pokud je povoleno bitem RCIE.
6. Přečíst obsah registru RCSTA a zjistit případné chyby při přenosu nebo určit případný devátý bit přenosu.
7. Přečíst přijatý bajt z registru RCREG.
8. Pokud byla během přenosu indikována chyba, nulovat bit CREN (a následně nastavit)
9. Při použití přerušení se ujistěte, že jsou správně nastaveny bity GIE a PEIE v registru INTCON.

Tabulka 11-13: Registry, které mají vztah k synchronnímu příjmu modulem AUSART v nastavení Slave

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druhích resetu
0Bh,8Bh,10Bh,18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IE	INT0IF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF (1)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
1Ah	RCREG	Příjmový registr								0000 0000	0000 0000
8Ch	PIE1	—	ADIE (1)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Konfigurační registr pro stanovení přenosové rychlosti								0000 0000	0000 0000

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s AUSART v synchronním režimu s nastavením Slave.

Poznámka 1: Tento bit je použit jen u mikropočítače PIC16F88. U mikropočítače PIC16F87 je čten jako nula.

12 MODUL AD PŘEVODNÍKU

Analogově digitální převodník (AD převodník) umožňuje převod analogového vstupního signálu na binární číslo o šířce 10 bitů. AD převodník má vstup pro horní i dolní mez referenčního napětí. Tato mez může být volena buď jako napájecí napětí nebo může být přivedena na vnější vývody RA2 a RA3. AD převodník má celkem 7 vstupů.

AD převodník je schopen činnosti i v režimu Sleep. V tomto režimu je nutné zajistit časování AD převodu pomocí vnitřního RC oscilátoru AD převodníku.

Modul AD převodníku má celkem 5 registrů:

- ADRESH je registr pro horní část výsledku
- ADRESL je registr pro dolní část výsledku
- ADCON0 je první řídicí registr
- ADCON1 je druhý řídicí registr
- ANSEL je registr pro konfiguraci analogových vstupů

Pomocí registru ADCON0 (viz registr 12-2) lze ovládat činnost AD převodníku. Pomocí registru ADCON1 (viz registr 12-3) a ANSEL (viz registr 12-1) se volí konfigurace zapojení vstupních obvodů. Každý z uvedených vývodů může být konfigurován jako analogový vstup nebo univerzální digitální vstup/výstup. Vývody RA3 a RA2 mohou být konfigurovány rovněž jako vstupy pro napěťovou referenci.

Další podrobnosti jsou uvedeny v literatuře DS33023 s názvem "PICmicro® Mid-Range MCU Family Reference Manual".

Registr 12-1: ANSEL – registr volby analogových vstupů (adresa: 9Bh, jen PIC16F88)

U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0
bit 7							bit 0

bit 7 **Nepoužít:** při čtení '0'

bit 6-0 **ANS6:ANS0:** volba analogového režimu na vstupu příslušného vývodu AN(6:0)

1 = Analogový vstup – vývod je konfigurován jako analogový vstup **(1,2)**

0 = Digitální vstup – vývod je použit jako obvyklý digitální vstup nebo výstup

Poznámka 1: Nastavením vývodu jako analogového vstupu je současně automaticky zakázán režim digitálního vstupu, použití Pull-up rezistorů a přerušení změnou na vývodu. Odpovídající řídicí bit v registru TRISA musí být nastaven ve vstupním režimu, aby napětí na vývodu nebylo ovlivňováno ničím jiným než externím analogovým obvodem. Pouze vývod AN2 je analogový vstup/výstup, ostatní vývody jsou jen vstupy.

2: Pro lepší pochopení ovládání vývodů pomocí registru ANSEL viz blokové schéma analogových vývodů.

Význam:

R = bit pro čtení

–n = hodnota po resetu POR

W = bit pro zápis

'1' = bit je nastaven

U = nepoužitý bit, čteno jako '0'

'0' = bit je v nule

x = hodnota bitu je neznámá

PIC16F87/88

Registr 12-2: ADCON0 – řídicí registr AD převodníku (adresa: 1Fh, jen PIC16F88)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	—	ADON
bit 7							bit 0

- bit 7-6 **ADCS1:ADCS0:** Volba zdroje časování pro AD převodník
 pro ADCS2 = 0
 00 = Fosc/2
 01 = Fosc/8
 10 = Fosc/32
 11 = Frc (časování je odvozeno od vnitřního RC oscilátoru AD převodníku)
 pro ADCS2 = 1
 00 = Fosc/4
 01 = Fosc/16
 10 = Fosc/64
 11 = Frc (časování je odvozeno od vnitřního RC oscilátoru AD převodníku)
- bit 5-3 **CHS2:CHS0:** volba analogového vstupu pro AD převodník, bit ADCS2 je v registru ADCON1
 000 = 1. vstup AD převodníku na vývodu AN0
 001 = 2. vstup AD převodníku na vývodu AN1
 010 = 3. vstup AD převodníku na vývodu AN2
 011 = 4. vstup AD převodníku na vývodu AN3
 100 = 5. vstup AD převodníku na vývodu AN4
 101 = 6. vstup AD převodníku na vývodu AN5
 110 = 7. vstup AD převodníku na vývodu AN6
- bit 2 **GO/DONE:** řídicí a stavový bit AD převodníku
 1 = probíhá AD převod. Nastavením tohoto bitu do log.'1' dojde k zahájení AD převodu
 tento bit je automaticky nulován po dokončení AD převodu
 0 = AD převod neprobíhá, resp. byl dokončen
- bit 1 **Nepoužito:** při čtení '0'
- bit 0 **ADON:** zapnutí AD převodníku
 1 = AD převodník je zapnutý
 0 = AD převodník je vypnutý a neodebírá žádný proud

Význam:

R = bit pro čtení W = bit pro zápis U = nepoužitý bit, čteno jako '0'
 -n = hodnota po resetu POR '1' = bit je nastaven '0' = bit je v nule x = hodnota bitu je neznámá

PIC16F87/88

Registr 12-3: **ADCON1 – druhý řídicí registr AD převodníku (adresa: 9Fh jen PIC16F88)**

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
ADFM	ADCS2	VCFG1	VCFG0	—	—	—	—
bit 7							bit 0

- bit 7 **ADFM:** Volba formátu výsledku AD převodu
1 = řazení zprava (šest nejvýznamnějších bitů ADRESH je čteno jako "0")
0 = řazení zleva (šest nejméně významných bitů ADRESL je čteno jako "0")
- bit 6 **ADCS2:** Volba zdroje časování pro AD převodník – nejvýznamnější bit, viz registr ADCON0
1 = časování AD převodu /2 při časování od taktovací frekvence
0 = zakázáno
- bit 5-4 **VCFG1:VCFG0:** Konfigurační bity reference AD převodníku

kombinace	VREF+	VREF-
00	AVDD	AVSS
01	AVDD	VREF-
10	VREF+	AVSS
11	VREF+	VREF-

Poznámka: Při použití vnější napěťové reference je nutné vývody AN3 a AN2 konfigurovat jako analogové vstupy

- bit 3-0 **Nepoužito:** při čtení '0'

Význam:

R = bit pro čtení W = bit pro zápis U = nepoužitý bit, čteno jako '0'
 -n = hodnota po resetu POR '1' = bit je nastaven '0' = bit je v nule x = hodnota bitu je neznámá

PIC16F87/88

AD převod je zahájen nastavením bitu $\overline{GO}/\overline{DONE}$. Probíhající AD převod lze přerušit programově nulováním bitu $\overline{GO}/\overline{DONE}$. V tomto případě nebudou aktualizovány registry obsahující výsledek AD převodu ADRESH a ADRESL. Tyto registry budou tedy obsahovat výsledek předchozího AD převodu. Po přerušeném AD převodu je vyžadováno zpoždění nejméně 2 TAD před aktivací dalšího vzorkování. Po uplynutí této doby je vzorkovací kondenzátor automaticky připojen na původně zvolený analogový vývod. Bit $\overline{GO}/\overline{DONE}$ v registru ADCON0 nesmí být nastaven stejnou instrukcí, která provede zapnutí AD převodníku nastavením bitu ADON ve stejném registru. Blokové schéma AD převodníku je uvedeno na obrázku 12-1.

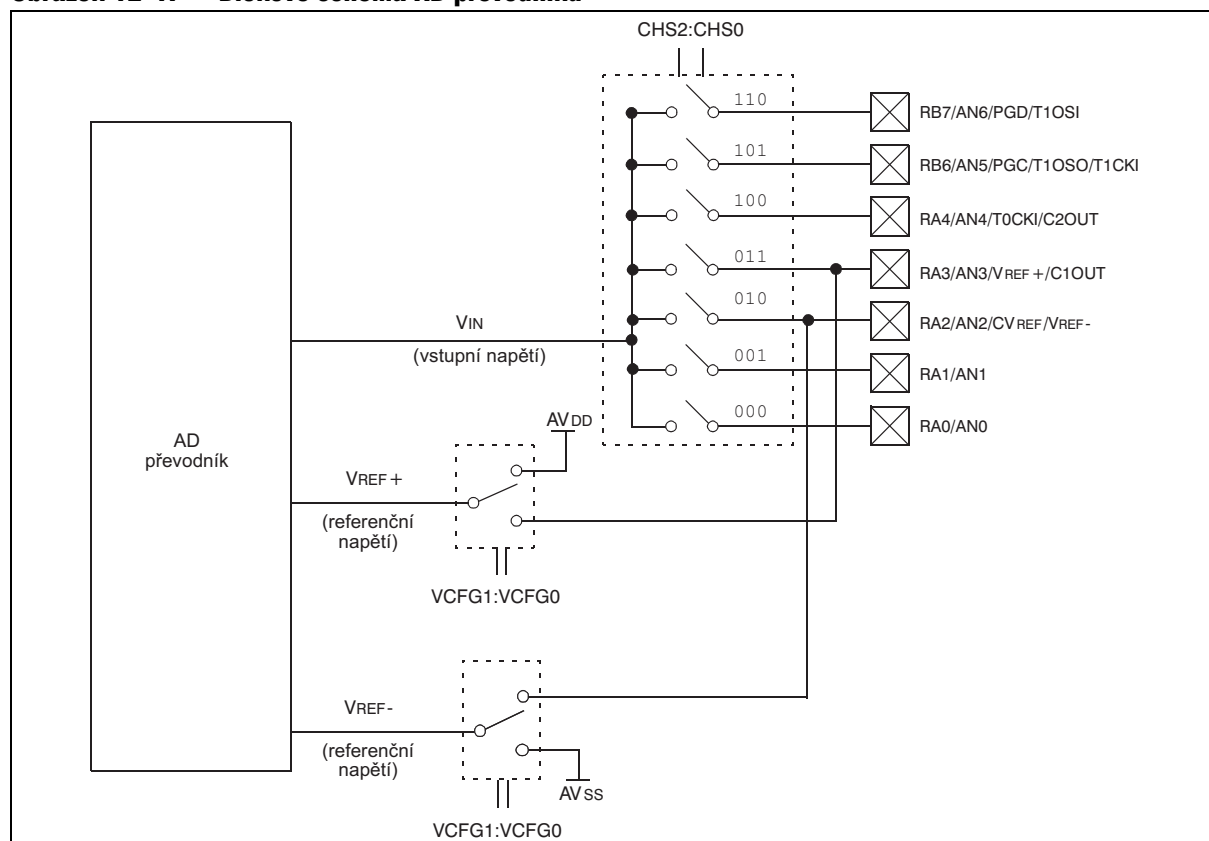
Modul AD převodníku je nutné správně nakonfigurovat a provést připojení vstupních vývodů. Konfigurace vývodu používaných jako analogové vstupy musí být odpovídající v registru TRISB.

Vstup s měřeným napětím musí být připojený na vstup AD převodníku po určitou minimální dobu, potřebnou k nabití vzorkovacího kondenzátoru. Teprve po uplynutí této doby je možné zahájit AD převod. Podrobnosti jsou uvedeny v **kapitole 12.1**.

Při práci s AD převodníkem je nutné zajistit následující kroky:

1. Konfigurovat AD převodník
 - rozdělit analogové a digitální vývody v registru ANSEL
 - určit napěťovou referenci v registru ADCON1
 - vybrat vstupní kanál AD převodníku v registru ADCON
 - určit zdroj časování AD převodu v registru ADCON0
 - zapnout AD převodník v registru ADCON0
2. Konfigurovat přerušovací systém (pokud je přerušení používáno)
 - vynulovat příznak ADIF
 - nastavit bit ADIE
 - nastavit bit PEIE
 - nastavit bit GIE
3. Počkat dobu nezbytnou pro nabití vzorkovacího kondenzátoru
4. Zahájit převod
 - nastavit bit $\overline{GO}/\overline{DONE}$ v registru ADON0
5. Čekat na dokončení AD převodu, a to jedním z následujících způsobů:
 - testovat bit $\overline{GO}/\overline{DONE}$, zda nebyl nulován (bez použití přerušení)
 - čekat na přerušení od dokončení AD převodu
6. Přečíst výsledek AD převodu z dvojice registrů ADRESH a ADRESL, nulovat příznak ADIF, pokud bude dále používáno přerušení
7. Pro další převod postupovat od bodu č.1 nebo č.2.

Obrázek 12-1: Blokové schéma AD převodníku



12.1 Požadavky na vzorkování AD převodníku

Pro dosažení maximální přesnosti AD převodu musí být zajištěno, aby vzorkovací kondenzátor CHOLD byl nabit na skutečnou úroveň vstupního napětí. Náhradní zapojení vstupního obvodu AD převodníku je uvedeno na obrázku 12-2. Impedance zdroje signálu R_S a velikost vzorkovacího kondenzátoru CHOLD přímo ovlivňují časovou konstantu nabíjení vzorkovacího kondenzátoru CHOLD. Odpor vzorkovacího spínače RSS je závislý na napájecím napětí V_{DD} jak je uvedeno na obrázku 12-2. **Nejvyšší doporučená impedance zdroje signálu je 10 k Ω .** Pokud je tato impedance nižší, je kratší i RC konstanta vstupního vzorkovacího obvodu. Po připojení analogového vstupu na vzorkovací kondenzátor musí být dodržen čas na nabití tohoto kondenzátoru na hodnotu vstupního napětí. Teprve po uplynutí této vzorkovací doby dojde k ustálení napětí na vzorkovacím kondenzátoru může dojít k zahájení AD převodu.

Pro výpočet nejkratší nutné doby pro ustálení napětí je možné použít vzorec 12-1. Tento vztah předpokládá maximální dovolenou chybu 1/2 LSB. Při 10-bitovém převodu můžeme získat 1 hodnotu z 1024. Pro zajištění rozlišení AD převodníku je nutná maximální chyba právě v rámci jedné poloviny nejnižšího bitu výsledku.

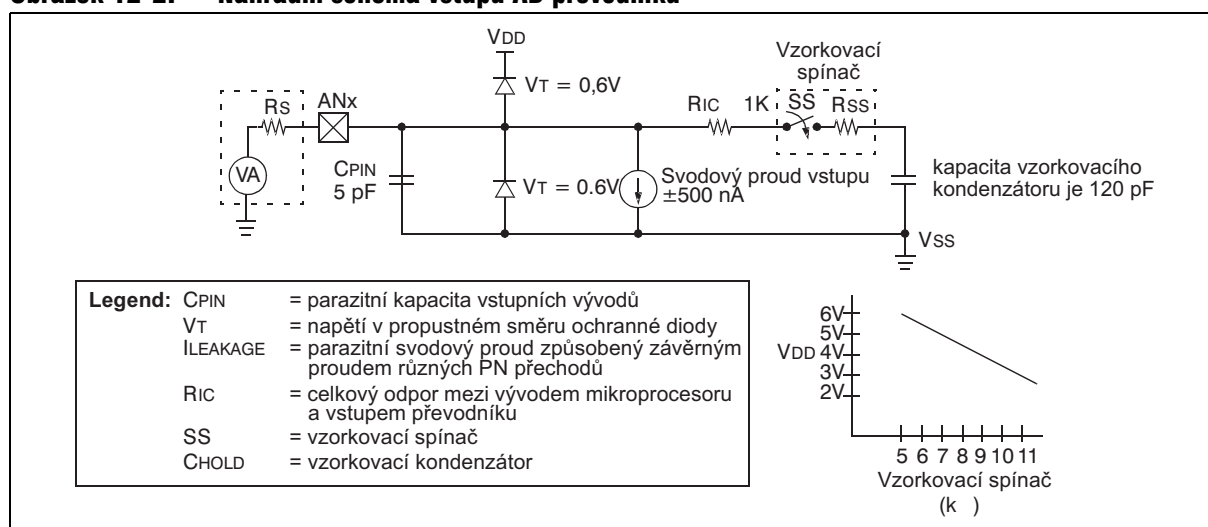
Další podrobnosti jsou uvedeny v literatuře DS33023 s názvem "PICmicro® Mid-Range MCU Family Reference Manual".

Vzorec 12-1 Vzorkovací doba

TACQ (vzorkovací doba)	=	doba ustálení zesilovače + doba nabití vzorkovacího kondenzátoru + teplotní koeficient
	=	TAMP + TC + TCOFF
	=	2 μ s + TC + [(teplota -25 °C)(0,05 μ s/°C)]
TC	=	CHOLD (RIC + RSS + RS) ln(1/2047)
	=	-120 pF (1 k Ω + 7 + 10) ln(0,0004885)
	=	16,47 μ s
TACQ	=	2 μ s + 16,47 μ s + [(50 °C - 25 °C)(0,05/ μ s/°C)]
	=	19,72 μ s

- Poznámka 1:** Velikost referenčního napětí nemá vliv na vzorkovací dobu
- 2:** Vzorkovací kondenzátor není vybíjen po ukončení převodu
- 3:** Nejvyšší doporučená impedance zdroje vstupního signálu je 10 k Ω .
S touto hodnotou je počítáno při udávání elektrických charakteristik součástky (zejména svodového proudu vstupu).
- 4:** Po dokončení AD převodu je nutné počkat minimálně po dobu 2,0 TAD.
Po tuto dobu není vzorkovací kondenzátor připojen k vybranému vstupu.

Obrázek 12-2: Náhradní schéma vstupu AD převodníku



12.2 Časování AD převodu

Kompletní AD převod vyžaduje celkem 9 časových cyklů označených jako T_{AD} . Doba tohoto cyklu může být odvozena od taktovacího oscilátoru nebo od vnitřního RC oscilátoru. Nastavením bitů $ADCS2:ADCS0$ v registrech $ADCON0$ a $ADCON1$ lze provést volbu jednoho ze sedmi druhů časování AD převodu:

- $F_{osc} / 2$
- $F_{osc} / 4$
- $F_{osc} / 8$
- $F_{osc} / 16$
- $F_{osc} / 32$
- $F_{osc} / 64$
- FRC (odvozeno od vnitřního RC oscilátoru AD převodníku)

Pro správný AD převod je nutné zajistit, aby doba T_{AD} byla zvolena v rozmezí od 1,6 μs do 6,4 μs . V tabulce 12-1 je uvedena kalkulace doby T_{AD} pro několik obvyklých taktovacích frekvencí.

12.3 Práce s AD převodníkem v režimu s řízenou spotřebou

AD převodník je možné používat i v režimu se sníženou spotřebou, kdy je zdrojem taktovací frekvence sekundární taktovací oscilátor.

Pokud bude mikropočítač přepnut na sekundární taktovací oscilátor a zdroj časování pro AD převodník je odvozen z taktovací frekvence, je nutné před samotným přepnutím překonfigurovat časování AD převodu na novou frekvenci. Zahájit AD převod je možné až po provedení přepnutí. Během probíhajícího AD převodu není možné přepínat zpět na primární taktovací oscilátor.

Pokud je frekvence sekundárního taktovacího oscilátoru menší než 1 MHz, je nutné použít jako zdroj časování AD převodníku jeho vlastní vnitřní RC oscilátor.

Tabulka 12-1: Závislost doby T_{AD} na taktovací frekvenci

Volba časování AD převodu			Maximální taktovací frekvence mikropočítače
Nastavení	$ADCS <2>$	$ADCS <1:0>$	Max.
2 T _{osc}	0	00	1,25 MHz
4 T _{osc}	1	00	2,5 MHz
8 T _{osc}	0	01	5 MHz
16 T _{osc}	1	01	10 MHz
32 T _{osc}	0	10	20 MHz
64 T _{osc}	1	10	20 MHz
RC (1,2,3)	x	11	(Poznámka 1)

Poznámka 1: Při použití vnitřního RC oscilátoru je obvyklá doba $T_{AD} = 4 \mu s$, ale může se pohybovat mezi 2 – 6 μs .

2: Pro taktovací oscilátor s frekvencí vyšší než 1 MHz je doporučeno použití zdroje časování AD převodu z vnitřního RC oscilátoru pouze v případě AD převodu během režimu Sleep

3: Při použití vnějších napěťových referencí viz **kapitulu 18.0**.

12.4 Konfigurace analogových vývodů

Konfiguraci vývodů pro práci s AD převodníkem je nutné provádět pomocí registrů ADCON1, ANSEL, TRISA a TRISB. Vývody použité jako analogové vstupy AD převodníku musí být příslušným bitem registru TRIS konfigurovány jako vstupy. Pokud by byly tyto vývody nastaveny jako výstupy, byl by prováděn AD převod odpovídajících napěťových úrovní pro log. 1 nebo log.0.

Operace AD převodníku jsou nezávislé na nastavení bitů CHS2:CHS0 a bitů registrů TRIS.

- Poznámka 1:** Při čtení pracovního registru PORTA nebo PORTB jsou bity vývodů konfigurovaných jako analogové vstupy čteny jako log.0. Analogové úrovně napětí přivedená na vstupy konfigurované jako digitální, mohou způsobit neočekávané stavy při AD převodu.
- 2:** Přivedení analogového napětí na vývod konfigurovaný jako digitální vstup může způsobit proud vývodem větší, než je uvedeno v elektrických charakteristikách mikropočítače.

12.5 AD převod

AD převod je zahájen nastavením bitu $\overline{GO/DONE}$. Probíhající AD převod lze přerušit programově nulováním bitu $\overline{GO/DONE}$. V tomto případě nebudou aktualizovány registry obsahující výsledek AD převodu ADRESH a ADRESL. Tyto registry budou tedy obsahovat výsledek předchozího AD převodu. Po přerušení AD převodu je vyžadováno zpoždění nejméně 2 TAD před aktivací dalšího vzorkování. Po uplynutí této doby je vzorkovací kondenzátor automaticky připojen na původně zvolený analogový vývod.

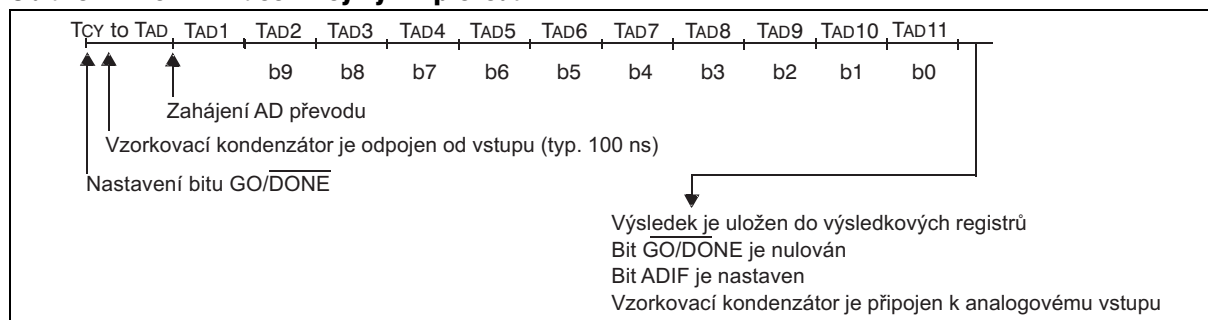
Situace je znázorněna na obrázku 12-3. Po nastavení bitu $\overline{GO/DONE}$ dojde k prodlevě v rozsahu od Tcy do Tad.

Poznámka: Bit $\overline{GO/DONE}$ v registru ADCON0 **nesmí** být nastaven stejnou instrukcí, která provede zapnutí AD převodníku nastavením bitu ADON ve stejném registru.

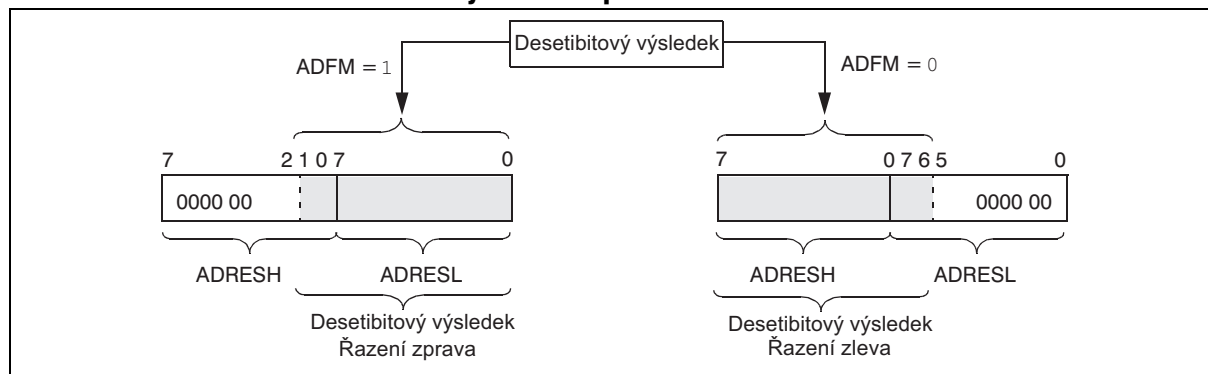
12.5.1 Výstup AD převodníku

Výstup AD převodníku je 10-bitové binární číslo a může být ve dvou výstupních registrech zapsáno ve dvou formátech. Buď je nejvýznamnější bit zapsán na krajní levou pozici registru ADRESH (tzv. řazení zleva) nebo je nejméně významný bit zapsán na krajní pravou pozici registru ADRESL (tzv. řazení zprava). Volba formátu se provádí nastavením bitu ADFM v registru ADCON1(7). Oba formáty jsou znázorněny na obrázku 12-4. Pokud není používán AD převodník je možné oba výsledkové registry používat jako univerzální 8-bitové registry.

Obrázek 12-3: Pracovní cykly AD převodu



Obrázek 12-4: Formát 10-bitového výsledku AD převodu



12.6 AD převod v režimu Sleep

Modul AD převodníku může pracovat i v režimu Sleep. Podmínkou je použití vlastního vnitřního jednoúčelového RC oscilátoru pro časování převodu. Po volbě RC oscilátoru převodník čeká jeden instrukční cyklus než je zahájen AD převod. To umožní vykonání instrukce SLEEP a omezí vliv šumu vniklého rychlým přepínáním zátěží na přesnost AD převodu. Po dokončení převodu je nulován bit GO/DONE a výsledek je umístěn do výsledkových registrů ADRESH a ADRESL. Pokud je povoleno přerušení, dojde vzápětí k probuzení ze stavu Sleep. Pokud přerušení povoleno není, modul AD převodníku se odpojí a mikropočítač zůstane ve stavu Sleep i přesto, že zůstane nastaven ovládací bit ADON.

Pokud zdroj časovacího signálu jiný než RC oscilátor, způsobí instrukce SLEEP přerušení probíhajícího převodu a vypnutí AD převodníku. Ovládací bit ADON zůstane nastaven.

Vypnutí AD převodníku v režimu Sleep zajistíme jeho minimální odběr proudu.

Poznámka: Pro použití AD převodníku v režimu Sleep je bezpodmínečně nutné zvolit časování od vlastního RC oscilátoru. Instrukce SLEEP musí následovat bezprostředně za instrukcí nastavující bit GO/DONE.

12.7 Důsledky signálu Reset

Signál Reset způsobí nastavení všech registrů do jejich inicializačního stavu. Modul AD převodníku je vypnut a případný probíhající převod je ukončen bez výsledku. Všechny příslušné vývody jsou konfigurovány jako analogové vstupy.

Obsah výsledkových registrů ADRESH a ADRESL se nemění. Obsah těchto registrů po resetu je zcela náhodný.

12.8 Použití spouštění pro modul CCP

AD převod může být zahájen speciální událostí způsobenou modulem CCP. To vyžaduje, aby byl AD převodník zapnut a bity CCP1M3: CCP1M0 byly nastaveny na kombinaci: "1011". Jakmile dojde ke zmíněné události, je automaticky nastaven bit GO/DONE, tím dojde i k zahájení AD převodu a současně je nulován čítač /časovač Timer1. Pro správnou činnost musí být odpovídající vstupy konfigurovány jako analogové a musí být zajištěna minimální doba pro nabití vzorkovacího kondenzátoru před zahájením AD převodu.

Pokud bude AD převodník vypnut, nedojde při speciální události způsobené modulem CCP k jeho spuštění. Čítač/časovač Timer1 je nulován vždy.

Tabulka 12-2: Přehled registrů a bitů, které mají vztah k AD převodníku

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druzích resetu
0Bh, 8Bh 10Bh, 18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF (1)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
8Ch	PIE1	—	ADIE (1)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
1Eh	ADRESH (2)	Vyšší bajt výsledku AD převodu								xxxx xxxx	uuuu uuuu
9Eh	ADRESL (2)	Nižší bajt výsledku AD převodu								xxxx xxxx	uuuu uuuu
1Fh	ADCON0 (2)	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	—	ADON	0000 00-0	0000 00-0
9Fh	ADCON1 (2)	ADFM	ADCS2	VCFG1	VCFG0	—	—	—	—	0000 ----	0000 ----
9Bh	ANSEL (2)	—	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	-111 1111	-111 1111
05h	PORTA (PIC16F87) (PIC16F88)	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx 0000 xxx0 0000	uuuu 0000 uuu0 0000
05h, 106hj	PORTB (PIC16F87) (PIC16F88)	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx 00xx xxxx	uuuu uuuu 00uu uuuu
85h	TRISA	TRISA7	TRISA6	TRISA5 (3)	Konfigurační registr brány PortA					1111 1111	1111 1111
86h, 186h	TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s AD převodníkem

Poznámka 1: Tento bit je použit jen u mikropočítače PIC16F88. U mikropočítače PIC16F87 je čten jako nula.

2: Jen PIC 16F88

3: Vývod 5 je pouze vstup, bit TRISA5 nemá žádný význam a jeho čtením získáme vždy log.1

13 MODUL KOMPARÁTORU

Mikropočítač má dva analogové komparátory. Vstupy komparátorů jsou sdíleny s vývody RA0-RA3, výstupy jsou sdíleny s vývody RA4 a RA3. Na vstup komparátoru lze rovněž připojit vnitřní napěťovou referenci (viz **kapitolu 14.0**).

V registru 13-1 je popsán řídicí registr komparátoru CMCON. Komparátor má celkem 8 různých možností konfigurace. Výběr konfigurace se provádí nastavením příslušných bitů v registru CMCON. Na obrázku 13-1 je uvedeno zapojení komparátoru pro všech osm možných konfigurací.

Registr 13-1: Řídicí registr komparátoru CMCON (adresa: 9Ch)

	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1	R/W-1
	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0
bit 7								bit 0
bit 7	C2OUT: výstup komparátoru C2 v případě, že je bit C2INV = 0 1 = C2 VIN+ je větší než C2 VIN- 0 = C2 VIN+ je menší než C2 VIN- v případě, že je bit C2INV = 1 1 = C2 VIN+ je menší než C2 VIN- 0 = C2 VIN+ je větší než C2 VIN-							
bit 6	C1OUT: výstup komparátoru C1 v případě, že je bit C1INV = 0 1 = C1 VIN+ je větší než C1 VIN- 0 = C1 VIN+ je menší než C1 VIN- v případě, že je bit C1INV = 1 1 = C1 VIN+ je menší než C1 VIN- 0 = C1 VIN+ je větší než C1 VIN-							
bit 5	C2INV: řídicí bit pro negaci výstupu komparátoru C2 1 = výstup komparátoru C2 je negován 0 = výstup komparátoru C2 není negován							
bit 4	C1INV: řídicí bit pro negaci výstupu komparátoru C1 1 = výstup komparátoru C1 je negován 0 = výstup komparátoru C1 není negován							
bit 3	CIS: řídicí bit přepínání invertujícího vstupu v případě, že konfigurace komparátoru odpovídá na bitech CM2-CM0 '001' 1 = invertující vstup komparátoru C1 je připojen na vývod RA3 0 = invertující vstup komparátoru C1 je připojen na vývod RA0 v případě, že konfigurace komparátoru odpovídá na bitech CM2-CM0 '010' 1 = invertující vstup komparátoru C1 je připojen na vývod RA3 invertující vstup komparátoru C2 je připojen na vývod RA2 0 = invertující vstup komparátoru C1 je připojen na vývod RA0 invertující vstup komparátoru C2 je připojen na vývod RA1							
bit 2-0	bity CM2 – CM0 určují jednu z osmi možností konfigurace komparátoru. Na obrázku 12-1 jsou znázorněny všechny možnosti konfigurace odpovídající všem kombinacím uvedených bitů.							

Význam:

R = bit pro čtení

W = bit pro zápis

U = nepoužitý bit, čteno jako '0'

–n = hodnota po resetu POR

'1' = bit je nastaven

'0' = bit je v nule

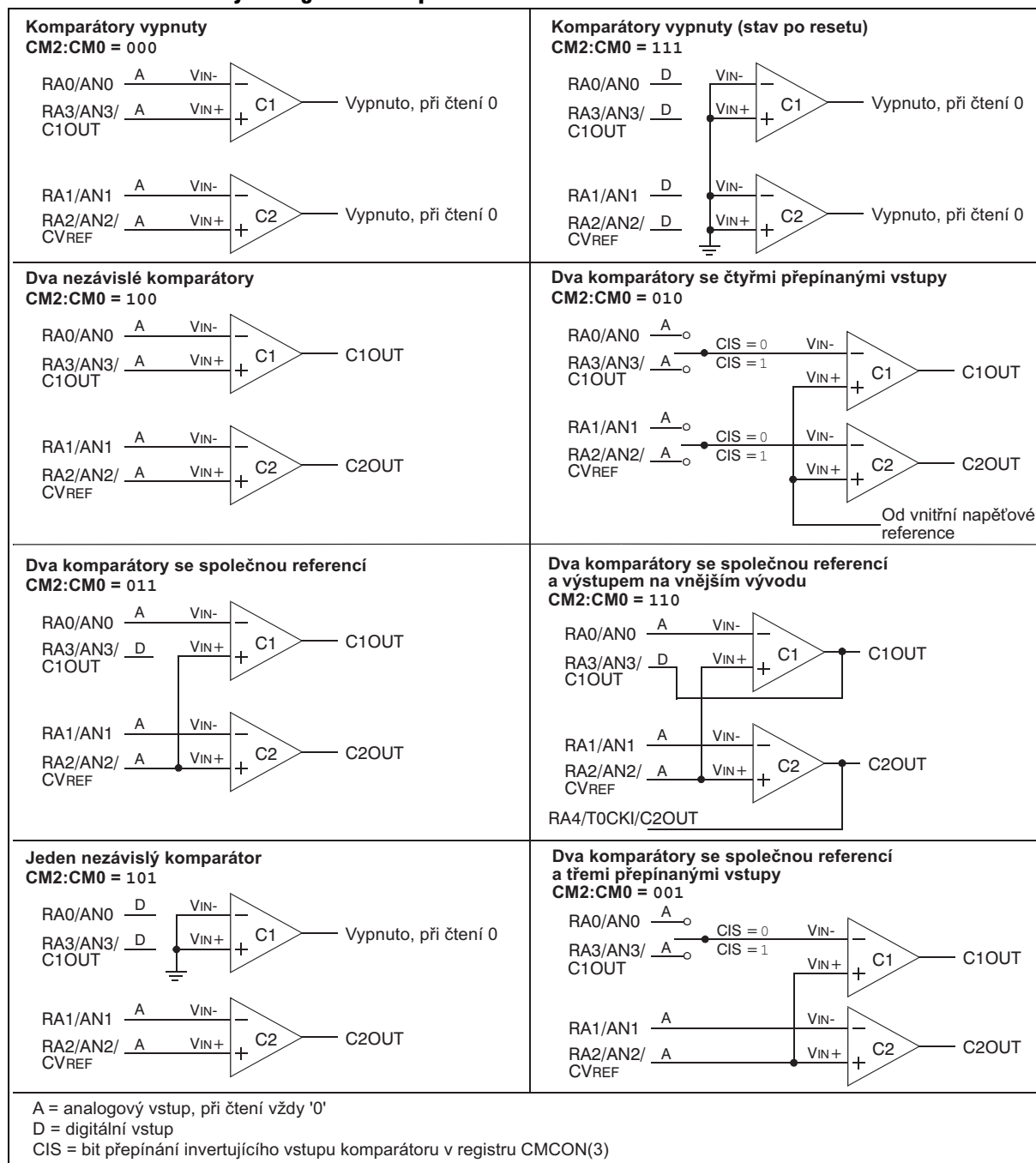
x = hodnota bitu je neznámá

13.1 Konfigurace komparátoru

Komparátor má celkem 8 různých možností konfigurace. Výběr konfigurace se provádí nastavením příslušných bitů v registru CMCON. Na obrázku 13-1 je uvedeno zapojení komparátoru pro všech osm možných konfigurací. Vývody, které používá komparátor v různých druhích konfigurace je nutné ošetřit rovněž v registru TRISA, který určuje, zda bude vývod ve vstupním nebo výstupním režimu. Při změně druhu konfigurace komparátoru není po určitou dobu zaručena platná hodnota na výstupu komparátoru. Podrobnosti jsou uvedeny v kapitole 18.0.

Poznámka: Při změně druhu konfigurace je nezbytné zakázat přerušení od komparátoru. V opačném případě může nastat při změně nežádoucí přerušení, způsobené nedefinovanými hodnotami.

Obrázek 13-1: Druhy konfigurace komparátoru



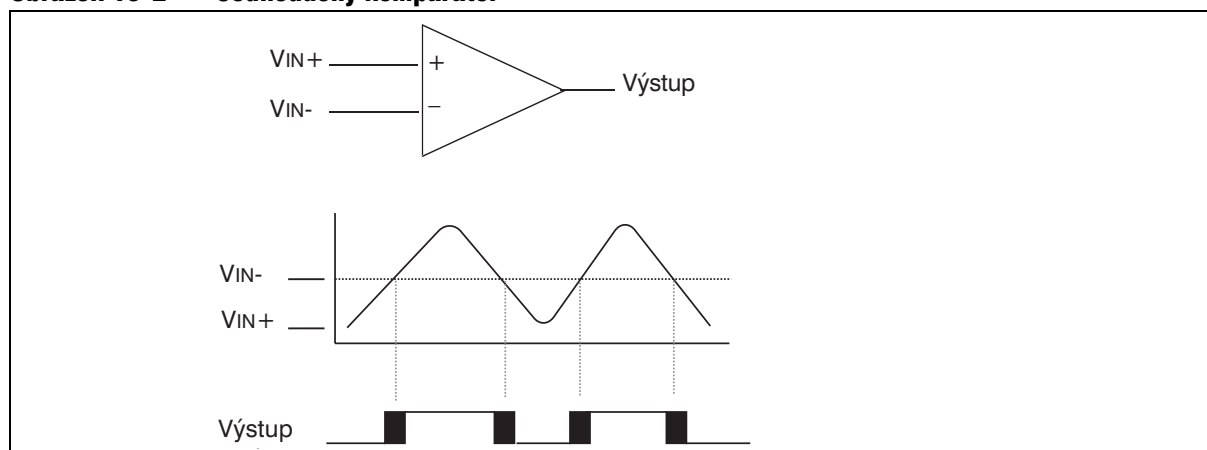
13.2 Funkce komparátoru

Schématická značka komparátoru sestávajícího z jediného operačního zesilovače je uvedena na obrázku 13-2. Dále jsou na obrázku znázorněny průběhy signálů na analogových vstupech a digitálním výstupu. Pokud je napětí na neinvertujícím vstupu V_{IN+} menší než na invertujícím vstupu V_{IN-} , výstup komparátoru je v logické '0'. Pokud je napětí na neinvertujícím vstupu V_{IN+} větší než na invertujícím vstupu V_{IN-} , výstup komparátoru je v logické '1'. K překlopení výstupu nedojde v nekonečně krátkém čase. Rovněž pokud jsou si vstupní napětí téměř rovná, nemusí být výstup komparátoru vždy stabilní. Vyznačená oblast na grafu průběhu znázorňuje dobu, po kterou nemá výstup komparátoru platnou hodnotu.

13.3 Napěťová reference komparátoru

Modul komparátoru umožňuje připojení na vstup komparátoru volitelné vnitřní napěťové reference nebo vnějšího napětí. Stav výstupu je závislý na poměru vstupních napětí, jak je uvedeno na obrázku 13-2.

Obrázek 13-2 Jednoduchý komparátor



13.3.1 Vnější napěťová reference

Při použití vnější napěťové reference mohou komparátory pracovat se společnou napěťovou referencí nebo používat dvě reference rozdílné. Velikost referenčního napětí musí být v každém případě v rozsahu napájecího napětí mikropočítače.

13.3.2 Vnitřní napěťová reference

Komparátor může pracovat i s vnitřní napěťovou referencí. Při volbě bitů $CM <2:0> = 010$ je vnitřní reference připojena na oba neinvertující vstupy komparátorů. Podrobněji je o vnitřní referenci pojednáno v kapitole 14.0.

13.4 Doba reakce komparátoru

Doba reakce komparátoru je nejkratší doba nutná k ustálení poměrů po volbě nové napěťové reference nebo po připojení nového zdroje signálu, aby byl zajištěn platný výstup dat z komparátoru. Při změně nastavení napěťové reference je nezbytné vytvořit programově zpoždění mezi touto změnou a použitím výstupu komparátoru. Je nutné použít vždy nejméně maximální dobu reakce komparátoru uvedenou v elektrických charakteristikách. Podrobněji viz kapitole 18.0.

13.5 Výstup komparátoru

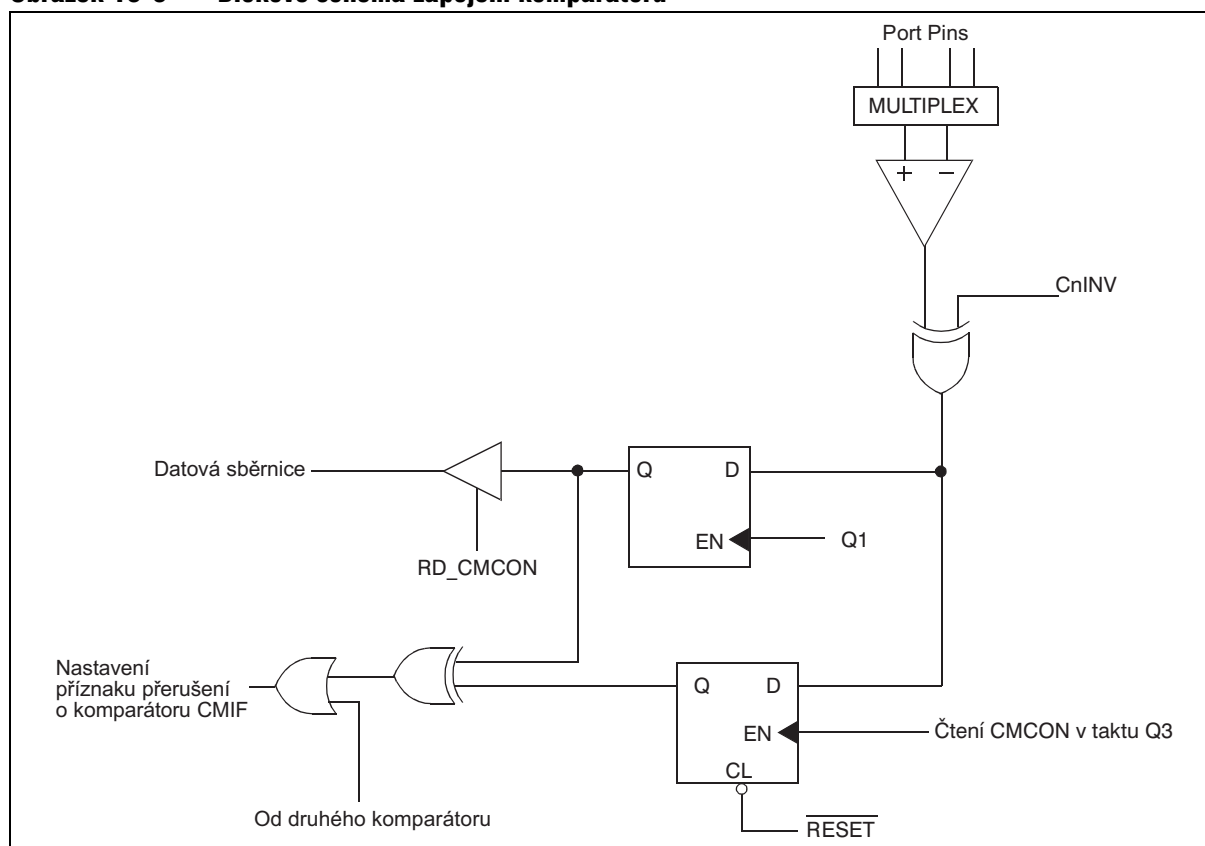
Výstupy komparátorů jsou k dispozici jako bity C1OUT a C2OUT v registru CMCON. Tyto bity je možné pouze číst, nikoli nastavovat. Tyto výstupy mohou být rovněž připojeny na vývody RA3 a RA4. V tomto případě není výstup komparátoru synchronizován s vnitřním taktom mikropočítače. Blokové schéma zapojení je uvedeno na obrázku 13-3.

Na funkci vývodů RA3 a RA4 má pochopitelně rovněž vliv nastavení registru TRISA. Pro funkci výstupu musí být v příslušných bitech registru TRISA zapsána log.0. Polarita výstupu je závislá na nastavení bitů C2INV a C1INV.

Poznámka 1: Při čtení registru PORTA jsou všechny bity odpovídající vývodům konfigurovaným jako analogové vstupy čteny vždy jako '0'. Z vývodů konfigurovaných jako digitální vstupy je analogový signál načten po ošetření Schmittovým klopným obvodem.

2: Analogové napětí přivedené na kterýkoli vývod konfigurovaný jako digitální vstup může mít za následek zvýšení vstupního proudu nad úroveň odpovídající připojení logických obvodů.

Obrázek 13-3 Blokové schéma zapojení komparátoru



13.6 Přerušení od komparátoru

Příznak přerušení od komparátoru CMIF je hardwarově nastaven vždy, když dojde ke změně na výstupu jakéhokoli komparátoru. Při práci s přerušením od komparátoru je nezbytné softwarově zjištění aktuální hodnoty bitů C1OUT a C2OUT v registru CMCON před povolením přerušení, aby bylo jasné, zda při vyvolaném přerušení od komparátoru došlo ke změně na výstupu z log.'1' do log.'0' nebo naopak. Bit CMIF, příznak přerušení od komparátoru, je v registru PIR2. Před povolením přerušení je nezbytné tento bit softwarově nulovat, stejně jako během rutiny obsluhy přerušení. Nastavením tohoto bitu do log.'1' můžeme simulovat přerušení od komparátoru.

Pro povolení přerušení je nezbytné nastavit maskovací bit přerušení od komparátoru CMIE v registru PIE2 a současně povolit přerušení od periférií nastavením bitu PEIE v registru INTCON. Následně je nutné povolit všeobecně přerušení nastavením bitu GIE. Pokud kterýkoli z uvedených bitů nebude nastaven, k přerušení od komparátoru nemůže dojít. Pokud budou splněny podmínky pro přerušení, dojde k nastavení příznaku přerušení od komparátoru CMIF i v tomto případě.

Poznámka: Pokud změna bitů C1OUT nebo C2OUT v registru CMCON nastane během instrukce čtení (počínaje začátkem druhého taktu instrukčního cyklu Q2), nemusí dojít k nastavení příznaku přerušení od komparátoru CMIF v registru PIR2.

Uživatel během rutiny obsluhy přerušení může obsloužit žádost o přerušení následujícím způsobem:

- Libovolné čtení nebo zápis do registru CMCON způsobí ukončení žádosti o přerušení od komparátoru.
- Nulování bitu CMIF ukončí příznak přerušení

Žádost o přerušení trvá po změně na výstupu komparátoru do té doby, dokud nedojde ke čtení nebo zápisu do registru CMCON. Po celou tuto dobu je také neustále nastavován příznak přerušení od komparátoru CMIF. Teprve po zapsání (čtení) do registru CMCON je hardwarově umožněno nulování příznaku CMIF.

13.7 Činnost komparátoru ve stavu Sleep

Komparátor i modul napěťové reference jsou schopny činnosti ve stavu Sleep, pokud před uvedením do stavu Sleep byly povoleny. Vedlejším účinkem činnosti obou modulů je vyšší odběr ve stavu Sleep než odpovídá specifikaci. Příslušné zvětšení odběru je popsáno odděleně v odběrových charakteristikách. Pro minimalizaci odběru ve stavu Sleep je nutné vypnout komparátor, bity CM2:CM0 = **111**. Pokud je komparátor zapnut během stavu Sleep, přerušení způsobené změnou jeho výstupu probudí mikropočítač ze stavu Sleep. Při probuzení ze stavu Sleep se obsah registru CMCON nijak nemění.

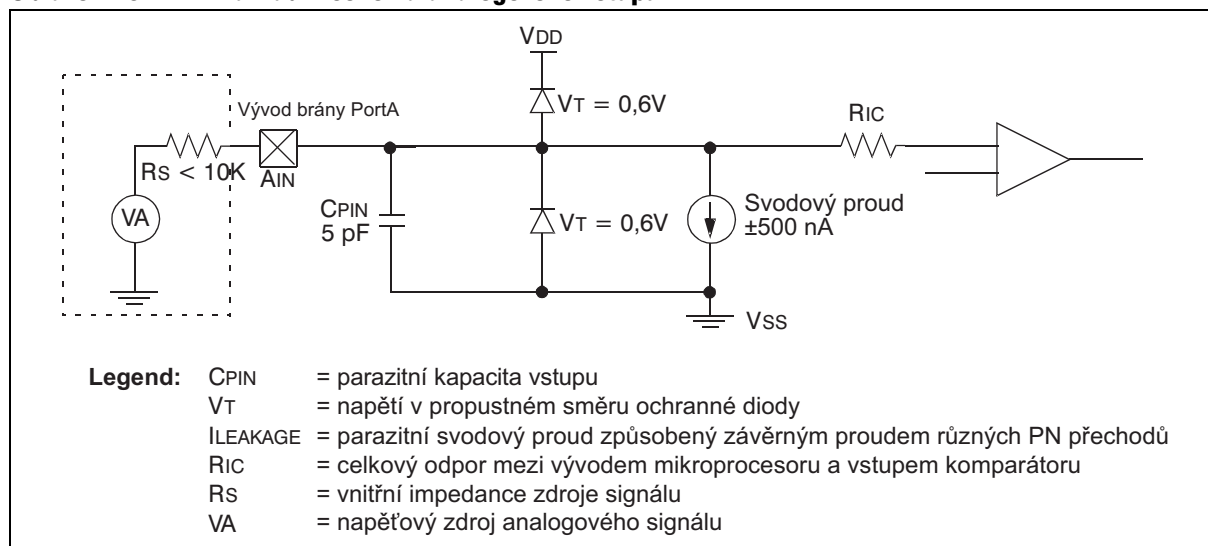
13.8 Důsledky signálu Reset pro komparátor

Signál Reset způsobí nastavení obsahu registru CMCON do inicializačních hodnot po resetu. Modul komparátoru bude nastaven do stavu CM2:CM0 = **111**.

13.9 Návrh analogového vstupu komparátoru

Zjednodušené náhradní schéma analogového vstupu komparátoru je uvedeno na obrázku 13-4. Analogový vstup je ošetřen dvojicí ochranných diod zapojených proti napájecímu napětí a zemi. Vstupní napětí musí být tedy vždy jen v rozsahu napájecího napětí. Pokud dojde k překročení tohoto rozsahu o více než 0,6 V, dojde k otevření příslušné diody a vstup bude zkratován proti příslušnému napájecímu vodiči. Vnitřní odpor zdroje signálu by neměl přesáhnout 10 kΩ. Připojení jakékoli vnější součástky jako je kondenzátor nebo Zenerova dioda způsobí vznik malého svodového proudu vytékajícího z vývodu do země.

Obrázek 13-4 Náhradní schéma analogového vstupu



Tabulka 13-1: Registry, které mají vztah ke komparátoru

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druzích resetu
9Ch	CMCON	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	0000 0111	0000 0111
9Dh	CVRCON	CVREN	CVR0E	CVRR	—	CVR3	CVR2	CVR1	CVR0	000 - 0000	000 - 0000
0Bh, 8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	0000 000x	0000 000u
0Dh	PIR2	OSFIF	CMIF	—	EEIF	—	—	—	—	00-0 ----	00-0 ----
8Dh	PIE2	OSFIE	CMIE	—	EEIE	—	—	—	—	00-0 ----	00-0 ----
05h	PORTA (PIC16F87) (PIC16F88)	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx 0000 xxx0 0000	uuuu 0000 uuu0 0000
85h	TRISA	TRISA7	TRISA6	TRISA5 (*)	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	1111 1111

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s komparátorem

Poznámka 1: Vývod 5 je pouze vstup, bit TRISA5 nemá žádný význam a jeho čtením získáme vždy log.1

PIC16F87/88

Poznámky:

14 MODUL NAPĚŤOVÉ REFERENCE

Na výstupu napěťové reference může být celkem 32 různých napěťových úrovní, 16 v nižším rozsahu a 16 ve vyšším rozsahu. Nastavení se provádí v registru bitem CVRR v registru CVRCON.

Vnitřní konstrukce modulu napěťové reference neumožňuje dosáhnout plynulého nastavení referenčního napětí v rozsahu celého napájecího napětí. Transistory na začátku a konci odporové sítě udržují CV_{REF} v závislosti na velikosti V_{DD} resp. V_{SS} . Referenční napětí je odvozeno od napájecího napětí V_{DD} a tedy každé zakolísání nebo zvlnění napájecího napětí se projeví i na napětí referenčním.

Výstup napěťové reference může být připojen na vývod RA2. To umožňuje realizaci velice jednoduchého DA převodníku nebo generování jednoduchých průběhů pro testovací účely.

Registr 14-1 Registr CVRCON – řídící registr napěťové reference (adresa 9Dh)

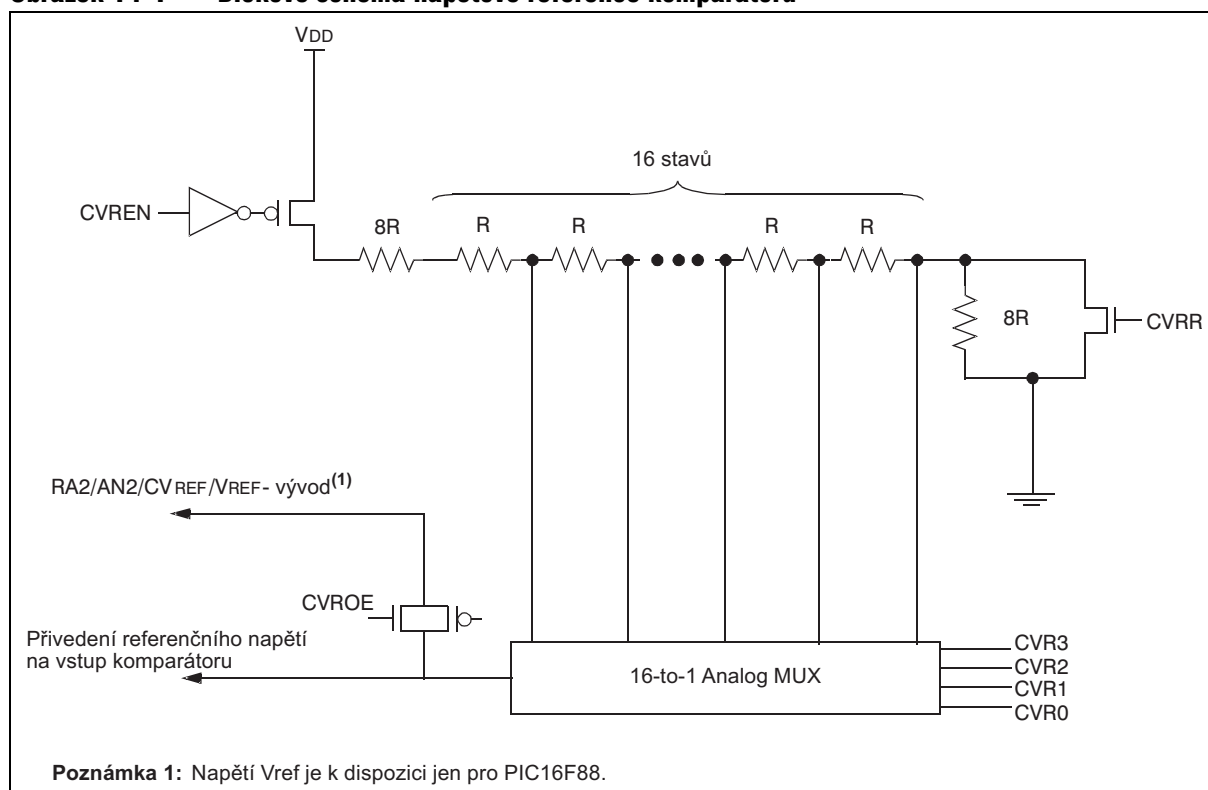
R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
CVREN	CVROE	CVRR	—	CVR3	CVR2	CVR1	CVR0
bit 7							bit 0

- bit 7 **CVREN:** povolení napěťové reference
1 = napěťová reference zapnuta
0 = napěťová reference vypnuta (není odebírán žádný proud)
- bit 6 **CVROE:** povolení výstupu na RA2
1 referenční napětí V_{REF} je přítomno na vývodu RA2
0 = referenční napětí V_{REF} není přítomno na vývodu RA2
- bit 5 **CVRR:** bit volby rozsahu napěťové reference
1 = nižší rozsah
0 = vyšší rozsah
- bit 4 **nepoužito:** při čtení '0'
- bit3-0 **CVR3:CVR0:** volba jedné ze 16 hodnot napětí ve zvoleném rozsahu
 pokud je $CVRR = 1$,
 je $CV_{REF} = V_{DD} \cdot (VR3, VR2, VR1, VR0) / 24$
 pokud je $CVRR = 0$,
 je $CV_{REF} = V_{DD} \cdot (0,25 + VR3, VR2, VR1, VR0 / 32)$

Význam:

R = bit pro čtení	W = bit pro zápis	U = nepoužitý bit, čteno jako '0'
–n = hodnota po resetu POR	'1' = bit je nastaven	'0' = bit je v nule x = hodnota bitu je neznámá

Obrázek 14-1 Blokové schéma napěťové reference komparátoru



Tabulka 14-1: Registry, které mají vztah k napěťové referenci

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Hodnota po resetu POR a BOR	Hodnota po ostatních druhých resetu
9Ch	CVRCON	CVREN	CVROE	CVRR	—	CVR3	CVR2	CVR1	CVR0	000 - 0000	000 - 0000
9Dh	CMCON	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	0000 0111	0000 0111

Vysvětlivky: — = nepoužito, při čtení "0", x = neznámý, u = nezměněný, stínované buňky nejsou použity pro práci s napěťovou referencí.

15 SPECIÁLNÍ VLASTNOSTI MIKROPOČÍTAČE

Mikropočítač je uzpůsoben pro aplikace v reálném čase díky svým speciálním vlastnostem a obvodům. Tyto obvody umožňují spolehlivý provoz v zapojení s minimem vnějších prvků s úsporným odběrem ze zdroje napájení. Jedná se o následující prvky:

Volba typu oscilátoru

- Reset
 - automatický po zapnutí napájecího napětí (POR)
 - časovač pro zpoždění resetu při zapnutí napájecího napětí (PWRT)
 - časovač pro rozběh krystalového oscilátoru (OST)
 - detektor poklesu napájecího napětí (BOD)
- Přerušení
- Hlídací časovač Watchdog pro případ kolize programu (WDT)
- Dvou rychlostní taktovací oscilátor pro rozběh
- Systém monitorování taktovací frekvence
- Režim Sleep s minimalizovanou spotřebou
- Ochrana paměti programu proti vyčtení
- Identifikační kód součástky (ID)
- Programování osazené součástky v koncové aplikaci (ICSP™)

Pro správné zahájení výkonu programu jsou k dispozici dva časovače. Časovač OST způsobuje prodlevu 1024 taktů krystalového oscilátoru a je určen k překlenutí počátečního kolísání při rozběhu. Časovač PWRT lze povolit v konfiguračním slově a způsobuje typicky zpoždění 72 ms určené pro ustálení napájecího napětí při zapnutí. Při použití těchto časovačů není nutný vnější resetovací obvod.

Režim Sleep je režim s nejmenší možnou spotřebou. Program se nevykonává, ale obsahy vnitřních registrů se nemění. Probuzení z režimu Sleep je možné buď vnějším resetem, přetečením časovače Watchdog nebo pomocí přerušení.

Další podrobnosti jsou uvedeny v literatuře DS33023 s názvem "*PICmicro® Mid-Range MCU Family Reference Manual*".

15.1 Konfigurační bity

Konfigurační slovo mikropočítače má celkem 16 bitů. V konfiguračním slově lze nastavit důležité volby pro činnost mikropočítače. Konfigurační slovo nelze měnit programem, k jeho nastavení dochází pouze při zavádění programu do součástky. Při dodání součástky jsou všechny bity nastaveny v log.1. Uživatel může tyto bity ponechat nenaprogramované, tedy v log.1 nebo je může naprogramovat, tedy vynulovat.

Konfigurační slovo je mapováno do dvou registrů v nepřístupné části paměti programu na adresách 2007h a 2008h.

PIC16F87/88

Registr 15-1 Konfigurační slovo č.1 CONFIG1 (adresa 2007h)

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
CP	CCPMX	DEBUG	WRT1	WRT0	CPD	LVP	BOREN	MCLRE	FOSC2	PWRTEN	WDTEN	FOSC1	FOSC0
bit 13													bit 0

- bit 13 **CP:** Ochrana paměti programu proti vyčtení
1 = ochrana kódu vypnuta
0 = celá paměť programu je chráněna proti vyčtení (0000h – 0FFFh)
- bit 12 **CCPMX:** Výběr vývodu pro CCP1
1 = funkce CCP1 na vývodu RB0
0 = funkce CCP1 na vývodu RB3
- bit 11 **DEBUG:** Povolení ladění v aplikaci ICD
1 = ICD je zakázáno, vývody RB6 a RB7 mají normální funkci
0 = ICD je povoleno, vývody RB6 a RB7 jsou rezervované pro debugger
- bit 10-9 **WRT1-WRT0:** Povolení zápisu do paměti programu typu Flash
11 = ochrana je vypnuta, lze zapisovat do celé paměti programu
10 = ochrana pro 0000h – 00FFh, lze zapisovat do 0100h – 1FFFh prostřednictvím EECON
01 = ochrana pro 0000h – 07FFh, lze zapisovat do 0800h – 1FFFh prostřednictvím EECON
00 = ochrana pro 0000h – 0FFFh
- bit 8 **CPD:** Nastavení ochrany proti vyčtení obsahu paměti EEPROM dat
1 = ochrana paměti dat typu EEPROM proti vyčtení je neaktivní
0 = ochrana paměti dat typu EEPROM proti vyčtení je aktivní (data nelze vyčíst)
- bit 7 **LVP:** Povolení programování nízkým napětím na vývodu $\overline{\text{MCLR}}$
1 = povoleno, vývod RB3/PGM je vyhrazen pro programování ICSP
0 = zakázáno, vývod RB3/PGM má normální funkci
- bit 6 **BOREN:** Povolení detekce poklesu napájecího napětí (BOR)
1 = povolen reset při poklesu napájecího napětí
0 = zakázán reset při poklesu napájecího napětí
- bit 5 **MCLRE:** Volba významu vývodu RA5/ $\overline{\text{MCLR}}$ /V_{PP}
1 = $\overline{\text{MCLR}}$
0 = digitální vstup/výstup, signál reset je vnitřně připojen k V_{DD}
- bit 3 **PWRTEN:** Povolení časovače pro zpoždění resetu při zapnutí napájecího napětí
1 = časovač PWRT je zakázán
0 = časovač PWRT je povolen
- bit 2 **WDTEN:** Povolení hlídacího časovače Watchdog (WDT)
1 = WDT je povolen (je nutné periodické nulování programem instrukcí CLRWDT)
0 = WDT je zakázán
- bit 4,1-0 **Fosc2 – Fosc0:** Volba typu oscilátoru
111 = oscilátor EXTRC s výstupem na vývodu RA6/OSC2/CLKO
110 = oscilátor EXTRC, vývod RA6/OSC2/CLKO je univerzální I/O
101 = vnitřní oscilátor INTRC s výstupem na RA6/OSC2/CLKO, vývod RA7 je univerzální I/O
100 = vnitřní oscilátor INTRC, vývod RA7 a RA6 jsou univerzální I/O
011 = oscilátor ECIO, vývod RA6 je univerzální I/O
010 = HS oscilátor
001 = XT oscilátor
000 = LP oscilátor

Význam:

R = bit pro čtení

W = bit pro zápis

U = nepoužitý bit, čteno jako '0'

–n = hodnota po resetu POR

'1' = bit je nastaven

'0' = bit je v nule

x = hodnota bitu je neznámá

PIC16F87/88

Registr 15-2 Konfigurační slovo č.2 CONFIG2 (adresa 2008h)

U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1	R/P-1	R/P-1
—	—	—	—	—	—	—	—	—	—	—	—	IESO	FCMEN
bit 13												bit 0	

bit 13-2 **nepoužito:** čteno jako '1'

bit 1 **IESO:** Povolení přepnutí vnitřního a vnějšího oscilátoru

1 = povoleno
0 = zakázáno

bit 0 **FCMEN:** Povolení systému monitorování taktovací frekvence

1 = povoleno
0 = zakázáno

Význam:

R = bit pro čtení

–n = hodnota po resetu POR

W = bit pro zápis

'1' = bit je nastaven

U = nepoužitý bit, čteno jako '0'

'0' = bit je v nule

x = hodnota bitu je neznámá

15.2 Reset

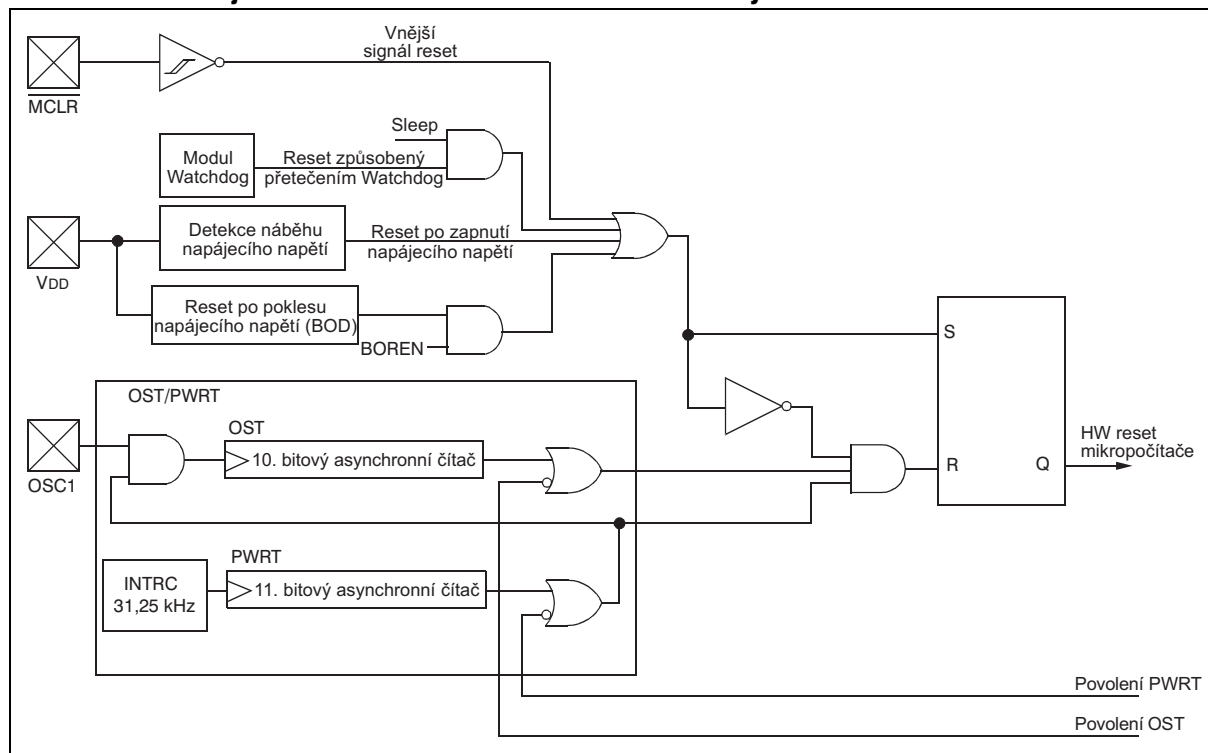
Mikroprocesor PIC16F87/88 rozlišuje mezi několika typy Resetů:

- Reset při zapnutí napájecího napětí (Power-on Reset – POR)
- Vnější Reset na vstupu $\overline{\text{MCLR}}$ během výkonu programu
- Vnější Reset na vstupu $\overline{\text{MCLR}}$ v režimu Sleep
- Reset způsobený ochranným čítačem Watchdog během výkonu programu
- Probuzení ze stavu Sleep prostřednictvím přetečení časovače Watchdog
- Reset způsobený poklesem napájecího napětí detekovaný obvodem BOD

Hodnoty některých registrů se nijak nemění při žádném z uvedených druhů Resetu. Jejich obsah je po Resetu POR náhodný a po všech ostatních druzích Resetu beze změny. Řada dalších registrů je nastavena v závislosti na druhu Resetu do počátečního stavu. Jediný druh Resetu, který zaručeně nemodifikuje žádné pracovní registry je Probuzení ze stavu Sleep prostřednictvím přetečení časovače Watchdog. Tento druh Resetu je totiž považován za přirozené pokračování běhu programu po ukončení režimu Sleep. Bity stavového registru (STATUS) $\overline{\text{TO}}$ a $\overline{\text{PD}}$ jsou nastavovány resetovacím systémem v závislosti na druhu posledního Resetu. Podle jejich stavu lze určit druh posledního Resetu. Možné kombinace jsou uvedeny v tabulce 15-4.

Blokové schéma zapojení obvodů resetu je uvedeno na obrázku 15-1.

Obrázek 15-1 Zjednodušené blokové schéma resetovacího systému



15.3 Vývod $\overline{\text{MCLR}}$

Vnější vstup $\overline{\text{MCLR}}$ je opatřen vstupním ochranným obvodem, který eliminuje případné nežádoucí krátkodobé pulsní rušení. Požadované parametry signálu $\overline{\text{MCLR}}$ včetně časových průběhů jsou uvedeny v zadní části této dokumentace.

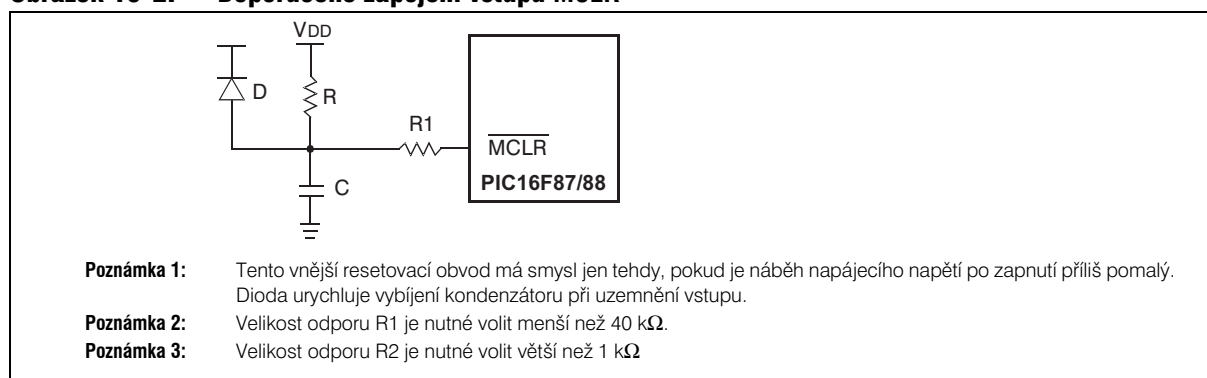
Vnitřní reset způsobený přetečením časovače Watchdog nezpůsobí pokles napětí na vývodu $\overline{\text{MCLR}}$.

Tento mikropočítač má vyřešený problém s ošetřením vývodu $\overline{\text{MCLR}}$ jinak než dřívější typy mikropočítačů. Doporučené zapojení vývodu $\overline{\text{MCLR}}$ je uvedeno na obrázku 15-2.

Poznámka: Vývod $\overline{\text{MCLR}}$ je nutné připojit k napájecímu napětí co možná nejkratší cestou.

Vývod RA5/ $\overline{\text{MCLR}}$ /VPP může být konfigurován jako vstup pro reset nebo jako univerzální vstup/výstup.

Obrázek 15-2: Doporučené zapojení vstupu $\overline{\text{MCLR}}$



15.2 Reset při zapnutí napájecího napětí (Power-on Reset – POR)

Tento druh resetu je generován tehdy, dosáhne-li narůstající napájecí napětí VDD rozhodovací úrovně, která leží v rozsahu od 1,2 V do 1,7 V. Podrobněji je pojednáno v kapitole 15.3 a 18.0.

Po ukončení signálu Reset začne mikropočítač vykonávat první instrukci v paměti programu. V této době již musí být dosaženo předepsaných hodnot napájecího napětí, taktovací frekvence, teploty a dalších parametrů. Pokud tyto podmínky nejsou splněny, je nezbytné prodloužit signál Reset do doby, než dojde k jejich splnění. V opačném případě by nebyla zaručena správná funkce mikropočítače. Typickým jevem při zapnutí napájecího napětí je jeho nelineární náběh a kolísání, související s odezníváním přechodových jevů celého napájeného obvodu.

Podrobněji je o problémech s napájecím napětím pojednáno v Aplikační příručce s označením AN607, v části nazvané “Power-up Trouble Shooting”.

15.5 Časovač pro zpoždění Resetu (Power-up Timer – PWRT)

Časovač pro zpoždění signálu Reset zabezpečí pevnou prodlevu 72 ms od okamžiku, kdy je obvod POR schopen reset generovat. Tato pevná doba (T_{PWRT}) je odvozena od nezávislého RC oscilátoru umístěného přímo na čipu mikropočítače. Účelem tohoto zpoždění je umožnit napájecímu napětí dosáhnout provozní hodnoty, případně překlenout dobu kolísání napájecího napětí bezprostředně po zapnutí. Po celou dobu, kdy je aktivní časovač PWRT, je signál Reset trvale v aktivním stavu (program se nevykonává).

Povolit nebo zakázat časovač PWRT lze pomocí bitu $\overline{\text{PWRTÉ}}$ v konfiguračním slově procesoru během zavádění programu do součástky. Hodnota 72 ms udávaná pro časovač PWRT je závislá nejen na teplotě, napájecím napětí VDD, výrobní toleranci, ale liší se i u každé vyrobené součástky. Bližší informace jsou uvedené v elektrických parametrech v zadní části této dokumentace (viz parametr #33).

15.6 Časovač pro rozběh oscilátoru (Oscillator Start-up Timer – OST)

Tento časovač je tvořen pevným desetibitovým čítačem, který je buzen ze vstupu OSC1. Jeho úkolem je po resetu POR, případně po zpoždění 72 ms zabezpečeném časovačem PWRT, podržet Reset aktivní po dobu 1024 taktů taktovacího oscilátoru počítače. Důvodem pro toto zpoždění je nutnost ustálení krystalového oscilátoru nebo rezonátoru. Doba trvání (T_{OST}) těchto 1024 cyklů je pochopitelně závislá na taktovací frekvenci.

Časovač OST je zařazen pouze při typu oscilátoru XT, LP nebo HS, pokud došlo k resetu POR nebo probuzení ze stavu Sleep.

15.7 Detekce poklesu napájecího napětí (BOR)

V konfiguračním slově lze zakázat nebo povolit činnost obvodu BOD. Jedná se o detektor poklesu napájecího napětí. Pokud dojde k poklesu napájecího napětí pod hodnotu V_{BOR} (zpravidla 4 V) na dobu delší než T_{BOR} (zpravidla 100 μ s) dojde k resetu mikropočítače BOR (viz parametr #33).

Jakmile byl aktivován reset BOR, je mikropočítač udržován ve stavu reset dokud napájecí napětí nestoupne na požadovanou úroveň. Časovač PWRT může dále vyvolat časovou prodlevu 72 ms.

Narozdí od předchozích mikropočítačů řady 16Fxx není časovač PWRT automaticky povolen vždy, pokud je povolen obvod BOD. Bity $\overline{PWRTEN}$ a $\overline{BOREN}$ jsou na sobě nezávislé.

15.8 Zapnutí napájecího napětí

Po zapnutí napájecího napětí je generován reset POR. Dále je aktivován časovač PWRT (pokud je v konfiguračním slově povolen). Při oscilátorech LP, XT a HS je aktivován časovač OST. Teprve poté je aktivován hardwarový reset mikropočítače.

Pokud je vstup $\overline{MCLR}$ držen v nule dostatečně dlouho, dojde k vyčerpání všech časových zpoždění a po uvolnění tohoto vstupu dojde k zahájení výkonu programu okamžitě. To je výhodné zejména pro testovací účely nebo pro synchronizaci více mikropočítačů.

V tabulce 15-1 jsou uvedeny obsahy některých registrů po různých druzích resetu.

15.9 Příznakový registr resetu PCON

Příznakový registr PCON obsahuje dva funkční bity. Podle těchto bitů lze určit poslední reset mikropočítače. Bit $\overline{BOR}$ je na pozici b0. Hodnota tohoto bitu je po resetu POR neznámá. Uživatel musí tento bit nastavit do log.1 a testovat při každém resetu. Hodnota 0 v tomto bitu znamená, že došlo k resetu způsobeném obvodem BOR. Pokud je obvod BOR zakázán, je stav bitu $\overline{BOR}$ nedefinovaný.

Bit POR je na pozici b1. Tento bit je nulován při resetu POR a při ostatních druzích resetu není nijak ovlivňován. Po resetu POR musí uživatel tento příznak nastavit do log. 1.

Tabulka 15-1: Prodleva při resetu v různých situacích

Konfigurace oscilátoru	Zapnutí napájení		Výpadek napájecího napětí (BOR)		Probuzení ze stavu Sleep
	$\overline{PWRTEN} = 0$	$\overline{PWRTEN} = 1$	$\overline{PWRTEN} = 0$	$\overline{PWRTEN} = 1$	
XT, HS, LP	$T_{PWRT} + 1024 \cdot T_{OSC}$	$1024 \cdot T_{OSC}$	$T_{PWRT} + 1024 \cdot T_{OSC}$	$1024 \cdot T_{OSC}$	$1024 \cdot T_{OSC}$
EXTRC, INTRC	T_{PWRT}	5 – 10 μ s ⁽¹⁾	T_{PWRT}	5 – 10 μ s ⁽¹⁾	5 – 10 μ s ⁽¹⁾
T1OSC	—	—	—	—	5 – 10 μ s ⁽¹⁾

Poznámka 1: Zahájení výkonu programu je vždy vyvoláno resetem POR, BOR nebo probuzením ze stavu Sleep. Zpoždění 5 – 10 μ s odpovídá taktovací frekvenci 1 MHz.

Tabulka 15-2: Stavové bity resetu a jejich význam

$\overline{POR}$	$\overline{BOR}$	$\overline{TO}$	$\overline{PD}$	Podmínka
0	x	1	1	Reset po zapnutí napájecího napětí (POR)
0	x	0	x	Zakázaný stav, $\overline{TO}$ je nastaven při POR
0	x	x	0	Zakázaný stav, $\overline{PD}$ je nastaven při POR
1	0	1	1	Reset způsobený výpadkem napájecího napětí (BOR)
1	1	0	1	Reset způsoben přetečením časovače Watchdog během výkonu programu
1	1	0	0	Probuzení ze stavu Sleep přetečením časovače Watchdog
1	1	u	u	Reset způsobený aktivací vývodu $\overline{MCLR}$ během výkonu programu
1	1	1	0	Probuzení ze stavu Sleep aktivací vývodu $\overline{MCLR}$ nebo pomocí přerušení

Význam: u = beze změny, x = stav není znám

Tabulka 15-3: Počáteční stav po Resetu pro některé speciální funkční registry.

Podmínka	Programový čítač PC	Stavový registr STATUS	Príznakový registr PCON
Reset po zapnutí napájecího napětí (POR)	000h	0001 1xxx	---- --0x
Reset způsobený aktivací vývodu MCLR během výkonu programu	000h	000u uuuu	---- --uu
Probuzení ze stavu Sleep aktivací vývodu MCLR	000h	0001 0uuu	---- --uu
Reset způsobený přetečením časovače Watchdog během výkonu programu	000h	0000 1uuu	---- --uu
Probuzení ze stavu Sleep přetečením časovače Watchdog	PC + 1	uuu0 0uuu	---- --uu
Reset způsobený výpadkem napájecího napětí (BOR)	000h	0001 1uuu	---- --u0
Probuzení ze stavu Sleep pomocí přerušení	PC + 1 (1)	uuu1 0uuu	---- --uu

Význam: u = bez změny, x = náhodný stav, – = nepoužito, při čtení 0

Poznámka 1: Pokud dojde k probuzení ze stavu Sleep prostřednictvím přerušení které je povoleno (GIE=1), do PC je uložen vektor přerušení (0004h)

Tabulka 15-4: Počáteční stav po Resetu pro všechny speciální funkční registry

Registr	Reset po zapnutí napájecího napětí (POR) nebo výpadkem napájecího napětí (BOR)	Reset způsobený aktivací vývodu MCLR nebo přetečením Watchdog během výkonu programu	Probuzení ze stavu Sleep přetečením časovače Watchdog nebo přerušením
W	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	N/A	N/A	N/A
TMR0	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	0000h	0000h	PC + 1 (2)
STATUS	0001 1xxx	000q quuu	uuuq quuu
FSR	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTA (PIC16F87)	xxxx 0000	uuuu 0000	uuuu uuuu
PORTA (PIC16F88)	xxx0 0000	uuu0 0000	uuuuu uuuuu
PORTA (PIC16F87)	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTA (PIC16F88)	00xx xxxx	00uu uuuu	uuuu uuuu
PCLATH	---0 0000	---0 0000	---u uuuu
INTCON	0000 000x	0000 000u	uuuu uuuu
PIR1	-000 0000	-000 0000	-uuu uuuu
PIR2	00-0 ----	00-0 ----	uu-u ----
TMR1L	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR1H	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1CON	-000 0000	-uuu uuuu	-uuu uuuu
TMR2	0000 0000	0000 0000	uuuu uuuu
T2CON	-000 0000	-000 0000	-uuu uuuu
SSPBUF	xxxx xxxx	uuuu uuuu	uuuu uuuu
SSPCON	0000 0000	0000 0000	uuuu uuuu
CCPR1L	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR1H	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP1CON	--00 0000	--00 0000	--uu uuuu
RCSTA	0000 000x	0000 000x	uuuu uuuu

Význam: u = bez změny, x = náhodný stav, — = nepoužito, při čtení 0, q = hodnota závisí na podmínkách

Poznámka 1: Jeden nebo více bitů v registrech INTCON, PIR1 a PIR2 mohou být ovlivněny

2: Pokud dojde k probuzení ze stavu SLEEP prostřednictvím přerušení které je povoleno (GIE=1), do PC je uložen vektor přerušení (0004h)

3: Počáteční stavy některých registrů jsou v tabulce 15-3.

Tabulka 15-4: Počáteční stav po Resetu pro všechny speciální funkční registry (pokračování)

Registr	Reset po zapnutí napájecího napětí (POR) nebo výpadkem napájecího napětí (BOR)	Reset způsobený aktivací vývodu MCLR nebo přetečením Watchdog během výkonu programu	Probuzení ze stavu Sleep přetečením časovače Watchdog nebo přerušením
TXREG	0000 0000	0000 0000	uuuu uuuu
RCREG	0000 0000	0000 0000	uuuu uuuu
ADRESH	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON0	0000 00-0	0000 00-0	uuuu uu-u
OPTION_REG	1111 1111	1111 1111	uuuu uuuu
TRISA	1111 1111	1111 1111	uuuu uuuu
TRISB	1111 1111	1111 1111	uuuu uuuu
PIE1	-000 0000	-000 0000	-uuu uuuu
PIE2	00-0 ----	00-0 ----	uu-u ----
PCON	---- --0q	---- --uu	---- --uu
OSCCON	-000 0000	-000 0000	-uuu uuuu
OSCTUNE	--00 0000	--00 0000	--uu uuuu
PR2	1111 1111	1111 1111	1111 1111
SSPADD	0000 0000	0000 0000	uuuu uuuu
SSPSTAT	0000 0000	0000 0000	uuuu uuuu
TXSTA	0000 -010	0000 -010	uuuu -u1u
SPBRG	0000 0000	0000 0000	uuuu uuuu
ANSEL	-111 1111	-111 1111	-111 1111
CMCON	0000 0111	0000 0111	uuuu u111
CVRCON	000- 0000	000- 0000	uuu- uuuu
WDTCON	---0 1000	---0 1000	---u uuuu
ADRESL	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON1	0000 ----	0000 ----	uuuu ----
EEDATA	xxxx xxxx	uuuu uuuu	uuuu uuuu
EEADR	xxxx xxxx	uuuu uuuu	uuuu uuuu
EEDATH	--xx xxxx	--uu uuuu	--uu uuuu
EEADRH	---- -xxx	---- -uuu	---- -uuu
EECON1	x--x x000	u--x u000	u--u uuuu
EECON2	---- ----	---- ----	---- ----

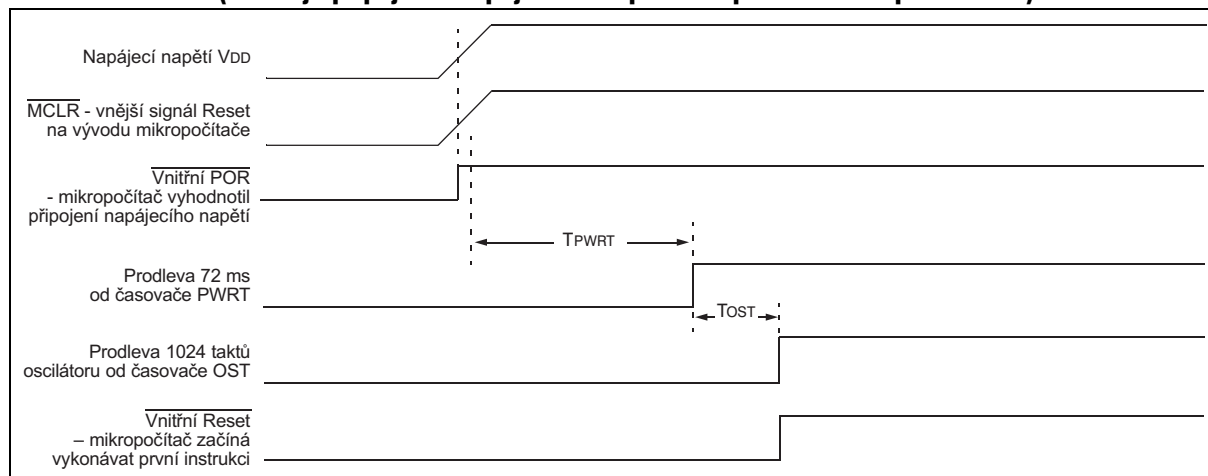
Význam: u = bez změny, x = náhodný stav, — = nepoužito, při čtení 0, q = hodnota závisí na podmínkách

Poznámka 1: Jeden nebo více bitů v registrech INTCON, PIR1 a PIR2 mohou být ovlivněny

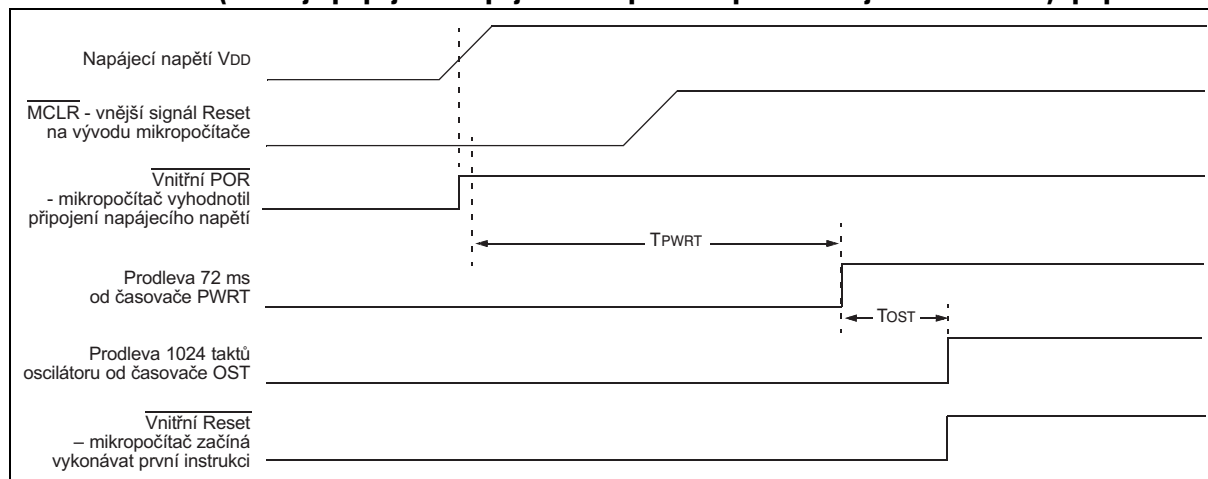
2: Pokud dojde k probuzení ze stavu Sleep prostřednictvím přerušení které je povolené (GIE=1), do PC je uložen vektor přerušení (0004h)

3: Počáteční stavy některých registrů jsou v tabulce 15-3.

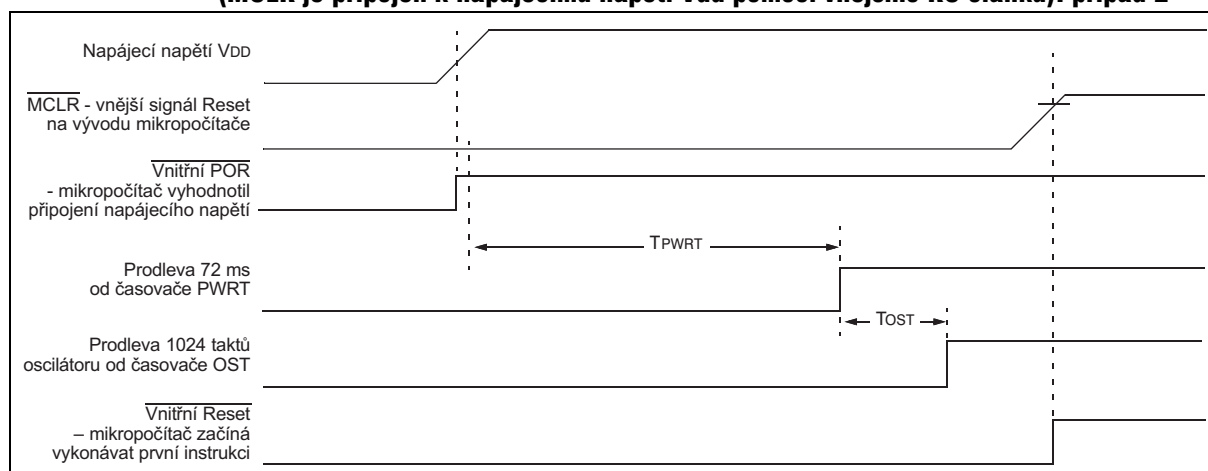
**Obrázek 15-3: Časové průběhy při zapnutí napájecího napětí
(MCLR je připojen k napájecímu napětí Vdd pomocí Pull-up rezistoru.)**



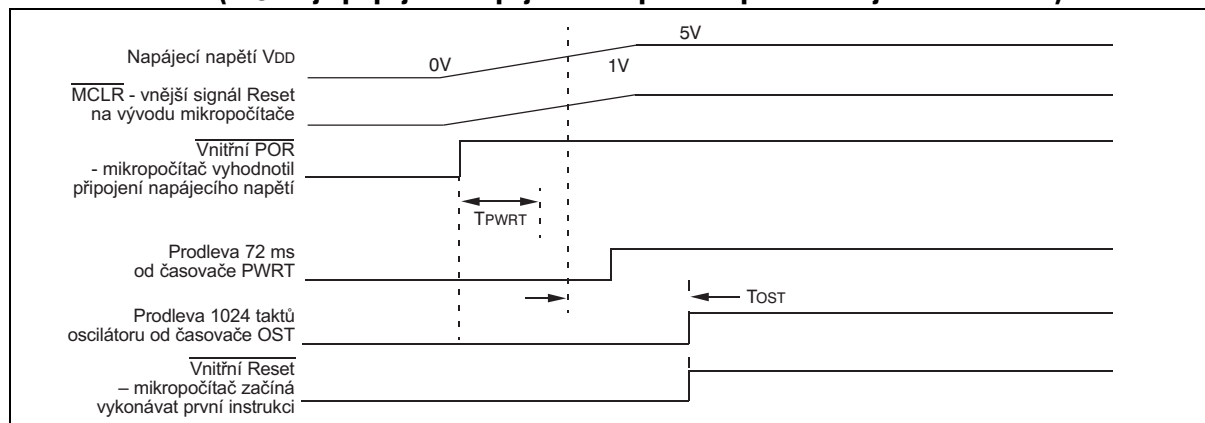
**Obrázek 15-4: Časové průběhy při zapnutí napájecího napětí
(MCLR je připojen k napájecímu napětí Vdd pomocí vnějšího RC článku): případ 1**



**Obrázek 15-5: Časové průběhy při zapnutí napájecího napětí
(MCLR je připojen k napájecímu napětí Vdd pomocí vnějšího RC článku): případ 2**



Obrázek 15-6: Časové průběhy při zapnutí napájecího napětí – pomalý nárůst VDD (MCLR je připojen k napájecímu napětí VDD pomocí vnějšího RC článku)



15.10 Přerušení

Mikroprocesor PIC16F87/88 má celkem 12 druhů přerušení. V řídicích registrech jsou příznakové bity, pomocí kterých lze určit k jakému druhu přerušení došlo. Tyto bity mají význam žádosti o přerušení (flag bit). V těchto registrech jsou i bity pro maskování (individuální zákaz) příslušných přerušení.

Poznámka: Příznakové bity jsou obvody mikroprocesoru nastaveny vždy, pokud jsou splněny podmínky vzniku přerušení a to i v případě, že přerušení je globálně zakázáno ($GIE=0$), nebo je maskováno konkrétní přerušení.

Přerušení lze zakázat globálně, tedy pro všechny čtyři zdroje, nebo lze globálně povolit nemaskovaná (individuálně povolená) přerušení nastavením bitu GIE (Global Interrupt Enable) v registru $INTCON(7)$. Nastavením do log "1" jsou všechna přerušení povolena. Pokud chceme zakázat konkrétní zdroj přerušení, je nutné maskovat pomocí odpovídajících bitů v různých registrech. Po signálu Reset je přerušení vždy globálně zakázáno ($GIE=0$).

Pro návrat z obsluhy přerušení lze použít instrukci **RETFIE**. Tato instrukce kromě řádného ukončení podprogramu a předání řízení na místo, odkud byl podprogram volán navíc povolí další přerušení ($GIE=1$).

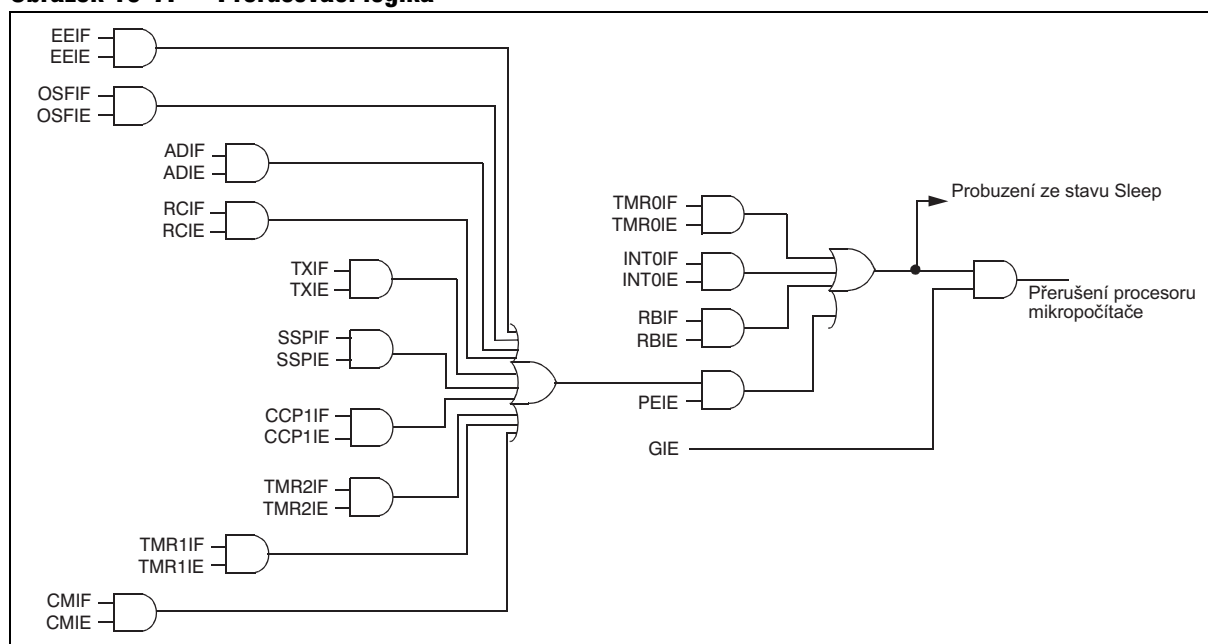
Příznakové bity přerušení na vývodu RBO , změnou na bráně PortB a přetečením čítače/časovače Timer0 jsou v registru $INTCON$.

Příznakové bity dalších přerušení od periférií jsou v registru $PIR1$. Odpovídající bity pro maskování přerušení jsou v registru $PIE1$. Bit povolující všechna přerušení od periférií je v registru $INTCON$.

Pokud přijde přerušení, které je povoleno, je okamžitě zakázáno jakékoli další přerušení ($GIE=0$). Obsah čítače instrukcí PC je uložen do zásobníku (STACK), čímž dojde k uchování návratové adresy. Dále je do čítače instrukcí PC uložen vektor přerušení (hodnota 0004h). Doba reakce na vnější přerušení je od třech do čtyř instrukčních cyklů. Přesná hodnota závisí na strojovém taktu, během kterého k žádosti došlo. Doba reakce na vnější přerušení nezávisí na tom, zda k žádosti došlo během výkonu instrukce trvající jeden nebo dva instrukční cykly. Pokud podprogram obsluhy přerušení obsluhuje více než jeden přerušovací zdroj, je nutné určit druh přerušení testováním příznakových bitů v registru $INTCON$. Tyto příznakové bity je nutné následně v rutině obsluhy přerušení softwarově vynulovat.

Při vnějších přerušeních na vývodu RBO nebo změnou na bráně PortB je zahájení obsluhy přerušení pozdrženo o 3 – 4 instrukční cykly. Přesná hodnota zpoždění závisí na konkrétních podmínkách. Zpoždění nezávisí na tom, zda k žádosti o přerušení dojde během instrukce trvající jeden nebo dva instrukční cykly. Příznakové bity přerušení se nastavují nezávisle na stavu příslušných maskovacích bitů.

Obrázek 15-7: Přerušovací logika



15.10.1 Vnější přerušování na vývodu RB0/INT

Vnější přerušování je spouštěno aktivní hranou signálu na vývodu RB0/INT. Pokud má být aktivní náběžná hrana tohoto signálu, je nutné nastavit řídicí bit (INTEDG) v registru OPTION_REG,6 do log"1". Pokud má být aktivní sestupná hrana, je nutné tento řídicí bit vynulovat. S příchodem aktivní hrany signálu na RB0/INT je vždy nastaven příznakový bit v registru INTCON,1 (INT0IF=1). Přerušování může být maskováno (zakázáno) nulováním řídicího bitu INTOIE v registru INTCON,4. Příznakový bit INT0IF musí být rutinou obsluhy přerušování nulován vždy před povolením dalšího přerušování. Přerušování lze probudit mikropočítač ze stavu Sleep za předpokladu, že před uvedením do stavu Sleep bude řídicí bit INTOIE nastaven do log"1". Pokud bude při probuzení povoleno globálně přerušování (GIE=1), dojde ke spuštění obsluhy přerušování na adrese 0004h. V opačném případě se bude pokračovat instrukcí, následující po instrukci SLEEP. Podrobněji je o stavu Sleep pojednáno v **části 15.13**.

15.10.2 Vnitřní přerušování přetečením čítače TMR0

Přetečení čítače/časovače TMR0 (FFh – 00h) způsobí nastavení příznakového bitu TMR0IF v registru INTCON,2. Přerušování může být povoleno nastavením řídicího bitu TMR0IE v registru INTCON,5 do log"1", nebo maskováno (zakázáno) nulováním tohoto bitu. Podrobněji je pojednáno v **kapitole 6.0**.

15.10.3 Vnější přerušování změnou na 4 bitech brány PORTB

Jakákoli změna stavu na vstupních bitech brány Portb (RB7–RB4) způsobí nastavení příznakového bitu RBIF v registru INTCON,0 (na uvedených 4 bitech lze vytvořit celkem 16 různých stavů). Přerušování může být povoleno nastavením řídicího bitu RBIE v registru INTCON,4 do log"1", nebo maskováno (zakázáno) nulováním tohoto bitu. Podrobněji je o využití tohoto přerušování pojednáno v **kapitole 3.2**.

15.11 Uložení obsahu registrů během přerušení

Během obsluhy přerušení se do zásobníku (STACK) ukládá pouze návratová adresa, tedy obsah programového čítače PC. Pokud uživatel požaduje uložení pracovních registrů, nejčastěji se jedná o střadač (W) nebo stavový registr (STATUS), musí jejich uložení (PUSH) a obnovení (POP) zabezpečit jako součást rutiny obsluhy přerušení. Hodnoty registrů se pochopitelně neukládají do zásobníku (STACK), ale do určené oblasti paměti RWM. Příklad možného řešení programové rutiny je uveden na příkladu 15-1.

Příklad 15-1: Uložení obsahu registru W a STATUS do RWM

MOVWF	WTEMP	;uložení střadače do záložního registru
SWAPF	STATUS, W	;načtení obsahu registru STATUS do střadače
		; (s přemístěním horních a dolních 4 bitů)
CLRF	STATUS	;banka 0 nezávisle na aktuální bance
MOVWF	STATUS_TEMP	;uložení obsahu registru STATUS
		;do záložního registru
MOVF	PCLATH, W	;požadováno jen při použití banky 1
MOVWF	PCLATH_TEMP	;uložení PCLATH do W
CLRF	PCLATH	;stránka 0, nezávisle na stránce paměti programu
:		
:(ISR)		;rutina obsluhy přerušení,
		;která může libovolně měnit volbu banky registrů
:		
MOVF	PCLATH_TEMP, W	;obnova PCLATH
MOVWF	PCLATH	;načtení střadače do PCLATH
SWAPF	STATUS_TEMP, W	;přepnutí STATUS_TEMP
		;obnova nastavení Banky
MOVWF	STATUS	;načtení záložního registru s původním STATUS
		; (s přemístěním horních a dolních 4 bitů)
SWAPF	W_TEMP, F	;přemístění horních a dolních 4 bitů
		;v registru W_TEMP
SWAPF	W_TEMP, W	;obnova (POP) pracovního registru W

PIC16F87/88

Registr 15-3: Řídící registr časovače Watchdog – WDTCON (adresa 105h)

U-0	U-0	U-0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-0
—	—	—	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN (1)
bit 7			bit 0				

bit 7-5 **nepoužito:** čteno jako '0'

bit 4-1 **WDTPS3:WDTPS0:** volba programovatelné předděličky

kombinace	=	dělicí poměr
0000	=	1:32
0001	=	1:64
0010	=	1:128
0011	=	1:256
0100	=	1:512
0101	=	1:1024
0110	=	1:2048
0111	=	1:4096
1000	=	1:8192
1001	=	1:16384
1010	=	1:32768
1011	=	1:65536

bit 0 **SWDTEN:** programové povolení časovače Watchdog

1 = povoleno
0 = zakázáno

Poznámka: Pokud je časovač Watchdog povolen v konfiguračním slově, je povolen vždy nezávisle na nastavení bitu SWDTEN

Význam:

R = bit pro čtení W = bit pro zápis U = nepoužitý bit, čteno jako '0'
-n = hodnota po resetu POR '1' = bit je nastaven '0' = bit je v nule x = hodnota bitu je neznámá

Tabulka 15-6: Registry, které mají vztah k časovači Watchdog

Adresa	Název	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
81h, 181h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
2007h	Konfigurační bity	LVP	BOREN	MCLRE	FOSC2	PWRTEN	WDTEN	FOSC1	FOSC0
105h	WDTCON	—	—	—	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN

Vysvětlivky: stínované buňky nejsou použity pro práci s časovačem Watchdog

Poznámka 1: Práce s těmito bity je popsána v kapitole 15-1

15.12.3 Dvourychlostní rozběh taktovacího oscilátoru

Použití dvourychlostního rozběhu taktovacího oscilátoru lze povolit nebo zakázat v konfiguračním slově č.2 pomocí bitu IESO. Aktivace tohoto systému umožňuje zkrátit dobu mezi zapnutím napájecího napětí a zahájením výkonu programu. V tomto režimu je jako taktovací oscilátor použit nejprve vnitřní oscilátor INTRC, teprve po náběhu a ustálení primárního oscilátoru je taktovací frekvence přepnuta.

Pokud je tento režim povolen, bude použit pro rozběh vnitřní oscilátor INTRC vždy, pokud bude splněna jedna z následujících podmínek:

- Reset POR a přetečení časovače PWRT (pokud je bit $\overline{\text{PWRTEN}} = 0$)
- nebo následně po probuzení ze stavu Sleep
- nebo při resetu během výkonu programu při taktování pomocí T1OSC nebo INTRC (po resetu jsou bity SCS1 a SCS0 vždy nulovány)

Poznámka: Po jakémkoli resetu jsou bity IRFC nulovány a frekvence vnitřního oscilátoru je stanovena na 31,25 kHz. Uživatel může programově modifikovat tyto bity a zajistit tak vyšší taktovací frekvenci.

Pokud je jako primární taktovací oscilátor konfigurován jakýkoli jiný oscilátor než XT, LP nebo HS, je dvou rychlostní rozběh nepoužitelný. Ostatní typy oscilátorů totiž nepotřebují žádnou časovou prodlevu od resetu nebo probuzení ze stavu Sleep do ustálení oscilací.

Pokud jsou před uvedením do režimu Sleep bity IRFC v registru OSCCON nastaveny na jiné než nulové hodnoty, je zdrojem taktovací frekvence výstup vnitřního oscilátoru INTOSC. Bity IOFC v registru OSCCON budou nulovány, dokud je INTOSC stabilní. To umožní uživateli určit, kdy bude použit vnitřní oscilátor pro časově kritické části programu.

Testováním bitu OSTs lze zjistit, zda je v činnosti primární oscilátor. Pokud není, bit OSTs zůstává v nule.

Pokud je mikropočítač autonomně přepnut do režimu INTRC následně po resetu POR nebo po probuzení ze stavu Sleep, pravidla pro ovládání oscilátorů zůstávají v platnosti. Bity SCS0 a SCS1 v registru OSCCON lze modifikovat před přetečením časovače OST. To může umožnit probuzení mikropočítače ze stavu Sleep, vykonání několika instrukcí a opětovné uvedení do stavu Sleep, aniž by se muselo čekat na aktivaci primárního oscilátoru.

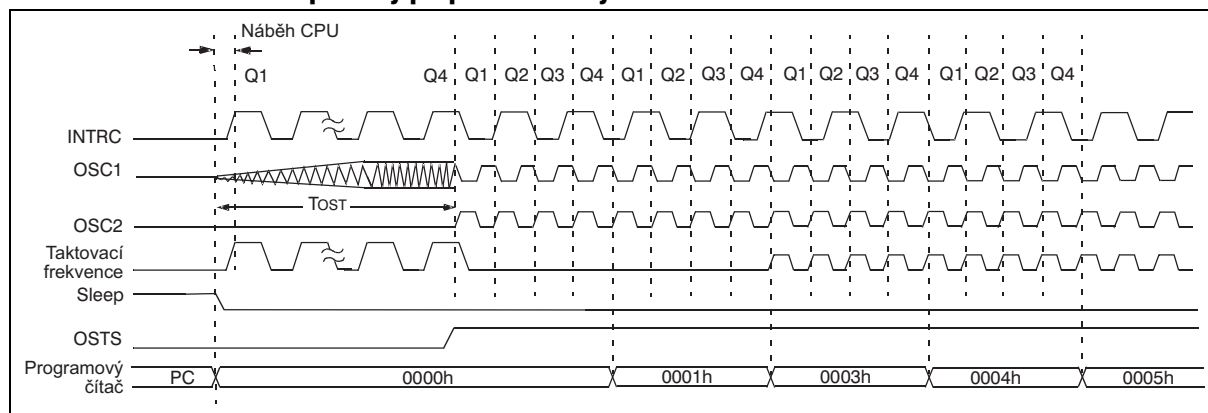
Poznámka: Vykonání instrukce Sleep ukončí předčasně činnost časovače pro rozběh oscilátoru a vynuluje bit OSTs.

15.12.3.1 Postup při použití dvourychlostního rozběhu taktovacího oscilátoru

1. Probuzení ze stavu Sleep, reset nebo POR.
2. Registr OSCCON je konfigurován pro INTRC (31,25 kHz).
3. Zahájení výkonu programu při taktu 31,25 kHz.
4. Časovač OST odpočítává 1024 taktů.
5. Po přetečení časovače OST se čeká na sestupnou hranu INTRC.
6. Bit OSTs je nastaven.
7. Taktovací signál je podržen v nule po dobu 8 sestupných hran nového taktovacího oscilátoru.
8. Je připojen nový taktovací oscilátor (LP, HS nebo XT)

Je možné programově detekovat bit OSTs a zjistit tak okamžik, ve kterém došlo k přepnutí taktovací frekvence.

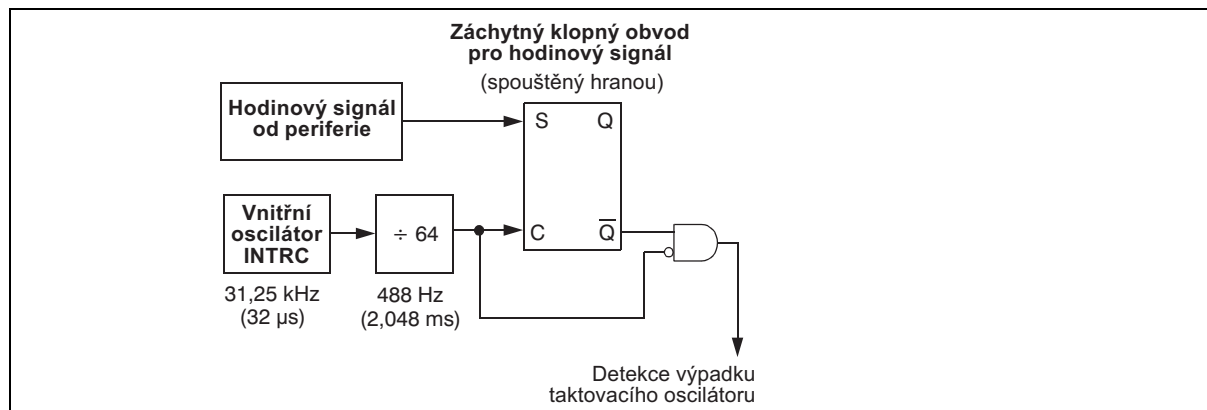
Obrázek 15-9: Časové průběhy při použití dvourychlostního rozběhu taktovacího oscilátoru



15.12.4 Ochrana taktovacího signálu

Modul FSCM je obvod pro monitorování taktovacího signálu, který umožňuje detekovat výpadek taktovacího oscilátoru. Současně je umožněn řádný výkon programu.

Obrázek 15-10: Blokové schéma modulu FSCM



Funkce modulu FSCM je povolena nastavením bitu FSCM v konfiguračním slově č.2.

V případě, že dojde k výpadku taktovacího oscilátoru, modul FSCM generuje žádost o přerušení od výpadku oscilátoru a přepne taktovací frekvenci na vnitřní oscilátor. Mikroprocesor je nadále taktován pomocí vnitřního oscilátoru až do té doby, než je ukončen chybový stav způsobený výpadkem původního taktovacího oscilátoru. Tento chybový stav může být ukončen buď resetem, vykonáním instrukce SLEEP nebo zápisem do registru OSCCON.

Frekvence vnitřního oscilátoru závisí na nastavení bitů IRCF. Pomocí bitů IRCF a SCS může být konfigurován i jiný zdroj taktovací frekvence.

Hodinový signál pro vzorkování taktovací frekvence je získán z frekvence vnitřního oscilátoru vydělením 64. Takto vzniklé prodlení mezi výpadkem taktovací frekvence a jeho zjištěním je v praxi dostatečné.

S náběžnou hranou hodinového signálu vzniklého dělením 64 dojde k nulování záchytného klopného obvodu ($CM=0$). Se sestupnou hranou taktovacího signálu od primárního nebo sekundárního zdroje je tento klopný obvod nastaven ($CM=1$). V případě, že v době sestupné hrany hodinového signálu vzniklého vydělením 64 nedojde k nastavení záchytného klopného obvodu, je detekován výpadek taktovacího oscilátoru.

Pokud je mikroprocesor v chybovém stavu způsobeném výpadkem taktovacího oscilátoru, může být tento stav ukončen signálem reset. Pokud je primární taktovací oscilátor konfigurován jako krystalový oscilátor, je aktivován časovač OST a po dobu následujících 1024 taktů potřebných pro jeho přetečení je takt zabezpečován vnitřním oscilátorem. Vykonání instrukce SLEEP nebo zápis do bitů SCS (pokud nejsou bity SCS rovny "00") způsobí uvedení mikroprocesoru do režimu se sníženou spotřebou.

Poznámka: Při povolení modulu FSCM je automaticky povolen i dvou rychlostní rozběh taktovacího oscilátoru.

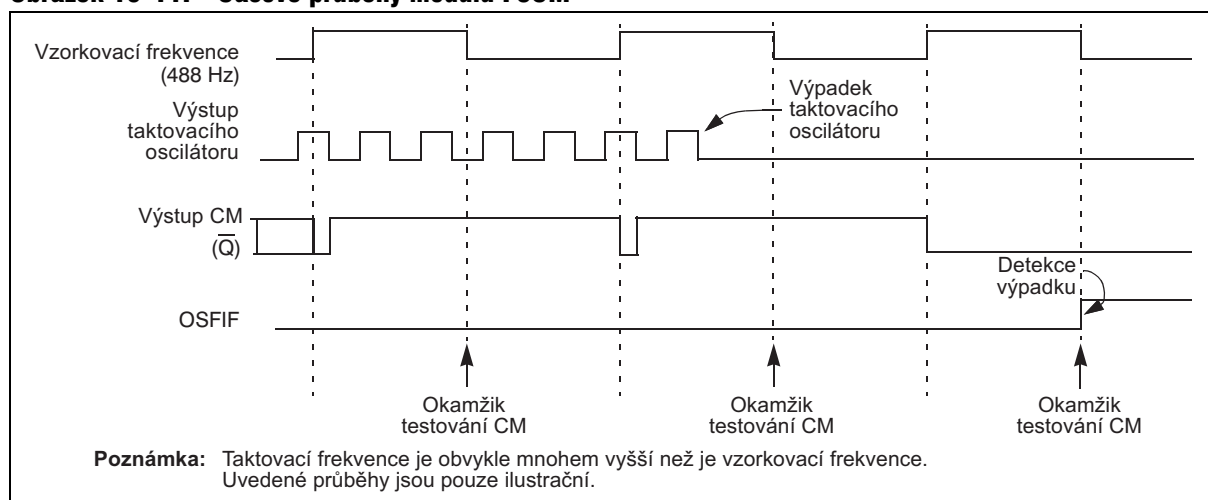
Pokud je primární taktovací oscilátor konfigurován jako EC nebo RC, způsobí reset v chybovém stavu způsobeném výpadkem taktovacího oscilátoru bezprostřední přepnutí do režimu EC resp. RC.

15.12.4.1 Ochrana taktovacího signálu v režimu Sleep

Chybový stav způsobený výpadkem taktovacího oscilátoru je ukončen vykonáním instrukce SLEEP nebo zápisem do registru OSCCON. Zdroj taktovacího signálu je implicitně určen bity SCS, když se jedná o konfiguraci T1OSC, INTRC nebo režim Sleep (žádná taktovací frekvence).

Modul FSCM dále pokračuje v monitorování taktovacího signálu. Pokud dojde k výpadku sekundárního oscilátoru, mikroprocesor je bezprostředně přepnut na vnitřní taktovací oscilátor. Pokud je nastaven bit OSFIE, dojde k vyvolání přerušení od výpadku oscilátoru.

Obrázek 15-11: Časové průběhy modulu FSCM



Poznámka: Taktovací frekvence je obvykle mnohem vyšší než je vzorkovací frekvence. Uvedené průběhy jsou pouze ilustrační.

15.12.4.2 Modul FSCM a časovač Watchdog

Pokud je detekován výpadek taktovacího signálu, jsou bity SCS1 a SCS0 uvedeny do stavu "10", což způsobí nulování časovače Watchdog (pokud je povolen).

15.12.4.3 Reset POR nebo probuzení ze stavu Sleep

Modul FSCM je schopen detekovat jakýkoli výpadek taktovacího oscilátoru následujícího po resetu POR nebo po probuzení ze stavu Sleep. Pokud je jako primární taktovací oscilátor konfigurován EC, RC nebo INTRC, dochází k zahájení monitorování bezprostředně po těchto událostech.

Při konfiguracích primárního taktovacího oscilátoru HS, LP nebo XT je situace poněkud jiná. V případě, že je doba potřebná pro rozběh a ustálení oscilátoru delší než je perioda vzorkovacího signálu, může dojít k detekci výpadku taktu. Aby k tomuto jevu nedocházelo, je po dobu rozběhu oscilátoru automaticky použit jako taktovací oscilátor vnitřní oscilátor. Po ustálení primárního oscilátoru je přepnut zdroj taktu na tento primární oscilátor. Jedná se o stejný princip, jaký je použit při dvou rychlostním rozběhu taktovacího oscilátoru. Po tomto přepnutí je vnitřní oscilátor INTRC opět použit jako zdroj vzorkovacího kmitočtu modulu FSCM.

Poznámka: Testováním bitu OSTS společně s programovou rutinou s využitím časové smyčky lze detekovat činnost taktovacího oscilátoru i bez využití přerušení.

15.12.4.4 Příklady různých situací při použití FSCM

1. SITUACE:

Mikropočítač je taktován pomocí krystalového oscilátoru, došlo k jeho výpadku a následně nastal přechod do režimu Sleep

OSTS = 0

SCS = 00

OSFIF = 1

ČINNOST UŽIVATELE:

Režim Sleep ukončí chybový stav způsobený výpadkem oscilátoru. Pokud není programově ošetřen vzniklý chybový stav ještě před vykonáním instrukce **SLEEP**, je po probuzení ze stavu Sleep mikropočítač taktován původním, tedy krystalovým oscilátorem. V tomto případě může dojít ke ztrátě informace o výpadku krystalového oscilátoru. Uživatel by neměl zahájit režim Sleep dříve, než zjistí stav bitu OSTS. Podle něj usoudí na činnost krystalového oscilátoru, ošetří případný výpadek tohoto oscilátoru a následně vykoná instrukci **SLEEP**.

2. SITUACE:

Po resetu POR je mikropočítač taktován pomocí dvou rychlostního rozběhu oscilátoru. K výpadku krystalového oscilátoru dojde před přetečením časovače OST. Pokud dojde k takovému výpadku během periody časovače OST nedojde k jeho detekci (bit OSFIF není nastaven).

OSTS = 0

SCS = 00

OSFIF = 0

ČINNOST UŽIVATELE:

Je potřeba otestovat bit OSTS. Pokud je tento bit v nule a v této době už mělo dojít k přetečení časovače OST, lze dovodit, že došlo k výpadku krystalového oscilátoru. Uživatel může následně změnit nastavení bitů SCS, což způsobí přepnutí taktovacího oscilátoru a současně uvolnění 10. bitového asynchronního čítače pro použití s obvodem Watchdog (pokud je povolen).

3. SITUACE:

Mikropočítač je během výkonu programu taktován pomocí krystalového oscilátoru, a dojde k jeho výpadku.

OSTS = 0

SCS = 00

OSFIF = 1

ČINNOST UŽIVATELE:

Je třeba nulovat bit OSFIF. Dále je nutné provést konfiguraci bitů SCS, což způsobí přepnutí taktovací frekvence a současně ukončení chybového stavu způsobeného výpadkem taktovacího oscilátoru. Později, pokud to uživatel uzná za vhodné, je možné pokusit se znovu o přepnutí taktovacího oscilátoru zpět na krystalový oscilátor. Pokud je toto provedeno, je vhodné zjistit chování krystalového oscilátoru dalším monitorováním stav bitu OSTS.

15.13 Úsporný režim (Sleep)

Mikropočítač může být uveden do úsporného režimu Sleep a následně může být z tohoto režimu uveden do normálního stavu (probuzení z režimu Sleep). Režim Sleep je vyvolán vykonáním instrukce **SLEEP**.

Při zahájení režimu Sleep je vynulován časovač Watchdog, pokud je jeho činnost povolena v konfiguračním slově (časovač Watchdog běží dále i během režimu Sleep). Dále je nulován bit $\overline{PD}$ v registru STATUS,3 a nastaven bit $\overline{TO}$ v registru STATUS,4 a je vypnut taktovací oscilátor mikropočítače. Univerzální vstupně výstupní brány mikropočítače zůstávají ve stavu, ve kterém byly před vykonáním instrukce **SLEEP** (log"1", log"0", nebo stav vysoké impedance).

Pro dosažení co nejmenší spotřeby v režimu Sleep je nutné zapojit výstupní vývody tak, aby nemohly dodávat proud do dalších obvodů. Vstupní vývody je nutné připojit na log"1" nebo log"0" aby se zabránilo vzniku ztrát způsobených náhodným překlápěním na plovoucích vstupech. Vstup čítače/časovače TOCKI je nutné rovněž připojit na definovanou logickou úroveň. Při návrhu zapojení je nutné počítat i se spotřebou vnitřních Pull-up rezistorů připojených na vstupní vývody brány Portb. Pro minimalizaci příkonu je nutné vypnout AD převodník.

Vývod $\overline{MCLR}$ musí být v režimu Sleep připojen na log"1"

15.13.1 Probuzení ze stavu Sleep

Mikropočítač může být probuzen ze stavu Sleep jedním z následujících způsobů:

1. Vnější Reset na vstupu $\overline{\text{MCLR}}$
2. Přetečení časovače Watchdog (pokud je časovač Watchdog povolen)
3. Vnější přerušení na vývodu RBO/INT nebo změnou na 4 bitech brány PORTB nebo přerušením od periférií.

K dalším druhům přerušení dojít nemůže, protože během režimu Sleep nefunguje taktovací oscilátor, nezbytný pro činnost příslušných periférií.

První způsob probuzení má za následek Reset mikropočítače a zahájení výkonu programu od adresy 0000h. Další způsoby mají za následek pokračování v programu výkonem instrukce následující za instrukcí **SLEEP**. Způsob probuzení ze stavu Sleep lze určit podle nastavení bitů $\overline{\text{PD}}$ a $\overline{\text{TO}}$. Bit $\overline{\text{PD}}$, který je nastaven do log“1“ při zapnutí napájecího napětí se nuluje při vykonání instrukce **SLEEP**. Bit $\overline{\text{TO}}$ se nuluje, pokud dojde k přetečení časovače Watchdog, které způsobí probuzení ze stavu Sleep.

Mikropočítač může být probuzen ze stavu Sleep jedním z následujících přerušení od periférií:

1. Přetečení čítače/časovače Timer1 v režimu asynchronního čítače
2. Přerušení od modulu CCP
3. Přerušení od spouštěcí funkce CCP (Timer1 musí pracovat jako asynchronní čítač s vnějším signálem na vstupu)
4. Přerušení od modulu SSP (detekce start nebo stop bitu)
5. Přerušení od modulu SSP (vysílání nebo příjem dat)
6. Dokončení AD převodu
7. Dokončení zápisu do EEPROM
8. Změna stavu na výstupu komparátoru
9. Přijetí nebo dokončení vyslání dat modulem AUSART (v synchronním režimu Slave)

Ostatní periférie nemohou způsobit probuzení mikropočítače ze stavu Sleep.

Během výkonu instrukce **SLEEP** je připravena k vykonání instrukce následující, ležící na adrese PC+1. Probuzení ze stavu Sleep prostřednictvím některého z přerušení proběhne nezávisle na tom, zda je globálně přerušení povoleno či zakázáno. Pokud je zakázáno ($\text{GIE}=0$), vykoná se tato následující instrukce a následně i instrukce další. Pokud je povoleno ($\text{GIE}=1$), vykoná se tato instrukce rovněž a následně je do PC umístěn vektor přerušení (0004h) a program vykonává rutinu přerušení. Pokud uživatel nechce, aby došlo k výkonu této následující instrukce před obsluhou přerušení, je nutné zařadit za instrukci **SLEEP** instrukci **NOP**.

15.13.2 Probuzení ze stavu Sleep pomocí přerušení

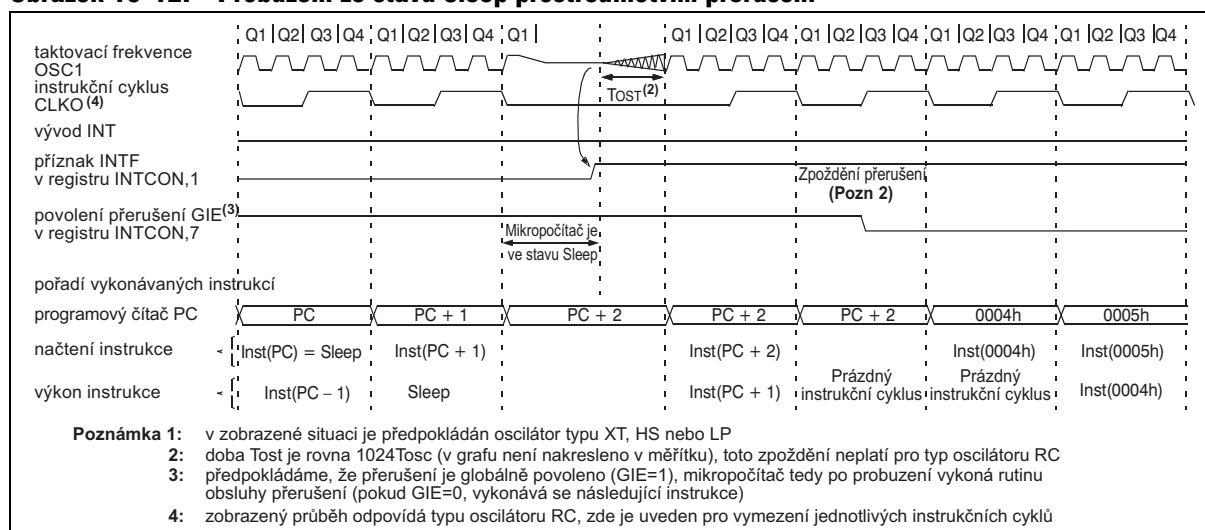
Pokud je přerušení globálně zakázáno ($\text{GIE}=0$), ale některý ze zdrojů přerušení má současně nastaven do log“1“ jak řídicí bit (maskovací), tak i příznak přerušení, může nastat jeden z následujících jevů:

- Pokud žádost o nemaskované přerušení nastane instrukcí **předcházející** instrukci **SLEEP**, bude tato instrukce **SLEEP** vykonána jako instrukce **NOP**. To znamená, že časovač Watchdog ani předdělička nebudou vynulovány, bit $\overline{\text{TO}}$ nebude nastaven do log“1“ a bit $\overline{\text{PD}}$ nebude nulován..
- Pokud žádost o nemaskované přerušení nastane **během** nebo **po** výkonu instrukce **SLEEP**, mikropočítač bude okamžitě probuzen ze stavu Sleep. Instrukce **SLEEP** bude kompletně dokončena před probuzením, tedy časovač Watchdog a předdělička budou vynulovány, bit $\overline{\text{TO}}$ bude nastaven do log“1“ a bit $\overline{\text{PD}}$ bude nulován.

Dokonce i v případě, že hodnota příznakového bitu přerušení byla testována těsně před vykonáním instrukce **SLEEP** je možné, že dojde k jeho nastavení ještě před dokončením instrukce **SLEEP**. Pro zjištění, zda instrukce **SLEEP** skutečně řádně proběhla je nutné testovat hodnotu bitu $\overline{\text{PD}}$. Pokud je tento bit nastaven, byla instrukce **SLEEP** vykonána jako instrukce **NOP**.

Před zahájením režimu **SLEEP** je vhodné vynulovat časovač WDT provedením instrukce **CLRWDT**.

Obrázek 15-12: Probuzení ze stavu Sleep prostřednictvím přerušení ⁽¹⁾



15.14 Ladění v aplikaci ICD

Pokud je ladění v aplikaci povoleno v konfiguračním slově, může být mikroprocesor ovládán pomocí dvou vývodů v režimu ladění (debugger). Tato jednoduchá funkce je podporována prostředím MPLAB® ICD. Povolení tohoto režimu způsobí určitá omezení pro použití mikroprocesoru. V tabulce 15-7 jsou tato omezení uvedena.

Tabulka 15-7: Omezení v režimu ICD

Vývody mikroprocesoru	RB6, RB7
Zásobník	1 úroveň
Paměť programu	na adrese 0000h musí být NOP posledních 100h adres
Paměť dat	0x070 (0x0F0, 0x170, 0x1F0) 0x1EB-0x1EF

Při použití ladění v aplikaci musí uživatel propojit s externím zařízením následující vývody mikroprocesoru: MCLR, VDD, VSS, RB7, RB6. Hardwarové prostředky podporující ladění v aplikaci firmy Microchip jsou uvedeny v zadní části této dokumentace.

15.15 Verifikace programu a ochrana kódu proti vyčtení

Pokud není v konfiguračním slově nastavena ochrana kódu proti vyčtení, je možné vyčíst paměť programu pro účely porovnání jejího obsahu s referenčním kódem, tedy provést tzv. verifikaci.

15.16 ID

Pro potřeby uložení identifikačních čísel nebo kontrolních součtů jsou k dispozici čtyři buňky na adresách 2000h-2003h. Tyto buňky nejsou přístupné během výkonu programu, lze do nich zapisovat nebo je číst pouze během zavádění programu do součástky. Pro potřeby uživatele lze použít pouze dolní čtyři bity každé buňky.

15.17. Programování mikroprocesoru na osazené desce (ICSP)

Do mikroprocesoru PIC16F87/88 může být zaveden program pomocí sériového rozhraní přímo v koncové aplikaci. Toto sériové synchronní programování se provádí pomocí dvou univerzálních vývodů brány Portb, když po jednom z vodičů přicházejí data, druhý je použit pro hodinový signál. Další tři vodiče jsou použity pro napájecí napětí VDD a VSS a pro přivedení programovacího napětí. Uživatel tak může na desku plošných spojů osadit mikroprocesor bez programu, a tento zavést těsně před použitím aplikace. Tato technologie umožňuje používat nové verze programu ve stávajícím zařízení případně používat různý firmware dle potřeb zákazníka.

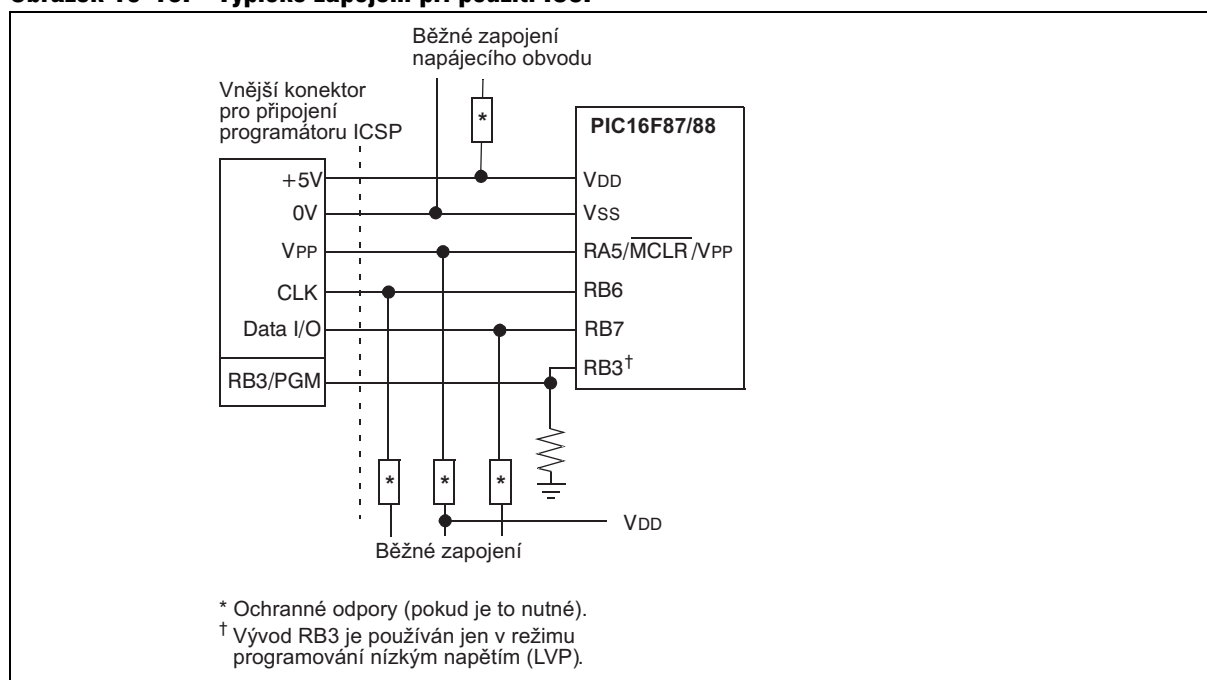
Typické schéma zapojení je uvedeno na obrázku 15-13.

Podrobnější informace o ICSP lze nalézt v dokumentaci s názvem *In-Circuit Serial Programming Guide* označené DS30277.

Poznámka: Vývody pro použití s čítačem /časovačem Timer1 T1OSI a T1OSO jsou sdílené s vývody PGD a PGC používanými v režimu ICSP.

Při použití oscilátoru Timer1 nemusí fungovat režim ICSP nebo ICD zcela správně. Zejména nemusí správně komunikovat vývojový prostředek s mikroprocesorem. Výsledkem současného propojení může být i zničení krystalu. Pokud je požadováno programování ICSP nebo ladění v aplikaci ICD, je nutné toto provádět bez připojeného krystalu Timer1. Tento krystal může být připojen po provedení programování nebo ladění. Ostatní prvky oscilátoru jako jsou vazební kondenzátory mohou být v zapojení ponechány.

Obrázek 15-13: Typické zapojení při použití ICSP



15.18 Programování nízkým napětím (LVP)

Programovací režim LVP je povolen řídicím bitem v konfiguračním slově. Tento režim umožňuje zavedení programu do součástky s užitím programovacího napětí v rozsahu běžného napájecího napětí. Na vstup $\overline{\text{MCLR}}$ tedy není nutné přivádět zvýšené programovací napětí. Mikroprocesor je dodáván s povoleným režimem LVP. Vývod RB3 je v tomto režimu rezervován pro LVP a není možné jej použít jako běžný vývod.

Pokud není režim LVP používán, může být vývod RB3 použit jako běžný číslicový vstup/výstup. Bit LVP může být programován jen tehdy, pokud je programovací režim aktivován zvýšeným programovacím napětím V_{ihh} na vývodu $\overline{\text{MCLR}}$. Změna nastavení tohoto bitu je možná jen tehdy, pokud je na vstupu $\overline{\text{MCLR}}$ aktivováno zvýšené programovací napětí.

Je nutné si uvědomit, že jakmile je bit LVP nulován, jeho nastavení je možné pouze v režimu programování zvýšeným programovacím napětím a současně lze celý mikroprocesor programovat pouze v tomto režimu.

Při použití programování nízkým napětím v režimu ICSP musí být mikroprocesor napájen napětím v rozsahu 4,5 – 5,5 V, pokud je prováděno mazání celého obsahu paměti. To zahrnuje i přeprogramování konfiguračních bitů ze stavu ON (zapnuto) do stavu OFF (vypnuto). Pro všechny ostatní případy programování nízkým napětím v režimu ICSP může být použito napájecího napětí v celém rozsahu napájecího napětí. Tím jsou míněny kalibrační konstanty, uživatelské kódy ID nebo programový kód, který může být přeprogramován nebo přidán.

Pokud je nastaven bit LVP v konfiguračním slově mikroprocesoru je nutné při programování dodržet následující postup:

1. Připojit napájecí napětí na vývod Vdd
2. Připojit napětí Vss na vstup $\overline{\text{MCLR}}$
3. Připojit napětí Vdd na vstup RB3/PGM
4. Připojit napájecí napětí na vstup $\overline{\text{MCLR}}$
5. Pokračovat v programování obvyklým způsobem

Poznámka 1: Režim programování zvýšeným programovacím napětím (normální programování) je dostupný vždy, nezávisle na nastavení bitu LVP. Do tohoto režimu se mikroprocesor dostane přivedením vysokého napětí na vstup $\overline{\text{MCLR}}$.

- 2: Při použití LVP nelze používat vývod RB3 jako normální vývod.
- 3: Při použití LVP a povolení pull-up na bráně PortB musí být odpovídající bit registru TRIS nulován.
- 4: Při povoleném LVP nelze ponechat vývod RB3 nezapojený. Vnější programátor by měl zajistit přidržení tohoto vstupu v log.0. Pokud bude na nezatíženém vstupu náhodně log.1, mikroprocesor to vyhodnotí jako signál k přepnutí do programovacího režimu LVP.
- 5: Implicitní nastavení LVP je "povoleno". Zákaz je možné provést nulováním bitu LVP v konfiguračním slově č.1.
- 6: Pro dosažení maximální kompatibility s ostatními mikroprocesory řady PIC16Cxxx je vhodné zakázat režim LVP.

PIC16F87/88

Poznámky:

16 INSTRUKČNÍ SADA

Každý mikropočítač z řady PIC16F8x má instrukční kód o délce 14 bitů. V tomto 14-bitovém slově je obsažen operační kód instrukce (Opcode), který určuje o jakou instrukci se jedná a dále je obsažen jeden nebo dva operandy a specifikován cílový registr. Přehled všech instrukcí je uveden v tabulce 16-2, kde jsou instrukce rozděleny do třech skupin. Operandem bajtově orientovaných instrukcí je celý registr, operandem bitově orientovaných instrukcí je určený bit libovolného registru v paměti RWM. Třetí skupina instrukcí pracuje s konstantami (Literal) nebo se jedná o instrukce řídicí. V tabulce 16-1 jsou vysvětlena používaná označení v přehledu instrukcí.

Instrukční soubor je logicky rozdělen do třech skupin:

- **Bajtově orientované** instrukce
- **Bitově orientované** instrukce
- **Řídicí** instrukce

Bajtově orientované instrukce používají pro určení operandu zápis ,f', když se jedná o adresu registru v paměti RWM. Výsledek operace bude uložen do cílového registru, který je určen ukazatelem ,d' ve formátu instrukce.

Pokud je ukazatel ,d' ve formátu instrukce roven ,0', je výsledek uložen do střadače (W). Pokud je tento ukazatel roven ,1', je výsledek uložen do registru, se kterým instrukce pracuje, tedy s operandem instrukce určeným pomocí zápisu ,f'.

Bitově orientované instrukce používají pro určení operandu ukazatel ,b', který určuje pořadí bitu v registru, který je určen pomocí zápisu ,f'. Krajní pravý bit má hodnotu ,b' rovnu ,0', krajní levý bit má hodnotu ,b' rovnu ,7'. Bitově orientovaná instrukce pracuje pouze s jediným adresovaným bitem.

Všechny instrukce se vykonávají v jediném instrukčním cyklu vyjma instrukcí, kdy je splněna podmínka instrukcí podmíněného větvení programu nebo v případech, kdy dojde k zápisu do čítače instrukcí PC. V těchto případech trvá vykonání instrukce dva instrukční cykly, přičemž druhý instrukční cyklus odpovídá vykonání instrukce NOP. Jeden instrukční cyklus obsahuje čtyři takty oscilátoru mikropočítače. Při frekvenci taktovacího oscilátoru 4 MHz trvá vykonání jedné instrukce právě 1 μ s. Pokud dojde ke splnění podmínky instrukce podmíněného větvení nebo bude výsledkem instrukce zápis do programového čítače PC, bude tato instrukce trvat právě 2 μ s.

Na obrázku 16-1 je uveden formát všech druhů instrukcí.

Poznámka: Pro zabezpečení zpětné kompatibility s budoucími mikropočítači řady PIC16F87/88 nepoužívejte zastaralé instrukce **OPTION** a **TRIS**.

Všechny uváděné příklady používají pro zápis hexadecimálního čísla formát 0xhh, kde h je hexadecimální číslice v rozsahu 0 – F

Řídicí instrukce používají zápis ,k' pro určení 8- nebo 11-bitové konstanty.

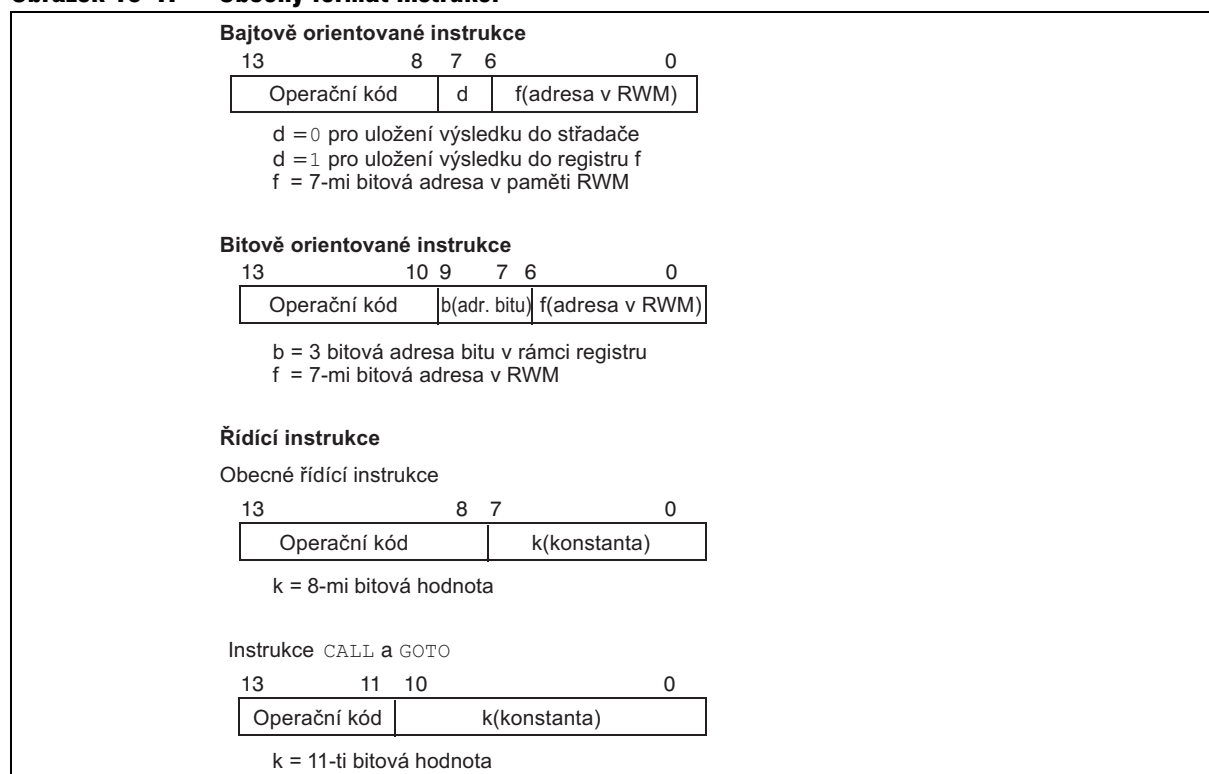
16.1 Operace typu čti-modifikuj-zapiš

Některé instrukce jsou prováděny postupem čti-modifikuj-zapiš (Read-Modify-Write). Obsah registru je načten, je provedena jeho změna v důsledku vykonání instrukce a výsledek je uložen do cílového registru. Takto probíhají např. instrukce BSF nebo BCF.

Tabulka 16-1: Označení používaná v přehledu instrukcí

Označení	Popis
f	adresa registru v paměti RWM v rozsahu od 0x00 do 0x7f
w	střadač (pracovní registr)
b	bitová adresa v rozsahu od 0 do 7 určující bit registru
k	datová konstanta nebo návěští (adresa v paměti programu)
x	nemá vliv, může nabývat hodnot ,0' nebo ,1'. Překladač generuje na místo x vždy ,0'. Jedná se o doporučený postup pro zajištění kompatibility se všemi vývojovými nástroji Microchip.
d	určení cílového registru. Pro d=0 je výsledek operace uložen do střadače (W), pro d=1 je výsledek uložen do registru, se kterým instrukce pracuje (f). Implicitní nastavení je d=1
PC	programový čítač (Program Counter)
TO	příznak přetečení čítače Watchdog
PD	příznak režimu Sleep

Obrázek 16-1: Obecný formát instrukcí



PIC16F87/88

Tabulka 16-2: Instrukční sada PIC16F87/88

Instrukce, Operandy	Popis		Instrukční cykly	14-bitový instrukční kód		Ovlivňované příznaky	Pozn.
				MSb	LSb		
Bajtové orientované instrukce							
ADDWF	f,d	ADD W and F	přičtení registru ke střadači	1	00 0111 dfff ffff	C,DC,Z	1,2
ANDWF	f,d	AND W with F	logický součin střadače a registru	1	00 0101 dfff ffff	Z	1,2
CLRF	f	CLeaR F	nulování registru	1	00 0001 1fff ffff	Z	2
CLRWF	—	CLeaR W	nulování střadače	1	00 0001 0xxx xxxx	Z	
COMF	f,d	COMplement F	negace registru	1	00 1001 dfff ffff	Z	1,2
DECf	f,d	DECrement F	snížení obsahu registru o jedničku	1	00 0011 dfff ffff	Z	1,2
DECFS	f,d	DECrement F Skip if Zero	dekrementace registru, skok při 0	1 (2)	00 1011 dfff ffff		1,2,3
INCF	f,d	INCrement F	zvýšení obsahu registru o jedničku	1	00 1010 dfff ffff	Z	1,2
INCFSZ	f,d	INCrement F Skip if Zero	inkrementace registru, skok při 0	1 (2)	00 1111 dfff ffff		1,2,3
IORWF	f,d	Inclusive OR W with F	logický součet střadače a registru	1	00 0100 dfff ffff	Z	1,2
MOVf	f,d	MOVe F to W	načtení registru do střadače	1	00 1000 dfff ffff	Z	1,2
MOVWF	f	MOVe W to F	přesun střadače do registru	1	00 0000 1fff ffff		
NOP	—	No OPeration	prázdná instrukce	1	00 0000 0xx0 0000		
RLF	f,d	Rotate Left F through carry	rotace registru doleva přes příznak C	1	00 1101 dfff ffff	C	1,2
RRF	f,d	Rotate Right F through carry	rotace registru doprava přes příznak C	1	00 1100 dfff ffff	C	1,2
SUBWF	f,d	SUBtract W from F	odečtení střadače od registru	1	00 0010 dfff ffff	C,DC,Z	1,2
SWAPf	f,d	SWAP nibbles in F	záměna dolního a horního půlbajtu	1	00 1110 dfff ffff		1,2
XORWF	f,d	eXclusive OR W with F	nonekvivalence střadače a registru	1	00 0110 dfff ffff	Z	1,2
Bitové orientované instrukce							
BCF	f,b	Bit Clear F	nulování bitu registru	1	01 00bb bfff ffff		1,2
BSF	f,b	Bit Set F	nastavení bitu registru	1	01 01bb bfff ffff		1,2
BTFSC	f,b	Bit Test F Skip if Clear	skok při nulovém bitu registru	1 (2)	01 10bb bfff ffff		3
BTFSS	f,b	Bit Test F Skip if Set	skok při nastaveném bitu registru	1 (2)	01 11bb bfff ffff		3
Řídící instrukce							
ADDLW	k	ADD Literal and W	přičtení konstanty ke střadači	1	11 111x kkkk kkkk	C,DC,Z	
ANDLW	k	AND Literal with W	logický součin střadače a konstanty	1	11 1001 kkkk kkkk	Z	
CALL	k	CALL subroutine	volání podprogramu	2	10 0kkk kkkk kkkk		
CLRWDt	—	CLeaR WatchDog Timer	nulování časovače Watchdog	1	00 0000 0110 0100	T0,PD	
GOTO	k	GO TO adress	skok na návesti	2	10 1kkk kkkk kkkk		
IORLW	k	Inclusive OR Literal with W	logický součet střadače a konstanty	1	11 1000 kkkk kkkk	Z	
MOVLW	k	MOVe Literal to W	načtení konstanty do střadače	1	11 00xx kkkk kkkk		
RETFIE	—	RETurn From Interrupt En.	návrat z podprogramu s povolením přerušení	2	00 0000 0000 1001		
RETLW	k	RETurn with Literal in W	návrat z podprogramu s konstantou	2	11 01xx kkkk kkkk		
RETURN	—	RETURN from subroutine	návrat z podprogramu	2	00 0000 0000 1000		
SLEEP	—	go into standby mode SLEEP	zahájení režimu Sleep	1	00 0000 0110 0011	T0,PD	
SUBLW	k	SUBtract W from Literal	odečtení střadače od konstanty	1	11 110x kkkk kkkk	C,DC,Z	
XORLW	k	eXclusive OR Literal with W	nonekvivalence střadače a konstanty	1	11 1010 kkkk kkkk	Z	

Poznámka 1: Pokud bude záchytný registr brány (PORTA nebo PORTB) modifikován (např. načtením sebe sama instrukcí **MOVf PORTB,1**), bude do něj zapsána hodnota, která bude v okamžiku vykonání instrukce fyzicky přítomna na vnějších vývodech součástky. Pokud bude například v záchytném registru vývodu konfigurovaném jako vstup hodnota log. '1' a tento vývod bude vnějším obvodem udržován ve stavu log. '0', bude po provedení instrukce modifikující příslušný záchytný registr zapsána na odpovídající bit log. '0'.

2: Pokud je operandem této instrukce (a případně i cílovým registrem při d=1) registr TMR0, dojde při vykonání této instrukce rovněž k vynulování předděličky, pokud je tato v této době připojena k čítači Timer0.

3: Pokud dojde k zápisu do programového čítače PC nebo pokud je splněna podmínka instrukce podmíněného větvení programu, trvá vykonání této instrukce dva instrukční cykly. Druhý instrukční cyklus odpovídá vykonání instrukce **NOP**.

Poznámka: Další informace o instrukční sadě jsou k dispozici v dokumentaci DS33023 s názvem: *PICmicro® Mid-Range MCU Family Reference Manual*.

PIC16F87/88

Poznámky:

16.2 Popis instrukcí (v abecedním pořádku)

ADDLW	ADD Literal and W při čtení konstanty ke střadači	BCF	Bit Clear F nulování bitu registru
Syntaxe:	návěští ADDLW k ; komentář	Syntaxe:	návěští BCF f,b ; komentář
Operandy:	k = (0-255)	Operandy:	f = (0-127) b = (0-7)
Operace:	(W) = (W) + k	Operace:	f(b) = 0
Ovlivňované příznaky:	C, DC, Z	Ovlivňované příznaky:	žádné
Popis:	Obsah střadače je sečten s osmi bitovou konstantou a výsledek je umístěn do střadače.	Popis:	Je vynulován bit s adresou ,b' v registru s adresou ,f'
ADDWF	ADD W and F přičtení registru ke střadači	BSF	Bit Set F nastavení bitu registru
Syntaxe:	návěští ADDWF f,d ; komentář	Syntaxe:	návěští BSF f,b ; komentář
Operandy:	f = (0-127) d = (0,1)	Operandy:	f = (0-127) b = (0-7)
Operace:	(cílový registr) = (W) + (f)	Operace:	f(b) = 1
Ovlivňované příznaky:	C, DC, Z	Ovlivňované příznaky:	žádné
Popis:	Obsah střadače je sečten s obsahem registru a výsledek je umístěn do střadače pro d=0 nebo do registru pro d=1.	Popis:	Je nastaven bit s adresou ,b' v registru s adresou ,f'
ANDLW	AND Literal with W logický součin střadače a konstanty	BTFS	Bit Test F Skip if Set skok při nastaveném bitu registru
Syntaxe:	návěští ANDLW k ; komentář	Syntaxe:	návěští BTFS f,b ; komentář
Operandy:	k = (0-255)	Operandy:	f = (0-127) b = (0-7)
Operace:	(W) = (W) AND k	Operace:	skok za podmínky f(b) = 1
Ovlivňované příznaky:	Z	Ovlivňované příznaky:	žádné
Popis:	Je provedena operace logického součinu mezi obsahem střadače a osmi bitovou konstantou. Výsledek je umístěn do střadače.	Popis:	Pokud je bit s adresou ,b' v registru s adresou ,f' roven ,0', je následující instrukce vykonána. Pokud je tento bit roven ,1', je místo následující instrukce vykonána vždy instrukce NOP. Doba vykonání instrukce je v tomto případě 2 instrukční cykly.
ANDWF	AND W with F logický součin střadače a registru	BTFS	Bit Test F Skip if Clear skok při nulovém bitu registru
Syntaxe:	návěští ANDWF f,d ; komentář	Syntaxe:	návěští BTFS f,b ; komentář
Operandy:	f = (0-127) d = (0,1)	Operandy:	f = (0-127) b = (0-7)
Operace:	(cílový registr) = (W) AND (f)	Operace:	skok za podmínky f(b) = 0
Ovlivňované příznaky:	Z	Ovlivňované příznaky:	žádné
Popis:	Je provedena operace logického součinu mezi obsahem střadače a obsahem registru. Výsledek je umístěn do střadače pro d=0 nebo do registru pro d=1.	Popis:	Pokud je bit s adresou ,b' v registru s adresou ,f' roven ,1', je následující instrukce vykonána. Pokud je tento bit roven ,0', je místo následující instrukce vykonána vždy instrukce NOP. Doba vykonání instrukce je v tomto případě 2 instrukční cykly.

CALL	CALL subroutine volání podprogramu
Syntaxe:	návěští CALL k ; komentář
Operandy:	k = (0-2047)
Operace:	STACK = (PC) + 1 PC(b10-b0) = k PC(b12,b11) = PCLATH(b4,b3)
Ovlivňované příznaky:	žádné
Popis:	Je volán podprogram ležící v paměti programu na adrese „k“. Do zásobníku je uložena návratová adresa. Adresa podprogramu v rozsahu (0-2047) je vložena do programového čítače PC. Horní dva bity programového čítače jsou současně doplněny ze zachytného registru PCLATH. Tři nižší bity PCLATH jsou v tomto případě ignorovány. Instrukce trvá dva instrukční cykly.

CLRF	CLear F nulování registru
Syntaxe:	návěští CLRF f ; komentář
Operandy:	f = (0-127)
Operace:	f = 0x00 Z = 1
Ovlivňované příznaky:	Z
Popis:	Je vynulován obsah registru f a je nastaven příznak nuly Z.

CLRW	CLear W nulování střadače
Syntaxe:	návěští CLRW ; komentář
Operandy:	žádné (implicitně je adresován střadač)
Operace:	W = 0x00 Z = 1
Ovlivňované příznaky:	Z
Popis:	Je vynulován obsah střadače a je nastaven příznak nuly Z.

CLRWDT	CleaR WatchDog Timer nulování časovače Watchdog
Syntaxe:	návěští CLRW DT ; komentář
Operandy:	žádné
Operace:	WDT = 0x00 předdělička Watchdog = 0 příznak TO = 1 příznak PD = 1
Ovlivňované příznaky:	$\overline{TO}$, $\overline{PD}$
Popis:	Je vynulován čítač Watchdog a jeho předdělička. Jsou nastaveny příznakové bity TO a PD.

COMF	COMplement F negace registru
Syntaxe:	návěští COMF f,d ; komentář
Operandy:	f = (0-127) d = (0,1)
Operace:	(cílový registr) = (f)
Ovlivňované příznaky:	Z
Popis:	Obsah registru je negován a výsledek je umístěn do střadače pro d=0 nebo do registru pro d=1

DECF	DECrement F snížení obsahu registru o jedničku
Syntaxe:	návěští DESF f,d ; komentář
Operandy:	f = (0-127) d = (0,1)
Operace:	(cílový registr) = (f) - 1
Ovlivňované příznaky:	Z
Popis:	Od obsahu registru je odečtena jednička a výsledek je umístěn do střadače pro d=0 nebo do registru pro d=1

DECFSZ	DECrement F Skip if Zero dekrementace registru, skok při 0
Syntaxe:	návěští DECFSZ f,d ; komentář
Operandy:	f = (0-127) d = (0,1)
Operace:	(cílový registr) = (f) - 1 Skok při výsledku rovném ,0'
Ovlivňované příznaky:	žádné
Popis:	Od obsahu registru je odečtena jednička a výsledek je umístěn do střadače pro d=0 nebo do registru pro d=1. Pokud není výsledek roven ,0', je následující instrukce vykonána. Pokud je výsledek roven ,0', je místo následující instrukce vykonána vždy instrukce NOP. Doba vykonání instrukce je v tomto případě 2 instrukční cykly

INCFSZ	INCrement F Skip if Zero inkrementace registru, skok při 0
Syntaxe:	návěští INCFSZ f,d ; komentář
Operandy:	f = (0-127) d = (0,1)
Operace:	(cílový registr) = (f) + 1 Skok při výsledku rovném ,0'
Ovlivňované příznaky:	žádné
Popis:	K obsahu registru je přičtena jednička a výsledek je umístěn do střadače pro d=0 nebo do registru pro d=1. Pokud není výsledek roven ,0', je následující instrukce vykonána. Pokud je výsledek roven ,0', je místo následující instrukce vykonána vždy instrukce NOP. Doba vykonání instrukce je v tomto případě 2 instrukční cykly

GOTO	GO TO adress skok na návěští
Syntaxe:	návěští GOTO k ; komentář
Operandy:	k = (0-2047)
Operace:	PC(b10-b0) = k PC(b12,b11) = PCLATH(b4,b3)
Ovlivňované příznaky:	žádné
Popis:	Je proveden nepodmíněný skok na určenou adresu. Adresa v rozsahu (0-2047) je vložena do programového čítače PC. Horní dva bity programového čítače jsou současně doplněny ze záchytného registru PCLATH. Tři nižší bity PCLATH jsou v tomto případě ignorovány. Instrukce trvá dva instrukční cykly.

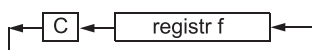
IORLW	Inclusive OR Literal with W logický součet střadače a konstanty
Syntaxe:	návěští IORLW k ; komentář
Operandy:	k = (0-255)
Operace:	(W) = (W) OR k
Ovlivňované příznaky:	Z
Popis:	Je provedena operace logického součtu mezi obsahem střadače a osmi bitovou konstantou. Výsledek je umístěn do střadače.

INCF	INCrement F zvýšení obsahu registru o jedničku
Syntaxe:	návěští INCF f,d ; komentář
Operandy:	f = (0-127) d = (0,1)
Operace:	(cílový registr) = (f) + 1
Ovlivňované příznaky:	Z
Popis:	K obsahu registru je přičtena jednička a výsledek je umístěn do střadače pro d=0 nebo do registru pro d=1

IORWF	Inclusive OR W with F logický součet střadače a registru
Syntaxe:	návěští IORWF f,d ; komentář
Operandy:	f = (0-127) d = (0,1)
Operace:	(cílový registr) = (W) OR (f)
Ovlivňované příznaky:	Z
Popis:	Je provedena operace logického součtu mezi obsahem střadače a obsahem registru. Výsledek je umístěn do střadače pro d=0 nebo do registru pro d=1.

MOVF	MOVe F to W načtení registru do střadače	NOP	No Operation prázdná instrukce
Syntaxe:	návěští MOVF f,d ; komentář	Syntaxe:	návěští NOP ; komentář
Operandy:	f = (0-127) d = (0,1)	Operandy:	žádné
Operace:	(cílový registr) = (f)	Operace:	žádná operace
Ovlivňované příznaky:	Z	Ovlivňované příznaky:	žádné
Popis:	Obsah registru je umístěn do střadače pro d=0 nebo do registru pro d=1. Operand d=1 je vhodné použít při testování obsahu registru na nulu, protože instrukce ovlivňuje příznak nuly Z.	Popis:	Během jednoho instrukčního cyklu nedochází k žádné operaci.
MOVLW	MOVe Literal to W načtení konstanty do střadače	RETfie	RETurn From Interrupt návrat s povolením přerušení
Syntaxe:	návěští MOVLW k ; komentář	Syntaxe:	návěští RETfie ; komentář
Operandy:	k = (0-255)	Operandy:	žádné
Operace:	(W) = k	Operace:	(PC) = STACK GIE = 1
Ovlivňované příznaky:	žádné	Ovlivňované příznaky:	žádné
Popis:	Do střadače je načtena osmibitová konstanta. Bity v operačním kódu, který nemají žádný význam budou přeloženy jako ,0'.	Popis:	Do programového čítače je umístěná návratová adresa z vrcholu zásobníku a současně je povoleno přerušení nastavením bitu GIE. Instrukce trvá dva instrukční cykly.
MOVWF	MOVe W to F přesun střadače do registru	RETLW	RETurn with Literal in W návrat z podprogramu s konstantou
Syntaxe:	návěští MOVWF f ; komentář	Syntaxe:	návěští RETLW k; komentář
Operandy:	f = (0-127)	Operandy:	k = (0-255)
Operace:	(f) = (W)	Operace:	(PC) = STACK (W) = k
Ovlivňované příznaky:	žádné	Ovlivňované příznaky:	žádné
Popis:	Obsah střadače je přesunut do registru.	Popis:	Do programového čítače je umístěná návratová adresa z vrcholu zásobníku a současně je do střadače načtena osmibitová konstanta, která je součástí instrukčního kódu. Instrukce trvá dva instrukční cykly.

RLF	Rotate Left F rotace registru doleva přes C
Syntaxe:	návěští RLF f,d ; komentář
Operandy:	f = (0-127) d = (0,1)
Operace:	(cílový registr) = rotace (f) doleva
Ovlivňované příznaky:	C
Popis:	Obsah registru je posunut o jeden bit doleva přes bit C - příznak přenosu CARRY (rotace probíhá přes devět bitů). Výsledek je umístěn do střadače pro d=0 nebo do registru pro d=1.

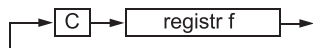


SLEEP	go into mode SLEEP zahájení režimu Sleep
Syntaxe:	návěští SLEEP ; komentář
Operandy:	žádné
Operace:	WDT = 0x00h předdělička Watchdog = 0 příznak $\overline{TO}$ = 1 příznak $\overline{PD}$ = 0
Ovlivňované příznaky:	TO, $\overline{PD}$
Popis:	Je vynulován čítač Watchdog a jeho předdělička. Je nastaven příznakový bit TO a vynulován příznakový bit PD. Procesor počítače je uveden do spánkového režimu Sleep, ve kterém je zastaven taktovací oscilátor.

RETURN	RETURN from subroutine návrát z podprogramu
Syntaxe:	návěští RETURN ; komentář
Operandy:	žádné
Operace:	(PC) = STACK
Ovlivňované příznaky:	žádné
Popis:	Je proveden návrat z podprogramu tím, že do programového čítače je umístěná návratová adresa z vrcholu zásobníku. Instrukce trvá dva instrukční cykly.

SUBLW	SUBtract W from Literal odečtení střadače od konstanty
Syntaxe:	návěští SUBLW k ; komentář
Operandy:	k = (0-255)
Operace:	(W) = k - (W)
Ovlivňované příznaky:	C, DC, Z
Popis:	Obsah střadače je odečten (metodou dvojkového doplňku) od osmi bitové konstanty a výsledek je umístěn do střadače.

RRF	Rotate Right F rotace registru doprava přes C
Syntaxe:	návěští RRF f,d ; komentář
Operandy:	f = (0-127) d = (0,1)
Operace:	(cílový registr) = rotace (f) doprava
Ovlivňované příznaky:	C
Popis:	Obsah registru je posunut o jeden bit doprava přes bit C - příznak přenosu CARRY (rotace probíhá přes devět bitů). Výsledek je umístěn do střadače pro d=0 nebo do registru pro d=1.



SUBWF	SUBtract W from F odečtení střadače od registru
Syntaxe:	návěští SUBWF f,d ; komentář
Operandy:	f = (0-127) d = (0,1)
Operace:	(cílový registr) = (f) - (W)
Ovlivňované příznaky:	C, DC, Z
Popis:	Obsah střadače je odečten (metodou dvojkového doplňku) od obsahu registru a výsledek je umístěn do střadače pro d=0 nebo do registru pro d=1.

SWAPF	SWAP nibbles in F záměna dolního a horního půlbajtu
Syntaxe:	návěští SWAPF f,d ; komentář
Operandy:	f = (0-127) d = (0,1)
Operace:	f(b3-b0) = f(b7-b4) f(b7-b4) = f(b3-b0)
Ovlivňované příznaky:	žádné
Popis:	Jsou vzájemně zaměněny čtyři nižší a čtyři vyšší bity registru, když je zachováno jejich pořadí. Výsledek je umístěn do střadače pro d=0 nebo do registru pro d=1.

XORLW	eXclusive OR Lit. with W nonekvivalence střadače a konst.
Syntaxe:	návěští XORLW k ; komentář
Operandy:	k = (0-255)
Operace:	(W) = (W) XOR k
Ovlivňované příznaky:	Z
Popis:	Je provedena logická operace nonekvivalence mezi obsahem střadače a osmi bitovou konstantou. Výsledek je umístěn do střadače.

XORWF	eXclusive OR W with F nonekvivalence střadače a registru
Syntaxe:	návěští XORWF f,d ; komentář
Operandy:	f = (0-127) d = (0,1)
Operace:	(cílový registr) = (W) XOR (f)
Ovlivňované příznaky:	Z
Popis:	Je provedena logická operace nonekvivalence mezi obsahem střadače a obsahem registru. Výsledek je umístěn do střadače pro d=0 nebo do registru pro d=1.

17 VÝVOJOVÉ PROSTŘEDKY

Pro mikropočítače PICmicro® je k dispozici široká škála hardwarových i softwarových vývojových prostředků.

- Integrované vývojové prostředí: program MPLAB IDE
- Assemblery, kompilátory a spojovací programy: MPASM, MPLAB C17, MPLAB C18, MPLINK, MPLAB C30.
- Simulátory: MPLAB SIM, MPLAB dsPIC30
- Emulátory: MPLAB ICE2000, MPLAB ICE4000
- Debugger pro ladění v aplikaci: MPLAB ICD2
- Programátory: PRO MATE, PICSTART, MPLAB MP3,
- Demonstrační desky: PICDEM 1, PICDEM.net, PICDEM 2 plus, PICDEM 3, PICDEM 4, PICDEM 17, PICDEM 18R, PICDEM LIN, PICDEM USB
- Vývojové kity: (vložit bez přepisu)

17.1 Integrované vývojové prostředí MPLAB

Program umožňuje snadné ovládání a vývoj programů, který nebyl dříve na trhu s 8-bitovými jednočipovými mikropočítači k dispozici. Program obsahuje:

- 4 režimy práce:
- simulátor
- programátor (nutno dokoupit samostatně)
- emulátor (nutno dokoupit samostatně)
- debugger pro ladění v aplikaci (nutno dokoupit samostatně)
- Editor
- Správce projektů
- Práce pomocí oken s možností přímé editace

MPLAB IDE umožňuje:

- Editovat zdrojové soubory v jazyku assembler nebo C.
- Spuštění překladu jednou klávesou se současnou aktualizací celého projektu
- Ladění s použitím zdrojových souborů nebo výpisu překladače (listing)

MPLAB poskytuje ucelené prostředky pro plnohodnotnou vývojovou práci. Pozdější přechod od jednoduchého simulátoru přes jednoduchý debugger v aplikaci k plnohodnotnému emulátoru je velmi snadný.

17.2 Assembler MPASM

Makroassembler MPASM je překladač jazyka symbolických adres pro počítače PC.

Podporuje všechny typy jednočipových mikropočítačů PIC. MPASM má plnou podporu maker, umožňuje podmíněný překlad, různé formáty zdrojového textu a několik možností výpisů (listing). Pro potřebu různých vývojových prostředků vytváří různé výstupní formáty.

MPASM má následující funkce:

- Integrace do projektů MPLAB IDE
- Podpora maker
- Podmíněný překlad
- Direktivy umožňující řízení překladu

17.3 Kompilátory jazyka C MPLAB C17 a MPLAB C18

Kompilátory MPLAB C17 a MPLAB C18 jsou překladače jazyka ANSI C pro řadu jednočipových mikropočítačů PIC17xxx a PIC18xxx. Oba překladače jsou plně integrovatelné do vývojového prostředí MPLAB IDE.

17.4 Programy MPLINK a MPLIB

Program MPLINK spojuje přemístitelný zdrojový kód vytvořený v programu MPASM nebo MPLAB C17 nebo MPLAB C18. Může rovněž spojit zdrojový kód s dříve vytvořenými knihovnami prvků.

Program MPLIB spravuje knihovnu programových rutin, které lze použít při tvorbě zdrojového kódu. Pokud je použita ve zdrojovém kódu rutina z knihovny, je do zdrojového kódu vložen jen ten knihovní modul, který danou rutinu obsahuje.

- Knihovní objekty lze libovolně upravovat a rozlišovat.
- Programy MPLINK a MPLIB mají následující funkce:
- Snadné spojování mezi jednotlivými knihovnami
- Snadná správa pomocí seskupování příbuzných rutin
- Pohodlné vytváření, modifikace, přidávání rutin

17.5 Kompilátor jazyka C MPLAB C30

Program MPLAB C30 je plnohodnotný překladač jazyka ANSI-C pro řadu jednočipových mikropočítačů dsPIC30Fxxx. Překladač je plně integrovatelný do vývojového prostředí MPLAB IDE.

MPLAB C30 je dodáván s kompletními standardními knihovnami. Knihovny obsahují potřebné rutiny pro přenos, konverze, matematické operace a další základní operace.

17.6 Program MPLAB ASM30

Program MPLAB ASM30 je kompilátor jazyka symbolických adres pro řadu dsPIC30F.

Překladač generuje přenositelný kód který lze pomocí knihovního programu archivovat nebo spojit do spustitelného kódu.

17.7 Softwarový simulátor MPLAB-SIM

Tento program umožňuje vývoj programů pro jednočipové mikropočítače PIC na počítačích PC mimo laboratorní prostředí, tedy pouhou simulací činnosti. Uživatel může simulovat činnost mikropočítače na úrovni jednotlivých instrukcí, zastavovat na libovolných místech kódu, nahlížet nebo modifikovat libovolné proměnné v libovolnou chvíli. Rovněž je možné simulovat buzení vstupů mikropočítače z vnějšího prostředí. Výstupní formát je volitelný a je možné vykonávat program v režimu trasování nebo běhu do splnění podmínky.

17.8 Softwarový simulátor MPLAB SIM30

Tento program umožňuje simulování činnosti mikropočítačů řady dsPIC30F na počítačích PC. Tento simulátor plně podporuje ladění na symbolické úrovni s využitím překladače MPLAB ASM30. Simulátor může být využíván samostatně nebo jako součást prostředí MPLAB IDE.

17.9 Obvodový emulátor MPLAB ICE 2000

Jedná se o univerzální obvodový emulátor jednočipových mikropočítačů PIC. Je určen jako podpora pro kompletní vývoj aplikací s využitím integrovaného vývojového prostředí MPLAB IDE. Umožňuje editovat, vytvářet a spouštět program v aplikacích. Během běhu programu je možné monitorovat veškeré registry, program je možné krokovat nebo spouštět do splnění podmínky. Je možné trasovat paměť programu. Emulátor reaguje na reálné hodnoty na vstupech mikropočítače a naopak poskytuje vnějším obvodům reálnou odezvu na svých výstupech.

Program je určen pro OS Windows.

17.10 Obvodový emulátor MPLAB ICE 4000

Jedná se o univerzální obvodový emulátor jednočipových mikropočítačů PIC včetně řady PIC18xxx a dsPIC30F. Je určen jako podpora pro kompletní vývoj aplikací s využitím integrovaného vývojového prostředí MPLAB IDE. Umožňuje editovat, vytvářet a spouštět program v aplikacích. Během běhu programu je možné monitorovat veškeré registry, program je možné krokovat nebo spouštět do splnění podmínky. Je možné trasovat paměť programu. Emulátor reaguje na reálné hodnoty na vstupech mikropočítače a naopak poskytuje vnějším obvodům reálnou odezvu na svých výstupech.

Program je určen pro OS Windows.

17.11 Debugger pro ladění v aplikaci MPLAB ICD-2

Tento hardwarový nástroj umožňuje provádět ladění programu pro mikropočítače PIC přímo v koncové aplikaci. Zařízení je připojené k mikropočítači pomocí dvou vývodů, obdobně jako při programování ICSP. K počítači typu PC je zařízení připojeno pomocí rozhraní RS232C nebo pomocí USB. Pomocí integrovaného vývojového prostředí MPLAB IDE lze spouštět nebo krokovat program, sledovat stav všech registrů, případně tyto registry modifikovat. Práce probíhá v reálném čase v reálné aplikaci.

17.12 Univerzální programátor PRO MATE II

Tento programátor může pracovat jak samostatně tak ve spolupráci s PC. Programátor je vybaven displejem LCD pro zobrazování chybových hlášení, tlačítky a patičí pro připojování modulů s pouzdry mikropočítačů. Programátor umožňuje čtení, verifikaci a programování všech pamětí všech mikropočítačů řady PIC.

Rovněž je možné nastavovat konfiguraci včetně ochrany kódu proti vyčtení.

17.13 Programátor MPLAB MP3

Programátor MPLAB MP3 je univerzální programátor s funkcemi verifikace s programovatelným napájecím napětím. Je vybaven grafickým displejem o velikosti 128x64 bodů. Součástí dodávky je kabel pro ICSP. Programátor je schopen číst, programovat a verifikovat mikropočítače PIC bez připojení k PC. Rovněž je schopen konfigurovat ochranu kódu proti vyčtení. Programátor lze připojit k PC pomocí RS232 nebo USB. Programátor umožňuje realizovat rychlé programovací algoritmy, obsahuje vlastní paměť, rozšiřitelnou kartami SD/MMC.

17.14 Vývojový systém pro začátečníky PICSTART Plus

Tento programátor je jednoduchý a levný programátor všech mikropočítačů PIC s pouzdrum do 40 vývodů. Typy s více vývody lze programovat pomocí redukční patice.

Programátor se připojuje k počítači PC pomocí RS232 a ovládá se pomocí integrovaného prostředí MPLAB IDE.

17.15 Levná demonstrační deska PICDEM 1

Jedná se o jednoduchou desku která demonstruje možnosti některých mikropočítačů PIC (PIC16C54 až PIC16C58A, PIC16C61, PIC16C62X, PIC16C71, PIC16C8X, PIC17C42, PIC17C43 a PIC17C44). Deska obsahuje veškerý potřebný hardware pro práci.

Uživatel může pro práci použít vzorkové množství mikropočítačů dodávaných s deskou. Desku lze připojit k emulátoru MPLAB-ICE. K dispozici je univerzální motiv plošného spoje pro realizaci vlastního zapojení. K dispozici je dále rozhraní RS 232C, potenciometr pro realizaci analogového vstupu, tlačítka a 8 LED připojených na PORTB.

17.16 Demonstrační deska PICDEM.net Internet/Ethernet

Jedná se o desku s mikropočítačem PIC18F452 s dodávaným firmware pro realizaci protokolu TCP/IP. Deska podporuje použití libovolného 40-vývodového mikropočítače od PIC16F877 do PIC18C452. Kit obsahuje realizaci TCP/IP, webový server s podporou HTML, sériovou paměť 24L256 pro stahování dat z webu, konektor pro připojení k ICSP nebo MPLAB ICD 2, připojení k Ethernet, rozhraní RS232C a dvouřádkový šestnáctiznakový LCD displej. K desce je dodávána dokumentace a CD.

17.17 Demonstrační deska PICDEM 2 Plus

Jedná se o desku podporující většinu 18, 28 a 40 vývodových mikropočítačů včetně PIC16F87x a PIC18Fxx2. Veškerý nezbytný hardware i software pro demonstraci základních operací je součástí dodávky. Dodávané vzorky mikropočítačů je možné programovat pomocí programátorů PRO MATE II, PICSTART Plus nebo pomocí MPLAB ICD 2 pomocí standardního rozhraní. Desku lze využít rovněž s emulátorem MPLAB-ICE nebo MPLAB ICD 2. Na desce je univerzální motiv plošného spoje pro realizaci vlastního zapojení, rozhraní RS232C, LCD displej, teplotní senzor a 4 LED.

Součástí dodávky jsou vzorky mikropočítačů PIC18F452 a PIC16F877.

17.18 Demonstrační deska PICDEM 3 PIC16C92x

Jedná se o desku podporující mikropočítače PIC16C923 a PIC16C924 v pouzdrech PLCC. Součástí dodávky je veškerý potřebný hardware i software potřebný ke spuštění demonstračních programů.

17.19 Demonstrační deska PICDEM 4 pro pouzdra 8/14/18 vývodů

Desku je možné použít pro demonstraci možností mikropočítačů v pouzdrech s 8, 14 nebo 18 vývody řady PIC16xxx nebo PIC18xxx včetně typů PIC16F818/819, PIC16F87/88, PIC 16F62XA a PIC18F1320. Na desce je možné demonstrovat řadu vlastností těchto mikropočítačů včetně LIN a ovládání motoru pomocí ECCP. Díky možnosti odpojit řadu obvodů lze demonstrovat i nízký odběr mikropočítačů v režimu se sníženým odběrem. Taktovací oscilátor lze volit krystalový, RC nebo vnitřní. Deska obsahuje stabilizátor napětí pro napájení z baterie 9V. Deska je vybavena rozhraním RS232C, konektorem ICD pro programování pomocí ICSP nebo pro připojení k MPLAB ICD 2, LCD displejem, motivem plošného spoje pro realizaci můstkového zapojení pro řízení motoru, komunikační rozhraní LIN a paměti EEPROM.

Na desce je rovněž 8 LED, 4 potenciometry, tři tlačítka a univerzální motiv plošného spoje. Součástí dodávky jsou mikropočítače PIC16F627A a PIC18F1320.

17.20 Demonstrační deska PICDEM 17

Jedná se o desku podporující mikropočítače PIC17C752, PIC17C756A, PIC17C762 a PIC17C766. Součástí dodávky jsou naprogramované vzorky mikropočítačů. Pro vlastní programování lze použít standardní programátory PRO MATE II nebo PICSTART Plus.

Deska umožňuje spouštět programy umístěné v externí paměti Flash. Uživatel má rovněž k dispozici univerzální motiv plošného spoje.

17.21 Demonstrační deska PICDEM 18R pro PIC18C601/801

Deska je určena pro vývojové práce s řadou pro PIC18C601/801. Umožňuje použití všech hardwarových módů mikropočítačů. Deska obsahuje 2 Mb externí paměti typu Flash, 128 Kb paměti SRAM a sériovou paměť EEPROM. Deska umožňuje využít i vyšší adresovatelný prostor řady PIC18C601/801.

17.22 Demonstrační deska PICDEM LIN PIC16C43x

Součástí kitu je hardwarová i softwarová podpora práce s LIN. Souprava obsahuje řadu jednoduchých desek s motivem plošného spoje a tři mikropočítače. PIC16C432 a PIC16C433 jsou použity v síti LIN jako podřízené (slave), zatímco mikropočítač PIC16F874 ovládá sběrnici (master). Ve všech třech mikropočítačích je nahrán firmware realizující LIN.

17.23 PICkit™ 1 Flash Starter Kit

Souprava obsahuje vše nezbytné pro zahájení práce s 8 nebo 14 vývodovými mikropočítači PIC. Deska umožňuje programování, ladění a vývoj, komunikace s PC je realizována pomocí USB. Součástí dodávky je CD s aplikačními programy, ukázkami a návody a integrované vývojové prostředí MPLAB IDE. Deska podporuje všechny vyráběné mikropočítače v pouzdrech 8 a 14 vývodů a bude podporovat i většinu nových mikropočítačů PIC do budoucna.

17.24 Demonstrační deska PICDEM USB PIC16C7x5

Tato deska ukazuje možnosti mikropočítačů PIC 16C745 a PIC16C765 používajících rozhraní USB.

17.25 Další vývojové prostředky

Firma Microchip poskytuje podporu veškerému svému dalšímu sortimentu. Podrobnosti a aktuální nabídka je uvedena na stránkách Microchip.com.

PIC16F87/88

Poznámky:

18 ELEKTRICKÉ CHARAKTERISTIKY

Mezní parametry

Pracovní teplota okolí	-40 °C až +125 °C
Skladovací teplota	-65 °C až +150 °C
Napětí na libovolném vývodu (kromě VDD a MCLR) proti Vss.....	-0,3 V až (VDD + 0,3 V)
Napájecí napětí na VDD proti Vss.....	-0,3 až 7,5 V
Napětí na vstupu MCLR proti Vss (Poznámka 2)	-0,3 až 14 V
Celkový příkon (Poznámka 1)	1 W
Maximální proud z vývodu Vss.....	200 mA
Maximální proud do vývodu VDD	200 mA
Vstupní proud ochrannými omezovacími diodami	±20 mA
Výstupní proud ochrannými omezovacími diodami.....	±20 mA
Maximální proud vtékající do libovolného vývodu.....	25 mA
Maximální proud vytékající z libovolného vývodu	25 mA
Maximální celkový proud vtékající současně do brány PORTA	100 mA
Maximální celkový proud vytékající současně z brány PORTA	100 mA
Maximální celkový proud vtékající současně do brány PORTB	100 mA
Maximální celkový proud vytékající současně z brány PORTB	100 mA

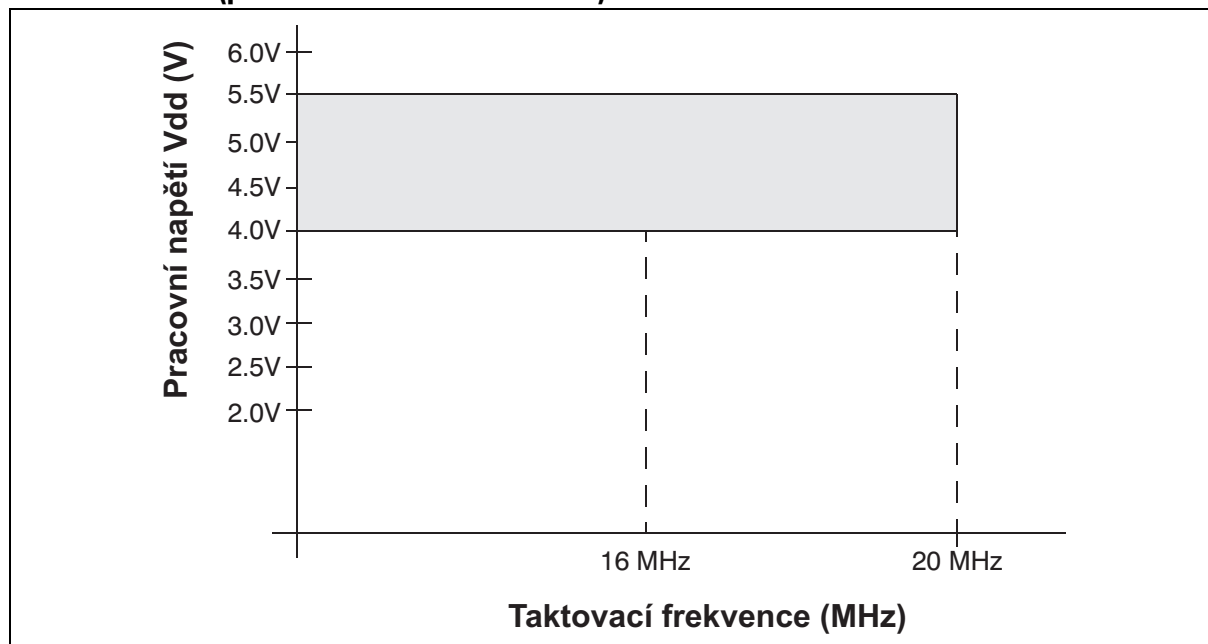
Poznámka 1: Celková výkonová ztráta je stanovena podle vzorce:
$$P_{dis} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$$

Poznámka 2: Pokles napětí na vývodu MCLR může být zachycen jako aktivní signál reset. Proto je doporučeno připojovat tento vývod na napětí Vss přes odpor min. 1 KΩ.

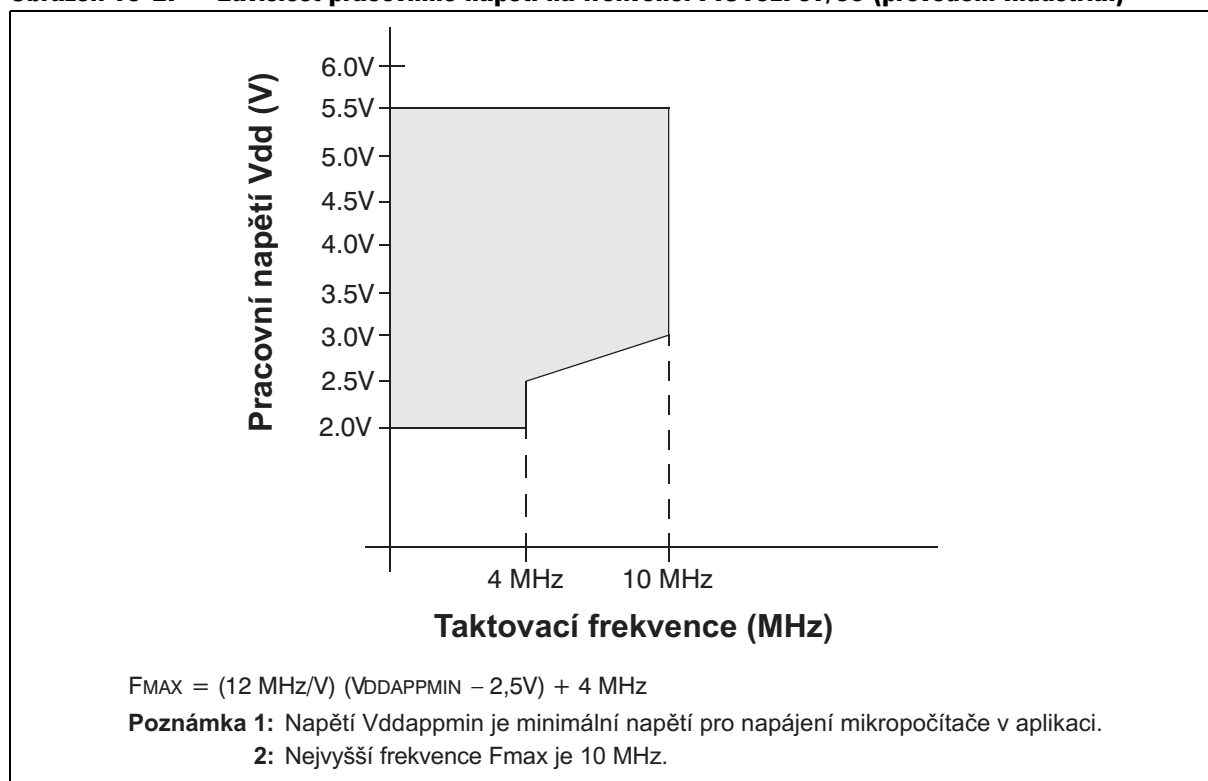
Upozornění: Hodnoty vyšší, než v přehledu uvedené jako maximální hodnota mohou vyvolat trvalé poškození součástky. Jedná se o hodnoty maximální za uvedených podmínek, nikoli o hodnoty pracovní. Při delším působení uvedených extrémních úrovní jednotlivých vlivů může dojít ke snížení spolehlivosti součástky.

PIC16F87/88

Obrázek 18-1: Závislost pracovního napětí na frekvenci PIC16F87/88 (provedení Industrial a Extended)



Obrázek 18-2: Závislost pracovního napětí na frekvenci PIC16LF87/88 (provedení Industrial)



18.1 DC parametry: Napájecí napětí

PIC16F87/88 (provedení "Industrial" a "Extended")

PIC16LF87/88 (provedení "Industrial")

PIC16LF87/88 (Industrial)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial"					
PIC16F87/88 (Industrial, Extended)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial" pracovní rozsah -40 až +125 °C pro provedení "Extended"					
Parametr	Značení	Charakteristika	Min.	Typický	Max.	Jednotky	Podmínky
D001	V _{DD}	Napájecí napětí					
		PIC16LF87/88	2,0	—	5,5	V	Konfigurace oscilátoru HS, XT, RC a LP
D001		PIC16F87/88	4,0	—	5,5	V	
D002	V _{DR}	Napětí pro zachování dat v RWM ⁽¹⁾	1,5	—	—	V	
D003	V _{POR}	Mezní napětí V_{DD} pro zajištění resetu POR		—	0,7	V	Podrobněji viz kapitolu 15.4
D004	S _{VDD}	Rychlost náběhu napětí V_{DD} pro zajištění resetu POR	0,05	—	—	V/ms	Podrobněji viz kapitolu 15.4
D005	V _{BOR}	Napětí pro aktivaci resetu BOR					
		PIC16LF87/88	3,65	—	4,35	V	
		PIC16F87/88	3,65	—	4,35	V	F _{MAX} = 14 MHz ⁽²⁾

Význam: Řádky odpovídající obvodům se standardním rozsahem napětí jsou stínované

Poznámka 1: Toto je mez napájecího napětí, na kterou může V_{DD} poklesnout aniž by došlo ke ztrátě dat v paměti RWM.

2: Při povoleném obvodu BOR bude obvod funkční, pokud napětí neklesne pod napětí aktivace resetu BOR

18.2 DC parametry: Napájecí proud v režimu Sleep PIC16F87/88 (provedení "Industrial" a "Extended") PIC16LF87/88 (provedení "Industrial")

PIC16LF87/88 (Industrial)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial"					
PIC16F87/88 (Industrial, Extended)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial" pracovní rozsah -40 až +125 °C pro provedení "Extended"					
Parametr	Mikropočítač	Typický	Max.	Jednotky	Podmínky		
	Napájecí proud v režimu Sleep ⁽¹⁾						
	PIC16LF87/88	0,1	0,4	μA	-40 °C	V _{DD} = 2,0 V	
		0,1	0,4	μA	+25 °C		
		0,4	1,5	μA	+85 °C		
	PIC16LF87/88	0,3	0,5	μA	-40 °C	V _{DD} = 3,0 V	
		0,3	0,5	μA	+25 °C		
		0,7	1,7	μA	+85 °C		
	Všechny typy	0,6	1,0	μA	-40 °C	V _{DD} = 5,0 V	
		0,6	1,0	μA	+25 °C		
		1,2	5,0	μA	+85 °C		
V provedení Extended	6	25	μA	+125 °C			

Význam: Řádky odpovídající obvodům se standardním rozsahem napětí jsou stínované

Poznámka 1: Proudová spotřeba v režimu Sleep se měří zásadně pokud je součástka v režimu Sleep a všechny vývody jsou uvedeny do třetího stavu a připojeny k V_{DD} nebo V_{SS}. Spotřeba nezávisí na typu oscilátoru. Napájení obvodů Watchdog, BOD, Timer1 a pod. je zakázáno.

- 2:** Spotřeba proudu závisí především na velikosti napájecího napětí a taktovací frekvenci. Dále spotřeba závisí na zatížení sběrnic, typu oscilátoru, typu vykonávaných instrukcí a teplotě pouzdra součástky. Testovací podmínky pro měření proudu I_{DD} v aktivním režimu jsou:
Taktování pomocí vnějšího signálu na OSC1, všechny vývody jsou ve třetím stavu a připojené na V_{DD}, vstup MCLR je připojen na V_{DD}. Stav časovače Watchdog (povolen – zakázán) je nutné specifikovat.
- 3:** Pro konfiguraci oscilátoru RC není do spotřeby zahrnut proud protékající odporem R_{EXT}. Tento proud lze vypočítat podle vztahu: (Hodnota R_{EXT} je uvedena v kΩ)

18.2 DC parametry: Napájecí proud v režimu Sleep PIC16F87/88 (provedení "Industrial" a "Extended") PIC16LF87/88 (provedení "Industrial") – pokračování

PIC16LF87/88 (Industrial)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial"					
PIC16F87/88 (Industrial, Extended)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial" pracovní rozsah -40 až +125 °C pro provedení "Extended"					
Parametr	Mikropočítač	Typický	Max.	Jednotky	Podmínky		
	Napájecí proud v režimu Sleep ^(2,3)						
	PIC16LF87/88	9	20	μA	-40 °C	VDD = 2,0 V	Fosc = 32 kHz (Konfigurace oscilátoru LP)
		7	15	μA	+25 °C		
		7	15	μA	+85 °C		
	PIC16LF87/88	16	30	μA	-40 °C	VDD = 3,0 V	
		14	25	μA	+25 °C		
		14	25	μA	+85 °C		
	Všechny typy	32	40	μA	-40 °C	VDD = 5,0 V	
		26	35	μA	+25 °C		
		26	35	μA	+85 °C		
V provedení Extended	35	53	μA	+125 °C			

Význam: Řádky odpovídající obvodům se standardním rozsahem napětí jsou stínované

Poznámka 1: Proudová spotřeba v režimu Sleep se měří zásadně pokud je součástka v režimu Sleep a všechny vývody jsou uvedeny do třetího stavu a připojeny k V_{DD} nebo V_{SS}. Spotřeba nezávisí na typu oscilátoru. Napájení obvodů Watchdog, BOD, Timer1 a pod. je zakázáno.

- Spotřeba proudu závisí především na velikosti napájecího napětí a taktovací frekvenci. Dále spotřeba závisí na zatížení sběrnic, typu oscilátoru, typu vykonávaných instrukcí a teplotě pouzdra součástky. Testovací podmínky pro měření proudu I_{DD} v aktivním režimu jsou:
Taktování pomocí vnějšího signálu na OSC1, všechny vývody jsou ve třetím stavu a připojené na V_{DD}, □ vstup MCLR je připojen na V_{DD}. Stav časovače Watchdog (povolen – zakázán) je nutné specifikovat.
- Pro konfiguraci oscilátoru RC není do spotřeby zahrnut proud protékající odporem R_{EXT}. Tento proud lze vypočítat podle vztahu: (Hodnota R_{EXT} je uvedena v kΩ)

18.2 DC parametry: Napájecí proud v režimu Sleep

PIC16F87/88 (provedení "Industrial" a "Extended")

PIC16LF87/88 (provedení "Industrial") – pokračování

PIC16LF87/88 (Industrial)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial"					
PIC16F87/88 (Industrial, Extended)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial" pracovní rozsah -40 až +125 °C pro provedení "Extended"					
Parametr	Mikropočítač	Typický	Max.	Jednotky	Podmínky		
	Napájecí proud v režimu Sleep ^(2,3)						
	PIC16LF87/88	72	95	μA	-40 °C	V _{DD} = 2,0 V	Fosc = 1 MHz (Konfigurace oscilátoru RC)
		76	90	μA	+25 °C		
		76	90	μA	+85 °C		
	PIC16LF87/88	138	175	μA	-40 °C	V _{DD} = 3,0 V	
		136	170	μA	+25 °C		
		136	170	μA	+85 °C		
	Všechny typy	310	380	μA	-40 °C	V _{DD} = 5,0 V	
		290	360	μA	+25 °C		
		280	360	μA	+85 °C		
	V provedení Extended	330	500	μA	+125 °C		
	PIC16LF87/88	270	335	μA	-40 °C	V _{DD} = 2,0 V	Fosc = 4 MHz (Konfigurace oscilátoru RC)
		280	330	μA	+25 °C		
		285	330	μA	+85 °C		
	PIC16LF87/88	460	610	μA	-40 °C	V _{DD} = 3,0 V	
		450	600	μA	+25 °C		
		450	600	μA	+85 °C		
	Všechny typy	900	1060	μA	-40 °C	V _{DD} = 5,0 V	
		890	1050	μA	+25 °C		
		890	1050	μA	+85 °C		
V provedení Extended	920	1,5	mA	+125 °C			

Význam: Řádky odpovídající obvodům se standardním rozsahem napětí jsou stínované

Poznámka 1: Proudová spotřeba v režimu Sleep se měří zásadně pokud je součástka v režimu Sleep a všechny vývody jsou uvedeny do třetího stavu a připojeny k V_{DD} nebo V_{SS}. Spotřeba nezávisí na typu oscilátoru. Napájení obvodů Watchdog, BOD, Timer1 a pod. je zakázáno.

2: Spotřeba proudu závisí především na velikosti napájecího napětí a taktovací frekvenci. Dále spotřeba závisí na zatížení sběrnic, typu oscilátoru, typu vykonávaných instrukcí a teplotě pouzdra součástky. Testovací podmínky pro měření proudu I_{DD} v aktivním režimu jsou:

Taktování pomocí vnějšího signálu na OSC1, všechny vývody jsou ve třetím stavu a připojené na V_{DD}, vstup MCLR je připojen na V_{DD}. Stav časovače Watchdog (povolen – zakázán) je nutné specifikovat.

3: Pro konfiguraci oscilátoru RC není do spotřeby zahrnut proud protékající odporem R_{EXT}. Tento proud lze vypočítat podle vztahu: (Hodnota R_{EXT} je uvedena v kΩ)

18.2 DC parametry: Napájecí proud v režimu Sleep

PIC16F87/88 (provedení "Industrial" a "Extended")

PIC16LF87/88 (provedení "Industrial") – pokračování

PIC16LF87/88 (Industrial)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial"					
PIC16F87/88 (Industrial, Extended)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial" pracovní rozsah -40 až +125 °C pro provedení "Extended"					
Parametr	Mikroprocesor	Typický	Max.	Jednotky	Podmínky		
	Napájecí proud v režimu Sleep (2,3)						
	Všechny typy	1,8	2,3	mA	-40 °C	V _{DD} = 4,0 V	Fosc = 32 kHz (Konfigurace oscilátoru LP)
		1,6	2,2	mA	+25 °C		
		1,3	2,2	mA	+85 °C		
	Všechny typy	3,0	4,2	mA	-40 °C	V _{DD} = 5,0 V	
		2,5	4,0	mA	+25 °C		
		2,5	4,0	mA	+85 °C		
	V provedení Extended	3,0	5,0	mA	+85 °C		

Význam: Řádky odpovídající obvodům se standardním rozsahem napětí jsou stínované

Poznámka 1: Proudová spotřeba v režimu Sleep se měří zásadně pokud je součástka v režimu Sleep a všechny vývody jsou uvedeny do třetího stavu a připojeny k V_{DD} nebo V_{SS}. Spotřeba nezávisí na typu oscilátoru. Napájení obvodů Watchdog, BOD, Timer1 a pod. je zakázáno.

- Spotřeba proudu závisí především na velikosti napájecího napětí a taktovací frekvenci. Dále spotřeba závisí na zatížení sběrnic, typu oscilátoru, typu vykonávaných instrukcí a teplotě pouzdra součástky. Testovací podmínky pro měření proudu I_{DD} v aktivním režimu jsou:
Taktování pomocí vnějšího signálu na OSC1, všechny vývody jsou ve třetím stavu a připojené na V_{DD}, vstup MCLR je připojen na V_{DD}. Stav časovače Watchdog (povolen – zakázán) je nutné specifikovat.
- Pro konfiguraci oscilátoru RC není do spotřeby zahrnut proud protékající odporem R_{EXT}. Tento proud lze vypočítat podle vztahu: (Hodnota R_{EXT} je uvedena v kΩ)

18.2 DC parametry: Napájecí proud v režimu Sleep

PIC16F87/88 (provedení "Industrial" a "Extended")

PIC16LF87/88 (provedení "Industrial") – pokračování

PIC16LF87/88 (Industrial)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial"				
PIC16F87/88 (Industrial, Extended)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial" pracovní rozsah -40 až +125 °C pro provedení "Extended"				
Parametr	Mikropočítač	Typický	Max.	Jednotky	Podmínky	
	Napájecí proud v režimu Sleep ^(2,3)					Fosc = 1 MHz (Konfigurace oscilátoru: vnitřní RC, režim RC_RUN)
	PIC16LF87/88	8	20	μA	-40 °C	
		7	15	μA	+25 °C	
		7	15	μA	+85 °C	
	PIC16LF87/88	16	30	μA	-40 °C	
		14	25	μA	+25 °C	
		14	25	μA	+85 °C	
	Všechny typy	32	40	μA	-40 °C	
		29	35	μA	+25 °C	
		29	35	μA	+85 °C	
	V provedení Extended	35	45	μA	+125 °C	
	PIC16LF87/88	132	160	μA	-40 °C	
		126	155	μA	+25 °C	
		126	155	μA	+85 °C	
	PIC16LF87/88	260	310	μA	-40 °C	
		230	300	μA	+25 °C	
		230	300	μA	+85 °C	
	Všechny typy	560	690	μA	-40 °C	
		500	650	μA	+25 °C	
		500	650	μA	+85 °C	
V provedení Extended	570	710	mA	+125 °C		

Význam: Řádky odpovídající obvodům se standardním rozsahem napětí jsou stínované

Poznámka 1: Proudová spotřeba v režimu Sleep se měří zásadně pokud je součástka v režimu Sleep a všechny vývody jsou uvedeny do třetího stavu a připojeny k V_{DD} nebo V_{SS}. Spotřeba nezávisí na typu oscilátoru. Napájení obvodů Watchdog, BOD, Timer1 a pod. je zakázáno.

- 2:** Spotřeba proudu závisí především na velikosti napájecího napětí a taktovací frekvenci. Dále spotřeba závisí na zatížení sběrnic, typu oscilátoru, typu vykonávaných instrukcí a teplotě pouzdra součástky. Testovací podmínky pro měření proudu I_{DD} v aktivním režimu jsou:
Taktování pomocí vnějšího signálu na OSC1, všechny vývody jsou ve třetím stavu a připojené na V_{DD}, vstup MCLR je připojen na V_{DD}. Stav časovače Watchdog (povolen – zakázán) je nutné specifikovat.
- 3:** Pro konfiguraci oscilátoru RC není do spotřeby zahrnut proud protékající odporem R_{EXT}. □
Tento proud lze vypočítat podle vztahu: (Hodnota R_{EXT} je uvedena v kΩ)

18.2 DC parametry: Napájecí proud v režimu Sleep

PIC16F87/88 (provedení "Industrial" a "Extended")

PIC16LF87/88 (provedení "Industrial") – pokračování

PIC16LF87/88 (Industrial)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial"					
PIC16F87/88 (Industrial, Extended)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial" pracovní rozsah -40 až +125 °C pro provedení "Extended"					
Parametr	Mikropočítač	Typický	Max.	Jednotky	Podmínky		
	Napájecí proud v režimu Sleep ^(2,3)						
	PIC16LF87/88	310	420	μA	-40 °C	VDD = 2,0 V	Fosc = 32 kHz (Konfigurace oscilátoru: vnitřní RC, režim RC_RUN)
		300	410	μA	+25 °C		
		300	410	μA	+85 °C		
	PIC16LF87/88	550	650	μA	-40 °C	VDD = 3,0 V	
		530	620	μA	+25 °C		
		530	620	μA	+85 °C		
	Všechny typy	1,2	1,5	mA	-40 °C	VDD = 5,0 V	
		1,1	1,4	mA	+25 °C		
		1,1	1,4	mA	+85 °C		
V provedení Extended	1,3	1,6	mA	+125 °C			

Význam: Řádky odpovídající obvodům se standardním rozsahem napětí jsou stínované

Poznámka 1: Proudová spotřeba v režimu Sleep se měří zásadně pokud je součástka v režimu Sleep a všechny vývody jsou uvedeny do třetího stavu a připojeny k V_{DD} nebo V_{SS}. Spotřeba nezávisí na typu oscilátoru. Napájení obvodů Watchdog, BOD, Timer1 a pod. je zakázáno.

- 2:** Spotřeba proudu závisí především na velikosti napájecího napětí a taktovací frekvenci. Dále spotřeba závisí na zatížení sběrnic, typu oscilátoru, typu vykonávaných instrukcí a teplotě pouzdra součástky. Testovací podmínky pro měření proudu I_{DD} v aktivním režimu jsou:
Taktování pomocí vnějšího signálu na OSC1, všechny vývody jsou ve třetím stavu a připojené na V_{DD}, vstup MCLR je připojen na V_{DD}. Stav časovače Watchdog (povolen – zakázán) je nutné specifikovat.
- 3:** Pro konfiguraci oscilátoru RC není do spotřeby zahrnut proud protékající odporem R_{EXT}. Tento proud lze vypočítat podle vztahu: $I_r = V_{DD}/2R_{EXT}$ (mA) (Hodnota R_{EXT} je uvedena v kW)

18.2 DC parametry: Napájecí proud v režimu Sleep

PIC16F87/88 (provedení "Industrial" a "Extended")

PIC16LF87/88 (provedení "Industrial") – pokračování

PIC16LF87/88 (Industrial)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial"						
PIC16F87/88 (Industrial, Extended)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial" pracovní rozsah -40 až +125 °C pro provedení "Extended"						
Parametr	Mikropočítač	Typický	Max.	Jednotky	Podmínky			
	Napájecí proud v režimu Sleep ^(2,3)							
	PIC16LF87/88	,950	1,3	mA	-40 °C	VDD = 3,0 V	Fosc = 1 MHz (Konfigurace oscilátoru: vnitřní RC, režim RC_RUN)	
		,930	1,2	mA	+25 °C			
		,930	1,2	mA	+85 °C			
	Všechny typy	1,8	3,0	mA	-40 °C	VDD = 5,0 V		
		1,7	2,8	mA	+25 °C			
		1,7	2,8	mA	+85 °C			
	V provedení Extended	2,0	4,0	mA	+125 °C			
	PIC16LF87/88	9	13	μA	-40 °C	VDD = 2,0 V		Fosc = 4 MHz (Konfigurace oscilátoru: Timer1, režim SEC_RUN)
		9	14	μA	+25 °C			
		11	16	μA	+85 °C			
	PIC16LF87/88	12	34	μA	-40 °C	VDD = 3,0 V		
		12	31	μA	+25 °C			
		14	28	μA	+85 °C			
	Všechny typy	20	72	μA	-40 °C	VDD = 5,0 V		
		20	65	μA	+25 °C			
25		59	μA	+85 °C				

Význam: Řádky odpovídající obvodům se standardním rozsahem napětí jsou stínované

Poznámka 1: Proudová spotřeba v režimu Sleep se měří zásadně pokud je součástka v režimu Sleep a všechny vývody jsou uvedeny do třetího stavu a připojeny k V_{DD} nebo V_{SS}. Spotřeba nezávisí na typu oscilátoru. Napájení obvodů Watchdog, BOD, Timer1 a pod. je zakázáno.

- 2:** Spotřeba proudu závisí především na velikosti napájecího napětí a taktovací frekvenci. Dále spotřeba závisí na zatížení sběrnic, typu oscilátoru, typu vykonávaných instrukcí a teplotě pouzdra součástky. Testovací podmínky pro měření proudu I_{DD} v aktivním režimu jsou:
Taktování pomocí vnějšího signálu na OSC1, všechny vývody jsou ve třetím stavu a připojené na V_{DD}, vstup MCLR je připojen na V_{DD}. Stav časovače Watchdog (povolen – zakázán) je nutné specifikovat.
- 3:** Pro konfiguraci oscilátoru RC není do spotřeby zahrnut proud protékající odporem R_{EXT}. Tento proud lze vypočítat podle vztahu: (Hodnota R_{EXT} je uvedena v kΩ)

18.2 DC parametry: Napájecí proud v režimu Sleep PIC16F87/88 (provedení "Industrial" a "Extended") PIC16LF87/88 (provedení "Industrial") – pokračování

PIC16LF87/88 (Industrial)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial"					
PIC16F87/88 (Industrial, Extended)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial" pracovní rozsah -40 až +125 °C pro provedení "Extended"					
Parametr	Mikropočítač	Typický	Max.	Jednotky	Podmínky		
D022 (ΔI_{wdt})	Napájecí proudy jednotlivých obvodů (ΔI_{wdt}, ΔI_{BOR}, ΔI_{LVD}, ΔI_{OSCB}, ΔI_{AD})						
	Časovač Watchdog	1,5	3,8	μA	-40 °C	$V_{DD} = 2,0 V$	
		2,2	3,8	μA	+25 °C		
		2,7	4,0	μA	+85 °C		
		2,3	4,6	μA	-40 °C	$V_{DD} = 3,0 V$	
		2,7	4,6	μA	+25 °C		
		3,1	4,8	μA	+85 °C		
		3,0	10,0	μA	-40 °C	$V_{DD} = 5,0 V$	
		3,3	10,0	μA	+25 °C		
		3,9	13,0	μA	+85 °C		
V provedení Extended	5,0	21,0	μA	+125 °C			
D022A (ΔI_{BOR})	Detekce poklesu napájecího napětí BOD	40	60	μA	-40 °C až +85 °C	$V_{DD} = 5,0 V$	
D025 (ΔI_{OSCB})	Oscilátor čítače/ časovače Timer1	1,7	2,3	μA	-40 °C	$V_{DD} = 2,0 V$	Oscilátor Timer1 na frekvenci 32 kHz
		1,8	2,3	μA	+25 °C		
		2,0	2,3	μA	+85 °C		
		2,2	3,8	μA	-40 °C	$V_{DD} = 3,0 V$	
		2,6	3,8	μA	+25 °C		
		2,9	3,8	μA	+85 °C		
		3,0	6,0	μA	-40 °C	$V_{DD} = 5,0 V$	
		3,2	6,0	μA	+25 °C		
		3,4	7,0	μA	+85 °C		
D026 (ΔI_{AD})	AD převodník	0,001	2,0	μA	-40 °C až +85 °C		AD převodník zapnut, neprobíhá převod, v režimu Sleep
		0,001	2,0	μA	-40 °C až +85 °C		
		0,003	2,0	μA	-40 °C až +85 °C		
	V provedení Extended	4,0	8,0	μA	-40 °C až +85 °C		

Význam: Řádky odpovídající obvodům se standardním rozsahem napětí jsou stínované

Poznámka 1: Proudová spotřeba v režimu Sleep se měří zásadně pokud je součástka v režimu Sleep a všechny vývody jsou uvedeny do třetího stavu a připojeny k V_{DD} nebo V_{SS} . Spotřeba nezávisí na typu oscilátoru. Napájení obvodů Watchdog, BOD, Timer1 a pod. je zakázáno.

2: Spotřeba proudu závisí především na velikosti napájecího napětí a taktovací frekvenci. Dále spotřeba závisí na zatížení sběrnic, typu oscilátoru, typu vykonávaných instrukcí a teplotě pouzdra součástky. Testovací podmínky pro měření proudu I_{DD} v aktivním režimu jsou:
Taktování pomocí vnějšího signálu na OSC1, všechny vývody jsou ve třetím stavu a připojené na V_{DD} , vstup $\overline{MCLR}$ je připojen na V_{DD} . Stav časovače Watchdog (povolen – zakázán) je nutné specifikovat.

3: Pro konfiguraci oscilátoru RC není do spotřeby zahrnut proud protékající odporem R_{EXT} . Tento proud lze vypočítat podle vztahu: (Hodnota R_{EXT} je uvedena v $k\Omega$)

18.3 DC parametry: Přesnost vnitřního RC oscilátoru PIC16F87/88 (provedení "Industrial" a "Extended") PIC16LF87/88 (provedení "Industrial")

PIC16LF87/88 (Industrial)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial"				
PIC16F87/88 (Industrial, Extended)		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial" pracovní rozsah -40 až +125 °C pro provedení "Extended"				
Parametr	Mikropočítač	Min.	Typický	Max.	Jednotky	Podmínky
Přesnost oscilátoru INTOSC při kmitočtu Freq = 8 MHz, 4 MHz, 2 MHz, 1 MHz, 500 kHz, 250 kHz, 125 kHz ⁽¹⁾						
	PIC16LF87/88	-2	±1	2	%	+25 °C
		-5	—	5	%	-10 °C až +85 °C
		-10	—	10	%	-40 °C až +85 °C
	PIC16LF87/88	-2	±1	2	%	+25 °C
		-5	—	5	%	-10 °C až +85 °C
		-10	—	10	%	-40 °C až +85 °C
	V provedení Extended	-15	—	15	%	-40 °C až +125 °C
Přesnost oscilátoru INTRC při kmitočtu Freq = 31 kHz ⁽²⁾						
	PIC16LF87/88	26,562	—	35,938	kHz	-40 °C až +85 °C
	PIC16F87/88	26,562	—	35,938	kHz	-40 °C až +85 °C

Význam: Řádky odpovídající obvodům se standardním rozsahem napětí jsou stínované

Poznámka 1: Frekvence oscilátoru je kalibrována při teplotě 25 °C.
Pro doladění frekvence při jiné teplotě lze použít registr OSCTUNE.

2: Frekvence INTRC po kalibraci.

18.4 DC charakteristiky: PIC16F87/88 (provedení "Industrial" a "Extended") PIC16LF87/88(provedení "Industrial")

DC parametry		Běžné pracovní podmínky (pokud není uvedeno jinak) pracovní rozsah -40 až +85 °C pro provedení "Industrial" pracovní rozsah -40 až +125 °C pro provedení "Extended" Rozsah napájecího napětí podle DC charakteristik					
Parametr	Značení	Charakteristika	Min.	Typ. ♦	Max.	Jedn.	Podmínky
D030 D030A D031 D032 D033 D034	Vil	Vstupní napětí v úrovni log.0					
		vstupně výstupní vývody: se vstupem TTL	Vss	—	0,15 V _{DD}	V	Pro celý rozsah napájecího napětí Pro napájecí napětí od 4,5 do 5,5 V
			Vss	—	0,8 V	V	
		se Schmittovým klopným obvodem na vstupu	Vss	—	0,2 V _{DD}	V	
		MCLR, OSC1 v konfiguraci RC	Vss	—	0,2 V _{DD}	V	(Poznámka 1)
		OSC1 v konfiguraci XT a LP	Vss	—	0,3 V	V	
		OSC1 v konfiguraci HS	Vss	—	0,3 V _{DD}	V	
		Vývody RB1 a RB4: Se Schmittovým klopným obvodem na vstupu	Vss	—	0,3 V _{DD}	V	Pro celý rozsah napájecího napětí
D040 D040A D041 D042 D042A D043 D044	Vih	Vstupní napětí v úrovni log.1					
		vstupně výstupní vývody: se vstupem TTL	2,0	—	V _{DD}	V	Pro napájecí napětí od 4,5 do 5,5 V Pro celý rozsah napájecího napětí Pro celý rozsah napájecího napětí
			0,25 V _{DD} + 0,8 V	—	V _{DD}	V	
		se Schmittovým klopným obvodem na vstupu	0,8 V _{DD}	—	V _{DD}	V	
		MCLR	0,8 V _{DD}	—	V _{DD}	V	(Poznámka 1)
		OSC1 v konfiguraci XT a LP	1,6 V	—	V _{DD}	V	
		OSC1 v konfiguraci HS	0,7 V _{DD}	—	V _{DD}	V	
		OSC1 v konfiguraci RC	0,9 V _{DD}	—	V _{DD}	V	Pro celý rozsah napájecího napětí
D070	I _{purb}	Proud odporu pull-up					
			50	250	400	μA	V _{DD} = 5V, V _{PIN} = Vss
D060 D061 D063	I _{il}	Vstupní svodový proud (2,3)					
		vstupně výstupní vývody	—	—	±1	μA	Vss ≤ V _{PIN} ≤ V _{DD} , vývod je ve stavu vysoké impedance
		MCLR	—	—	±5	μA	Vss ≤ V _{PIN} ≤ V _{DD}
		OSC1	—	—	±5	μA	Vss ≤ V _{PIN} ≤ V _{DD} , pro konfiguraci oscilátoru XT, HS a LP

- ♦ Údaje ve sloupci "Typický" jsou pro napájecí napětí 5 V při teplotě 25 °C pokud není uvedeno jinak.
Údaje jsou pouze pro potřeby návrhu, nejsou testovány.

- * Údaje jsou pouze pro potřeby návrhu, nejsou testovány

Poznámka 1: Při konfiguraci oscilátoru RC je na vstupu OSC1 Schmittův klopný obvod.
Není doporučeno, aby byl v tomto režimu použit externí taktovací signál.

- 2:** Svodový proud vývodu MCLR silně závisí na přiloženém napětí.
Uvedený údaj platí pro normální pracovní podmínky.
Při jiných vstupních napětích může být naměřena vyšší hodnota svodového proudu.

- 3:** Záporný proud je definován jako proud vytékající z vývodu.

18.4 DC charakteristiky: PIC16F87/88 (provedení "Industrial" a "Extended") PIC16LF87/88 (provedení "Industrial") – pokračování

DC parametry			Běžné pracovní podmínky (pokud není uvedeno jinak)				
			pracovní rozsah -40 až +85 °C pro provedení "Industrial"				
			pracovní rozsah -40 až +125 °C pro provedení "Extended"				
			Rozsah napájecího napětí podle DC charakteristik				
Parametr	Značení	Charakteristika	Min.	Typ. ♦	Max.	Jedn.	Podmínky
D080	V _{OL}	Výstupní napětí v úrovni log.0					
		Vstupně výstupní vývody	—	—	0,6	V	I _{OL} = 8,5 mA, V _{DD} = 4,5 V, -40 °C až +125 °C
D083		OSC2/CLKO v konfiguraci RC	—	—	0,6	V	I _{OL} = 1,6 mA, V _{DD} = 4,5 V, -40 °C až +125 °C
D090	V _{OH}	Výstupní napětí v úrovni log.1					
		Vstupně výstupní vývody (3)	V _{DD} – 0,7	—	—	V	I _{OL} = -3,0 mA, V _{DD} = 4,5 V, -40 °C až +125 °C
D092		OSC2/CLKO v konfiguraci RC	V _{DD} – 0,7	—	—	V	I _{OL} = -1,3 mA, V _{DD} = 4,5 V, -40 °C až +125 °C
D100	C _{osc2}	Kapacitní zátěž na výstupních vývodech					
		vývod OSC2	—	—	15	pF	V konfiguraci XT, HS a LP pokud je OSC1 vstup vnějšího taktovacího signálu
		Všechny vst/výst. vývody a OSC2 v konfiguraci RC	—	—	50	pF	
D102	C _B	SCL, SDA (v režimu I ² C™)	—	—	400	pF	
D120	E _D	Paměť dat typu EEPROM					
		počet zápisů	100 K	1M	—	E/W	-40 °C až +85 °C
		10 K	100 K	—	E/W	+85 °C až +125 °C	
D121	V _{DRW}	V _{DD} pro čtení/zápis	V _{MIN}	—	5,5	V	Při použití registru EECON pro zápis a čtení, V _{MIN} = minimální pracovní napětí
D122	T _{DEW}	Cyklus mazání/zápis	—	4	8	ms	
D130	E _P	Paměť programu typu Flash					
		počet zápisů	10 K	100 K	—	E/W	-40 °C až +85 °C
		1 K	10 K	—	E/W	+85 °C až +125 °C	
D131	V _{PR}	V _{DD} pro čtení	V _{MIN}	—	5,5	V	Při použití registru EECON pro zápis a čtení, V _{MIN} = minimální pracovní napětí
V _{DD} pro čtení/zápis		V _{MIN}	—	5,5	V		
D132A							
D133	T _{PE}	Doba mazání	—	4	4	ms	
D134	T _{PW}	Doba zápisu	—	4	4	ms	

- ♦ Údaje ve sloupci "Typický" jsou pro napájecí napětí 5 V při teplotě 25 °C pokud není uvedeno jinak.
Údaje jsou pouze pro potřeby návrhu, nejsou testovány.
- * Údaje jsou pouze pro potřeby návrhu, nejsou testovány

- Poznámka 1:** Při konfiguraci oscilátoru RC je na vstupu OSC1 Schmittův klopný obvod.
Není doporučeno, aby byl v tomto režimu použit externí taktovací signál.
- 2:** Svodový proud vývodu $\overline{\text{MCLR}}$ silně závisí na přiloženém napětí.
Uvedený údaj platí pro normální pracovní podmínky.
Při jiných vstupních napětích může být naměřena vyšší hodnota svodového proudu.
- 3:** Záporný proud je definován jako proud vytékající z vývodu.

PIC16F87/88

Tabulka 18-1: Vlastnosti komparátoru

Pracovní podmínky: $3,0\text{ V} < V_{DD} < 5,5\text{ V}$, $-40\text{ °C} < T_A < +85\text{ °C}$ (pokud není uvedeno jinak)							
Parametr	Značení	Charakteristika	Min.	Typ.	Max.	Jedn.	Komentář
D300	V _{IOFF}	Vstupní napěťová nesymetrie	—	±5,0	±10	mV	
D301	V _{ICM}	Vstupní napětí *	0	—	$V_{DD} - 1,5$	V	
D302	CMRR	Poměr potlačení soufázového napětí*	55	—	—	dB	
300 300A	T _{RESP}	Doba přeběhu (t) [*]	—	150	400 600	ns ns	PIC16F87/88 PIC16LF87/88
301	T _{mc2ov}	Doba platnosti výstupu po změně módu *	—	—	10	μs	

* tyto parametry jsou typické, nejsou však testovány

Poznámka 1: Doba přeběhu je měřena při změně napětí na vstupu komparátoru z hodnoty V_{ss} na V_{DD} za současného připojení druhého vstupu komparátoru na napětí $U = (V_{DD} - 1,5)/2$.

Tabulka 18-2: Vlastnosti napěťové reference

Pracovní podmínky: $3,0\text{ V} < V_{DD} < 5,5\text{ V}$, $-40\text{ °C} < T_A < +85\text{ °C}$ (pokud není uvedeno jinak)							
Parametr	Značení	Charakteristika	Min.	Typ.	Max.	Jedn.	Komentář
D310	V _{RES}	Rozlišení	$V_{DD}/24$	—	$V_{DD}/32$	LSb	
D311	V _{RAA}	Absolutní přesnost	—	—	1/2 1/2	LSb LSb	nižší rozsah (CVRR = 1) vyšší rozsah (CVRR = 0)
D312	V _{RUR}	Hodnota rezistoru *	—	2 k	—	Ω	
310	T _{SET}	Doba ustálení (t) [*]	—	—	10	μs	

* tyto parametry jsou typické, nejsou však testovány

Poznámka 1: Doba ustálení je měřena při V_{rr} = 1 a při změně bitů VR3, VR2, VR1 a VR0 ze stavu "0000" do stavu "1111".

18.5 Symbolika časových parametrů

Symboły časových parametrů jsou vytvořeny jedním z následujících způsobů:

- | | | |
|-------------|-----------|------------------------------------|
| 1. TppS2ppS | 3. Tcc:st | (platí pouze pro I ² C) |
| 2. TppS | 4. Ts | (platí pouze pro I ² C) |

T			
F	Kmitočet	T	Čas

Symboły značené malým písmem a jejich význam:

pp			
cc	CCP1	osc	OSC1
ck	CLKO	rd	$\overline{RD}$
cs	CS	rw	$\overline{RD}$ nebo $\overline{WR}$
di	SDI	sc	SCK
do	SDO	ss	$\overline{SS}$
dt	vstup dat	t0	T0CKI
io	vstupní/výstupní brána	t1	T1CKI
mc	$\overline{MCLR}$	wr	$\overline{WR}$

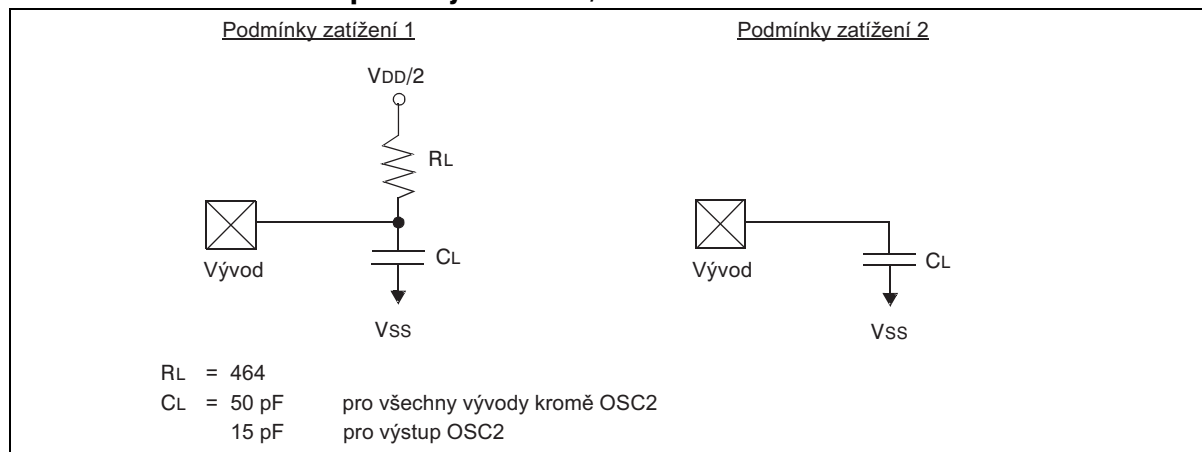
Symboły značené velkými písmeny a jejich význam:

S			
F	Pokles (sestupná hrana)	P	Perioda
H	Vysoká (úroveň)	R	Nárůst (vzestupná hrana)
I	Zakázaný (nebo vysoká impedance)	V	Platný
L	Nízká (úroveň)	Z	Z Vysoká impedance
platí jen pro I²C			
AA	přístup na výstup	High	úroveň H
BUF	sběrnice je volná	Low	úroveň L

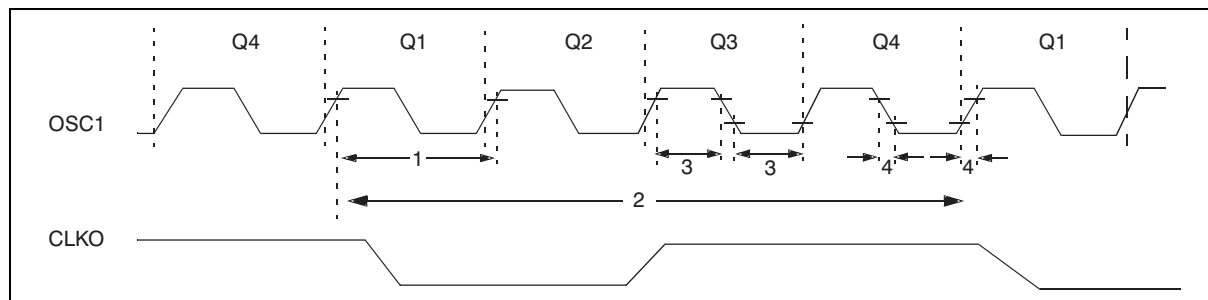
Tcc:st (platí pouze pro I²C):

CC			
HD	Podržení	SU	Nastavení
ST			
DAT	podržení vstupních dat	ST0	Podmínka STOP
STA	podmínka START		

Obrázek 18-3: Zatěžovací podmínky PIC 16F87/88



Obrázek 18-4: Externí taktování



Tabulka 18-3: Požadavky na externí taktování

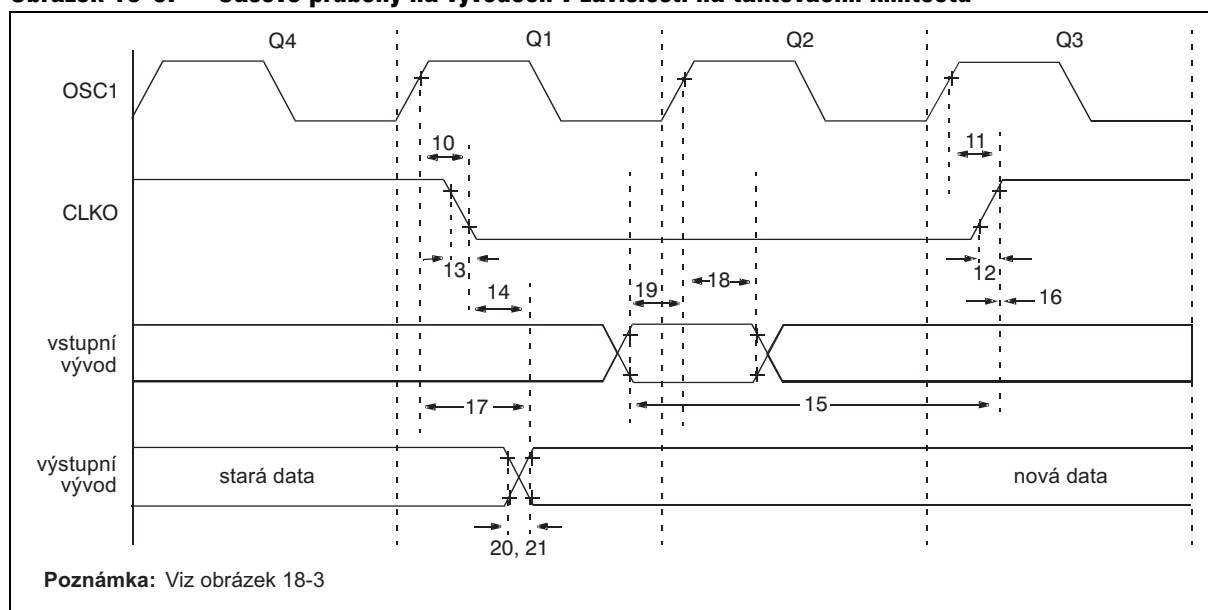
Parametr	Značení	Charakteristika	Min.	Typ. ♦	Max.	Jedn.	Komentář
	Fosc	vnější frekvence na CLKI (Poznámka 1)	DC	—	1	MHz	XT a RC oscilátor
			DC	—	20	MHz	HS oscilátor
			DC	—	32	kHz	LP oscilátor
		frekvence oscilátoru (Poznámka 1)	DC	—	4	MHz	RC oscilátor
			0,1	—	4	MHz	XT oscilátor
1	Tosc	vnější perioda na CLKI (Poznámka 1)	4	—	20	MHz	HS oscilátor
			5	—	200	kHz	LP oscilátor
			1000	—	—	ns	XT a RC oscilátor
		perioda oscilátoru (Poznámka 1)	50	—	—	ns	HS oscilátor
			5	—	—	ms	LP oscilátor
2	Tcy	doba instrukčního cyklu (Poznámka 1)	200	Tcy	DC	ns	Tcy = 4/Fosc
3	TosL, TosH	doba trvání vysoké $\square$ a nízké úrovně na OSC1	500	—	—	ns	XT oscilátor
			2,5	—	—	ms	LP oscilátor
			15	—	—	ns	HS oscilátor
4	TosR, TosF	doba trvání náběžné nebo sestupné hrany na OSC1	—	—	25	ns	XT oscilátor
			—	—	50	ns	LP oscilátor
			—	—	15	ns	HS oscilátor

- ♦ Údaje ve sloupci "Typický" jsou pro napájecí napětí 5 V při teplotě 25 °C pokud není uvedeno jinak.
Údaje jsou pouze pro potřeby návrhu, nejsou testovány.

Poznámka 1: Doba instrukčního cyklu odpovídá čtyřnásobku doby trvání jednoho strojového taktu. Všechny uváděné hodnoty se vztahují ke konkrétnímu typu oscilátoru při standardních podmínkách během výkonu programu. Překročení uvedených hodnot může vést k nestabilitě oscilátoru. Všechny mikropočítače jsou testovány v rozsahu minimálních hodnot s externím taktováním na OSC1. Při použití externího taktování platí hodnoty DC.

PIC16F87/88

Obrázek 18-5: Časové průběhy na vývodech v závislosti na taktovacím kmitočtu



Tabulka 18-4: Požadavky na práci s vstupně výstupními vývody

Parametr	Značení	Charakteristika	Min.	Typ. ♦	Max.	Jedn.	Podmínky
10*	TosH2ckL	od OSC1 ↑ do CLKOUT ↓	—	75	200	ns	(poznámka 1)
11*	TosH2ckH	od OSC1 ↑ do CLKOUT ↓	—	75	200	ns	(poznámka 1)
12*	TckR	doba náběhu CLK0	—	35	100	ns	(poznámka 1)
13*	TckF	doba poklesu CLK0	—	35	100	ns	(poznámka 1)
14*	TckL2ioV	doba od poklesu CLK0 ↓ do platnosti dat	—	—	0,5 Tcy + 20	ns	(poznámka 1)
15*	TioV2ckH	platnost dat před CLKOUT ↑	Tosc + 200	—	—	ns	(poznámka 1)
16*	TckH2ioL	doba podržení dat po CLKOUT ↑	0	—	—	ns	(poznámka 1)
17*	TosH2ioV	OSC1 ↑ doba od Q1 do platnosti dat na výstupu	—	100	255	ns	
18*	TosH2ioL	OSC1 ↑ doba od Q2 do platnosti dat na vstupu	PIC16F87/88	100	—	ns	
			PIC16LF87/88	200	—	ns	
19*	TioV2osH	předstih dat na vstupu před OSC1 ↑	0	—	—	ns	
20*	TioR	doba náběhu na výstupu	PIC16F87/88	—	10	40	ns
			PIC16LF87/88	—	—	145	ns
21*	TioF	doba poklesu napětí na výstupu	PIC16F87/88	—	10	40	ns
			PIC16LF87/88	—	—	145	ns
22♦♦*	TINP	doba trvání úrovně H nebo L na vstupu INT	Tcy	—	—	ns	
23♦♦*	TRBP	doba trvání změny pro přerušení změnou na bráně PortA	Tcy	—	—	ns	

* tyto parametry jsou typické, nejsou však testovány

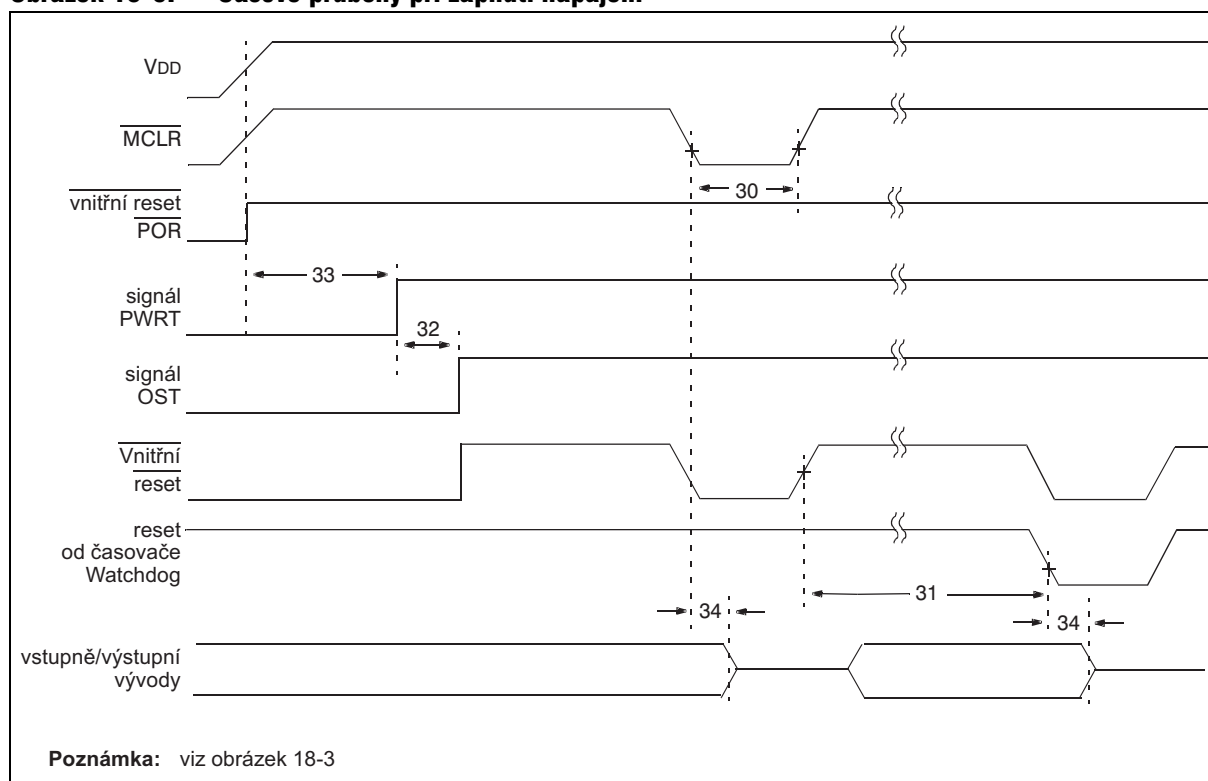
♦ Údaje ve sloupci "Typický" jsou pro napájecí napětí 5 V při teplotě 25 °C pokud není uvedeno jinak. Údaje jsou pouze pro potřeby návrhu, nejsou testovány.

♦♦ jedná se o asynchronní události nezávislé na taktu mikropočítače

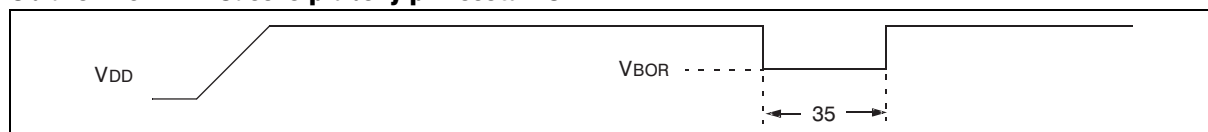
Poznámka 1: Měření je prováděno v konfiguraci RC když na výstupu CLK0 je 4*Fosc

PIC16F87/88

Obrázek 18-6: Časové průběhy při zapnutí napájení



Obrázek 18-7: Časové průběhy při resetu BOR



Tabulka 18-5: Parametry signálů při zapnutí napájení

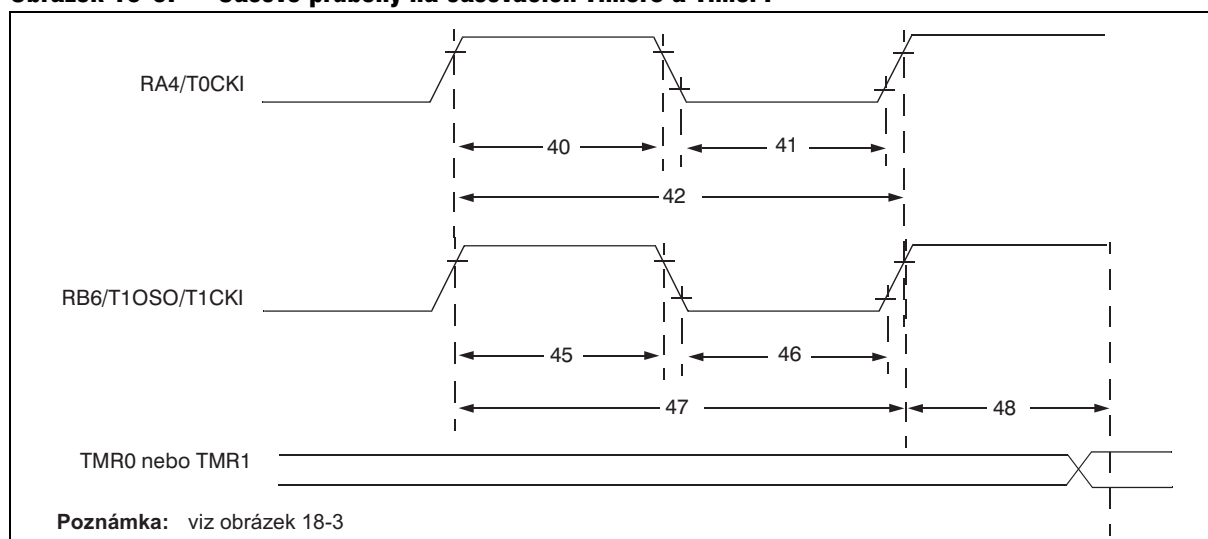
Parametr	Značení	Charakteristika	Min.	Typ. ♦	Max.	Jedn.	Podmínky
30	T _{mcl}	doba trvání MCLR v log.0	2	—	—	μs	V _{DD} = 5V, -40 °C až +85 °C
31*	T _{WDT}	perioda časovače Watchdog (předdělička 16-bitů = 0100 a bez předděličky)	13,6	16	18,4	ms	V _{DD} = 5V, -40 °C až +85 °C
32	T _{OST}	doba zdržení resetu od časovače OST	—	1024 T _{osc}	—	—	T _{osc} = perioda signálu na OSC1
33*	T _{PWRT}	doba zdržení resetu od časovače PWRT	61,2	72	82,8	ms	V _{DD} = 5V, -40 °C až +85 °C
34	T _{IOZ}	doba zpoždění stavu vysoké impedance na vývodu způsobená signálem reset	—	—	2,1	μs	
35	T _{BOR}	doba trvání výpadku napětí detekovaného BOR	100	—	—	μs	V _{DD} ≤ V _{BOR} (D005)

* tyto parametry jsou typické, nejsou však testovány

♦ Údaje ve sloupci "Typický" jsou pro napájecí napětí 5 V při teplotě 25 °C pokud není uvedeno jinak. Údaje jsou pouze pro potřeby návrhu, nejsou testovány.

PIC16F87/88

Obrazek 18-8: Časové průběhy na časovačích Timer0 a Timer1



Tabulka 18-6: Parametry hodinového signálu časovačů Timer0 a Timer1

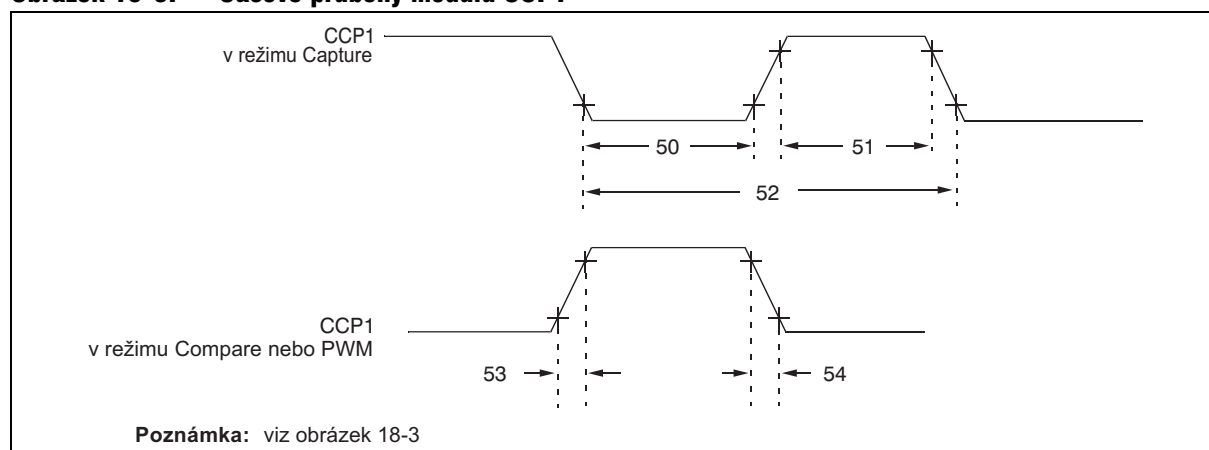
Param.	Značení	Charakteristika			Min.	Typ. ♦	Max.	Jedn.	Podmínky
40*	Tt0H	šířka impulsu T0CKI v úrovni H		bez předděličky	0,5 Tcy + 20	—	—	ns	současně musí být splněn parametr 42
	s předděličkou			10	—	—	ns		
41*	Tt0L	šířka impulsu T0CKI v úrovni L		bez předděličky	0,5 Tcy + 20	—	—	ns	současně musí být splněn parametr 42
	s předděličkou			10	—	—	ns		
42*	Tt0P	perioda T0CKI		bez předděličky	Tcy + 40	—	—	ns	N = nastavení předděličky (2, 4, ..., 256)
	s předděličkou			větší z parametrů 20 nebo $\frac{Tcy+40}{N}$	—	—	ns		
45*	Tt1H	doba setrvání T1CKI ve stavu H	synchronně, předdělička = 1		0,5 Tcy + 20	—	—	ns	současně musí být splněn parametr 47
	synchronně, předdělička = 2,4,8		PIC16F87/88	15	—	—	ns		
			PIC16LF87/88	25	—	—	ns		
	asynchronně		PIC16F87/88	30	—	—	ns		
			PIC16LF87/88	50	—	—	ns		
46*	Tt1L	doba setrvání T1CKI ve stavu L	synchronně, předdělička = 1		0,5 Tcy + 20	—	—	ns	současně musí být splněn parametr 47
	synchronně, předdělička = 2,4,8		PIC16F87/88	15	—	—	ns		
			PIC16LF87/88	25	—	—	ns		
	asynchronně		PIC16F87/88	30	—	—	ns		
			PIC16LF87/88	50	—	—	ns		
47*	Tt1P	perioda T1CKI	synchronně	PIC16F87/88	větší z parametrů 30 nebo $\frac{Tcy+40}{N}$	—	—	ns	N = nastavení předděličky (1, 2, 4, 8)
				PIC16LF87/88	větší z parametrů 50 nebo $\frac{Tcy+40}{N}$				N = nastavení předděličky (1, 2, 4, 8)
		asynchronně	PIC16F87/88	60	—	—	ns		
			PIC16LF87/88	100	—	—	ns		
	Ft1	rozsah hodnot na vstupu Timer1 (oscilátor je povolen nastavením bitu T1OSCEN)			DC	—	32,768	kHz	
48	TCKEZtmr1	zpoždění mezi aktivní hranou na vstupu a inkrementací čítače			2 Tosc*	—	7 Tosc*	—	

* tyto parametry jsou typické, nejsou však testovány

♦ Údaje ve sloupci "Typický" jsou pro napájecí napětí 5 V při teplotě 25 °C pokud není uvedeno jinak.
Údaje jsou pouze pro potřeby návrhu, nejsou testovány.

PIC16F87/88

Obrázek 18-9: Časové průběhy modulu CCP1



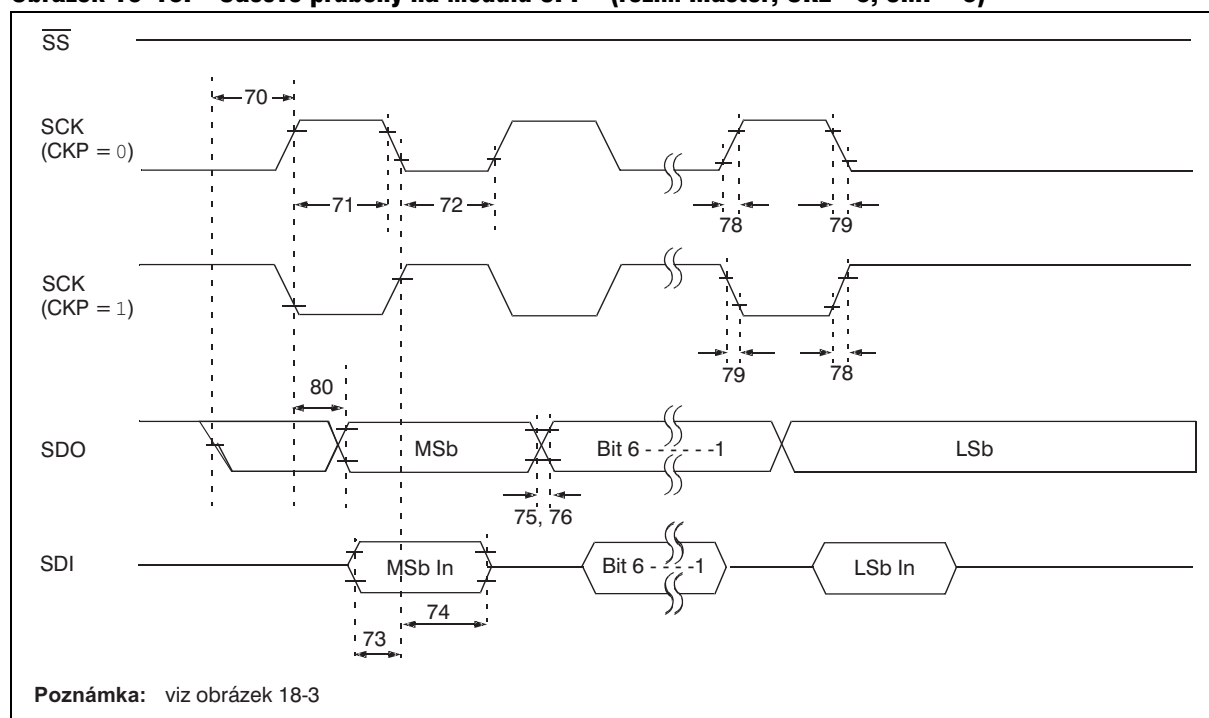
Tabulka 18-7: Parametry modulu CCP1

Param.	Značení	Charakteristika		Min.	Typ. ♦	Max.	Jedn.	Podmínky
50*	TccL	doba trvání impulsu na vstupu CCP1 v log.0	bez předděličky	$0,5 T_{CY} + 20$	—	—	ns	
			s předděličkou PIC16F87/88	10	—	—	ns	
			PIC16LF87/88	20	—	—	ns	
51*	TccH	doba trvání impulsu na vstupu CCP1 v log.1	bez předděličky	$0,5 T_{CY} + 20$	—	—	ns	
			s předděličkou PIC16F87/88	10	—	—	ns	
			PIC16LF87/88	20	—	—	ns	
52*	TccP	perioda vstupního signálu CCP1		$\frac{3 T_{CY} + 40}{N}$	—	—	ns	N = poměr předděličky (1,4 nebo 16)
53*	TccR	doba náběžné hrany signálu na výstupu CCP1	PIC16F87/88	—	10	25	ns	
			PIC16LF87/88	—	25	50	ns	
54*	TccF	doba sestupné hrany na výstupu CCP1 a CCP2	PIC16F87/88	—	10	25	ns	
			PIC16LF87/88	—	25	40	ns	

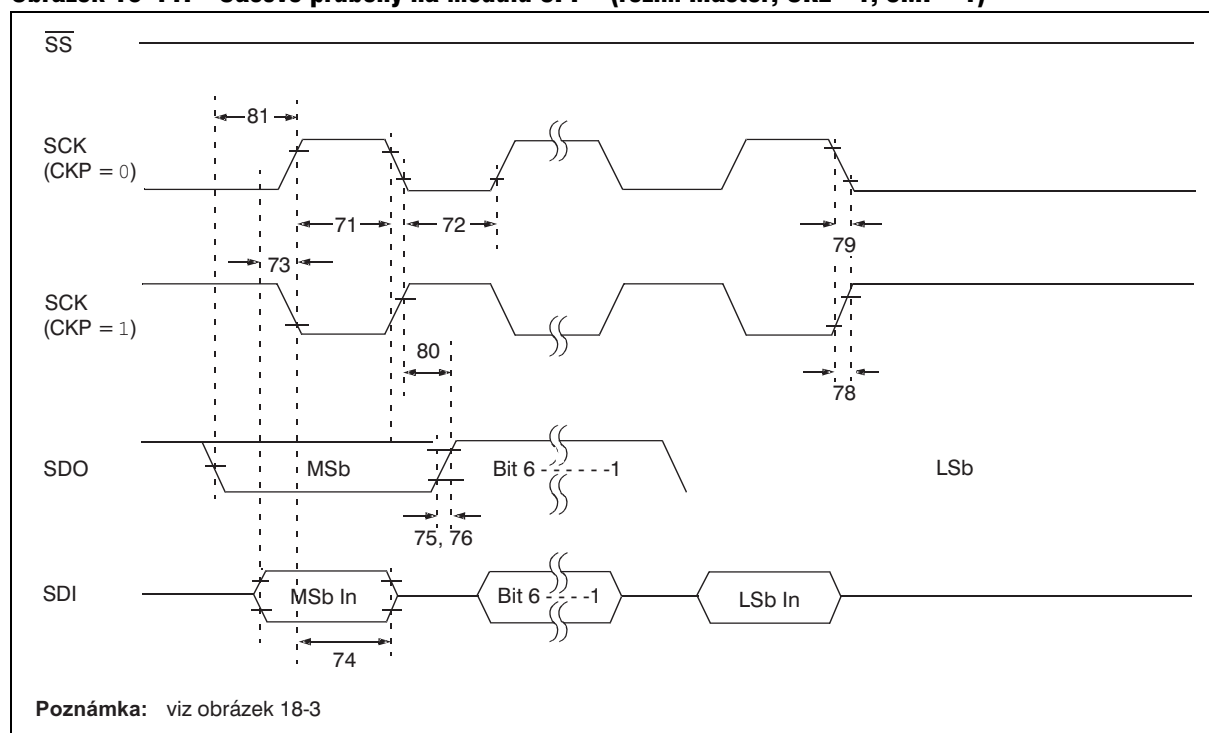
* tyto parametry jsou typické, nejsou však testovány

♦ Údaje ve sloupci "Typický" jsou pro napájecí napětí 5 V při teplotě 25 °C pokud není uvedeno jinak.
Údaje jsou pouze pro potřeby návrhu, nejsou testovány.

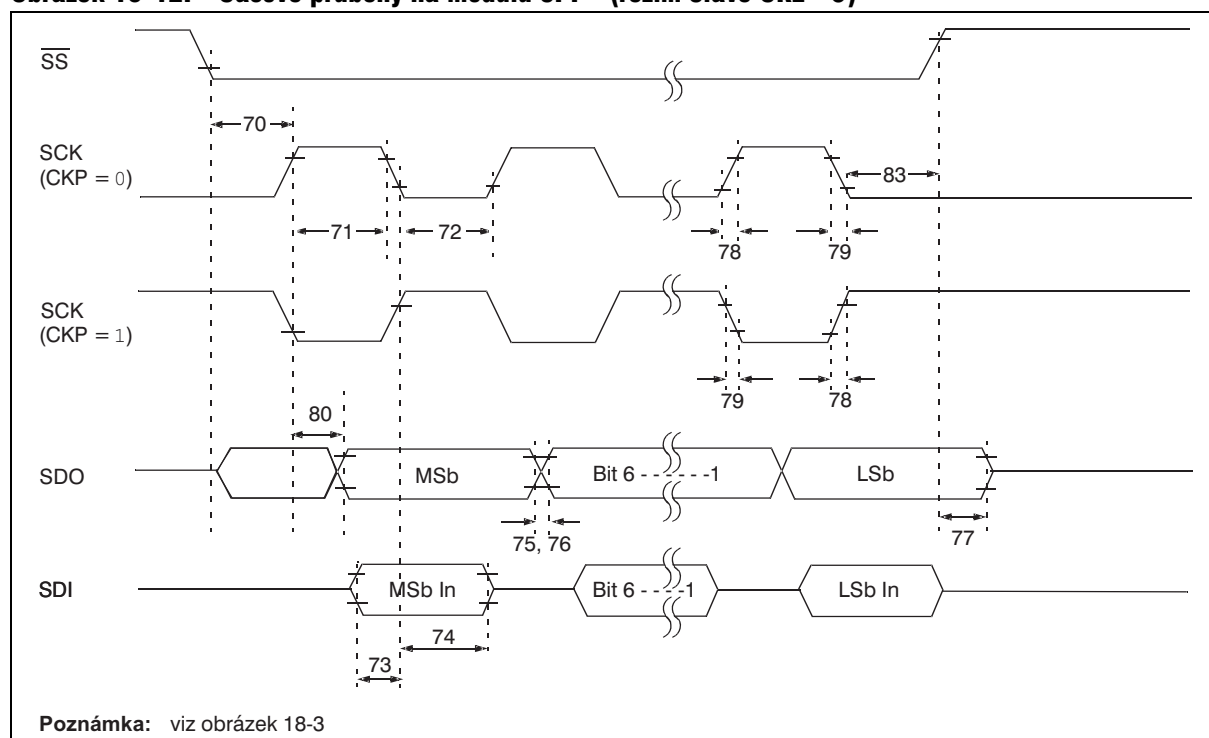
Obrázek 18-10: Časové průběhy na modulu SPI™ (režim Master, CKE=0, SMP=0)



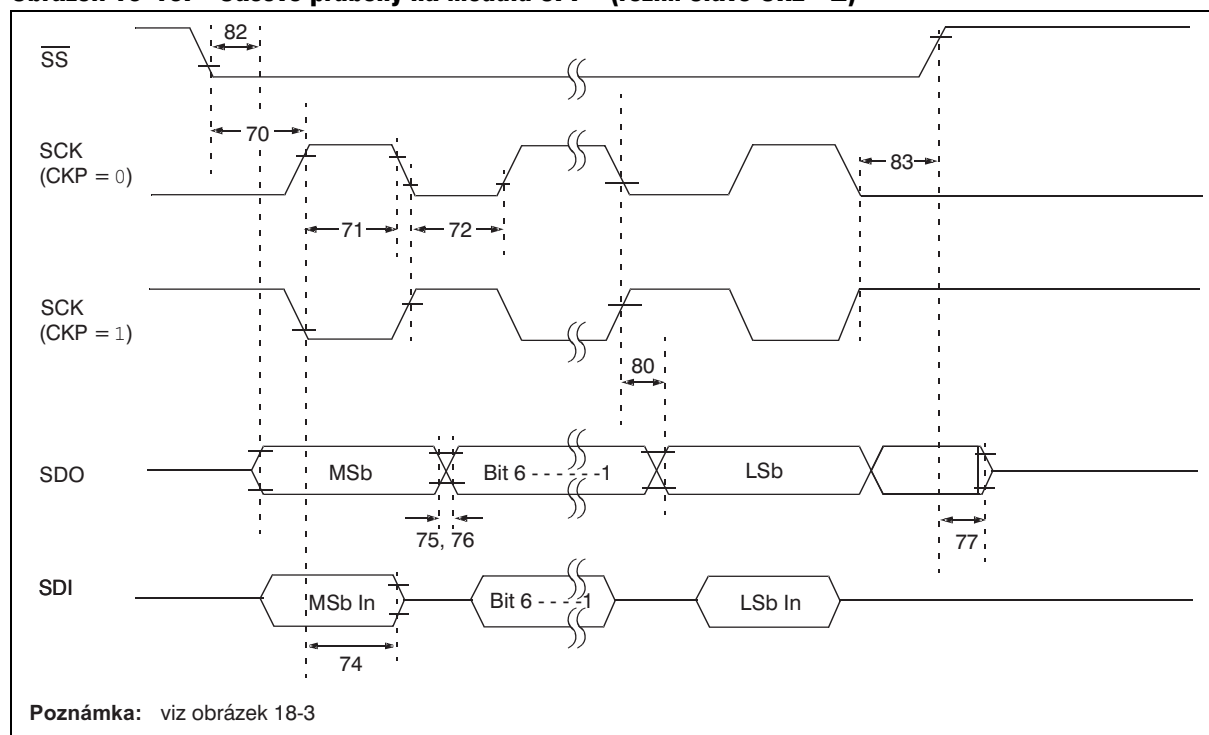
Obrázek 18-11: Časové průběhy na modulu SPI™ (režim Master, CKE=1, SMP=1)



Obrázek 18-12: Časové průběhy na modulu SPI™ (režim Slave CKE=0)



Obrázek 18-13: Časové průběhy na modulu SPI™ (režim Slave CKE=1)

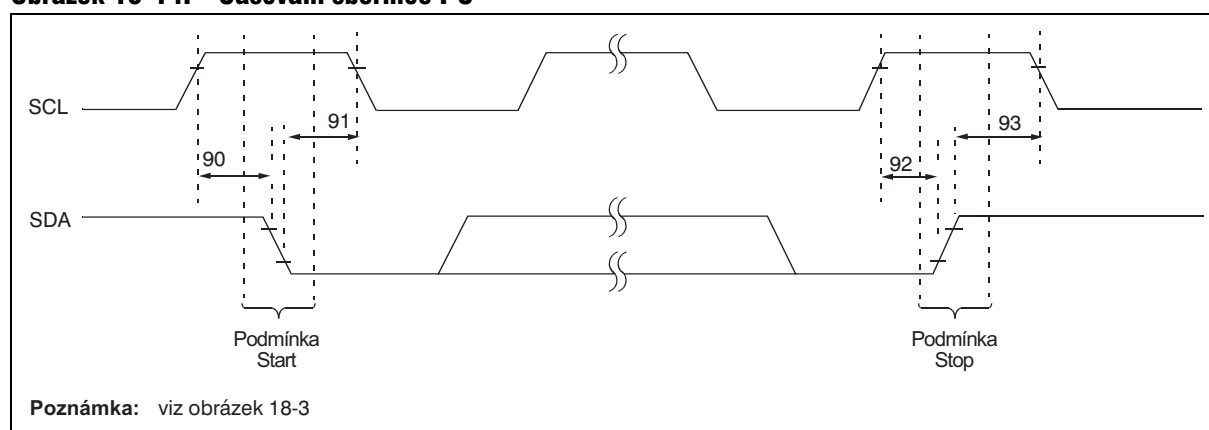


Tabulka 18-8: Parametry modulu SPI™

Parametr	Značení	Charakteristika	Min.	Typ. ♦	Max.	Jedn.	Podmínky
70*	TssL2scH, TssL2scL	doba od $\overline{SS} \downarrow$ do SCK $\downarrow$ nebo SCK $\uparrow$	T_{cy}	—	—	ns	
71*	TscH	doba trvání impulsu na vstupu SCK v log.1 (režim Slave)	$T_{cy} + 20$	—	—	ns	
72*	TscL	doba trvání impulsu na vstupu SCK v log.0 (režim Slave)	$T_{cy} + 20$	—	—	ns	
73*	TdiV2scH, TdiV2scL	doba trvání platných dat na vstupu SDI před SCK	100	—	—	ns	
74*	Tsch2diL, TscL2diL	doba trvání platných dat na vstupu SDI po SCK	100	—	—	ns	
75*	TdoR	doba náběžné hrany signálu na výstupu SDO	PIC16F87/88 —	10	25	ns	
			PIC16LF87/88 —	25	50	ns	
76*	TdoF	doba sestupné hrany signálu na výstupu SDO	—	10	25	ns	
77*	TssH2doZ	doba od $\overline{SS} \uparrow$ do zahájení třetího stavu na výstupu SDO	10	—	50	ns	
78*	TscR	doba náběžné hrany signálu na výstupu SCK (režim Master)	PIC16F87/88 —	10	25	ns	
			PIC16LF87/88 —	25	50	ns	
79*	TscF	doba sestupné hrany signálu na výstupu SCK (režim Master)	—	10	25	ns	
80*	Tsch2doV, TscL2doV	doba trvání platných dat na výstupu SDO po SCK	PIC16F87/88 —	—	50	ns	
			PIC16LF87/88 —	—	145	ns	
81*	TdoV2scH, TdoV2scL	doba trvání platných dat na výstupu SDO do SCK	T_{cy}	—	—	ns	
82*	TssL2doV	doba trvání platných dat na výstupu SDO po hraně $\overline{SS} \downarrow$	—	—	50	ns	
83*	Tsch2ssH, TscL2ssH	doba od náběžné hrany $\overline{SS} \uparrow$ do hrany SCK	$1,5 T_{cy} + 40$	—	—	ns	

- * tyto parametry jsou typické, nejsou však testovány
- ♦ Údaje ve sloupci "Typický" jsou pro napájecí napětí 5 V při teplotě 25 °C pokud není uvedeno jinak. Údaje jsou pouze pro potřeby návrhu, nejsou testovány.

Obrázek 18-14: Časování sběrnice I²C™



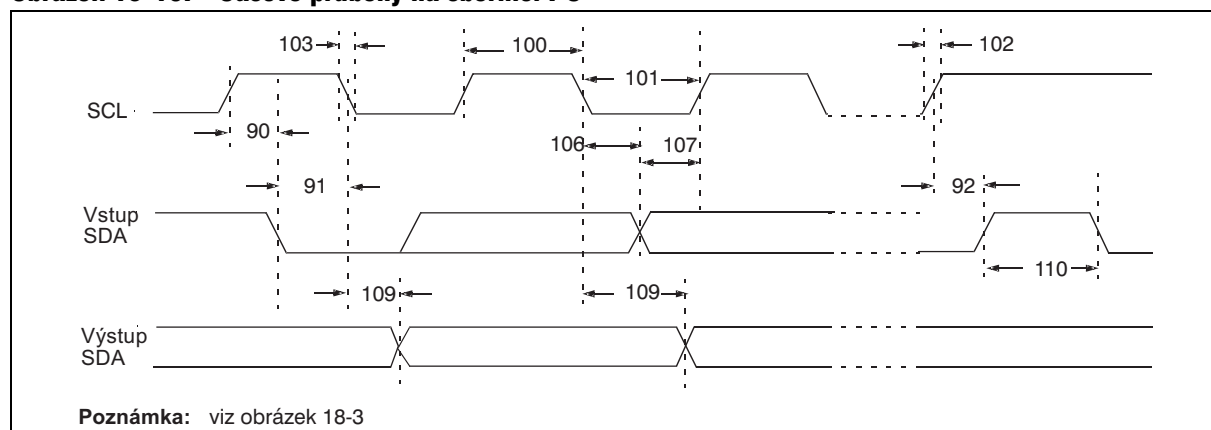
PIC16F87/88

Tabulka 18-9: Parametry signálů Start a Stop na sběrnici I²C™

Parametr	Značení	Charakteristika		Min.	Typ. ♦	Max.	Jedn.	Podmínky
90*	T _{SU:STA}	Podmínka Start	100 kHz režim	4700	—	—	ns	platí pouze při opakované podmínce Start
		doba nastavení dat	400 kHz režim	600	—	—		
91*	T _{HD:STA}	Podmínka Start	100 kHz režim	4000	—	—	ns	po této době je generován první impuls hodinového signálu
		doba podržení dat	400 kHz režim	600	—	—		
92*	T _{SU:STO}	podmínka Stop	100 kHz režim	4700	—	—	ns	
		doba nastavení dat	400 kHz režim	600	—	—		
93*	T _{HD:STO}	podmínka Stop	100 kHz režim	4000	—	—	ns	
		doba podržení dat	400 kHz režim	600	—	—		

* tyto parametry jsou typické, nejsou však testovány

Obrázek 18-15: Časové průběhy na sběrnici I²C™



Tabulka 18-10: Parametry sběrnice I²C™

Parametr	Značení	Charakteristika	Min.	Max.	Jedn.	Podmínky
100*	T _{HIGH}	doba trvání hodinového signálu v log.1	100 kHz režim	4,0	—	μs
			400 kHz režim	0,6	—	μs
			modul SSP	1,5 T _{CY}	—	
101*	T _{LOW}	doba trvání hodinového signálu v log.0	100 kHz režim	4,7	—	μs
			400 kHz režim	1,3	—	μs
			modul SSP	1,5 T _{CY}	—	
102*	T _R	doba náběžné hrany signálu na SDA a SCL	100 kHz režim	—	1000	ns
			400 kHz režim	20 + 0,1 C _B	300	ns hodnota C _B je od 10 do 400 pF
103*	T _F	doba sestupné hrany signálu na SDA a SCL	100 kHz režim	—	300	ns
			400 kHz režim	20 + 0,1 C _B	300	ns hodnota C _B je od 10 do 400 pF
90*	T _{SU:STA}	doba trvání log.1 před podmínkou Start	100 kHz režim	4,7	—	μs
			400 kHz režim	0,6	—	μs
91*	T _{HD:STA}	doba trvání log.1 po podmínce Start	100 kHz režim	4,0	—	μs
			400 kHz režim	0,6	—	μs
106*	T _{HD:DAT}	doba platnosti dat na vstupu	100 kHz režim	0	—	ns
			400 kHz režim	0	0,9	μs
107*	T _{SU:DAT}	doba předstihu dat na vstupu	100 kHz režim	250	—	ns
			400 kHz režim	100	—	ns
92*	T _{SU:STO}	doba trvání log.1 před podmínkou Stop	100 kHz režim	4,7	—	μs
			400 kHz režim	0,6	—	μs
109*	T _{AA}	platnost dat na výstupu	100 kHz režim	—	3500	ns
			400 kHz režim	—	—	ns
110*	T _{BUF}	doba, po kterou je sběrnice volná	100 kHz režim	4,7	—	μs
			400 kHz režim	1,3	—	μs
	C _B	kapacitní zátěž sběrnice	—	400	pF	

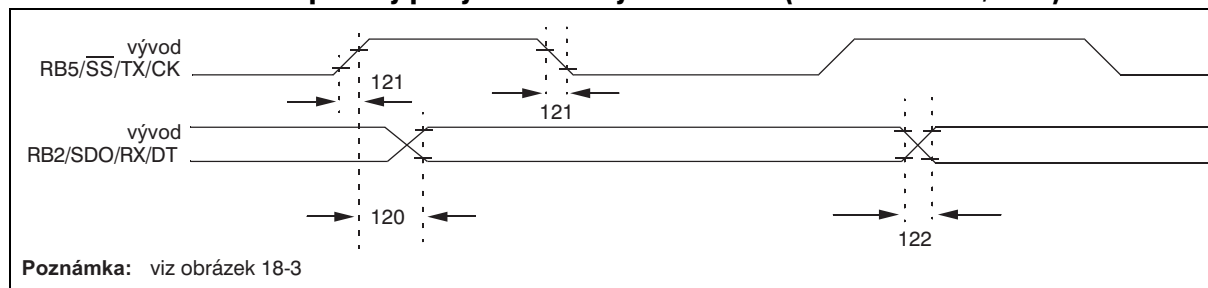
* tyto parametry jsou typické, nejsou však testovány

Poznámka 1: Jako vysílač musí mikropočítač zajistit tyto minimální doby, aby se zabránilo nechtěnému generování podmínky Start nebo Stop.

2: Obvody v režimu 400 kHz je možné použít pro režim 100 kHz. Musí být však splněna podmínka T_{SU:DAT} je menší než 250 ns. To platí automaticky pro případ, že nebude prodloužen signál SCL v log.0.
Pro součet časů T_{RMAX} + T_{SU:DAT} musí platit:
T_{RMAX} + T_{SU:DAT} = 1000 + 250 = 1250 ns.

PIC16F87/88

Obrázek 18-16: Časové průběhy při synchronním vysílání USART (časování master/slave)

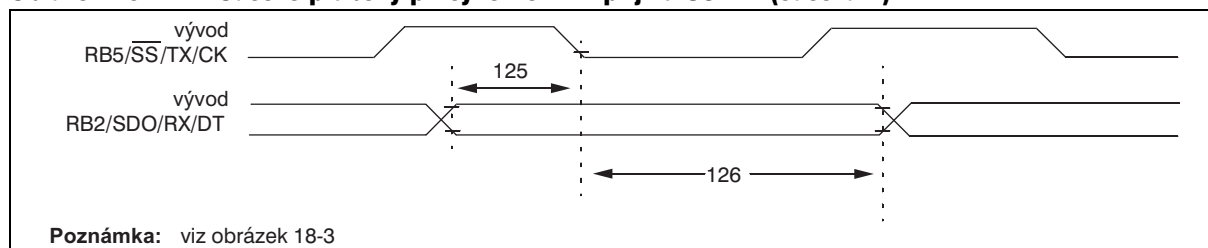


Tabulka 18-11: Parametry synchronního vysílání modulu USART

Parametr	Značení	Charakteristika	Min.	Typ. ♦	Max.	Jedn.	Podmínky
120	TckH2dtV	SYNC XMIT (MASTER & SLAVE) doba trvání hodinového signálu ve stavu H	PIC16F87/88 —	—	80	ns	
121	Tckrf	náběžná a sestupná hrana hodinového signálu	PIC16F87/88 —	—	45	ns	
122	Tdtrf	náběžná a sestupná hrana datového signálu	PIC16F87/88 —	—	45	ns	
			PIC16LF87/88 —	—	50	ns	

+ Údaje ve sloupci "Typický" jsou pro napájecí napětí 5V při teplotě 25 °C pokud není uvedeno jinak. Údaje jsou pouze pro potřeby návrhu, nejsou testovány.

Obrázek 18-17: Časové průběhy při synchronním příjmu USART (časování)



Tabulka 18-12: Parametry synchronního příjmu modulu USART

Parametr	Značení	Charakteristika	Min.	Typ. ♦	Max.	Jedn.	Podmínky
125	TdtV2ckL	SYNC XMIT (MASTER & SLAVE) Předstih dat před sestupnou hranou CK ↓	15	—	—	ns	
126	TckL2dtI	Platnost dat po sestupné hraně CK ↓	15	—	—	ns	

♦ Údaje ve sloupci "Typický" jsou pro napájecí napětí 5 V při teplotě 25 °C pokud není uvedeno jinak. Údaje jsou pouze pro potřeby návrhu, nejsou testovány.

PIC16F87/88

**Tabulka 18-13: Parametry AD převodníku PIC16F87/88 (provedení Industrial a Extended)
PIC16LF87/88 (provedení Industrial)**

Parametr	Znač.	Charakteristika	Min.	Typ. ♦	Max.	Jedn.	Podmínky
A01	NR	Rozlišení	—	—	10-bit	bit	$V_{REF} = V_{DD} = 5,12 \text{ V}$, $V_{SS} \leq V_{AIN} \leq V_{REF}$
A03	EIL	Integrační chyba	—	—	$< \pm 1$	LSb	$V_{REF} = V_{DD} = 5,12 \text{ V}$, $V_{SS} \leq V_{AIN} \leq V_{REF}$
A04	EDL	Diferenciální chyba	—	—	$< \pm 1$	LSb	$V_{REF} = V_{DD} = 5,12 \text{ V}$, $V_{SS} \leq V_{AIN} \leq V_{REF}$
A06	E _{OFF}	Chyba offsetu	—	—	$< \pm 2$	LSb	$V_{REF} = V_{DD} = 5,12 \text{ V}$, $V_{SS} \leq V_{AIN} \leq V_{REF}$
A07	E _{GN}	Chyba zesílení	—	—	$< \pm 1$	LSb	$V_{REF} = V_{DD} = 5,12 \text{ V}$, $V_{SS} \leq V_{AIN} \leq V_{REF}$
A10	—	Monotónnost	—	zaručeno (3)	—	—	$V_{SS} \leq V_{AIN} \leq V_{REF}$
A20	V _{REF}	Referenční napětí (V _{REF+} – V _{REF-})	2,0	—	V _{DD} + 0,3	V	
A21	V _{REF+}	Horní mez referenčního napětí	AV _{DD} – 2,5 V	—	AV _{DD} + 0,3 V	V	
A22	V _{REF-}	Dolní mez referenčního napětí	AV _{SS} – 0,3 V	—	V _{REF+} – 2,0 V	V	
A25	V _{AIN}	Vstupní napětí	V _{SS} – 0,3 V	—	V _{REF} + 0,3 V	V	
A30	Z _{AIN}	Doporučená impedance zdroje vstupního napětí	—	—	2,5	kΩ	(Poznámka 4)
A40	I _{AD}	Proud AD převodníku	PIC16F87/88	—	220	—	Průměrná spotřeba AD převodníku ve stavu zapnuto (Poznámka 1)
			PIC16LF87/88	—	90	—	
A50	I _{REF}	Proud do vstupu referenčního napětí (Poznámka 2)	—	—	5	μA	Vstupní napětí je udržováno na vzorkovacím kondenzátoru, který je tímto napětím nabíjen (viz kapitulu 12.1)
			—	—	150	μA	

* tyto parametry jsou typické, nejsou však testovány

♦ Údaje ve sloupci "Typický" jsou pro napájecí napětí 5 V při teplotě 25 °C pokud není uvedeno jinak.
Údaje jsou pouze pro potřeby návrhu, nejsou testovány.

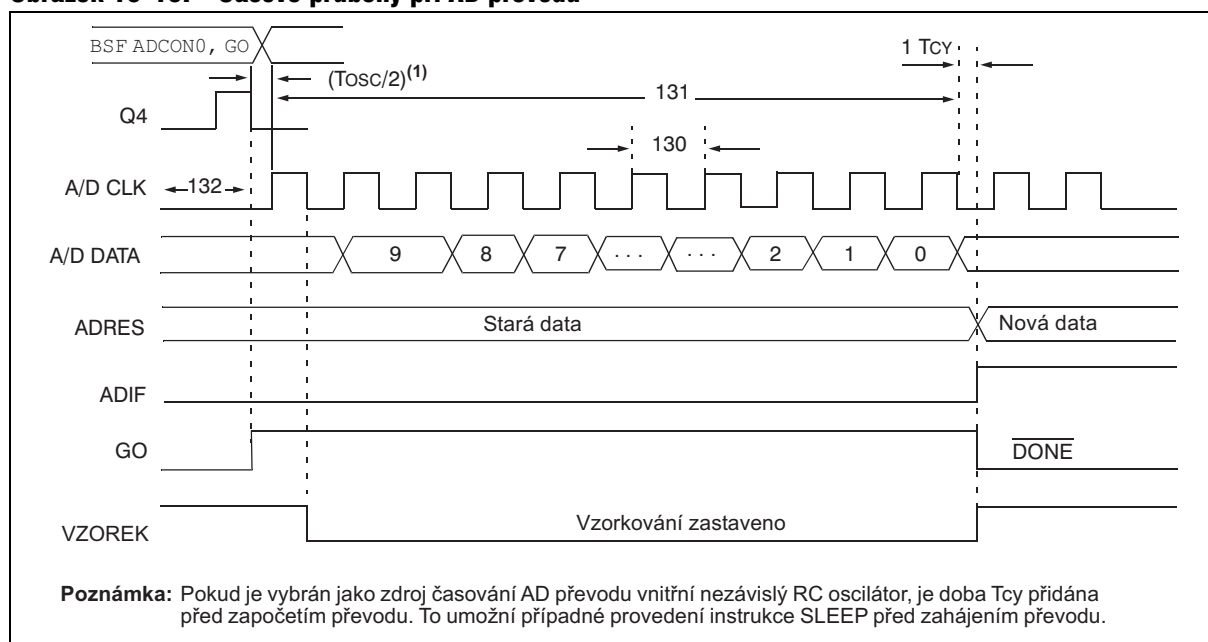
Poznámka 1: Pokud je AD převodník vypnut, nemá žádnou proudovou spotřebu (vyjma svodových proudů)

2: Proud odebíraný z referenčního zdroje napětí teče do vývodu RA3 nebo V_{DD}, podle konfigurace AD převodníku

3: Přesnost AD převodníku nezávisí na velikosti vstupního napětí

4: Nejvyšší povolená impedance zdroje vstupního napětí je 10 kΩ.

Obrázek 18-18: Časové průběhy při AD převodu



Tabulka 18-14: Parametry AD převodu

Parametr	Znač.	Charakteristika		Min.	Typ. ♦	Max.	Jedn.	Podmínky
130	Tad	Doba vzorkování jednoho bitu	PIC16 F 87/88	1,6	—	—	μs	Tosc odvozen od taktu mikropočítače, VREF ≥ 3,0 V
			PIC16 L F87/88	3,0	—	—	μs	Tosc odvozen od taktu mikropočítače, VREF ≥ 2,0 V
			PIC16 F 87/88	2,0	4,0	6,0	μs	Tosc odvozen od vlastního RC oscilátoru
			PIC16 L F87/88	3,0	6,0	9,0	μs	Tosc odvozen od vlastního RC oscilátoru
131	Tcnv	Doba převodu celého výsledku AD převodu (Poznámka 1)			—	12	Tad	
132	Tacq	Doba přítomnosti vstupního napětí	(Pozn. 2) 10☐	40	—	—	μs	Nejmenší doba je doba pro ustálení zesilovače. Lze použít pouze tehdy, pokud na dalším vstupu není vstupní napětí odlišné o více než 1 LSb (20 mV při Udd = 5,12 V)
				—	—	—	μs	
134	Tgo	Doba od Q4 do zahájení AD převodu		—	Tosc/2 §	—	—	Pokud je vybrán jako zdroj časování AD převodu vnitřní nezávislý RC oscilátor, je doba Tcy přidána před započítáním převodu. To umožní případné provedení instrukce SLEEP před zahájením převodu.

* tyto parametry jsou typické, nejsou však testovány

♦ Údaje ve sloupci "Typický" jsou pro napájecí napětí 5V při teplotě 25 °C pokud není uvedeno jinak. Údaje jsou pouze pro potřeby návrhu, nejsou testovány.

§ tato hodnota vyplývá z návrhových pravidel

Poznámka 1: Výsledkový registr ADRES může být čten již následující cyklus Tcy

2: Viz kapitolu 12.1

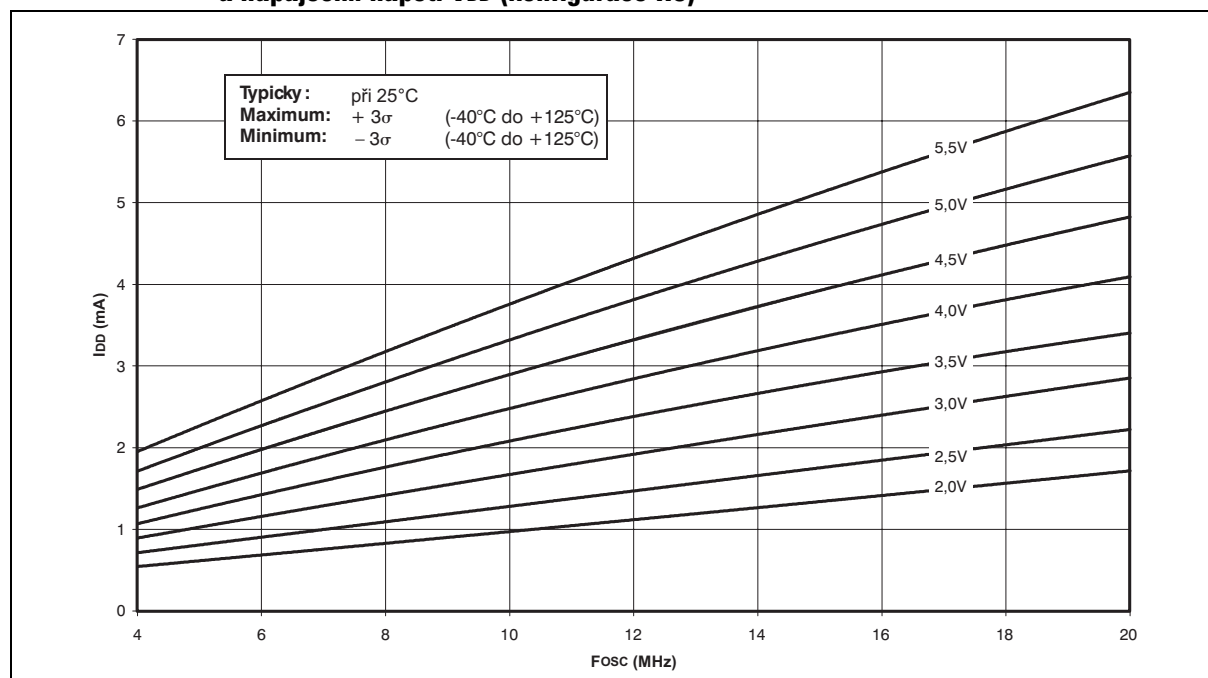
PIC16F87/88

Poznámky:

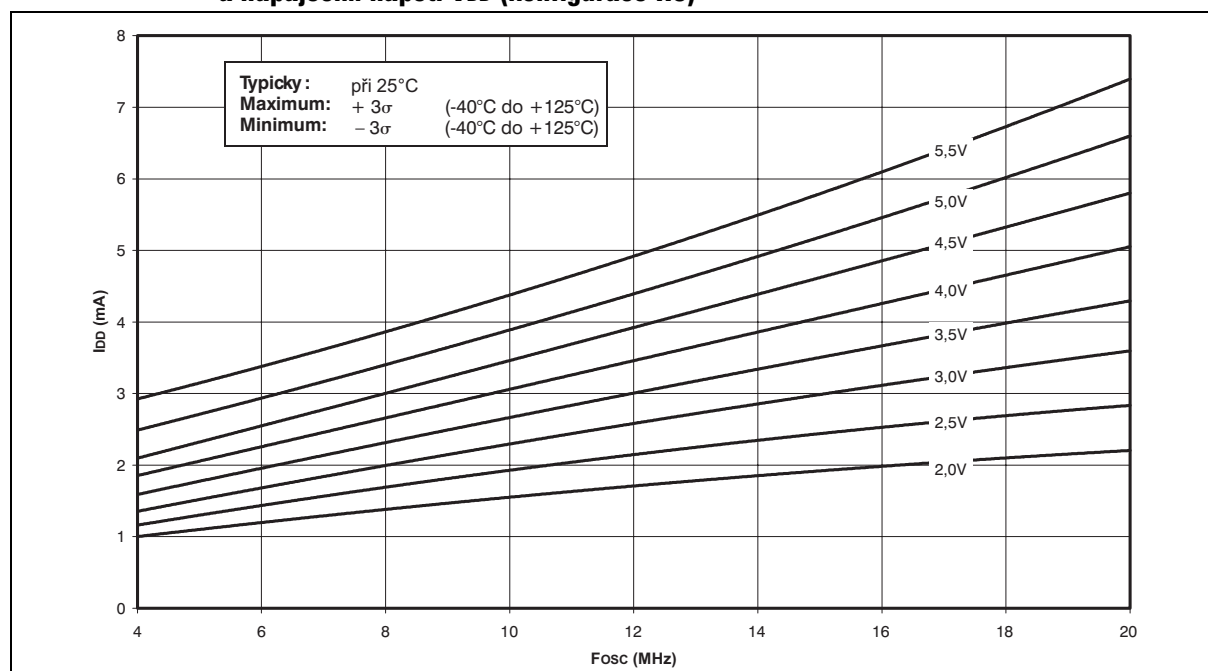
19 GRAFY

Parametry a průběhy uvedené v této kapitole jsou pouze informativní. Údaje byli získány statistickým zpracováním naměřených hodnot. Podrobnosti lze získat na stránkách výrobce www.microchip.com. Typické hodnoty odpovídají teplotě okolí 25 °C.

Obrázek 19-1: Typická závislost hodnoty proudu I_{DD} na taktovací frekvenci a napájecím napětí V_{DD} (konfigurace HS)

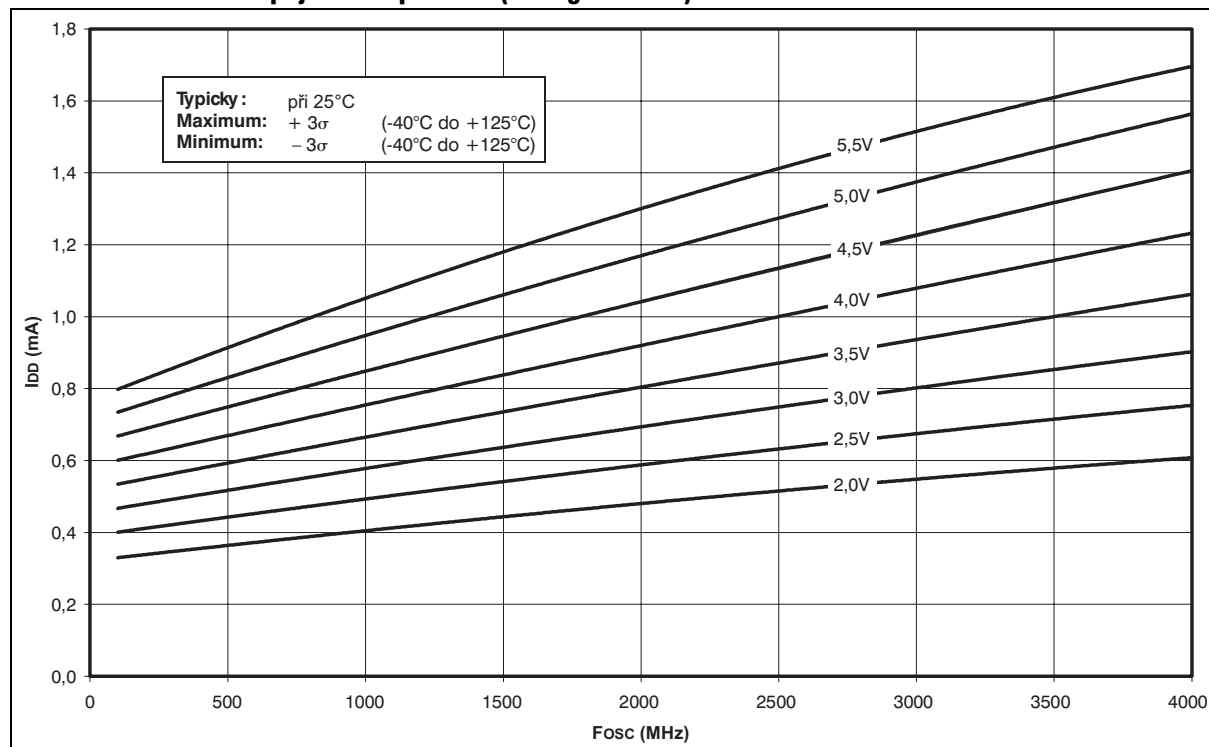


Obrázek 19-2: Závislost maximální hodnoty proudu I_{DD} na taktovací frekvenci a napájecím napětí V_{DD} (konfigurace HS)

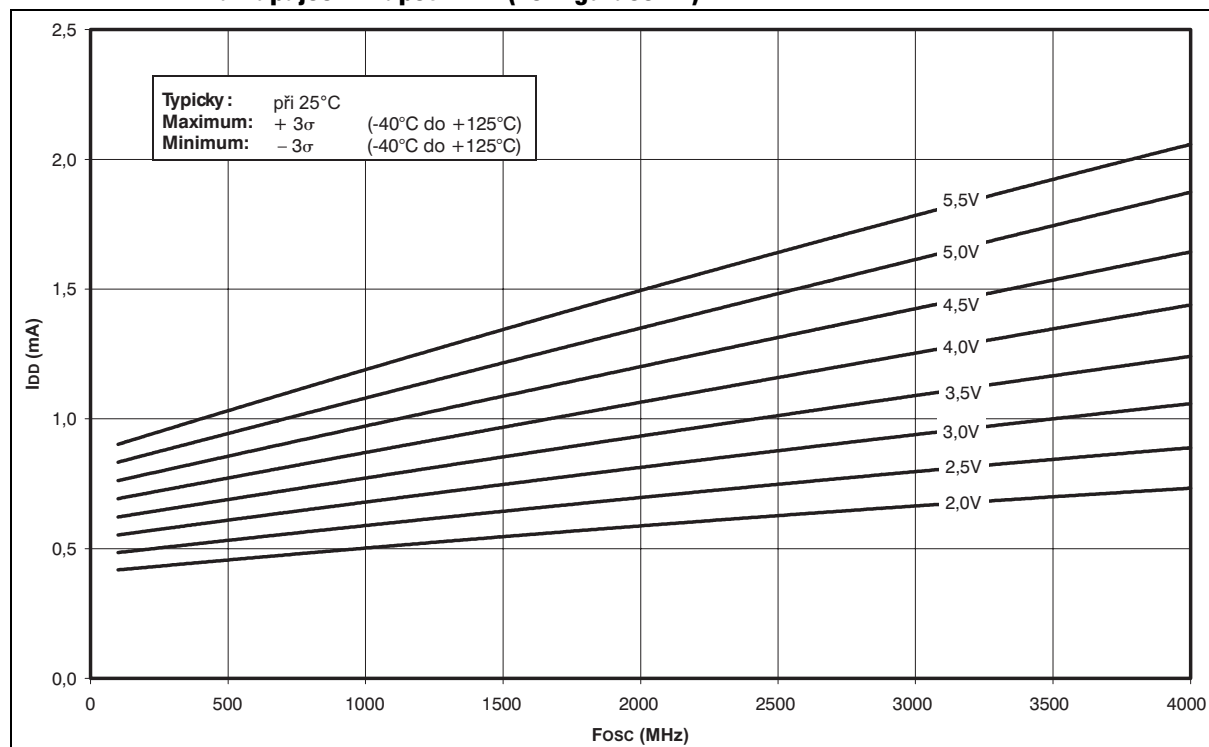


PIC16F87/88

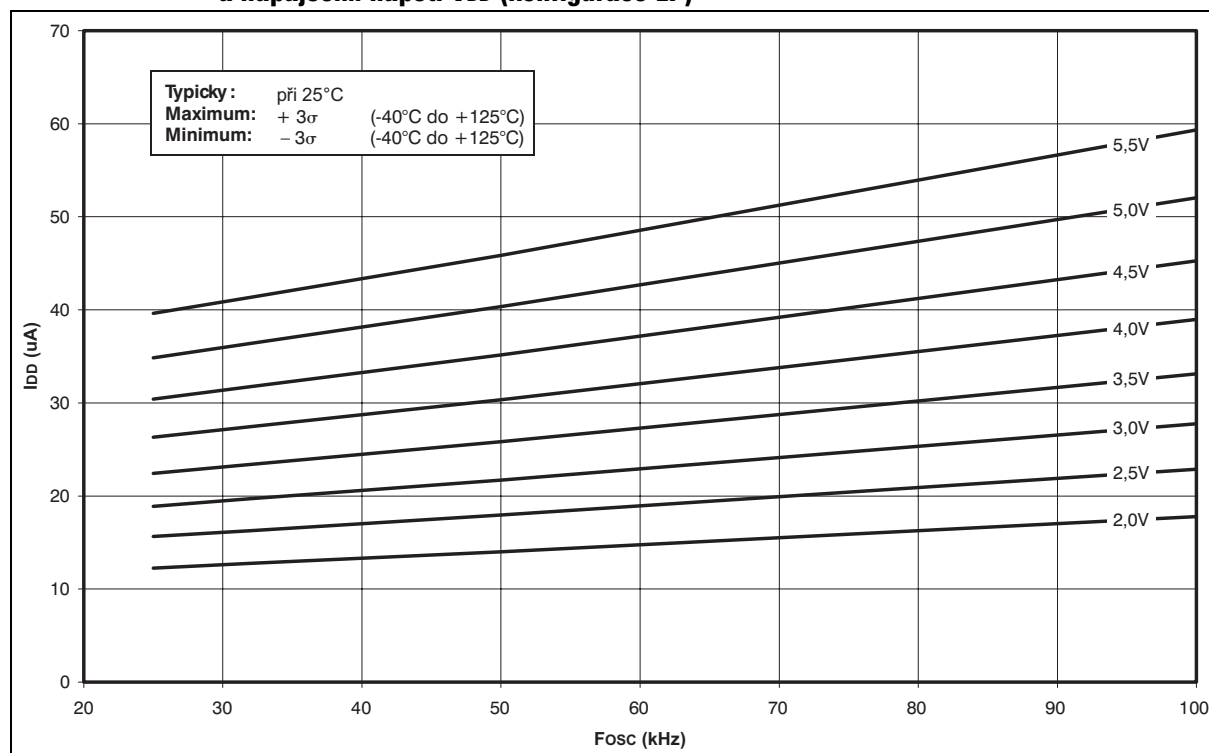
Obrázek 19-3: Typická závislost hodnoty proudu I_{DD} na taktovací frekvenci a napájecím napětí V_{DD} (konfigurace XT)



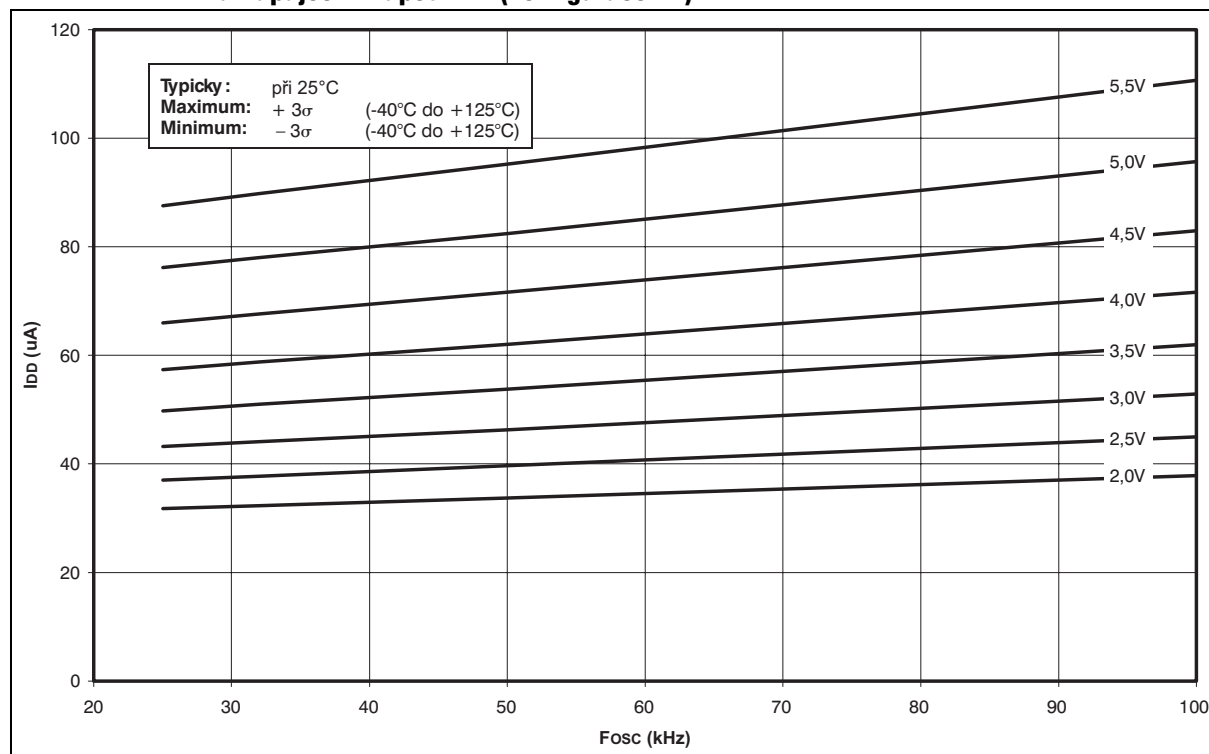
Obrázek 19-4: Závislost maximální hodnoty proudu I_{DD} na taktovací frekvenci a napájecím napětí V_{DD} . (konfigurace XT)



Obrázek 19-5: Typická závislost hodnoty proudu I_{DD} na taktovací frekvenci a napájecím napětí V_{DD} (konfigurace LP)

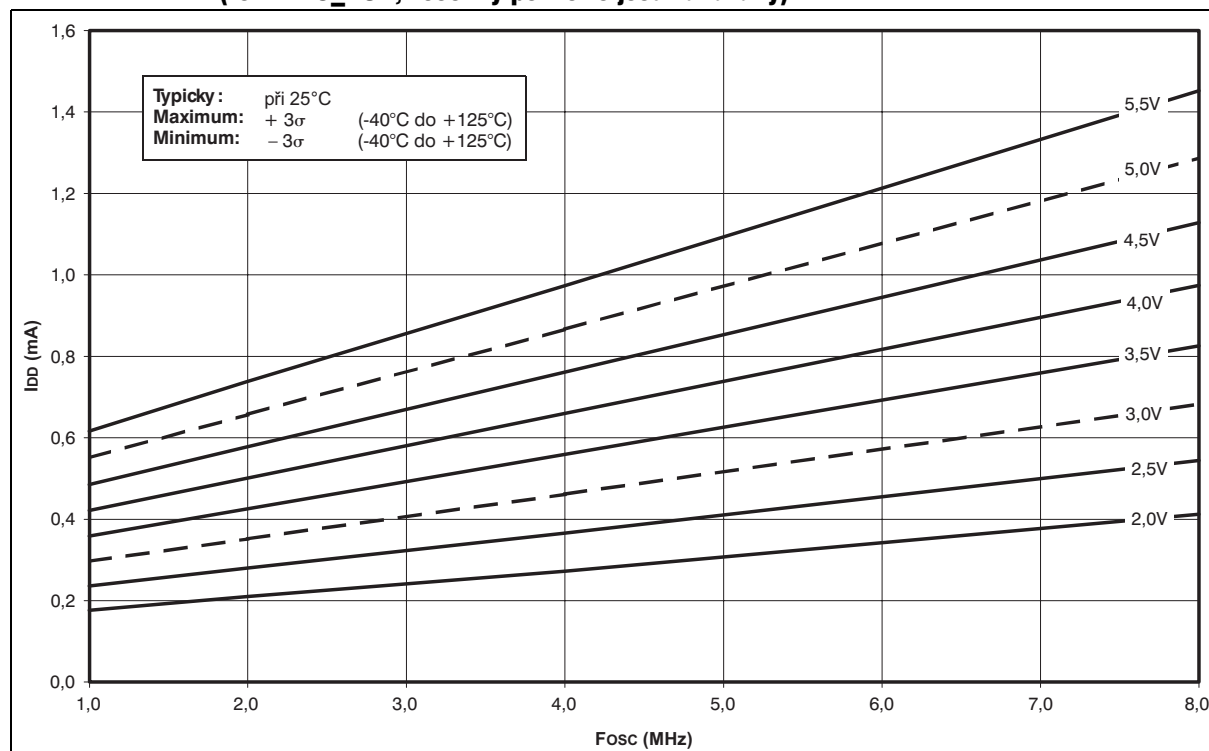


Obrázek 19-6: Závislost maximální hodnoty proudu I_{DD} na taktovací frekvenci a napájecím napětí V_{DD} (konfigurace LP)

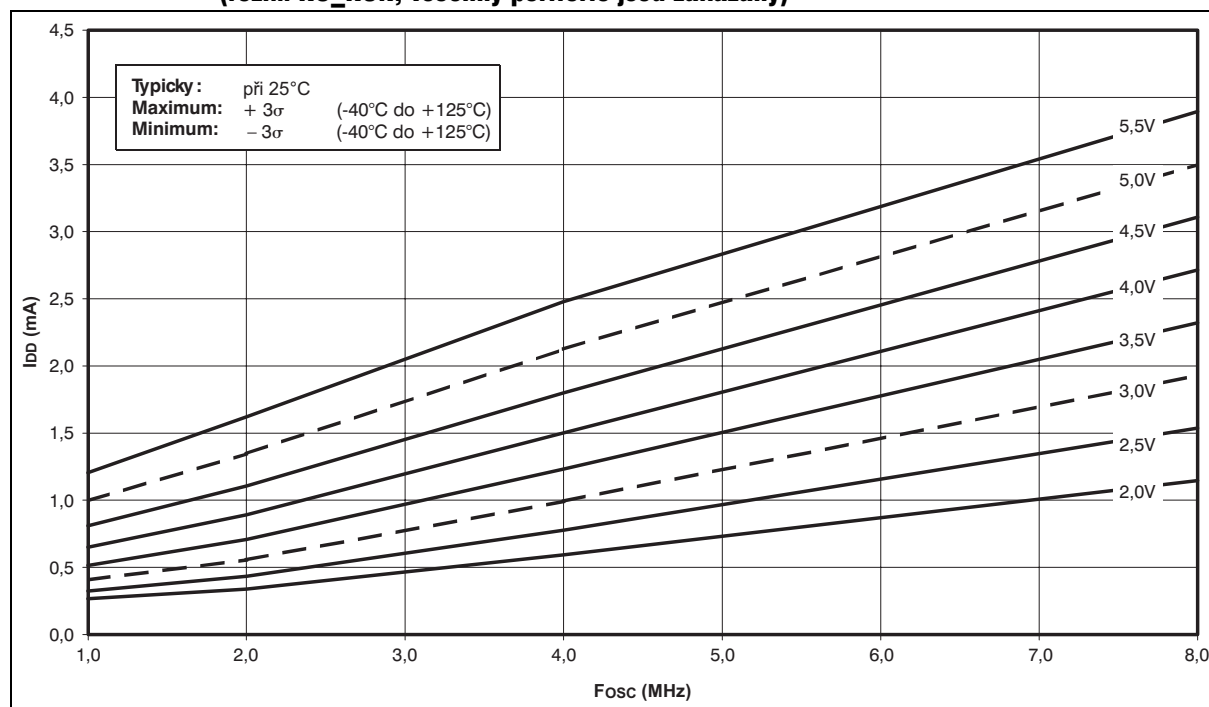


PIC16F87/88

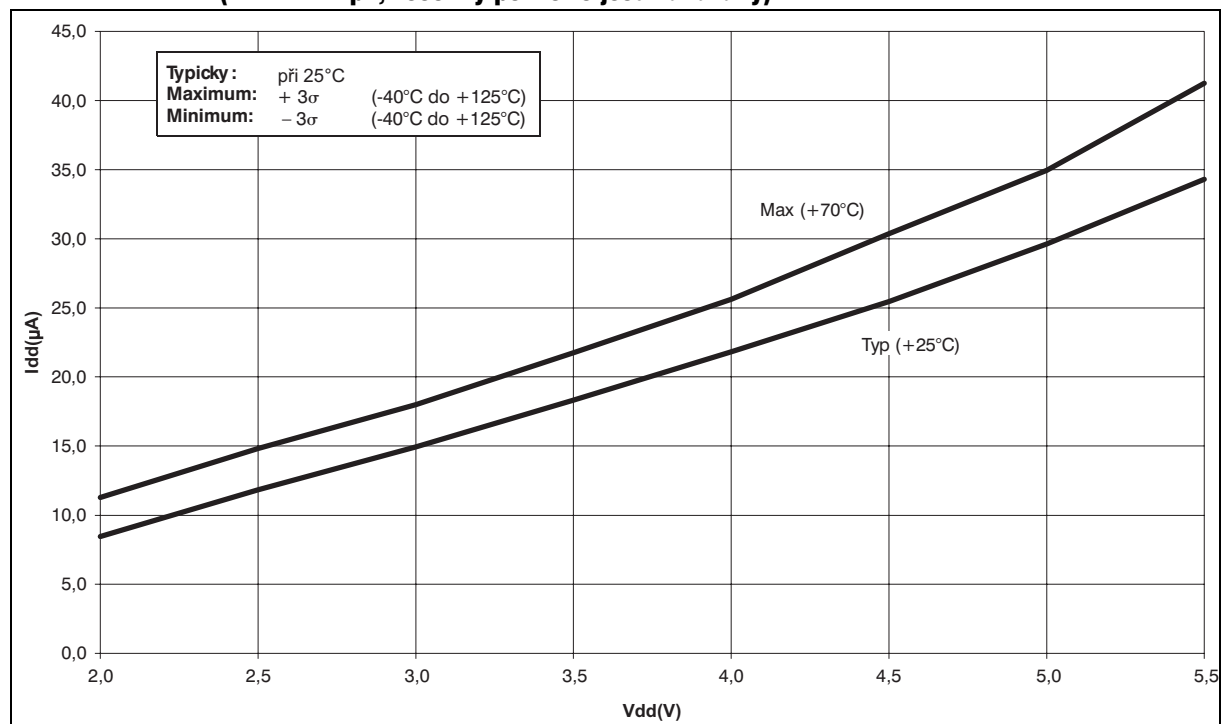
Obrázek 19-7: Typická závislost hodnoty proudu I_{DD} na napájecím napětí V_{DD} v rozsahu teplot -40 až $+125$ °C a rozsahu frekvencí od 1 MHz do 8 MHz (režim RC_RUN, všechny periferie jsou zakázány)



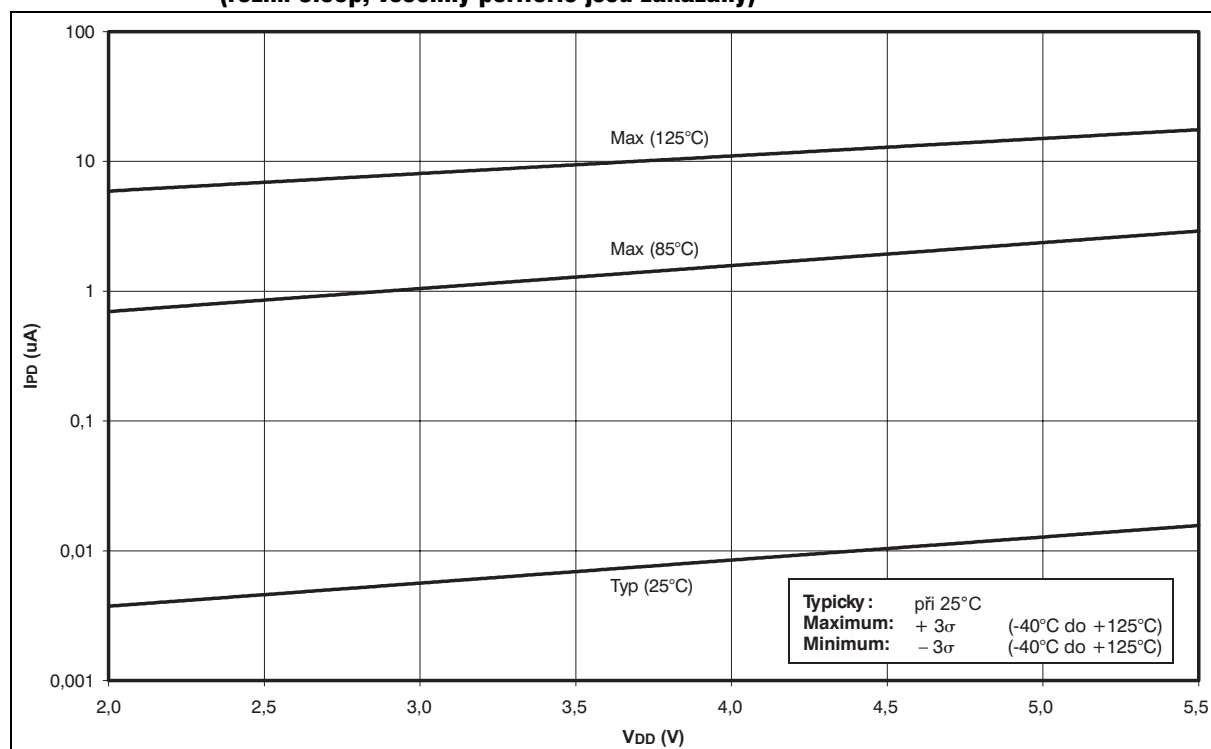
Obrázek 19-8: Závislost maximální hodnoty proudu I_{DD} na napájecím napětí V_{DD} v rozsahu teplot -40 až $+125$ °C a rozsahu frekvencí od 1 MHz do 8 MHz (režim RC_RUN, všechny periferie jsou zakázány)



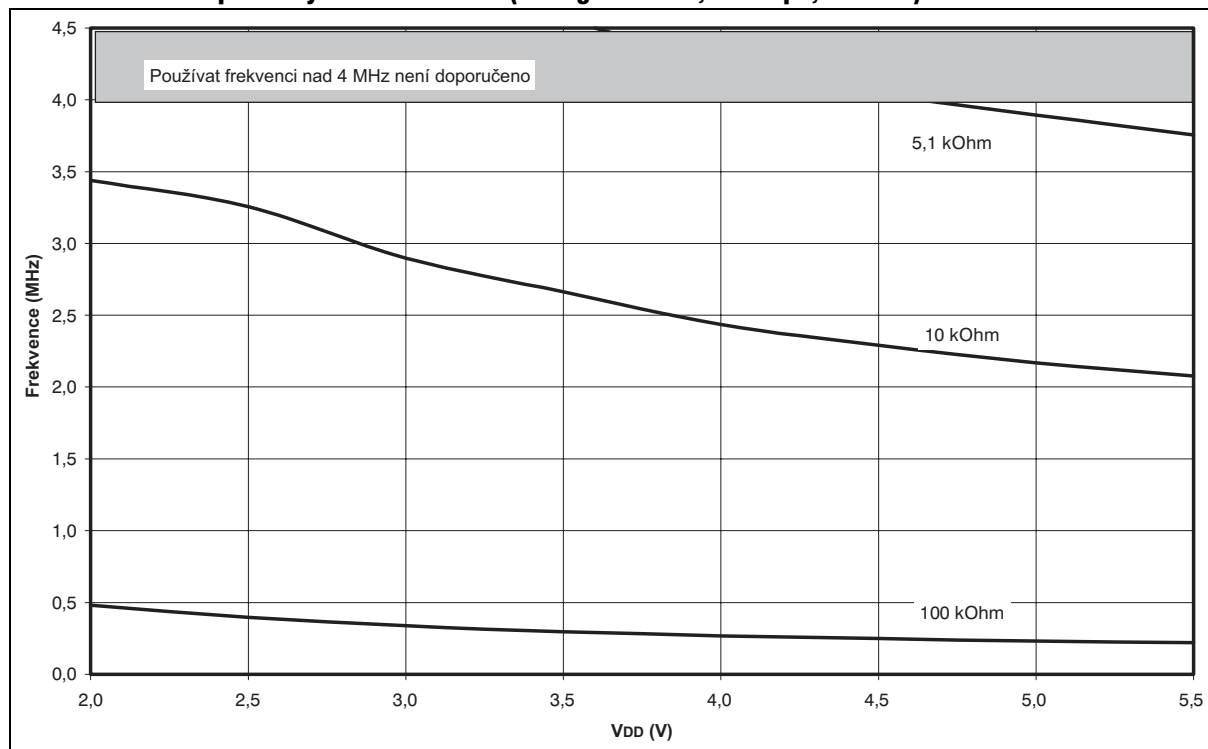
Obrázek 19-9: Závislost hodnoty proudu I_{DD} na napájecím napětí V_{DD} v rozsahu teplot -10 až +125 °C při frekvenci 32,768 kHz (XTAL 2x22pF, všechny periferie jsou zakázány)



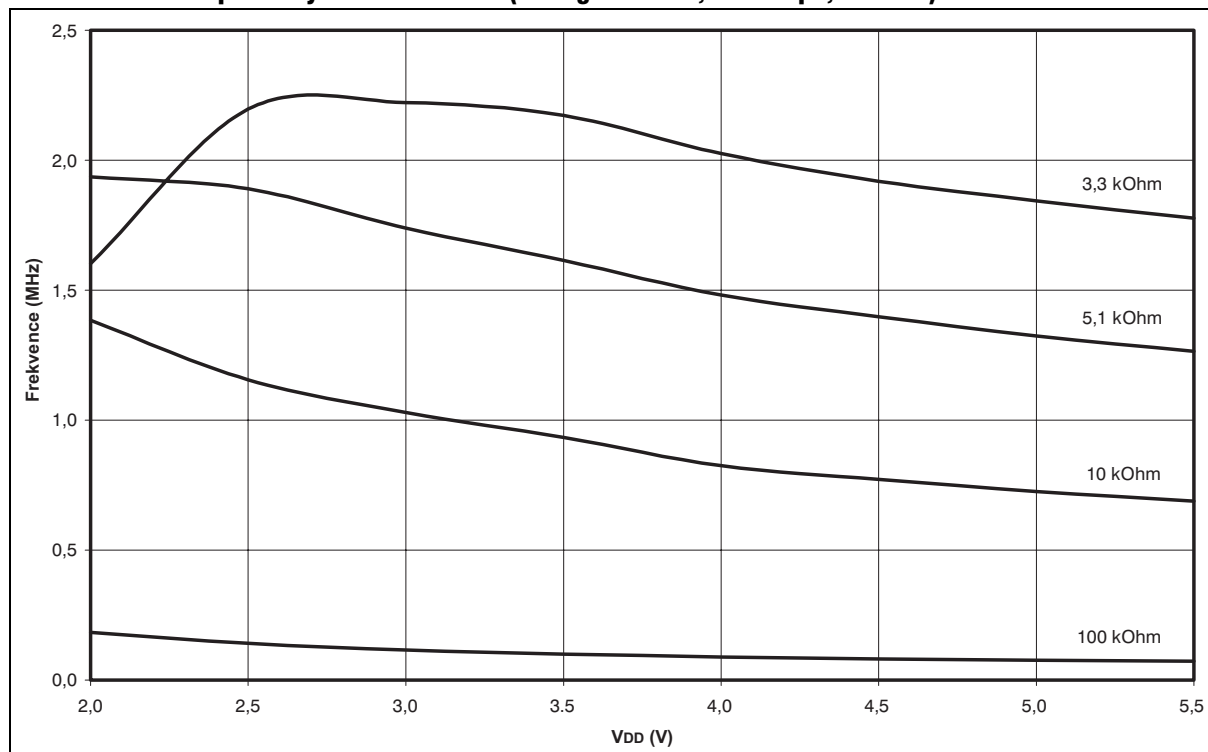
Obrázek 19-10: Závislost hodnoty proudu I_{DD} na napájecím napětí V_{DD} v rozsahu teplot -10 až +125 °C (režim Sleep, všechny periferie jsou zakázány)



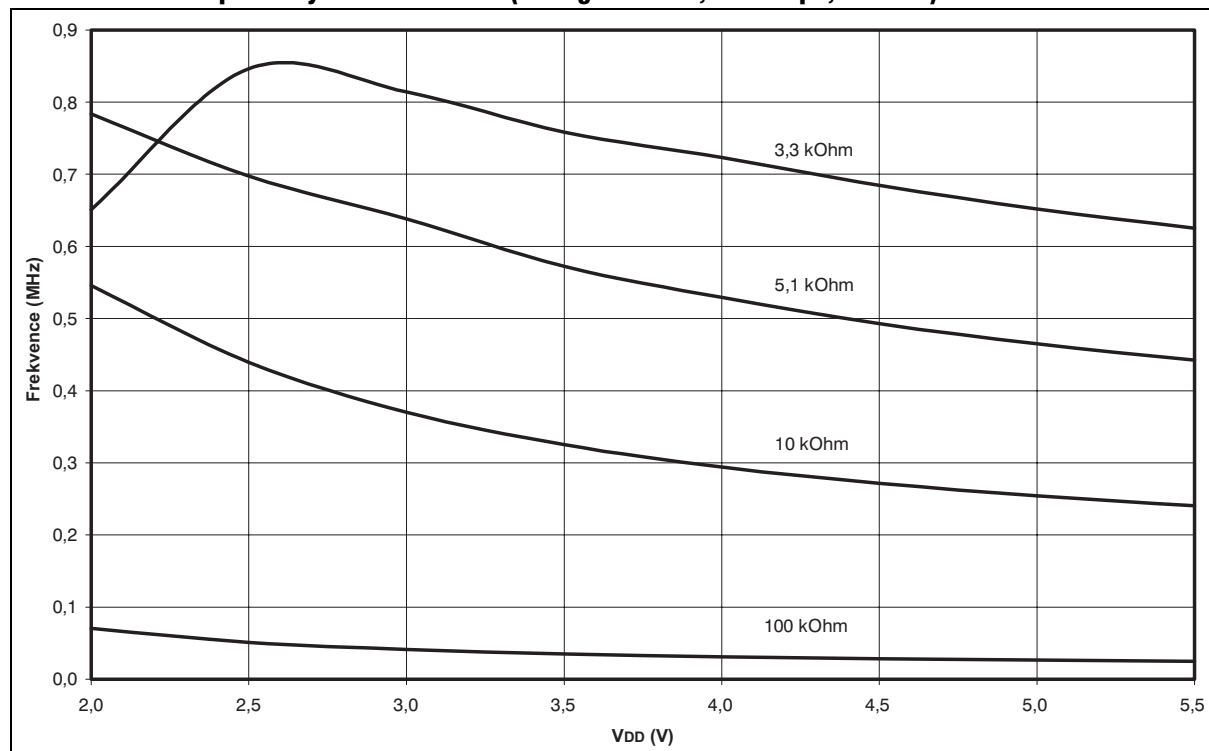
Obrázek 19-11: Závislost průměrné taktovací frekvence na napájecím napětí V_{DD} při různých hodnotách R (konfigurace RC, $C=20\text{pF}$, $+25\text{ }^{\circ}\text{C}$)



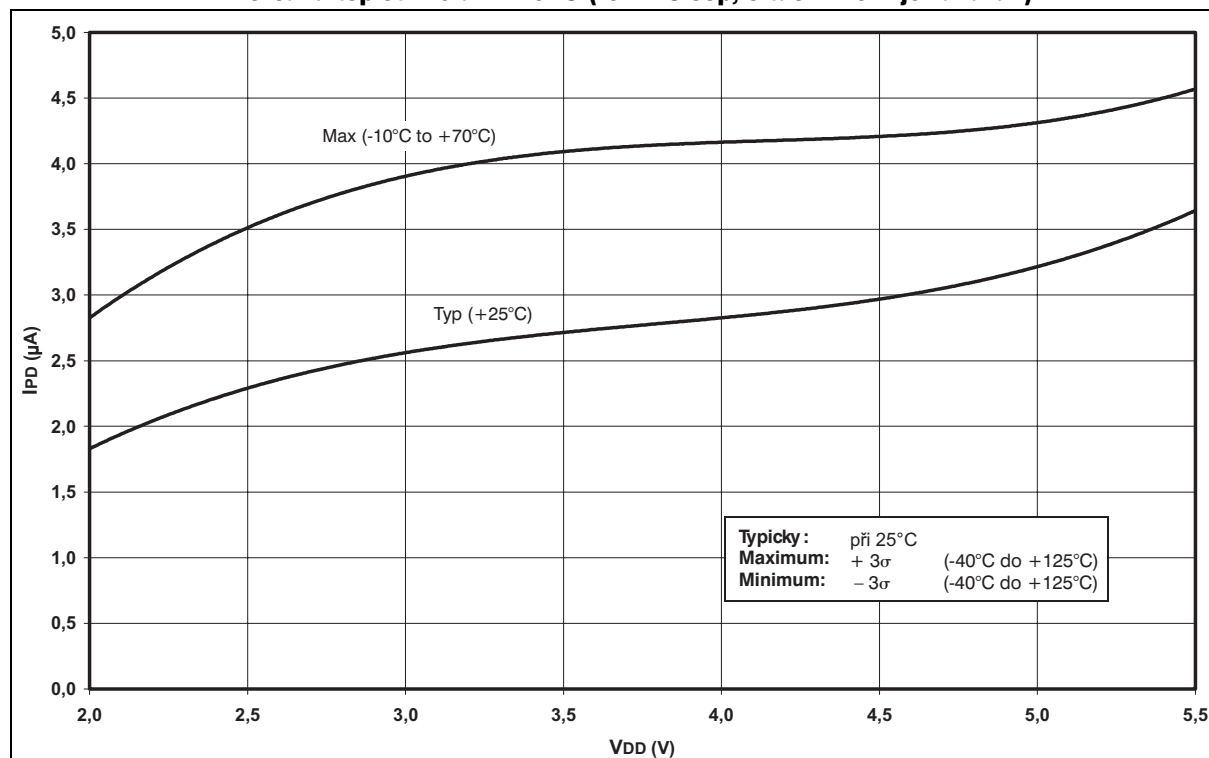
Obrázek 19-2: Závislost průměrné taktovací frekvence na napájecím napětí V_{DD} při různých hodnotách R (konfigurace RC, $C=100\text{pF}$, $+25\text{ }^{\circ}\text{C}$)



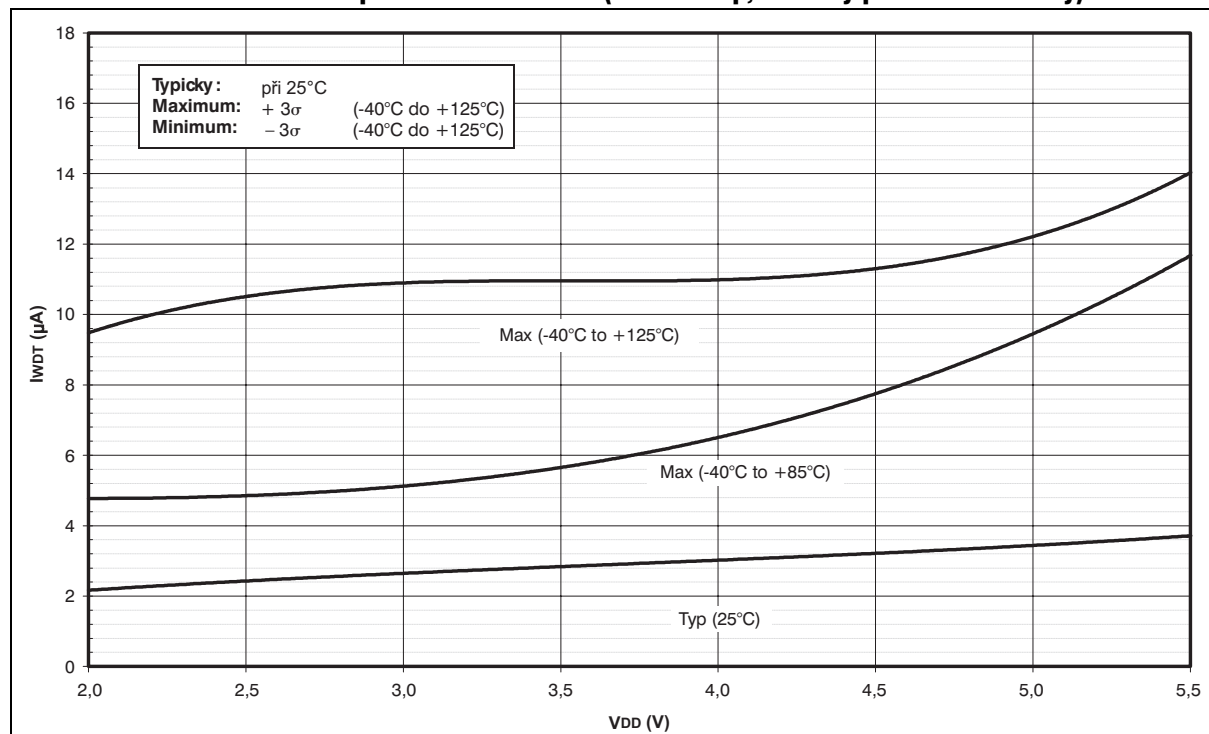
Obrázek 19-13: Závislost průměrné taktovací frekvence na napájecím napětí V_{DD} při různých hodnotách R (konfigurace RC, C=300pF, +25 °C)



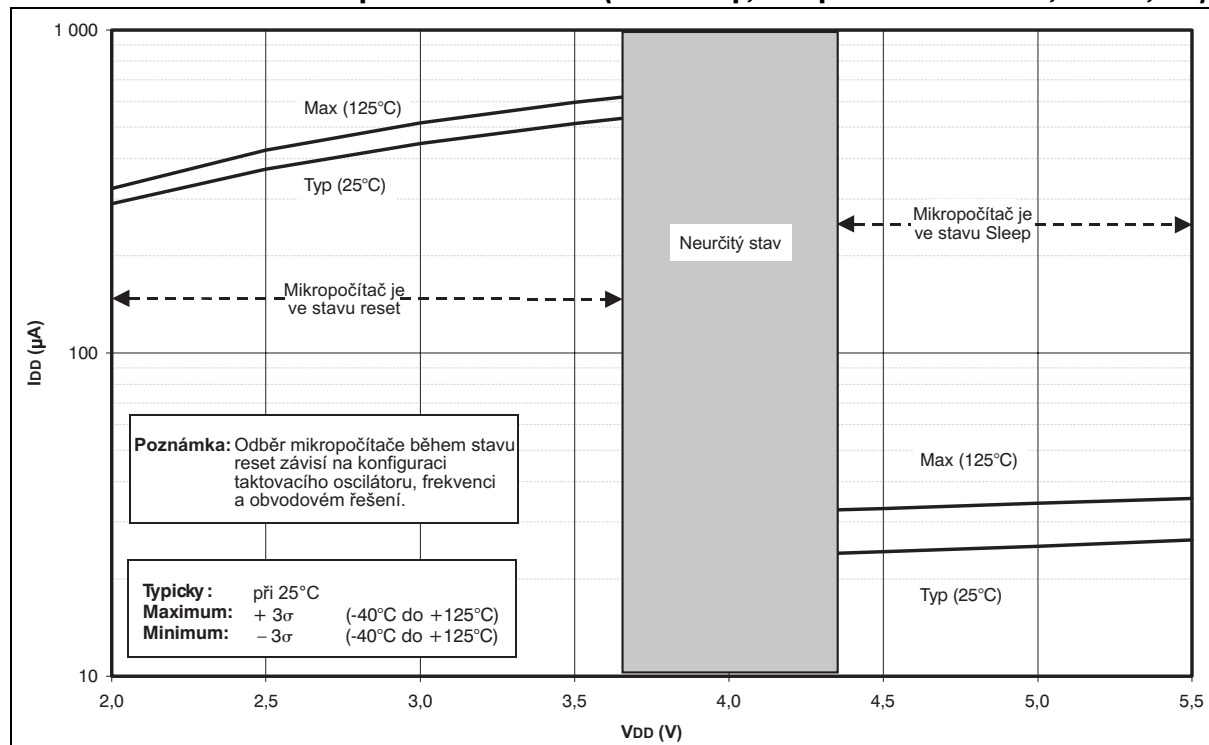
Obrázek 19-14: Závislost proudu delta I_{PD} oscilátoru Timer1 na napájecím napětí V_{DD} v rozsahu teplot -10 až +70 °C (režim Sleep, čítač Timer1 je zakázán)



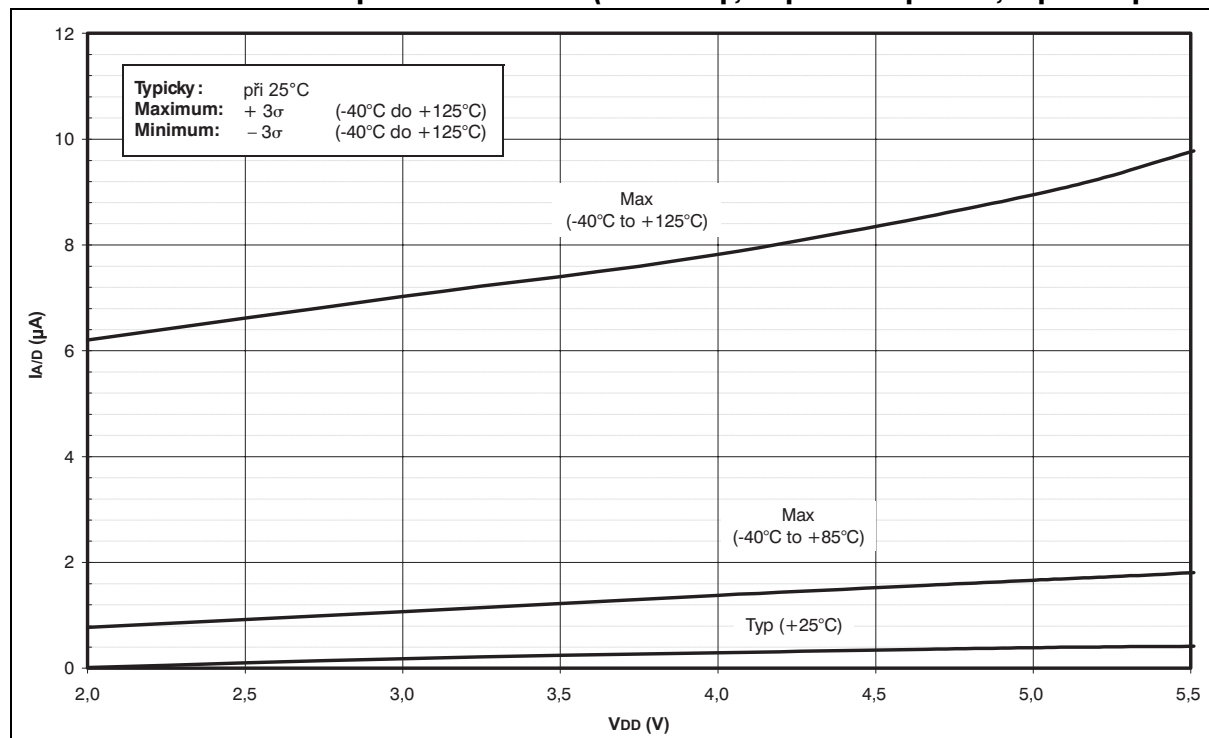
Obrázek 19-15: Závislost proudu delta I_{WDT} časovače Watchdog na napájecím napětí V_{DD} v rozsahu teplot -40 až +125 °C (režim Sleep, všechny periferie zakázány)



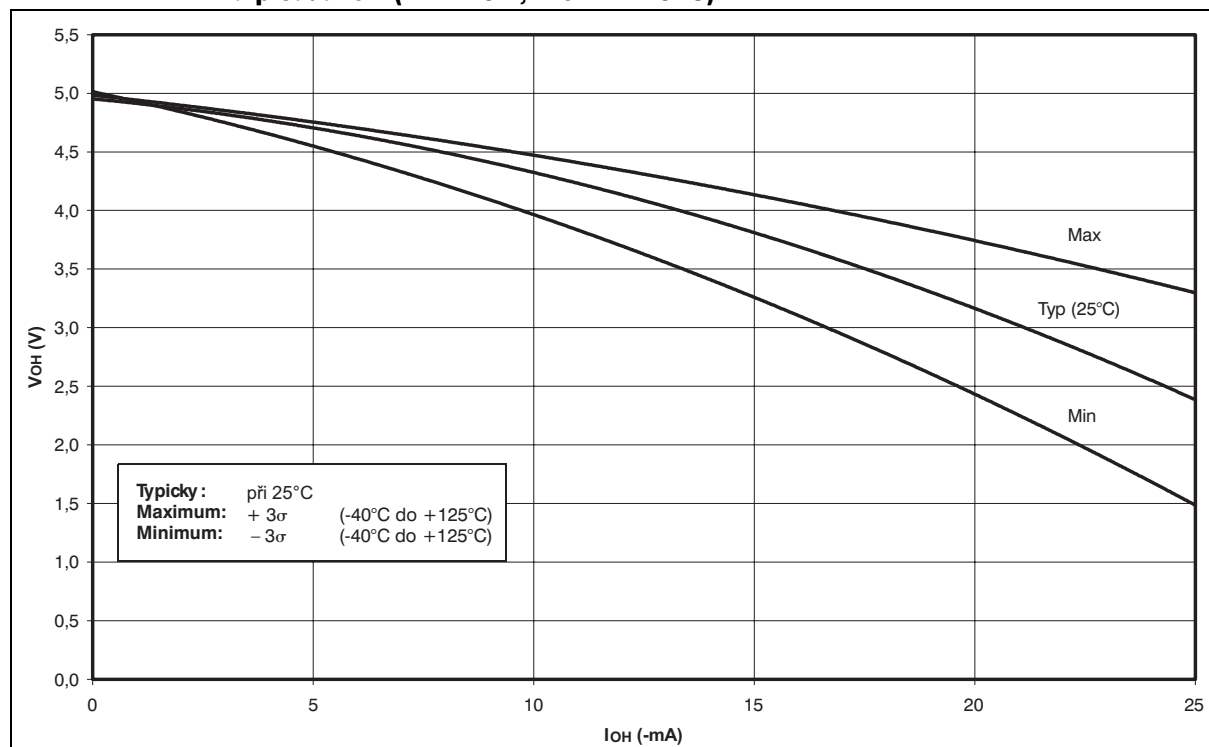
Obrázek 19-16: Závislost proudu delta I_{DD} obvodu BOR na napájecím napětí V_{DD} v rozsahu teplot -40 až +125 °C (režim Sleep, BOR povolen v rozsahu 2,00V – 2,16V)



Obrázek 19-17: Závislost proudu delta I_{PD} AD převodníku na napájecím napětí V_{DD} v rozsahu teplot -40 až +125 °C (režim Sleep, AD převodník povolen, neprobíhá převod)

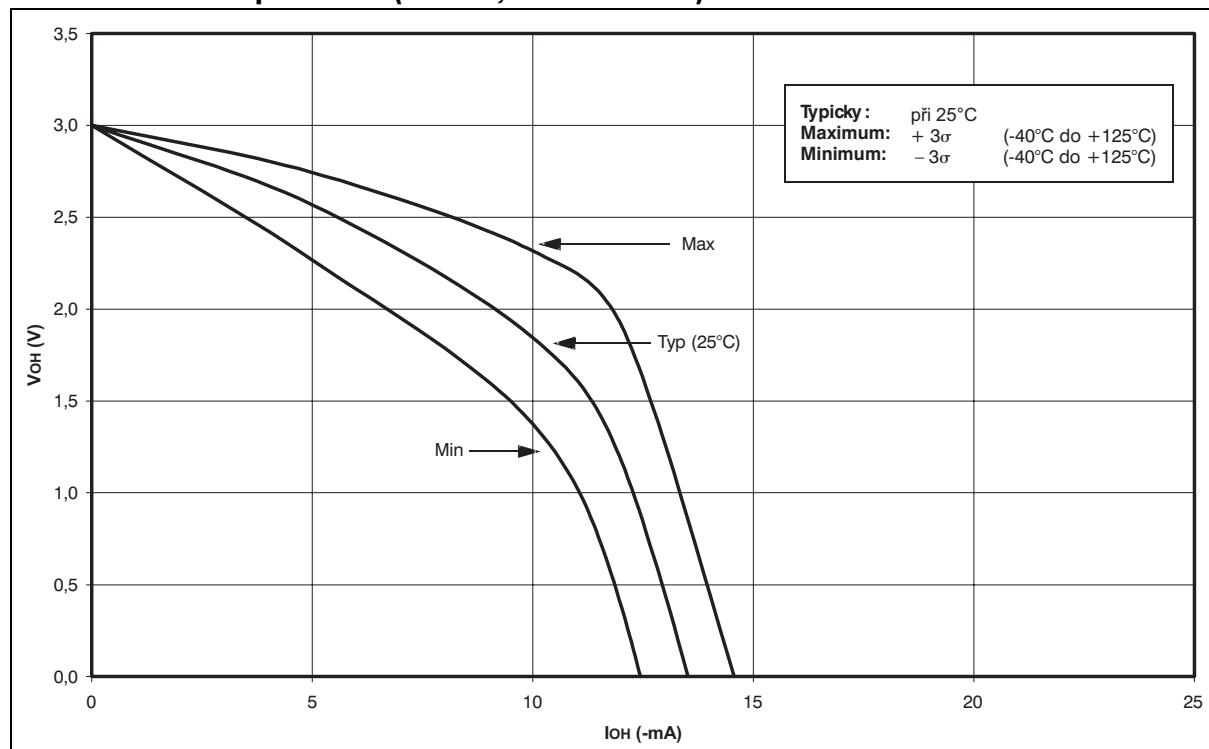


Obrázek 19-18: Závislost typické, minimální a maximální hodnoty napětí V_{OH} na proudu I_{OH} . ($V_{DD} = 5$ V, -40 – +125 °C)

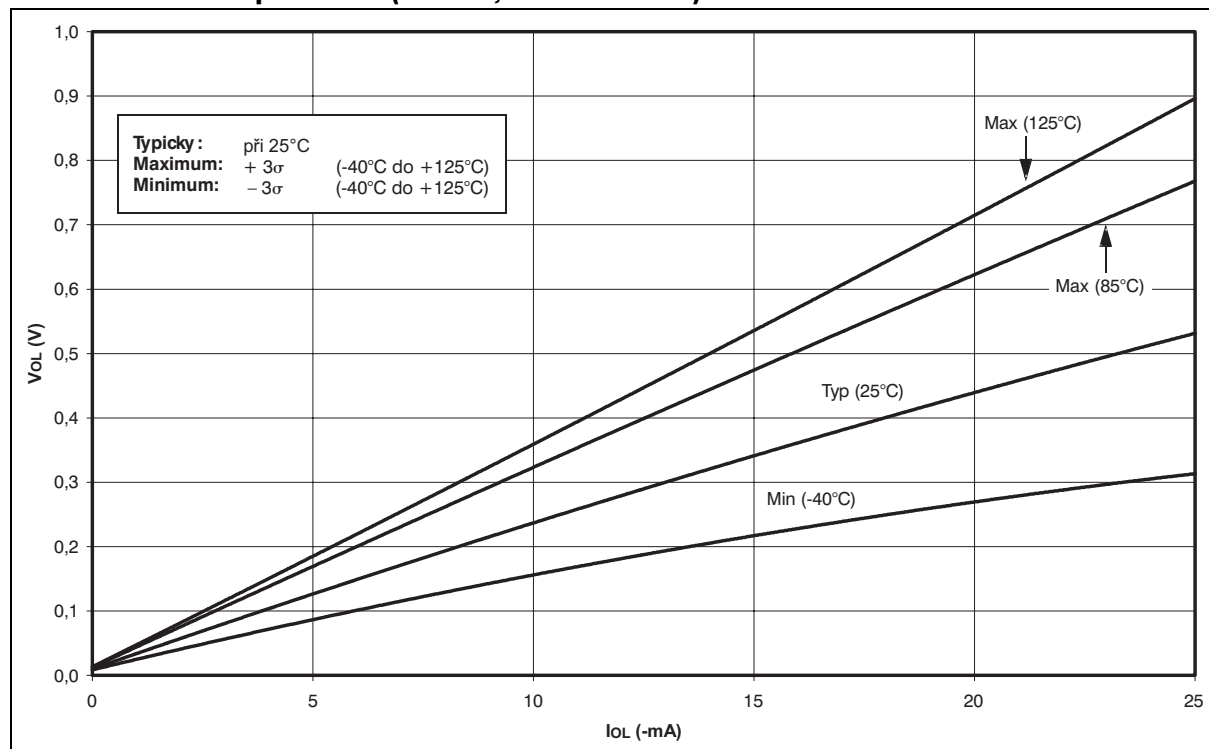


PIC16F87/88

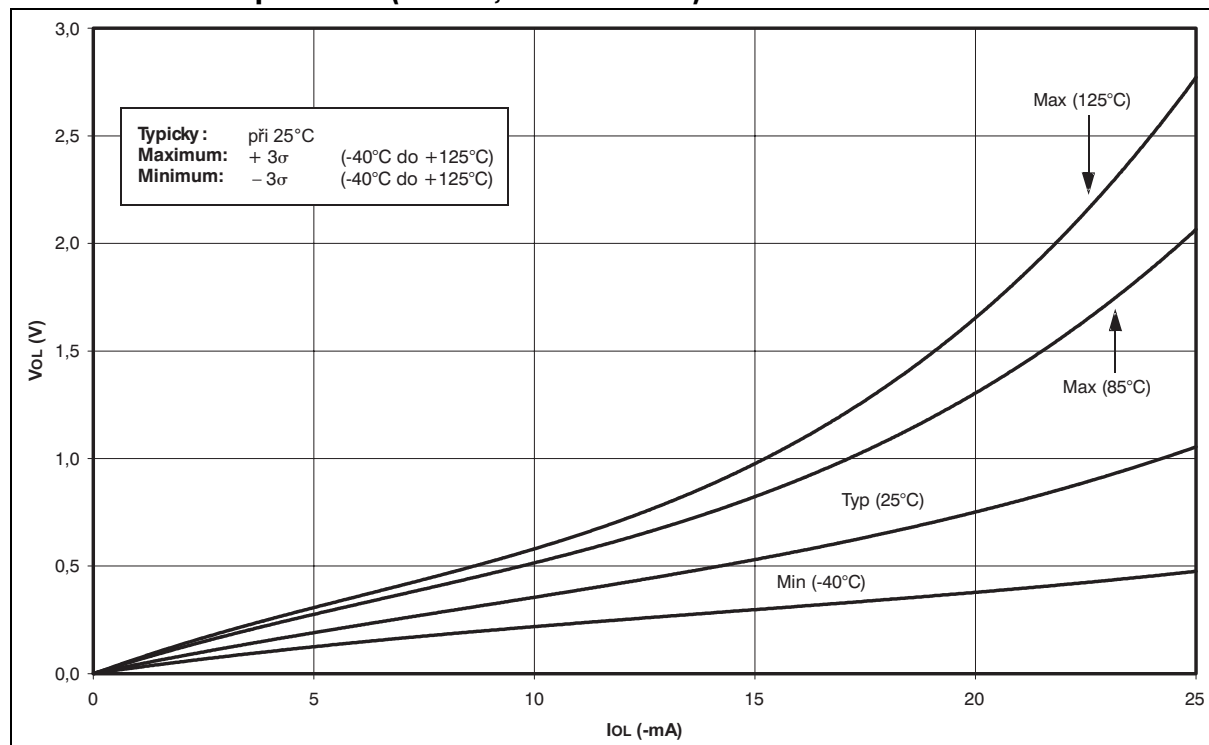
Obrázek 19-19: Závislost typické, minimální a maximální hodnoty napětí V_{OH} na proudu I_{OH} . ($V_{DD}=3V$, -40 až $+125$ °C)



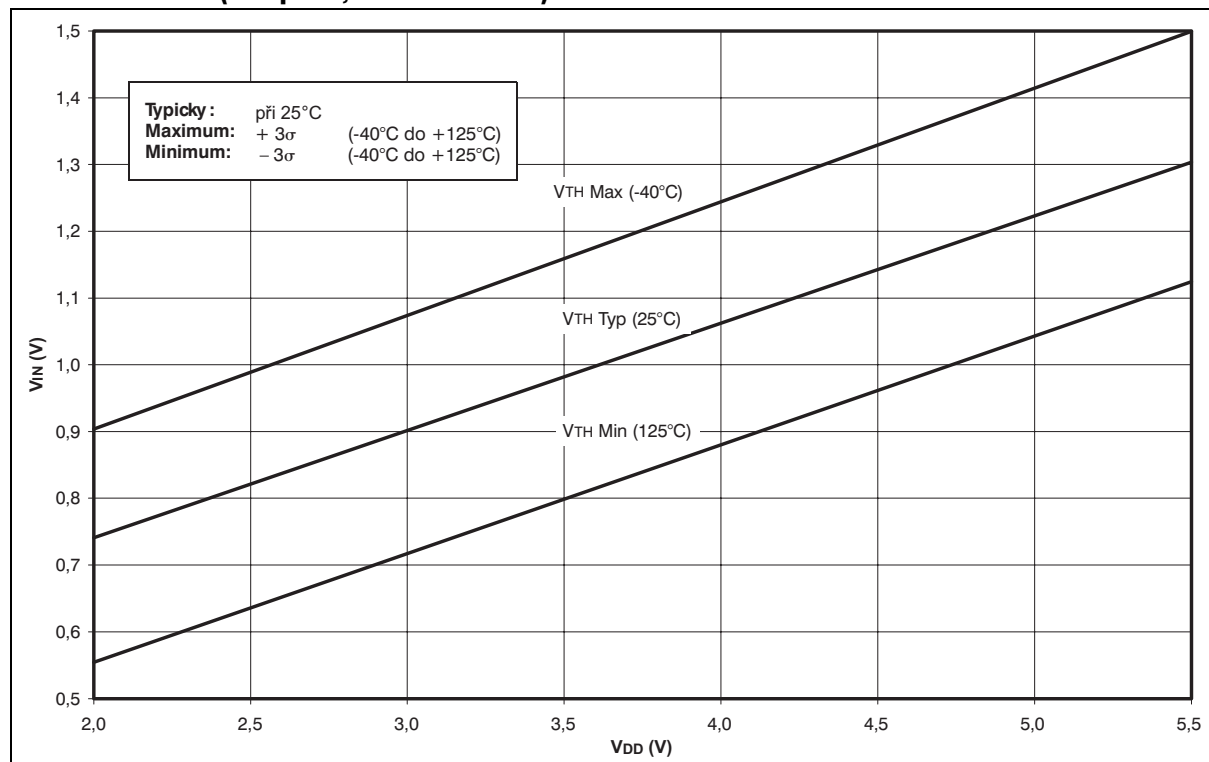
Obrázek 19-20: Závislost typické, minimální a maximální hodnoty napětí V_{OL} na proudu I_{OL} . ($V_{DD}=5V$, -40 až $+125$ °C)



Obrázek 19-21: Závislost typické, minimální a maximální hodnoty napětí V_{OL} na proudu I_{OL} . ($V_{DD}=3V$, -40 až $+125$ °C)

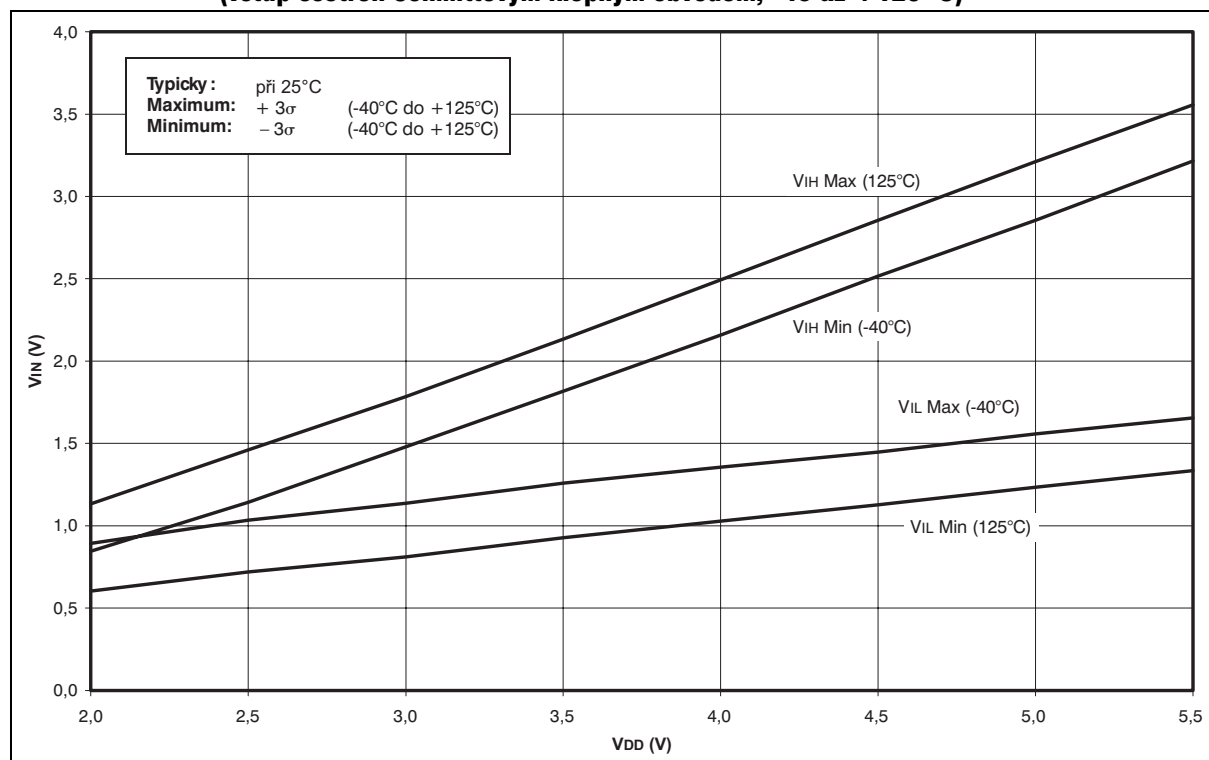


Obrázek 19-22: Závislost minimální a maximální hodnoty napětí V_{IN} na napájecím napětí V_{DD} . (Vstup TTL, -40 až $+125$ °C)

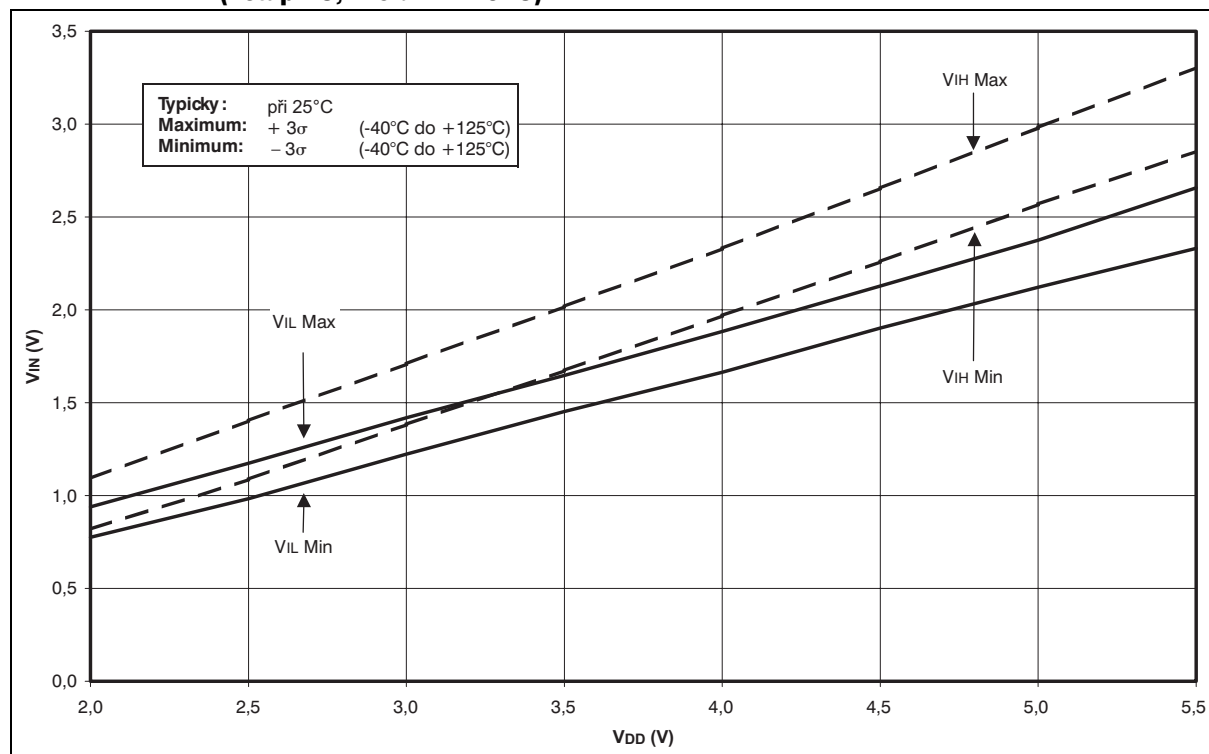


PIC16F87/88

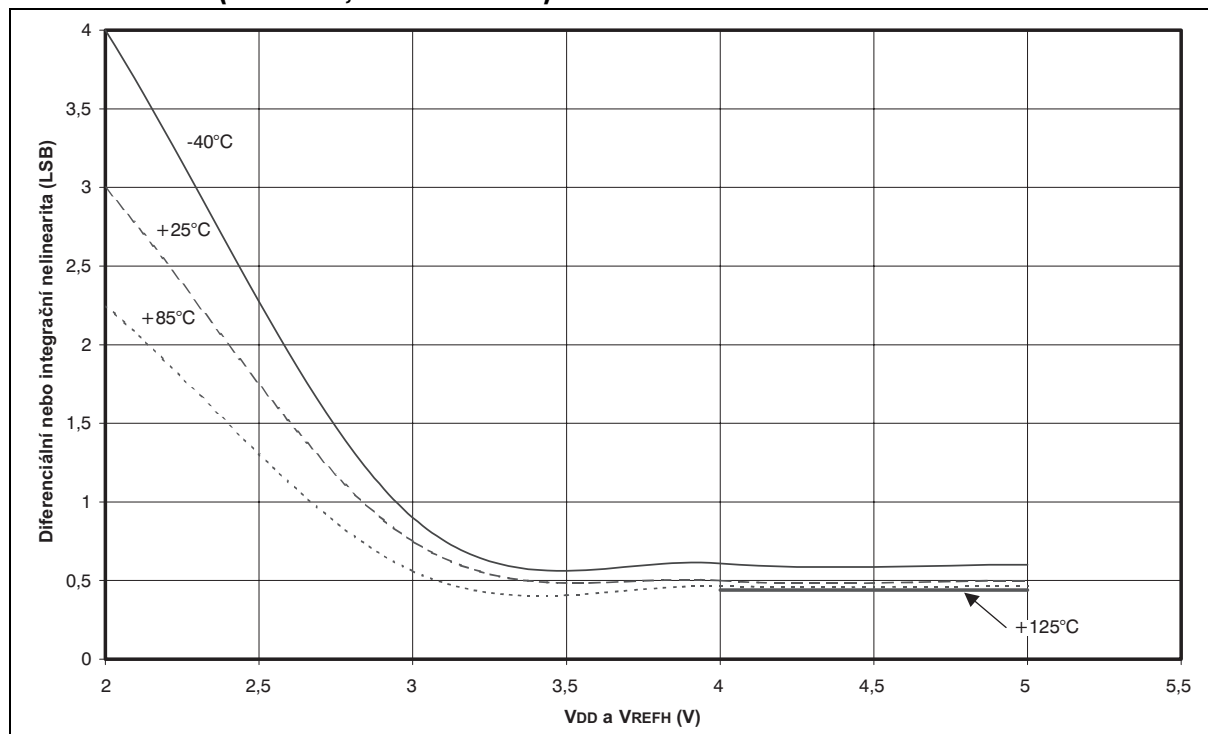
Obrazek 19-23: Závislost minimální a maximální hodnoty napětí V_{IH} na napájecím napětí V_{DD} (vstup ošetřen Schmittovým klopným obvodem, -40 až +125 °C)



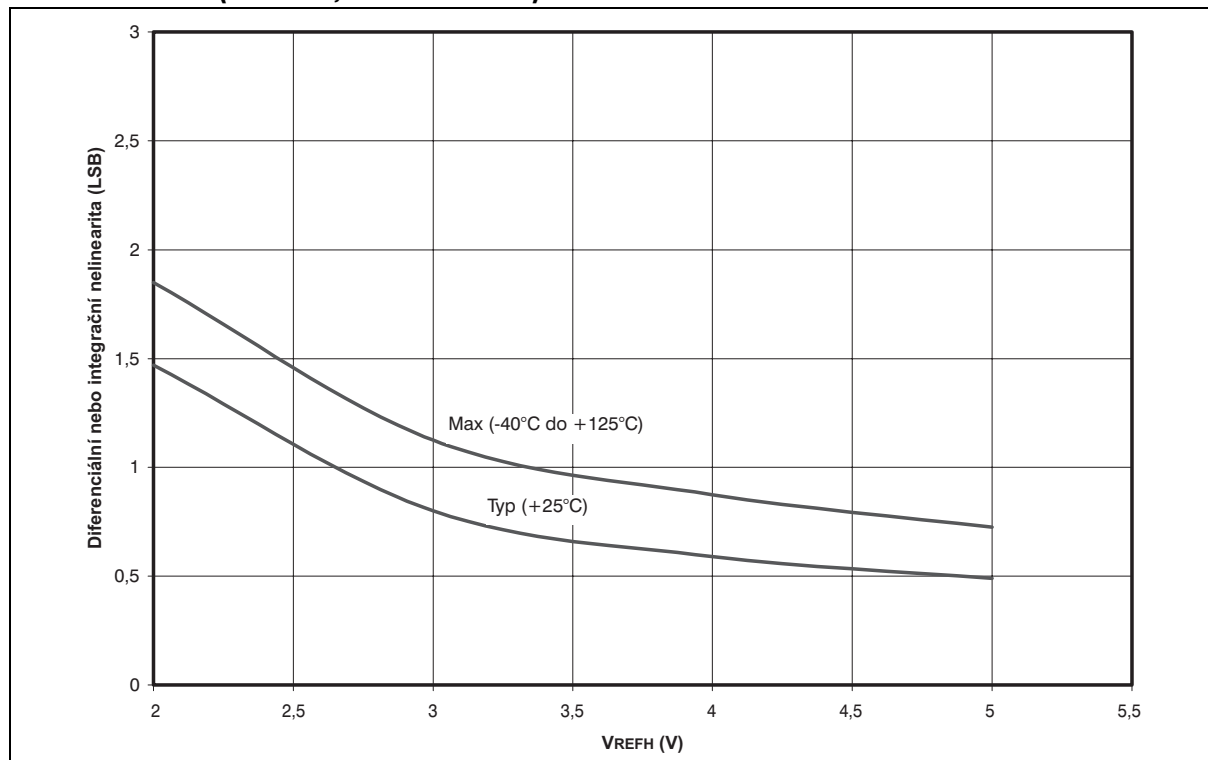
Obrazek 19-24: Závislost minimální a maximální hodnoty napětí V_{IH} na napájecím napětí V_{DD} (Vstup I²C, -40 až +125 °C)



Obrázek 19-25: Závislost nelinearity AD převodníku na referenčním napětí VREF (VDD=VREF, -40 až +125 °C)



Obrázek 19-26: Závislost nelinearity AD převodníku na referenčním napětí VREFH (VREF=5V, -40 až +125 °C)

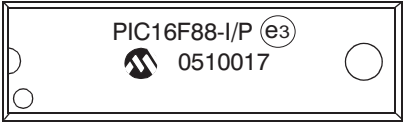
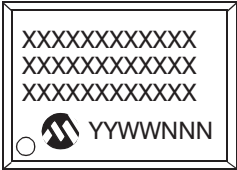
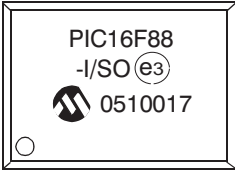
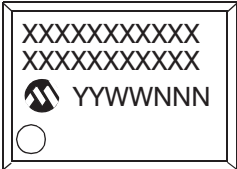
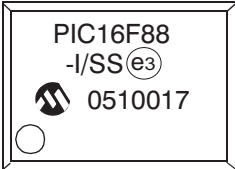


PIC16F87/88

Poznámky:

20 POUZDRA

20.1 Značení pouzder

18-Lead PDIP 	příklad 
18-Lead SOIC 	příklad 
20-Lead SSOP 	příklad 
28-Lead QFN 	příklad 

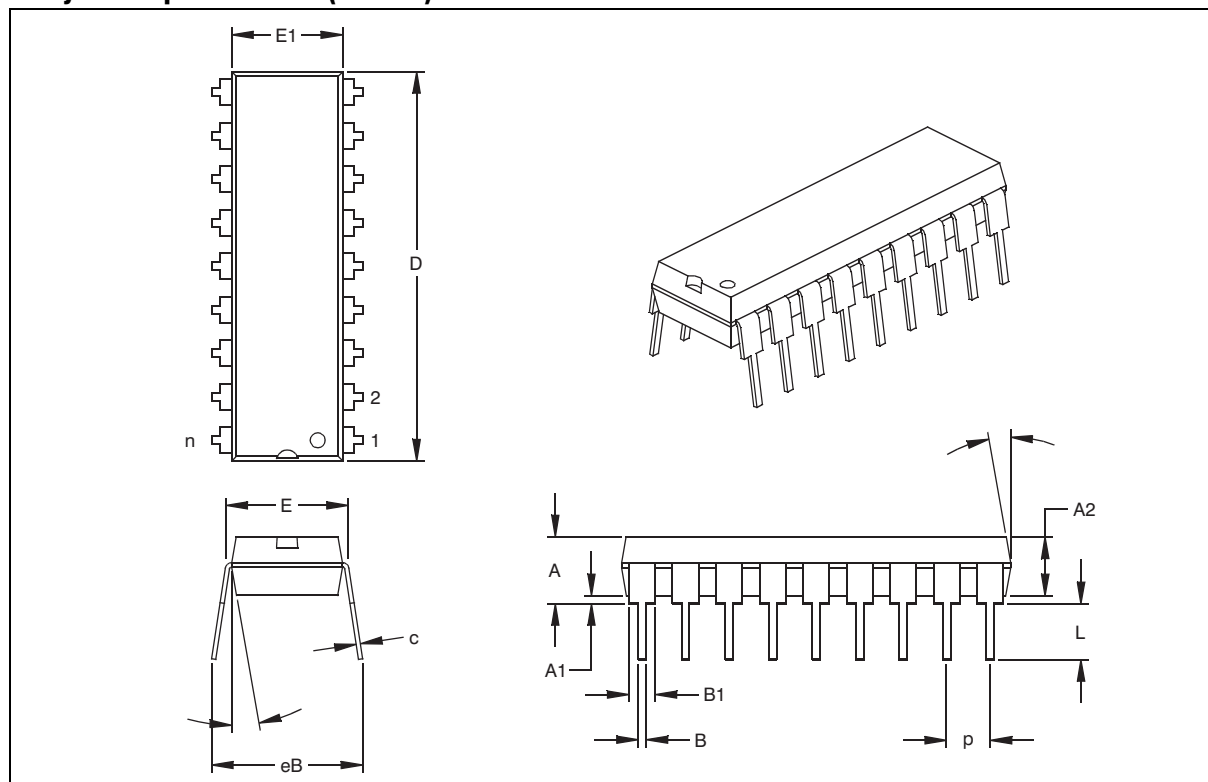
Význam:

XX...X	Zákaznický specifikovaná informace
Y	Rok výroby (poslední číslo kalendářního roku výroby)
YY	Rok výroby (poslední dvojčíslí kalendářního roku výroby)
WW	Týden výroby (první týden v roce je značen "01")
NNN	Alfanumerický identifikační kód
(e3)	bezolovnaté provedení
*	Toto pouzdro je v bezolovnatém provedení. Značka (e3) může být umístěna na vnějším obalu pouzder.

Poznámka: Pokud plné značení nemůže být umístěno na jednom řádku, pokračuje značení na druhém řádku s omezeným počtem znaků zákaznický specifikované informace.

PIC16F87/88

18-vývodové pouzdro PDIP (300 mil)



Jednotky		Palce*			Milimetry		
Tolerance		Minimální	Nominální	Maximální	Minimální	Nominální	Maximální
Počet vývodů	n	18			18		
Rozteč vývodů	p		,100			2,54	
Výška osazeného pouzdra	A	,140	,155	,170	3,56	3,94	4,32
Výška pouzdra bez vývodů	A2	,115	,130	,145	2,92	3,30	3,68
Vzdálenost dolní části pouzdra od dosedací části vývodů	A1	,015			0,38		
Vzdálenost mezi zaoblením vývodů	E	,300	,313	,325	7,62	7,94	8,26
Šířka pouzdra	E1	,240	,250	,260	6,10	6,35	6,60
Délka pouzdra	D	,890	,898	,905	22,16	22,80	22,99
Vzdálenost od dosedací úrovně vývodu k jeho konci	L	,125	,130	,135	3,18	3,30	3,43
Tloušťka vývodu	c	,008	,012	,015	0,20	0,29	0,38
Šířka horní části vývodu	B1	,045	,058	,070	1,14	1,46	1,78
Šířka dolní části vývodu	B	,014	,018	,022	0,36	0,46	0,56
Vzdálenost mezi dolními okraji vývodů	§ eB	,310	,370	,430	7,87	9,40	10,92
Úhel horního skosení čela pouzdra	α	5	10	15	5	10	15
Úhel dolního skosení boku pouzdra	β	5	10	15	5	10	15

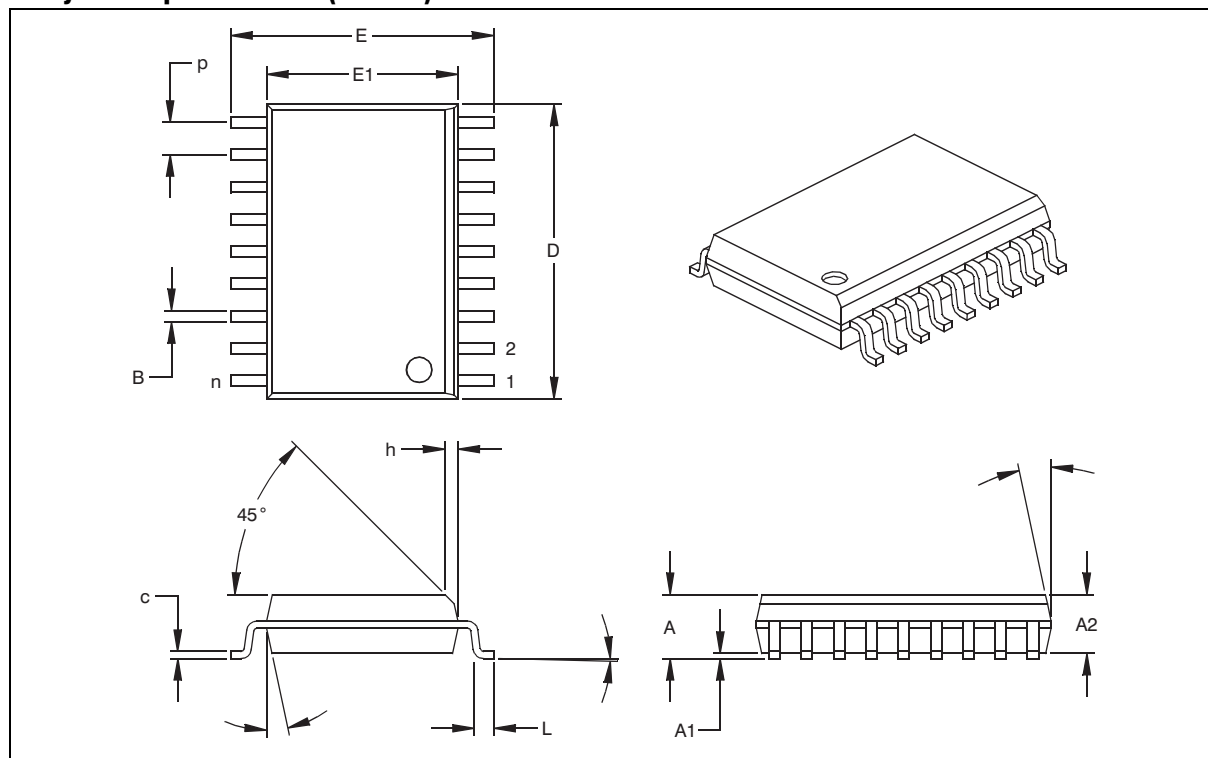
* Základní parametr

§ Významná hodnota

Poznámky: Rozměry D a E1 nezahrnují otřepy a výstupky.
Otřepy a výstupky nesmějí přesáhnout 0,01" (0,254 mm) na každé straně.

PIC16F87/88

18-vývodové pouzdro SOIC (300 mil)



	Jednotky	Palce			Milimetry		
	Tolerance	Minimální	Nominální	Maximální	Minimální	Nominální	Maximální
Počet vývodů	n	18			18		
Rozteč vývodů	p		,050			1,27	
Výška osazeného pouzdra	A	,093	,099	,104	2,36	2,50	2,64
Výška pouzdra bez vývodů	A2	,088	,091	,094	2,24	2,31	2,39
Tloušťka vývodu	§ A1	,004	,008	,012	0,10	0,20	0,30
Šířka pouzdra s vývody	E	,394	,407	,420	10,01	10,34	10,67
Šířka pouzdra	E1	,291	,295	,299	7,39	7,49	7,59
Délka pouzdra s vývody	D	,446	,454	,462	11,33	11,53	11,73
Délka pouzdra	h	,010	,020	,029	0,25	0,50	0,74
Délka pájecí části vývodu	L	,016	,033	,050	0,41	0,84	1,27
Úhel pájecí části vývodu	Φ	0	4	8	0	4	8
Tloušťka vývodu	c	,009	,011	,012	0,23	0,27	0,30
Šířka horní části vývodu	B	,014	,017	,020	0,36	0,42	0,51
Úhel horního skosení čela pouzdra	α	0	12	15	0	12	15
Úhel dolního skosení boku pouzdra	β	0	12	15	0	12	15

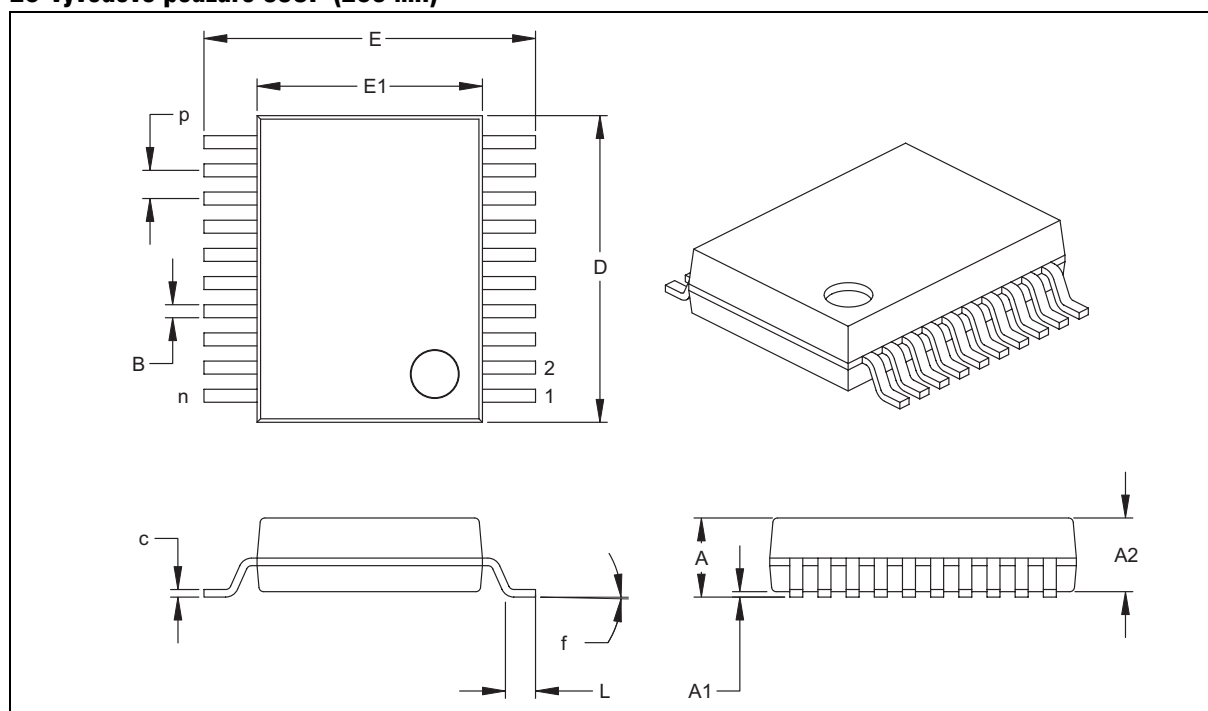
* Základní parametr

§ Významná hodnota

Poznámky: Rozměry D a E1 nezahrnují otřepy a výstupky.
Otřepy a výstupky nesmějí přesáhnout 0,01" (0,254 mm) na každé straně.

PIC16F87/88

20-vývodové pouzdro SSOP (209 mil)



Jednotky		Palce			Milimetry		
Tolerance		Minimální	Nominální	Maximální	Minimální	Nominální	Maximální
Počet vývodů	n	20			20		
Rozteč vývodů	p		,026			0,65	
Výška osazeného pouzdra	A	—	—	,079	—	—	2,00
Výška pouzdra bez vývodů	A2	,065	,069	,073	1,65	1,75	1,85
Mezera mezi dnem pouzdra a rovinou osazení	A1	,002	—	—	0,05	—	—
Šířka pouzdra s vývody	E	,291	,307	,323	7,40	7,80	8,20
Šířka pouzdra	E1	,197	,209	,220	5,00	5,30	5,60
Délka pouzdra s vývody	D	,272	,283	,289	,295	7,20	7,50
Délka pájecí části vývodu	L	,022	,030	,037	0,55	0,75	0,95
Tloušťka vývodu	c	,004	—	,010	0,09	—	0,25
Úhel pájecí části vývodu	f	0°	4°	8°	0°	4°	8°
Šířka horní části vývodu	B	,009	—	,015	0,22	—	0,38

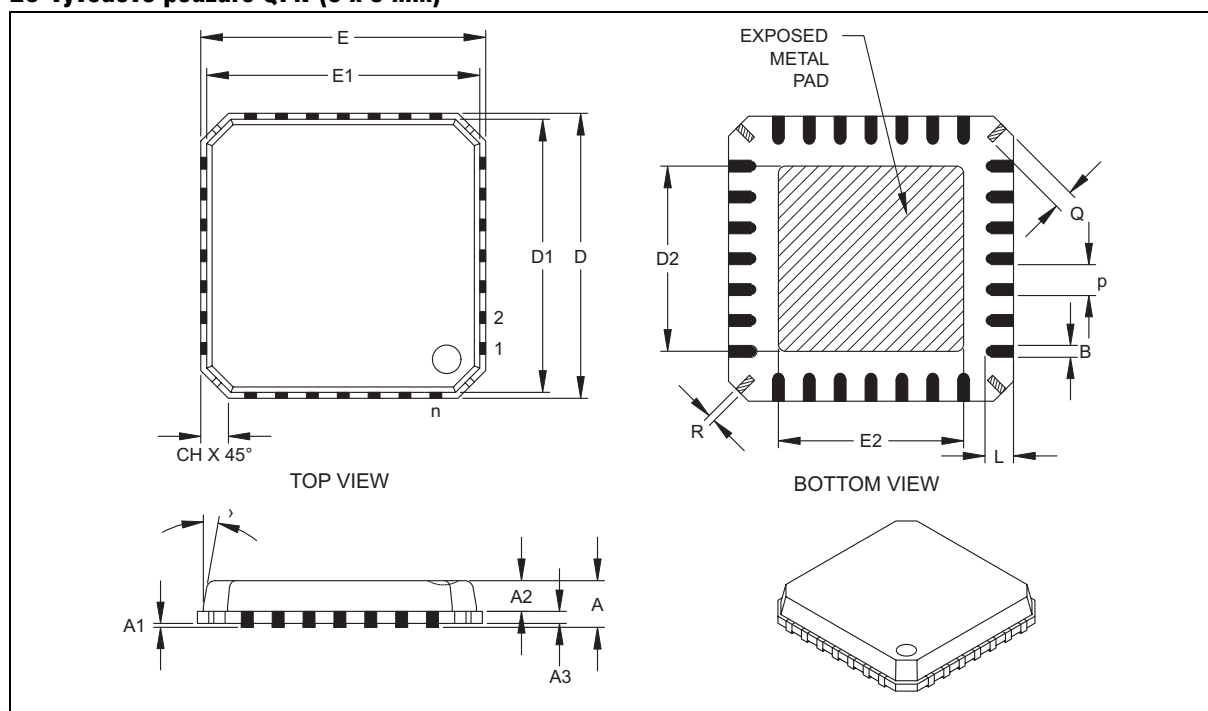
* Základní parametr

§ Významná hodnota

Poznámky: Rozměry D a E1 nezahrnují otřepy a výstupky.
Otřepy a výstupky nesmějí přesáhnout 0,01" (0,254 mm) na každé straně.

PIC16F87/88

28-vývodové pouzdro QFN (6 x 6 mm)



	Jednotky	Palce			Milimetry		
	Tolerance	Minimální	Nominální	Maximální	Minimální	Nominální	Maximální
Počet vývodů	n	28			28		
Rozteč vývodů	p	,026 BSC			0,65 BSC		
Výška osazeného pouzdra	A		,033	,039		0,85	1,00
Výška pouzdra bez vývodů	A2		,026	,031		0,65	0,80
Mezera mezi dnem pouzdra a rovinou osazení	A1	,000	,0004	,002	0,00	0,01	0,05
Výška vývodu	A3	,008 REF			0,20 REF		
Šířka pouzdra s vývody	E	,236 BSC			6,00 BSC		
Šířka pouzdra	E1	,226 BSC			5,75 BSC		
Rozteč krajních vývodů	E2	,140	,146	,152	3,55	3,70	3,85
Délka pouzdra s vývody	D	,236 BSC			6,00 BSC		
Délka pouzdra	D1	,226 BSC			5,75 BSC		
Rozteč krajních vývodů	D2	,140	,146	,152	3,55	3,70	3,85
Šířka vývodu	B	,009	,011	,014	0,23	0,28	0,35
Délka vývodu	L	,020	,024	,030	0,50	0,60	0,75
Šířka drážky	R	,005	,007	,010	0,13	0,17	0,23
Délka drážky	Q	,012	,016	,026	0,30	0,40	0,65
Délka skosení	CH	,009	,017	,024	0,24	0,42	0,60
Úhel skosení horní části pouzdra	α			12°			12°

* Základní parametr

Poznámky: Rozměry D a E1 nezahrnují otřepy a výstupky.
Otřepy a výstupky nesmějí přesáhnout 0,01" (0,254 mm) na každé straně.

PIC16F87/88

Poznámky:

PIC16F87/88

Identifikační systém PIC16F87/88

Pro přesné objednání kontaktujte svého dodavatele

<u>PART NO.</u>	<u>X</u>	<u>/XX</u>	<u>XXX</u>
Typ	Teplotní rozsah	Pouzdro	Speciální požadavky
Typ PIC16F87: Běžný rozsah pracovního napětí PIC16F87T: (balení v pásce pro povrchovou montáž) PIC16LF87: Rozšířený rozsah pracovního napětí			
Teplotní rozsah - = 0°C to +70°C I = -40°C to +85°C (Industrial) E = -40°C to +125°C (Extended)			
Pouzdro P = PDIP SO = SOIC SS = SSOP ML = QFN			
Speciální požadavky Kódy pro QTP, SQTP, ROM nebo speciální požadavky. Verze OTP a EPROM bez označení.			

Příklady:

a) PIC16F87-I/P = průmyslové provedení (Industrial), pouzdro PDIP, rozšířený rozsah Vdd.

b) PIC16LF87-I/SO = průmyslové provedení (Industrial), pouzdro SOIC, běžný rozsah Vdd.

Poznámka 1: F = CMOS Flash
LF = nízkopříkonná CMOS Flash

2: T = balení v pásce pro SMT - SOIC a SSOP