# A PIC18 mikrovezérlő család

Elektronikai rendszerek laboratóriumi mérést előkészítő előadás

#### PIC mikrovezérlők

8 bites 16 bites

**10Fxxx** (6-pin)

**12Cxxx**, **12Fxxx** (8-pin)

16C5x (baseline)

16Cxxx, 16Fxxx (mid-range)

**17Cxxx** (high performance)

18Fxxx (enhanced architecture)

24Fxxx

dsPIC30Fxxx (DSC)

dsPIC33Fxxx (DSC)

Kb. 250 típus

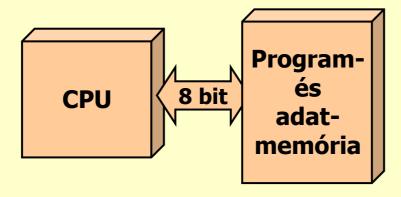
## Főbb jellemzők:

- 16 bites utasításhossz
- 1Mw FLASH programmemória (max.)
- 4kB RAM adatmemória (max.)
- EEPROM adatmemória
- 2 megszakításvektor
- 10 bites A/D átalakító
- Széles tápfeszültség-tartomány (2 5,5V)
- ICSP (2 lábon, pl. 64kw programmemória: 2s alatt

## A PIC18 család szerkezetének legfontosabb jellemzői:

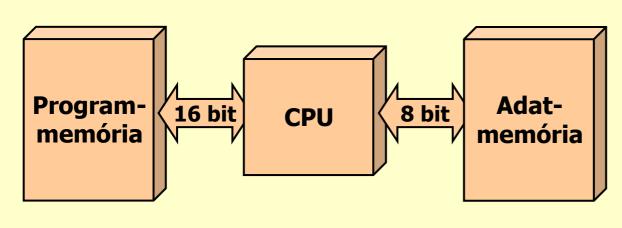
- Harvard architektúra
- Utasítás pipelining
- Regiszterfájl (speciális funkciójú regiszterek + adatmemória)
- Csökkentett utasításkészlet (RISC)
- Egyciklusú utasítás végrehajtás (néhány kivétel)
- Hosszúszavas utasítások (LWI)
- Ortogonális utasításkészlet
- Hardveres veremtár

#### Neumann architektúra



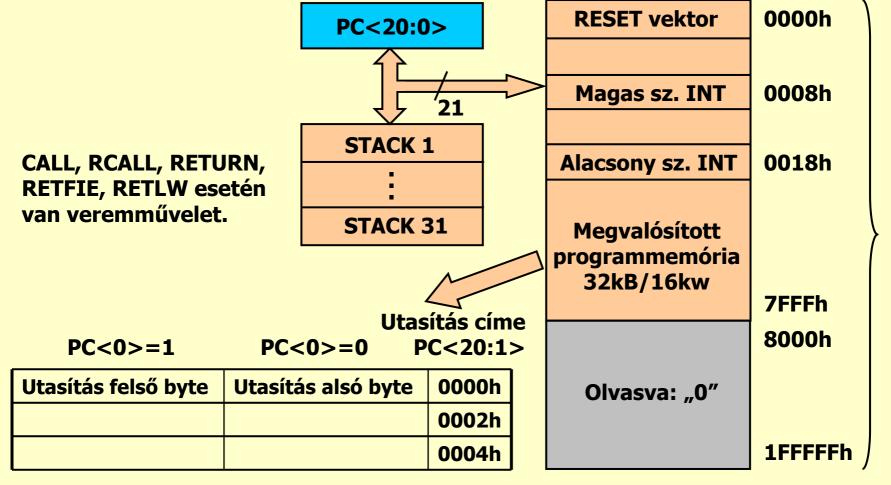
- A program és az adatok fizikailag egy memóriában vannak eltárolva
- Korlátozza a programvégrehajtás sebességét

#### Harvard architektúra



- A program és az adatok fizikailag külön memóriában vannak eltárolva
- Növeli a programvégrehajtás sebességét
- Különböző busz-szélesség lehet a két memória felé

# Programmemória (18F452)



# **Utasítás pipelining**

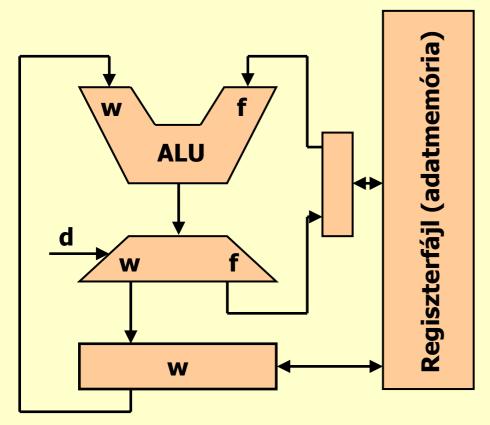
 $T_c=1$  $T_c=2$  $T_c=3$  $T_C=4$  $T_c=5$ **Utasításciklus:**  $T_c=0$ 1. ut. lehív. 1. ut. vh. 1. MOVLW 0x55 2. ut. vh. 2. ut. lehív. 2. MOVWF PORTA 3. ut. lehív. 3. ut. vh. 3. CALL **SUB** 4. ut. lehív. ürítés PORTA, 0 4. BSF SUB 1. ut. lh. SUB 1. ut. vh. 5. SUB első utasítása

- $\cdot T_{c} = 4xT_{osc}, T_{osc} = 1/f_{osc}$
- Az utasítások lehívása és végrehajtása átlapoltan történik, így minden utasításciklusban újabb utasítás hajtódik végre (kivétel: ugró utasítások)

ut. lehív: utasítás lehívása (fetch)

ut. vh.: utasítás végrehajtása (execute) ürítés: erőltetett NOP végrehajtása (flush)

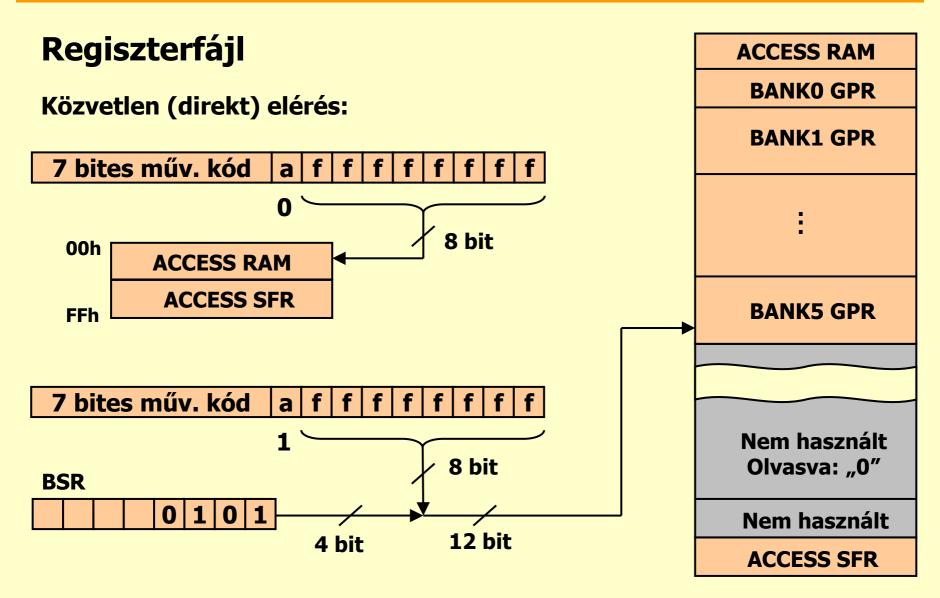
## Regiszterfájl



- A teljes adatmemória egy regiszterfájlban van elhelyezve
- A hosszúszavas utasítások lehetővé teszik a teljes memória közvetlen címzését
- Bármely memóriarekesz közvetlenül elérhető
- Minden periféria az adatmemóriában elhelyezkedő regisztereken keresztül kezelhető
- Az utasításkészlet ortogonális, azaz minden parancs minden regiszteren használható

Pl.: addwf regiszter, w addwf regiszter, f

	_			
Regiszte	erfájl (18F45	2)	000h 07Fh	ACCESS RAM
		BANKO -	080h	BANKO GPR
a=0 eseté	en:		100h	BANK1 GPR
	000h		1FFh	
	07Fh 080h		200h	
	0FFh			:
Bankváltás	nem lehetséges (A	ccess Bank)	4FFh	
			500h	
a=1 eseté	en:		5FFh	BANK5 GPR
BSR<3:0>	Kiválasztott BANK		600h	
0000	BANK0			
0001	BANK1			Nem használt
0010	BANK2			Olvasva: "0"
			EFFh	
1111	BANK15	BANK15 -	F00h F7Fh	Nem használt
		DAINTS	F80h	ACCESS SFR
DSK. Dalik	Select Register		FFFh	

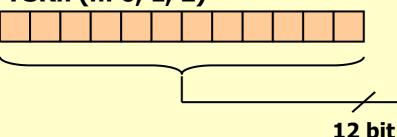


**ACCESS RAM** 

# Regiszterfájl

Közvetett (indirekt) elérés:

FSRn (n: 0, 1, 2)



Az INDFn (n: 0, 1, 2) regiszterbe írás, vagy olvasás hatására az FSR által kijelölt regiszteren hajtódik végre a művelet.

INDFn NEM fizikai regiszter! Indirekt módon elérve INDFn-t, olvasáskor 00h, az írásnak nincs hatása. Nem használt

Olvasva: "0"

**ACCESS SFR** 

# Regiszterfájl

#### SFR - Speciális Funkciójú Regiszterek

- A processzor és a perifériák működtetésével kapcsolatos regisztercsoport.
- A fontosabb perifériákkal kapcsolatos SFR-eket az adott perifériánál tárgyaljuk.
- Processzormaggal kapcsolatos példa: STATUS regiszter

**ACCESS SFR** 

## Regiszterfájl

STATUS regiszter (az ALU aritmetikai állapotát mutatja)

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
-	-	-	N	OV	Z	DC	С

N: negatív bit: értéke 1, ha az előjeles 2's compl. számmal végzett művelet eredménye negatív.

OV: túlcsordulás bit: értéke 1, ha az előjeles 2's compl. számmal végzett műveletkor a szám abszolút értéke nem fér el 7 biten.

Z: nulla bit: értéke 1, ha az elvégzett aritmetikai vagy logikai művelet eredménye nulla.

DC: köztes átvitel/áthozat bit. Összeadásnál értéke 1, ha a az alsó félbájtról volt átvitel a felsőre. Kivonásnál értéke 1, ha nem volt köztes áthozat.

C: átvitel/áthozat bit. Összeadásnál értéke 1, ha az MSB-ről volt átvitel. Kivonásnál értéke 1, ha nem volt áthozat.

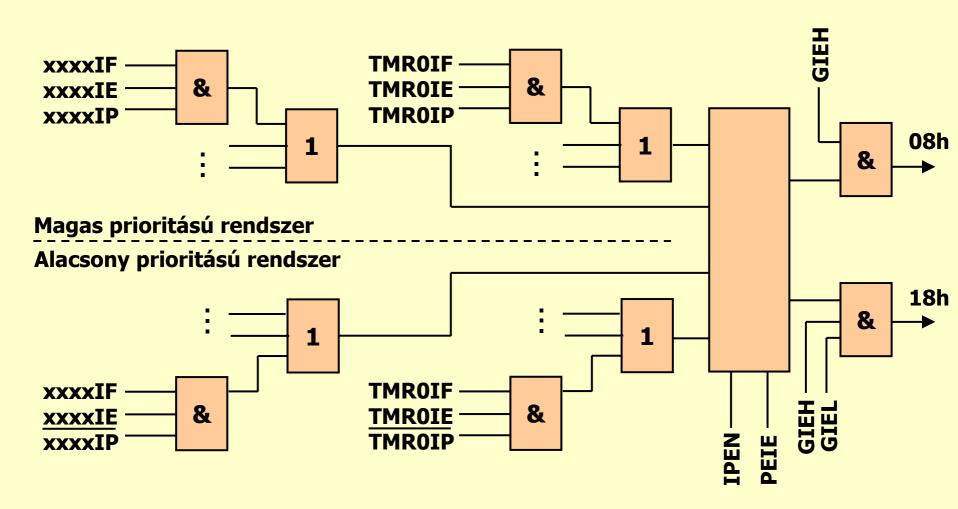
## Megszakítások

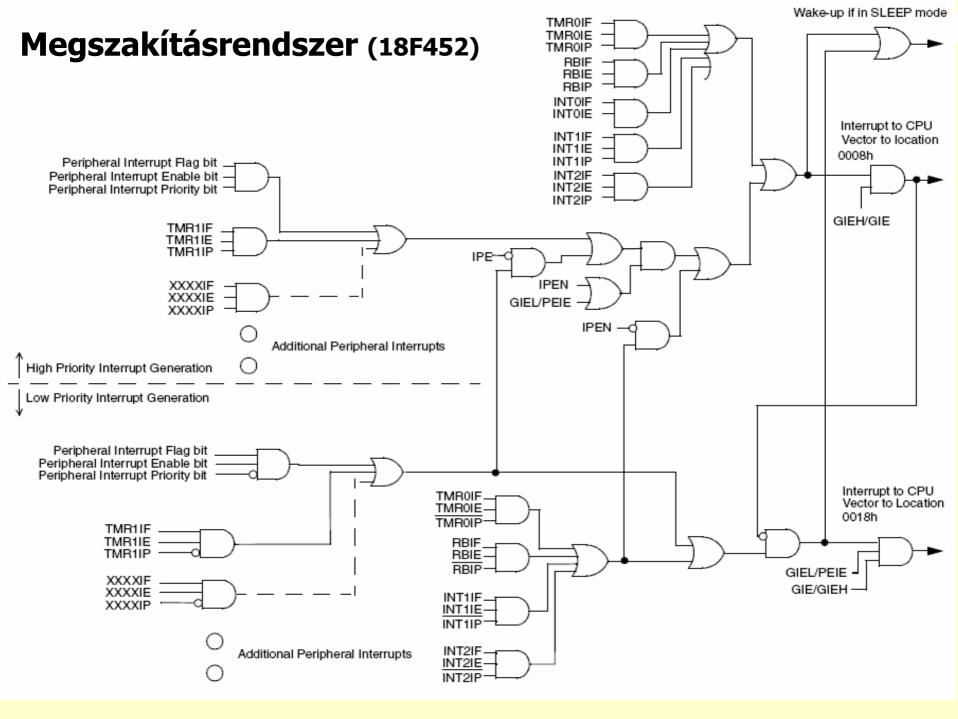
- Két megszakításvektor:
  - Magas prioritású: 0008h
  - Alacsony prioritású: 0018h
- A megszakítások egyedileg és globálisan engedélyezhetőek, egyedileg állítható a prioritásuk
- Minden megszakításhoz tartozik:
  - xxxxIF: megszakítást jelző bit (flag)
  - xxxxIE: egyedi megszakítás engedélyező bit
  - xxxxIP: prioritás beállító bit
- · Globális állítási lehetőségek:
  - IPEN: prioritásos rendszer engedélyezése
  - GIEH: magas szintű megszakítások globális engedélyezése (GIE)
  - · GIEL: alacsony szintű megszakítások globális engedélyezése

# Megszakítások

- A megszakítás forrásának azonosítását és a prioritás beállítását szoftverből kell elvégezni
- A megszakításjelző bitek törlése (bittől függően):
  - Az adott bit szoftveres törlésével
  - A kiváltó ok megszüntetésével (pl. vételi regiszter kiolvasása)
- Megszakításvektorra ugráskor a PC mentése történik a 31 szó mély verembe. A többi regiszter mentéséről gondoskodni kell!
- Fast Return lehetőség: az 1 mélységű Fast Register Stack-ból az automatikusan mentett W, BSR és a STATUS regiszterek visszatöltése
- A megszakítás megjelenésének késése 3 utasításciklus

# Megszakításrendszer (egyszerűsített vázlat)





## Megszakításokkal kapcsolatos SFR-ek

- INTCONx: globális megszakítás engedélyező bitek, alapvető perifériák megszakításjelző, megszakítás-engedélyező és prioritás beállító bitjei, valamint egyéb beállítások (pl. külső megszakítás élének kiválasztása)
- PIRx: perifériák megszakításjelző bitjei
- PIEx: perifériák egyedi megszakítás engedélyező bitjei
- IPRx: perifériák prioritás beállító bitjei
- RCON: MSB-je a prioritásos megszakításrendszert engedélyezi (IPEN)

PIR1	PSPIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
PIE1	PSPIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
IPR1	PSPIP	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP

# Megszakításrendszer inicializálási példa

Állítsuk be a TMR0 megszakítását magas, az A/D átalakítóét pedig alacsony prioritásúra. Engedélyezzük mindkét megszakítást, és a teljes megszakításrendszert is!

IPEN=1

TMR0IP=1 ADIP=0

(TMR0IF=0) (ADIF=0)

TMR0IE=1 ADIE=1

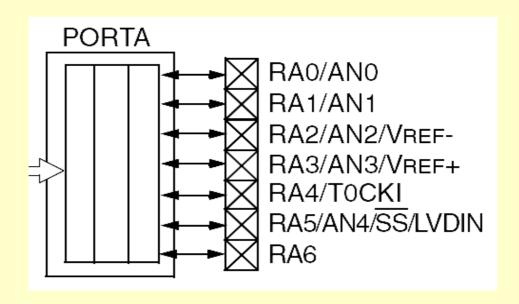
GIEH=1 GIEL=1

### Perifériák (blokkvázlat helyett)

- Legfontosabb perifériák (18F452):
  - Digitális I/O portok (PORTA, PORTB, PORTC, PORTD, PORTE)
  - Időzítők/számlálók (TIMER0, TIMER1, TIMER2, TIMER3)
  - CCP modulok (CCP1, CCP2)
  - Soros kommunikációs modulok (MSSP, USART)
  - A/D átalakító
  - EEPROM

# Digitális portok

- A legtöbb láb más funkciókkal van multiplexelve
- Bitenként állítható irány (I/O)
- 3 állapotú láb megvalósításának lehetősége
- Lábanként 25mA forrás/nyelő (Összesen max. 250mA!)



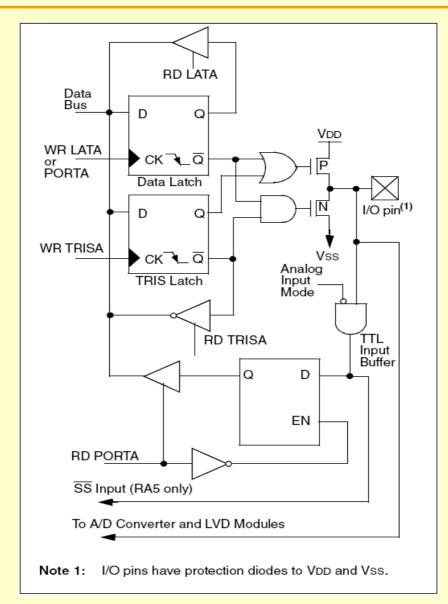
# Digitális portok

- A portokhoz tartozó regiszterek:
  - TRISx: irány beállítása (adott biten: 1-bemenet, 0-kimenet)
  - PORTx: portlábaknak megfelelő bitek (főként olvasásra)
  - LATx: portlábakhoz tartozó kimeneti tárolók bitjei (LATn≠PORTn multiplexelt funkciók engedélyezésekor, vagy ha a portláb bemenet)
- Multiplexelt funkciókhoz tartozó regiszterek (változó)

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTA	_	RA6	RA5	RA4	RA3	RA2	RA1	RA0
LATA	_	LATA Dat	_ATA Data Output Register					
TRISA	_	PORTA D	PORTA Data Direction Register					
ADCON1	ADFM	ADCS2	_	_	PCFG3	PCFG2	PCFG1	PCFG0

# Digitális portok

• Pl.: RA0:3, RA5



## Digitális port inicializálási példa

- Inicializáljuk a PORTA-t a következők szerint:
  - Alsó 4 bit (PORTA<3:0>) bemenet
  - Felső 3 bit (PORTA<6:4>) kimenet
  - A kimenetek 0-ban
  - Multiplexelt funkciók letiltva

PORTA	_	RA6	RA5	RA4	RA3	RA2	RA1	RA0
LATA	_	LATA Dat	ATA Data Output Register					
TRISA	_	PORTA D	PORTA Data Direction Register					
ADCON1	ADFM	ADCS2	_	_	PCFG3	PCFG2	PCFG1	PCFG0

TRISA < 3:0> = 1111

TRISA < 6:4 > = 000

PORTA < 6:4 > = 000

(PCFG<3:0> = 011x, lásd az A/D átalakítónál)

#### Időzítők

#### TIMER0:

- 8/16 bites időzítő/számláló
- 8 bites programozható előosztó
- Belső/külső órajel-forrás
- Megszakítást ad, ha a számláló regiszter túlcsordul

#### • TIMER1:

- 16 bites időzítő/számláló
- · Belső/külső órajel-forrás (külső kvarc-oszcillátor csatlakoztatható)
- Megszakítást ad, ha a számláló regiszter túlcsordul
- (Kapcsolatban van a CCP modulokkal)

#### Időzítők

#### TIMER2:

- 8 bites időzítő/számláló
- 4 bites programozható előosztó és utóosztó
- Belső órajel-forrás
- Megszakítást ad, ha a számláló regiszter túlcsordul
- (Kapcsolatban van a CCP modulokkal)

#### • TIMER3:

- 16 bites időzítő/számláló
- 3 bites előosztó
- Belső/külső órajel-forrás (külső kvarc-oszcillátor csatlakoztatható)
- Megszakítást ad, ha a számláló regiszter túlcsordul
- (Kapcsolatban van a CCP modulokkal)

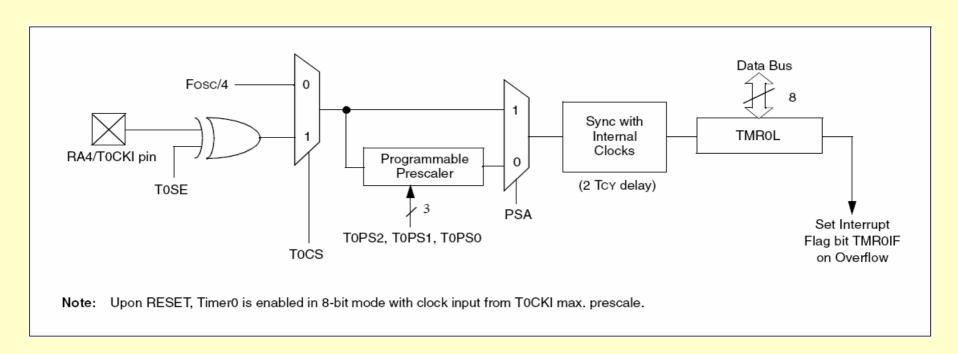
### **TIMERO**

TIMER0-val kapcsolatos regiszterek:

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
TMR0L	Timer0 Modu	Module Low Byte Register							
TMR0H	Timer0 Modu	ule High Byte f	Register						
INTCON	GIE/GIEH	PEIE/GIEL	PEIE/GIEL TMR0IE INT0IE RBIE TMR0IF INT0IF RBIF						
T0CON	TMR0ON	T08BIT	T08BIT T0CS T0SE PSA T0PS2 T0PS1 T0PS0						
TRISA	_	PORTA Data Direction Register							

#### **TIMERO**

TIMERO 8 bites üzemmódban:

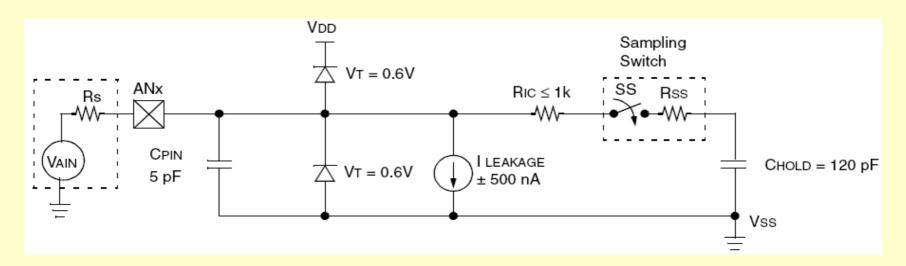


## TIMERO inicializálási példa

- Inicializáljuk TIMER0-t a következők szerint:
  - 8 bites üzemmód
  - Belső órajel-forrás
  - Előosztó kikapcsolva
  - TMR0 kezdeti értéke: 55h

```
T08BIT = 1
T0CS = 0
PSA = 1
TMR0L = 55h
(TMR0ON = 1)
```

- 1 db A/D SAR átalakító
- 10 bites felbontás
- 8 A/D bemenet (18F452)
- Mintavevő/tartó (T<sub>ACO</sub>, forrás impedancia!)
- Választható A/D órajel (T<sub>AD</sub>, órajel-frekvencia!)
- Megszakítást ad, ha kész az átalakítás

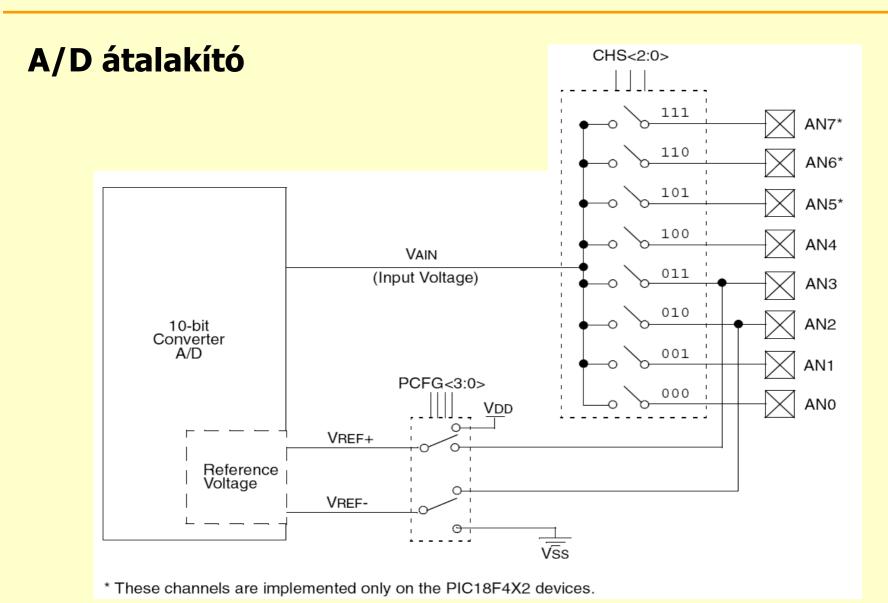


#### Az A/D átalakítóval kapcsolatos regiszterek:

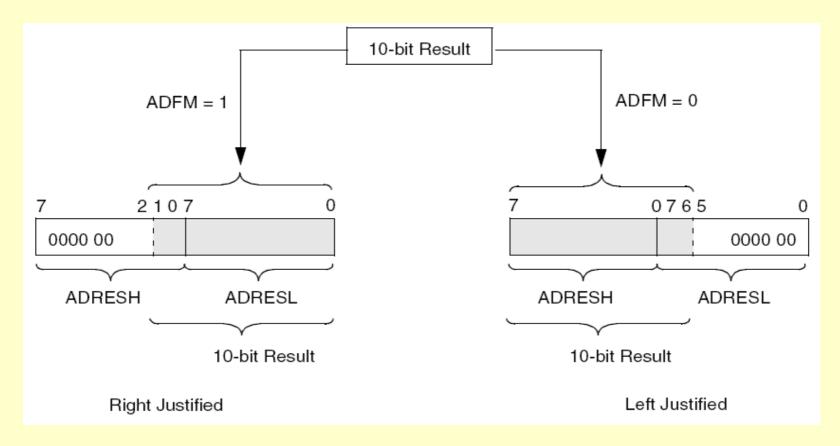
Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on All Other RESETS
INTCON	GIE/ GIEH	PEIE/ GIEL	TMR0IE	INTOIE	RBIE	TMR0IF	INTOIF	RBIF	0000 000x	0000 000u
PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
IPR1	PSPIP <sup>(1)</sup>	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	0000 0000	0000 0000
PIR2	_	_	_	EEIF	BCLIF	LVDIF	TMR3IF	CCP2IF	0 0000	0 0000
PIE2	_	_		EEIE	BCLIE	LVDIE	TMR3IE	CCP2IE	0 0000	0 0000
IPR2	_	_		EEIP	BCLIP	LVDIP	TMR3IP	CCP2IP	1 1111	1 0000
ADRESH	A/D Resul	t Register			-				xxxx xxxx	uuuu uuuu
ADRESL	A/D Resul	t Register							xxxx xxxx	uuuu uuuu
ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	_	ADON	0000 00-0	0000 00-0
ADCON1	ADFM	ADCS2	_	_	PCFG3	PCFG2	PCFG1	PCFG0	000	000
PORTA	_	RA6	RA5	RA4	RA3	RA2	RA1	RA0	0x 0000	0u 0000
TRISA	_	PORTA Data Direction Register							11 1111	11 1111
PORTE				_	_	RE2	RE1	RE0	000	000
LATE	_	_	_	_	_	LATE2	LATE1	LATE0	xxx	uuu
TRISE	IBF	OBF	IBOV	PSPMODE	_	PORTE Data	a Direction I	oits	0000 -111	0000 -111

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used for A/D conversion.

Note 1: The PSPIF, PSPIE and PSPIP bits are reserved on the PIC18F2X2 devices; always maintain these bits clear.



Az A/D 10 bites eredményének igazítása



 A multiplexelt analóg bemenetek konfigurációs lehetőségei (részletek a teljes táblázatból)

PCFG <3:0>	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	VREF+	VREF-	C/R
0000	Α	Α	Α	Α	Α	Α	Α	Α	VDD	Vss	8 / 0
0001	Α	Α	Α	Α	VREF+	Α	Α	Α	AN3	Vss	7 / 1
011x	D	D	D	D	D	D	D	D	_	_	0/0
1110	D	D	D	D	D	D	D	Α	VDD	Vss	1/0

Az analóg bemenetként használt lábak TRIS bitjeinek 1-ben kell lenniük!

#### Az A/D órajelének beállítása

ADCON1 <adcs2></adcs2>	ADCON0 <adcs1:adcs0></adcs1:adcs0>	Clock Conversion			
0	0.0	Fosc/2			
0	01	Fosc/8			
0	10	Fosc/32			
0	11	FRC (clock derived from the internal A/D RC oscillator)			
1	00	Fosc/4			
1	01	Fosc/16			
1	10	Fosc/64			
1	11	FRC (clock derived from the internal A/D RC oscillator)			

AD Clock	Source (TAD)	Maximum Device Frequency				
Operation	ADCS2:ADCS0	PIC18FXX2	PIC18LFXX2			
2 Tosc	000	1.25 MHz	666 kHz			
4 Tosc	100	2.50 MHz	1.33 MHz			
8 Toec	0.01	5 00 MHz	2 67 MHz			

## A/D átalakító inicializálási példa

- Konfiguráljuk az A/D-t a következők szerint:
  - ANO/RAO analóg bemenet
  - Referencia a tápfeszültség
  - Órajel: f<sub>osc</sub>/4
  - Eredmény balra igazítva (8 bites kompatibilis mód)

```
(ADON = 1)
TRISA0 = 1
PCFG<3:0> = 1110
CHS<2:0> = 000
ADCS<2:0> = 100
ADFM = 0
(GO/DONE = 1)
```

## **CCP** modulok

- 2 db CCP modul (CCP1, CCP2)
- 16 bites Capture/Compare/PWM üzemmód

## **Capture üzemmód**

- Esemény hatására eltárolódik TMR1 vagy TMR3 16 bites értéke, és megszakítás generálódik
- · Négyféle esemény lehetséges (a modul bemenetén):
  - Le/felfutó él
  - Minden negyedik/tizenhatodik felfutó él

## **Compare üzemmód**

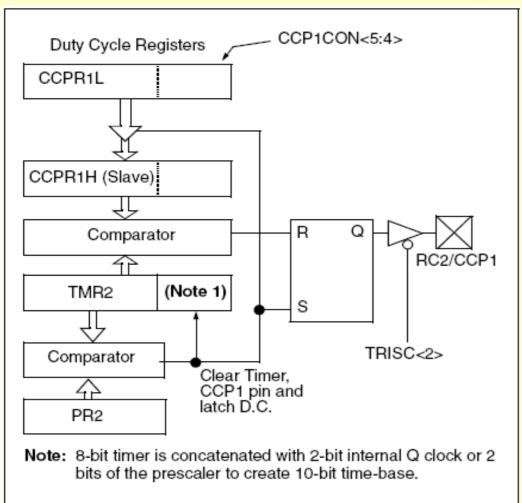
- Konstans érték összehasonlítása TMR1 vagy TMR3 16 bites értékével
- Egyezés esetén megszakítás generálódik, valamint:
  - A modul kimenete magas/alacsony szintű lesz
  - A modul kimenetén pozitív/negatív impulzust ad
  - Nem történik változás (a megszakítás kivételével)

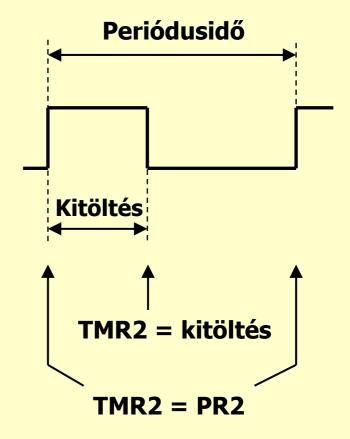
## **CCP** modulok

#### **PWM üzemmód**

- 10 bit felbontású PWM jel generálás
- · Állítható PWM frekvencia (periódusidő) és kitöltés
- A CCP modulokkal kapcsolatos regiszterek:
  - TMR2 regiszterei és TMR2CON
  - PR2, a TMR2-höz kötődő periódus regiszter
  - CCPRxL, CCPRxH CCP modulok regiszterei
  - CCPxCON CCP modulok vezérlőregiszterei
  - Az ide tartozó megszakításokat jelző és maszkoló regiszterek

# A CCP1 modul PWM üzemmódja





# A CCP1 modul PWM üzemmódja

A PWM periódusidő beállítása:

PWMperiod = [PR2+1]\*4\*T<sub>osc</sub>\*TMR2prescale\_value

A PWM kitöltési idő beállítása:

PWMduty\_cycle = (CCPR1L:CCPCON<5:4>)\*T<sub>OSC</sub>\*TMR2prescale\_value

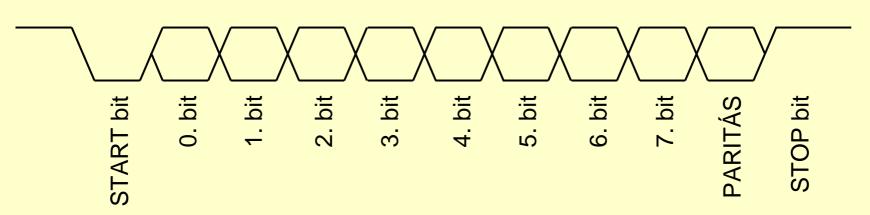
A PWM jel felbontása (állítási lépcsője) függ a  $f_{OSC}/f_{PWM}$  hányadostól. Ha a hányados csökken, a felbontás is csökken!

További részletek: lásd katalógus.

#### **USART**

- Univerzális szinkron/aszinkron adó/vevő
- Aszinkron üzemmódban full duplex
- Szinkron üzemmódban half duplex
- Hardveres bitsebesség (baud rate) generátor
- Kettős pufferelés
- Paritás bit átvitel lehetősége (szoftveres kiszámítás)
- Külön adás és vételi megszakítás

#### Az aszinkron soros jelátvitel:

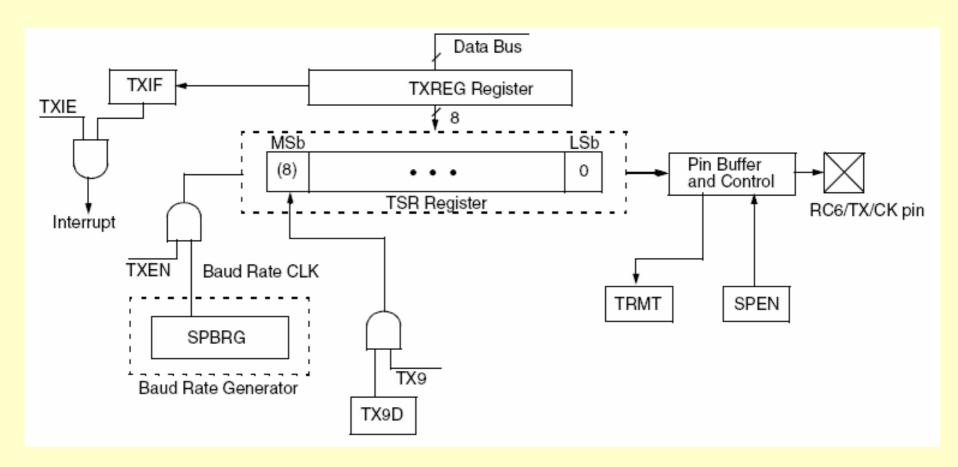


#### **USART**

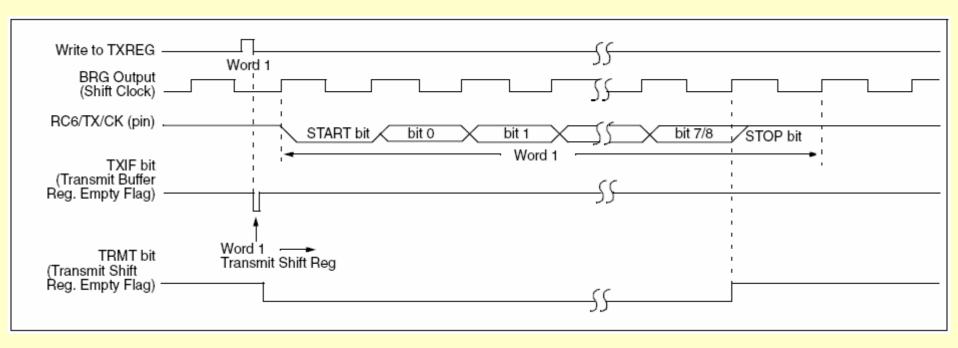
#### Az USART-tal kapcsolatos regiszterek:

- TXSTA: adás vezérlő és státuszjelző bitek
- RCSTA: vétel vezérlő és státuszjelző bitek
- SPBRG: bitsebesség generátor beállító regisztere
- TXREG: adás puffer
- RCREG: vételi puffer
- Ide vonatkozó megszakításokkal kapcsolatos regiszterek

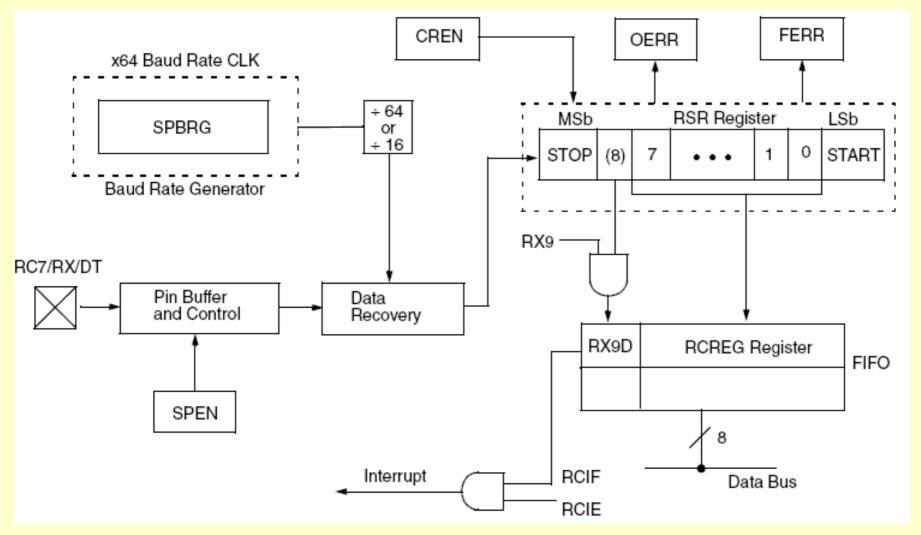
## **USART** aszinkron adás



## **USART** aszinkron adás

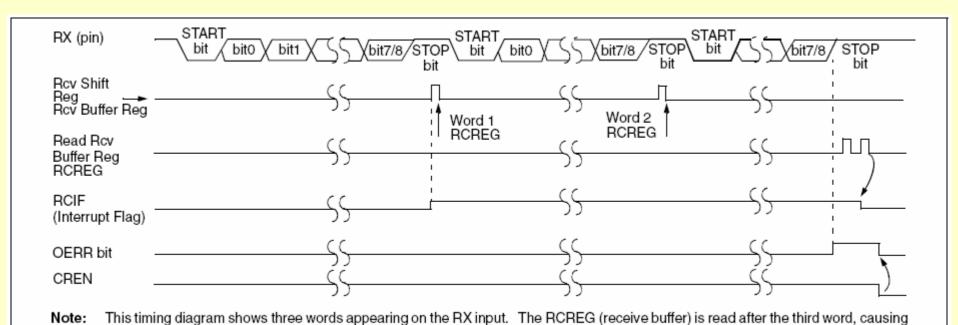


## **USART** aszinkron vétel



## **USART** aszinkron vétel

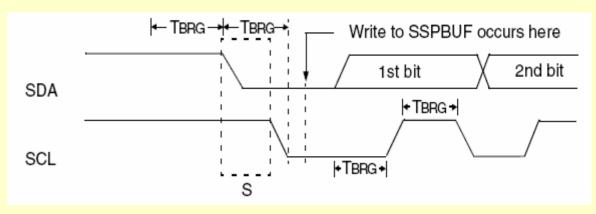
the OERR (overrun) bit to be set.



BMF KVK MAI, Molnár Zsolt, 2007.

#### **MSSP**

- Master szinkron soros port
- I<sup>2</sup>C master, multi-master, slave üzemmódok
- SPI master, slave üzemmódok (négyféle órajel-üzemmód)
- Az I<sup>2</sup>C kapcsolat jelei:
  - SCL: órajel
  - SDA: soros adat ki/bemenet
- Cím + adat átvitele történik
- START/STOP "feltétel"
- ACKnowledge

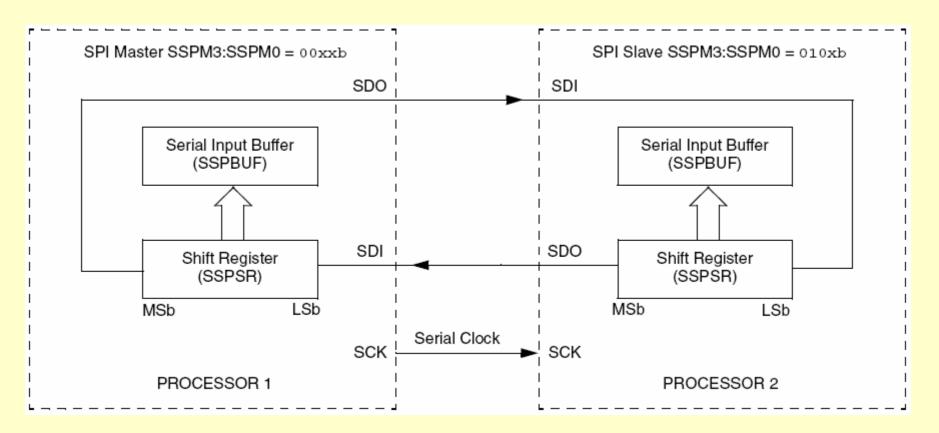


#### **MSSP**

- Az SPI kapcsolat jelei:
  - SCK: órajel
  - SDI: soros adat bemenet
  - SDO: soros adat kimenet
  - SS: kiválasztás (slave módban)
- Adatátviteli lehetőségek:
  - Master adatot küld Slave haszontalan adatot küld
  - Master adatot küld Slave adatot küld
  - Master haszontalan adatot küld Slave adatot küld

## **MSSP**

#### **SPI** kapcsolat:

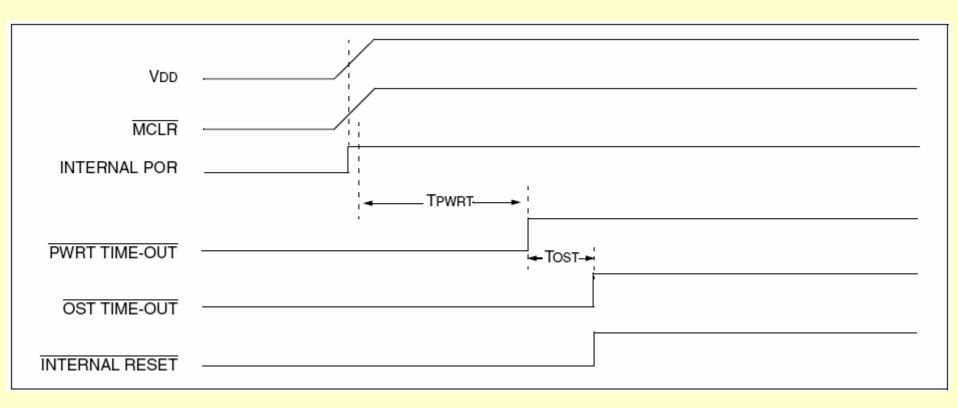


# **Órajel-forrás típusok**

- Az órajel típusa konfigurációs bitekkel állítható (a beprogramozáskor)
- Kiválasztható órajel-források:
  - LP: kvarckristály 200kHz-ig (alacsony áramfelvételű üzemmód)
  - XT: kvarckristály 4MHz-ig
  - HS: kvarckristály 20MHz-ig
  - HS-PLL: HS kvarc, órajel 4x szorzása PLL-el (40MHz-ig)
  - RC: külső RC oszcillátor
  - RCIO: külső RC oszcillátor, az órajel portlábon kiadva
  - EC: külső órajel-forrás csatlakoztatása
  - · ECIO: külső órajel-forrás csatlakoztatása, az órajel portlábon kiadva

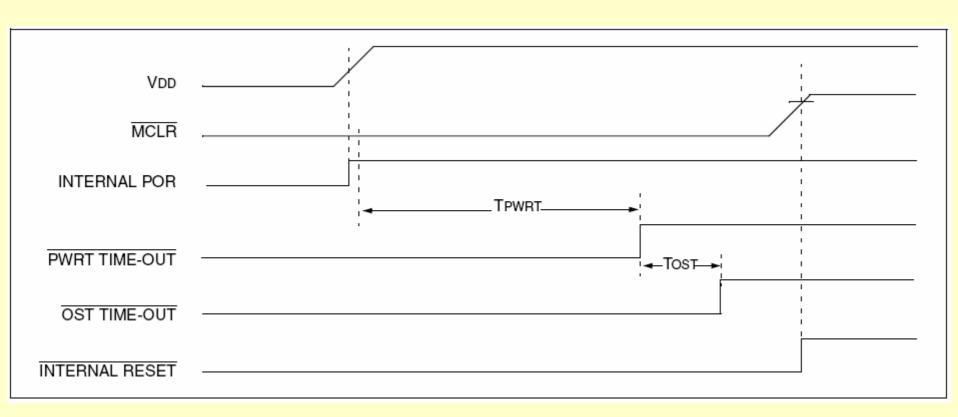
## Reset

## 1. MCLR láb a tápfeszültségre kötve



## Reset

## 2. MCLR láb nagy időállandójú RC tagra kötve



## Watchdog

- Feladata: ha a beállított időn belül nem nullázzuk, akkor túlcsordul, és újraindítja a processzort
- Előre nem látható, rejtett szoftver hibák jelentkezése esetén, vagy zavarjelek bejutása esetén lép működésbe, amikor a rendszeres nullázás elmarad
- Képes éleszteni a processzort energiatakarékos állapotból
- Saját, belső RC oszcillátor
- Konfigurációs bitekkel állítható be az üzemmódja