Hardver alapok

(VIIIBA01)

Időzítők gyakorlati alkalmazása

Rácz György gyuriracz@iit.bme.hu

2022/23. tanév őszi félév

Bevezetés

- Szoftveres időzítést már láttuk
- Hátrányai:
  - 100% CPU lefoglaltságot eredményez
  - Időzítése függ a processzor órajelétől
  - Ha van engedélyezve hardveres megszakítás, ami többször is lefut, akkor az időzítés pontatlanná válik
- Célszerű lenne külön hardveregységet alkalmazni a pontos időzítésekhez→ TIMER

Kitekintés - Megszakítás

- Láttuk a szoftveres várakozó hurok és szoftveres késleltetés megvalósítását
- Mivel ez 100%-ban lefoglalja a CPU-t feleslegesen, így a CPU nem tud más tevékenységet végezni
- Megoldás: „automatikus szubrutin hívás”
- A gomb megnyomása, vagy az időzítés letelte indítson el egy szubrutint

  →megszakítási szubrutin

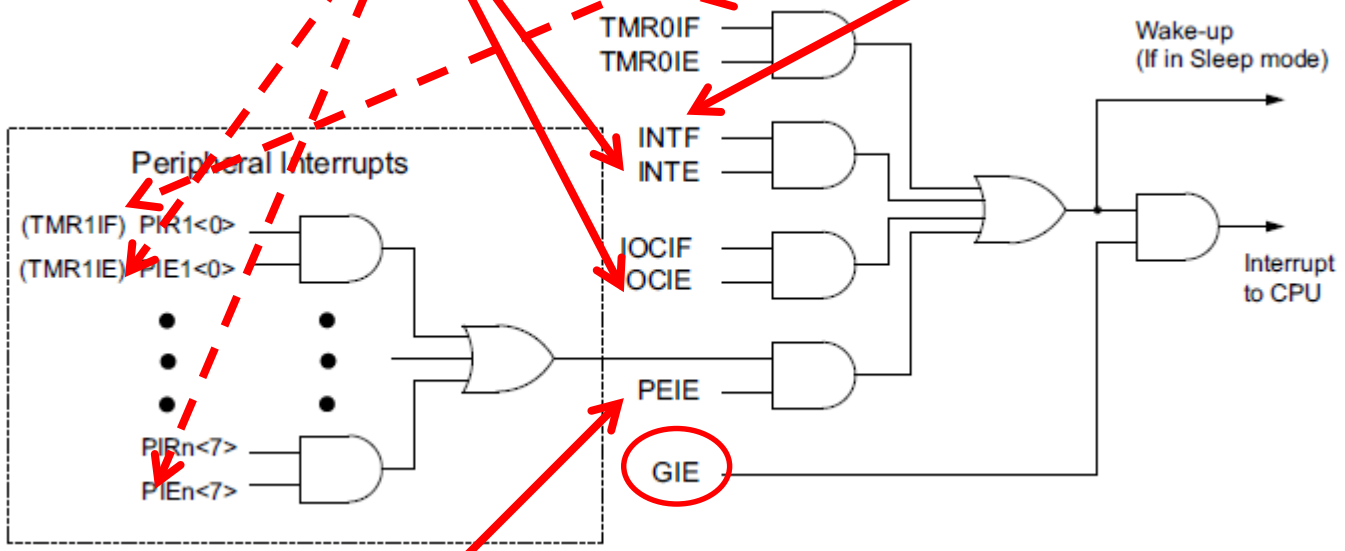
PIC16F és a megszakítás

- Sok megszakítási forrás lehet (szinte minden periféria)
- Minden forráshoz külön engedélyezés (IE) és külön jelzőbit (IF)
- Egyetlen vektor →fixen 0x0004 cím a belépési pont
- Programból kell az okát kitalálni (melyik engedélyezett Interrupt Flag (IF) aktív)
- Belépéskor környezet mentése, visszatéréskor pedig a környezet visszaállítása→később még foglalkozunk vele…

Megszakítás logika

Minden forrás egyedileg engedélyezhető

Minden forrásnak van egy jelzőbitje (IF)

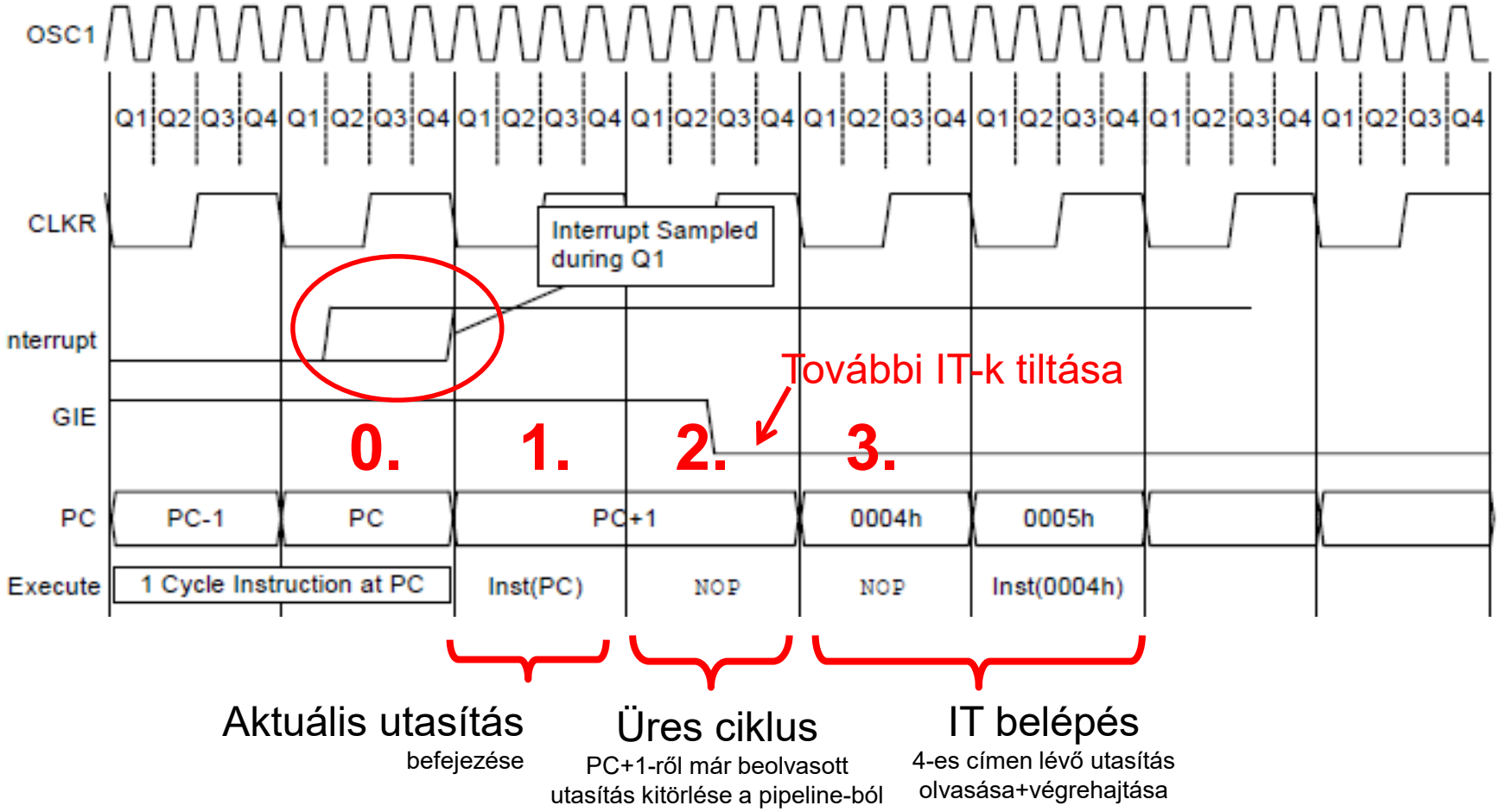


Perifériák külön is tilthatók

Megszakítások engedélyezése

Megszakítás időzítése

1 ciklusú utasítás esetén



2 és 3 ciklusú utasítások esetén a lappangási idő 3..5 utasításciklus (CLKR) lehet

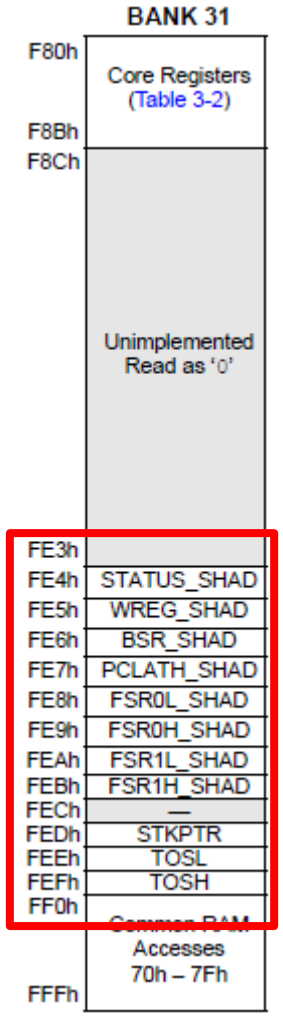
Automatikus környezet mentés

Megszakításba lépéskor automatikusan elmentődnek:

- W regiszter
- STATUS regiszter
- FSR0 és FSR1 regiszterek
- BSR regiszter
- PCLATH regiszter

Visszatéréskor az elmentett értékek mindig helyreállításra kerülnek (RETFIE utasításra)

Minden más „rombolt” változó mentését a programozónak kell megoldania!



Időzítők felhasználása

- Számláló áramkörök különböző kiegészítőkkel
- 8 vagy 16 bites
- Ciklus rövidítés, jelzőbit az időzítés végén

  →megszakítást is tud okozni
- Használható:
  - Adott frekvencia előállítása
  - Változtatható kitöltési tényező előállítása (PWM)
  - Bemeneti váltás időpontjának megmérése
  - Frekvencia mérés
  - Periódus idő mérés

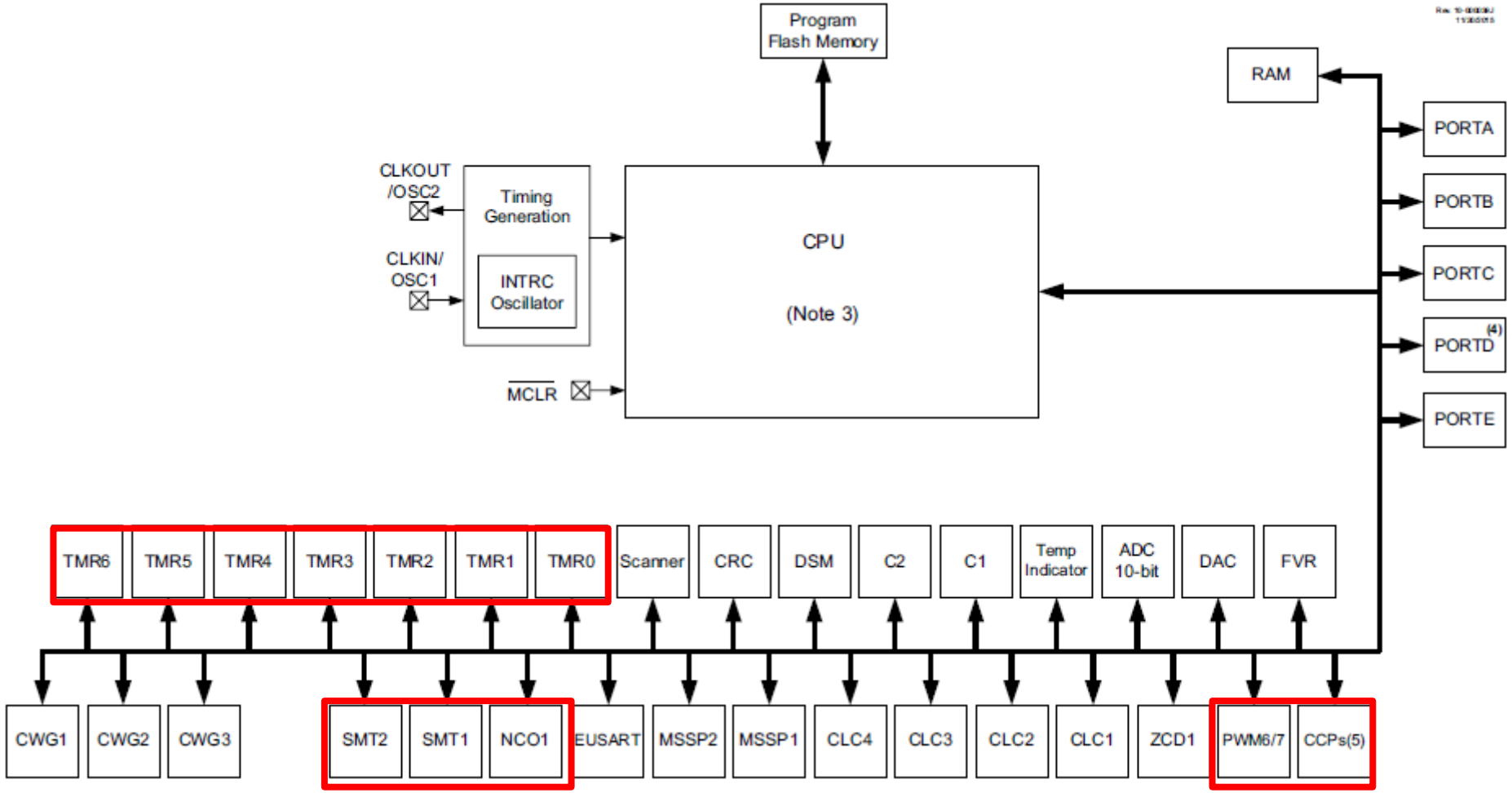
Hardveres időzítők a PIC16F18875-ben

- Watchdog
- 3db 8 bites timer (TMR2, TMR4, TMR6)
- 4db 16 bites timer (TMR0, TMR1, TMR3, TMR5)
- 2db SMT (Signal Measurement Timer, 24 bites)

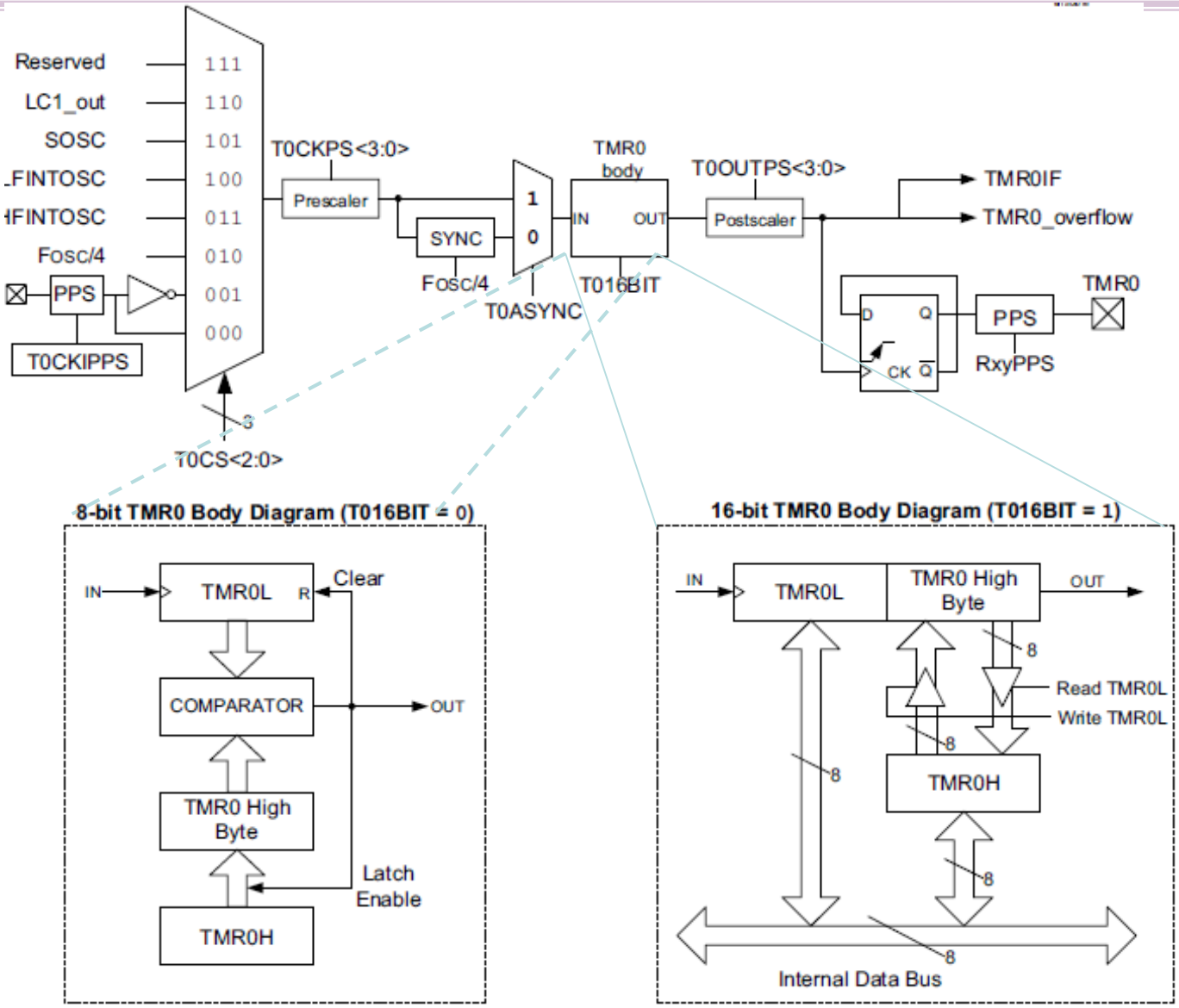
Időzítőt használó perifériák:

- 5db CCP modul (16 bites mérés, 10 bites PWM)
- 2db 10 bites PWM (Pulse Width Modulator) modul
- NCO (Numerically Controlled Oscillator)

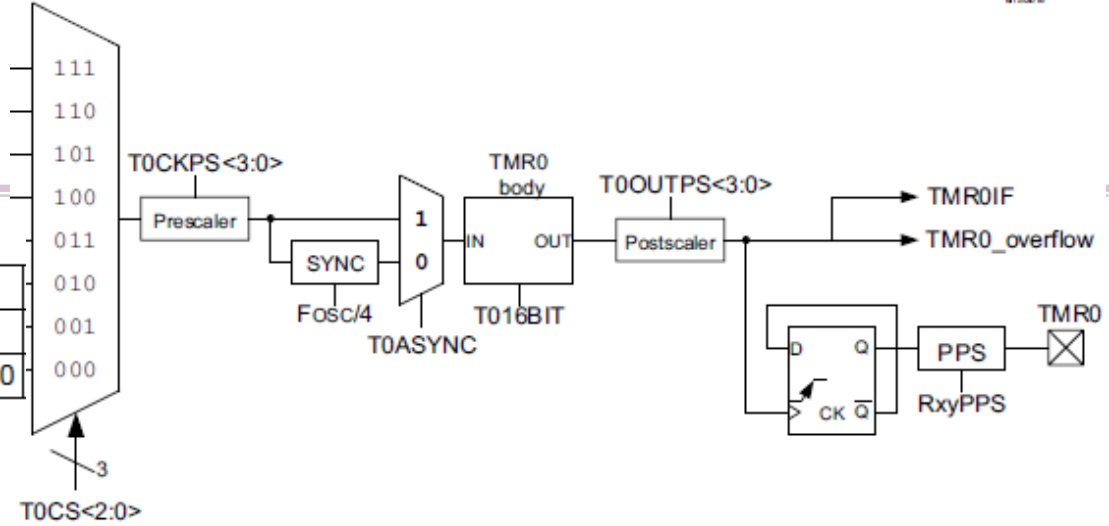
Hardveres időzítők a PIC16F18875-ben



TMR0


Reserved
LC1_out
SOSC
LFINTOSC
HFINTOSC
Fosc/4
PPS
T0CKIPPS
111
110
101
100
011
010
001
000
T0CKPS<3:0>
Prescaler
SYNC
Fosc/4
T0ASYNC
1
0
TMR0 body
IN
OUT
T016BIT
T0OUTPS<3:0>
Postscaler
TMR0IF
TMR0_overflow
D
Q
CK
Q
PPS
RxyPPS
TMR0
T0CS<2:0>
8-bit TMR0 Body Diagram (T016BIT = 0)
IN
TMR0L
R
Clear
COMPARATOR
OUT
TMR0 High Byte
Latch Enable
TMR0H
16-bit TMR0 Body Diagram (T016BIT = 1)
IN
TMR0L
TMR0 High Byte
OUT
Read TMR0L
Write TMR0L
TMR0H
Internal Data Bus

TMR0



REGISTER 27-1: T0CON0: TIMER0 CONTROL REGISTER 0

R/W-0/0	U-0	R-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
T0EN	—	T0OUT	T016BIT	T0OUTPS<3:0>			
bit 7							bit 0

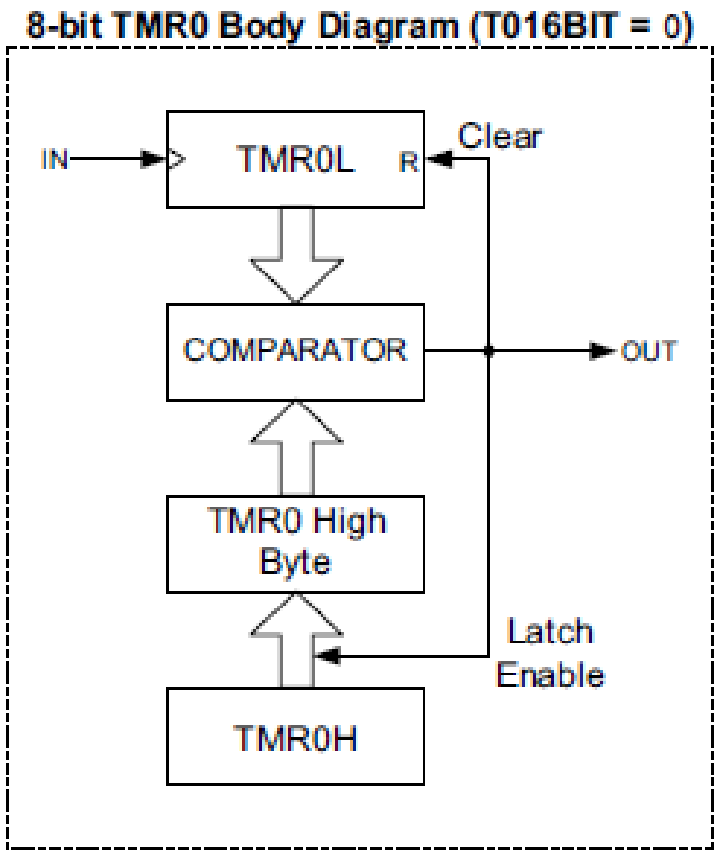
- Választható 8 vagy 16 bites működés
- Programozható utóosztó (output postscaler) 1:1…1:16 között

REGISTER 27-2: T0CON1: TIMER0 CONTROL REGISTER 1

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
T0CS<2:0>			T0ASYNC	T0CKPS<3:0>			
bit 7							bit 0

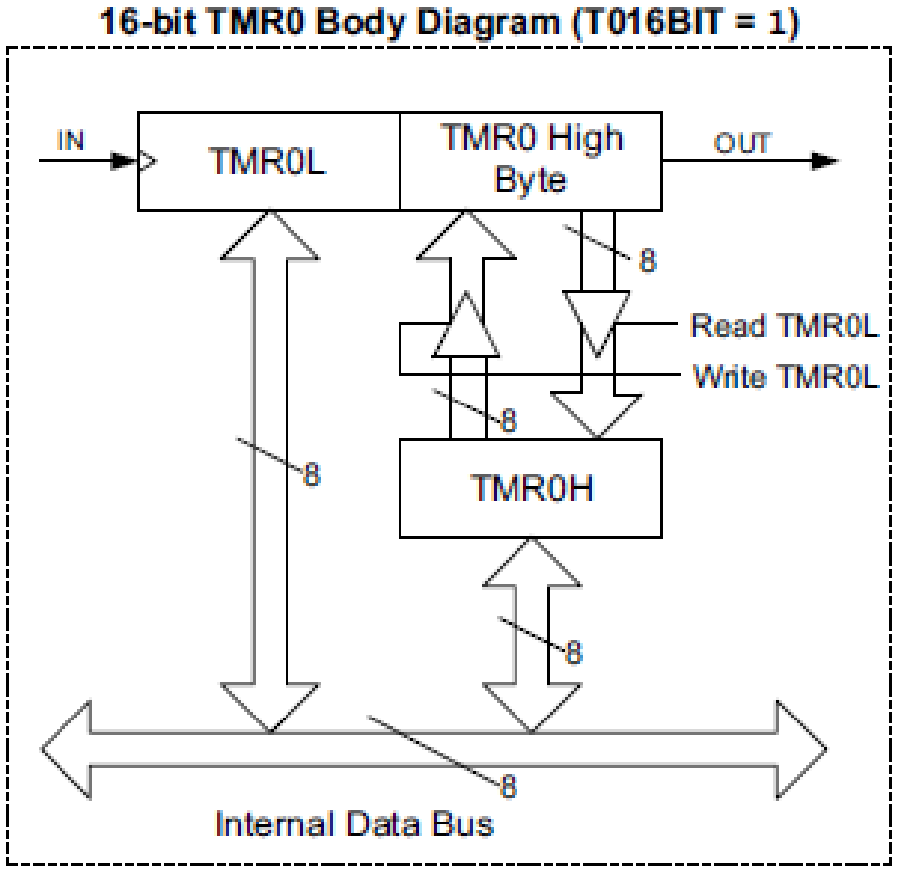
- Választható órajel forrás
- Működése szinkronizálható az utasítások végrehajtásához
- Programozható előosztó (prescaler) 1:1…1:32768
- Túlcsordulás kimenet portlábra is kivezethető

TMR0-automatikus újratöltés mód



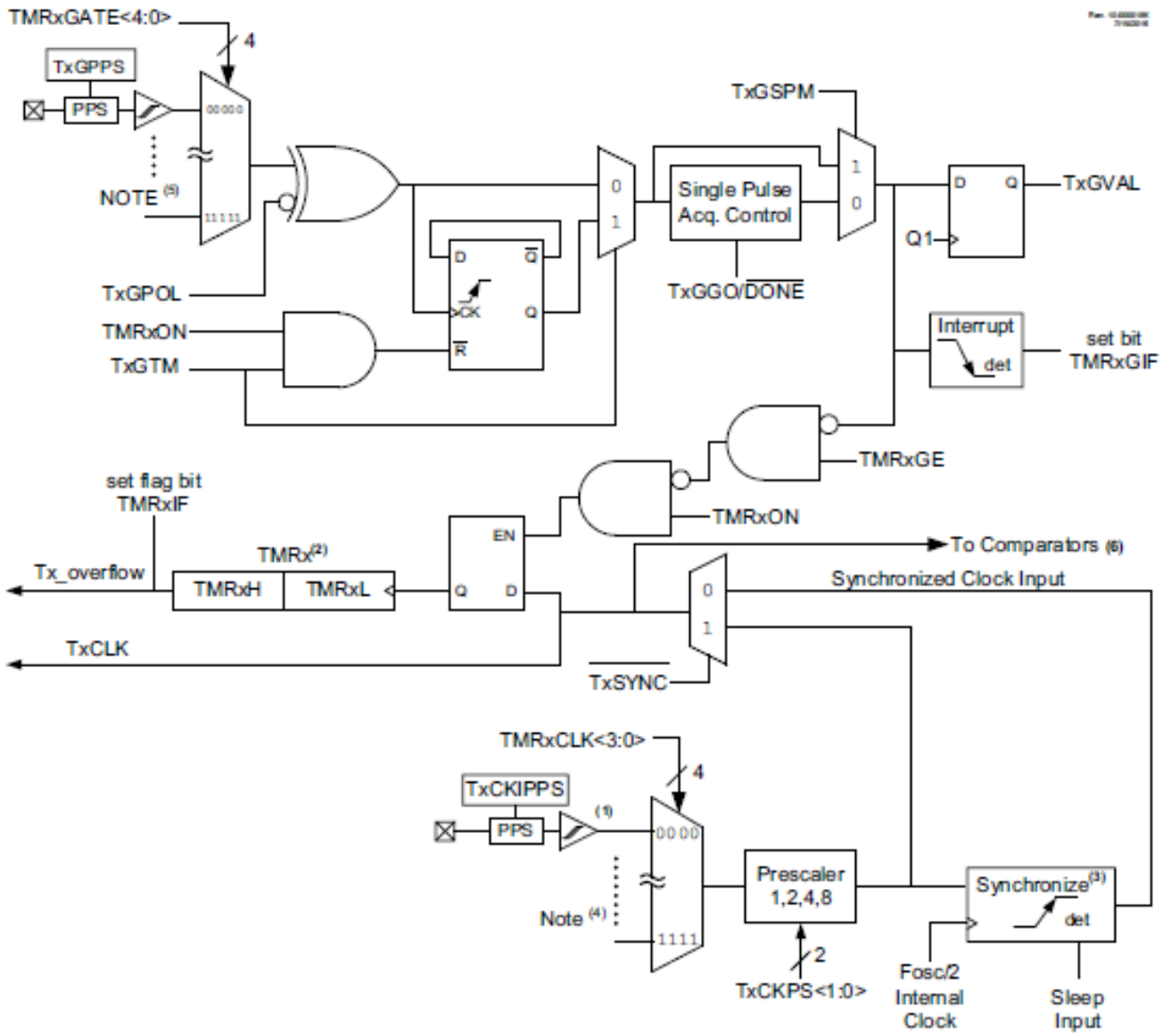
- Felfelé számlálás
- Beállított érték elérésekor törlés+kimenet állítása
    - TMR0L számol
    - TMR0H-ban van a beállított érték
- Alkalmazási példák:
    - Órajel előállítás külső eszköznek
    - Program ütemezés
    - Jelzőhang generálás

TMR0 16 bites számláló mód



- Felfelé számlálás
- 16 biten (2db 8 bites regiszter)
  - TMR0H:TMR0L együtt egy 16 bites számláló
- Túlcsordulásnál kimenet beállítása
- Alkalmazási példák:
  - Külső események számolása
  - Idő/frekvencia mérés

TMR1, 3, 5

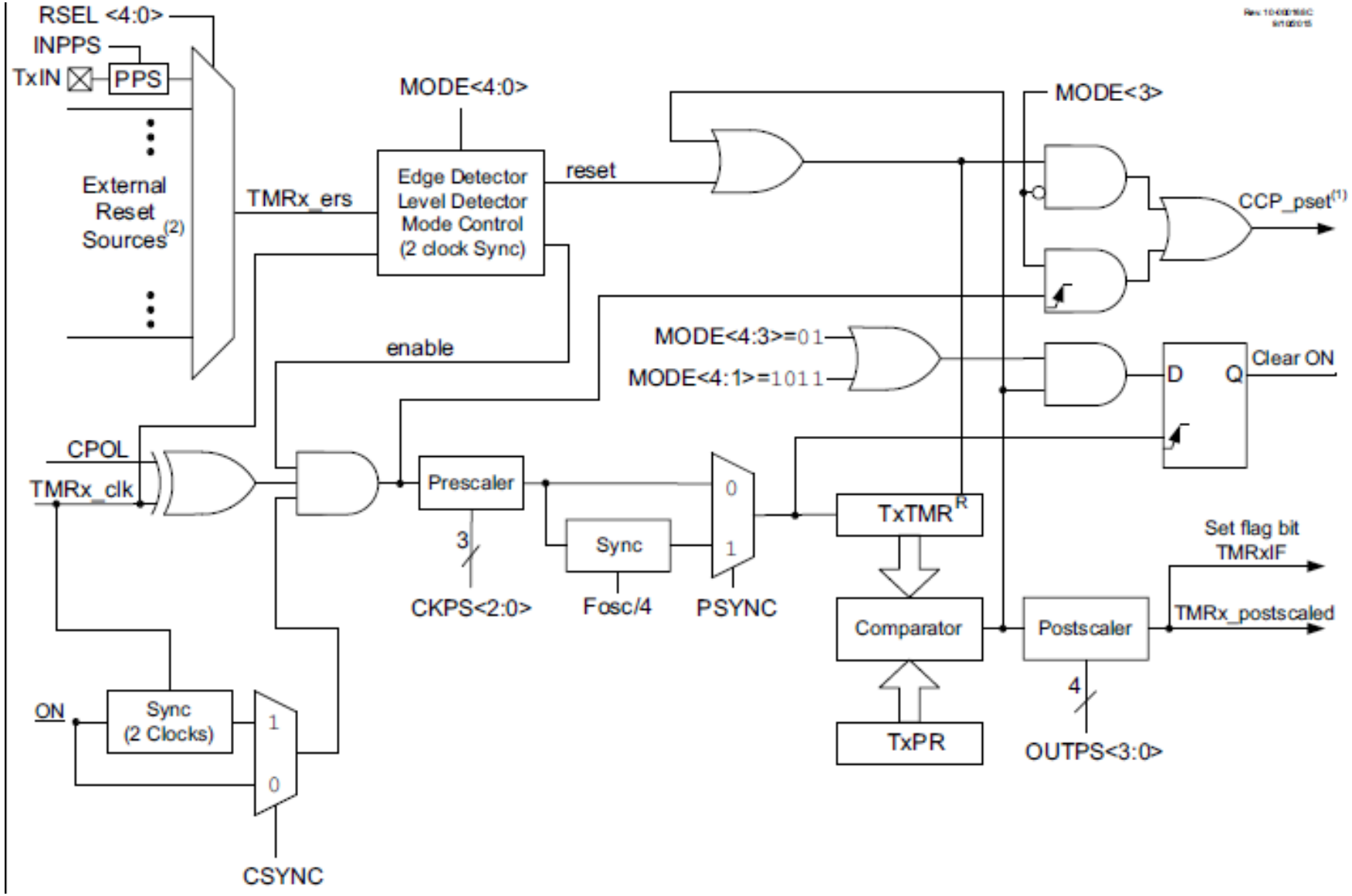

TMRxGATE<4:0>
4
TxGPPS
PPS
NOTE (5)
TxGPOL
TMRxON
TxGTM
D
Q
CK
R
0
1
Single Pulse Acq. Control
TxGSPM
TxGGO/DONE
Q1
TxGVAL
Interrupt
det
set bit TMRxGIF
TMRxGE
TMRxON
set flag bit TMRxIF
Tx_overflow
TMRx(2)
TMRxH
TMRxL
EN
To Comparators (6)
Synchronized Clock Input
TxCLK
TxSYNC
TMRxCLK<3:0>
TxCKIPPS
(1)
Note (4)
Prescaler 1,2,4,8
2
TxCKPS<1:0>
Synchronize(3)
Fosc/2 Internal Clock
Sleep Input

TMR1,3,5 jellemzők

- Felfelé számlálás
- 16 bites időzítő/számláló
- Választható belső vagy külső órajel forrás
- 2 bites előosztó
- Túlcsorduláskor megszakítás kérés
- Választható számlálás engedélyezés (Gate)

© Pilászy György BME-IIT

TMR2, 4, 6


RSEL <4:0>
INPPS
TxIN
PPS
External Reset Sources(2)
TMRx_ers
MODE<4:0>
Edge Detector
Level Detector
Mode Control
(2 clock Sync)
reset
MODE<3>
CCP_pset(1)
enable
MODE<4:3>=01
MODE<4:1>=1011
Clear ON
D
Q
CPOL
TMRx_clk
Prescaler
3
CKPS<2:0>
Sync
Fosc/4
0
1
PSYNC
TxTMR
Set flag bit
TMRxIF
Comparator
Postscaler
TMRx_postscaled
4
OUTPS<3:0>
TxPR
ON
Sync
(2 Clocks)
1
0
CSYNC

TMR2,4,6 jellemzők

- 8 bites számláló
- 8 bites periódus regiszter
- Választható külső törlési lehetőségek
- Programozható előosztó (1:1…1:128)
- Programozható utóosztó (1:1…1:16)
- Választható órajel források
- Megszakítás kérés a periódus végén
- Működési módok:
  - Szabadon futó mód
  - Egyszer lefutó mód

Időzítők gyakorlati alkalmazásai

- Program ütemezés
- Óra
- Órajel előállítás külső egységnek
- Külső jel frekvencia mérése
- Külső jel periódusának mérése
- Külső jel kitöltési tényezőjének mérése
- Külső pulzusok számolása

Program ütemezése TMR0-al

Példa: késleltessünk 10ms-időt a TMR0-al

32MHz-es órajelünk van→8MHz az utasításciklus

$$100Hz = 8000000Hz \cdot \frac{1}{32} \cdot \frac{1}{250} \cdot \frac{1}{10}$$

REGISTER 27-1: T0CON0: TIMER0 CONTROL REGISTER 0

R/W-0/0	U-0	R-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
T0EN	—	T0OUT	T016BIT	T0OUTPS<3:0>			
bit 7							bit 0

REGISTER 27-2: T0CON1: TIMER0 CONTROL REGISTER 1

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
T0CS<2:0>			T0ASYNC	T0CKPS<3:0>			
bit 7							bit 0

Program ütemezése TMR0-al

REGISTER 27-1: T0CON0: TIMER0 CONTROL REGISTER 0

R/W-0/0	U-0	R-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
T0EN	—	T0OUT	T016BIT	T0OUTPS<3:0>			
bit 7							bit 0
1	0	0	0	1	0	0	1

= H'89'

bit 7 **T0EN:** TMR0 Enable bit
1 = The module is enabled and operating
0 = The module is disabled and in the lowest power mode

bit 6 **Unimplemented:** Read as '0'

bit 5 **T0OUT:** TMR0 Output bit (read-only)
TMR0 output bit

bit 4 **T016BIT:** TMR0 Operating as 16-bit Timer Select bit
1 = TMR0 is a 16-bit timer
0 = TMR0 is an 8-bit timer

bit 3-0 **T0OUTPS<3:0>:** TMR0 output postscaler (divider) select bits
1111 = 1:16 Postscaler
1110 = 1:15 Postscaler
1101 = 1:14 Postscaler
1100 = 1:13 Postscaler
1011 = 1:12 Postscaler
1010 = 1:11 Postscaler
1001 = 1:10 Postscaler
1000 = 1:9 Postscaler
0111 = 1:8 Postscaler
0110 = 1:7 Postscaler
0101 = 1:6 Postscaler
0100 = 1:5 Postscaler
0011 = 1:4 Postscaler
0010 = 1:3 Postscaler
0001 = 1:2 Postscaler
0000 = 1:1 Postscaler

$$100Hz = 8000000Hz \cdot \frac{1}{32} \cdot \frac{1}{250} \cdot \frac{1}{10}$$

Program ütemezése TMR0-al

REGISTER 27-2: T0CON1: TIMER0 CONTROL REGISTER 1

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
T0CS<2:0>			T0ASYNC	T0CKPS<3:0>			
bit 7							bit 0
0	1	0	0	0	1	0	1

= H'45'

bit 7-5 **T0CS<2:0>:** Timer0 Clock Source select bits
111 = Reserved
110 = LC1_out
101 = SOSC
100 = LFINTOSC
011 = HFINTOSC
010 = Fosc/4
001 = T0CKIPPS (Inverted)
000 = T0CKIPPS (True)

bit 4 **T0ASYNC:** TMR0 Input Asynchronization Enable bit
1 = The input to the TMR0 counter is not synchronized to system clocks
0 = The input to the TMR0 counter is synchronized to Fosc/4

bit 3-0 **T0CKPS<3:0>:** Prescaler Rate Select bit
1111 = 1:32768
1110 = 1:16384
1101 = 1:8192
1100 = 1:4096
1011 = 1:2048
1010 = 1:1024
1001 = 1:512
1000 = 1:256
0111 = 1:128
0110 = 1:64
0101 = 1:32
0100 = 1:16
0011 = 1:8
0010 = 1:4
0001 = 1:2
0000 = 1:1

$$100Hz = 8000000Hz \cdot \frac{1}{32} \cdot \frac{1}{250} \cdot \frac{1}{10}$$

Program ütemezése TMR0-al

Példa: késleltessünk 10ms-időt a TMR0-al

$$100Hz = 8000000Hz \cdot \frac{1}{32} \cdot \frac{1}{250} \cdot \frac{1}{10}$$

Regiszter név	**Érték (binárisan)**	Érték(Hexa)	**Megjegyzés**
T0CON0	B’1000 1001’	**H’89’**	Bekapcsolás + 1:10 utóosztás
T0CON1	B’0100 0101’	**H’45’**	Fosc/4 + 1:32 előosztás
TMR0H	B’1111 1001’	H’F9’	0…**249** között számol

REGISTER 27-1: T0CON0: TIMER0 CONTROL REGISTER 0

R/W-0/0	U-0	R-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
T0EN	—	T0OUT	T016BIT	T0OUTPS<3:0>			
bit 7 1	0	0	0	1	0	0	1 bit 0

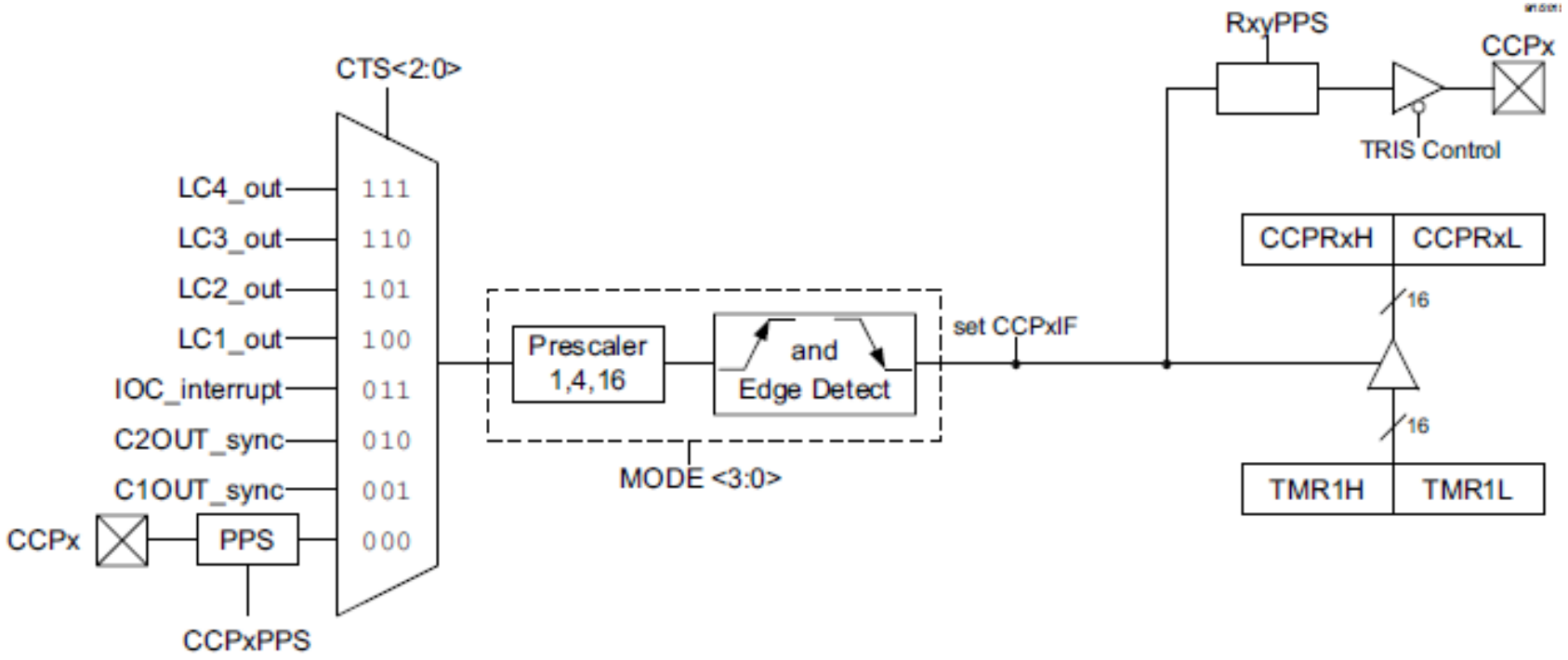
REGISTER 27-2: T0CON1: TIMER0 CONTROL REGISTER 1

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
T0CS<2:0>			T0ASYNC	T0CKPS<3:0>			
bit 7 0	1	0	0	0	1	0	1 bit 0

Várakozás timerrel

```
TMR0INIT:
   MOVLW     B'10001001' ;ez lehetne H'89' is
   MOVWF     T0CON0
   MOVLW     B'01000101' ;ez lehetne H'45' is
   MOVWF     T0CON1
   MOVLW     D'249'
   MOVWF     TMR0H
   RETURN

Delay10ms:
   CLRF      TMR0L
   BANKSEL   PIR0         ; BANK7
   BCF       PIR0,TMR0IF
   BTFSS     PIR0,TMR0IF ; sw várakozás TMR flagbitre
   BRA       $-1   ;menj egy sorral visszább (ha nem akarunk címkét)
   BANKSEL   TMR0L ; BANK0
   RETURN
```

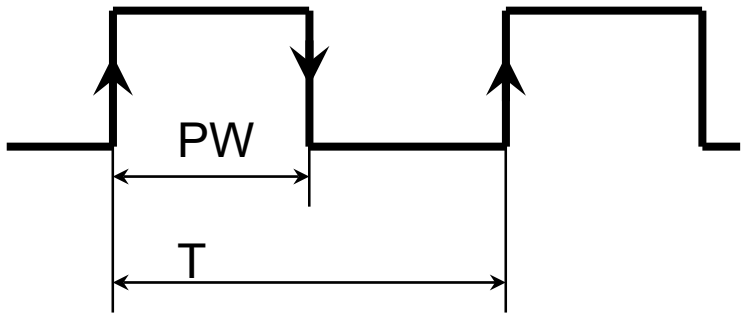
Input capture



Adott (beállítható) él hatására elteszi a számláló pillanatnyi értékét annak leállítása nélkül: CCPRxH:CCPRxL = TMR1H:TMR1L

Input capture alkalmazása

- Impulzus hosszának megmérése
  PW = CAP2↑-CAP1↓



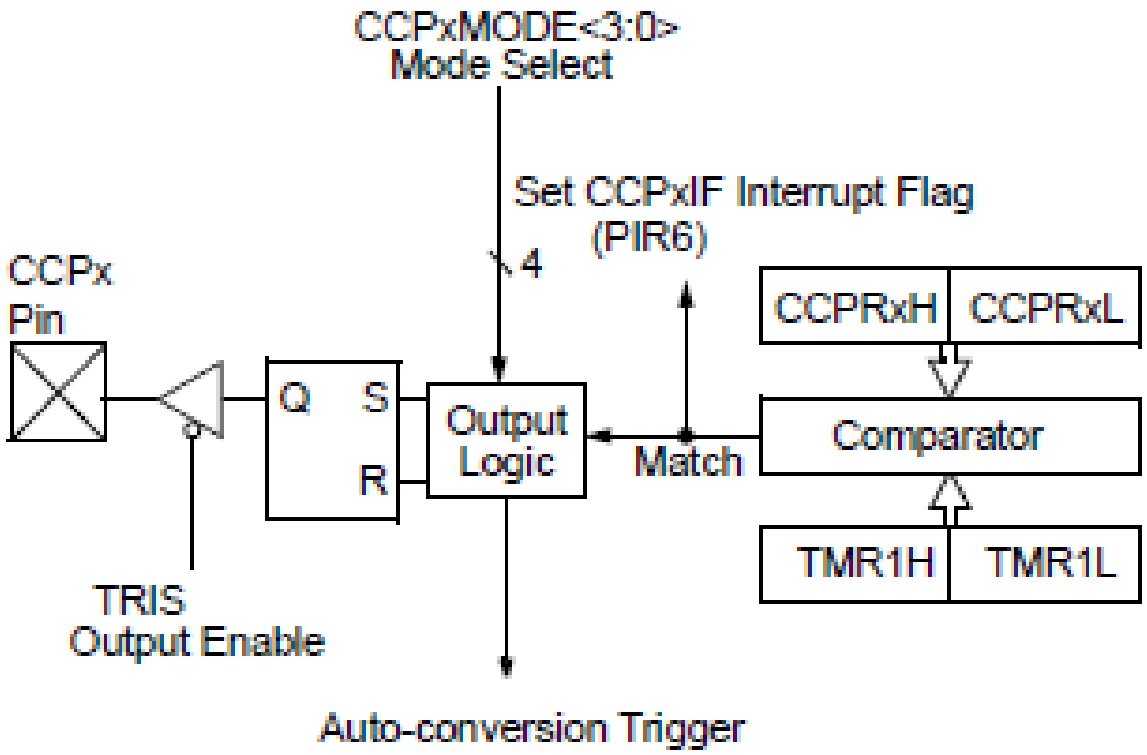
- Jel periódus idejének mérése
  T= CAP2↑-CAP1↑

- Frekvencia mérés N darab minta után:

$$F = \frac{N}{T\ddot{o}sszes} = \frac{darabszám}{\ddot{o}ssz.id\ddot{o}}$$

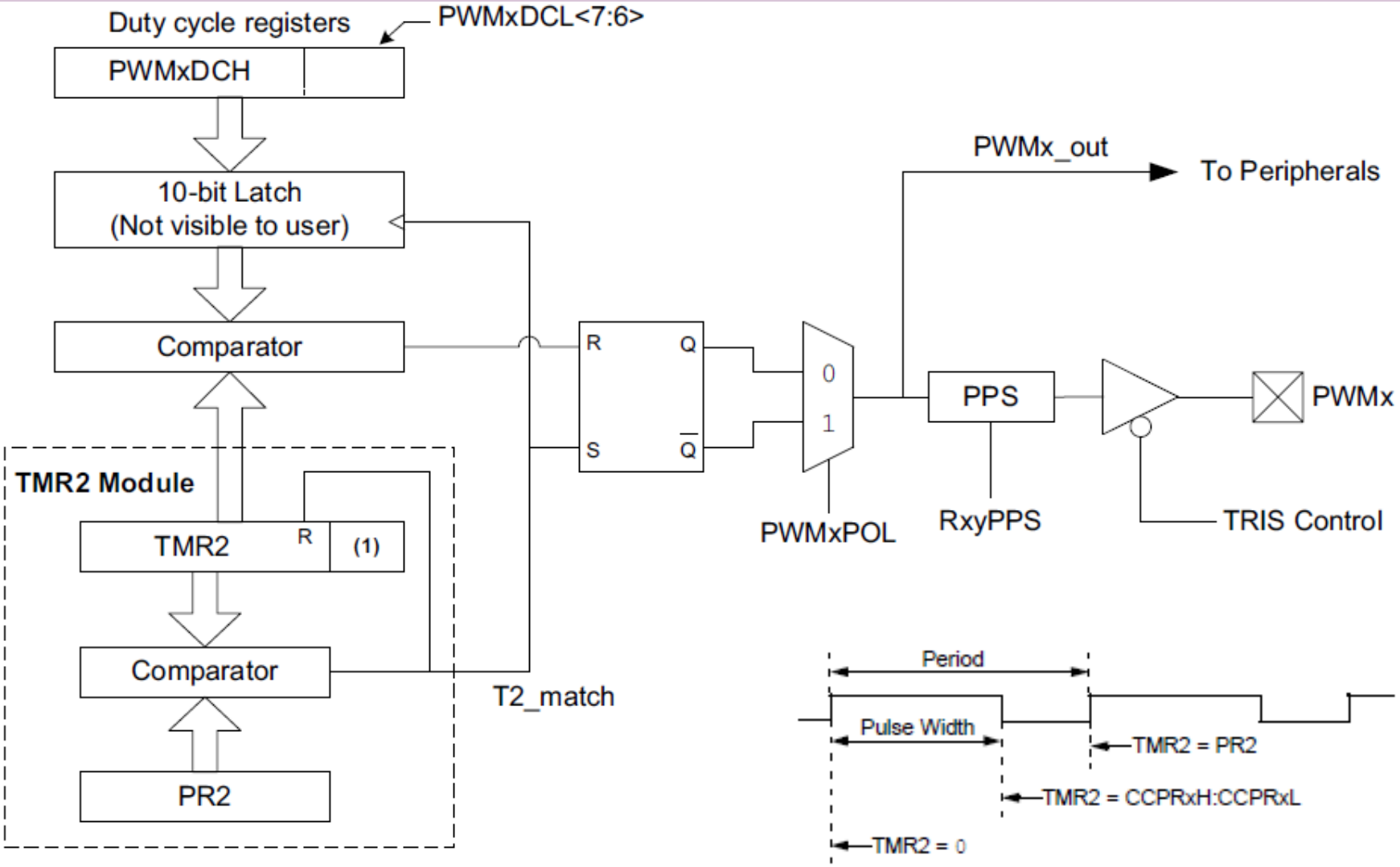
- Időbélyeg előállítása kommunikációhoz

Output compare



- Kimenet beállítása a komparátor jelzésére
- Adott hosszúságú impulzus hardveres előállítása

PWM jel előállítása



PWM előállítása

- Külön állítható a periódus (PR2) és külön a kitöltés (CCPRxH:CCPRxL)
- Ciklus elején kimenet 1-be áll
- Kitöltés elérésekor kimenet 0-ba áll
- Alkalmazási példák:
  - D/A átalakítás (külső szűrés!)
  - Kijelző fényerő állítása
  - PWM jel továbbítás (pl.: rádió csatornán)
  - DC motor sebességének változtatása (pl. PC ventilátor)

PWM bekapcsolása

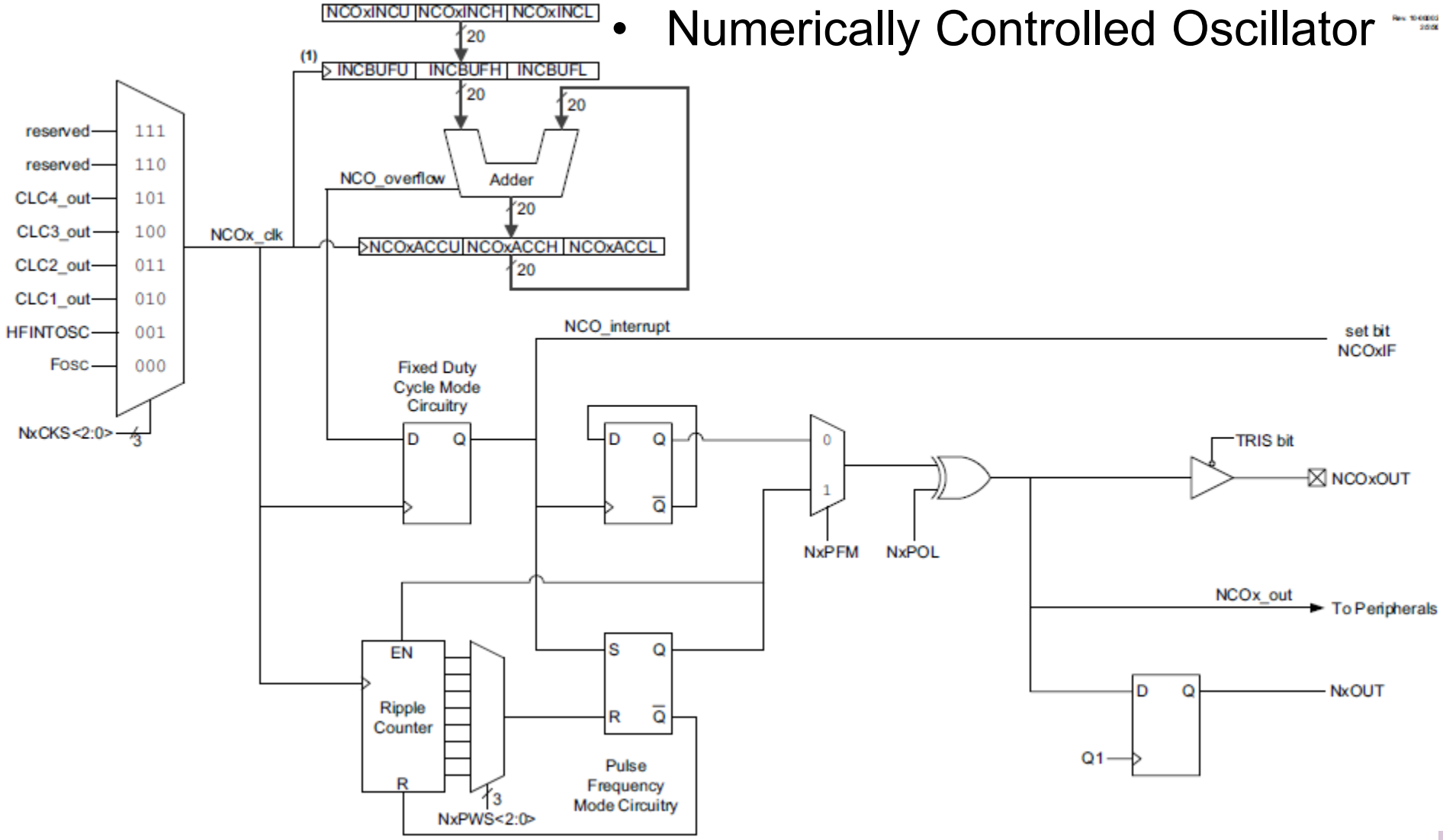
```
PWMINIT:
    BANKSEL   T2CON
    MOVLW     H'01'
    MOVWF     T2CLKCON ;timer2 órajele fosc/4 -> 8MHz
    MOVLW     B'10100000' ;on, elosztó(3 bit), utósztó (4 bit)
    MOVWF     T2CON   ;T2 elosztó 1:4-es -> 2MHz
    MOVLW     D'255'
    MOVWF     T2PR     ;a timer2 255-ig számol, majd újraindul (8MHz/4/255=7,8kHz)
    MOVLW     D'0'
    BANKSEL   PWM7CON
    MOVWF     PWM7DCH ;kezdeti kitöltés 0%
    BSF       PWM7CON,PWM7EN ;PWM modul indul
    BANKSEL RC2PPS
    MOVLW   H'0F'    ; PWM7output
    MOVWF   RC2PPS   ; PORT RC2-re

    BANKSEL   LATA
    BCF       TRISC,2       ; RC2 portbit kimenetre
    RETURN
```

PWM használata

```
PWMSET: ; kitöltés beállítása W szerint 0..255
        BANKSEL PWM7DCH
        MOVWF   PWM7DCH
        BANKSEL LATA
        RETURN
```

NCO modul

- Numerically Controlled Oscillator



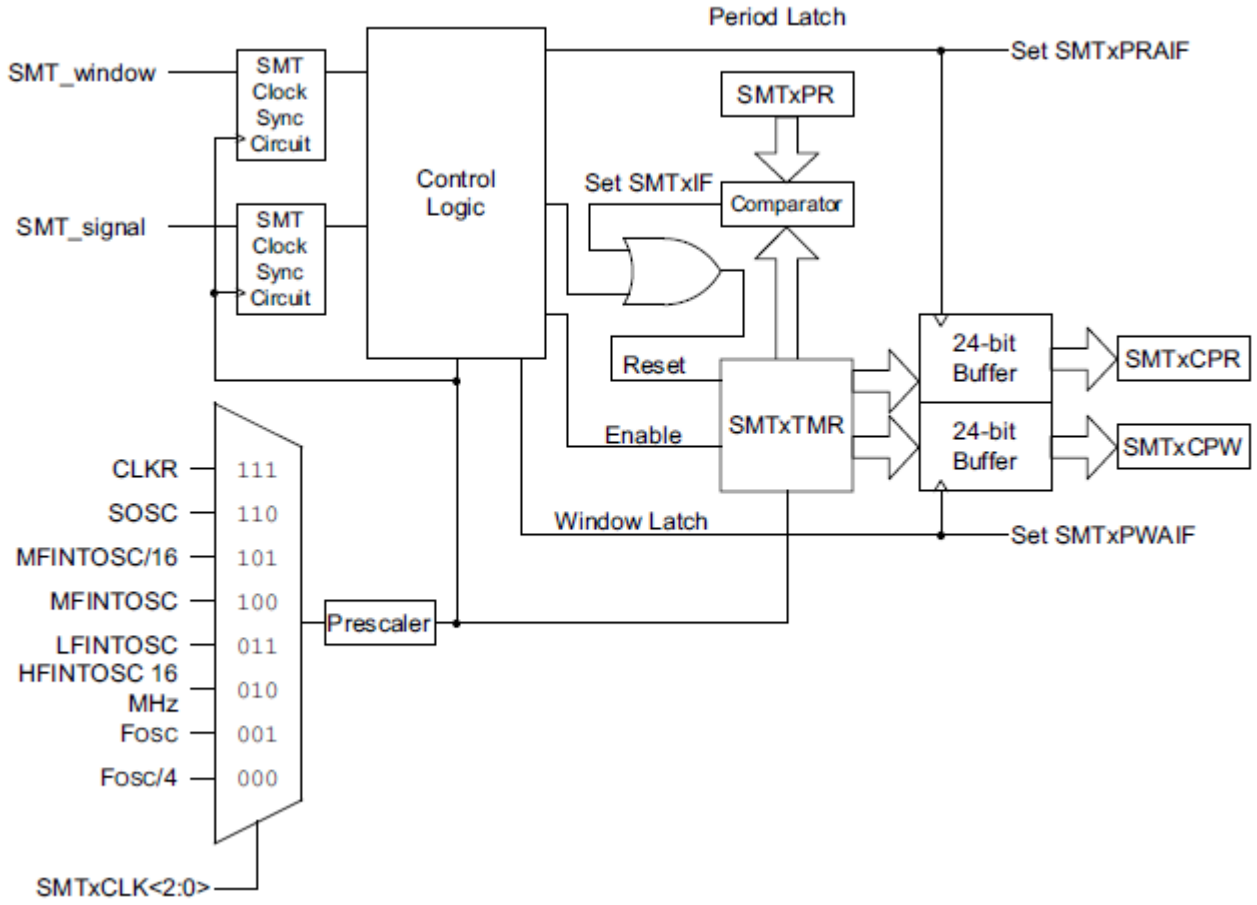
NCO modul

- A külső órajel ütemében (NCOx_clk)
- egy regiszter (NCOxACCU) értékét növeli
- megadható lépésközzel (NCOxINC)
- 20 bites
- Működhet fix kitöltési tényezővel vagy
- Megadott impulzusszélességgel
- A kimeneti frekvencia nagy felbontással állítható

$$\Delta = \frac{\mathrm{NCOx_clk}}{2^{20}}$$

SMT modul

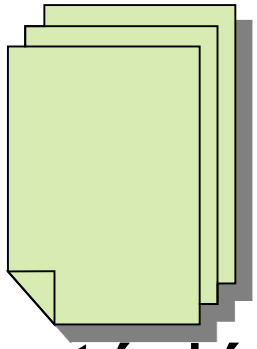
Signal measurement timer



SMT modul

- 24 bites időzítő
- Precíziós mérési feladatok időzítéséhez
- Képes megszakítást okozni
- Más perifériákat is tud működtetni (indítani)

A témához kapcsolódó anyagok



- Microchip weboldala

  http://www.microchip.com
- MPLABX fejlesztői környezet és dokumentáció letöltése

  https://www.microchip.com/mplab/mplab-x-ide
- Online help a mikrokontrollerhez és a fejlesztői környezethez

  http://microchipdeveloper.com/mcu1102:start
- Tárgy honlapján

  https://www.iit.bme.hu