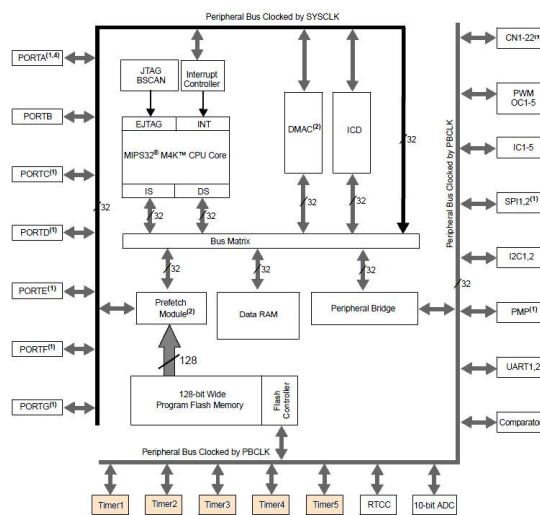


מערכות משובצות מחשב

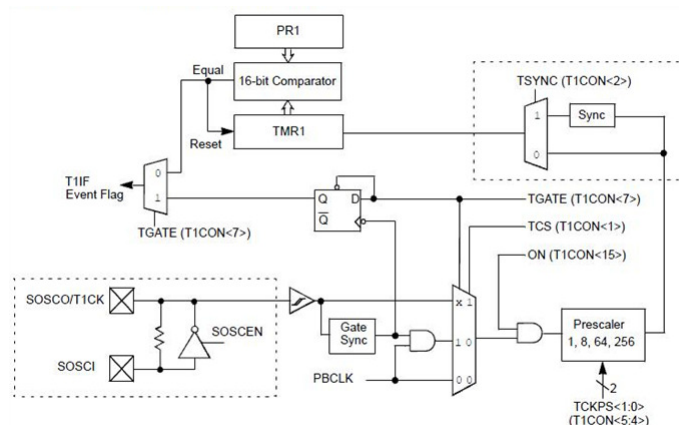
Computer Embedded Systems

Timers In PIC32

PIC32 – דיאגראמת בלוקים



Timer1 – דיאגראמת בלוקים



Timer1: Configuration

- שעון ה-CPU מחובר למעגל ה-PreScaler אשר קובע את יחס חלוקת ה-clock בין הערכים: 1,8,64,256
- מעגל ה-Tsync קובע את אופן העבודה של הטיימר – סינכרוני / א-סינכרוני.
- בכל פעימת שעון TMR1 מונה כלפי מעלה והמשווה של-16 ביט בודק שוויון בין ה-TMR1 לבין ה-PR1.
- טיימר זה פעול ב-16 ביט בלבד.

Timer	Low-Power Oscillator	Asynchronous External Clock	16-bit Synchronous Timer/Counter	32-bit Synchronous Timer/Counter	Gated Timer	Special Event Trigger
Timer 1	Yes	Yes	Yes	No	Yes	No

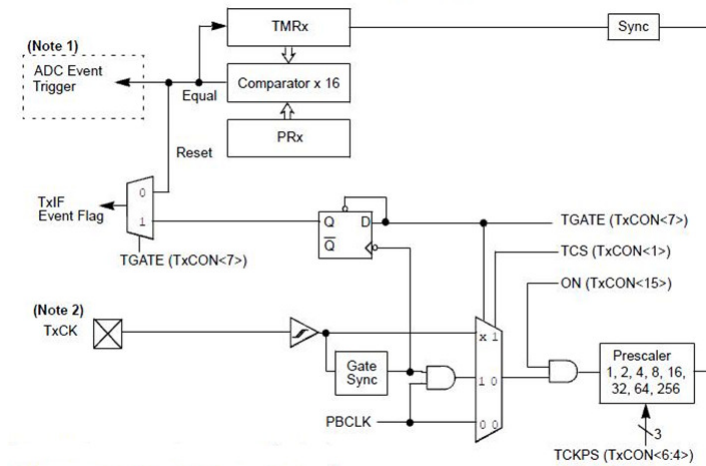
Timer configuration Register: T1CON

REGISTER 13-1: T1CON: TIMER1 CONTROL REGISTER

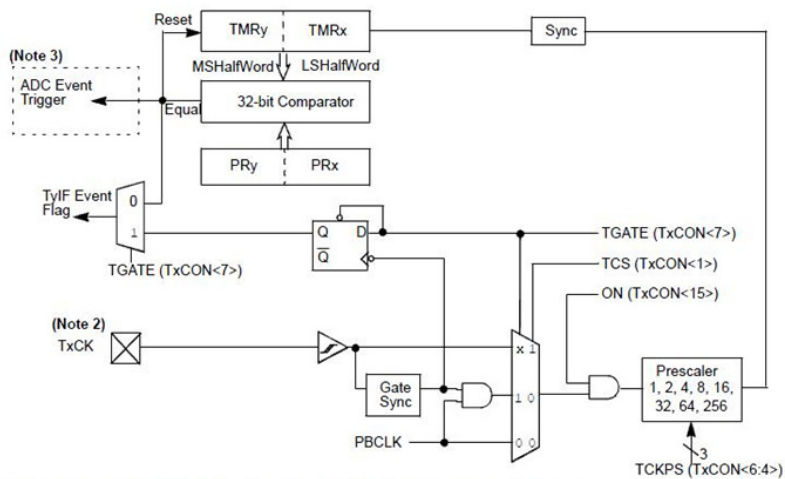
r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x
---	---	---	---	---	---	---	---
bit 31							bit 24
r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x
---	---	---	---	---	---	---	---
bit 23							bit 16
R/W-0	R/W-0	R/W-0	R/W-0	R-0	r-x	r-x	r-x
ON	FRZ	SIDL	TWDIS	TWIP	---	---	---
bit 15							bit 8
R/W-0	r-x	R/W-0	R/W-0	r-x	R/W-0	R/W-0	r-x
TGATE	---	TCKPS<1:0>	---	TSYNC	TCS	---	---
bit 7							bit 0

- bit 31-16 **Reserved:** Write '0'; ignore read
- bit 15 **ON:** Timer On bit
1 = Timer is enabled
0 = Timer is disabled
- bit 14 **FRZ:** Freeze in Debug Exception Mode bit
1 = Freeze operation when CPU is in Debug Exception mode
0 = Continue operation when CPU is in Debug Exception mode
Note: FRZ is writable in Debug Exception mode only, it is forced to '0' in normal mode.
- bit 13 **SIDL:** Stop in Idle Mode bit
1 = Discontinue operation when device enters idle mode
0 = Continue operation in idle mode
- bit 12 **TWDIS:** Asynchronous Timer Write Disable bit
In Asynchronous Timer mode:
1 = Writes to asynchronous TMR1 are ignored until pending write operation completes
0 = Back-to-back writes are enabled (legacy asynchronous timer functionality)
In Synchronous Timer mode:
This bit has no effect.
- bit 11 **TWIP:** Asynchronous Timer Write in Progress bit
In Asynchronous Timer mode:
1 = Asynchronous write to TMR1 register in progress
0 = Asynchronous write to TMR1 register complete
In Synchronous Timer mode:
This bit is read as '0'.
- bit 10-8 **Reserved:** Write '0'; ignore read
- bit 7 **TGATE:** Gated Time Accumulation Enable bit
When TCS = 1:
This bit is ignored and read '0'.
When TCS = 0:
1 = Gated time accumulation is enabled
0 = Gated time accumulation is disabled
- bit 6 **Reserved:** Write '0'; ignore read
- bit 5-4 **TCKPS<1:0>:** Timer Input Clock prescaler Select bits
11 = 1:256 prescale value
10 = 1:64 prescale value
01 = 1:8 prescale value
00 = 1:1 prescale value
- bit 3 **Reserved:** Write '0'; ignore read
- bit 2 **TSYNC:** Timer External Clock Input Synchronization Selection bit
When TCS = 1:
1 = External clock input is synchronized
0 = External clock input is not synchronized
When TCS = 0:
This bit is ignored and read '0'.
- bit 1 **TCS:** Timer Clock Source Select bit
1 = External clock from T1CK1 pin
0 = Internal peripheral clock
- bit 0 **Reserved:** Write '0'; ignore read

Timers2÷5 16 bit mode



Timers2÷5 32 bit mode



מעגל Timers2÷5

- ההבדל בין קונפיגורציית 16\32 ביט אינו משפיעה על אופן הפעולה של המעגל, והמעגל פועל בדומה למעגל של Timer1
- על מנת להשתמש במונה של 32 ביט יש לשלב אחד מן צמד המונים הבאים: Timer2/3, Timer4,5.
- כאשר משלבים 2 טיימרים ליצירת טיימר 32 ביט טיימרים 2,4 יהוו את 16 סיביות ה LSB.

Timers	Low-Power Oscillator	Asynchronous External Clock	16-bit Synchronous Timer	32-bit Synchronous Timer ⁽¹⁾	Gated Timer	Special Event Trigger
2, 4	No	No	Yes	Yes	Yes	No
3, 5	No	No	Yes	Yes	Yes	Yes ⁽²⁾

T2CON/T4CON Registers

REGISTER 14-9: T2CON, T4CON: TIMER2 AND TIMER4 CONTROL REGISTER

r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x
—	—	—	—	—	—	—	—	—
bit 31								bit 24
r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x
—	—	—	—	—	—	—	—	—
bit 23								bit 16
R/W-0	R/W-0	R/W-0	r-x	r-x	r-x	r-x	r-x	r-x
ON	FRZ	SIDL	—	—	—	—	—	—
bit 15								bit 8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	r-x	R/W-0	r-x	r-x
TGATE	TCKPS<2:0>			T32	—	TCS	—	—
bit 7								bit 0

bit 31-16 **Reserved:** Write '0'; ignore read

bit 15 **ON:** Timer On bit

1 = Timer is enabled

0 = Timer is disabled

bit 14 **FRZ:** Freeze in Debug Exception Mode bit

1 = Freeze operation when CPU is in Debug Exception mode

0 = Continue operation when CPU is in Debug Exception mode

Note: FRZ is writable in Debug Exception mode only, it is forced to '0' in normal mode.

bit 13 **SIDL:** Stop in Idle Mode bit

1 = Discontinue operation when device enters Idle mode

0 = Continue operation in Idle mode

bit 12-8 **Reserved:** Write '0'; ignore read

bit 7 **TGATE:** Gated Time Accumulation Enable bit

When TCS = 1:

This bit is ignored and read '0'.

When TCS = 0:

1 = Gated time accumulation is enabled

0 = Gated time accumulation is disabled

bit 6-4 **TCKPS<2:0>:** Timer Input Clock Prescaler Select bits

111 = 1:256 prescale value

110 = 1:64 prescale value

101 = 1:32 prescale value

100 = 1:16 prescale value

011 = 1:8 prescale value

010 = 1:4 prescale value

001 = 1:2 prescale value

000 = 1:1 prescale value

bit 3

T32: 32-bit Timer Mode Select bits

1 = TMRx and TMRy form a 32-bit timer

0 = TMRx and TMRy form separate 16-bit timers

bit 2

Reserved: Write '0'; ignore read

bit 1

TCS: Timer Clock Source Select bit

1 = External clock from TxCK pin

0 = Internal peripheral clock

bit 0

Reserved: Write '0'; ignore read

T3CON/T5CON Registers

REGISTER 14-10: T3CON, T5CON: TIMER3 AND TIMER5 CONTROL REGISTER

r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x
—	—	—	—	—	—	—	—	—
bit 31								bit 24
r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x
—	—	—	—	—	—	—	—	—
bit 23								bit 16
R/W-0	R/W-0	R/W-0	r-x	r-x	r-x	r-x	r-x	r-x
ON	FRZ	SIDL	—	—	—	—	—	—
bit 15								bit 8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	r-x	R/W-0	r-x	r-x
TGATE	TCKPS<2:0>			—	—	TCS	—	—
bit 7								bit 0

bit 31-16 **Reserved:** Write '0'; ignore read

bit 15 **ON:** Timer On bit

1 = Module is enabled

0 = Module is disabled

bit 14 **FRZ:** Freeze in Debug Exception Mode bit

1 = Freeze operation when CPU is in Debug Exception mode

0 = Continue operation when CPU is in Debug Exception mode

Note: FRZ is writable in Debug Exception mode only, it is forced to '0' in normal mode.

bit 13 **SIDL:** Stop in Idle Mode bit

1 = Discontinue operation when device enters Idle mode

0 = Continue operation in Idle mode

bit 12-8 **Reserved:** Write '0'; ignore read

bit 7 **TGATE:** Gated Time Accumulation Enable bit

When TCS = 1:

This bit is ignored and read '0'.

When TCS = 0:

1 = Gated time accumulation is enabled

0 = Gated time accumulation is disabled

bit 6-4 **TCKPS<1:0>:** Timer Input Clock Prescaler Select bits

111 = 1:256 prescale value

110 = 1:64 prescale value

101 = 1:32 prescale value

100 = 1:16 prescale value

011 = 1:8 prescale value

010 = 1:4 prescale value

001 = 1:2 prescale value

000 = 1:1 prescale value

bit 3-2 **Reserved:** Write '0'; ignore read

bit 1

TCS: Timer Clock Source Select bit

1 = External clock from TxCK pin

0 = Internal peripheral clock

bit 0

Reserved: Write '0'; ignore read

תוכנית לדוגמא:

```
#include "p32xxxx.h"
#pragma config FPLLMUL = MUL_20, FPLLDIV = DIV_2, FPLLODIV = DIV_1, FWDTEN = OFF
#pragma config POSCMOD = HS, FNOSC = PRIPLL, FPBDIV = DIV_8
#define TRIS_MAIN_ADDR_DECODER {unsigned int portfMap = TRISF; portfMap &= 0xFFFFFEF8; TRISF=portfMap;}
#define MAIN_ADDR_DECODER(CBA) PORTF = CBA
#define MAIN_DECODER_CS_RF8
#define TRIS_DATA_OUT {unsigned int porteMap = TRISE; porteMap &= 0xFFFFF00; TRISE = porteMap;}
#define TRIS_LED_CLK {unsigned int portdMap = TRISD; portdMap &= 0xFFFFFEF; TRISD = portdMap;}
#define LED_CLK_RD4
#define LED_EN 0x04
#define DISABLE 0x01
void delay(void);
void main(void)
{
    TRIS_MAIN_ADDR_DECODER // portMap ← TRISF, portMap bits 1,2,3,9 = '0', TRISF ← portMap
    MAIN_DECODER_CS = DISABLE; // Decoder 3x8 Disable
    TRIS_DATA_OUT // portMap ← TRISE, portMap bits 1÷8 = '0', TRISE ← portMap
    TRIS_LED_CLK // portMap ← TRISD, portMap bit 5 = '0', TRISD ← portMap
    LED_CLK = 0x0;
    MAIN_ADDR_DECODER(LED_EN); // Decoder 3x8 LED Select
}
```

```
while(1) // Endless Loop
{
    PORTE = 0x55; // LED 0x55, 0101 0101
    LED_CLK = 0x01; // Enable Pulse 1 → 0
    LED_CLK = 0x00;
    delay(); // Delay Loop - Timer
    PORTE = 0xAA; // LED 0xAA, 1010 1010
    LED_CLK = 0x01; // Enable Pulse 1 → 0
    LED_CLK = 0x00;
    delay(); // Delay Loop - Timer
}

void delay(void) // Delay Loop with Timer1
{
    T1CONbits.ON=0; // Timer1 = OFF
    T1CONbits.TGATE=0; // Bit Is ignored
    T1CONbits.TCS=0; // Timer1 Clock Source (internal)
    T1CONbits.TCKPS0=1; // Timer PreScaler TCKPS<'1'> = 1:256
    T1CONbits.TCKPS1=1;
    T1CONbits.TSYNC=1; // Sync Clock
    TMR1=0; // Timer1 Start Value
    PR1=0X04FF; // Timer1 Period Value
    T1CONbits.ON=1; // Timer1 = ON
    IFS0bits.T1IF=0; // Reset Interrupt Flag
    while(!IFS0bits.T1IF); // Done Check – Interrupt Flag
}
```

Delay with 32 bit timer

```
void delay(void)
{
    T2CONbits.ON=0;           // Timer2 = OFF
    T2CONbits.TGATE=0;       // Gated Accumulation Disabled
    T2CONbits.TCS=0;         // Timer Clock Source – Internal Peripheral Clock
    T2CONbits.T32=1;         // Timer In 32bit Mode
    T2CONbits.TCKPS0=1;      // Timer PreScaler TCKPS<'111'> = 1:256
    T2CONbits.TCKPS1=1;
    T2CONbits.TCKPS2=1;
    T3CONbits.ON=0;          // Timer3 = OFF
    TMR3=0;                  // Timer3 Start Value, TMRy – Upper Word Half
    TMR2=0;                  //Timer2 Start Value, TMRx - Lower Word Half
    PR2=0X5000;              //Timer 2 Period Value, PR2x
    PR3=0;                   // Timer3 Period Value, PR2y
    T2CONbits.ON=1;          // Timer2 = ON
    IFS0bits.T2IF=0;         // Reset Timer2 Interrupt Flag
    IFS0bits.T3IF=0;         // Reset Timer3 Interrupt Flag
    while(!IFS0bits.T3IF);   // Done Check – Timer3 Interrupt Flag
}
```