

Using Timer1/3/5 and the Capture-Compare-PWM Circuit

Corrado Santoro

ARSLAB - Autonomous and Robotic Systems Laboratory

Dipartimento di Matematica e Informatica - Università di Catania, Italy

santoro@dmf.unict.it



L.A.P. 1 Course

The other timers of of PIC18

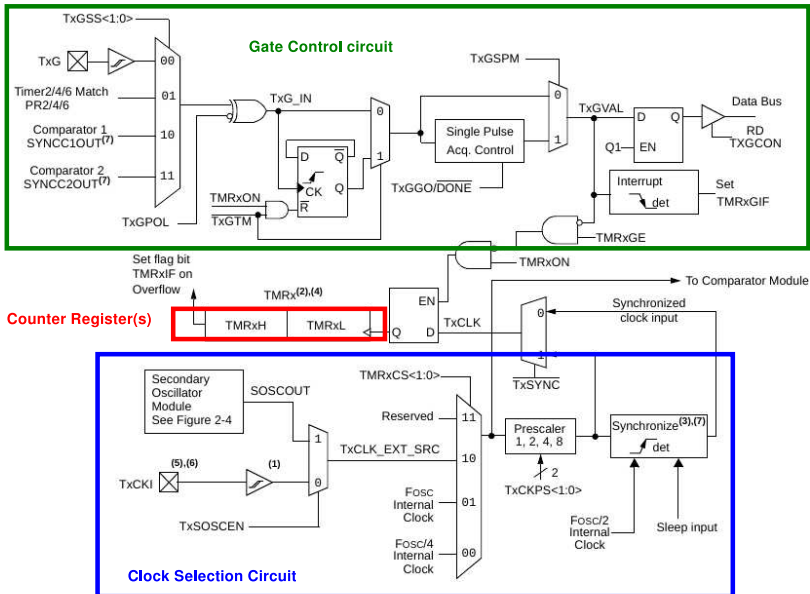
- In addition to **TIMER0**, PIC18 Family has other 6 timers, subdivided in two families
- First family includes **TIMER1**, **TIMER3** and **TIMER5**
- Second family includes **TIMER2**, **TIMER4** and **TIMER6**
- Timers of the same family have the same structure and working scheme

The timers 1/3/5 of PIC18

These timers are more versatile than the others and, therefore, are a little bit more complex:

- 16-bit timers
- Clock source selectable from:
 - $FOSC$
 - $FOSC/4$
 - External input
 - Secondary oscillator module
- Prescaler with division by 1, 2, 4, 8;
- Overflow ($0xFFFF \rightarrow 0$) triggers an interrupt.
- **Gated timers**, i.e. counting can be enabled by:
 - Bit TMRxON (as in other timers)
 - External input **TxG**
 - Specific event source coming from other peripherals

The timers 1/3/5 of of PIC18



Control Register of timers 1/3/5

REGISTER 12-1: TXCON: TIMER1/3/5 CONTROL REGISTER

R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/0	R/W-0/u
TMRxCS<1:0>		TxCKPS<1:0>		TxSOSCEN	TxSYNC	TxRD16	TMRxON
bit 7							bit 0

- **TMRxCS**: Clock source select
 - **00**, $F_{OSC}/4$
 - **01**, F_{OSC}
 - **10** & **TxSOSCEN** = **0**, External clock from TxCKI pin
 - **11** & **TxSOSCEN** = **1**, Clock from external crystal
- **TxCKPS**: Prescaler setup
 - **00** = 1:1 prescaler
 - **01** = 1:2 prescaler
 - **10** = 1:4 prescaler
 - **11** = 1:8 prescaler
- **TxRD16**: 16-bit R/W Operations
- **TMRxON**: Timer On/Off control

Gate Control Register of timers 1/3/5

REGISTER 12-2: TXGCON: TIMER1/3/5 GATE CONTROL REGISTER

R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W/HC-0/u	R-x/x	R/W-0/u	R/W-0/u
TMRxGE	TxGPOL	TxGTM	TxGSPM	TxGGO/DONE	TxGVAL	TxGSS<1:0>	
bit 7							bit 0

- **TMRxGE**: Gate Enable bit
 - **0**, Gate is disabled
 - **1**, Gate is enabled and gate function is controlled by other bits

Interrupt flags of timers 1/3/5

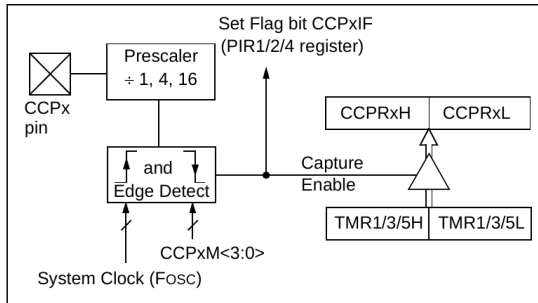
Interrupt (overflow) flags of timers 1/3/5 belong to various special function registers:

- **TMR1IF** ⇒ **PIR1bits.TMR1IF**
- **TMR1IE** ⇒ **PIE1bits.TMR1IE**
- **TMR3IF** ⇒ **PIR2bits.TMR3IF**
- **TMR3IE** ⇒ **PIE2bits.TMR3IE**
- **TMR5IF** ⇒ **PIR5bits.TMR5IF**
- **TMR5IE** ⇒ **PIE5bits.TMR5IE**

The Capture-Compare-PWM Circuit (CCP)

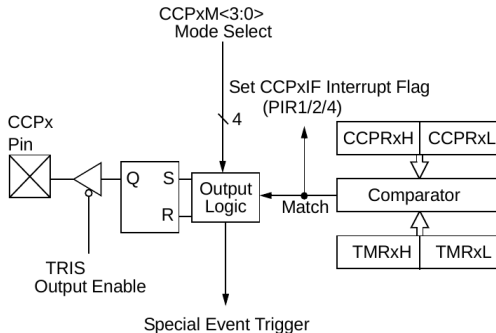
- The CCP is a special circuit which performs certain signal manipulation and measurement functions. It has 3 working modes:
 - **Capture**: at the occurrence of a pre-programmed **edge** in an input signal, it takes a snapshot of the value of a timer
 - **Compare**: when the value of a timer *matches* a pre-programmed constant, the circuit **generates** a specific output signal
 - **PWM**: it generates a PWM signal *totally in hardware*
- Capture/Compare events can drive a proper interrupt
- The 18F25K22 has 5 CCP modules
- Each one can be programmed to work in one of the three modes above
- Each CCP module has its specific input/output pin

Capture Mode



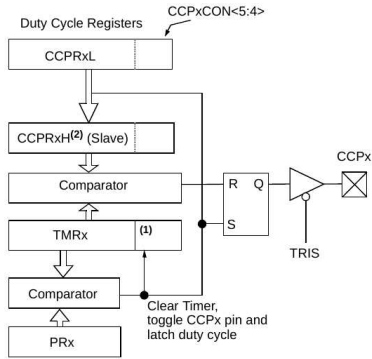
- **Input signal** is taken from **CCPx** pin (which is multiplexed with a digital I/O port)
- The **edge detector** is able to identify a *falling-* or *rising-edge* in the input signal, according to the value of bits **CCPxM** (CCP Mode)
- The programmable **prescaler** is able to identify each edge, each 4th edge or each 16th edge
- When the (programmed) edge is detected, the value of TIMER1/3/5 is copied into the **CCPRx** register and (optionally) an interrupt is generated
- All the process occurs **in hardware**

Compare Mode



- **CCPx** pin acts as **output**
- The software can program a value in the **CCPRx** register
- When the value of **TIMER1/3/5 matches** the value in **CCPRx** one of the following programmed events occurs:
 - The output is set to 0
 - The output is set to 1
 - The output is *toggle*d (0 → 1, 1 → 0)
- An optional interrupt is also generated
- All the process occurs **in hardware**

PWM Mode



- **CCPx** pin acts as **output**
- One of the **TIMER2/4/6** is used
- The **PWM period** is programmed into the **PRx** register (timer period register)
- The **duty cycle** is programmed into the **CCPRx** register
- All the process occurs **in hardware**

CCP Control Register

REGISTER 14-1: CCPxCON: STANDARD CCPx CONTROL REGISTER

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
—	—	DCxB<1:0>		CCPxM<3:0>				
bit 7								bit 0

- **DCxB**: duty cycle bits 8 and 9 (used in PWM Mode)
- **CCPxM**: CCP Mode
 - **0000**, circuit is off
 - **0010**, Compare mode, toggle output on match
 - **1000**, Compare mode, “set” output on match
 - **1001**, Compare mode, “clear” output on match
 - **0100**, Capture mode, every falling edge
 - **0101**, Capture mode, every rising edge
 - **0110**, Capture mode, every 4th rising edge
 - **0111**, Capture mode, every 16th rising edge
 - **11xx**, PWM mode

CCP Timer Selection

CCPTMRS0 and CCPTMRS1 registers are used to select the timer to use in each CCP circuit.

REGISTER 14-3: CCPTMRS0: PWM TIMER SELECTION CONTROL REGISTER 0

R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
C3TSEL<1:0>	—	C2TSEL<1:0>	—	C1TSEL<1:0>			
bit 7						bit 0	

- **CxTSEL**: Timer selection
 - **00**, Capture/compare uses Timer1, PWM uses Timer2
 - **01**, Capture/compare uses Timer3, PWM uses Timer4
 - **10**, Capture/compare uses Timer5, PWM uses Timer6

Interrupt flags of CCP 1–5

Interrupt (event) flags of CCP circuits from 1 to 5 belong to various special function registers:

- **CCP1IF** ⇒ **PIR1bits.CCP1IF**
- **CCP1IE** ⇒ **PIE1bits.CCP1IE**
- **CCP2IF** ⇒ **PIR2bits.CCP2IF**
- **CCP2IE** ⇒ **PIE2bits.CCP2IE**
- **CCP3IF** ⇒ **PIR4bits.CCP3IF**
- **CCP3IE** ⇒ **PIE4bits.CCP3IE**
- **CCP4IF** ⇒ **PIR4bits.CCP4IF**
- **CCP4IE** ⇒ **PIE4bits.CCP4IE**
- **CCP5IF** ⇒ **PIR4bits.CCP5IF**
- **CCP5IE** ⇒ **PIE4bits.CCP5IE**

CCP Circuit PIN Assignment in PIC18F25K22

- **CCP1** $\Rightarrow$ **RC2**
- **CCP2** $\Rightarrow$ **RC1** or **RB3** (settable through a configuration register)
- **CCP3** $\Rightarrow$ **RC6** or **RB5** (settable through a configuration register)
- **CCP4** $\Rightarrow$ **RB0**
- **CCP5** $\Rightarrow$ **RA4**

Using Timer1/3/5 and the Capture-Compare-PWM Circuit

Corrado Santoro

ARSLAB - Autonomous and Robotic Systems Laboratory

Dipartimento di Matematica e Informatica - Università di Catania, Italy

santoro@dmf.unict.it



L.A.P. 1 Course