Timer1 Module

HIGHLIGHTS

This section of the manual contains the following topics:

© 2016-2018 Microchip Technology Inc.

Note: This family reference manual section is meant to serve as a complement to device data sheets. Depending on the device, this manual section may not apply to all dsPIC33/PIC24 devices. Some dsPIC33/PIC24 devices are dual core and contain both a Master and Slave CPU core. For single core dsPIC33/PIC24 devices, disregard any Slave-specific references.

Please consult the note at the beginning of the "**Timer1**" chapter in the current device data sheet to check whether this document supports the device you are using.

Device data sheets and family reference manual sections are available for download from the Microchip Worldwide Web site at: http://www.microchip.com

1.0 INTRODUCTION

The Timer1 module found in some dsPIC33/PIC24 family devices shares many features with a classic Type A timer. There is typically a single Timer1 module implemented for each CPU core present in a device. Timers are useful for generating accurate time-based periodic interrupt events for software applications or real-time operating systems. Other uses include counting external pulses or accurate timing measurement of external events by using the timer's gate feature.

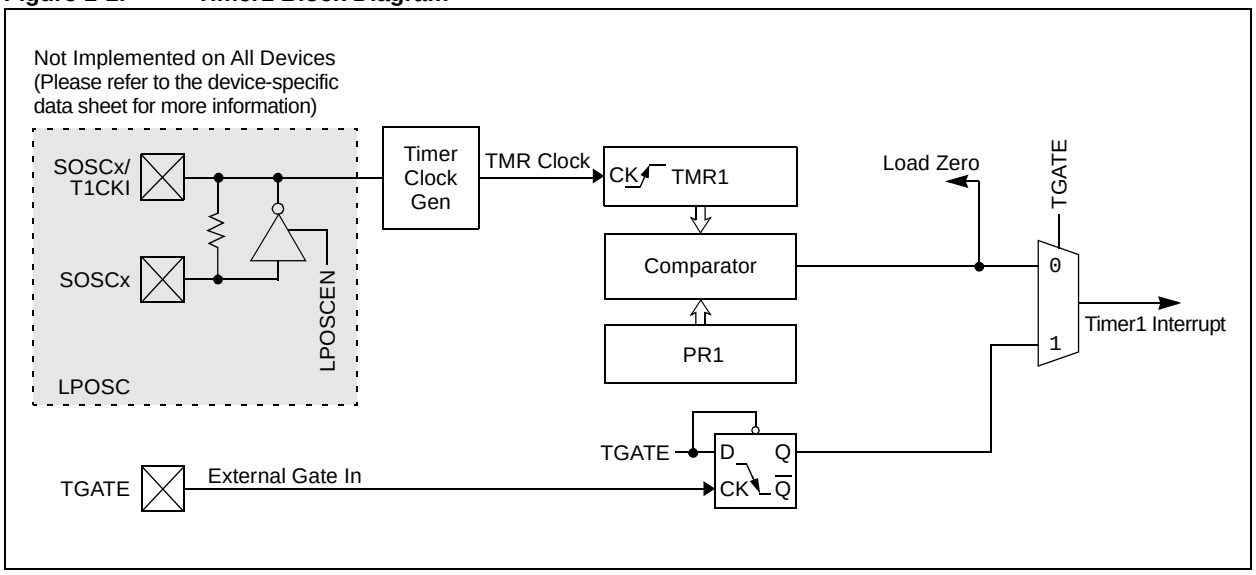
The timers found in some dsPIC33/PIC24 family devices include the following features:

- Asynchronous and Synchronous Operation
- Software-Selectable Internal or External Clock Sources
- Programmable Interrupt Generation and Priority
- Gated External Pulse Counter
- Operation during Sleep mode
- Software Prescalers: 1:1, 1:8, 1:64 and 1:256

Timer1 does not support 32-bit mode.

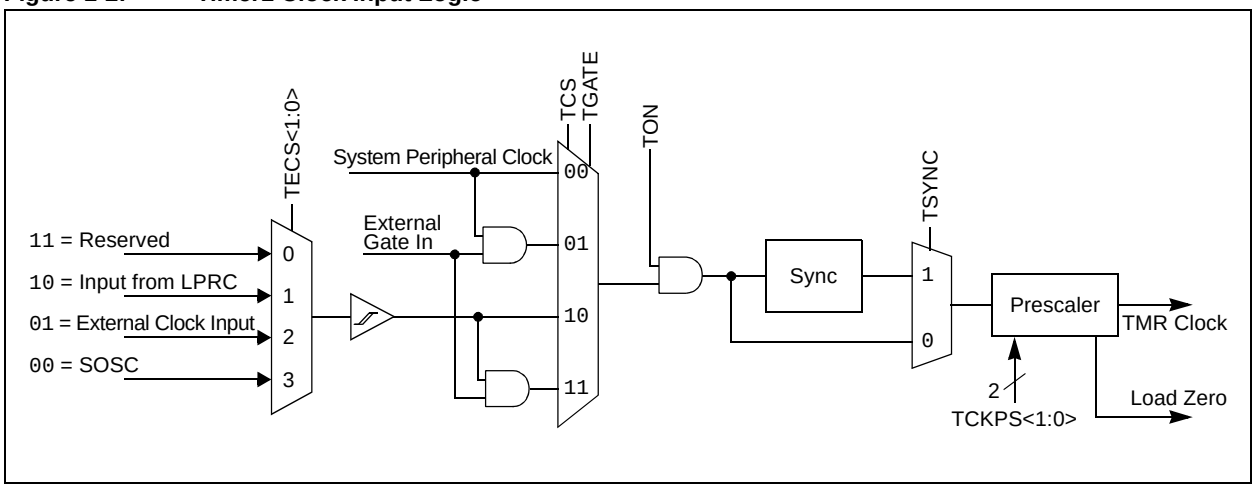
The unique features of the Timer1 module allow it to be used for Real-Time Clock (RTC) applications. Figure 1-1 illustrates the block diagram of a Timer1 module and Figure 1-2 shows the Timer1 clock input logic.

Figure 1-1: Timer1 Block Diagram



© 2016-2018 Microchip Technology Inc.

Figure 1-2: Timer1 Clock Input Logic



2.0 CONTROL REGISTERS

Note: Each dsPIC33/PIC24 family device may have one or more timer modules. An 'x' used in the names of pins, control/status bits and registers denotes the particular module. For more information, refer to the specific device data sheet.

Each Timer module is a 16-bit timer/counter that consists of the following Special Function Registers (SFRs), which are summarized in Table 2-1:

- **T1CON: Timer1 Control Register**
  This register provides control features for the timer.
- **TMR1: Timer1 Register**
  This register provides the count for the timer.
- **PR1: Timer1 Period Register**
  This register provides the TMR1 match value.

  Each Timer module also has the following associated bits for interrupt control:
  - T1IE: Timer1 Interrupt Enable bit in the IECx Interrupt register
  - T1IF: Timer1 Interrupt Flag bit in the IFSx Interrupt register
  - T1IP<2:0>: Timer1 Interrupt Priority bits in the IPCx Interrupt register

Note: Refer to **"Interrupts"** in the "*dsPIC33/PIC24 Family Reference Manual*" (DS70000600) for more information on these registers.

© 2016-2018 Microchip Technology Inc.

Table 2-1: Timer1 SFR Summary

Register Name	Bit Range	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T1CON	15:0	TON	—	SIDL	TWDIS	TWIP	PRWIP	TECS1	TECS0	TGATE	—	TCKPS1	TCKPS0	—	TSYNC	TCS	—
TMR1	15:0	TMR1<15:0>															
PR1	15:0	PR1<15:0>															

Legend: — = unimplemented, read as '0'.

© 2016-2018 Microchip Technology Inc.

Register 2-1: T1CON: Timer1 Control Register

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TON(1)	—	SIDL	TWDIS	TWIP	PRWIP	TECS1	TECS0
bit 15							bit 8

R/W-0	U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0
TGATE	—	TCKPS1	TCKPS0	—	TSYNC	TCS	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **TON:** Timer1 Enable bit(1)
1 = Timer1 is enabled
0 = Timer1 is disabled

bit 14 **Unimplemented:** Read as '0'

bit 13 **SIDL:** Timer1 Stop in Idle Mode bit
1 = Discontinues operation when device enters Idle mode
0 = Continues operation when device enters Idle mode

bit 12 **TWDIS:** Asynchronous Timer1 Write Disable bit
1 = Writes to TMR1 are ignored until pending write operation completes
0 = Back-to-back writes are enabled (legacy asynchronous timer functionality)

bit 11 **TWIP:** Asynchronous Timer1 Write in Progress bit
In Asynchronous Timer mode:
1 = Asynchronous write to TMR1 register is in progress
0 = Asynchronous write to TMR1 register is complete
In Synchronous Timer mode:
This bit is read as '0'.

bit 10 **PRWIP:** Asynchronous Period Write in Progress bit
1 = Write to the Period register in Asynchronous mode is pending
0 = Write to the Period register in Asynchronous mode is complete

bit 9-8 **TECS<1:0>:** Timer1 Extended Clock Select bits
11 = Reserved
10 = Clock input from internal Low-Power RC (LPRC) Oscillator
01 = External Clock (EC) input from T1CK pin
00 = Secondary Oscillator (SOSC)(2)

bit 7 **TGATE:** Timer1 Gated Time Accumulation Enable bit
When TCS = 1:
This bit is ignored.
When TCS = 0:
1 = Gated time accumulation is enabled
0 = Gated time accumulation is disabled

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

Note 1: When using the 1:1 PBCLK divisor, the user's software should not read/write the peripheral SFRs in the SYSCLK cycle immediately following the instruction that clears the module's TON bit.

2: Not all devices implement a Secondary Oscillator. Please refer to the device-specific data sheet for more information.

© 2016-2018 Microchip Technology Inc.

Register 2-1: T1CON: Timer1 Control Register (Continued)

bit 5-4	**TCKPS<1:0>:** Timer1 Input Clock Prescale Select bits 11 = 1:256 prescale value 10 = 1:64 prescale value 01 = 1:8 prescale value 00 = 1:1 prescale value
bit 3	**Unimplemented:** Read as '0'
bit 2	**TSYNC:** Timer1 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.
bit 1	**TCS:** Timer1 Clock Source Select bit When the TECS<1:0> bits are Implemented: TCS is derived from the TECS<1:0> selection. When the TECS<1:0> bits are Unimplemented: 1 = External Clock is from the T1CK pin 0 = Internal Peripheral Bus Clock
bit 0	**Unimplemented:** Read as '0'

Note 1: When using the 1:1 PBCLK divisor, the user's software should not read/write the peripheral SFRs in the SYSCLK cycle immediately following the instruction that clears the module's TON bit.

2: Not all devices implement a Secondary Oscillator. Please refer to the device-specific data sheet for more information.

Register 2-2: TMR1: Timer1 Register

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TMR1<15:0>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TMR1<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **TMR1<15:0>:** Timer1 Count Register bits

Register 2-3: PR1: Timer1 Period Register

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PR1<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PR1<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **PR1<15:0>:** Timer1 Period Register bits

© 2016-2018 Microchip Technology Inc.

3.0 MODES OF OPERATION

3.1 16-Bit Modes

The Timer1 modules found in dsPIC33/PIC24 family devices support the following modes of operation:

- 16-Bit Synchronous Clock Counter
- 16-Bit Synchronous External Clock Counter
- 16-Bit Gated Timer
- 16-Bit Asynchronous External Counter

The 16-Bit Timer modes are determined by the following bits:

- TCS (T1CON<1>): Timer1 Clock Source Select bit
- TGATE (T1CON<7>): Timer1 Gated Time Accumulation Enable bit
- TSYNC (T1CON<2>): Timer1 External Clock Input Synchronization Selection bit

3.1.1 16-BIT TIMER CONSIDERATIONS

The following should be considered when using a 16-bit timer:

- All timer module SFRs can be written to as a byte (8 bits) or as a half-word (16 bits)
- All timer module SFRs can be read from as a byte or as a half-word

3.2 16-Bit Synchronous Clock Counter Mode

The Synchronous Clock Counter mode operation provides the following capabilities:

- Elapsed time measurements
- Time delays
- Periodic timer interrupts

In this mode, the input clock source for the timer is the internal Peripheral Bus Clock, PBCLK. It is selected by clearing the Timer1 Clock Source Select bit, TCS (TxCON<1> = 0). Timer1 External Clock Input Synchronization Selection bit, TSYNC (T1CON<2>), is ignored in this mode.

Timers that use a 1:1 timer input clock prescale, operate at a timer clock rate that is the same as the PBCLK, and which increments the TMR1 Count register on every rising timer clock edge. The timer continues to increment until the TMR1 Count register matches the PR1 Period register value. The TMR1 Count register resets to 0x0000 on the next timer clock cycle, and then continues to increment and repeats the period match until the timer is disabled. If the PR1 Period register value = 0x0000, the TMR1 Count register resets to 0x0000 on the next timer clock cycle, but does not continue to increment.

Timers using a timer input clock prescale = *N* (other than 1:1) operate at a timer clock rate (PBCLK ÷ *N*), and the TMR1 Count register increments on every *N*th timer clock rising edge. For example, if the timer input clock prescale is 1:8, the timer increments on every eighth timer clock cycle. The timer continues to increment until the TMR1 Count register matches the PR1 Period register value. The TMR1 Count register then resets to 0x0000 after '*N*' more timer clock cycles, and then continues to increment and repeats the period match until the timer is disabled. If the PR1 Period register value = 0x0000, the TMR1 Count register resets to 0x0000 on the next *N*th timer clock cycle, but will not continue to increment.

Timer1 generates a timer event one-half timer clock cycle (on the falling edge) after the TMR1 Count register matches the PR1 Period register value. T1IF is set within one PBCLK, plus two SYSCLK cycles of this event, and if the Timer1 Interrupt Enable bit, T1IE, is set, an interrupt is generated.

3.2.1 16-BIT SYNCHRONOUS CLOCK COUNTER CONSIDERATIONS

The timer period is determined by the value in the PR1 Period register. To initialize the timer period, a user may write to the PR1 Period register directly at any time while the timer is disabled (TON bit = 0) or during a timer match Interrupt Service Routine (ISR) while the timer is enabled (TON bit = 1). In all other cases, writing to the Period register while the timer is enabled is not recommended and may allow unintended period matches to occur. The maximum period that can be loaded is 0xFFFF.

Writing 0x0000 to the PR1 Period register allows a TMR1 match to occur; however, no interrupt is generated.

3.2.2 16-BIT SYNCHRONOUS COUNTER INITIALIZATION STEPS

The following steps must be performed to configure the timer for 16-bit Synchronous Timer mode.

1. Clear the TON control bit (T1CON<15> = 0) to disable the timer.
2. Clear the TCS control bit (T1CON<1> = 0) to select the internal PBCLK source.
3. Select the desired timer input clock prescale.
4. Load/clear the Timer1 register, TMR1.
5. Load the Timer1 Period register, PR1, with the desired 16-bit match value.
6. If interrupts are used:
   a) Clear the T1IF interrupt flag bit in the IFSx register.
   b) Configure the interrupt priority levels in the IPCx register.
   c) Set the T1IE interrupt enable bit in the IECx register.
7. Set the TON control bit (T1CON<15> = 1) to enable the timer.

Example 3-1: 16-Bit Synchronous Clock Counter Example Code

```
T1CON = 0x0;              // Stop timer and clear control register,
                          // set prescaler at 1:1, internal clock source
TMR1 = 0x0;               // Clear timer register
PR1 = 0xFFFF;             // Load period register
T1CONbits.TON = 1;        // Start timer
```


© 2016-2018 Microchip Technology Inc.

3.3 16-Bit Synchronous External Clock Counter Mode

The Synchronous External Clock Counter mode operation provides the following capabilities:

- Counting periodic or non-periodic pulses
- Using External Clock as the time base for timers

For Timer1, the External Clock source can be connected to either the T1CK pin or it can be selected based on the settings of the Timer1 Extended Clock Select bits, TECS<1:0> (T1CON<9:8>). The Timer1 Clock Source Select bit, TCS (T1CON<1>), must be set to '1' to enable operation from an External Clock source.

Timer1 timers that use a 1:1 timer input clock prescale increment the TMR1 Count register on every rising External Clock edge after synchronization. The timer continues to increment until the TMR1 Count register matches the PR1 Period register value. The TMR1 Count register resets to 0x0000 on the next rising External Clock edge after synchronization. The timer interrupt flag is set and the CPU executes the timer Interrupt Service Routine (ISR) if the interrupt is enabled. The TMR1 Count register continues to increment and repeats the period match until the timer is disabled. If the PR1 Period register value = 0x0000, the TMR1 Count register resets to 0x0000 on the next timer clock cycle, but will not continue to increment.

Timer1 timers using a timer input clock prescale = *N* (other than 1:1) operate at a timer clock rate (External Clock ÷ *N*), and the TMR1 Count register increments on every *N*th External Clock rising edge after synchronization. For example, if the timer input clock prescale is 1:8, the timer increments on every eight External Clock cycles. The timer continues to increment until the TMR1 Count register matches the PR1 Period register value. The TMR1 Count register then resets to 0x0000 after '*N*' more External Clock cycles, and then continues to increment and repeats the period match until the timer is disabled. If the PR1 Period register value = 0x0000, the TMR1 Count register resets to 0x0000 on the next External Clock cycle, but does not continue to increment.

Timer1 generates a timer event one-half timer clock cycle (on the falling edge) after the TMR1 Count register matches the PR1 Period register value. T1IF is set within one PBCLK, plus two SYSCLK cycles of this event, and if the Timer1 Interrupt Enable bit, T1IE, is set, an interrupt is generated.

3.3.1 16-BIT SYNCHRONOUS EXTERNAL CLOCK COUNTER CONSIDERATIONS

This section describes items that should be considered when using the 16-bit synchronous External Clock counter.

Timer1 timers operating from a synchronized External Clock source will not operate in Sleep mode, since the synchronization circuit is disabled during Sleep mode.

Timer1 timers using a timer input clock prescale = *N* (other than 1:1) require two to three External Clock cycles, after the TON bit = 1, before the TMR1 Count register increments. For more information, see **Section 3.8 "Timer Latency Considerations"**.

When operating the timer in Synchronous Counter mode, the external input clock must meet certain minimum high time and low time requirements. Refer to the **"Electrical Characteristics"** chapter in the specific device data sheet for further details.

© 2016-2018 Microchip Technology Inc.

3.3.2 16-BIT SYNCHRONOUS EXTERNAL COUNTER INITIALIZATION STEPS

The following steps must be performed to configure the timer for 16-Bit Synchronous Counter mode:

1. Clear the TON control bit (T1CON<15> = 0) to disable the timer.
2. If available on the device, set the External Clock source using the TECS<1:0> bits (T1CON<9:8>).
3. Set the TCS control bit (T1CON<1> = 1) to enable the clock selection.
4. Set the TSYNC control bit (T1CON<2> = 1) to enable clock synchronization.
5. Select the desired timer input clock prescale.
6. Load/clear the Timer1 register, TMR1.
7. If using period match:
   a) Load the Timer1 Period register, PR1, with the desired 16-bit match value.
8. If interrupts are used:
   a) Clear the T1IF interrupt flag bit in the IFSx register.
   b) Configure the interrupt priority levels in the IPC1 register.
   c) Set the T1IE interrupt enable bit in the IEC1 register.
9. Set the TON control bit (T1CON<15> = 1) to enable the timer.

Example 3-2: 16-Bit Synchronous External Counter Example Code

```
T1CON = 0x0;              // Stop timer and clear control register
T1CON = 0x0106;           // Set prescaler at 1:1, external clock source
TMR1 = 0x0;               // Clear timer register
PR1 = 0xFFFF;             // Load period register
T1CONbits.TON = 1;        // Start timer
```


© 2016-2018 Microchip Technology Inc.

3.4 16-Bit Gated Timer Mode

The gate operation starts on a rising edge of the signal applied to the T1CK pin. The TMR1 Count register increments while the external gate signal remains high. The gate operation terminates on the falling edge of the signal applied to the T1CK pin. The Timer1 Interrupt Flag, T1IF, is set.

The timer clock source is the internal Peripheral Bus Clock, PBCLK, and is selected by clearing the TCS control bit (T1CON<1>) = 0. Timer1 timers automatically provide synchronization to the Peripheral Bus Clock; therefore, the Timer1 External Clock Input Synchronization Selection bit, TSYNC (T1CON<2>), is ignored in this mode. In Gated Timer mode, the input clock is gated by the signal applied to the T1CK pin. The Gated Timer mode is enabled by setting the TGATE control bit (T1CON<7>) = 1.

Timer1 timers using a 1:1 timer input clock prescale operate at a timer clock rate the same as the PBCLK and increment the TMR1 Count register on every rising timer clock edge. The timer continues to increment until the TMR1 Count register matches the PR1 Period register value. The TMR1 Count register then resets to 0x0000 on the next timer clock cycle, and then continues to increment and repeats the period match until the falling edge of the gate signal or the timer is disabled by clearing TON (T1CON<15> = 0). The timer does not generate an interrupt when a timer period match occurs.

Timer1 timers using a timer input clock prescale = *N* (other than 1:1) operate at a timer clock rate (PBCLK ÷ *N*), and the TMR1 Count register increments on every *N*th timer clock rising edge. For example, if the timer input clock prescale is 1:8, the timer increments on every eighth timer clock cycle. The timer continues to increment until the TMR1 Count register matches the PR1 Period register value. The TMR1 Count register then resets to 0x0000 after '*N*' more timer clock cycles, and continues to increment and repeats the period match until the falling edge of the gate signal or the timer is disabled by clearing TON (T1CON<15> = 0). The timer does not generate an interrupt when a timer period match occurs.

On the falling edge of the gate signal, the count operations terminate, a Timer event is generated, and the Timer1 Interrupt Flag bit (T1IF) is set one PBCLK, plus two SYSCLK system clock cycles, after the falling edge of the signal on the gate pin. The TMR1 Count register is not reset to 0x0000. Reset the TMR1 Count register if it is desired to start from zero on the next rising edge gate input.

The resolution of the timer count is directly related to the timer clock period. When the timer input clock prescale is 1:1, the timer clock period is one Peripheral Bus Clock cycle, T_{PBCLK}. For a timer input clock prescale of 1:8, the timer clock period is eight times the Peripheral Bus Clock cycle.

3.4.1 SPECIAL GATED TIMER MODE CONSIDERATIONS

This section describes the items that should be considered when using the special Gated Timer mode.

Note:	Gated Timer mode is overridden if the Timer1 Clock Source Select bit (TCS) is set to an External Clock source, TCS = 1. For Gated Timer mode operation, the internal clock source must be selected (TCS = 0).

Timer1 timers using a timer input clock prescale = *N* (other than 1:1) require two to three timer clock cycles after the TON bit = 1, before the TMR1 Count register increments. For more information, see **Section 3.8 "Timer Latency Considerations"**.

For details on gate width pulse requirements, refer to the **"Electrical Characteristics"** chapter in the specific device data sheet.

© 2016-2018 Microchip Technology Inc.

3.4.2 16-BIT GATED TIMER INITIALIZATION STEPS

The following steps must be performed to configure the timer for 16-Bit Gated Timer mode:

1. Clear the TON control bit (TxCON<15> = 0) to disable the timer.
2. Clear the TCS control bit (TxCON<1> = 0) to select the internal PBCLK source.
3. Set the TGATE control bit (T1CON<7> = 1) to enable Gated Timer mode.
4. Select the desired prescaler.
5. Clear the Timer1 register, TMR1.
6. Load the Timer1 Period register, PR1, with the desired 16-bit match value.
7. If interrupts are used:
   a) Clear the T1IF interrupt flag bit in the IFSx register.
   b) Configure the interrupt priority levels in the IPCx register.
   c) Set the T1IE interrupt enable bit in the IECx register.
8. Set the TON control bit (TxCON<15> = 1) to enable the timer.

Example 3-3: 16-Bit Gated Timer Example Code

```
T1CON = 0x0;       // Stop timer and clear control register
T1CON = 0x0080;    // Gated Timer mode, prescaler at 1:1, internal clock source
TMR1 = 0x0;        // Clear timer register
PR1 = 0xFFFF;      // Load period register with 16-bit match value
T1CONbits.TON = 1; // Start timer
```


© 2016-2018 Microchip Technology Inc.

3.5 Asynchronous Clock Counter Mode

The Asynchronous Timer Clock Counter mode operation can operate during Sleep mode and can generate an interrupt on a period register match that will wake the processor from Sleep or Idle mode.

Timer1 has the ability to operate in an Asynchronous Counting mode using an External Clock source. The External Clock source can be connected to either the T1CK pin or it can be selected based on the settings of the Timer Extended Clock Select bits, TECS<1:0> (TxCON<9:8>). The Timer1 Clock Source Select bit, TCS (T1CON<1>), must be set to '1' to enable operation from an External Clock source. This mode requires that the External Clock synchronization be disabled by setting the TSYNC bit (T1CON<2>) = 0.

Timer1, using a 1:1 timer input clock prescale, operates at the same clock rate as the applied External Clock rate and increments the TMR1 Count register on every rising timer clock edge. The timer continues to increment until the TMR1 Count register matches the Timer1 PR1 Period register value. The TMR1 Count register resets to 0x0000 on the next timer clock cycle, and then continues to increment and repeats the period match until the timer is disabled. If the PR1 Period register value = 0x0000, the TMR1 Count register resets to 0x0000 on the next timer clock cycle, but will not continue to increment.

Timer1 generates a timer event when the TMR1 Count register matches the PR1 Period register value. The Timer1 Interrupt Flag bit, T1IF, is set within one PBCLK, plus two SYSCLK system clock cycles of this event. If the Timer1 Interrupt Enable bit is set, T1IE = 1, an interrupt is generated.

3.5.1 ASYNCHRONOUS MODE TMR1 READ AND WRITE OPERATIONS

Due to the asynchronous nature of Timer1 operating in this mode, reading and writing to the TMR1 Count register requires synchronization between the asynchronous clock source and the internal PBCLK Peripheral Bus Clock. Timer1 features a control bit, the Asynchronous Timer1 Write Disable bit (TWDIS), and a status bit, the Asynchronous Timer1 Write in Progress bit (TWIP), to provide users with two options for safely writing to the TMR1 Count register while Timer1 is enabled. These bits have no effect in Synchronous Clock Counter modes.

Option 1 is the legacy Timer1 Write mode, TWDIS bit = 0. To determine when it is safe to write to the TMR1 Count register, it is recommended to poll the TWIP bit. When TWIP = 0, it is safe to perform the next write operation to the TMR1 Count register. When TWIP = 1, the previous write operation to the TMR1 Count register is still being synchronized and any additional write operations should wait until TWIP = 0.

Option 2 is the new Synchronized Timer1 Write mode, TWDIS bit = 1. A write to the TMR1 Count register can be performed at any time. However, if the previous write operation to the TMR1 Count register is still being synchronized, any additional write operations are ignored.

When performing a write to the TMR1 Count register, two to three asynchronous External Clock cycles are required for the value to be synchronized into the register.

Note:	A write to the TMR1 Count register must be performed prior to configuring Timer1 for Asynchronous mode if the proper procedure to check TWIP and TWDIS is not followed.

When performing a read from the TMR1 Count register, synchronization requires two PBCLK cycle delays between the current unsynchronized value in the TMR1 Count register and the synchronized value returned by the read operation. In other words, the value read is always two PBCLK cycles behind the actual value in the TMR1 Count register.

3.5.2 ASYNCHRONOUS CLOCK COUNTER CONSIDERATIONS

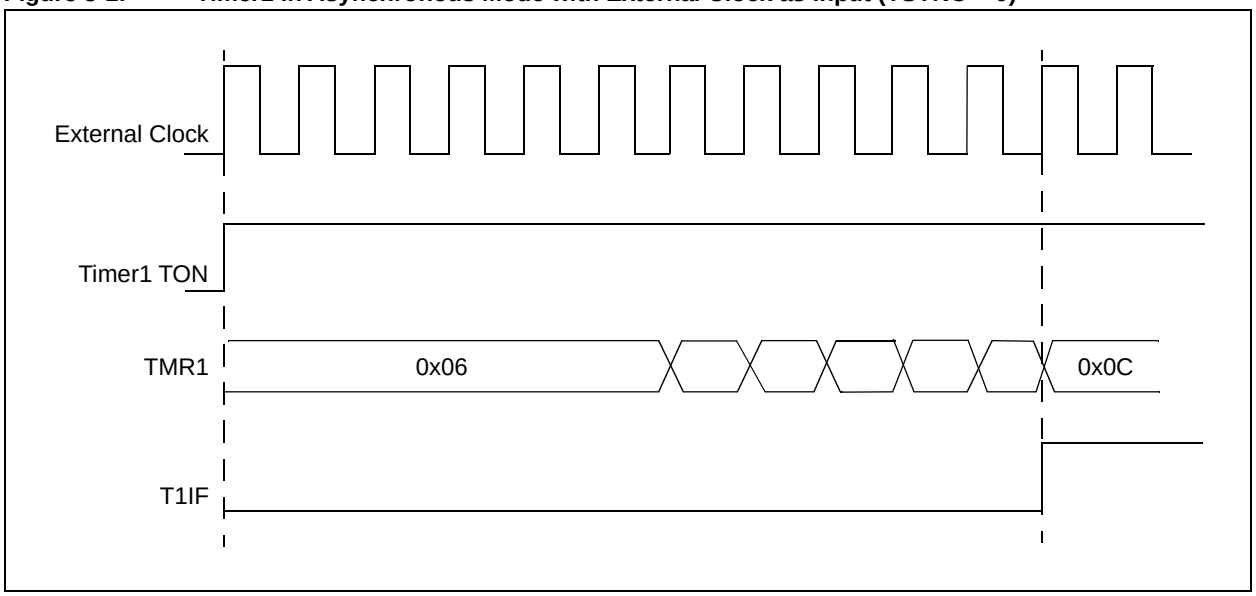
This section describes items that should be considered when using the Asynchronous Clock Counter mode.

After being enabled, Timer1 does not start counting immediately when an External Clock source is used as an input and the TSYNC bit (T1CON<2>) = 0. To circumvent this limitation, the Timer1 register can be preset with an offset value or the TSYNC bit should be set to '1'. The preset value has been empirically determined to be six.

Figure 3-1 illustrates the scenario previously mentioned, where the Timer1 register does not begin counting until the sixth clock pulse after Timer1 is enabled. The timer period match occurs after the twelfth clock pulse and the Timer1 Interrupt Flag (T1IF) is set. For more information, see **Section 3.8 "Timer Latency Considerations"**.

The external input clock must meet certain minimum high time and low time requirements when used in Asynchronous Counter mode. For more information, refer to the **"Electrical Characteristics"** chapter in the specific device data sheet.

Figure 3-1: Timer1 in Asynchronous Mode with External Clock as Input (TSYNC = 0)



© 2016-2018 Microchip Technology Inc.

3.5.3 ASYNCHRONOUS EXTERNAL CLOCK COUNTER INITIALIZATION STEPS

The following steps must be performed to configure the timer for 16-bit Asynchronous Counter mode.

1. Clear the TON control bit (T1CON<15> = 0) to disable the timer.
2. If available on the device, set the External Clock source using the TECS<1:0> bits (T1CON<9:8>).
3. Set the TCS control bit (T1CON<1> = 1) to enable External Clock selection.
4. Clear the TSYNC control bit (T1CON<2> = 0) to disable clock synchronization.
5. Select the desired prescaler.
6. Load/clear the Timer1 register, TMR1.
7. If using period match, load the Timer1 Period register, PR1, with the desired 16-bit match value.
8. If interrupts are used:
   a) Clear the T1IF interrupt flag bit in the IFSx register.
   b) Configure the interrupt priority levels in the IPCx register.
   c) Set the T1IE interrupt enable bit in the IECx register.
9. Set the TON control bit (T1CON<15> = 1) to enable the timer.

Example 3-4: 16-Bit Asynchronous Counter Mode Code Example

```
/*  16-bit Asynchronous Counter Mode Example */

T1CON = 0x0;       // Stops the Timer1 and resets the control register
TMR1 = 0x0;        // Clear timer register
T1CON = 0x0102;    // Set prescaler 1:1, external clock, asynchronous mode
PR1 = 0xFFFF;      // Load period register
T1CONbits.TON = 1; // Start timer
```

3.6 Timer Prescalers

Timer1 timers provide input clock (Peripheral Bus Clock or External Clock) prescale options of 1:1, 1:8, 1:64 and 1:256, which can be selected by using the TCKPS<1:0> bits (T1CON<5:4>).

The prescaler counter is cleared when any of the following occurs:

- A write to the TMR1 register
- Disabling the Timer1 TON bit (T1CON<15>) = 0
- Any device Reset

3.7 Writing to T1CON, TMR1 and PR1 Registers

The Timer1 module is disabled and powered off when the TON bit (T1CON<15>) = 0, thus providing maximum power savings.

To prevent unpredictable timer behavior, it is recommended that the timer be disabled before writing to any of the T1CON register bits or the timer input clock prescaler. Attempting to set the TON bit = 1 and writing to any T1CON register bits in the same instruction may cause erroneous timer operation.

The PR1 Period register can be written to while the module is operating. However, to prevent unintended period matches, writing to the PR1 Period register while the timer is enabled (TON bit = 1) is not recommended.

The TMR1 Count register can be written to while the module is operating. The user should be aware of the following points when byte writes are performed:

- If the timer is incrementing and the low byte of the timer is written to, the upper byte of the timer is not affected. If 0xFF is written into the low byte of the timer, the next timer count clock after this write will cause the low byte to roll over to 0x00 and generate a carry into the high byte of the timer.
- If the timer is incrementing and the high byte of the timer is written to, the low byte of the timer is not affected. If the low byte of the timer contains 0xFF when the write occurs, the next timer count clock will generate a carry from the timer low byte and this carry will cause the upper byte of the timer to increment.

Additionally, the TMR1 Count register can be written to while the module is operating. For information on Asynchronous Clock mode operations, see **Section 3.5.1 “Asynchronous Mode TMR1 Read and Write Operations”**.

When the TMR1 register is written to (a word, half-word or byte) via an instruction, the TMR1 register increment is masked and does not occur during that instruction cycle.

The TMR1 Count register is not reset to zero when the module is disabled.

© 2016-2018 Microchip Technology Inc.

3.8 Timer Latency Considerations

The Timer1 module can use the internal Peripheral Bus Clock (PBCLK) or an External Clock (EC). There are considerations regarding latencies of operations performed on the timer. These latencies represent the time delay between the moment an operation is executed (read or write) and the moment its first effect begins, as shown in Table 3-1.

Reading and writing the T1CON, TMR1 and PR1 registers in any Synchronized Clock mode do not require synchronization of data between the main SYSCLK clock domain and the Timer1 module clock domain. Therefore, the operation is immediate. However, when operating Timer1 in Asynchronous Clock mode, reading the TMR1 count register requires two PBCLK cycles for synchronization, while writing to theTMR1 count register requires two to three timer clock cycles for synchronization.

For example, if Timer1 is using an asynchronous clock source and a read operation of the TMR1 register is being executed, two PBCLK peripheral bus clocks are required to synchronize this data to the TMR1 Count register. Therefore, the value of the TMR1 read will be two PBCLK cycles behind the actual TMR1 count.

Additionally, if an External Clock source is being used, two to three External Clock cycles will be required after the TON bit (T1CON<15>) is set (= 1) before the timer will start incrementing.The interrupt flag latency represents the time delay between the timer event and the moment the timer interrupt flag is active.

The interrupt flag latency represents the time delay between the timer event and the moment the timer interrupt flag is active.

Table 3-1: Timer1 Latencies

Operation	PBCLK Internal Clock	Synchronous External Clock	Asynchronous External Clock
Set TON = 1 (Enables Timer)	0 PBCLKs	2-3 TMRCLKCY	2-3 TMRCLKCY
Set TON = 0 (Disables Timer)	0 PBCLKs	2-3 TMRCLKCY	2-3 TMRCLKCY
Read PR1	0 PBCLKs	0 PBCLKs	0 PBCLKs
Write PR1	0 PBCLKs	0 PBCLKs	0 PBCLKs
Read TMR1	0 PBCLKs	0 PBCLKs	2 PBCLKs
Write TMR1	0 PBCLKs	0 PBCLKs	2-3 TMRCLKCY
Interrupt Flag INTF = 1	1 PBCLK + 2 to 3 SYSCLKs	1 PBCLK + 2 to 3 SYSCLKs	(TMRCLKCY ÷ 2) + 2 to 3 SYSCLKs

Legend: TMRCLKCY = External synchronous or asynchronous timer clock cycles.

4.0 INTERRUPTS

The Timer1 module has the ability to generate an interrupt on a period match or falling edge of the external gate signal, depending on the operating mode.

The T1IF bit is set when one of the following conditions is true:

- When the TMR1 count matches the respective PR1 register and the Timer1 module is not operating in Gated Time Accumulation mode
- When the falling edge of the gate signal is detected while the Timer1 is operating in Gated Time Accumulation mode

The T1IF bit must be cleared in software.

The Timer1 module is enabled as a source of interrupt via the Timer1 Interrupt Enable bit, T1IE. The Timer1 Interrupt Priority Level bits, T1IP<2:0>.

Note:	A special case occurs when the PR1 register is loaded with '0' and the timer is enabled. An interrupts is not generated for this configuration.

4.1 Interrupt Configuration

The Timer1 module has a dedicated interrupt flag bit (T1IF) and a corresponding interrupt enable/mask bit (T1IE).

The T1IF bit is set when the timer count matches the respective Period register and the timer module is not operational in Gated Time Accumulation mode. This bit is also set if the falling edge of the gate signal is detected when the timer is operating in Gated Time Accumulation mode. The T1IF bit is set, regardless of the state of the corresponding T1IE bit. If required, the T1IF bit can be polled by software.

The T1IE bit is used to define the behavior of the interrupt controller when the T1IF bit is set. When the T1IE bit is clear, the interrupt controller does not generate a CPU interrupt for the event. If the T1IE bit is set, the interrupt controller generates an interrupt to the CPU when the T1IF bit is set (subject to the interrupt priority).

It is the responsibility of the user's software routine that services a particular interrupt to clear the appropriate interrupt flag bit before the service routine is complete.

The priority of the Timer1 module can be set with the T1IP<2:0> bits. This priority defines the priority group to which the interrupt source will be assigned. The priority groups range from a value of 7 (the highest priority) to a value of 0 (which does not generate an interrupt). An interrupt being serviced will be preempted by an interrupt in a higher priority group.

After an enabled interrupt is generated, the CPU will jump to the vector assigned to that interrupt. The vector number for the interrupt is the same as the natural order number. The CPU will then begin executing code at the vector address. The user's code at this vector address should perform any application-specific operations and clear the T1IF interrupt flag, and then exit. For more information on interrupts and vector address details, refer to **"Interrupts"** in the *"dsPIC33/PIC24 Family Reference Manual"* (DS70000600).

© 2016-2018 Microchip Technology Inc.

5.0 OPERATION IN POWER-SAVING MODES

5.1 Timer Operation in Sleep Mode

As the device enters Sleep mode, the System Clock (SYSCLK) and Peripheral Bus Clock (PBCLK) are disabled.

The Timer1 timer can operate asynchronously from an External Clock source. Therefore, the Timer1 module can continue to operate during Sleep mode.

To operate in Sleep mode, the Timer1 module is configured as follows:

- Timer1 module is enabled, TON bit (T1CON<15>) = 1
- Timer1 clock source is selected as external, TCS bit (T1CON<1>) = 1
- TSYNC bit (T1CON<2>) is set to a logic '0' (Asynchronous Counter mode enabled)

When these conditions are met, Timer1 continues to count and detect period matches when the device is in Sleep mode. When a match between the timer and the Period register occurs, the T1IF status bit is set. If the T1IE bit is set, and its priority is greater than the current CPU priority, the device wakes from Sleep or Idle mode and executes the Timer1 Interrupt Service Routine.

If the assigned priority level of the Timer1 interrupt is less than, or equal to, the current CPU priority level, the CPU is not awakened and the device enters Idle mode.

5.2 Timer Operation in Idle Mode

When the device enters Idle mode, the system clock sources remain functional and the CPU stops executing code. The Timer1 module can optionally continue to operate in Idle mode.

The setting of the SIDL bit (T1CON<13>) determines whether the Timer1 module stops in Idle mode, or continues to operate normally. If SIDL = 0, the Timer1 Module continues operation in Idle mode. If SIDL = 1, the Timer1 module stops in Idle mode.

6.0 EFFECTS OF VARIOUS RESETS

6.1 Device Reset

All Timer1 registers are forced to their Reset states on a device Reset.

6.2 Power-on Reset (POR)

All Timer1 registers are forced to their Reset states on a Power-on Reset (POR).

6.3 Watchdog Timer Reset

All Timer1 registers are forced to their Reset states on a Watchdog Timer Reset.

© 2016-2018 Microchip Technology Inc.

7.0 RELATED APPLICATION NOTES

This section lists application notes that are related to this section of the manual. These application notes may not be written specifically for the dsPIC33/PIC24 device families, but the concepts are pertinent and could be used with modification and possible limitations. The current application notes related to the Timer1 module are:

Title	Application Note #
No related application notes at this time.	N/A

Note: Please visit the Microchip web site (www.microchip.com) for additional application notes and code examples for the dsPIC33/PIC24 families of devices.

© 2016-2018 Microchip Technology Inc.

8.0 REVISION HISTORY

Revision A (July 2016)

This is the initial version of this document.

Revision B (June 2018)

Removes the Advance Information watermark from the footer.

Notes:

© 2016-2018 Microchip Technology Inc.

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE**.** Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
═ ISO/TS 16949 ═

Trademarks

The Microchip name and logo, the Microchip logo, AnyRate, AVR, AVR logo, AVR Freaks, BitCloud, chipKIT, chipKIT logo, CryptoMemory, CryptoRF, dsPIC, FlashFlex, flexPWR, Heldo, JukeBlox, KeeLoq, Kleer, LANCheck, LINK MD, maXStylus, maXTouch, MediaLB, megaAVR, MOST, MOST logo, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, Prochip Designer, QTouch, SAM-BA, SpyNIC, SST, SST Logo, SuperFlash, tinyAVR, UNI/O, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

ClockWorks, The Embedded Control Solutions Company, EtherSynch, Hyper Speed Control, HyperLight Load, IntelliMOS, mTouch, Precision Edge, and Quiet-Wire are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, BodyCom, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, EtherGREEN, In-Circuit Serial Programming, ICSP, INICnet, Inter-Chip Connectivity, JitterBlocker, KleerNet, KleerNet logo, memBrain, Mindi, MiWi, motorBench, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PowerSmart, PureSilicon, QMatrix, REAL ICE, Ripple Blocker, SAM-ICE, Serial Quad I/O, SMART-I.S., SQI, SuperSwitcher, SuperSwitcher II, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

Silicon Storage Technology is a registered trademark of Microchip Technology Inc. in other countries.

GestIC is a registered trademark of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

© 2018, Microchip Technology Incorporated, All Rights Reserved.

ISBN: 978-1-5224-3241-8

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Australia - Sydney
Tel: 61-2-9868-6733

China - Beijing
Tel: 86-10-8569-7000

China - Chengdu
Tel: 86-28-8665-5511

China - Chongqing
Tel: 86-23-8980-9588

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115

China - Hong Kong SAR
Tel: 852-2943-5100

China - Nanjing
Tel: 86-25-8473-2460

China - Qingdao
Tel: 86-532-8502-7355

China - Shanghai
Tel: 86-21-3326-8000

China - Shenyang
Tel: 86-24-2334-2829

China - Shenzhen
Tel: 86-755-8864-2200

China - Suzhou
Tel: 86-186-6233-1526

China - Wuhan
Tel: 86-27-5980-5300

China - Xian
Tel: 86-29-8833-7252

China - Xiamen
Tel: 86-592-2388138

China - Zhuhai
Tel: 86-756-3210040

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444

India - New Delhi
Tel: 91-11-4160-8631

India - Pune
Tel: 91-20-4121-0141

Japan - Osaka
Tel: 81-6-6152-7160

Japan - Tokyo
Tel: 81-3-6880- 3770

Korea - Daegu
Tel: 82-53-744-4301

Korea - Seoul
Tel: 82-2-554-7200

Malaysia - Kuala Lumpur
Tel: 60-3-7651-7906

Malaysia - Penang
Tel: 60-4-227-8870

Philippines - Manila
Tel: 63-2-634-9065

Singapore
Tel: 65-6334-8870

Taiwan - Hsin Chu
Tel: 886-3-577-8366

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600

Thailand - Bangkok
Tel: 66-2-694-1351

Vietnam - Ho Chi Minh
Tel: 84-28-5448-2100

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-67-3636

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-7289-7561

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820

© 2016-2018 Microchip Technology Inc.