MICROCHIP

# TB062

# Frequently Asked Questions (FAQs) About dsPIC® DSC SMPS Devices

## INTRODUCTION

This document provides answers to Frequently Asked Questions (FAQs) about dsPIC33FJ06GS101/X02 and dsPIC33FJ16GSX02/X04 devices, which are designed to meet the requirements of power conversion applications. These devices were developed using a newer, advanced architecture and include many new features over the dsPIC30F1010/202X devices.

The following are the operating specifications for the dsPIC33FJ06GS101/X02 and dsPIC33FJ16GSX02/X04 devices:

- Voltage range: 3.0V to 3.6V
- Operating frequency: up to 40 MIPS
- Operating temperature range: -40°C to +125°C

| **Note:** | For more information about operating specifications, refer to **Section 24.0 "Electrical Characteristics"** in the *"dsPIC33FJ06GS101/X02 and dsPIC33FJ16GSX02/X04 Data Sheet"* (DS70318), which is available for download from the Microchip web site (www.microchip.com). |
|---|---|

***Question 1: How do I configure the Oscillator for 40 MIPS operation?***

**Answer:** The code sequence to configure the dsPIC® DSC SMPS device for 40 MIPS operation is shown in Example 1. For more information on oscillator configuration, refer to **Section 42. "Oscillator (Part IV)"** (DS70307) in the *"dsPIC33F Family Reference Manual"* or **Section 8.0 "Oscillator"** in the *"dsPIC33FJ06GS101/X02 and dsPIC33FJ16GSX02/X04 Data Sheet"* (DS70318).

### EXAMPLE 1: OSCILLATOR CONFIGURATION

```
// Configure the oscillator to operate the device at 40 MHz
// The Fast RC (FRC) internal oscillator runs at a nominal frequency of 7.37 MHz
// FOSC = Fin * M/(N1 * N2), FCY = FOSC/2
// FOSC = 7.37 * (43)/(2 * 2) = 80 MHz for FOSC, FCY = 40 MHz

// In order to configure the device to operate at 40 MHz, configure the PLL prescaler,
// PLL postscaler, and PLL divisor
PLLFBD = 41;                       // M = PLLFBD + 2
CLKDIVbits.PLLPOST = 0;            // N1 = 2
CLKDIVbits.PLLPRE = 0;             // N2 = 2

__builtin_write_OSCCONH(0x01);     // New oscillator FRC w/ PLL
__builtin_write_OSCCONL(0x01);     // Enable clock switch as per dsPIC oscillator start-up
                                   // guidelines. For more information, refer to
                                   // Section 42. "Oscillator (Part IV)" (DS70307) in the
                                   // "dsPIC33F Family Reference Manual".


while(OSCCONbits.COSC! = 0b001);   // Wait for the new Oscillator to become FRC w/ PLL
while(OSCCONbits.LOCK! = 1);       // Wait for PLL to lock
```

***Question 2: How do I configure the Pulse-Width Modulator (PWM) module?***

**Answer:** Depending on the device variant, the dsPIC DSC SMPS device family offers two, three or four PWM generators on a single device**.** Each PWM generator has two output pins, PWMxH and PWMxL, which can be configured as independent PWM outputs. Therefore, up to eight independent PWM signals can be generated.

For each single PWM channel, the user application can select a common time base (i.e., all the PWM channels share a common time base) or individual time base (i.e., each PWM channel has an independent time base).

Example 2 illustrates the code sequence to initialize the PWM peripheral. The code configures the minimum number of registers required to enable the PWM module. For more information, refer to **Section 43. “High-Speed PWM”** (DS70323) in the “*dsPIC33F Family Reference Manual*”.

### EXAMPLE 2: PWM CONFIGURATION

```
// PTCON: PWM Time Base Control register
// bit15 PTCON: PTEN enables the operation of the PWM. It must be set to ‘0’
// during the initialization phase; it will be set at the end, starting the
// peripheral operation

PTCON = 0;

// PTCON2: PWM Clock Divider Select register
// When PTCON2 is set to ‘0’, it ensures the maximum speed and time resolution of the PWM

PTCON2 = 0;

// PTPER: PWM Master Time Base register
// When selected as the period generator, this register contains the value of
// the PWM period. The PWM period can be computed using the following equation:

// PWM_PERIOD = (1.0e9/(1.04 * PWM_FREQUENCY)), where PWM_FREQUENCY is in Hz

PTPER = PWM_PERIOD;

SEVTCMP = 0;

// MDC: PWM Master Duty Cycle register
// It must be loaded with a non-zero value if one or more PWM channels use
// this register as their duty cycle value

// PWMCONx: PWM Control register
// It contains the following major bits:
// ITB: Independent Time Base Mode bit
//  1 = PHASEx/SPHASEx register provides the time base period for this PWMx generator
//  0 = PTPER register provides timing for this PWMx generator

// MDCS: Master Duty Cycle Register Select bit
//  1 = MDC register provides duty cycle information for this PWMx generator
//  0 = PDCx and SDCx registers provide duty cycle information for this PWMx generator
```

### EXAMPLE 2: PWM CONFIGURATION (CONTINUED)

```
// DTC<1:0>: Dead Time Control bits

//  00 = Positive dead time actively applied for all output modes
//  01 = Negative dead time actively applied for all output modes
//  10 = Dead time function is disabled
//  11 = Reserved

// IUE: Immediate Update Enable bit
//  1 = Updates to the active MDC/PDCx/SDCx registers are immediate
//  0 = Updates to the active PDCx registers are synchronized to the PWM time base

// In this example, the PTPER register is used as the period register
// Each channel will have a different duty
// Positive dead time is enabled
// Duty value is updated in synchronism with the PWM time base

PWMCON1bits.ITB = 0;
PWMCON1bits.MDCS = 0;
PWMCON1bits.DTC = 0;
PWMCON1bits.IUE = 0;

// PDCx: PWM Generator Duty Cycle register
// It selects the duty cycle
// In this example, the duty is half the PWM period

PDC1 = (PWM_PERIOD/2);

// SDCx: PWM Secondary Duty Cycle register
// The secondary duty cycle is not used because the PWM channel is programmed to
// output one (complimentary) signal

// (S)PHASEx: PWM Primary Phase Shift register
// In this example, one single PWM output signal is used without any phase shift

PHASE1 = 0;

SPHASE1 = 0;

// DTRx: PWM Dead Time register
// Dead time is useful to avoid shoot-through in the power switches
// The highest resolution is 1.04 ns
// Two registers are available to select different rise and fall dead
// times: DTRx and ALTDTRx
// In this example, the inserted dead times are close to 100 ns and 150 ns respectively

DTR1 = 100;

// ALTDTRx: PWM Alternate Dead Time register

ALTDTR1 = 150;
```

### EXAMPLE 2: PWM CONFIGURATION (CONTINUED)

```
// IOCONx: PWM I/O Control register
// This register controls a number of different functionality of the I/O
// pins of the PWM peripheral. It includes the following major bits:

// PENH: PWMxH Output Pin Ownership bit
//  1 = PWM module controls PWMxH pin
//  0 = GPIO module controls PWMxH pin

// PENL: PWMxL Output Pin Ownership bit
//  1 = PWM module controls PWMxL pin
//  0 = GPIO module controls PWMxL pin

// POLH: PWMxH Output Pin Polarity bit
//  1 = PWMxH pin is active-low
//  0 = PWMxH pin is active-high

// POLL: PWMxL Output Pin Polarity bit
//  1 = PWMxL pin is active-low
//  0 = PWMxL pin is active-high

// PMOD<1:0>: PWM # I/O Pin Mode bits
//  00 = PWM I/O pin pair is in the Complementary Output mode
//  01 = PWM I/O pin pair is in the Redundant Output mode
//  10 = PWM I/O pin pair is in the Push-Pull Output mode
//  11 = PWM I/O pin pair is in the True Independent Output mode

// In this example, the pins are driven by the PWM peripheral (PENH/L = 1)
// These pins are active high (POLL/H = 0), and are configured in
// complimentary mode (PMOD = 0);

IOCON1 = 0;
IOCON1bits.PENH = 1;
IOCON1bits.PENL = 1;
IOCON1bits.POLH = 0;
IOCON1bits.PMOD = 0;

// FLCONx: PWM Fault Current-Limit Control register

// FLTMOD<1:0>: Fault Mode bits for PWM Generator #
//  00 = The selected Fault source forces PWMxH, PWMxL pins to FLTDAT
//       values (latched condition)
//  01 = The selected Fault source forces PWMxH, PWMxL pins to FLTDAT
//       values (cycle)
//  10 = Reserved
//  11 = Fault input is disabled
// This register enables the fault operation and current limit
// operation of the PWM. None of these features are used in this example.
// FCLCON1bits.FLTMOD = 3;
```

***Question 3:*** ***How do I configure the PWM resolution (e.g., 4.16 ns) to optimize the power consumption?***

**Answer:** The input clock of the High-Speed PWM module has prescaler (divider) options of 1:1 to 1:64, which can be selected using the PWM Input Clock Prescaler Select (PCLKDIV) bits in the PWM Clock Divider Select (PTCON2<2:0>) register. The prescaled value also reflects the PWM resolution, which helps in reducing the power consumption of the High-Speed PWM module. The prescaled clock is an input to the PWM clock control logic block. The maximum clock speed provides the duty cycle, dead time, phase shift and period resolution of 1.04 ns. For example:

- If a prescaler option of 1:2 is selected, the PWM duty cycle and period resolution can be set at 2.08 ns. Thereby, the power consumption of High-Speed PWM module would be reduced by approximately 50 percent of the maximum speed operation.
- If a prescaler option of 1:4 is selected, the PWM duty cycle and period resolution can be set at 4.16 ns. Thereby, the power consumption of the High-Speed PWM module would be reduced by approximately 75 percent of the maximum speed operation**.**

***Question 4:*** ***How do I configure the Analog Comparator as a fault/current-limit source in a dsPIC DSC SMPS device?***

**Answer:** The following steps help to configure an analog comparator as a fault source in the dsPIC DSC SMPS device:

1. Map the analog comparator to a virtual pin.
2. On the same virtual pin, map one of the Current-Limit/Fault Signal Source (FCLCONx Register) :
   - RPOR16 (bit names: RP32R and RP33R)
   - RPOR17 (bit names: RP34R and RP35R)

   The virtual pins are not available as I/O pins, but are listed on the device. Example 3 illustrates the code sequence that shows how to use the comparators as a fault source.
3. Map the fault to any of the fault sources listed in the FCLCONx register of the PWM module. Here, this fault is mapped to Fault1 (FLT1).

For more information, refer to the code example **“CE157 – Cycle-by-Cycle PWM Fault ”** available on the Microchip web site (www.microchip.com).

### EXAMPLE 3: FAULT CONFIGURATION

```
RPINR29bits.FLT1R = 32;       // Select FLT1 as an input for the trigger source
                              // and connect to output of the Comparator
RPOR16bits.RP32R = 0b101000;  // Output Analog Comparator 2 to the “virtual pin” 32
```

***Question 5:*** ***How do I configure the PWM and the Comparator for cycle-by-cycle control?***

**Answer:** The analog comparator can operate at high-speed with a typical delay of 20 ns. The negative input of the comparator is always connected to the DAC circuit. The positive input of the Comparator is connected to an analog multiplexer, which selects the desired source pin. To use the Comparator output as one of the fault sources, follow these steps:

1. Remap the Comparator output to a virtual pin.
2. Remap one of the Current-Limit/Fault Signal Source (FCLCONx Register) as an input to the same virtual pin.
3. Configure this virtual pin as the source for the fault/current-limit feature.
4. Enable the fault/current-limit in Cycle-by-Cycle mode.

For more information, refer to the code example **“CE157 – Cycle-by-Cycle PWM Fault”** available on the Microchip web site (www.microchip.com).

***Question 6:*** ***How do I configure the PWM module when implementing the prominent topologies such as Push-Pull, Complementary, Redundant and True Independent mode?***

**Answer:** The following guidelines help to configure various PWM operating modes when implementing the prominent topologies:

- Push-Pull Output mode: This mode can be configured by setting the PMOD bits in the PWM I/O Control (IOCONx) register to '10'.
- Complementary Output mode: This mode can be configured by setting the PMOD bits in the PWM I/O Control (IOCONx) register to '00'.
- Redundant Output mode: This mode can be configured by setting the PMOD bits in the PWM I/O Control (IOCONx) register to '01'.
- True Independent Output mode: This mode can be configured by setting the PMOD bits in the PWM I/O Control (IOCONx) register to '11'.

Table 1 provides the PMOD bit selection for the different topologies and configurations.

For more information, refer to sections **43.9 "PWM Operating Modes"** and **43.16 "Application Information"** in **Section 43. "High-Speed PWM"** (DS70323) of the "*dsPIC33F Family Reference Manual*".

**TABLE 1: PMOD BIT SELECTION**

| Item | Topology[1] | Configuration | PMOD bit Setting |
|---|---|---|---|
| 1 | Flyback Converter | True Independent Output mode/Redundant Output mode | 11/01 |
| 2 | Boost/PFC Converter | True Independent Output mode/Redundant Output mode | 11/01 |
| 3 | Interleaved PFC Converter | True Independent Output mode with Master Time Base | 11 |
| 4 | Forward Converter | True Independent Output mode/Redundant Output mode | 11/01 |
| 5 | Double Ended Forward Converter | True Independent Output mode/Redundant Output mode | 11/01 |
| 6 | Active Clamp Forward Converter | Complementary Output mode | 00 |
| 7 | LLC Half-Bridge Series Converter | Complementary Output mode | 00 |
| 8 | Half-Bridge Converter | Push-Pull Output mode | 10 |
| 9 | Push-Pull Converter | Push-Pull Output mode | 10 |
| 10 | Full-Bridge Converter | Push-Pull Output mode | 10 |
| 11 | Phase Shifted Full-Bridge Converter | Complementary Output mode | 00 |
| 12 | Single-Phase Synchronous Buck Regulator | Complementary Output mode | 00 |
| 13 | Multi-Phase Buck Regulator | True Independent Output Mode with Master Time Base | 11 |

**Note 1:** The listed topologies can be configured both in the voltage and in the current (i.e., Average and Peak Current) mode control.

***Question 7:*** ***How do I exit from a latched Fault?***

**Answer:** In Latched mode, the PWM outputs follow the states defined in the FLTDAT bits in the IOCONx registers when the fault pin is asserted. The PWM outputs remain in this state until the fault pin is deasserted, and the corresponding interrupt flag is cleared in software. When both actions occur, the PWM outputs return to the normal operation at the beginning of the next PWM cycle boundary.

For more information, refer to section **43.10.4 "Fault Exit"** in **Section 43. "High-Speed PWM"** (DS70323) of the "*dsPIC33F Family Reference Manual*", and the code example **"CE158_Latch_Fault"**, which are available from the Microchip web site (www.microchip.com).

***Question 8:*** ***How do I implement the PWM external synchronization feature?***

**Answer:** The master time base can be synchronized with the external synchronization signal via the master time base synchronization signal (SYNCI1/SYNCI2). The synchronization source (SYNCI1 and SYNCI2) can be selected using the Synchronous Source Selection (SYNCSRC<1:0>) bits in the PWM Time Base Control (PTCON<5:4>) register. The Synchronize Input Polarity (SYNCPOL) bit in the PWM Time Base Control (PTCON<9>) register selects the rising or falling edge of the synchronization pulse, which resets the timer (PMTMR). The external synchronization feature can be enabled or disabled using the External Time Base Synchronization Enable (SYNCEN) bit in the PWM Time Base Control (PTCON<7>) register. The pulse-width of the external synchronization signal (SYNCI1/SYNCI2) should be more than 200 ns to ensure the reliable detection by the master time base.

For more information about the external synchronization feature, refer to section **43.5.6 “Time Base Synchronization”** in **Section 43. “High-Speed PWM”** (DS70323) of the “*dsPIC33F Family Reference Manual”*.

***Question 9:*** ***How do I implement the Current Reset/Variable Frequency control using the PWM and the Comparator?***

**Answer:** The Current Reset PWM is a variable frequency mode, where the actual PWM period is less than or equal to the specified period value. The independent time base is externally reset after some time the PWM signal is deasserted. This is known as Constant PWM On-time mode. To operate in PWM Current Reset, the PWM generator should be in Independent Time Base mode. If an external reset signal is not received, the PWM period uses the PHASEx register value by default.

For more information, refer to section **43.16.5 “Current Reset PWM”** in **Section 43. “High-Speed PWM”** (DS70323) of the “*dsPIC33F Family Reference Manual*”.

Example 4 illustrates the code sequence on how to implement the Variable Frequency mode. For more details, refer to the code example **“CE168_Current_Reset_Variable_Frequency”** available on the Microchip web site (www.microchip.com).

### EXAMPLE 4: CODE EXAMPLE

```
IOCON1bits.PENH = 1;          // PWM1H is controlled by PWM module
IOCON1bits.PENL = 0;          // PWM1L is controlled by I/O module
IOCON1bits.PMOD = 3;          // PWM pair operates in Independent mode

IOCON1bits.OVRDAT = 0b00;     // PWM1H/L output is low and override is enabled
IOCON1bits.OVRENH = 0;        // PWM provides data for PWM1H

PWMCON1bits.IUE = 0;          // Disable immediate duty cycle updates
PWMCON1bits.ITB = 1;          // Select Independent Time Base mode
PWMCON1bits. XPRES = 1;       // Current Reset mode

FCLCON1bits.CLSRC = 0;        // FLT1 to reset timer base (FLT1 remapped to Comparator 2 output)
FCLCON1bits.CLPOL = 1;        // Current limit signal is active low

RPINR29bits.FLT1R = 32;       // Select FLT1 as input for current limit trigger source
RPOR16bits.RP32R = 0b101000;  // Output Analog Comparator 2 to “virtual pin” 32
CMPDAC2bits.CMREF = 20;       // The period timer is reset when non-inverting pin voltage of
                              // Analog Comparator reaches this value. For this example, the
                              // reference inverting pin voltage will be 20 * 1.65/1023 volts
CMPCON2bits.INSEL = 0b01;     // CMP2B is selected
CMPCON2bits.EXTREF = 0;       // Internal reference for DAC
CMPCON2bits.RANGE = 1;        // In this example, Maximum DAC output = VDD/2 = 1.65V
CMPCON2bits.CMPPOL = 0;       // Comparator output is not inverted
```

***Question 10: How do I configure the Analog-to-Digital Converter (ADC) for maximum speed operation?***

**Answer:** The auxiliary clock generator can be used to generate the clock for the ADC module independent of the system clock. The Primary Oscillator Clock (POSCCLK) and the Internal FRC Clock (FRCCLK) can be used with an auxiliary PLL to obtain the Auxiliary Clock (ACLK). The auxiliary PLL has a fixed 16x multiplication factor. To configure the ADC for maximum speed, enable the slow clock divider (ADCONbits.SLOWCLK = 1), and select the Analog-to-Digital (A/D) conversion clock divider value of 6 (ADCONbits.ADCS = 5). This will provide the maximum ADC speed of 4 Msps for two SAR devices and 2 Msps for a single SAR device.

### EXAMPLE 5: CODE EXAMPLE

```
// Input clock: FRCCLK
// Multiply by 16, FPWM/ADC = 7.37 MHz * 16 = 117.92 MHz
// ACLKCON: Auxiliary Clock Divisor Control register

ACLKCON = 0;
ACLKCONbits.FRCSEL = 1;                  // Source clock is FRC
ACLKCONbits.SELACLK = 1;                 // Primary auxiliary oscillator
ACLKCONbits.APSTSCLR = AUX_OSC_DIVIDE;   // Divide factor
ACLKCONbits.ENAPLL = TRUE;               // Enable auxiliary clock generation

// loop waiting for auxiliary PLL lock
while(!ACLKCONbits.APLLCK)
{
};

The instructions for the clock selection are:
ADCONbits.SLOWCLK = 1;
ADCONbits.ADCS = 5;                      // FADC
                                         // Clock divider is set up for FADC/6
                                         // TAD = 41.66 ns
                                         // For simultaneous sampling, the total
                                         // conversion time for one pair is 0.625 μs
```

***Question 11: How do I trigger an ADC conversion?***

**Answer:** To start the ADC operation, a trigger should be generated using any of the possible trigger sources listed in Example 6. The trigger source is selected by programming the Trigger Source Selection (TRGSRCx) bits in the A/D Convert Pair Control (ADCPCx) register.

### EXAMPLE 6: TRIGGER SOURCES

The available trigger sources are as follows:

`11111` = Timer2 period match<br>
•<br>
•<br>
•<br>
`11011` = Reserved<br>
`11010` = PWM Generator 4 current limit ADC trigger<br>
`11001` = PWM Generator 3 current limit ADC trigger<br>
`11000` = PWM Generator 2 current limit ADC trigger<br>
`10111` = PWM Generator 1 current limit ADC trigger<br>
`10110` = Reserved<br>
•<br>
•<br>
•<br>
`10010` = Reserved<br>
`10001` = PWM Generator 4 current limit ADC trigger<br>
`10000` = PWM Generator 3 current limit ADC trigger<br>
`01111` = PWM Generator 2 current limit ADC trigger<br>
`01110` = PWM Generator 1 current limit ADC trigger<br>
`01101` = Reserved<br>
`01100` = Timer1 period match<br>
•<br>
•<br>
•<br>
`01000` = Reserved<br>
`00111` = PWM Generator 4 current limit ADC trigger<br>
`00110` = PWM Generator 3 current limit ADC trigger<br>
`00101` = PWM Generator 2 current limit ADC trigger<br>
`00100` = PWM Generator 1 current limit ADC trigger<br>
`00011` = PWM special event trigger selected<br>
`00010` = Global software trigger selected<br>
`00001` = Individual software trigger selected

***Question 12:*** ***How do I set up the ADC clock speed/sample rate?***

**Answer:** The ADC requires 14 * TAD to complete a conversion. Therefore, to achieve the 2 Msps sampling rate, the FADC should be set to 24 MHz. For information about setting up the ADC clock speed, refer to section **44.3.1 "ADC Clock Selection"** in **Section 44. "High-Speed 10-Bit ADC"** (DS70321) of the "*dsPIC33F Family Reference Manual*".

If SLOWCLK = `0`, the primary PLL is used as the ADC clock source. FVCO is the output of the Primary oscillator.

### EQUATION 1:

$$F_{ADC} = \frac{F_{VCO}}{N}$$

Where *N* = ADC clock divider ratio, which is selected using the ADCS bits in ADCON register.

To operate the ADC at 2 Msps and to set FADC to 24 MHz, choose the ADCS<2:0> bits setting based on the FVCO frequency.

If SLOWCLK = `1`, the auxiliary PLL is used as the ADC clock source. The ACLK is the output of the auxiliary oscillator.

### EQUATION 2:

$$F_{ADC} = \frac{ACLK}{N}$$

**Note:** The ADC clock divider ratio is selected using the ADCS bits in ADCON register.

To operate the ADC at 2 Msps and to set FADC to 24 MHz, choose the ADCS<2:0> bits setting based on the ACLK frequency.

***Question 13:*** ***How do I configure the Idle mode?***

**Answer:** The Idle mode halts the CPU and code execution, but allows the peripheral modules to continue their operation. For information about Idle mode configuration, refer to **Section 9.0 "Power-Saving Features"** in the *"dsPIC33FJ06GS101/X02 and dsPIC33FJ16GSX02/X04 Data Sheet"* (DS70318).

### EXAMPLE 7: `PWRSAV` INSTRUCTION SYNTAX

```
PWRSAV #IDLE_MODE; //Put the device into Idle mode
```

***Question 14:*** ***How do I configure the Sleep mode?***

**Answer:** The Sleep mode stops the clock operation and halts the code execution. For information on Sleep mode configuration, refer to **Section 9.0 "Power-Saving Features"** in the *"dsPIC33FJ06GS101/X02 and dsPIC33FJ16GSX02/X04 Data Sheet"* (DS70318).

### EXAMPLE 8: `PWRSAV` INSTRUCTION SYNTAX

```
PWRSAV #SLEEP_MODE; // Put the device into Sleep mode
```

***Question 15:  How do I configure the DOZE mode?***

**Answer:** The DOZE mode is a simple, effective and alternative method to reduce the power consumption while the device is still executing the code. In this mode, the system clock continues to operate from the same source and at the same speed. The peripheral modules continue to be clocked at the same speed, while the CPU clock speed is reduced. Synchronization between the two clock domains is maintained, allowing the peripherals to access the SFRs while the CPU executes code at a slower rate.

### EXAMPLE 9: DOZE MODE CONFIGURATION

```
CLKDIVbits.DOZE = 4; //DOZE ratio is FCY/4
CLKDIVbits.DOZEN = 1; // DOZE enable
```

## REVISION HISTORY

### Revision A (November 2009)

This is the initial released version of this document.

**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.

**Trademarks**

The Microchip name and logo, the Microchip logo, dsPIC, KEELOQ, KEELOQ logo, MPLAB, PIC, PICmicro, PICSTART, rfPIC and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

FilterLab, Hampshire, HI-TECH C, Linear Active Thermistor, MXDEV, MXLAB, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Application Maestro, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, dsSPEAK, ECAN, ECONOMONITOR, FanSense, HI-TIDE, In-Circuit Serial Programming, ICSP, Mindi, MiWi, MPASM, MPLAB Certified logo, MPLIB, MPLINK, mTouch, Octopus, Omniscient Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICkit, PICtail, PIC$^{32}$ logo, REAL ICE, rfLAB, Select Mode, Total Endurance, TSHARC, UniWinDriver, WiperLock 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.

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

♻ Printed on recycled paper.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
== ISO/TS 16949:2002 ==

*Microchip received ISO/TS-16949:2002 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.*

# 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://support.microchip.com
Web Address:
www.microchip.com

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

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

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

**Cleveland**
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

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

**Detroit**
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

**Kokomo**
Kokomo, IN
Tel: 765-864-8360
Fax: 765-864-8387

**Los Angeles**
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

**Santa Clara**
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

**Toronto**
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

## ASIA/PACIFIC

**Asia Pacific Office**
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

**Australia - Sydney**
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

**China - Beijing**
Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

**China - Chengdu**
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

**China - Hong Kong SAR**
Tel: 852-2401-1200
Fax: 852-2401-3431

**China - Nanjing**
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

**China - Qingdao**
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

**China - Shanghai**
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

**China - Shenyang**
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

**China - Shenzhen**
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

**China - Wuhan**
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

**China - Xiamen**
Tel: 86-592-2388138
Fax: 86-592-2388130

**China - Xian**
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

**China - Zhuhai**
Tel: 86-756-3210040
Fax: 86-756-3210049

## ASIA/PACIFIC

**India - Bangalore**
Tel: 91-80-3090-4444
Fax: 91-80-3090-4080

**India - New Delhi**
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

**India - Pune**
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

**Japan - Yokohama**
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

**Korea - Daegu**
Tel: 82-53-744-4301
Fax: 82-53-744-4302

**Korea - Seoul**
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

**Malaysia - Kuala Lumpur**
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

**Malaysia - Penang**
Tel: 60-4-227-8870
Fax: 60-4-227-4068

**Philippines - Manila**
Tel: 63-2-634-9065
Fax: 63-2-634-9069

**Singapore**
Tel: 65-6334-8870
Fax: 65-6334-8850

**Taiwan - Hsin Chu**
Tel: 886-3-6578-300
Fax: 886-3-6578-370

**Taiwan - Kaohsiung**
Tel: 886-7-536-4818
Fax: 886-7-536-4803

**Taiwan - Taipei**
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

**Thailand - Bangkok**
Tel: 66-2-694-1351
Fax: 66-2-694-1350

## EUROPE

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

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

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

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

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

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

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

**UK - Wokingham**
Tel: 44-118-921-5869
Fax: 44-118-921-5820

03/26/09