



---

---

## Section 24. Programming and Diagnostics

---

---

### HIGHLIGHTS

This section of the manual contains the following topics:

|      |  |       |
|------|--|-------|
| 24.1 | Introduction .....                           | 24-2  |
| 24.2 | In-Circuit Serial Programming™ (ICSP™) ..... | 24-3  |
| 24.3 | Enhanced ICSP .....                          | 24-6  |
| 24.4 | JTAG Boundary Scan .....                     | 24-7  |
| 24.5 | Related Application Notes .....              | 24-16 |
| 24.6 | Revision History .....                       | 24-17 |

**Note:** This family reference manual section is meant to serve as a complement to device data sheets. Depending on the device variant, this manual section may not apply to all dsPIC33E/PIC24E devices.

Please consult the note at the beginning of the “**Special Features**” 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>

## 24.1 INTRODUCTION

dsPIC33E/PIC24E devices provide a complete range of programming and diagnostic features that can increase the flexibility of any application using them. These features allow system designers to include:

- Simplified field programmability using two-wire interfaces
- Enhanced debugging capabilities
- Boundary scan testing for device and board diagnostics

dsPIC33E/PIC24E devices incorporate three different programming and diagnostic modalities that provide a range of functions useful to the application developer. They are summarized in [Table 24-1](#).

**Table 24-1: Comparison of dsPIC33E/PIC24E Programming and Diagnostic Features**

| Feature   | Interface                  | Device Integration   | Functions                               |
|---|----------------------------|--|---|
| In-Circuit Serial Programming™ (ICSP™) programming method | PGCx and PGDx pins         | Integrated with device core                                  | Programming, debugging                  |
| Enhanced ICSP programming method                          | PGCx and PGDx pins         | Hardware integrated with device core; firmware-based control | Programming                             |
| Joint Test Action Group (JTAG)                            | TDI, TDO, TMS and TCK pins | Peripheral to device core; partly integrated with I/O logic  | Boundary Scan Testing (BST) diagnostics |

### 24.2 IN-CIRCUIT SERIAL PROGRAMMING™ (ICSP™)

The In-Circuit Serial Programming (ICSP) programming capability is Microchip's proprietary process for microcontroller programming in the target application. Originally introduced for 8-bit PIC16 devices, this method is used for virtually all Microchip microcontrollers. ICSP is the most direct method to program the device, whether the controller is embedded in a system or loaded into a device programmer.

#### 24.2.1 ICSP Interface

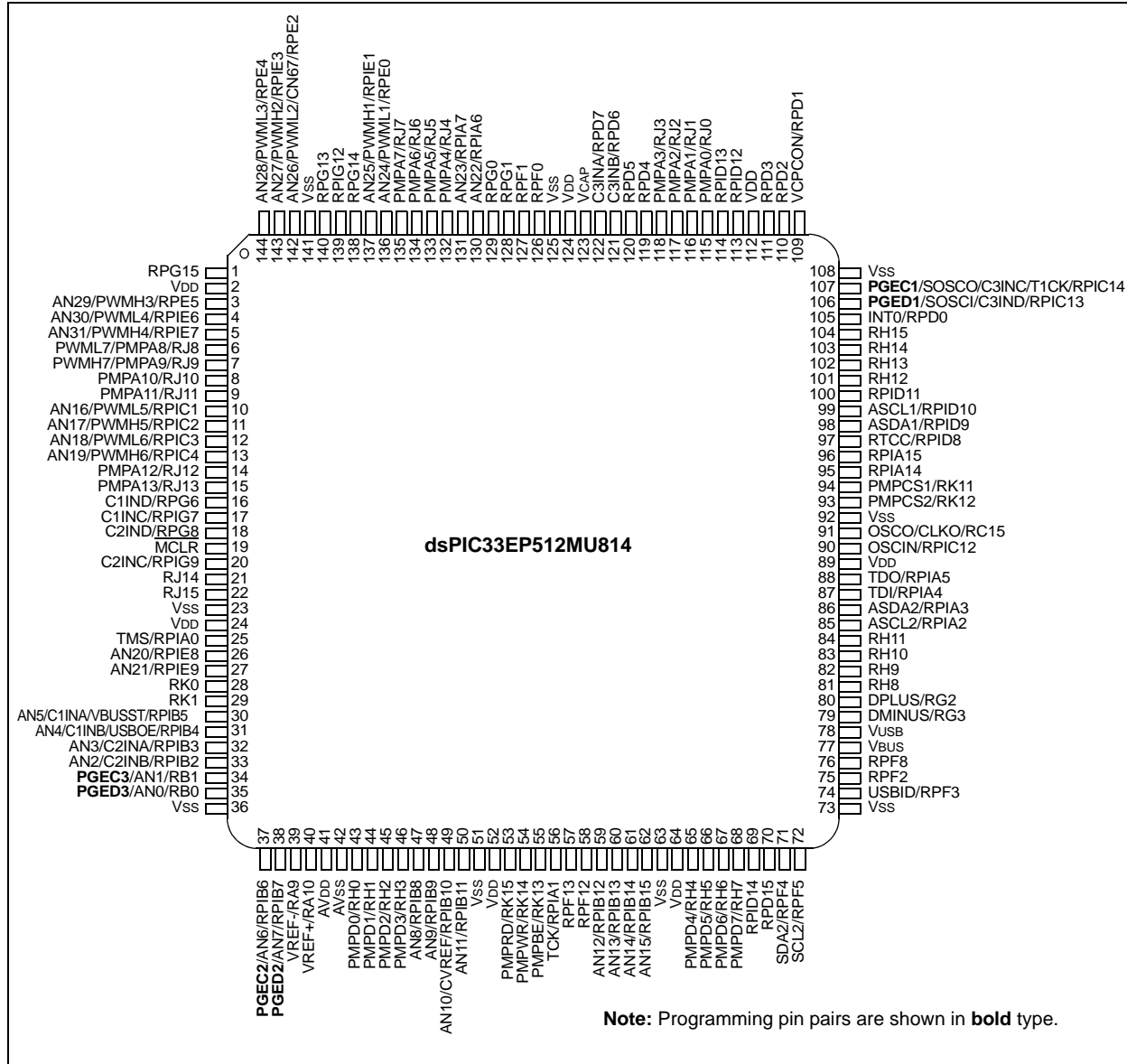
The ICSP method uses a two-pin communication interface. The Programming Data (PGD) pin functions as both an input and an output, allowing programming data to be read in and device information to be read out on command. The Programming Clock (PGC) pin clocks in data and controls the overall process.

Most dsPIC33E/PIC24E devices have more than one pair of PGC and PGD pins; these are multiplexed with other I/O or peripheral functions as shown in [Figure 24-1](#). Individual ICSP pin pairs are indicated by number (e.g., PGC1/PGD1, etc.) and are generically referred to as PGCx and PGDx. The multiple PGCx/PGDx pairs provide additional flexibility in system design by allowing you to incorporate ICSP on the pair of pins least constrained by the circuit design. All PGCx and PGDx pins are functionally tied together and behave identically. Any one pair can be used for successful device programming. The only limitation is that both pins from the same pair must be used.

In addition to the PGCx and PGDx pins, ICSP requires that all voltage supply and ground pins on the device must be connected. The  $\overline{\text{MCLR}}$  pin, which is used with PGCx to enter and control the programming process, must also be connected to the programming device.

# dsPIC33E/PIC24E Family Reference Manual

Figure 24-1: Example of Programming Pin Pairs on a dsPIC33E Device



## 24.2.2 ICSP Operation

ICSP mode uses a combination of internal hardware and external control to program the target device. Programming data and instructions are provided on the PGD pin. A special set of 4-bit commands, combined with standard dsPIC33E/PIC24E instructions, controls the overall process of writing to the program memory. The PGD pin also returns data to the external programmer in response to queries.

The programming process is controlled by manipulating the PGC and  $\overline{\text{MCLR}}$  pins. Entry into and exit from ICSP mode involves applying (or removing) voltage to  $\overline{\text{MCLR}}$ , while supplying a code sequence to PGD and a clock to PGC.

Any one of the PGCx/PGDx pairs can be used for programming. During programming, the clock train on PGC is also used to indicate the difference between 4-bit commands, programming control commands, and payload data to be programmed.

The internal process is regulated by a state machine built into the dsPIC33E/PIC24E core logic; however, overall control of the process must be provided by the external programming tool. Microchip programming tools, such as the MPLAB® PM3 Universal Device Programmer (used with the MPLAB® IDE development software), include the necessary hardware and algorithms to manage the programming process for dsPIC33E/PIC24E devices. Users who are interested in a more detailed description, or who are considering designing their own programming interface for a dsPIC33E/PIC24E device, should refer to the “dsPIC33E/PIC24E Flash Programming Specification” (DS70619).

### 24.2.3 ICSP and In-Circuit Debugging

The ICSP method also provides a hardware channel for in-circuit debugging, which allows external control of software debugging. Using the appropriate hardware interface and software environment, you can force the device to single-step through its code, track the actual content of multiple registers, and set software breakpoints.

To use in-circuit debugging, an external system must load a debug executive program into the microcontroller. This task is handled automatically by many debugging tools, such as MPLAB REAL ICE™ and MPLAB ICD 3. For dsPIC33E/PIC24E devices, the program is loaded into the executive program memory in the configuration memory space. Although memory is implemented and code can be executed from these locations, the executive memory space is not available to the user application during normal operating modes. For details, refer to the “dsPIC33E/PIC24E Flash Programming Specification” (DS70619).

Because of the memory location, use of the debug executive has no impact on the size of the application being examined. The executive memory space allows use of the entire program memory for program code, without needing to reserve space for application debugging. In addition, its use means that the program memory content in normal and debug states is identical, which helps to simplify troubleshooting.

Depending on the particular dsPIC33E/PIC24E device, one or more ICSP ports can be used for programming. However, only one of these ICSP ports can be used for in-circuit debugging. Use the following process to select which part to activate for debugging via your MPLAB IDE setup:

1. In MPLAB IDE, select the *Configure > Configuration Bits* menu to display the Configurations Bits window.
2. In the Configuration Bits window, select the appropriate debug pair setting under the Comm Channel Select Category.

**Note:** For details on the configuration memory space, refer to the “dsPIC33E/PIC24E Flash Programming Specification” (DS70619).

The dsPIC33E and PIC24E devices contain a new and improved Debug module that expands their debugging capabilities. Some of the new capabilities (provided with MPLAB X IDE in conjunction with MPLAB REAL ICE and MPLAB ICD 3) are:

- The ability to non-intrusively probe and/or modify internal registers and memory locations at run-time without halting the CPU execution
- Up to six data sources can be monitored and streamed out to the host computer whenever they are modified without any impact on CPU operation. They can be examined at run-time either via the watch window or plotted graphically (MPLAB REAL ICE only).
- Application input/output. The ability to exchange application data with the CPU without halting CPU execution and with minimum impact on the CPU.
- The capability of the user’s application to stream out data to the host computer via a 16-word hardware FIFO (MPLAB REAL ICE only)
- Up to six complex breakpoints that can be set up to monitor address and/or data and events
- Advanced modes for breakpoints such as: Break on Address Greater Than or Equal to

All of these capabilities use the same ICD interface pins (PGC and PGD), which are used for programming. No external components are needed.

## 24.3 ENHANCED ICSP

The Enhanced ICSP protocol is an extension of the ICSP method. Enhanced ICSP uses the same physical interface as the original, but changes the location and execution of programming control.

ICSP mode uses a simple state machine to control each step of the programming process; however, the state machine is controlled by an external programmer. In contrast, Enhanced ICSP uses an on-board bootloader, known as the programming executive, to manage the programming process. While overall device programming is still overseen by an external programmer, the programming executive manages most of the things that must be directly controlled by the programmer in standard ICSP.

The programming executive implements its own command set, wider in range than the original ICSP, that can directly erase, program, and verify the microcontroller's program memory. This avoids the need to repeatedly run ICSP command sequences to perform simple tasks. As a result, Enhanced ICSP mode can program or reprogram a device more quickly than ICSP mode.

Like the in-circuit debug executive, the programming executive does not reside in the user application program memory space. It is also loaded into the executive program memory. Since the debugger and Enhanced ICSP executives both use this memory space, in-circuit debugging is not available while Enhanced ICSP mode is being used for programming.

The programming executive is not preprogrammed into dsPIC33E/PIC24E devices. If you need Enhanced ICSP, you must use standard ICSP to program the executive to the executive memory space. You can set this up directly in your software, or automatically using a compatible Microchip programming system.

For additional information on Enhanced ICSP and the programming executive, refer to the *"dsPIC33E/PIC24E Flash Programming Specification"* (DS70619).

## 24.4 JTAG BOUNDARY SCAN

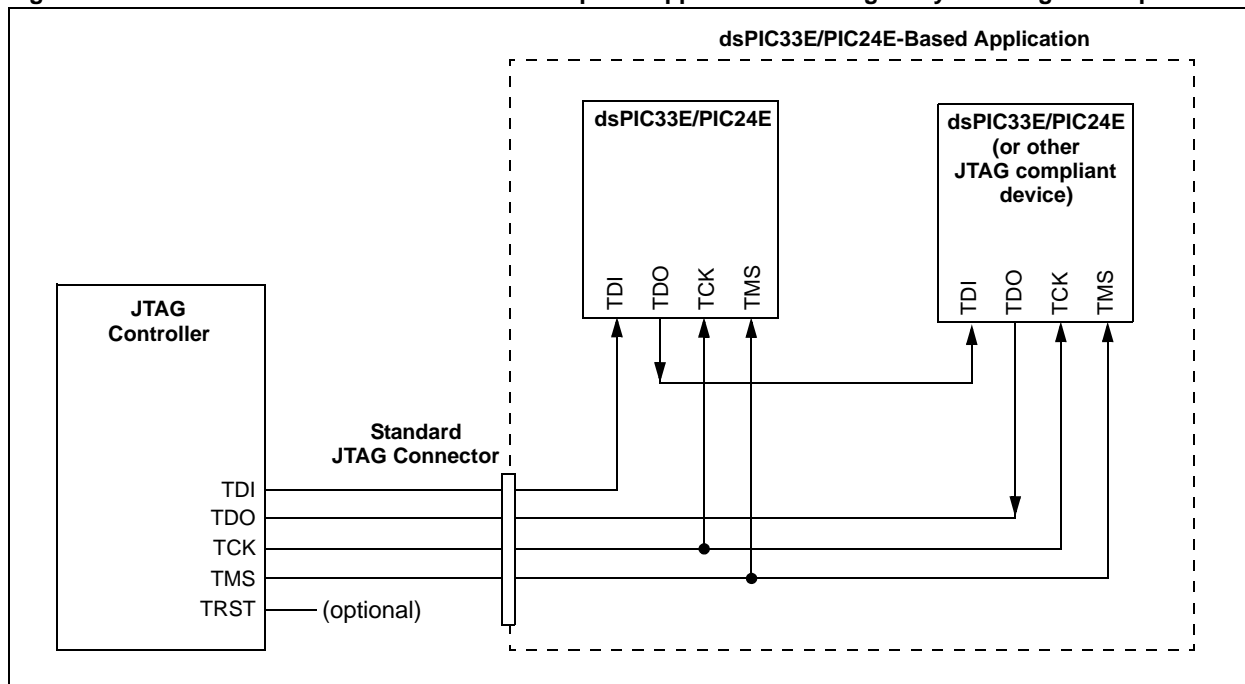
As the complexity and density of board designs increase, testing electrical connections between the components on fully-assembled circuit boards poses many challenges. To address these challenges, the Joint Test Action Group (JTAG) developed a method for boundary scan testing that was later standardized as IEEE 1149.1-2001, “IEEE Standard Test Access Port and Boundary Scan Architecture”.

The JTAG boundary scan method adds a shift register stage adjacent to each of the component’s I/O pins, which permits signals at the component boundaries to be controlled and observed using a defined set of scan test principles. An external tester or controller provides instructions and reads the results serially. The external device also provides common clock and control signals. Depending on the implementation, access to all test signals is provided through a standardized 4-pin or 5-pin interface.

In system level applications, individual JTAG-enabled components are connected through their individual testing interfaces (in addition to their more standard application-specific connections). Devices are connected in a series or daisy-chained fashion, with the test output of one device connected exclusively to the test input of the next device in the chain. Instructions in the JTAG boundary scan protocol allow the testing of any one device in the chain, or any combination of devices, without testing the entire chain. In this method, connections between components, as well as connections at the boundary of the application, can be tested.

Figure 24-2 shows a typical application incorporating the JTAG boundary scan interface. In this example, a dsPIC33E/PIC24E device is daisy-chained to a second JTAG-compliant device. The Test Data Input (TDI) line from the external tester supplies data to the Test Data Input (TDI) pin of the first device in the chain (in this case, the DSC). The resulting test data for this two-device chain is provided from the Test Data Output (TDO) pin of the second device to the TDO line of the tester.

Figure 24-2: dsPIC33E/PIC24E-Based JTAG-Compliant Application Showing Daisy Chaining of Components



# dsPIC33E/PIC24E Family Reference Manual

In the dsPIC33E/PIC24E device family, the hardware for the JTAG boundary scan is implemented as a peripheral module (i.e., outside of the CPU core) with additional integrated logic in all I/O ports. The dsPIC33E/PIC24E family implements a 4-pin JTAG interface (see [Table 24-2](#)).

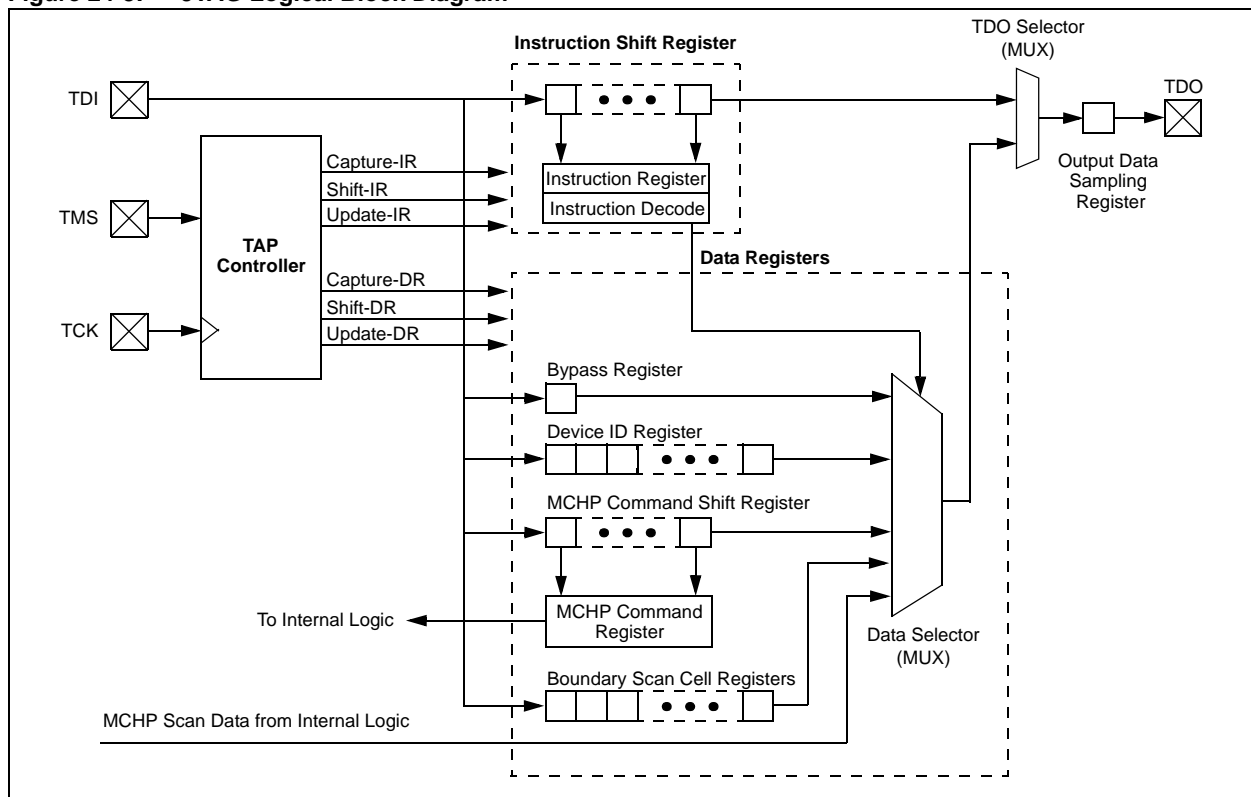
**Table 24-2: JTAG Pin Functions**

| Interface Pin                | Function  |
|------------------------------|---|
| Test Clock Input (TCK)       | Provides the clock for test logic                             |
| Test Mode Select Input (TMS) | Used by the Test Access Port (TAP) to control test operations |
| Test Data Input (TDI)        | Serial input for test instructions and data                   |
| Test Data Output (TDO)       | Serial output for test instructions and data                  |

A logical block diagram of the JTAG module is shown in [Figure 24-3](#), and consists of the following key elements:

- TAP Interface Pins (TDI, TMS, TCK and TDO)
- TAP Controller
- Instruction Shift Register and Instruction Register (IR)
- Data Registers

**Figure 24-3: JTAG Logical Block Diagram**





## 24.4.1 Test Access Port (TAP) and TAP Controller

The Test Access Port (TAP) on the dsPIC33E/PIC24E device family is a general purpose port that provides test access to many built-in support functions and test logic defined in IEEE Standard 1149.1. The TAP controller and the associated boundary scan pins are disabled by programming the JTAG Enable (JTAGEN) bit to '0' in the FICD Configuration register. The TAP controller, by default, is enabled in the bit's unprogrammed state. While enabled, the designated I/O pins become dedicated TAP pins. Use the following process to enable or disable the JTAG port via your MPLAB IDE setup:

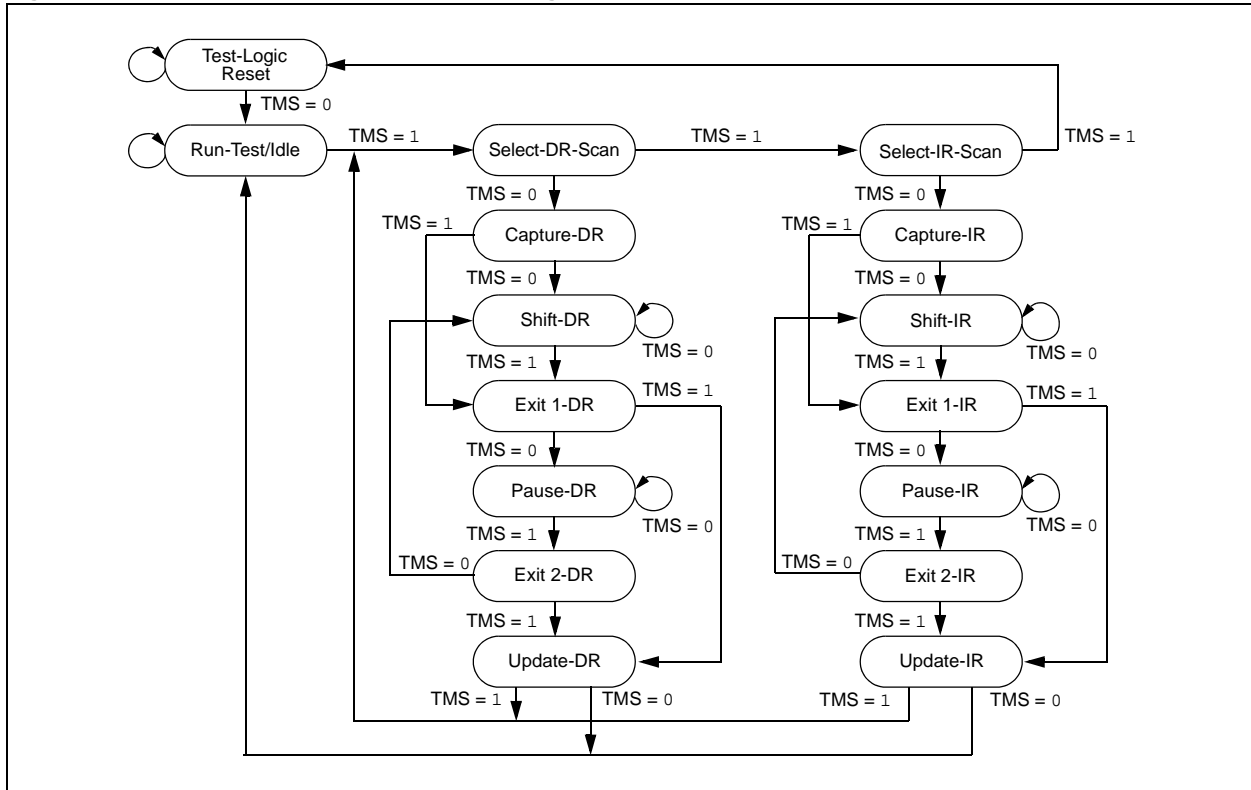
1. In the MPLAB IDE click *Configure > Configuration Bits* menu to display the Configuration Bits window.
2. In the Configuration Bits window, select the Enable/Disable setting under the JTAG Port Enable Category.

**Note:** For information on the FICD register, refer to the “dsPIC33E/PIC24E Flash Programming Specification” (DS70619).

To minimize I/O loss due to JTAG scans, the optional TAP Reset ( $\overline{\text{TRST}}$ ) input pin, specified in the standard, is not implemented on dsPIC33E/PIC24E devices. For convenience, a “soft” TAP Reset is included in the TAP controller, using the TMS and TCK pins. To force a port Reset, apply a logic high to the TMS pin for at least 5 rising edges of TCK. Device Resets (including POR) do not automatically result in a TAP Reset. This must be done by the external JTAG controller using the soft TAP Reset.

The TAP controller on the dsPIC33E/PIC24E family devices is a synchronous finite state machine that implements the standard 16 states for JTAG scans. Figure 24-4 shows all the module states of the TAP controller. All Boundary Scan Test (BST) instructions and test results are communicated through the TAP via the TDI pin in a serial format, Least Significant bit first.

**Figure 24-4: TAP Controller Module State Diagram**



# dsPIC33E/PIC24E Family Reference Manual

By manipulating the state of TMS and the clock pulses on TCK, the TAP controller can be moved through all of the defined module states to capture, shift, and update various instruction and/or data registers. Figure 24-4 shows the state changes on TMS as the controller cycles through its state machine. Figure 24-5 shows the timing of TMS and TCK, while transitioning the controller through the appropriate module states for shifting in an instruction. In this example, the sequence demonstrates how a TAP controller reads an instruction.

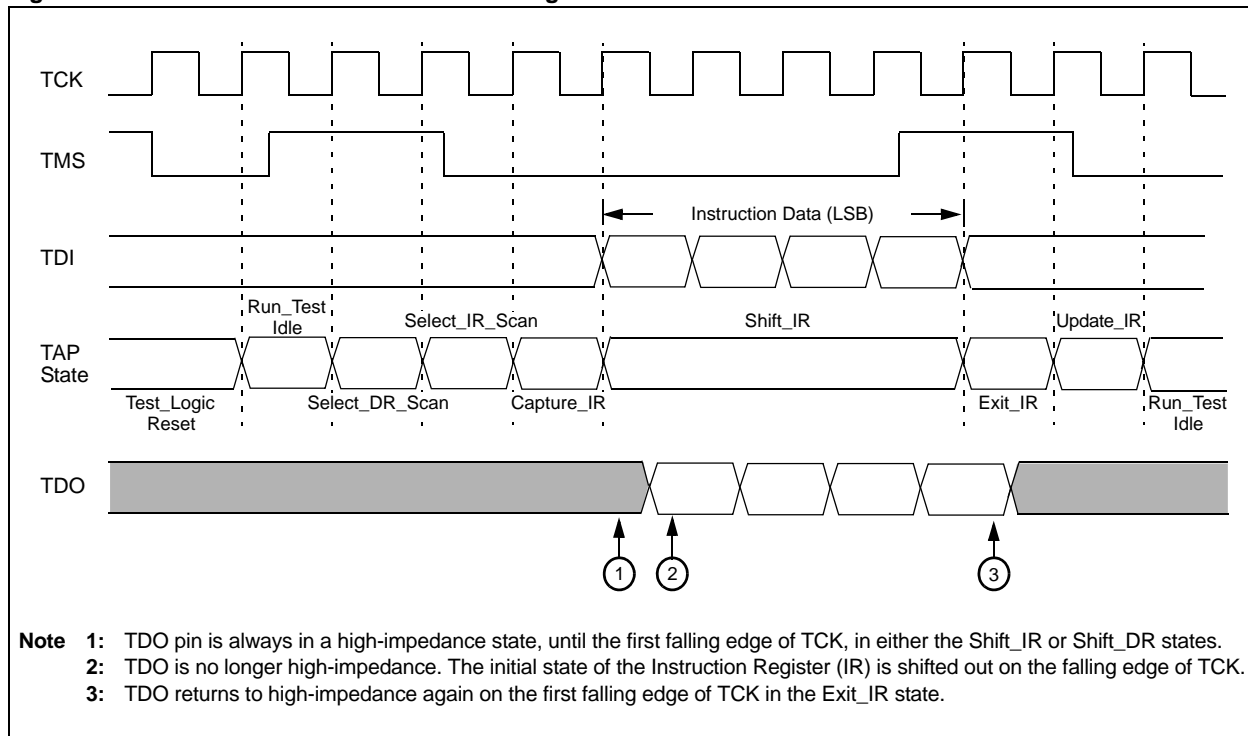
All TAP controller states are entered on the rising edge of the signal on the TCK pin. The TAP controller starts in the Test-Logic Reset state. Since the state of the TAP controller is dependent on the previous instruction, and therefore could be unknown, it is good programming practice to begin in the Test-Logic Reset state.

When TMS is asserted low on the next rising edge of TCK, the TAP controller moves into the Run-Test/Idle state. On the next two rising edges of TCK, TMS is high, which moves the TAP controller to the Select-IR-Scan state.

On the next two rising edges of TCK, TMS is held low, which moves the TAP controller into the Shift-IR state. An instruction is shifted in to the Instruction Shift register via the TDI on the next four rising edges of TCK. After the TAP controller enters this state, the TDO pin goes from a high-impedance state to active. The controller shifts out the initial state of the Instruction Register (IR) on the TDO pin, on the falling edges of TCK, and continues to shift out the contents of the IR while in the Shift-IR state. The TDO returns to the high-impedance state on the first falling edge of TCK upon exiting the shift state.

On the next three rising edges of TCK, the TAP controller exits the Shift-IR state, updates the IR and then moves back to the Run-Test/Idle state. Data, or another instruction, can now be shifted in to the appropriate data or IR.

**Figure 24-5: TAP State Transitions for Shifting in an Instruction**



## 24.4.2 JTAG Registers

The JTAG module uses a number of registers of various sizes as part of its operation. None of the JTAG registers are located within the device data memory space. They cannot be directly accessed by the user application in normal operating modes.

### 24.4.2.1 INSTRUCTION SHIFT REGISTER AND INSTRUCTION REGISTER

The 4-bit IR allows an instruction to be shifted into the device. The instruction selects the data register to access.

The parallel output from the Instruction register is latched to protect from the transient data patterns that occur in its shift register stages as new instruction data is entered. The latched parallel output is controlled, so that it can change state only in the Update-IR and Test-Logic-Reset controller states.

A list and description of implemented instructions is provided in [24.4.4 “JTAG Instructions”](#).

### 24.4.2.2 DATA REGISTERS

The dsPIC33E/PIC24E device family supports the JTAG data registers listed in [Table 24-3](#).

**Table 24-3: JTAG Data Registers**

| Register                         | Function  |
|----------------------------------|---|
| Bypass Register                  | Provides a minimum-length serial path for the movement of test data between TDI and TDO. This path can be selected when no other test data register needs to be accessed during a board-level test operation.<br>Use of the Bypass register in a component speeds access to test data registers in other components on a board-level test data path.  |
| Microchip Command Shift Register | This 8-bit shift register shifts in Microchip device-specific commands. The parallel output from the shift register is latched to protect from the transient data patterns that occur in its shift register stages as a new command is entered.   |
| JTAG Device ID Register          | This 32-bit device IR allows the manufacturer, part number, and variant of a component to be determined. The bit format of the dsPIC33E/PIC24E device consists of an 11-bit manufacturer ID assigned by the IEEE (0x29 for Microchip Technology), device part number, and device revision number. Refer to the <i>“dsPIC33E/PIC24E Flash Programming Specification”</i> (DS70619) or more information on the bit format.<br>For example, the JTAG ID for a dsPIC33EP512MU814 device is: <ul style="list-style-type: none"> <li>• Manufacturer ID = 0x29</li> <li>• Part number = 0x1873</li> <li>• Silicon revision = A0</li> <li>• JTAG ID = 0x01873053</li> </ul> |
| Boundary Scan Register           | Consists of a number of cells combined to form a single shift-register-based path that is connected between TDI and TDO when an appropriate instruction is selected.  |

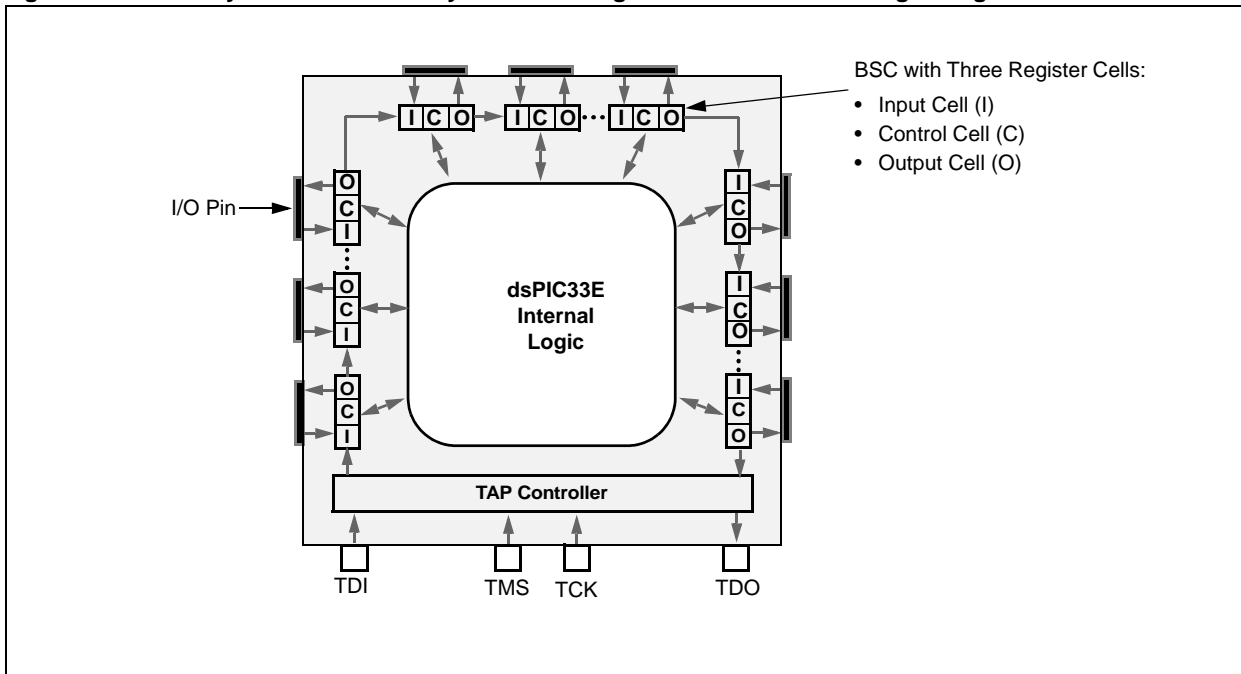
## 24.4.3 Boundary Scan Register

The Boundary Scan Register (BSR) is a large shift register that consists of all the I/O Boundary Scan Cells daisy-chained together, as shown in Figure 24-6. Each I/O pin has one Boundary Scan Cell (BSC). Each BSC contains three BSC registers: an input cell register, an output cell register and a control cell register. When the `SAMPLE/PRELOAD` or `EXTEST` instructions are active, the BSR is placed between the TDI and TDO pins, with the TDI pin as the input and the TDO pin as the output.

The size of the BSR depends on the number of I/O pins on the device. For example, the dsPIC33EP512MU814 has 122 I/O pins. Three BSC registers for each of the 122 I/Os yields a Boundary Scan register length of 366 bits. Information on the I/O port pin count for a specific device is found in the specific BSDL files.

**Note:** The Boundary Scan Cell is not used for power supply pins (VDD, VCAP, VSS, AVDD, AVSS). The pins that have the JTAG interconnect function and JTAG control are not part of the scan-chain and are not JTAG testable.

Figure 24-6: Daisy-Chained Boundary Scan Cell Registers on a dsPIC33E Digital Signal Controller



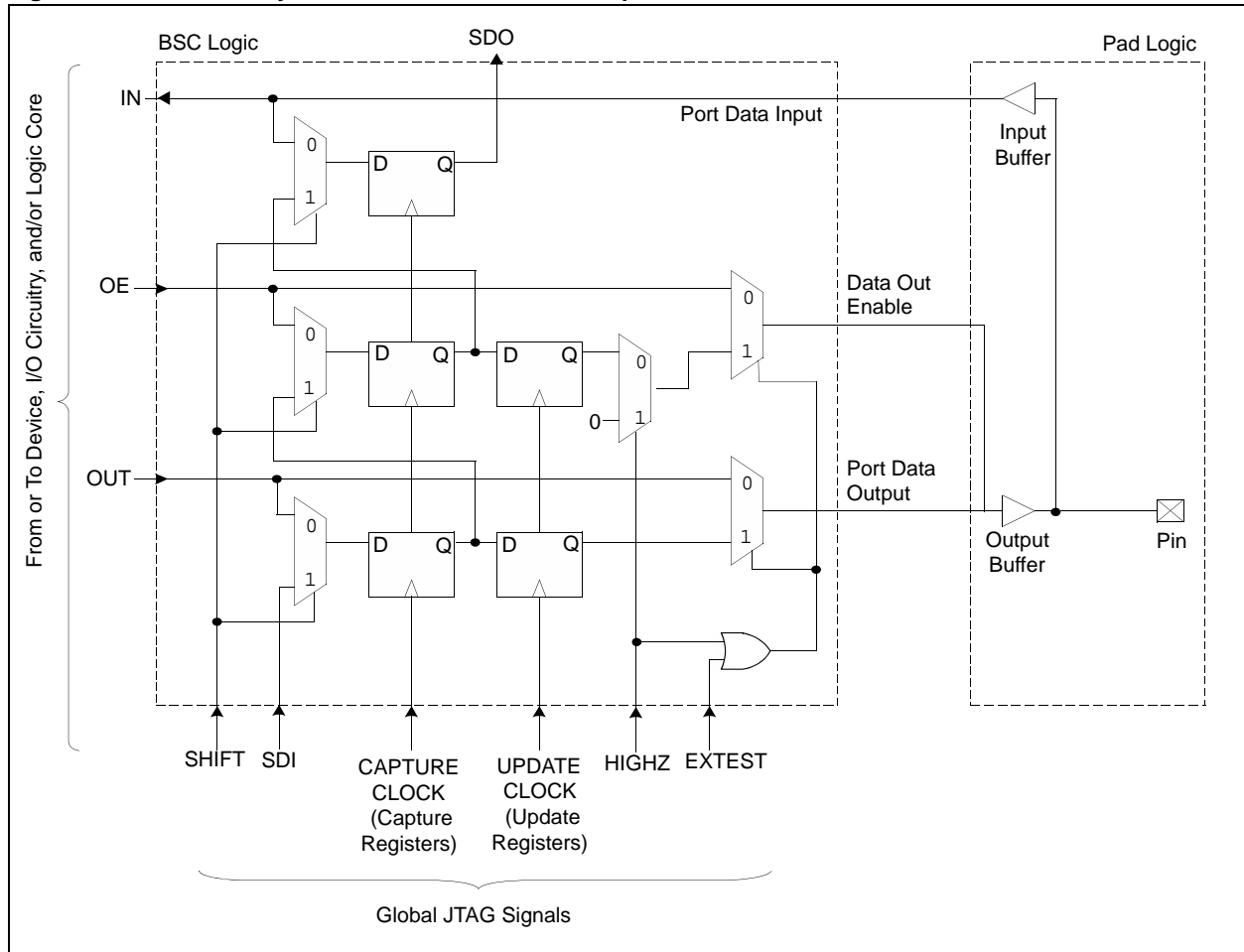
## 24.4.3.1 BOUNDARY SCAN CELL

The Boundary Scan Cell (BSC) captures and overrides I/O input or output data values when JTAG is active. The BSC consists of three single-bit capture register cells and two single-bit holding register cells. The capture cells are daisy-chained to capture the port's input, output and control (output-enable) data. The capture cells also pass JTAG data along to the Boundary Scan register. Command signals from the TAP controller determine if the JTAG data is captured, and how and when it is clocked out of the BSC.

The first register either captures internal data sent to the output driver, or provides serially scanned-in data for the output driver. The second register captures internal output-enable control from the output driver, and also provides serially-scanned output-enable values. The third register captures the input data from the I/O's input buffer.

Figure 24-7 shows a typical BSC and its relationship to the I/O port.

**Figure 24-7: Boundary Scan Cell and Its Relationship to the I/O Port**



# dsPIC33E/PIC24E Family Reference Manual

## 24.4.4 JTAG Instructions

dsPIC33E/PIC24E devices support the mandatory instruction set specified by IEEE 1149.1, as well as several optional public instructions defined in the specification. These devices also implement Microchip-specific instructions. Table 24-4 describes these mandatory, optional, and Microchip-specific JTAG instructions.

**Table 24-4: JTAG Instructions**

| JTAG Instruction                            | Description   |
|---|---|
| <b>Mandatory JTAG Instructions:</b>         |   |
| BYPASS (0x0F)                               | Bypasses a device in a test chain. In Bypass mode, a single shift register stage provides a minimum-length serial path between the TDI and TDO pins.  |
| SAMPLE/PRELOAD (0x01)                       | <p>Takes snapshots of the component's input and output signals without interfering with the normal operation of the assembled board. The snapshot is taken on the rising edge of TCK in the Capture-DR controller state. The data can be viewed by shifting through the component's TDO output.</p> <p>This instruction also allows the scanning of the BSR without interfering with normal operation of the on-chip system logic. For example, before the EXTEST instruction is selected, data can be loaded onto the latched parallel outputs using PRELOAD. As soon as the EXTEST instruction is transferred to the parallel output of the Instruction register, the preloaded data is driven through the system output pins. This ensures that known data, consistent at the board level, is driven immediately when the EXTEST instruction is entered. Without PRELOAD, indeterminate data would be driven until the first scan sequence had been completed.</p> |
| EXTEST (0x03)                               | Allows testing of off-chip circuitry and board level interconnections. Data typically is loaded onto the latched parallel outputs of the Boundary Scan shift register stages by using the PRELOAD instruction before the EXTEST instruction is selected. BSR cells at output pins are used to apply test stimuli. Those at input pins are used to capture test results.   |
| <b>Optional JTAG Instructions</b>           |   |
| IDCODE (0x02)                               | Selects a 32-bit identification register to be connected for serial access between TDI and TDO in the Shift-DR controller state. This instruction causes the 32-bit device identification word to be shifted out on the TDO pin.  |
| HIGHZ (0x04)                                | Places the component in a state in which all of its system logic outputs are placed in an inactive drive state (e.g., high-impedance). In this state, an in-circuit test system drives the signals onto the connections normally driven by a component output without damaging the component. In the HIGHZ mode, the Bypass register is connected between TDI and TDO in the Shift-DR state.  |
| <b>Microchip-specific JTAG Instructions</b> |   |
| MCHP_SCAN (0x07)                            | Selects the internal Microchip-specific scan register to be connected for serial access between the TDI and TDO in the Shift-DR controller state.   |
| MCHP_CMD (0x08)                             | <p>Selects 8-bit Microchip Command shift register to be connected for serial access between the TDI and TDO in the Shift-DR controller state. This shift register supports up to 256 commands. The following command is available for the user; the rest are reserved:</p> <p>JTAG_MCLR (0x01): Performs a device Master Clear Reset while the JTAG interface is active; functionally equivalent to hardware MCLR. The TAP interface itself is not reset.</p>   |

### 24.4.5 Boundary Scan Testing

Boundary Scan Testing (BST) is the method of controlling and observing the boundary pins of the JTAG-compliant device with software. BST can be used to test connectivity between devices by daisy-chaining JTAG compliant devices to form a single scan chain. Several scan chains can exist on a printed circuit board to form multiple scan chains. These multiple scan chains can then be driven simultaneously to test many components in parallel. Scan chains can contain both JTAG compliant devices and non-JTAG compliant devices.

A key advantage of BST is that it can be implemented without physical test probes. All that is needed is a 4-wire or 5-wire interface and an appropriate test platform. Since JTAG boundary scan has been available for many years, many software tools exist for testing scan chains without the need for extensive physical probing. The main drawback to BST is that it can only evaluate digital signals and circuit continuity. It cannot measure input or output voltage levels or currents.

#### 24.4.5.1 RELATED JTAG FILES

To implement BST, all JTAG test tools require a Boundary Scan Description Language (BSDL) file. BSDL is a subset of VHSIC Hardware Description Language (VHDL), and is described as part of IEEE 1149.1. The device-specific BSDL file describes how the standard is implemented on a particular device and how it operates. The BSDL file for a particular device includes the following:

- Pinout and package configuration for the particular device
- Physical location of the TAP pins
- Device ID register and the device ID
- Length of the IR
- Supported BST instructions and their binary codes
- Length and structure of the Boundary Scan register
- Boundary scan cell definition

Device-specific BSDL files are available at Microchip's web site, [www.microchip.com](http://www.microchip.com). The name for each BSDL file is the device name and silicon revision. For example, dsPIC33EP512MU814.BSD is the BSDL file for the dsPIC33EP512MU814 device.

## 24.5 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 dsPIC33E/PIC24E device family, but the concepts are pertinent and could be used with modification and possible limitations. The current application notes related to Programming and Diagnostics are:

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

**Note:** Visit the Microchip web site ([www.microchip.com](http://www.microchip.com)) for additional application notes and code examples for the dsPIC33E/PIC24E family of devices.



## 24.6 REVISION HISTORY

### Revision A (August 2009)

This is the initial release of the document.

### Revision B (December 2010)

This revision includes the following updates:

- Added a note at the beginning of the section, which provides information on complementary documentation
- Updated the dsPIC33E references in the entire document as dsPIC33E/PIC24E
- Updated the first paragraph in [24.2.1 “ICSP Interface”](#)
- Changed the name of pin 123 to VCAP (see [Figure 24-1: “Example of Programming Pin Pairs on a dsPIC33E Device”](#))
- Add information on the new and improved Debug module capabilities (see [24.2.3 “ICSP and In-Circuit Debugging”](#))
- Removed the last sentence of the first paragraph and removed the fifth paragraph in [24.4 “JTAG Boundary Scan”](#)
- Updated the description for MCHP\_CMD (0x08) in [Table 24-4](#)
- Removed 24.4.6 “JTAG Device Programming”
- Updates to formatting and minor text changes have been incorporated throughout the document

NOTES:

---

**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.

---

DS70608B24 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, PIC<sup>32</sup> logo, rPIC 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, Omniscient Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICkit, PICTail, REAL ICE, rLAB, 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.

© 2010, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

 Printed on recycled paper.

ISBN: 978-1-60932-761-3

*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<sup>®</sup> MCUs and dsPIC<sup>®</sup> DSCs, KEELOQ<sup>®</sup> 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 16949:2002 ==**



# MICROCHIP

## 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](http://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 - Chongqing**  
Tel: 86-23-8980-9588  
Fax: 86-23-8980-9500

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

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

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

### ASIA/PACIFIC

**India - Bangalore**  
Tel: 91-80-3090-4444  
Fax: 91-80-3090-4123

**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: 886-7-213-7830  
Fax: 886-7-330-9305

**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

08/04/10