



Programming and Diagnostics

HIGHLIGHTS

This section of the manual contains the following topics:

1.0	Introduction	2
2.0	In-Circuit Serial Programming™ (ICSP™)	3
3.0	Enhanced ICSP	6
4.0	JTAG Boundary Scan	7
5.0	Related Application Notes	16
6.0	Revision History	17

dsPIC33/PIC24 Family Reference Manual

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 dsPIC33/PIC24 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 Website at: <http://www.microchip.com>

1.0 INTRODUCTION

dsPIC33/PIC24 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

dsPIC33/PIC24 devices incorporate three different programming and diagnostic modalities that provide a range of functions useful to the application developer. They are summarized in [Table 1-1](#).

Table 1-1: Comparison of dsPIC33/PIC24 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

2.0 IN-CIRCUIT SERIAL PROGRAMMING™ (ICSP™)

The In-Circuit Serial Programming (ICSP) 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.

2.1 ICSP Interface

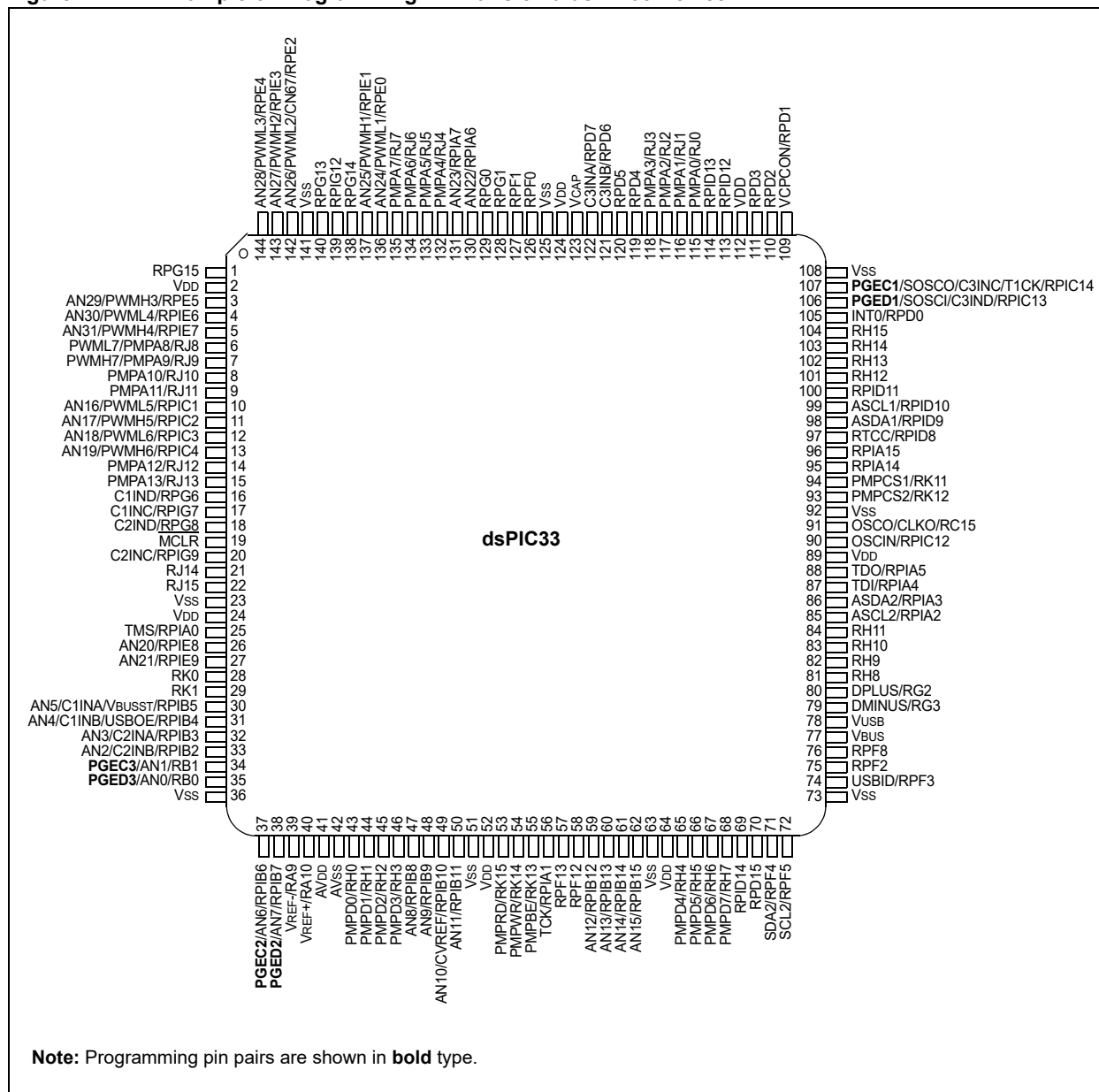
The ICSP method uses a two-pin communication interface. The Programming Data (PGD) pin functions as both an input and an output, allowing programming data to be read in and device information to be read out on command. The Programming Clock (PGC) pin clocks in data and controls the overall process.

Most dsPIC33/PIC24 devices have more than one pair of PGC and PGD pins; these are multiplexed with other I/Os or peripheral functions, as shown in [Figure 2-1](#). 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 MCLR pin, which is used with PGCx to enter and control the programming process, must also be connected to the programming device.

dsPIC33/PIC24 Family Reference Manual

Figure 2-1: Example of Programming Pin Pairs on a dsPIC33 Device



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 dsPIC33/PIC24 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 MCLR pins. Entry into and exit from ICSP mode involves applying (or removing) voltage to 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 dsPIC33/PIC24 core logic; however, overall control of the process must be provided by the external programming tool. Microchip programming tools include the necessary hardware and algorithms to manage the programming process for dsPIC33/PIC24 devices. Users who are interested in a more detailed description, or who are considering designing their own programming interface for a dsPIC33/PIC24 device, should refer to the device specific data sheet for the applicable dsPIC33/PIC24 Flash Programming Specification.

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 dsPIC33/PIC24 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 device specific data sheet for the applicable dsPIC33/PIC24 Flash Programming Specification.

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 dsPIC33/PIC24 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 *Window > Target Memory Views > Configuration Bits* menu to display the Configurations Bits window.
2. In the Configuration Bits window, select the appropriate debug pair setting under the ICD Communication Channel Select Category.

Note: For details on the configuration memory space, refer to the device specific data sheet for the applicable dsPIC33/PIC24 Flash Programming Specification.
--

The dsPIC33 and PIC24 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.

dsPIC33/PIC24 Family Reference Manual

3.0 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 (PE), 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 dsPIC33/PIC24 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 device specific data sheet for the applicable dsPIC33/PIC24 Flash Programming Specification.

4.0 JTAG BOUNDARY SCAN

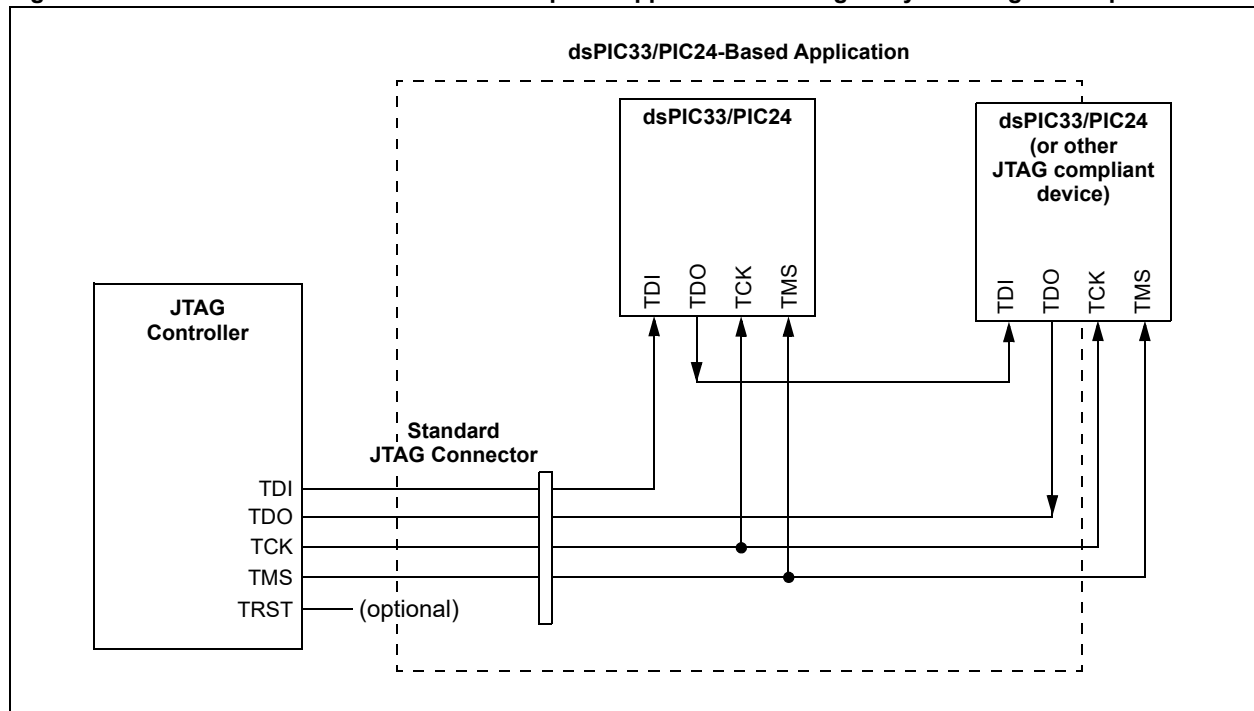
As the complexity and density of board designs increase, testing electrical connections between the components on fully assembled circuit boards poses many challenges. To address these challenges, the Joint Test Action Group (JTAG) developed a method for boundary scan testing that was later standardized 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 4-1 shows a typical application incorporating the JTAG boundary scan interface. In this example, a dsPIC33/PIC24 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 are provided from the Test Data Output (TDO) pin of the second device to the TDO line of the tester.

Figure 4-1: dsPIC33/PIC24-Based JTAG-Compliant Application Showing Daisy Chaining of Components



dsPIC33/PIC24 Family Reference Manual

In the dsPIC33/PIC24 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 dsPIC33/PIC24 device family implements a 4-pin JTAG interface (see [Table 4-1](#)).

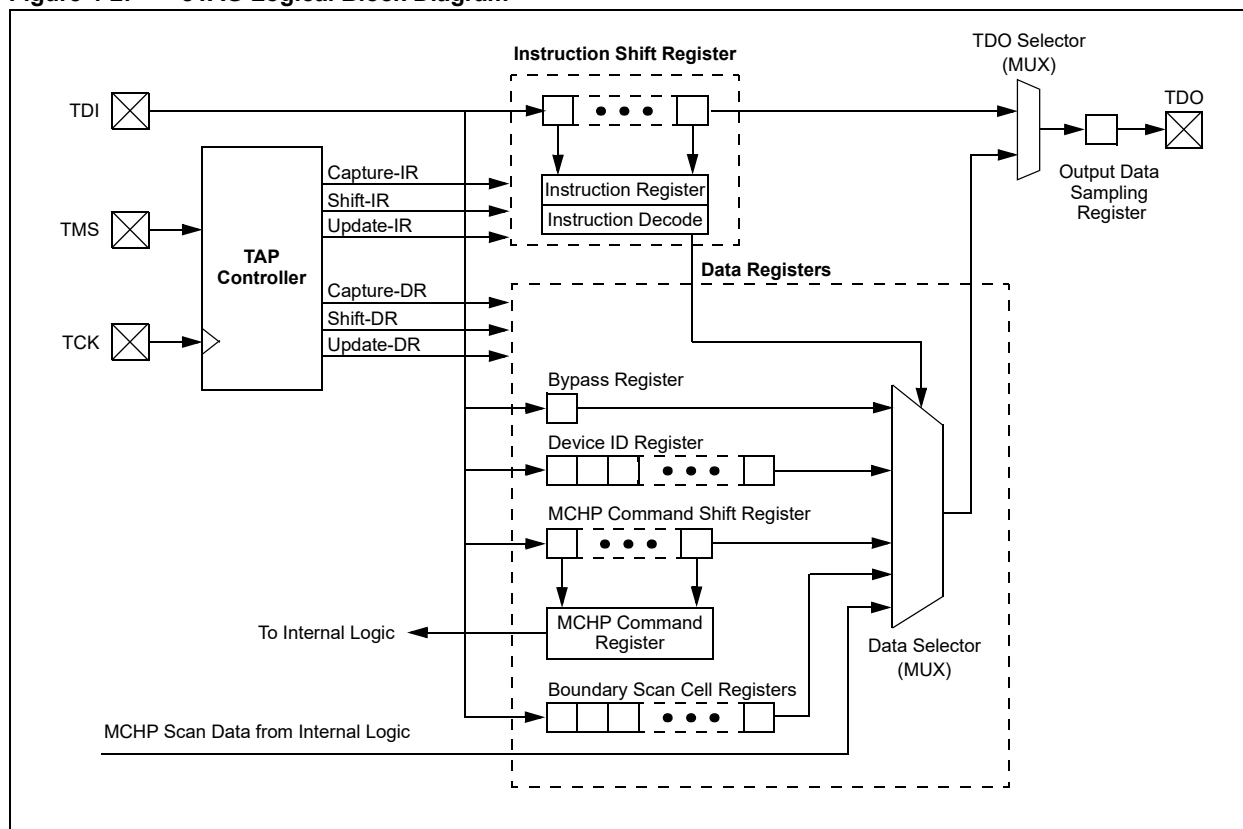
Table 4-1: JTAG Pin Functions

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 4-2](#) 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 4-2: JTAG Logical Block Diagram



4.1 Test Access Port (TAP) and TAP Controller

The Test Access Port (TAP) on the dsPIC33/PIC24 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:

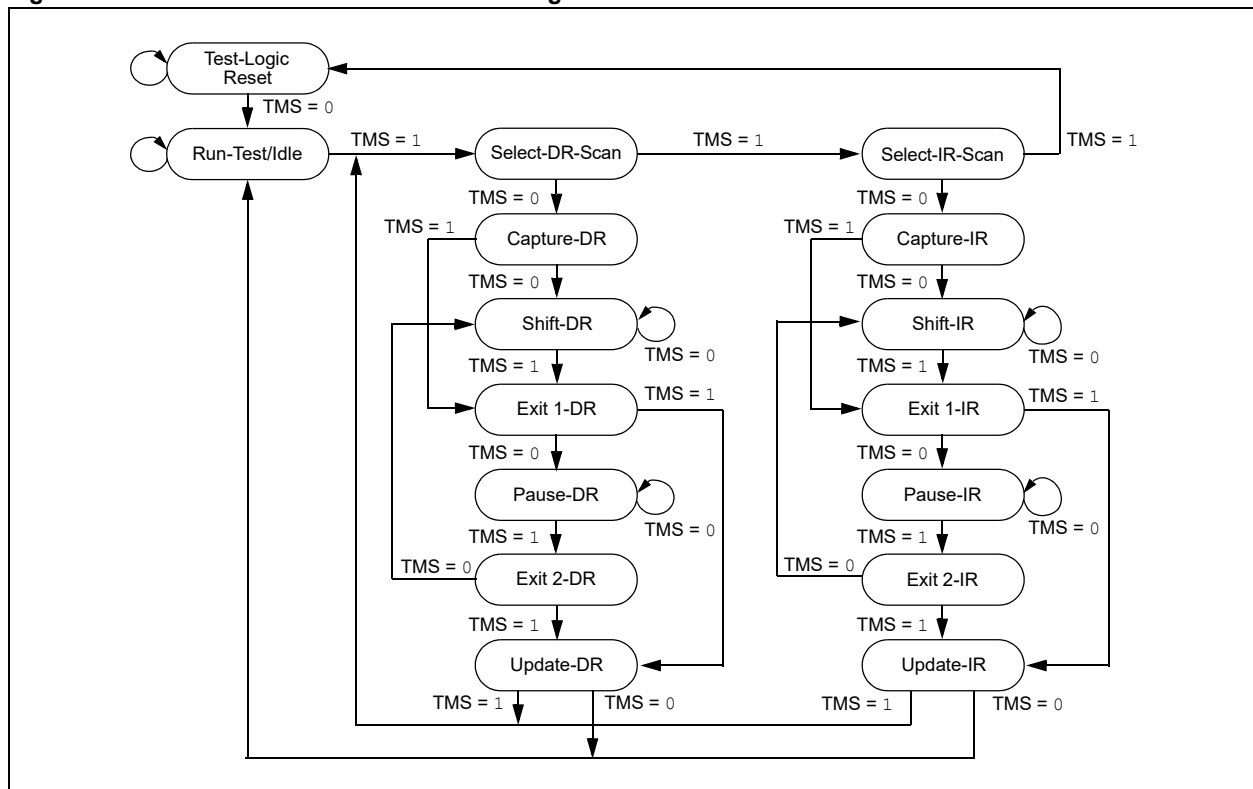
1. In the MPLAB IDE, click the *Configure > Configuration Bits* menu to display the Configuration Bits window.
2. In the Configuration Bits window, select the Enable/Disable setting under the JTAG Port Enable Category.

Note: For information on the FICD register, refer to the device specific data sheet for the applicable dsPIC33/PIC24 Flash Programming Specification.

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 dsPIC33/PIC24 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 five 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 dsPIC33/PIC24 family devices is a synchronous finite state machine that implements the standard 16 states for JTAG scans. Figure 4-3 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 4-3: TAP Controller Module State Diagram



dsPIC33/PIC24 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 4-3 shows the state changes on TMS as the controller cycles through its state machine. Figure 4-4 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.

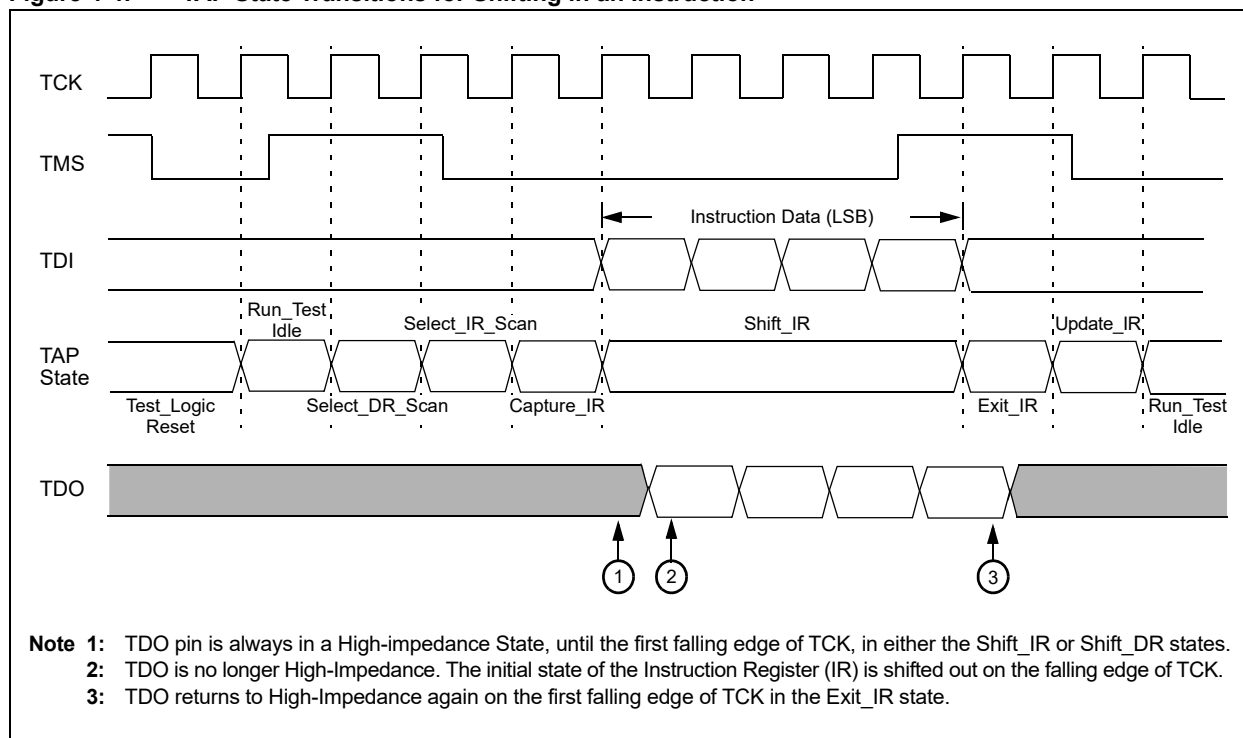
All TAP controller states are entered on the rising edge of the signal on the TCK pin. The TAP controller starts in the Test-Logic Reset state. Since the state of the TAP controller is dependent on the previous instruction, and therefore could be unknown, it is good programming practice to begin in the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

Figure 4-4: TAP State Transitions for Shifting in an Instruction



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.

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 are 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 [Section 4.4 “JTAG Instructions”](#).

4.2.2 DATA REGISTERS

The dsPIC33/PIC24 device family supports the JTAG data registers listed in [Table 4-2](#).

Table 4-2: 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. 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 dsPIC33/PIC24 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 device specific data sheet for the applicable dsPIC33/PIC24 Flash Programming Specification for more information on the bit format. For example, the JTAG ID for a dsPIC33EP512MU814 device is: • Manufacturer ID = 0x29 • Part number = 0x1873 • Silicon revision = A0 • JTAG ID = 0x01873053
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.

dsPIC33/PIC24 Family Reference Manual

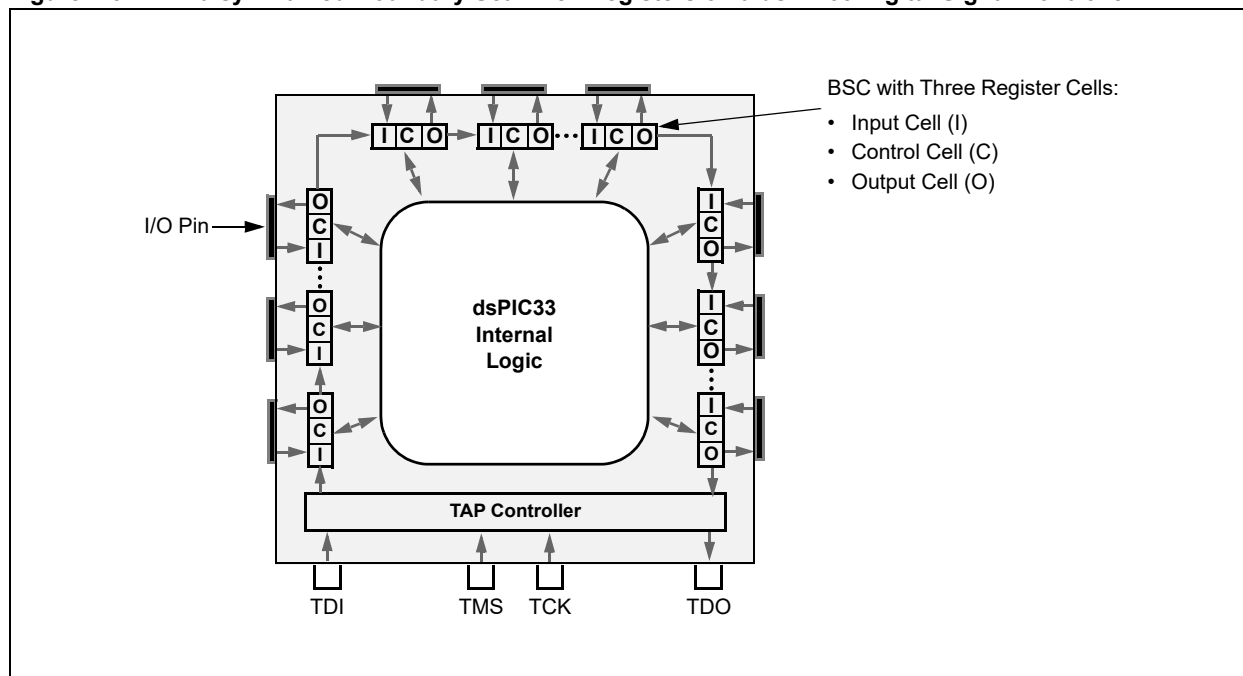
4.3 Boundary Scan Register

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 4-5. 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 4-5: Daisy-Chained Boundary Scan Cell Registers on a dsPIC33 Digital Signal Controller



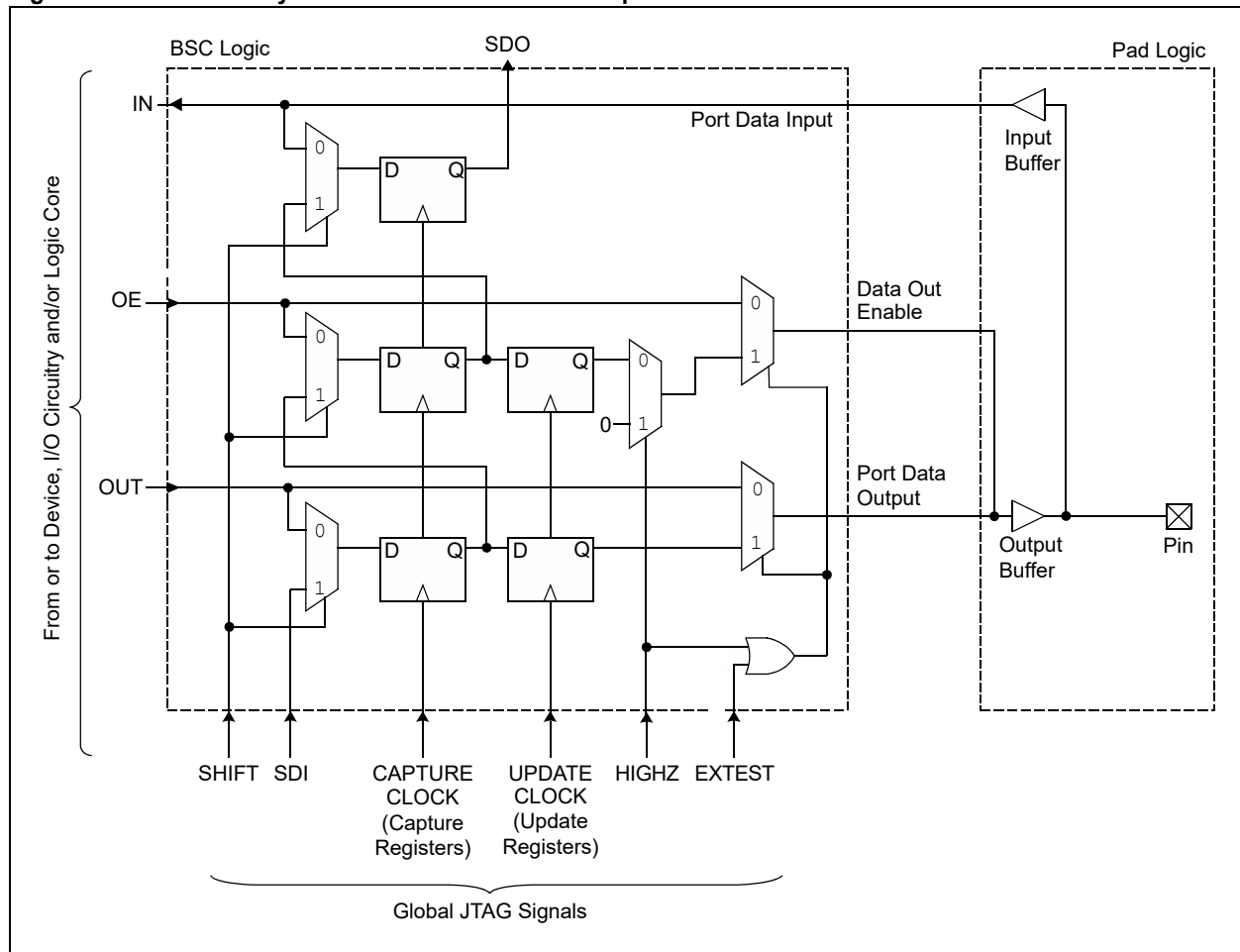
4.3.1 BOUNDARY SCAN CELL

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 are captured, and how and when they are 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 4-6 shows a typical BSC and its relationship to the I/O port.

Figure 4-6: Boundary Scan Cell and Its Relationship to the I/O Port



dsPIC33/PIC24 Family Reference Manual

4.4 JTAG Instructions

dsPIC33/PIC24 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 4-3 describes these mandatory, optional, and Microchip-specific JTAG instructions.

Table 4-3: JTAG Instructions

JTAG Instruction	Description
Mandatory JTAG Instructions:	
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)	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. 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 are driven through the system output pins. This ensures that known data, consistent at the board level, are driven immediately when the EXTEST instruction is entered. Without PRELOAD, indeterminate data would be driven until the first scan sequence had been completed.
EXTEST (0x03)	Allows testing of off-chip circuitry and board-level interconnections. Data typically are 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.
Optional JTAG Instructions	
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 High-Z mode, the Bypass register is connected between TDI and TDO in the Shift-DR state.
Microchip-Specific JTAG Instructions	
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)	Selects the 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: 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.

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 (PCB) 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 four-wire or five-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.

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 website, 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.

dsPIC33/PIC24 Family Reference Manual

5.0 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 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 website (www.microchip.com) for additional application notes and code examples for the dsPIC33/PIC24 family of devices.

6.0 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 dsPIC33/PIC24
- Updated the first paragraph in [Section 2.1 “ICSP Interface”](#)
- Changed the name of pin 123 to VCAP (see [Figure 2-1: “Example of Programming Pin Pairs on a dsPIC33 Device”](#))
- Add information on the new and improved Debug module capabilities (see [Section 2.3 “ICSP and In-Circuit Debugging”](#))
- Removed the last sentence of the first paragraph and removed the fifth paragraph in [Section 4.0 “JTAG Boundary Scan”](#)
- Updated the description for MCHP_CMD (0x08) in [Table 4-3](#)
- Removed 24.4.6 “JTAG Device Programming”
- Updates to formatting and minor text changes have been incorporated throughout the document

Revision C (August 2022)

This revision includes the following updates:

- Applied current Family Reference Manual format.
- Updated all dsPIC33E/PIC24E references to dsPIC33/PIC24.

dsPIC33/PIC24 Family Reference Manual

NOTES:

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

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
 - Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
 - Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip product is strictly prohibited and may violate the Digital Millennium Copyright Act.
 - Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is "unbreakable" Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.
-

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications 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. Contact your local Microchip sales office for additional support or, obtain additional support at <https://www.microchip.com/en-us/support/design-help/client-support-services>.

THIS INFORMATION IS PROVIDED BY MICROCHIP "AS IS". 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 ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP'S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

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.

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

Trademarks

The Microchip name and logo, the Microchip logo, Adaptec, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, CryptoMemory, CryptoRF, dsPIC, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Klear, LANCheck, LinkMD, maxStylus, maxTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

AgileSwitch, APT, ClockWorks, The Embedded Control Solutions Company, EtherSynch, Flashtec, Hyper Speed Control, HyperLight Load, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet-Wire, SmartFusion, SyncWorld, Temux, TimeCesium, TimeHub, TimePictra, TimeProvider, TrueTime, and ZL 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, Augmented Switching, BlueSky, BodyCom, Clockstudio, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, Espresso T1S, EtherGREEN, GridTime, IdealBridge, In-Circuit Serial Programming, ICSP, INICnet, Intelligent Paralleling, IntelliMOS, Inter-Chip Connectivity, JitterBlocker, Knob-on-Display, KoD, maxCrypto, maxView, memBrain, Mindi, MiWi, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PowerSmart, PureSilicon, QMatrix, REAL ICE, Ripple Blocker, RTAX, RTG4, SAM-ICE, Serial Quad I/O, simpleMAP, SimpliPHY, SmartBuffer, SmartHLS, SMART-I.S., storClad, SQI, SuperSwitcher, SuperSwitcher II, Switchtec, SynchroPHY, Total Endurance, Trusted Time, TSHARC, USBCheck, VariSense, VectorBlox, VeriPHY, ViewSpan, WiperLock, XpressConnect, 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.

The Adaptec logo, Frequency on Demand, Silicon Storage Technology, and Symmcom are registered trademarks 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.

© 2010-2022, Microchip Technology Incorporated and its subsidiaries.

All Rights Reserved.

ISBN: 978-1-6683-1141-7

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: 45-4485-5910
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: 49-7131-72400

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: 47-7288-4388

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