



MPLAB[®] Connect Configurator GUI

User's Guide

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

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

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

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in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates.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

Trademarks

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

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

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

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

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

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

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

© 2018, Microchip Technology Incorporated, All Rights Reserved.

ISBN: 978-1-5224-3582-2

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

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELQ® 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.

Table of Contents

Preface	5
Introduction	5
Document Layout	5
Conventions Used in this Guide	7
The Microchip Website	8
Customer Support	8
Document Revision History	8
Chapter 1. Overview	
1.1 Introduction	9
1.1.1 Terms and Abbreviations	9
1.2 Drivers	9
1.2.1 USB Hub Drivers	9
1.2.2 LAN78xx Driver	9
1.2.3 LAN74xx Drivers	9
1.3 Logging	10
1.4 Version	10
1.5 Reference	10
Chapter 2. Prerequisites	
2.1 Disable Power Management	11
2.2 JAVA Installation	11
Chapter 3. Endianness	
Chapter 4. Configuration File Formats	
4.1 JSON Format	15
4.2 Binary Format	15
Chapter 5. USB Hub Devices	
5.1 Online Hub Configuration Page	17
5.1.1 Basic Features	19
5.1.2 Advanced Features	26
5.1.3 Preview Changes	42
5.1.4 Special Features	42
5.2 Programming Page	50
5.2.1 Device Selection	51
5.2.2 Configuration Programming	51
5.2.3 Firmware Programming	52
5.2.4 Configuration and SPI Flash Firmware Programming	52
5.2.5 Types of Programming	53
5.3 Offline Hub Configuration Page	54

Chapter 6. LAN78xx Devices

6.1 Online LAN78xx Configuration Page	57
6.1.1 Basic Features	57
6.1.2 Advanced Features	60
6.1.3 Preview Changes	66
6.2 Programming Page	66
6.2.1 Program Once	66
6.2.2 Mass Program	68
6.3 Offline LAN78xx Configuration Page	69

Chapter 7. LAN74XX Devices

7.1 Online LAN74XX Configuration Page	73
7.1.1 Basic Features	74
7.1.2 Advanced Features	78
7.1.3 Special Features	84
7.2 Programming Page	85
7.2.1 Program Once	85
7.3 Offline LAN74xx Configuration Page	86

Appendix A. Troubleshooting and Error Codes

A.1 Troubleshooting	89
A.2 Error Codes	89

Appendix B. Disable Power Management

B.1 Windows 7	91
B.2 Windows 8.1	94

Appendix C. Protouch/MPLABConnect Interoperability

C.1 Introduction	95
------------------------	----

Appendix D. Verifying MPLABConnect Driver Installation

D.1 Introduction	99
D.1.1 VSM as Class Filter	99
D.1.2 WinUSB Driver	104

Appendix E. Supported Configuration Items

E.1 Introduction	107
------------------------	-----

Worldwide Sales and Service	116
--	------------

Preface

NOTICE TO CUSTOMERS

All documentation becomes dated, and this manual is no exception. Microchip tools and documentation are constantly evolving to meet customer needs, so some actual dialogs and/or tool descriptions may differ from those in this document. Please refer to our website (www.microchip.com) to obtain the latest documentation available.

Documents are identified with a “DS” number. This number is located on the bottom of each page, in front of the page number. The numbering convention for the DS number is “DSXXXXA”, where “XXXX” is the document number and “A” is the revision level of the document.

For the most up-to-date information on development tools, see the MPLAB® IDE online help. Select the Help menu, and then Topics to open a list of available online help files.

INTRODUCTION

This chapter contains general information that will be useful to know before using MPLAB® Connect Configurator GUI User's Guide. Items discussed in this chapter include:

- [Document Layout](#)
- [Conventions Used in this Guide](#)
- [The Microchip Website](#)
- [Customer Support](#)
- [Document Revision History](#)

DOCUMENT LAYOUT

This document describes how to use the MPLAB Connect Configurator GUI configuration and programming tool for Microchip USB hubs and LAN products. The manual layout is as follows:

- **Chapter 1. “Overview”** – This chapter shows a brief description of the MPLAB Connect Configurator GUI.
- **Chapter 2. “Prerequisites”** – This chapter shows prerequisites and instructions for using the MPLAB Connect Configurator GUI.
- **Chapter 3. “Endianness”** – This chapter explains the *endian* format.
- **Chapter 4. “Configuration File Formats”** – This chapter shows the configuration file formats of the MPLAB Connect Configurator GUI.
- **Chapter 5. “USB Hub Devices”** – This chapter shows information on the USB hub device configuration pages of the MPLAB Connect Configurator GUI.
- **Chapter 6. “LAN78xx Devices”** – This chapter shows information on the LAN78xx device configuration pages of the MPLAB Connect Configurator GUI.
- **Chapter 7. “LAN74XX Devices”** – This chapter shows information on the LAN74xx device configuration pages of the MPLAB Connect Configurator GUI.

- **Appendix A. “Troubleshooting and Error Codes”** – This appendix shows the troubleshooting information for the MPLAB Connect Configurator GUI.
- **Appendix B. “Disable Power Management”** – This appendix details how to disable power management for the MPLAB Connect Configurator GUI.
- **Appendix C. “Protouch/MPLABConnect Interoperability”** – This appendix details the Protouch/MPLAB Connect Configurator GUI interoperability.
- **Appendix D. “Verifying MPLABConnect Driver Installation”** – This appendix shows how to check whether drivers were installed correctly for the MPLAB Connect Configurator GUI.
- **Appendix E. “Supported Configuration Items”** – This appendix details a list of supported configuration items for the MPLAB Connect Configurator GUI.

CONVENTIONS USED IN THIS GUIDE

This manual uses the following documentation conventions:

DOCUMENTATION CONVENTIONS

Description	Represents	Examples
Arial font:		
Italic characters	Referenced books	<i>MPLAB® IDE User's Guide</i>
	Emphasized text	...is the <i>only</i> compiler...
Initial caps	A window	the Output window
	A dialog	the Settings dialog
	A menu selection	select Enable Programmer
Quotes	A field name in a window or dialog	"Save project before build"
Underlined, italic text with right angle bracket	A menu path	<u><i>File>Save</i></u>
Bold characters	A dialog button	Click OK
	A tab	Click the Power tab
N'Rnnnn	A number in verilog format, where N is the total number of digits, R is the radix and n is a digit.	4'b0010, 2'hF1
Text in angle brackets < >	A key on the keyboard	Press <Enter>, <F1>
Courier New font:		
Plain Courier New	Sample source code	#define START
	Filenames	autoexec.bat
	File paths	c:\mcc18\h
	Keywords	_asm, _endasm, static
	Command-line options	-Opa+, -Opa-
	Bit values	0, 1
	Constants	0xFF, 'A'
Italic Courier New	A variable argument	<i>file.o</i> , where <i>file</i> can be any valid filename
Square brackets []	Optional arguments	mcc18 [options] <i>file</i> [options]
Curly brackets and pipe character: { }	Choice of mutually exclusive arguments; an OR selection	errorlevel {0 1}
Ellipses...	Replaces repeated text	var_name [, var_name...]
	Represents code supplied by user	void main (void) { ... }

THE MICROCHIP WEBSITE

Microchip provides online support via our website at www.microchip.com. This website is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the website contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases, and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors, and factory representatives

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the website at:

<http://www.microchip.com/support>

DOCUMENT REVISION HISTORY

Revision	Section/Figure/Entry	Correction
DS50002634B (10-1-18)	Chapter 5. "Application"	Removed chapter
	Chapter 5. "USB Hub Devices"	Added new figures and revised some sections.
	Chapter 6. "LAN78xx Devices"	Revised some sections
	Chapter 7. "LAN74XX Devices"	New chapter added to the user's guide
	Table E-8	New table added to Appendix E. "Supported Configuration Items"
	All	Added information on USB70xx, LAN74xx, and USB57xx. Changed all "USB57x4" to "USB57xx." Made minor text changes throughout the document.
DS50002634A (08-16-17)	Initial Microchip release	

Chapter 1. Overview

1.1 INTRODUCTION

MPLAB® Connect Configurator GUI is a configuration and programming tool for the following Microchip USB hubs, LAN, and PCIe products:

- USB253x/USB(8)4604/USB3x13
- USB57xx
- USB58xx/USB59xx
- USB49xx/USB4715
- USB70xx
- LAN74xx
- LAN78xx

1.1.1 Terms and Abbreviations

MPLABConnect – MPLAB Connect Configurator

HFC – Hub Feature Controller (Internal USB Device)/ Hub Controller Endpoint (Internal USB Device)

UUID – Universal Unique Identifier

HSIC – High-Speed Inter-Chip

NCM – Network Control Model

1.2 DRIVERS

1.2.1 USB Hub Drivers

The WinUSB driver must be installed before launching the MPLAB Connect Configurator GUI tool. Hub class filter installation can be done by using the command line tool.

Note: The VSM filter driver must be installed if the internal HFC device is disabled.

Please refer to *Appendix F* in the *MPLAB Connect Configurator CLI User's Guide* for WinUSB driver installation.

1.2.2 LAN78xx Driver

The LAN78xx driver must be installed before connecting the LAN device to a computer. It is available in {MPLAB Connect tool folder}\Drivers\LAN78xx-Driver\install.exe.

1.2.3 LAN74xx Drivers

LAN74xx driver needs to be installed before connecting the PCIe device to a computer. It is available in {MPLAB Connect tool folder}\Drivers\LAN74xx-Driver\install.exe.

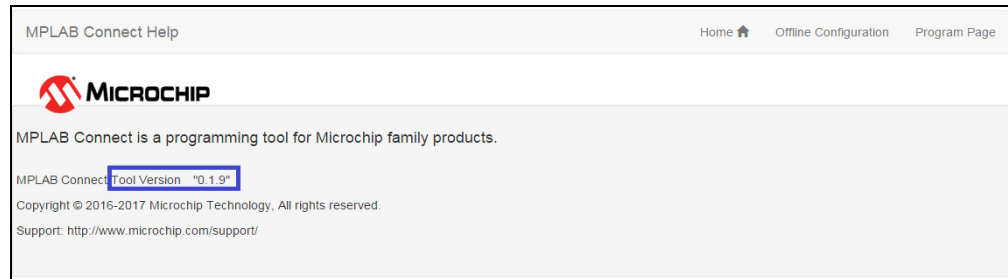
1.3 LOGGING

A log file with the name `MPLABConnect.log` that contains detailed messages is created in the same path where the application is running.

1.4 VERSION

The version number of the tool can be found on the help page. See [Figure 1-1](#).

FIGURE 1-1: APPLICATION HELP PAGE



1.5 REFERENCE

For more information on supported operating systems, Microchip devices, and USB controllers as well as a list of known limitations, refer to:

- *MPLAB® Connect Configurator Release Notes*

Chapter 2. Prerequisites

2.1 DISABLE POWER MANAGEMENT

Power management for the corresponding USB hub, LAN device, or PCIe device should be disabled before launching the tool. If power management is enabled, the USB hub, LAN device or PCIe device would be put into Sleep mode and the tool would fail to access the device. Refer to [Appendix B. “Disable Power Management”](#) for details on how to disable this field.

Note: If using a USB 3.1 Gen1 Hub, power management should be disabled on both the USB 2.0 Hub and USB 3.1 Gen1 Hub.

2.2 JAVA INSTALLATION

1. Download JDK1.8.0_60 from the following website:
<http://www.oracle.com/technetwork/java/javase/downloads/java-archive-javase8-2177648.html#jdk-8u60-oth-JPR>

Download the version: Windows x86

2. Click on to the downloaded installer and install Java.

The default location of the Java Installation is as follows:

- 32-bit machine: C:\Program Files (x86)\Java\jdk1.8.0_60
- 64-bit machine: C:\Program Files\Java\jdk1.8.0_60

NOTES:

Chapter 3. Endianness

All the widgets except direct register access follow the *little-endian* format. Direct register access alone follows the *big-endian* format. Refer to [Chapter 5. “USB Hub Devices”](#) for USB and [Chapter 6. “LAN78xx Devices”](#) for LAN devices, and [Chapter 7. “LAN74XX Devices”](#) for PCIe devices.

NOTES:

Chapter 4. Configuration File Formats

The configuration file holds the data to be programmed onto the hub. The configuration file used for programming can be in Binary format (.*cfg*) or JSON format (.*json*) for USB hub products and Binary format (.*bin*) for LAN78xx products. These files are generated when changes are done to the configuration items in the hub.

4.1 JSON FORMAT

JavaScript Object Notation (JSON) files use human-readable text-to-program. JSON files start with open brace "{" and end with closed brace "}". A number of properties using "*name*": "*value*" pairing separated by commas (,) can be declared inside these braces. There should not be any empty space or empty lines.

Use only the case-sensitive names listed in [Appendix E. "Supported Configuration Items"](#) when using JSON files. A JSON file should be in the format shown in [Example 4-1](#).

EXAMPLE 4-1: JSON FORMAT

```
{  
  "name": "value"  
}
```

Note: Use a comma to separate when there are more entries.

In [Example 4-2](#), "pid" and "did" are configuration items and 0x1234 and 0x5678 are the new values, respectively.

EXAMPLE 4-2: JSON FORMAT EXAMPLE

```
{  
  "pid": "0x1234",  
  "did": "0x5678"  
}
```

Refer to [Appendix E. "Supported Configuration Items"](#) for the list of supported configuration items.

4.2 BINARY FORMAT

Configuration files in binary format can also be generated and programmed using the MPLAB Connect Configurator GUI tool. This is the format in which data is stored in the Configuration memory and is not human readable. These files are generated and accepted with extension .*cfg* for USB hub products and .*bin* for LAN78xx products.

NOTES:

Chapter 5. USB Hub Devices

5.1 ONLINE HUB CONFIGURATION PAGE

Microchip hubs contain several registers that can change the behavior of the hub. These can be modified to a new value through the MPLAB® Connect Configurator GUI application by programming the configuration memory.

Microchip hub configuration items are divided into the three following categories and are available under three different tabs as shown in [Figure 5-1](#) to [Figure 5-6](#).

- Basic Features (Shown by default)
- Advanced Features
- Special Features

FIGURE 5-1: USB253X/USB(8)460X/USB3X13

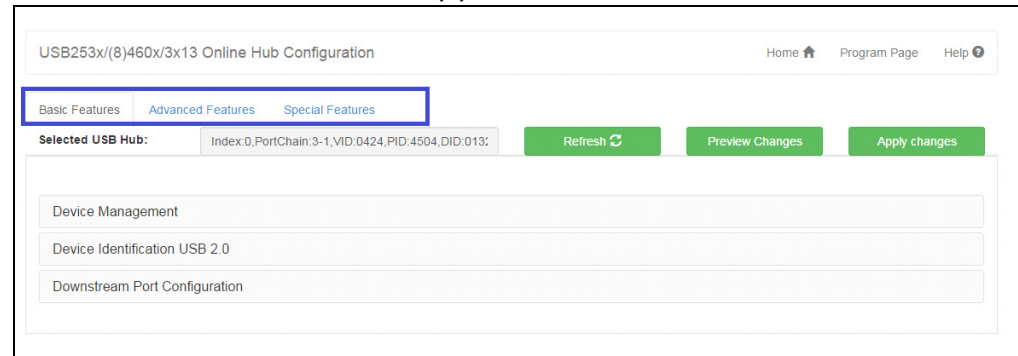


FIGURE 5-2: USB57XX

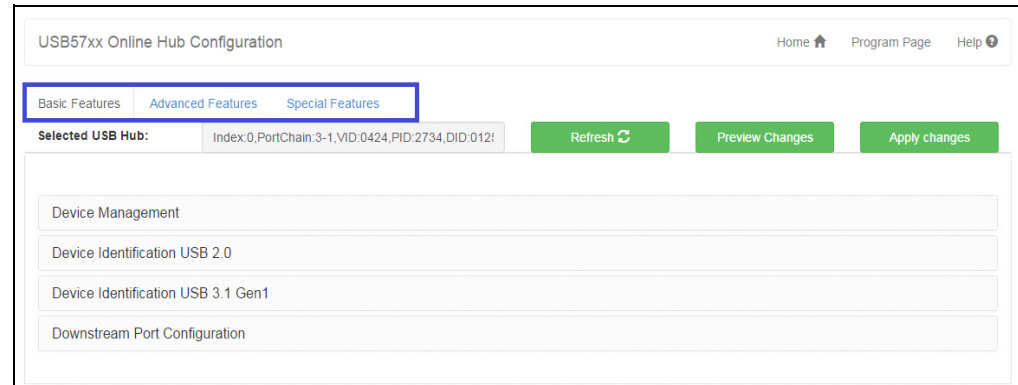


FIGURE 5-3: USB58XX/USB59XX

USB58xx/USB59xx Online Hub Configuration

Home Program Page Help

Basic Features Advanced Features Special Features

Selected USB Hub: Index:0,PortChain:2-4,VID:0424,PID:2807,DID:020f Refresh Preview Changes Apply changes

Device Management

Device Identification USB 2.0

Device Identification USB 3.1 Gen1

Downstream Port Configuration

FIGURE 5-4: USB49XX

USB49xx Online Hub Configuration

Home Program Page Help

Basic Features Advanced Features Special Features

Selected USB Hub: Index:0,PortChain:2-2,VID:0424,PID:4916,DID:010f Refresh Preview Changes Apply changes

Device Management

Device Identification

Downstream Port Configuration

FIGURE 5-5: USB4715

USB4715 Online Hub Configuration

Home Program Page Help

Basic Features Advanced Features Special Features

Selected USB Hub: Index:0,VID:0424,PID:4715,DID:0121 Refresh Preview Changes Apply changes

Device Management

Device Identification

Downstream Port Configuration

FIGURE 5-6: USB70XX

USB70xx Online Hub Configuration

Home Program Page Help

Basic Features Advanced Features Special Features

Selected USB Hub: Index:0,PortChain:3-1,VID:0424,PID:4002,DID:123- Refresh Preview Changes Apply changes

Device Management

Device Identification USB 2.0

Device Identification USB 3.1 Gen1

Downstream Port Configuration

5.1.1 Basic Features

Features that are widely used and modified by many customers are grouped in the **Basic Features** tab.

5.1.1.1 DEVICE MANAGEMENT

FIGURE 5-7: SELECT USB HUB

Figure 5-7 shows the screen displayed once the device family is selected in the landing page. When the MPLAB Connect Configurator GUI is launched, all the GUI elements (in all pages) would remain disabled except the “Select USB Hub,” and the default item selected would be “Hub at index 0”.

The MPLAB Connect Configurator GUI tool would prompt users to connect the available hubs if the family of the hub is not available at index 0 and if multiple hubs of the same family are connected to the host. The MPLAB Connect Configurator GUI tool would prompt users based on the VID and PID added in the `MPLABConnect.ini` file. If the VID/PID of the hub is changed, it must be updated in the `MPLABConnect.ini` file. See Figure 5-8.

FIGURE 5-8: MPLAB CONNECT CONFIRMATION

5.1.1.1.1 INI Format

Example 5-1 to Example 5-5 show the INI format for various Microchip devices:

EXAMPLE 5-1: FOR USB253X/USB (8)460X/USB3X13 FAMILY HUBS

```
[USB253x_(8)460x_3x13_HUBS]
HUB_VID_PID1=0x0424:0x4504
```

EXAMPLE 5-2: FOR USB57XXFAMILY HUBS

```
[USB57xx_HUBS]
HUB_VID_PID1=0x0424:0x2734HUB_VID_PID1=0x0424:0x4504
```

EXAMPLE 5-3: FOR USB58XX/USB59XX FAMILY HUBS

```
[USB58xx_USB59xx_HUBS]
HUB_VID_PID1=0x0424:0x2807
HUB_VID_PID2=0x0424:0x2806
```

EXAMPLE 5-4: FOR USB49XX FAMILY HUBS

```
[USB49xx_HUBS]
HUB_VID_PID1=0x0424:0x4916
HUB_VID_PID2=0x0424:0x4715
```

EXAMPLE 5-5: FOR USB70XX FAMILY HUBS

```
[USB70xx_HUBS]
HUB_VID_PID1=0x0424:0x4002
HUB_VID_PID2=0x0424:0x4050
```

Note: Do not duplicate the entries of the VID and PID of the hub. Otherwise, the tool would interpret it as multiple hubs connected with the same VID and PID.

- Auto Identified Part Number

This provides information about the devices' default PID. The Auto-identification is independent of VID/PID changes.

- Configuration Memory and Firmware Memory

There can be only two possibilities as follows:

- Firmware is running from ROM. Configuration is always loaded from OTP.
- Firmware is running from SPI flash. Configuration is always loaded from SPI.

- Number of Configurations Programmed

The number of configurations programmed is displayed based on the data read from the hub. The tool allows programming as long as there is free space in the configuration memory.

- Configuration Memory Usage

The configuration space usage progress bar shows the percentage of configuration memory used.

Color coding is based on the usage percentage. For example:

- Green, if usage is less than 50%
- Yellow, if usage is between 50% and 75%
- Red, if usage is greater than 75%

- Restore Factory Defaults

Restore factory defaults is the restoration of a hub to its original system state by erasing all of the information stored on the device in an attempt to restore the device to its original manufacturer settings.

For USB253x SKU, if the number of configurations programmed is 16+ and running from ROM, then MPLAB Connect Configurator GUI would ask for the default SKU to restore. See [Figure 5-9](#).

FIGURE 5-9: DEFAULT SKU

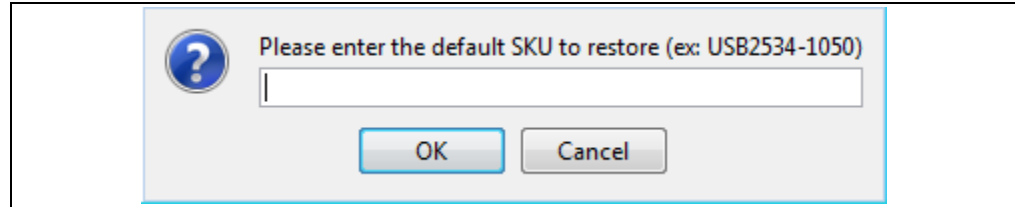


Figure 5-9 shows a dialog box with a question mark icon. The text inside says "Please enter the default SKU to restore (ex: USB2534-1050)". There is a text input field below the text, and two buttons labeled "OK" and "Cancel" at the bottom.

5.1.1.2 DEVICE IDENTIFICATION USB 2.0

USB 2.0 Vendor ID, Product ID, bcdDevice, bcdUSB, and string descriptors can be customized by writing appropriate values. See [Figure 5-10](#).

FIGURE 5-10: .DEVICE IDENTIFICATION USB 2.0

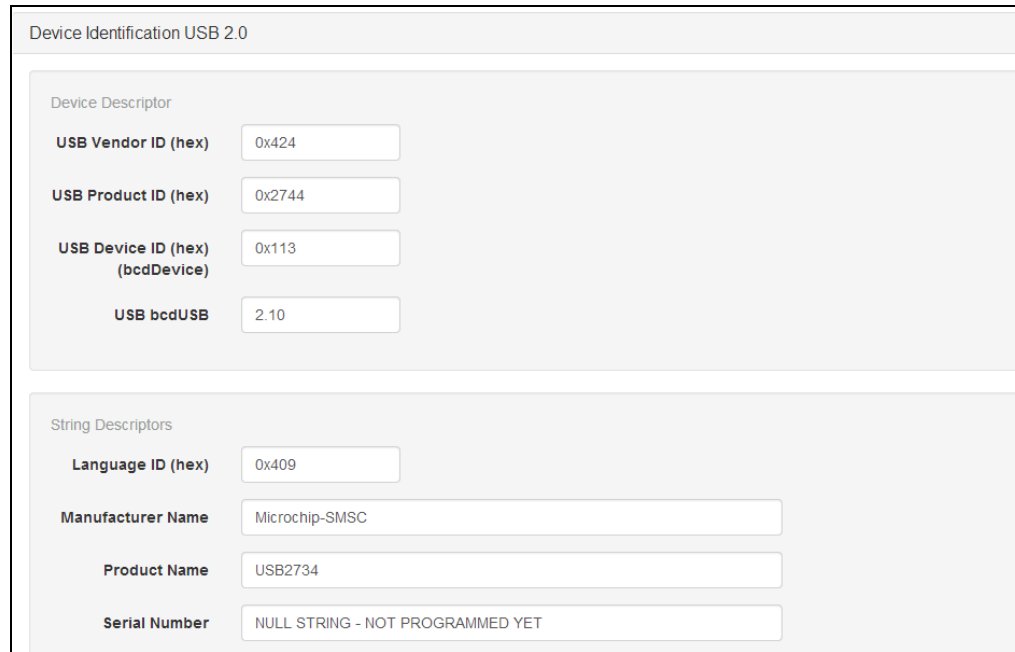


Figure 5-10 shows the "Device Identification USB 2.0" window. It is divided into two main sections: "Device Descriptor" and "String Descriptors".

Device Descriptor:

- USB Vendor ID (hex): 0x424
- USB Product ID (hex): 0x2744
- USB Device ID (hex) (bcdDevice): 0x113
- USB bcdUSB: 2.10

String Descriptors:

- Language ID (hex): 0x409
- Manufacturer Name: Microchip-SMSC
- Product Name: USB2734
- Serial Number: NULL STRING - NOT PROGRAMMED YET

The USB 2.0 String can be disabled by programming an empty string, as shown in [Figure 5-11](#). USB 3.1 Gen1 and HFC strings can also be disabled by this method.

FIGURE 5-11: STRING DESCRIPTORS

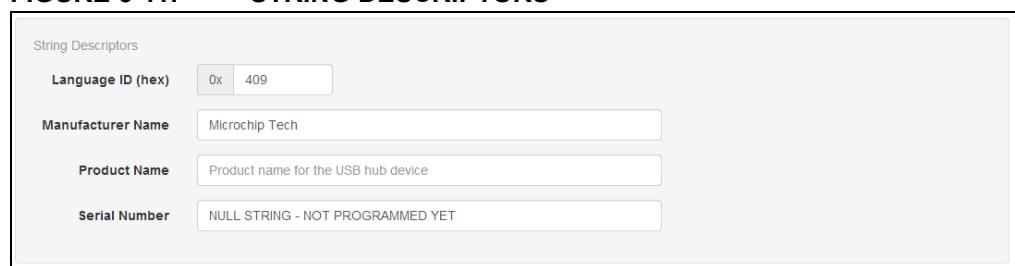


Figure 5-11 shows the "String Descriptors" window. It contains the following fields:

- Language ID (hex): 0x 409
- Manufacturer Name: Microchip Tech
- Product Name: Product name for the USB hub device
- Serial Number: NULL STRING - NOT PROGRAMMED YET

FIGURE 5-12: USB492X

Device Management

Device Identification

Primary Hub

Secondary Hub

Downstream Port Configuration

Device Identification for the USB492x SKU family would be different from other SKUs and provides the following widgets: Primary Hub Vendor ID, Primary Hub Product ID, Primary Hub bcdDevice, Primary Hub bcdUSB and Primary Hub String Descriptors, Primary Hub UUID, Secondary Hub Vendor ID, Secondary Hub Product ID, Secondary Hub bcdDevice, Secondary Hub bcdUSB and Secondary Hub String Descriptors, and Secondary Hub UUID. See [Figure 5-12](#) to [Figure 5-16](#).

FIGURE 5-13: PRIMARY HUB

Primary Hub

Device Descriptor

USB Vendor ID (hex)

0x424

USB Product ID (hex)

0x4925

USB Device ID (hex) (bcdDevice)

0x121

USB bcdUSB

0x2.01

String Descriptors

Language ID (hex)

0x409

Manufacturer Name

Microchip Tech

Product Name

USB4925

Serial Number

NULL STRING - NOT PROGRAMMED YET

FIGURE 5-14: PRIMARY HUB (CONTINUED)

BOS Descriptors

UUID (hex)

0x100f0e0d0c0b0a090807060504030201

FIGURE 5-15: SECONDARY HUB

Secondary Hub

Device Descriptor

USB Vendor ID (hex) 0x 424

USB Product ID (hex) 0x 4931

USB Device ID (hex) (bcdDevice) 0x 121

USB bcdUSB 0x 2.01

String Descriptors

Language ID (hex) 0x 409

Manufacturer Name Microchip Tech

Product Name USB4931

Serial Number NULL STRING - NOT PROGRAMMED YET

FIGURE 5-16: SECONDARY HUB (CONTINUED)

BOS Descriptors

UUID (hex) 0x 100f0e0d0c0b0a090807060504030201

5.1.1.3 DEVICE IDENTIFICATION USB 3.1 GEN1

“Device Identification USB3.1 Gen1” is displayed for all the USB 3.0 hubs, namely USB57xx, USB70xx, and USB58xx/USB59xx. USB 3.1 Gen1 Vendor ID, Product ID, bcdDevice, USBbcd, String Descriptors, and UUID can be customized by writing appropriate values. See [Figure 5-17](#) to [Figure 5-19](#).

FIGURE 5-17: DEVICE IDENTIFICATION USB 3.1 GEN1

Device Identification USB 3.1 Gen1

Device Descriptor

USB 3.1 Gen1 Vendor ID (hex) 0x 424

USB 3.1 Gen1 Product ID (hex) 0x 5744

USB 3.1 Gen1 Device ID (hex) (bcdDevice) 0x 202

USB 3.1 Gen1 bcdUSB 0x 3.00

FIGURE 5-18: DEVICE IDENTIFICATION USB 3.1 GEN1 (CONTINUED)

String Descriptors

Language ID (hex)

0x409

Manufacturer Name

Microchip Tech

Product Name

USB5734

Serial Number

NULL STRING - NOT PROGRAMMED YET

FIGURE 5-19: DEVICE IDENTIFICATION USB 3.1 GEN1 (CONTINUED)

Other Descriptors

UUID (hex)

0x10010e0d0c0b0a090807060504030201

5.1.1.4 DOWNSTREAM PORT CONFIGURATION

Based on the selected hub, downstream ports are displayed.
For example, the USB253x, USB(8)460x, and USB57xx have four downstream ports
See [Figure 5-20](#).

FIGURE 5-20: DOWNSTREAM 4-PORT CONFIGURATION

Downstream Port Configuration

Port 1

Port 2

Port 3

Port 4

The USB3X13 (USB3613 and USB3813) only have three downstream ports. (See [Figure 5-21](#).)The upstream port and the third downstream port of USB3613 are HSIC enabled. In the USB3813, the first downstream port is HSIC enabled.

FIGURE 5-21: DOWNSTREAM 3-PORT CONFIGURATION

Downstream Port Configuration

Port 1

Port 2

Port 3

5.1.1.4.1 Port Configuration

- Basic downstream port configurations are:
1. Port – Enable/Disable
 2. Non-Removable Port Feature – Enable/Disable
 3. Battery Charging
- See [Figure 5-22](#).

FIGURE 5-22: DOWNSTREAM PORT CONFIGURATION



5.1.1.4.2 Port Enable/Disable

A physical port can be enabled/disabled through the GUI. (See [Figure 5-23.](#)) If any one of the physical ports is enabled or disabled, the GUI would remap appropriate logical ports internally.

FIGURE 5-23: PORT ENABLE/DISABLE



5.1.1.4.3 Non-Removable Ports

USB 2.0 and USB 3.0 ports can be configured as non-removable ports.

- If the check box is not checked, the port is removable.
- If the check box is checked, the port is non-removable.

This configuration item informs the Host that one of the active ports has a permanent device that is not detachable from the hub.

5.1.1.4.4 Battery Charging

- [USB2.0 Standard USB Port- Maximum 500 mA](#)
- [BC 1.2-Compliant Port- Maximum 1.5A](#)
- [Most Devices-Supported Mode-Maximum 2A](#)

- USB2.0 Standard USB Port- Maximum 500 mA

The “USB 2.0 Standard USB Port-Maximum 500 mA” is the default battery charging method. This also means battery charging is disabled and the device can draw only 500 mA, which is the standard for a USB device.

- BC 1.2-Compliant Port- Maximum 1.5A

When there is no upstream VBUS, and consequently no USB host connected to the upstream port, the downstream battery charging-enabled ports would operate as “BC 1.2-Compliant Port-Maximum 1.5A.”

- Most Devices-Supported Mode-Maximum 2A

When there is an upstream VBUS and an upstream connection, the downstream battery charging-enabled ports operate as “Most Devices-supported Mode-Max 2A.”

Battery charging is supported for disabled ports in USB57xx hubs and is not supported for disabled ports in USB253x/USB(8)4604 hubs.

In HSIC devices (USB3x13), if a downstream port is HSIC enabled, battery charging is not supported on that port.

5.1.2 Advanced Features

MPLAB Connect Configurator GUI also allows to change some advanced configurations and dump memory. See [Figure 5-24](#) to [Figure 5-27](#).

FIGURE 5-24: USB57XX

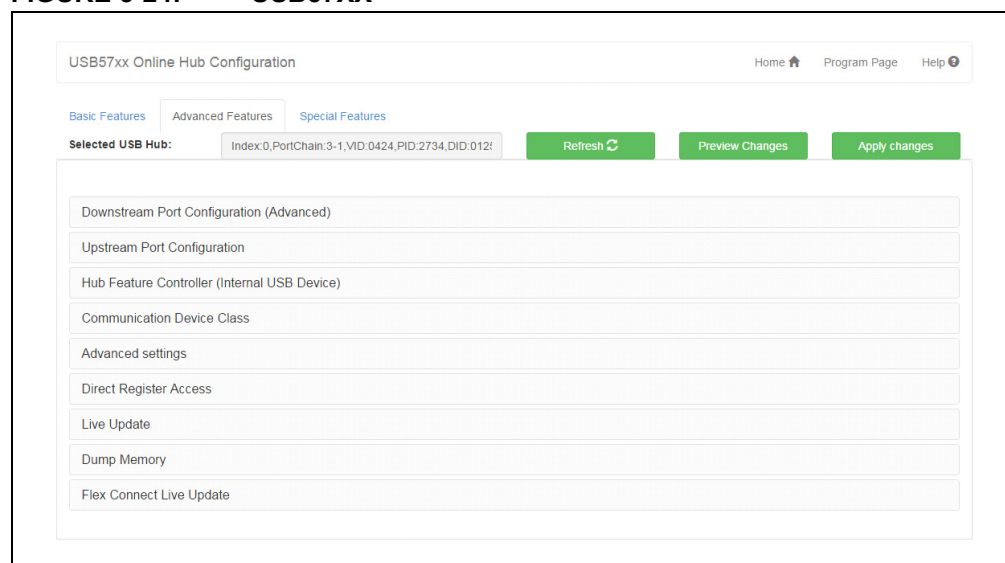


FIGURE 5-25: USB253X/USB(8)4604/USB3X13

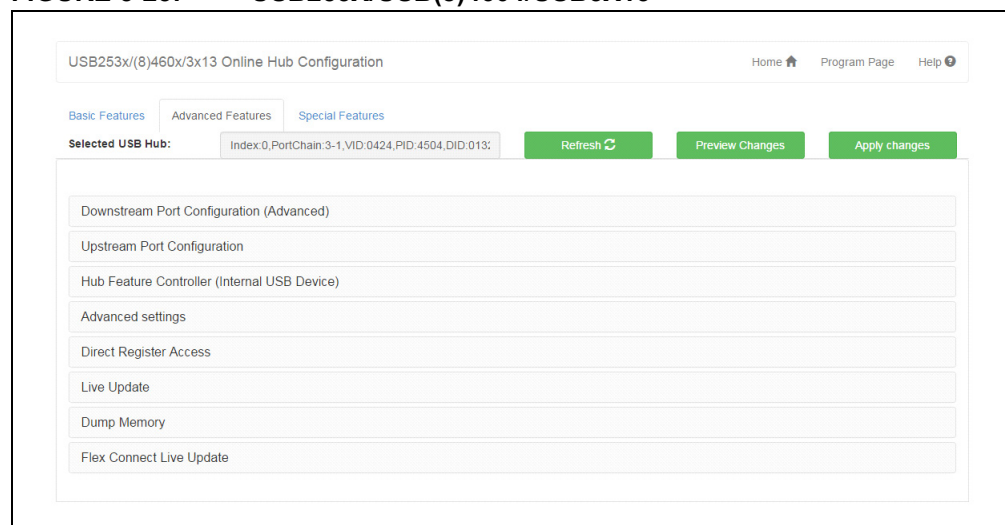


FIGURE 5-26: USB491X

USB49xx Online Hub Configuration

Home Program Page Help

Basic Features Advanced Features Special Features

Selected USB Hub: Index:0,VID:0424,PID:4914,DID:0121 Refresh Preview Changes Apply changes

Downstream Port Configuration (Advanced)

Upstream Port Configuration

Hub Feature Controller (Internal USB Device)

NCM Device

I2S Interface Configuration

Advanced settings

Direct Register Access

Live Update

Dump Memory

FIGURE 5-27: USB70XX

USB70xx Online Hub Configuration

Home Program Page Help

Basic Features Advanced Features Special Features

Selected USB Hub: Index:0,PortChain:3-1,VID:0424,PID:4002,DID:123- Refresh Preview Changes Apply changes

Downstream Port Configuration (Advanced)

Upstream Port Configuration

Hub Controller (Internal USB Device)

NCM Device

Communication Device Class (CDC)

I2S Interface Configuration

Advanced settings

MiniHost

Direct Register Access

Live Update

Dump Memory

5.1.2.1 DOWNSTREAM PORT CONFIGURATION (ADVANCED)

5.1.2.1.1 Swap D+/D-

USB2.0 DP and DM Pins can be swapped using the GUI.

5.1.2.1.2 PHY Boost and Varisense

USB signal drive strength in upstream and downstream transceivers can be changed using PHY Boost. USB signal receiver sensitivity can be changed by using Varisense. Downstream port PHY Boost and Varisense values can be changed by using the GUI, as shown in [Figure 5-28](#) and [Figure 5-29](#).

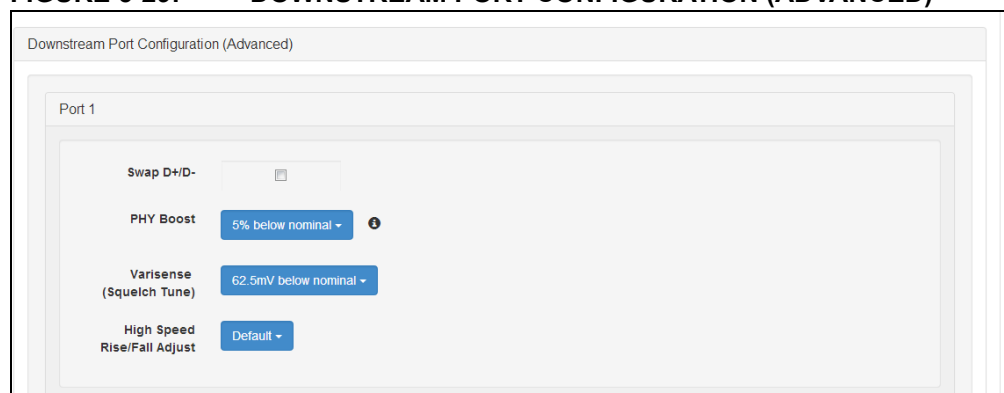
5.1.2.1.3 High-Speed Rise/Fall Adjust

Rise/Fall time for high-speed devices vary based on the cable length. This can be changed using the “High-Speed Rise/Fall Adjust”.

FIGURE 5-28: DOWNSTREAM PORT CONFIGURATION (ADVANCED)



FIGURE 5-29: DOWNSTREAM PORT CONFIGURATION (ADVANCED)



5.1.2.2 HSIC-RELATED CONFIGURATION

The following are the HSIC-related configurations which can be performed on HSIC-enabled ports (See [Figure 5-30](#) and [Figure 5-31](#).):

5.1.2.2.1 Swap HSIC DATA/STROBE

HSIC Data and Strobe Pins can be swapped using the GUI.

5.1.2.2.2 HSIC Driver Output Impedance

HSIC Driver Output Impedance can be configured to one of the options given below using the GUI:

- 40 ohm
- 50 ohm

5.1.2.2.3 Slew Tune

Similarly, Slew Tune in HSIC-enabled ports can be configured as shown below:

- Default
- +30%

FIGURE 5-30: DOWNSTREAM PORT HSIC CONFIGURATION (ADVANCED)

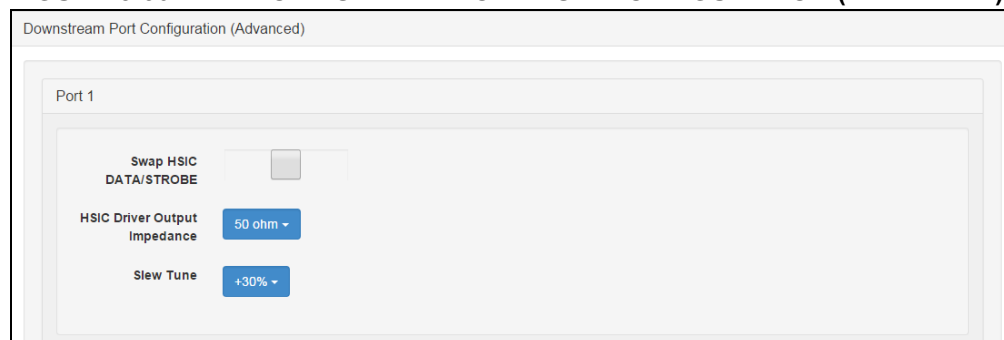
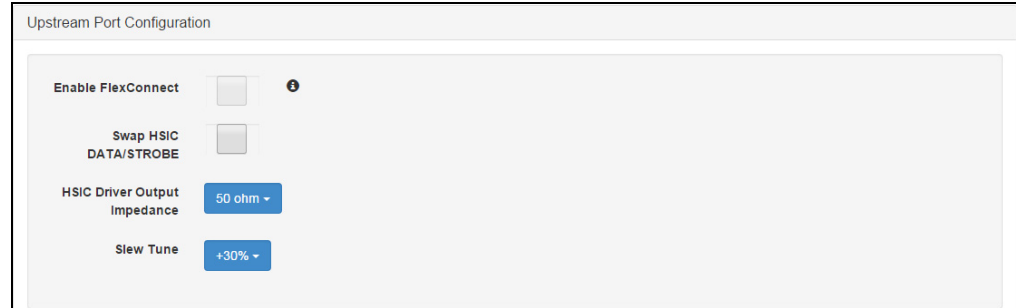


FIGURE 5-31: UPSTREAM PORT HSIC CONFIGURATION



5.1.2.3 UPSTREAM PORT CONFIGURATION

Features like Flexconnect, Swap D+/D-, PHY Boost, Varisense, High-Speed Rise/Fall adjust are available under Upstream Port Configuration. See [Figure 5-33](#) and [Figure 5-34](#).

5.1.2.3.1 Flex Connect

USB 2.0/ USB 3.1 Gen1 downstream port 1 can be configured as an upstream port.

- If check box is not checked, the upstream port is not Flex connected. (Default behavior)
- If check box is checked, the upstream port is Flex connected, meaning the upstream port is the downstream port and downstream port 1 is the upstream port.

Note: MPLAB Connect Configurator GUI does not provide Flex Connect option through OTP/Pseudo OTP.

FIGURE 5-32: USB4715



5.1.2.3.2 Flex Port

For the USB4715, FlexConnect can be enabled for any of the downstream ports. If the “FlexConnect” checkbox is checked, flexport would denote the downstream port with which the flex has happened. See [Figure 5-32](#).

5.1.2.3.3 Swap D+/D-

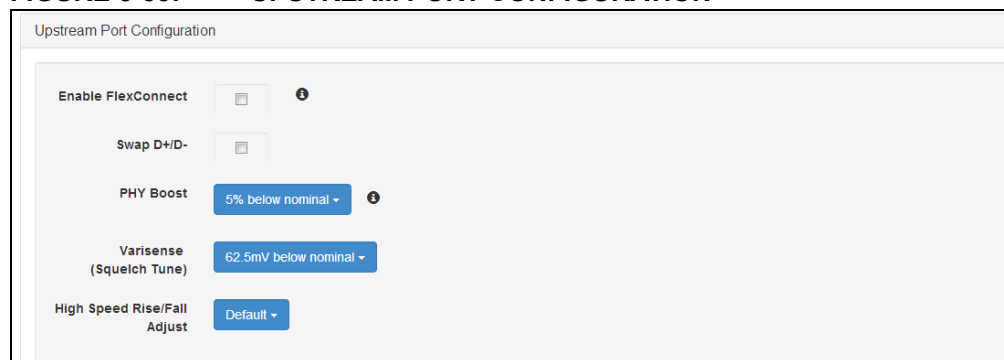
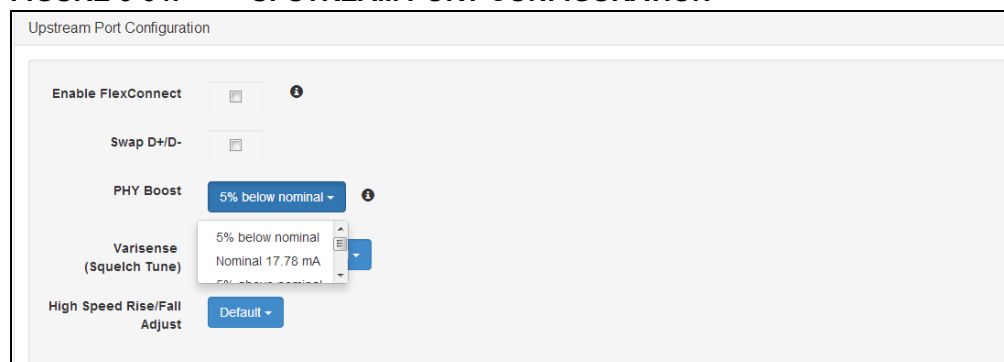
USB 2.0 Upstream DP and DM Pins can be swapped using the GUI.

5.1.2.3.4 PHY Boost and Varisense

USB signal drive strength in upstream and downstream transceivers can be changed using PHY Boost. USB signal receiver sensitivity can be changed by using Varisense. Downstream Ports PHY Boost and Varisense values can be changed by using GUI as shown below.

5.1.2.3.5 High-Speed Rise/Fall Adjust

Rise/Fall time for high-speed device varies based on the cable length. This can be changed using “High Speed Rise/Fall Adjust”.

FIGURE 5-33: UPSTREAM PORT CONFIGURATION

FIGURE 5-34: UPSTREAM PORT CONFIGURATION


5.1.2.4 HFC (INTERNAL USB DEVICE)

5.1.2.4.1 Enable HFC

The Hub Feature Controller (HFC) device is used to access USB Device. HFC may always be enabled by using the GUI and programming the configuration memory.

- If check box is not checked, the HFC is not enabled.
- If check box is checked, the HFC is always enabled.

The HFC Vendor ID, Product ID, bcdDevice, and string descriptors may be customized by writing appropriate values. See [Figure 5-35](#).

Note: If the VID or PID of the HFC device is changed from the default value, the new VID and PID needs to be added into the INI file. PID should be added to "HCE_DEV_INFO" and VID should be added to "HUB_VID_LIST" in the INI file. If the default Vendor ID/Product ID is changed for the hub controller, the Microchip hub controller driver would not be loaded for the hub controller and a new WinUSB driver package should be generated for the hub controller.

FIGURE 5-35: HUB CONTROLLER

Hub Controller (Internal USB Device)

Enable Hub Controller ☒

Device Descriptor

USB Vendor ID (hex) 0x 424 ⓘ

USB Product ID (hex) 0x 2740 ⓘ

USB bcdDevice (hex) 0x 200

String Descriptors

Language ID (hex) 0x 409

Manufacturer Name Microchip Tech

Product Name Hub Controller

Note: The automatic and manual WinUSB driver installation would not work if the hub controller Vendor ID and Product ID are changed.

5.1.2.5 NCM DEVICE

The NCM Vendor ID, Product ID, bcdDevice and string descriptors may be customized by writing appropriate values. See [Figure 5-36](#) to [Figure 5-38](#).

FIGURE 5-36: USB491X, USB7002, AND USB7050

NCM Device

NCM Device 1

NCM Device 2

FIGURE 5-37: USB491X, USB7002, AND USB7050 (CONTINUED)

NCM Device 1

Device Descriptor

USB Vendor ID (hex) 0x 424 ⓘ

USB Product ID (hex) 0x 4910 ⓘ

USB bcdDevice (hex) 0x 121

String Descriptors

Language ID (hex) 0x 409

Manufacturer Name Microchip Tech

Product Name USB49XX NCM/IAP Bridge

Serial Number NULL STRING - NOT PROGRAMMED YET

MAC Address 02:00:00:00:00:01

FIGURE 5-38: USB491X, USB7002, AND USB7050 (CONTINUED)

NCM Device 2

Device Descriptor

USB Vendor ID (hex)0x424 ⓘ

USB Product ID (hex)0x4920 ⓘ

USB bcdDevice (hex)0x121

String Descriptors

Language ID (hex)0x409

Manufacturer NameMicrochip Tech

Product NameUSB49XX NCM/IAP Bridge

Serial NumberNULL STRING - NOT PROGRAMMED YET

MAC Address02:00:00:00:00:02

The USB491x, USB7002, and USB7050 have two NCM devices—NCM Device 1 and NCM Device 2—used for bridging purpose. (See [Figure 5-39](#).) The following widgets are available for the user to configure: NCM Device 1 Vendor ID, NCM Device 1 Product ID, NCM Device 1 bcdDevice, NCM Device 1 bcdUSB, and NCM Device 1 String Descriptors, NCM Device 2 Vendor ID, NCM Device 2 Product ID, NCM Device 2 bcdDevice, NCM Device 2 bcdUSB, and NCM Device 2 String Descriptors.

NCM widgets other than MAC address are configurable only through “Hub Controller.” Hub controller values can be copied to NCM Device 1 and NCM Device 2 by selecting the option in [Figure 5-40](#).

FIGURE 5-39: COPY VALUES FROM HUB CONTROLLER UNCHECKED

NCM Device

Copy Values from Hub controller☐

NCM Device 1

NCM Device 2

FIGURE 5-40: COPY VALUES FROM HUB CONTROLLER CHECKED

Copy Values from Hub controller ☒

NCM Device 1

Device Descriptor

USB Vendor ID (hex) 0x 424 ⓘ

USB Product ID (hex) 0x 494a ⓘ

USB bcdDevice (hex) 0x 121

String Descriptors

Language ID (hex) 0x 409

Manufacturer Name Microchip Tech

Product Name USB2 Controller Hub

Serial Number NULL STRING - NOT PROGRAMMED YET

5.1.2.6 I²S INTERFACE CONFIGURATION

FIGURE 5-41: USB49XX/USB4715 AND USB70XX

I2S Interface Configuration

I2S_FEATURE_UNIT_SEL

☐ Disable ⓘ

☐ Enable Audio IN

☐ Enable Audio OUT

☒ Enable Audio IN and Audio OUT

HID_FEATURE_UNIT_SEL

☒ Disable ⓘ

☐ Audio IN Control

☐ Audio OUT Control

☐ Audio IN Control and Audio OUT Control

5.1.2.6.1 I2S_FEATURE_UNIT_SEL

This field controls the I²S interface features, if enabled through programmable function pins.

- Disable – I²S is completely disabled.
- Enable Audio IN – Audio IN is enabled.
- Enable Audio OUT – Audio OUT is enabled.
- Enable Audio IN and Audio OUT – Audio IN and Audio OUT are enabled. (See [Figure 5-41](#).)

5.1.2.6.2 HID_FEATURE_UNIT_SEL

This field controls what units are governed by the HID interface.

- Disable – HID is disabled.
- Audio IN Control – HID interface controls only Audio IN.
- Audio OUT Control – HID interface controls only Audio OUT.
- Audio IN Control and Audio OUT Control – HID interface controls both Audio IN and Audio OUT. (See [Figure 5-41](#).)

5.1.2.7 COMMUNICATION DEVICE CLASS (CDC)

- If the check box is not checked, CDC is always not enabled. (See [Figure 5-42](#).)
- If the check box is checked, CDC is always enabled. When the WinUSB interface is present, CDC enumerates at interface 1 or interface 0.

FIGURE 5-42: CDC



Note: CDC is not available in the USB253x/USB(8)4604.

5.1.2.7.1 For USB49xx/USB4715 and USB70xx

CDC Enable is based on Strap.

5.1.2.8 ADVANCED SETTINGS

The MPLAB Connect Configurator GUI tool allows to change Hub Power mode, Maximum power, LPM, Full-Speed mode under advanced settings. See [Figure 5-43](#) to [Figure 5-45](#).

5.1.2.8.1 Hub Power Mode

The Hub Power mode can be configured by using this GUI. If the hub is self-powered, then the external power supply is applied to the hub. If the hub is bus-powered, then power is obtained from the USB port to which the hub is connected.

FIGURE 5-43: USB57XX

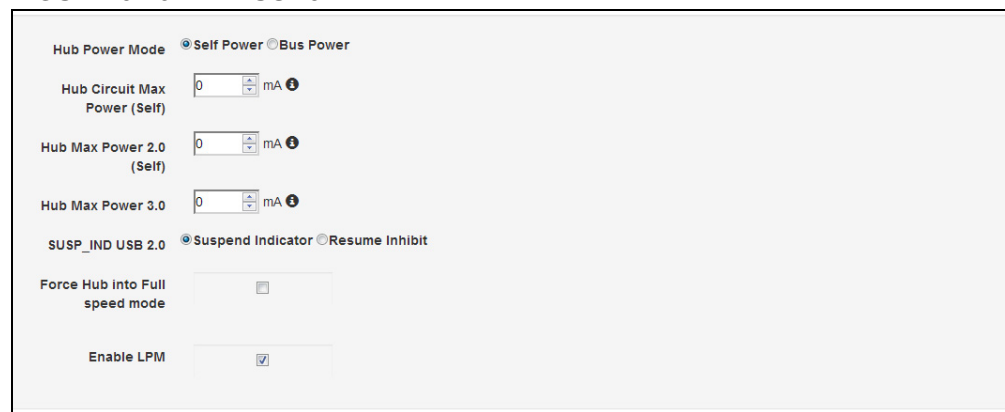


FIGURE 5-44: USB253X/USB(8)4604

Hub Power Mode ☒ Self Power ☐ Bus Power

Hub Circuit Max Power (Self) 1 mA ⓘ

Hub Max Power 2.0 (Self) 2 mA ⓘ

Upstream Charger Detection ☒ Disabled ☐ All supported Chargers ☐ All SE1 chargers ☐ CDP charger ☐ DCP charger

SUSP_IND USB 2.0 ☒ Disable ☐ Suspend Indicator ☐ Resume Inhibit

Force Hub into Full speed mode ☐

Enable LPM ☐

FIGURE 5-45: USB49XX /USB4715 AND USB70XX

Hub Power Mode ☒ Self Power ☐ Bus Power

Hub MaxCurrent (Self) 0 mA ⓘ

Hub MaxPower 2.0 (Self) 0 mA ⓘ

Power-On Time 100 ms ⓘ

Force Hub into Full speed mode ☐

Enable LPM ☐

5.1.2.8.2 Hub Maximum Power and Hub Circuit Maximum Power

If the Hub Power mode is configured as a self-powered device, then less than 1 mA of upstream VBUS current is consumed, and all ports are available with each port being capable of sourcing 500 mA of current. If the Hub Power mode is configured as a bus-powered device, the hub consumes less than 100 mA of current prior to being configured. After configuration, the bus-powered hub (along with all associated hub circuitry, any embedded devices that are part of a compound device, and a 100 mA per externally available downstream port) must consume no more than 500 mA of the upstream VBUS current. The hub circuit maximum power field gets input and displays output for the USB 2.0 Hub. The hub circuit maximum power for the USB 3.1 Gen1 Hub is updated based on input given to this field. See [Figure 5-46](#).

FIGURE 5-46: HUB MAXIMUM POWER AND HUB CIRCUIT MAXIMUM POWER

Hub Circuit Max Power (Self) 0 mA ⓘ

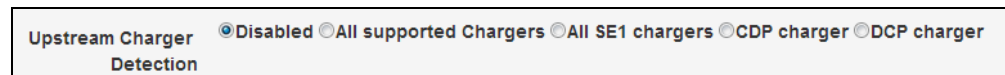
Hub Max Power 2.0 (Self) 0 mA ⓘ

Hub Max Power 3.0 0 mA ⓘ

5.1.2.8.3 Upstream Charger Detection

Battery charger detection is available in USB253x/USB(8)4604 hubs on the upstream facing port. The detection sequence identifies chargers which conform to the Chinese battery charger specification, chargers which conform to the USB-IF Battery Charger Specification 1.2, and single-ended 1 charger (SE1). See [Figure 5-47](#).

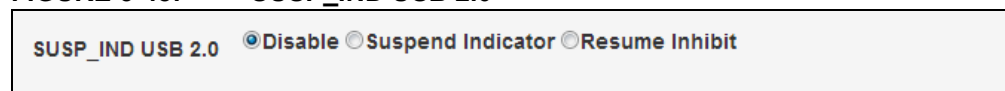
FIGURE 5-47: UPSTREAM CHARGER DETECTION



5.1.2.8.4 SUSP_IND USB 2.0

If the suspend indicator is enabled, the device is configured and is active (not in suspend). When resume inhibit is enabled and if the hub detects any wakeup event, the suspend indication would be driven for a time. The SUSP_IND pin can be configured by using GUI. See [Figure 5-48](#).

FIGURE 5-48: SUSP_IND USB 2.0



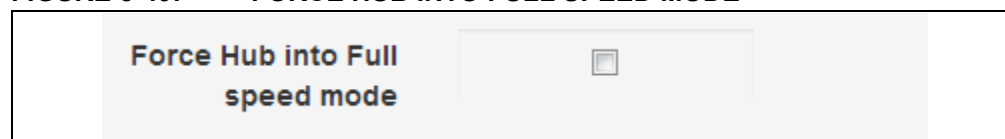
Note: “Disable” option is not available for USB57xx Hubs.

5.1.2.8.5 Force Hub into Full-Speed Mode

The speed of the device changes to 12 Mbit/s by configuring the USB full-speed device in the GUI. See [Figure 5-49](#).

- If check box is not checked, the USB full-speed device is disabled.
- If check box is checked, the USB full-speed device is enabled.

FIGURE 5-49: FORCE HUB INTO FULL SPEED MODE



5.1.2.8.6 Link Power Management (LPM)

Low transitional latencies can be offered by enabling LPM using the GUI. See [Figure 5-50](#).

FIGURE 5-50: LPM



5.1.2.8.7 Power-On Time

The PwrOn2PwrGood field of class-specific hub descriptor is the period (in 2 ms intervals) from the time the power-on sequence begins on a port until power is good on that port. See [Figure 5-51](#).

FIGURE 5-51: POWER-ON TIME

The screenshot shows a configuration field labeled "Power-On Time" with a text input box containing the number "100" and a unit selector set to "ms". An information icon is visible to the right of the unit selector.

5.1.2.9 MINIHOST

5.1.2.9.1 USB70xx

Minihost rules such as VID PID, VID PID Mask, and Device class can be programmed. If any Minihost rules are already programmed in configuration area, then the rules are displayed in the “Rules Configured” area. See [Figure 5-52](#).

FIGURE 5-52: MINIHOST

The screenshot shows a scrollable area titled "Rules Configured". It contains a list of "VID PID Rules" with the following details:

- Rule Index:1, VID:0x0424, PID:0x7050, Connection type:USB 1.1 DFP, Rule Id:0x00
- Rule Index:2, VID:0x4051, PID:0x7051, Connection type:USB 1.1 DFP, Rule Id:0x01
- Rule Index:3, VID:0x0202, PID:0x7053, Connection type:USB 1.1 DFP, Rule Id:0x02

Below this list, there is a section for "VID PID Mask Rules" which is partially visible.

New minihost rule can be created or existing minihost rule can be overwritten.

Click to add this Mini Host rule button is used to add more than one rule. A maximum of 20 rules can be added for VID PID rule. A maximum of 10 rules can be added for VID PID Mask rule, and a maximum of 5 rules can be added for Device Class rule. See [Figure 5-53](#) to [Figure 5-55](#).

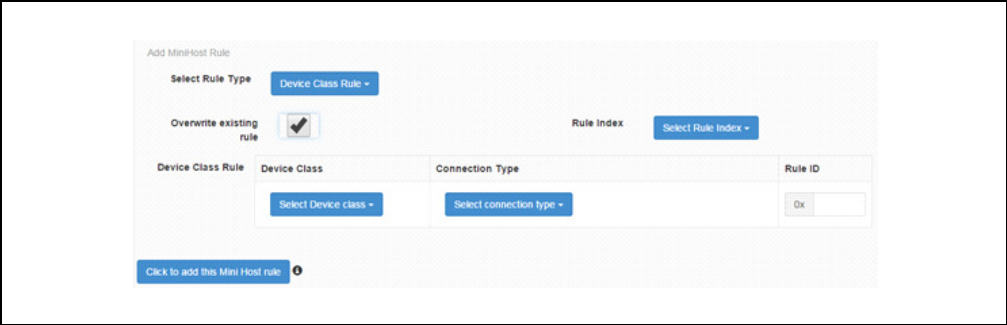
FIGURE 5-53: VID PID RULE

The screenshot shows the "Add Minihost Rule" form. The "Select Rule Type" dropdown is set to "VID PID Rule". The "Overwrite existing rule" checkbox is checked. The "Rule Index" dropdown is set to "Select Rule Index". The form includes input fields for VID (0x), PID (0x), Connection Type (Select connection type), and Rule ID (0x). A blue button labeled "Click to add this Mini Host rule" is at the bottom.

FIGURE 5-54: VID PID MASK RULE

The screenshot shows the "Add Minihost Rule" form. The "Select Rule Type" dropdown is set to "VID PID Mask Rule". The "Overwrite existing rule" checkbox is checked. The "Rule Index" dropdown is set to "Select Rule Index". The form includes input fields for VID (0x), PID (0x), VID PID Mask (0x), Connection Type (Select connection type), and Rule ID (0x). A blue button labeled "Click to add this Mini Host rule" is at the bottom.

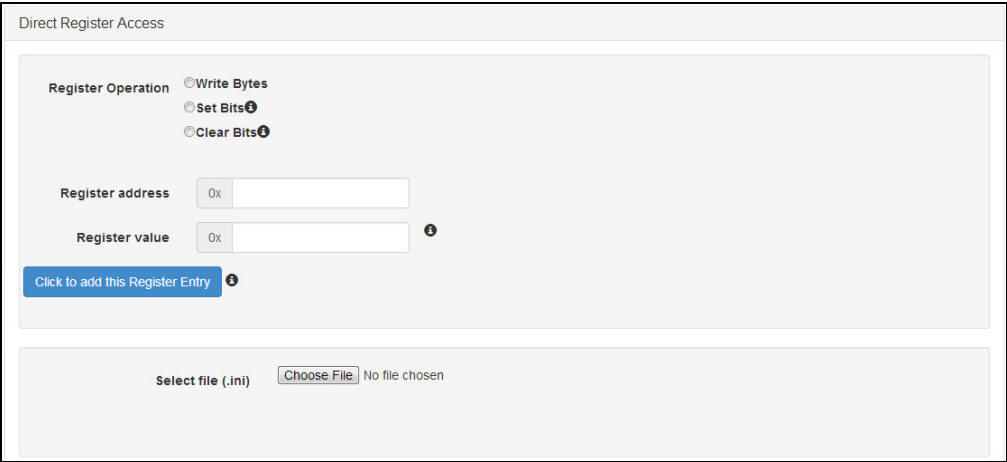
FIGURE 5-55: DEVICE CLASS RULE



5.1.2.10 DIRECT REGISTER ACCESS

The XDATA Address can be programmed directly by using the “Direct Register Access” option. This method follows the big-endian format. See [Figure 5-56](#).

FIGURE 5-56: DIRECT REGISTER ACCESS



Example:

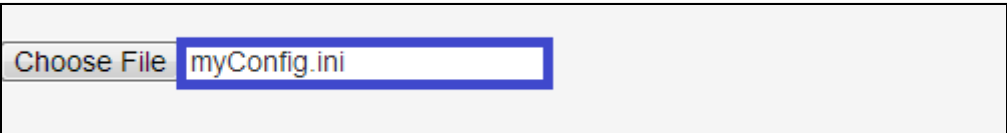
Write value 1234 in register 0x3000 as shown in [Figure 5-57](#). Here, 12 will be written to 0x3000 and 34 will be written to 0x3001.

FIGURE 5-57: EXAMPLE OF DIRECT REGISTER ACCESS



Alternatively, an.ini file can be given as input. See [Figure 5-58](#).

FIGURE 5-58: EXAMPLE



INI Format:

```
[OTPGEN_CONFIG]
; XDATA Write byte. i.e XDATA[address] = yy
; XWRE_ is the identifier to initiate xdata write byte operation.
; For example to write XDATA[3000]to 1234
XWRE_3000=12
XWRE_3001=34

; for example to write XDATA[3000]to 1234
XWRE_3002=12 34

; XDATA Set bits. i.e XDATA[address] |= yy
; XSET_ is the identifier to initiate xdata set bits operation.
; for example to write XDATA[3000]to 1234
XSET_3000=10 ; sets bit 4 in xdata address 0x3000
XSET_3001=02; sets bit 1 in xdata address 0x3001

XSET_3000=02 14 ; sets bit 1 in xdata address 0x3000 and sets bit 2,4
in xdata address 0x3001
; XDATA Clear bits. i.e XDATA[address] &= (~yy)
; XCLR_ is the identifier to initiate xdata clear bits operation.

XCLR_3000=13 ; clears bit 0,1,4 in xdata address 0x3000
XCLR_3001=01 ; clears bit 2 in xdata address 0x3001

XCLR_3000=01 40 ; clears bit 2 in xdata address 0x3000 and clears bit
5 in xdata address 0x3001
```

- Note 1:** Follow either one of the examples of respective operation.
- 2:** No line breaks should be inserted while specifying multi-byte values.
- 3:** “;” on line start comments for that whole line.
- 4:** No space to be inserted before and after “=”.
- 5:** [OTPGEN_CONFIG] field is mandatory in the beginning of ini file.

5.1.2.11 LIVE UPDATE

Register Read and Write can be done in Live Update by using the GUI. (See [Figure 5-59](#).) The register value would be reflected for a given register address when the “Read” option is selected. Data can be written to a register address by selecting the “Write” option. The number of bytes to be read or written can be mentioned in the “Length” field. Selecting “Advanced Options” allows users to store read values in a file and write values to registers from a given file. The file extension is .regdmp.

FIGURE 5-59: LIVE UPDATE

Live Update

Register address(hex) 0x ⓘ

Length ⓘ

Register Operation ☒ Read ☐ Write

Register value(hex) 0x

Advanced Options ⓘ

Live Update

Examples are shown in [Figure 5-60](#), [Figure 5-61](#), and [Figure 5-62](#).

FIGURE 5-60: TO READ REGISTER VALUE

The screenshot shows the 'Live Update' window with the following fields and controls:

- Register address(hex):** 0x 3000
- Length:** 2
- Register Operation:** ☒ Read ☐ Write
- Register value(hex):** 0x 24 04
- Advanced Options:** ☐
- Live Update:** Button

FIGURE 5-61: ADVANCED OPTION

The screenshot shows the 'Live Update' window with the following fields and controls:

- Register address(hex):** 0x 3000
- Length:** 20
- Register Operation:** ☒ Read ☐ Write
- Register value(hex):** 0x 24 04 04 45 28 01 9b 28 08 20 00 00 02 50 01 50 32 00 00 00
- Advanced Options:** ☒
- Select file (.regdmp):** Choose File RegRead
- Live Update:** Button

FIGURE 5-62: USB49XX/USB4715 AND USB70XX

The screenshot shows the 'Live Update' window with the following fields and controls:

- Register address(hex):** 0x bf803000
- Length:** 12
- Register value(hex):** 0x 24 04 14 49 21 01 9b 28 09 e0 00 00
- Advanced Options:** ☒
- Select file (.regdmp):** Choose File No file chosen
- Read:** Button
- Write:** Button

Values read are stored in the selected file. This file can be used as an input when “Write” option is selected.

5.1.2.12 DUMP MEMORY

Dump memory allows users to dump OTP memory, SPI memory, SPI with Pseudo-OTP memory, or Live registers present in the connected device. OTP Memory Dump is saved in a given file with extension .dump. SPI Memory, SPI with Pseudo OTP Dump, is saved in a given file with extension .bin. Live registers is saved in a file with extension .regdmp. See [Figure 5-63](#) to [Figure 5-65](#).

FIGURE 5-63: DUMP MEMORY

FIGURE 5-64: USB49XX/USB4715 AND USB70XX

FIGURE 5-65: USB49XX/USB4715 AND USB70XX

5.1.2.13 FLEXCONNECT FEATURE

This page is used to issue Flex Feature Command to the hub. Both the USB(8)4604 and USB57xx hubs respond to valid FlexConnect commands. See [Figure 5-66](#).

FIGURE 5-66: FLEXCONNECT FEATURE

5.1.3 Preview Changes

Preview changes allows users to view information such as the number of bytes to be programmed and the configuration memory usage before programming. Users can also save a configuration file without programming. See [Figure 5-67](#) and [Figure 5-68](#).

FIGURE 5-67: PREVIEW CHANGES

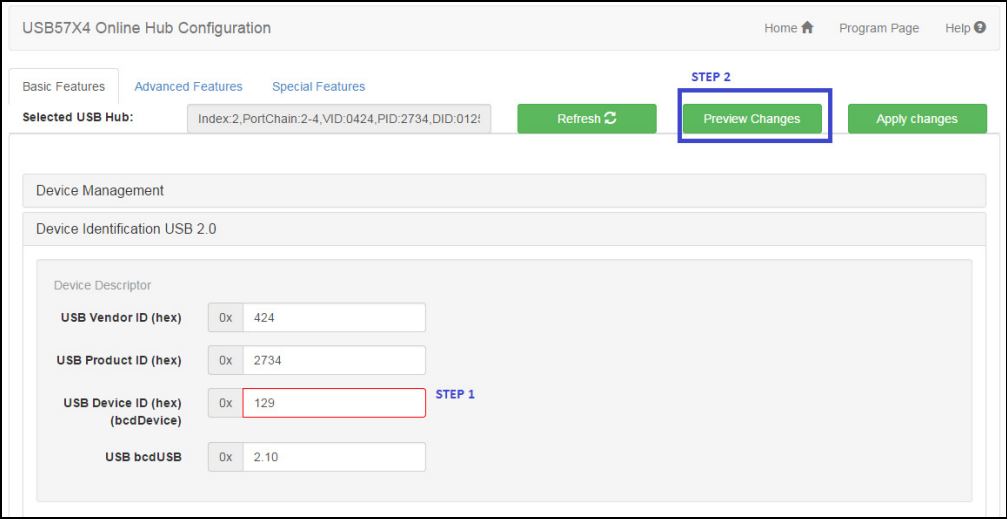
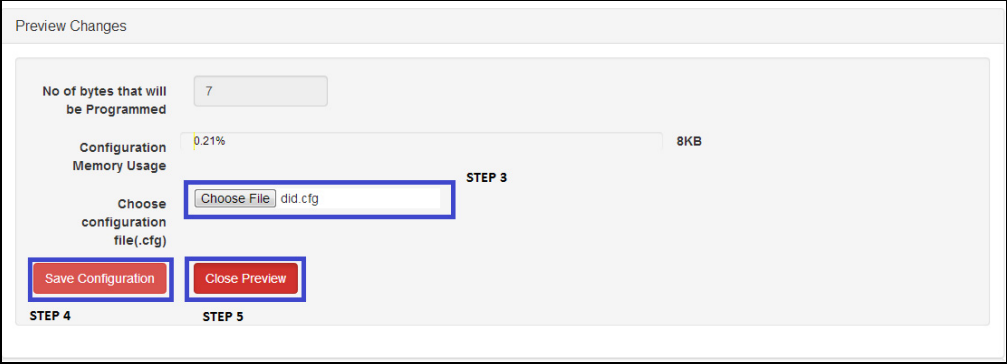


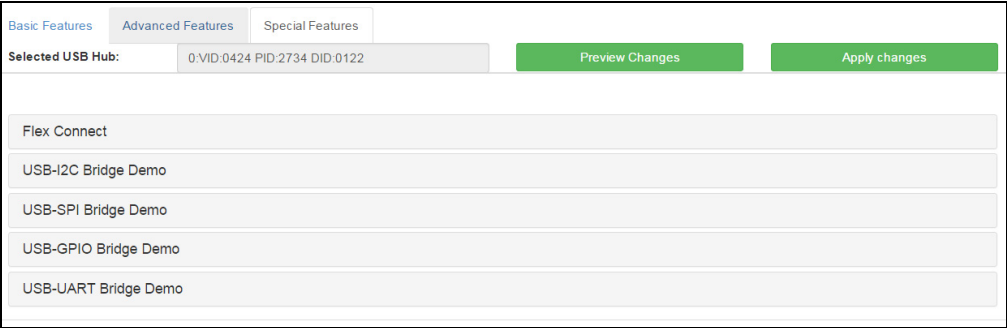
FIGURE 5-68: PREVIEW CHANGES (CONTINUED)



5.1.4 Special Features

The **Special Feature** tab is used for demo purposes. It helps explain the concept of FlexConnect, I²C Bridging, SPI Bridging, GPIO Bridging, and UART Bridging. UART Bridging is available only for the USB2530 hub. See [Figure 5-69](#).

FIGURE 5-69: SPECIAL FEATURES



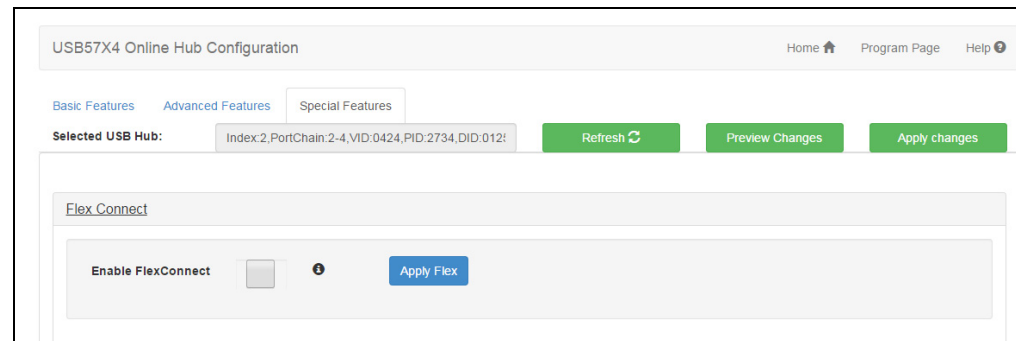
5.1.4.1 ENABLE LIVE FLEXCONNECT

USB 2.0/USB 3.1 Gen1 downstream port 1 can be configured as the upstream port, which reconfigures the upstream port as downstream port 1.

- If check box is not checked, the upstream port is not FlexConnected.
- If check box is checked, the upstream port is FlexConnected.

Live FlexConnect can be disabled by resetting the device. Note that this is a live update wherein the configuration memory is not changed, and only the registers are changed. Therefore, the functionality is lost when resetting the device. See [Figure 5-70](#).

FIGURE 5-70: LIVE FLEXCONNECT

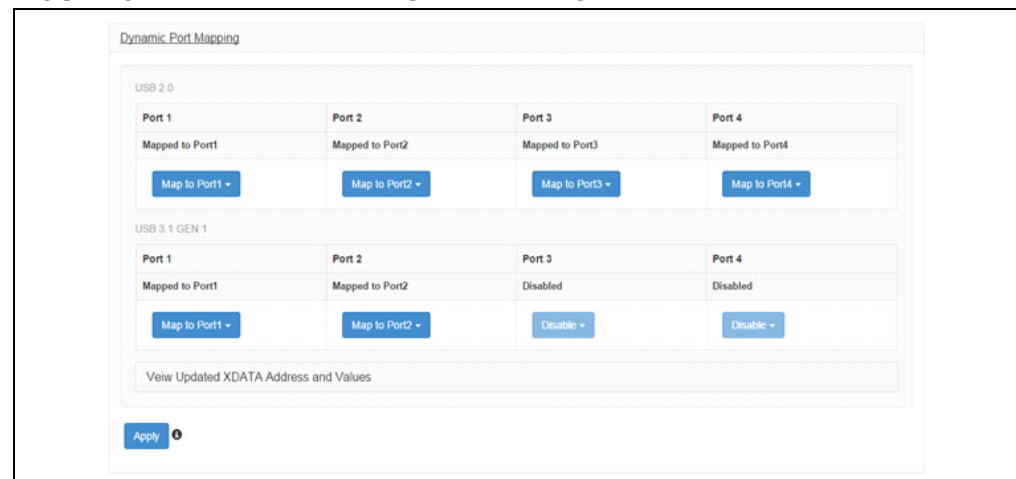


5.1.4.2 DYNAMIC PORT MAPPING

5.1.4.2.1 USB70xx

This feature allows to dynamically map USB2 and USB3 physical ports to logical port. The applied changes are reflected once after the removal and reinsertion of the hub. The updated XDATA registers and respective values can be viewed by clicking on **View Updated XDATA Registers and Values**. See [Figure 5-71](#).

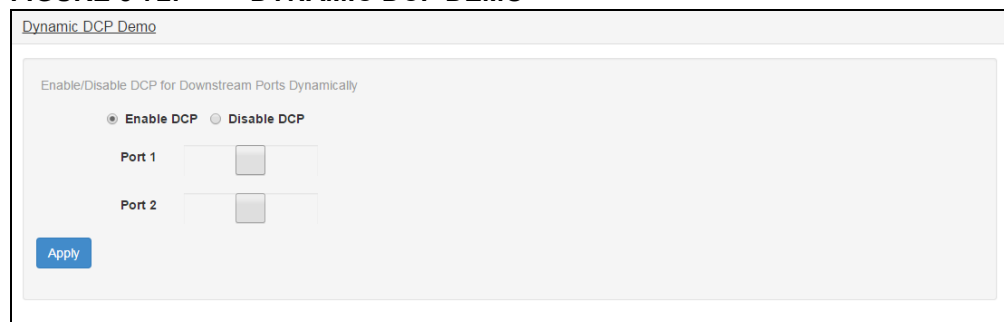
FIGURE 5-71: DYNAMIC PORT MAPPING



5.1.4.3 DYNAMIC DCP DEMO

5.1.4.3.1 USB49xx /USB4715 and USB70xx

This feature allows any port of the USB hub to be put in and out of Dedicated Charging Port (DCP) mode with the USB port still exposed to the host but would continue to report in a Device-disconnected state as long as it is in DCP mode. This USB command is a Control transfer (NO DATA command) to be issued to Endpoint 0 of the hub controller endpoint attached internally to the hub. See [Figure 5-72](#).

FIGURE 5-72: DYNAMIC DCP DEMO


5.1.4.4 ROLE SWITCH DEMO

Role Switch Demo is used to perform Flexconnect, Multi Host Bridging or Dual Upstream based on SKU. See [Figure 5-73](#).

5.1.4.4.1 Role Switch ON

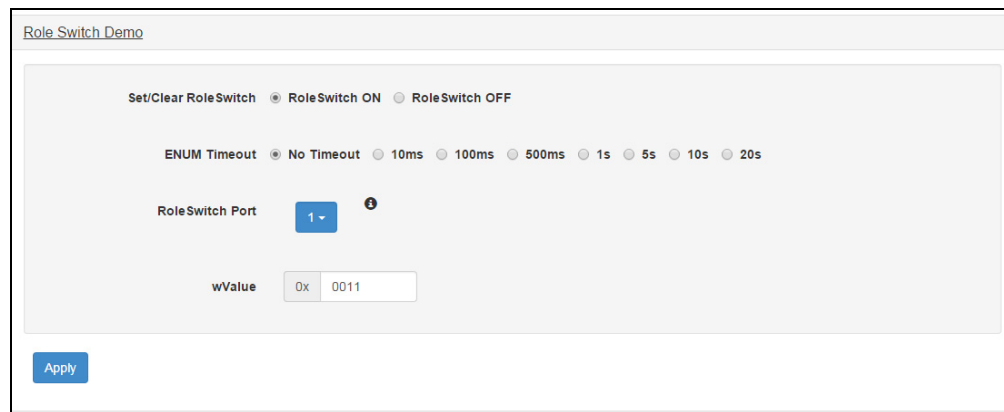
Role Switch ON refers to the state in which the upstream-facing port connects to the device and the downstream-facing port connects to the host.

5.1.4.4.2 Role Switch OFF

Role Switch OFF refers to the normal hub functionality in which the upstream-facing port connects to the host and the downstream-facing port connects to the device.

5.1.4.4.3 ENUM_TIMEOUT

This is the maximum time the firmware waits for the USB device to enumerate after transitioning into the Role-switched state. After which, it transitions out of Role-switched state automatically.

FIGURE 5-73: USB49XX ROLE SWITCH DEMO


5.1.4.4.4 FLEXCFG Area

This refers to a portion of the data memory where register configurations such as port disable/enable and battery charging configuration are written that needs to be updated over OTP/Pseudo OTP during next CPU reset. See [Figure 5-74](#) and [Figure 5-75](#).

FIGURE 5-74: USB4715 ROLE SWITCH DEMO

FIGURE 5-75: USB70XX ROLE SWITCH DEMO

5.1.4.5 USB-I²C BRIDGE DEMO

Microchip USB hubs facilitate USB-I²C Bridging through an embedded USB device (5th port) control point. See [Figure 5-76](#) and [Figure 5-77](#).

FIGURE 5-76: USB-I²C BRIDGE DEMO

The screenshot shows the 'USB-I2C Bridge Demo' window. It contains the following fields and controls:

- Frequency:** A button labeled 'Select Frequency'.
- Select I2C Operation:** Two radio buttons, 'Read' (selected) and 'Write'.
- Slave Address:** A text box with '0x' and an empty field.
- Length:** A text box with an empty field.
- Data:** A text box with '0x' and an empty field.
- Execute:** A blue button at the bottom left.

FIGURE 5-77: USB49XX/USB4715 AND USB70XX

The screenshot shows the 'USB-I2C Bridge Demo' window with the following settings:

- Frequency:** A button labeled '400KHz'.
- Select I2C Operation:** Two radio buttons, 'Read' and 'Write' (selected).
- Slave Address:** A text box with '0x' and '50'.
- Length:** A text box with '4'.
- Data:** A text box with '0x' and '00 01 03 05'.
- Read/Write:** Two blue buttons at the bottom left.

Below are the steps for executing an I²C write example:

1. Fill in the fields of the USB-I²C Bridge Demo page as shown in [Figure 5-78](#).
 - "0x50" is the I²C Slave address.
 - Data field has 3 entries: "00" represents the start address, and "11 22" is the data to be written.

FIGURE 5-78: STEP 1

The screenshot shows the 'USB-I2C Bridge Demo' window with the following settings for Step 1:

- Frequency:** A button labeled '238KHz'.
- Select I2C Operation:** Two radio buttons, 'Read' and 'Write' (selected).
- Slave Address:** A text box with '0x' and '50'.
- Length:** A text box with '3'.
- Data:** A text box with '0x' and '00 11 22'. The '00' is highlighted with a blue box.
- Execute:** A blue button at the bottom left.

Annotations in the image:

- A blue arrow points from the '00' in the Data field to the text: "Start Address to which data is written."
- A blue arrow points from the '11 22' in the Data field to the text: "Rest of the values represent the data to be written."

2. Perform a write again for slave address "0x50", data "00", and length "1".
3. Perform an I²C read as illustrated in [Figure 5-79](#).

FIGURE 5-79: STEP 3

USB-I2C Bridge Demo

Frequency: 238KHz

Select I2C Operation: ☒ Read ☐ Write

Slave Address: 0x 50

Length: 2

Data: 0x 11 22

Execute

The values read in Step 1 are read successfully.

5.1.4.6 USB-SPI BRIDGE DEMO

5.1.4.6.1 SPI Read

In [Figure 5-80](#), “0x50” is the start address of the SPI flash from where read operation starts. “Length” represents the number of bytes to be read. This option is available only in USB253X/USB(8)4604, USB57xx, USB58xx/USB59xx devices.

FIGURE 5-80: SPI READ

USB-SPI Bridge Demo

Select SPI Operation: ☒ Read ☐ Write ☐ Transfer

Start Address: 0x 50

Length: 2

Data: 0x 8a 2d

Execute

5.1.4.6.2 SPI Write

“SPI Write” is used to update SPI flash with firmware data. This option is available only in USB253X/USB(8)4604, USB57xx, and USB58xx/USB59xx devices. See [Figure 5-81](#).

FIGURE 5-81: SPI WRITE

USB-SPI Bridge Demo

Select SPI Operation: ☐ Read ☒ Write ☐ Transfer

Select SPI Firmware: Choose File USB2534_SPI_V128.bin

Execute

Binary file to be written in SPI Flash

5.1.4.6.3 SPI Transfer

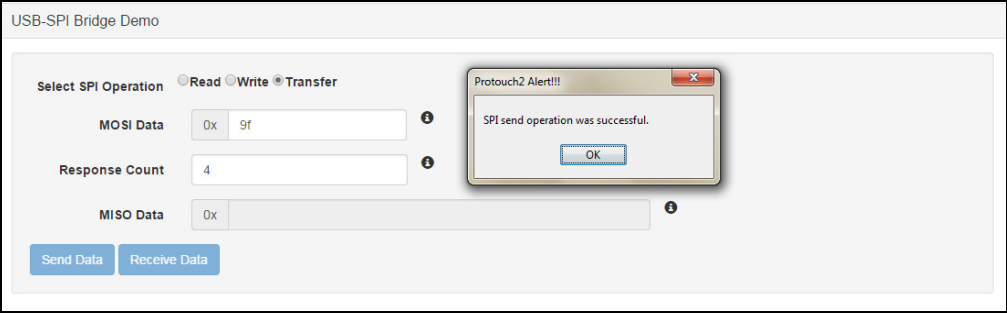
This is a demo for low-level SPI pass-through command reads/writes. All commands to the SPI interface are directed as SPI pass-through writes. SPI pass-through read is an XDATA read from a specified offset where the response is stored.

SPI transfer demo can be done only when the USB hub boots up from ROM.

5.1.4.6.4 SPI Send Operation

Send Data is used to send given data to the SPI slave. See [Figure 5-82](#).

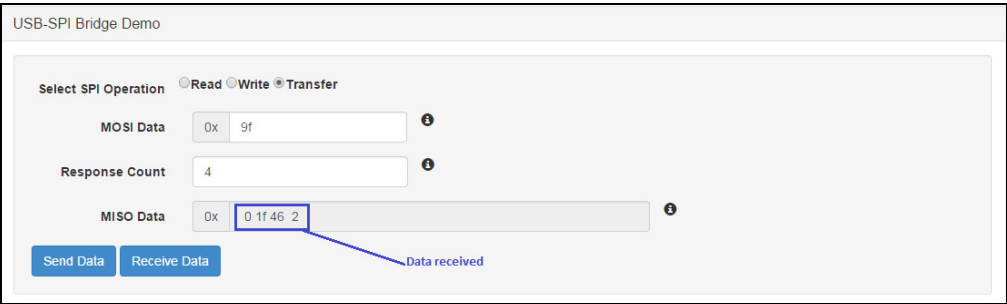
FIGURE 5-82: SPI SEND OPERATION



5.1.4.6.5 SPI Receive Operation

During an SPI data transfer, response data from the SPI slave is copied starting at memory location 0xBFDD2310. This response data can be read using Receive Data. See [Figure 5-83](#).

FIGURE 5-83: SPI RECEIVE OPERATION



5.1.4.7 USB-GPIO BRIDGE DEMO

This is used for low-level control of GPIO pins in Microchip USB hubs. The user can configure the direction, pull-up/down, read data, and write data to any GPIO. See [Figure 5-84](#).

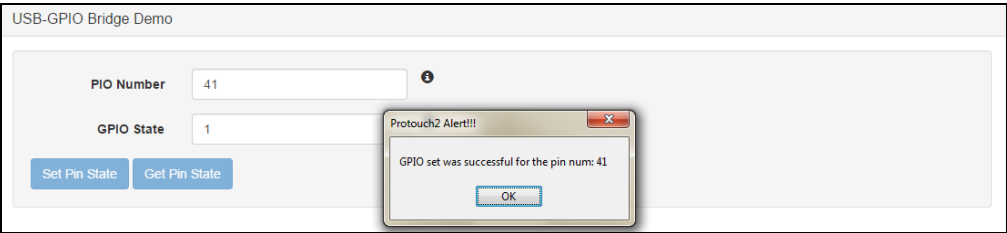
FIGURE 5-84: USB-GPIO BRIDGE DEMO



5.1.4.7.1 GPIO Set Operation

This demo allows the user to set the state of a specified GPIO pin. See [Figure 5-85](#).

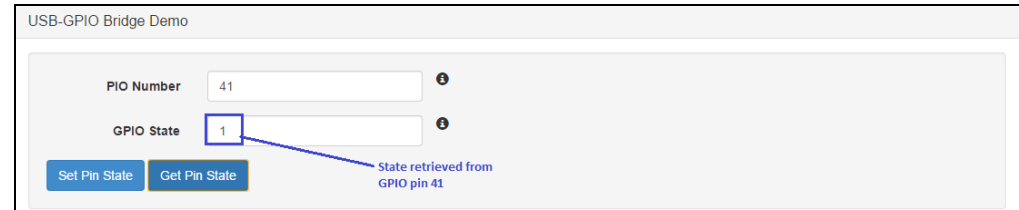
FIGURE 5-85: GPIO SET OPERATION



5.1.4.7.2 GPIO Get Operation

This demo allows the user to read the state of a specified GPIO pin. See [Figure 5-86](#).

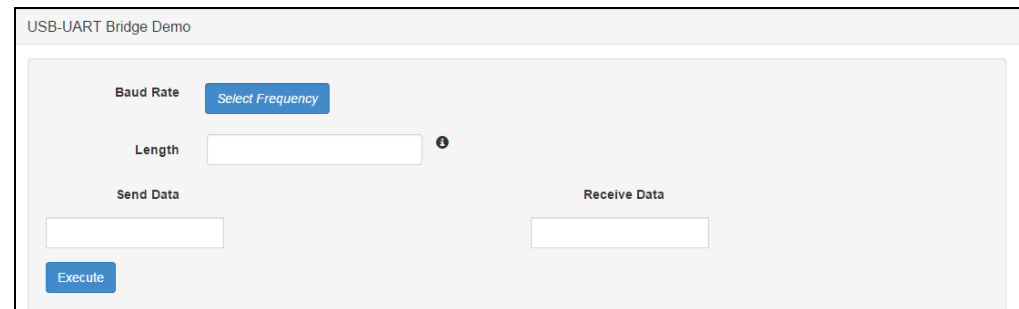
FIGURE 5-86: GPIO GET OPERATION



5.1.4.8 USB-UART BRIDGE DEMO

MPLAB Connect Configurator GUI supports UART demo only for USB253x /(8)460x /3x13 Hubs. See [Figure 5-87](#).

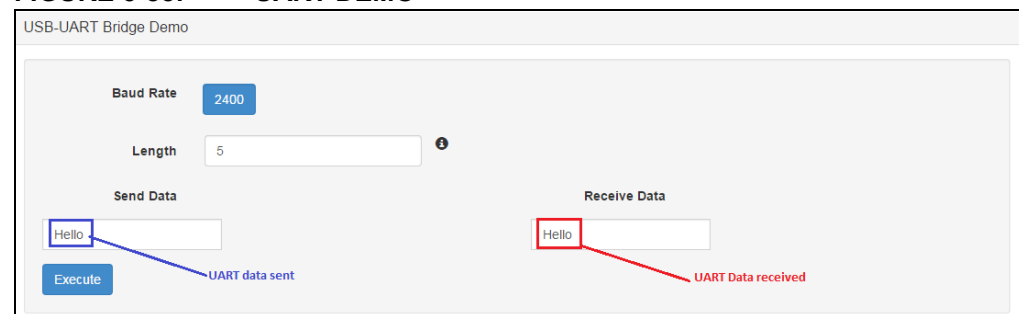
FIGURE 5-87: USB-UART BRIDGE DEMO



5.1.4.8.1 UART Demo

With UART demo, the data is transferred through a serial port to the connected serial peripheral and received there. See [Figure 5-88](#).

FIGURE 5-88: UART DEMO



5.2 PROGRAMMING PAGE

MPLAB Connect Configurator GUI supports programming configuration file and SPI Firmware file. See [Figure 5-89](#) to [Figure 5-93](#).

FIGURE 5-89: PROGRAMMING PAGE

The screenshot shows the 'Protouch2 USB hub - Program' window. At the top right are links for 'Home', 'Configuration', and 'Help'. Below the title bar, there are two green buttons: 'Program Once' and 'Mass Program'. The main area is divided into three sections. The first section, 'Select USB Hub', contains a dropdown menu showing 'Index:0,VID:0424,PID:2532,DID:0182' and a green 'Refresh' button. The second section, 'Configuration programming', contains a 'Select Configuration file' label, a 'Choose File' button, and the text 'No file chosen'. The third section, 'SPI Firmware programming', contains a 'Select Firmware file' label, a 'Choose File' button, and the text 'No file chosen'. Below this, there is an 'Erase Pseudo OTP' label and a checkbox.

FIGURE 5-90: USB49XX/USB4715

The screenshot shows the 'USB hub - Program' window. At the top right are links for 'Home', 'Online Configuration', and 'Help'. Below the title bar, there are two green buttons: 'Program Once' and 'Mass Program'. The main area is divided into three sections. The first section, 'Select USB Hub', contains a dropdown menu showing 'Index:0,VID:0424,PID:4715,DID:0121' and a green 'Refresh' button. The second section, 'Configuration programming', contains a 'Select Configuration file' label, a 'Choose File' button, and the text 'No file chosen'. The third section, 'SPI Firmware programming', contains a 'Select Firmware file' label, a 'Choose File' button, and the text 'No file chosen'. Below this, there are two labels: 'Erase Pseudo OTP' and 'Erase SPI Flash', each with a checkbox.

FIGURE 5-91: USB491X, USB7002, AND USB7050

The screenshot shows the main configuration interface. At the top right are two green buttons: "Program Once" and "Mass Program". Below them is a "Select USB Hub" section with a dropdown menu showing "Index:0,VID:0424,PID:4914,DID:0121" and a "Refresh" button. A sidebar on the left contains three options: "Configuration programming", "SPI Firmware programming", and "MAC Address programming", with the last one highlighted by a red box.

FIGURE 5-92: USB491X, USB7002, AND USB7050

The screenshot shows the "MAC Address programming" section for "Hub Feature Controller1 / NCM Device 1". It includes a toggle for "Enable Mass Program". Below this, there are fields for "Program MAC Address" (with a toggle), "Start MAC Address" (with a "Current MAC Addr:" button), "End MAC Address" (with a "Maximum MAC Add" button), and an "Increment Value" dropdown menu.

FIGURE 5-93: USB491X, USB7002 AND USB7050

The screenshot shows the "MAC Address programming" section for "Hub Feature Controller2 / NCM Device 2". It includes a toggle for "Program MAC Address", a "Start" button with "Current MAC Addr:" text, a "Maximum" button with "Maximum MAC Add" text, and an "Increment by" dropdown menu.

5.2.1 Device Selection

The hub of interest can be selected from the dropdown list.

5.2.2 Configuration Programming

Both JSON and binary format configuration files are permitted. The configuration file is saved by changing the configuration items of interest and clicking on the **Preview changes** button in the Online page. The configuration file generated on the Offline page can be used as an input. See [Figure 5-94](#).

FIGURE 5-94: CONFIGURATION PROGRAMMING

The screenshot shows the "Configuration programming" section. It features a "Select Configuration file" label, a "Choose File" button, and the text "No file chosen".

5.2.3 Firmware Programming

Firmware programming is required only if the user intends to program the SPI flash. Otherwise, this section can be skipped. Contact Microchip for the latest version of the SPI flash firmware file. The “Erase Pseudo OTP” checkbox should be selected with firmware programming to erase the SPI flash configuration memory along with firmware download. The default behavior is to append to the configuration memory with firmware programming and not erasing the configuration memory. Only binary files are permitted. See [Figure 5-95](#) and [Figure 5-96](#).

FIGURE 5-95: FIRMWARE PROGRAMMING

FIGURE 5-96: USB49XX/USB4715 AND USB70XX

5.2.4 Configuration and SPI Flash Firmware Programming

The configuration file and firmware file can be programmed at once. See [Figure 5-97](#).

FIGURE 5-97: PROGRAM ONCE OR MASS PROGRAM

5.2.4.1 MAC ADDRESS PROGRAMMING

Note: This is applicable only for (USB4916/USB4914, USB7002, and USB7050).

MAC address programming can be performed in the various ways listed below:

1. Programming the configuration file with the NCM1 MAC address
2. Programming the configuration file with the NCM2 MAC address
3. Programming the configuration file with the NCM1 and NCM2 MAC addresses
4. Programming the NCM1 and NCM2 MAC address
5. Programming the configuration file and firmware along with the NCM1 and NCM2 MAC addresses

EXAMPLE 5-6: EXAMPLE FOR SINGLE PROGRAMMING

NCM1 MAC Address Programming:
Start: "00:80:0F:78:00:02"

NCM2 MAC Address Programming:
Start: "00:80:0F:78:00:06"

If the above values are given, the selected device is programmed with NCM1 MAC Address "00:80:0F:78:00:02" and NCM2 MAC address "00:80:0F:78:00:06". See [Example 5-6](#).

EXAMPLE 5-7: EXAMPLE FOR MASS PROGRAMMING

MAC Address in Mass Programming:

NCM1 MAC Address:

Start: "00:80:0F:78:00:00"
Maximum: "00:80:0F:78:00:10"
Increment: 2

NCM2 MAC Address:

Start: "00:80:0F:78:00:05"
Maximum: "00:80:0F:78:00:10"
Increment: 1

If the above values are given, the Device1 is programmed with NCM1 MAC Address "00:80:0F:78:00:00" and NCM2 MAC address "00:80:0F:78:00:05". See [Example 5-7](#).

Device 2 is programmed with NCM1 MAC Address "00:80:0F:78:00:02" and NCM2 MAC address "00:80:0F:78:00:06".

5.2.5 Types of Programming

5.2.5.1 PROGRAM ONCE

The **Program Once** option allows a single device to be programmed at a time. This is performed by selecting the hub to be programmed along with the configuration file and/or firmware file, followed by clicking the **Program Once** button.

5.2.5.2 MASS PROGRAM

Multiple devices can be programmed one after the other by using the **Mass Program** option. This is performed by first selecting the hub of interest along with the configuration file and/or firmware file before starting the mass program.

5.2.5.3 PROGRAMMING TIME

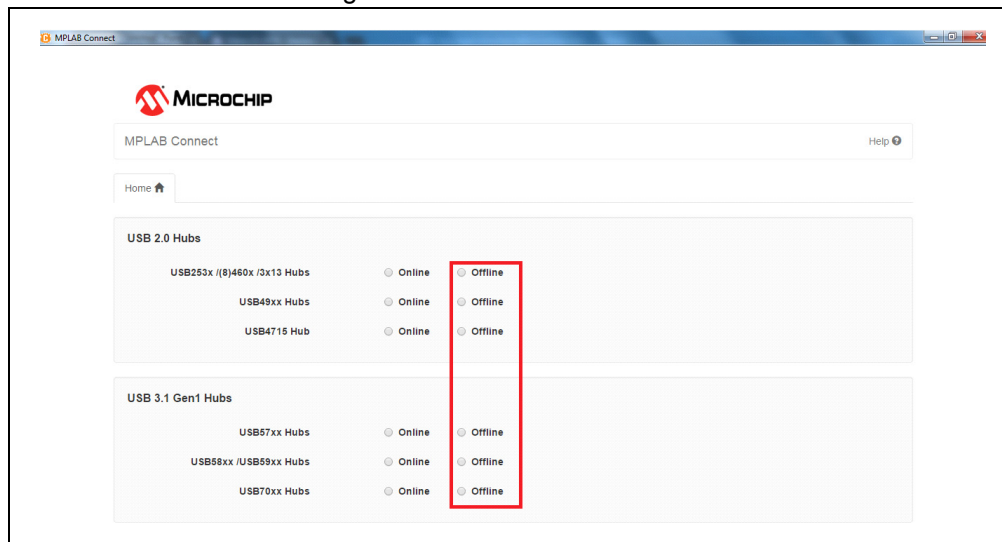
Programming takes approximately 3 to 5 seconds if the internal HFC device is enabled. Otherwise, it takes about 14 to 20 seconds depending on the OS (Windows 7 or 8.1) and architecture (32 bit or 64 bit).

5.3 OFFLINE HUB CONFIGURATION PAGE

Offline support allows for generation of a configuration file without the hub being connected to the system.

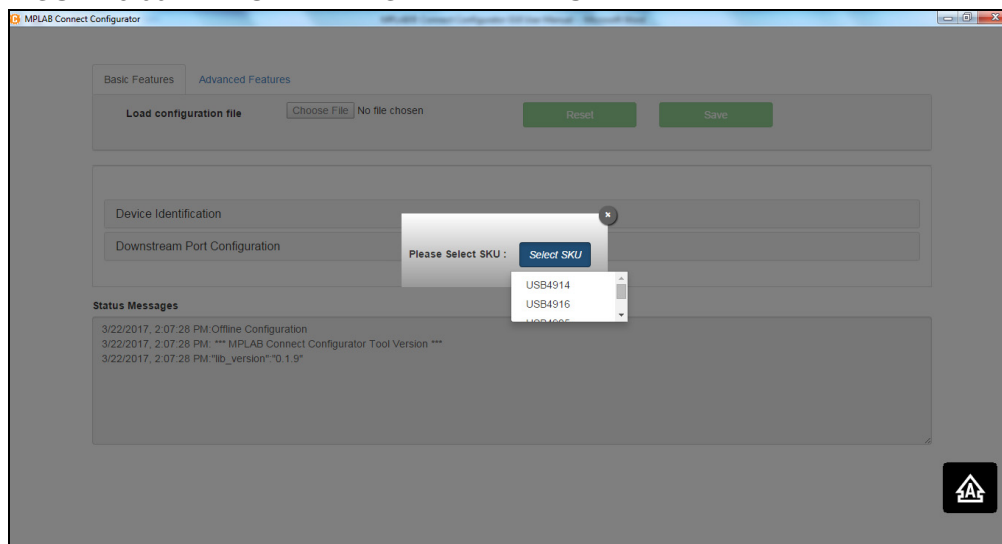
1. Select the **Radio** button on the landing page as shown in [Figure 5-98](#).

FIGURE 5-98: Selecting Offline Mode



2. Select the part number of interest from the dropdown as shown in [Figure 5-99](#).

FIGURE 5-99: SELECTING THE PART NUMBER



3. If the user wants to generate a configuration file for a device whose configuration is already modified, the device's dump file should be loaded using the "Load Configuration File" option. (See [Figure 5-100](#).) This should be done before any changes are done to the configuration settings. Once changes are made to the default settings, the "Load Configuration File" becomes active instead of the

“Load Configuration File.”

Note: The **Save** button is inactive at this time since no changes are made to the configuration settings.

FIGURE 5-100: LOADING A CONFIGURATION FILE ON TOP OF THE DEFAULT SKU SETTINGS

The screenshot shows the 'USB57X4 Offline Hub Configuration' web application. At the top, there are navigation links: Home, Program Page, Online Configuration, and Help. Below this, there are two tabs: 'Basic Features' and 'Advanced Features'. Under 'Basic Features', there is a blue button labeled 'USB5734'. To its right, there is a section titled 'Load configuration file' which contains a 'Choose File' button and the text 'No file chosen'. This entire section is highlighted with a green rectangular box. To the right of this section are two green buttons: 'Reset' and 'Save'. At the bottom of the interface, there is a section titled 'Device Identification USB 2.0'.

4. Configuration settings of the hub as desired can be changed from the default or default+ Load configuration settings as shown in [Figure 5-101](#). The **Save** button becomes active as soon as any changes are done.

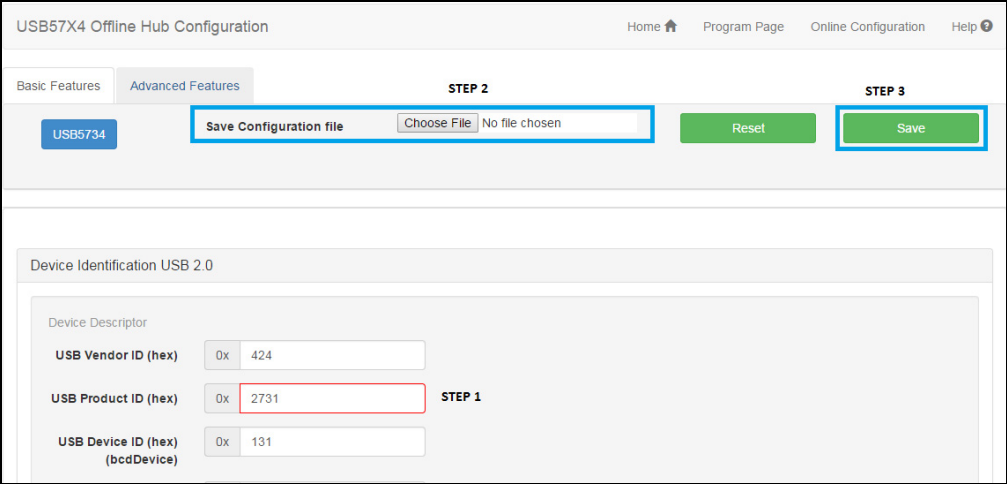
FIGURE 5-101: CHANGE CONFIGURATION SETTINGS

The screenshot shows the same web application as Figure 5-100, but with changes. The 'Load configuration file' section is no longer highlighted. Instead, the 'Save Configuration file' button is now active and highlighted with a green box. The 'Device Identification USB 2.0' section is expanded, showing a 'Device Descriptor' table with the following values:

Device Descriptor	
USB Vendor ID (hex)	0x 424
USB Product ID (hex)	0x 2731
USB Device ID (hex) (bcdDevice)	0x 131
USB bcdUSB	0x 2.10

5. Once the required changes are made, the user can select the file name and click on the **Save** button. The **Reset** button can be used to reverse the changes and reset to default settings. See [Figure 5-102](#).

FIGURE 5-102: SAVE THE NEW CONFIGURATION FILE



Chapter 6. LAN78xx Devices

6.1 ONLINE LAN78XX CONFIGURATION PAGE

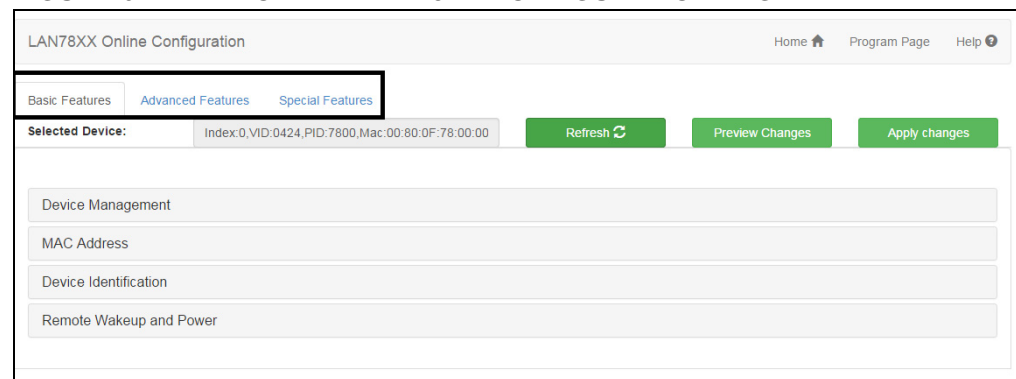
The Microchip LAN78xx contains several registers that can alter the behavior of the device. These can be changed to new values through the MPLAB® Connect Configurator GUI application by programming the EEPROM or OTP Memory.

- Note 1:** To use the online LAN78xx configuration page, the tool must be opened with administrator rights.
- 2:** The tool supports EEPROM or OTP memory only.
- 3:** If the device boots up from EEPROM and the user programs the OTP, the device would boot from EEPROM and not from OTP after programming.

The LAN78xx configuration items are divided into the two following categories and are available under two different tabs as shown in [Figure 6-1](#).

- [Basic Features](#) (Shown by default)
- [Advanced Features](#)

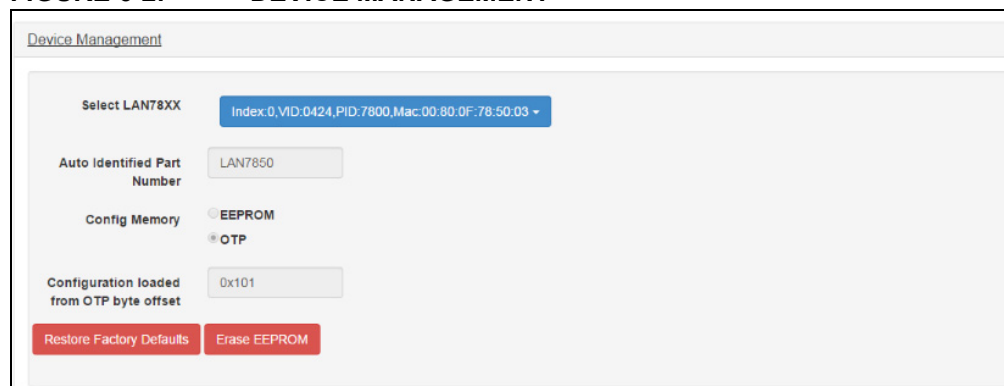
FIGURE 6-1: ONLINE LAN78XX CONFIGURATION PAGE



6.1.1 Basic Features

6.1.1.1 DEVICE MANAGEMENT

The following items will be part of Device management: Device selection, Refresh device list, Auto Identified Part Number of selected device, Config Memory of the selected device— OTP or SPI, and Memory Usage. See [Figure 6-2](#).

FIGURE 6-2: DEVICE MANAGEMENT

6.1.1.1.1 Select LAN78xx

Figure 6-2 is the default screen displayed once the LAN78xx online page is launched. At launch, all the GUI elements (in all pages) remain disabled except “Select USB Hub.” The default item selected would be “LAN at index 0.”

6.1.1.1.2 Auto Identified Part Number

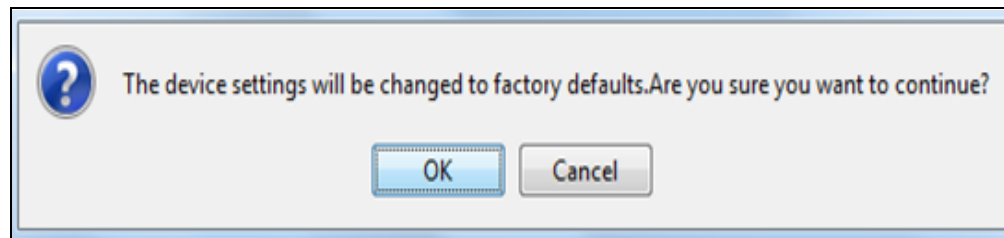
This section provides the default PID of the device. Auto-identification is independent of VID/PID changes.

6.1.1.1.3 Configuration Memory

The LAN78xx device has two types of configuration memory: EEPROM and OTP. If EEPROM memory is enabled, all the configuration values are loaded from EEPROM. If OTP memory is enabled, all the configuration values are loaded from OTP memory.

6.1.1.1.4 Restore Factory Defaults

This option restores the device to the original manufacturer settings by erasing all of the stored information. Only EEPROM memory supports the “Restore Factory Defaults” option. Refer to Figure 6-3.

FIGURE 6-3: MPLAB CONNECT CONFIRMATION

6.1.1.1.5 Erase EEPROM

EEPROM content can be erased using this button.

6.1.1.2 MAC ADDRESS

This is the 6-byte universally unique MAC address the board uses. Bytes are separated by a colon. See [Figure 6-4](#).

FIGURE 6-4: MAC ADDRESS

6.1.1.3 DEVICE IDENTIFICATION

For the LAN7850, the Device Descriptor fields configure USB 2.0 VID, USB 2.0 PID, and USB 2.0 DID. For the LAN7800, the Device Descriptor fields configure both the USB 2.0 and USB 3.0 VIDs, PIDs, and DIDs. See [Figure 6-5](#).

FIGURE 6-5: DEVICE IDENTIFICATION

USB String Descriptors can be disabled by programming empty string as in [Figure 6-6](#).

FIGURE 6-6: STRING DESCRIPTORS

6.1.1.4 REMOTE WAKEUP AND POWER

“Remote Wakeup” is used to wake up the device from a U1/U2/U3 Low-power state to the U0 state. When in Self-powered mode, an external supply of 5V is required for device operation. Refer to [Figure 6-7](#).

FIGURE 6-7: REMOTE WAKEUP AND POWER

Remote Wakeup and Power

Remote Wakeup	<input checked="" type="checkbox"/>	
Self Power	<input checked="" type="checkbox"/>	
HS Max Power	<input type="text" value="2"/>	mA
FS Max Power	<input type="text" value="2"/>	mA
SS Max Power	<input type="text" value="8"/>	mA

6.1.2 Advanced Features

6.1.2.1 INTERRUPT ENDPOINT BINTERVAL

The polling interval for the Interrupt Endpoint can be configured for “Full Speed,” “High Speed” and “Super Speed” operations, as shown in [Figure 6-8](#).

FIGURE 6-8: INTERRUPT ENDPOINT BINTERVAL

Interrupt Endpoint bInterval

Full Speed	<input type="text" value="1"/>
High Speed	<input type="text" value="4"/>
Super Speed	<input type="text" value="6"/>

6.1.2.2 LED CONFIGURATION

All the Four LEDs (LED0, LED1, LED2, and LED3) can be enabled and configured in any of the modes below:

- **Link/Activity:** LED monitors the Link at any speed and its activity.
- **Link1000/Activity:** LED monitors the Link in 1000BASE-T and its activity.
- **Link100/Activity:** LED monitors the Link in 100BASE-TX and its activity.
- **Link10/Activity:** LED monitors the Link in 10BASE-T and its activity.
- **Link100/1000/Activity:** LED monitors the Link in 1000BASE-T and 100BASE-TX and its activity.
- **Link10/1000/Activity:** LED monitors the Link in 1000BASE-T and 10BASE-T and its activity.
- **Link10/100/Activity:** LED monitors the Link in 100BASE-TX and 10BASE-T and its activity.
- **Duplex/Collision:** LED monitors the Half Duplex, Full Duplex, and their Collision Status.
- **Collision:** LED monitors the Collision in the Link.
- **Activity:** LED monitors the Activity in the Link.
- **Auto-Negotiation Fault:** LED monitors the Auto-Negotiation Fault in Link.

- **Serial Mode:** LED monitors the Serial Stream.
- **Force LED OFF:** This de-asserts the LED.
- **Force LED ON:** This asserts the LED.

All the Four LEDs (LED0, LED1, LED2, and LED3) behavior can be configured using the fields shown in [Figure 6-9](#) and [Figure 6-10](#).

FIGURE 6-9: LED CONFIGURATION

FIGURE 6-10: LED CONFIGURATION (CONTINUED)

6.1.2.3 LED COMMON FEATURES

All four LED behaviors can be configured using the fields in [Figure 6-11](#).

FIGURE 6-11: LED COMMON FEATURES

6.1.2.4 GPIO CONFIGURATION

GPIO Configuration fields are used to Enable/Disable all 7 GPIOs (GPIO0, GPIO1, GPIO2, GPIO3, GPIO4, GPIO5, and GPIO6). See [Figure 6-12](#).

FIGURE 6-12: GPIO CONFIGURATION



The output buffer for the GPIO pins can be configured as a push/pull driver or open/drain driver. GPIOs can be configured as input/output by GPIO Direction, as shown in [Figure 6-13](#).

FIGURE 6-13: OUTPUT BUFFER



When the GPIO is set as an output, the output value must be given in the GPIO data field, as shown in [Figure 6-14](#).

FIGURE 6-14: GPIO AS OUTPUT

GPIO 7	<input checked="" type="radio"/> enable <input type="radio"/> disable
GPIO 7 Buffer	<input checked="" type="radio"/> Open Drain <input type="radio"/> Push Pull
GPIO 7 Direction	<input type="radio"/> Input <input checked="" type="radio"/> Output
GPIO 7 Wake Up	<input checked="" type="radio"/> Disable <input type="radio"/> Enable
GPIO 7 WakeUp Polarity	<input checked="" type="radio"/> Low <input type="radio"/> High
GPIO 7 Data	<input type="radio"/> Set <input checked="" type="radio"/> Clear

When the GPIO is set as an input, the GPIO data field is hidden as shown in [Figure 6-15](#).

FIGURE 6-15: GPIO AS INPUT

GPIO 5 Buffer	<input checked="" type="radio"/> Open Drain <input type="radio"/> Push Pull
GPIO 5 Direction	<input checked="" type="radio"/> Input <input type="radio"/> Output

6.1.2.5 POWER MANAGEMENT

The LAN78xx can wake up the host processor via the following types of PME wakeup events (See [Figure 6-16](#).):

- [GPIO Pins](#)
- [PHY Link Change](#)
- [PME Packet](#)
- [PME WUFF](#)

6.1.2.5.1 GPIO Pins

GPIO PME wakeup events can be enabled by the “GPIO PME” field. A GPIO PME event can be signaled via a level or pulse which can be set in the GPIO PME configuration. GPIO PME polarity is used to specify the level as low or high. If set as a pulse, the duration of the pulse for which the event should occur can be set in the “GPIO PME Length” field.

The GPIO PME can be enabled for the desired GPIO using the “GPIO Wakeup” field, and the polarity can be set in the “GPIO Wakeup” polarity field.

6.1.2.5.2 PHY Link Change

Detection of a PHY link partner when in PME mode results in a PME being asserted. This can be enabled or disabled in “PHY Link Change” field.

6.1.2.5.3 PME Packet

Reception of a Packet when in PME mode results in a PME being asserted. This can be enabled or disabled in “PME Packet” field.

6.1.2.5.4 PME WUFF

Reception of a packet matching the WUFF when in PME mode results in a PME being asserted. This can be enabled or disabled in “PME WUFF” field.

FIGURE 6-16: POWER MANAGEMENT

The screenshot shows the 'Power Management' configuration window. It contains the following settings:

- GPIO PME:** ☒ disable, ☐ enable
- GPIO PME Configuration:** ☒ Level, ☐ Pulse
- GPIO PME Length:** ☒ Pulse Length = 1.5ms, ☐ Pulse Length = 150ms
- GPIO PME Polarity:** ☒ Signalling Polarity Low, ☐ Signalling Polarity High
- GPIO Buffer Type:** ☒ Open Drain Driver, ☐ Push-Pull Driver
- PHY Link Change:** ☐ Wake up Not supported, ☒ Wake up Supported
- PME Packet:** ☐ Packet Event Wakeup disabled, ☒ Packet Event Wakeup enabled
- PME Perfect DA:** ☒ Perfect DA Event Wakeup disabled, ☐ Perfect DA Event Wakeup enabled
- PME WUFF:** ☒ Wakeup Frame Detection disable, ☐ Wakeup Frame Detection enable

6.1.2.6 LIVE UPDATE

Live Update enables the option of Register Read/Write for LAN registers and PHY registers. The register address and the length of bytes to read/write are set in “Register Address” and “Length” fields, respectively. See [Figure 6-17](#) and [Figure 6-18](#).

FIGURE 6-17: LIVE UPDATE: PHY REGISTERS

Live Update

Registers ☐ LAN Registers ☒ PHY Registers

Register address(hex) ⓘ

Length ⓘ

Register Operation ☒ Read ☐ Write

Register value(hex)

```
00001000 000079ed 00000007 0000c131 000005e1 0000cde1
0000000f 00002001 00006001 00000200 00003800 00000000
00000000 00004007 00000000
```

0x

Live Update

FIGURE 6-18: LIVE UPDATE: LAN REGISTERS

Live Update

Registers ☒ LAN Registers ☐ PHY Registers

Register address(hex) ⓘ

Length ⓘ

Register Operation ☒ Read ☐ Write

Register value(hex)

```
0008380c 05ee0025 00000001 6000ffff 00009876 00000000
00000000 780f8000 00000840 000079ed 00000000 00000000
00000000 0021001e 00000000
```

0x

Live Update

6.1.2.7 OTP/EEPROM DUMP MEMORY

Dump memory allows the user to dump OTP or EEPROM memory. The OTP or EEPROM memory dump are saved in the selected file with the .bin extension. See [Figure 6-19](#).

FIGURE 6-19: OTP/EEPROM DUMP MEMORY

OTP/EEPROM Dump Memory

Choose Memory ☒ EEPROM Memory ☐ OTP Memory

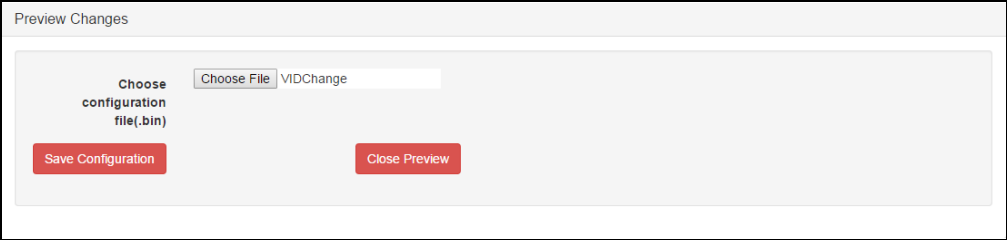
Save Dump file No file chosen

Dump Memory

6.1.3 Preview Changes

The “Preview Changes” option allows the user to generate the configuration file that is going to be programmed in the device. (See [Figure 6-20](#).) A configuration file with .bin extension is generated.

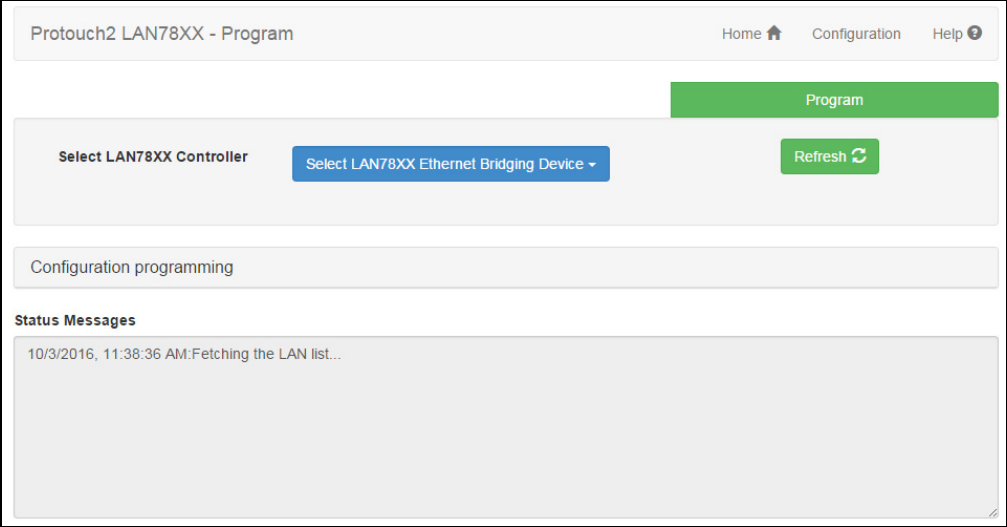
FIGURE 6-20: PREVIEW CHANGES



6.2 PROGRAMMING PAGE

The Programming Page enables the user to program the configuration file in EEPROM or OTP memory. See [Figure 6-21](#).

FIGURE 6-21: PROGRAMMING PAGE



6.2.1 Program Once

A single device can be programmed at a time by disabling the “Enable Mass Program” option. Select the device of interest from the “Select the LAN7800 Controller” dropdown menu and choose the memory to be programmed. Refer to [Figure 6-22](#) and [Example 6-1](#).

FIGURE 6-22: PROGRAM ONCE

Single programming can be performed in the following ways:

- [Programming Configuration file with Serial number and MAC address](#)
- [Programming Configuration file with Serial number alone](#)
- [Programming Configuration file with MAC address alone](#)

6.2.1.1 PROGRAMMING CONFIGURATION FILE WITH SERIAL NUMBER AND MAC ADDRESS

The configuration file is programmed in either OTP or EEPROM with the given serial number and MAC address overwriting the values in the configuration file.

MAC addresses for programming each device in Mass Program mode can be given in the “Start” field. Serial number for programming a device in Single Program mode can be given in the “Prefix” and “Start” field.

6.2.1.2 PROGRAMMING CONFIGURATION FILE WITH SERIAL NUMBER ALONE

The configuration file is programmed in either OTP or EEPROM with the given serial number overwriting the values in the configuration file.

6.2.1.3 PROGRAMMING CONFIGURATION FILE WITH MAC ADDRESS ALONE

The configuration file is programmed in either OTP or EEPROM with the given MAC address overwriting the values in the configuration file.

EXAMPLE 6-1: EXAMPLE FOR SINGLE PROGRAMMING

Serial Number in Single Programming:
Prefix: MCHP
Start: 1

MAC Address in Single Programming:
Start: “00:80:0F:78:00:02”

If the above example values are given, then Device 1 would be programmed with serial number MCHP1 and MAC Address "00:80:0F:78:00:02".

6.2.2 Mass Program

Multiple devices can be programmed one after the other by enabling the "Mass Program" option. (Refer to [Example 6-2](#).) Select the device of interest from the **Select the LAN9800 controller** dropdown and choose the memory to be programmed.

Mass programming can be performed in the following ways:

- [Programming Configuration File with Serial Number and MAC Address](#)
- [Programming Configuration File with Serial Number Alone](#)
- [Program Configuration File with MAC Address Alone](#)

6.2.2.1 PROGRAMMING CONFIGURATION FILE WITH SERIAL NUMBER AND MAC ADDRESS

The configuration file is programmed in either OTP or EEPROM with the given serial number and MAC address, overwriting the values in the configuration file.

MAC addresses for programming each device in Mass Program mode can be given in the "Start", "Maximum", and "Increment by" fields. The serial number for programming each device in Mass Program mode can be given in the "Prefix", "Start", "Maximum", and "Increment by" fields. See [Figure 6-23](#) and [Figure 6-24](#).

6.2.2.2 PROGRAMMING CONFIGURATION FILE WITH SERIAL NUMBER ALONE

The configuration file is programmed in either OTP or EEPROM with the given serial number, overwriting the values in the configuration file.

6.2.2.3 PROGRAM CONFIGURATION FILE WITH MAC ADDRESS ALONE

The configuration file is programmed in either OTP or EEPROM with the given MAC address, overwriting the values in the configuration file.

EXAMPLE 6-2: EXAMPLE FOR MASS PROGRAMMING

Serial Number in Mass Programming:

Prefix: MCHP

Start: 1

Maximum: 3

Increment By: 2

MAC Address in Mass Programming:

Start: "00:80:0F:78:00:00"

Maximum: "00:80:0F:78:00:10"

Increment: 10

If the above example values are given, then Device 1 would be programmed with serial number MCHP1 and MAC Address "00:80:0F:78:00:00". Device 2 would be programmed with serial number MCHP3 and MAC Address "00:80:0F:78:00:10".

FIGURE 6-23: CONFIGURATION PROGRAMMING

FIGURE 6-24: MAC ADDRESS

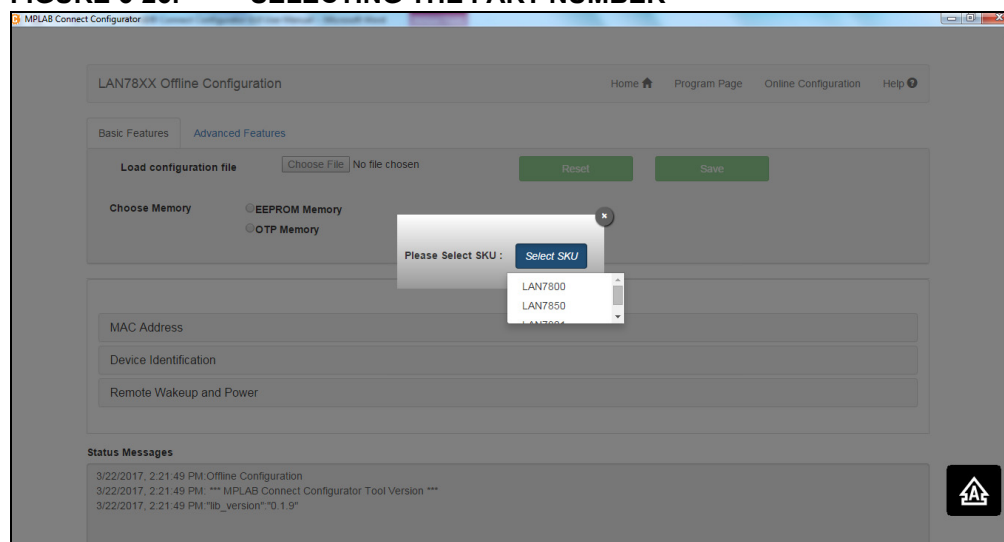
6.3 OFFLINE LAN78XX CONFIGURATION PAGE

Offline support allows for generation of a configuration file without the hub being connected to the system.

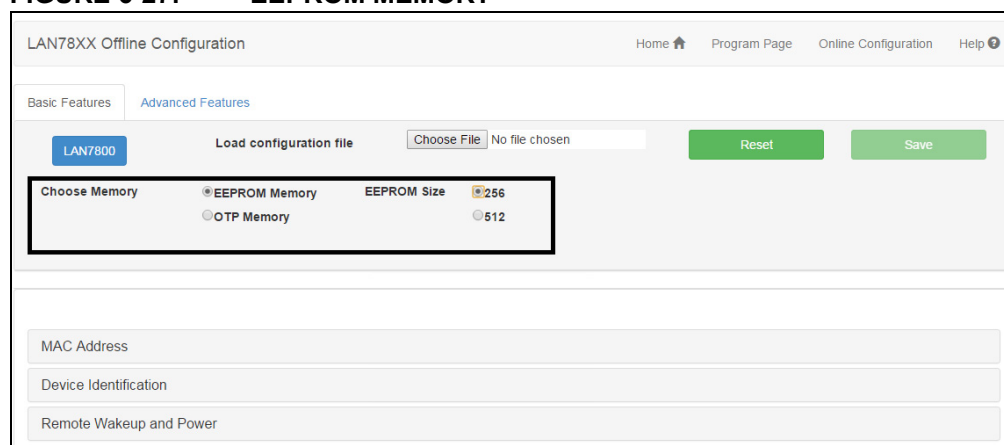
1. Select the **Radio** button on the landing page as shown in [Figure 6-25](#).

FIGURE 6-25: SELECTING OFFLINE MODE

2. Select the part number of interest from the dropdown as shown in [Figure 6-26](#). Choose the desired memory type for which the configuration file is to be generated.

FIGURE 6-26: SELECTING THE PART NUMBER


3. If EEPROM is chosen, then the size of EEPROM to be programmed must be selected as shown in [Figure 6-27](#).

FIGURE 6-27: EEPROM MEMORY


4. If the user wants to generate a configuration file for a device whose configuration is already modified, then that device's dump file should be loaded using the “Load Configuration File” option. This should be done before any changes are performed to the configuration settings. Once changes are made to the default settings, the “Save Configuration File” option becomes active instead of the “Load Configuration File”. The **Save** button is inactive at this time since no changes are made to the configuration settings. See [Figure 6-28](#).

Note: If the user wants to generate a configuration file for a device whose OTP is already programmed once, then the OTP dump file of the device should be loaded using “Load Configuration File” option.

FIGURE 6-28: LOADING A CONFIGURATION FILE ON TOP OF THE DEFAULT SKU SETTINGS

The screenshot shows the 'LAN78XX Offline Configuration' web interface. At the top, there are navigation links: Home, Program Page, Online Configuration, and Help. Below this, there are tabs for 'Basic Features' and 'Advanced Features'. The 'Basic Features' tab is active, and a 'LAN7800' button is visible. A text box labeled 'Load configuration file' contains the text 'Choose File' and 'No file chosen'. To the right of this text box are two green buttons: 'Reset' and 'Save'. Below the text box, there are radio buttons for 'EEPROM Memory' (selected) and 'OTP Memory'. To the right of these radio buttons is a label 'EEPROM Size' with a value of '256' and a radio button for '512'. Below these options, there are three input fields: 'MAC Address', 'Device Identification', and 'Remote Wakeup and Power'.

5. The configuration settings of the hub can be changed from the default as shown in Figure 6-29. The **Save** button becomes active as soon as any changes are done.

FIGURE 6-29: CHANGE CONFIGURATION SETTINGS

The screenshot shows the 'LAN78XX Offline Configuration' web interface. At the top, there are navigation links: Home, Program Page, Online Configuration, and Help. Below this, there are tabs for 'Basic Features' and 'Advanced Features'. The 'Basic Features' tab is active, and a 'LAN7800' button is visible. A text box labeled 'Save Configuration file' contains the text 'Choose File' and 'No file chosen'. To the right of this text box are two green buttons: 'Reset' and 'Save'. Below the text box, there are radio buttons for 'EEPROM Memory' and 'OTP Memory' (selected). Below these options, there are three input fields: 'MAC Address', 'Device Identification', and 'Remote Wakeup and Power'. The 'MAC Address' field is highlighted with a red border and contains the value '00:80:0F:78:00:05'.

6. Once the required changes are made, the user can select the file name and click on the **Save** button. (See Figure 6-30.) If the changes are to be reversed, then the **Reset** button can be used to reset to default settings.

FIGURE 6-30: SAVE THE NEW CONFIGURATION FILE

The screenshot shows the 'LAN78XX Offline Configuration' web interface. At the top, there are navigation links: Home, Program Page, Online Configuration, and Help. Below this, there are tabs for 'Basic Features' and 'Advanced Features'. The 'Basic Features' tab is active, and a 'LAN7800' button is visible. A text box labeled 'Save Configuration file' contains the text 'Choose File' and 'No file chosen'. To the right of this text box are two green buttons: 'Reset' and 'Save'. Below the text box, there are radio buttons for 'EEPROM Memory' and 'OTP Memory' (selected). Below these options, there are three input fields: 'MAC Address', 'Device Identification', and 'Remote Wakeup and Power'. The 'MAC Address' field is highlighted with a red border and contains the value '00:80:0F:78:00:05'. Annotations are present: 'STEP 2' is above the 'Save Configuration file' text box, 'STEP 3' is above the 'Save' button, and 'STEP 1' is to the right of the 'MAC Address' input field.

NOTES:

Chapter 7. LAN74XX Devices

7.1 ONLINE LAN74XX CONFIGURATION PAGE

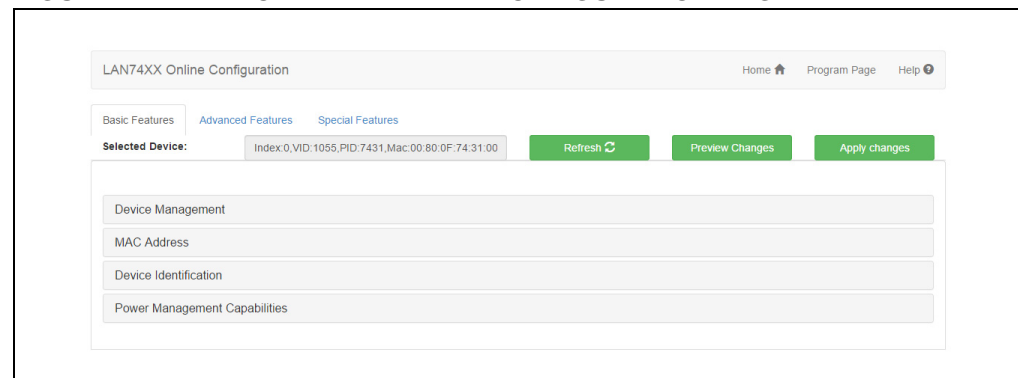
The Microchip LAN74xx contains several registers that can change the behavior of the device. These can be modified to new values through the MPLAB® Connect Configurator GUI application by programming the EEPROM or OTP Memory.

- Note 1:** PCIe device is restarted by changing PCIE_RESTART_RESCAN option in MPLABConnect.ini to TRUE. The MPLAB Connect Configurator GUI tool would try to restart the selected device after programming. The default value of PCIE_RESTART_RESCAN option is TRUE.
- 2:** The tool should be opened with administrator rights to use the Online LAN74xx configuration page to automatically restart the device. Otherwise, the tool need not be opened with administrator rights.
- 3:** The tool supports either EEPROM or OTP memory only.
- 4:** If the device boots up from EEPROM and the user programs the OTP, the device would boot from EEPROM and not from OTP after programming.

The LAN74xx configuration items are divided into the three following categories and are available under two different tabs as shown in [Figure 7-1](#):

- [Basic Features](#) (Shown by default)
- [Advanced Features](#)
- [Special Features](#)

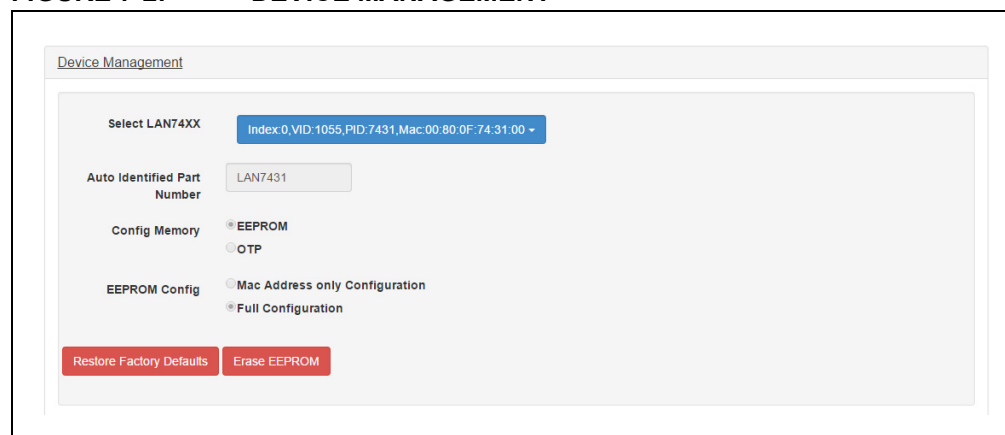
FIGURE 7-1: ONLINE LAN74XX CONFIGURATION PAGE



7.1.1 Basic Features

7.1.1.1 DEVICE MANAGEMENT

FIGURE 7-2: DEVICE MANAGEMENT



7.1.1.1.1 Select LAN74xx

Figure 7-2 is the default screen displayed once the LAN74xx online page is launched. At launch, all the GUI elements (in all pages) remain disabled except the “Select LAN74xx Controller”. The default item selected would be “LAN at index 0.”

7.1.1.1.2 Auto Identified Part Number

This section provides information on the SKU, which is the default PID of the device. Auto-identification is independent of VID/PID changes.

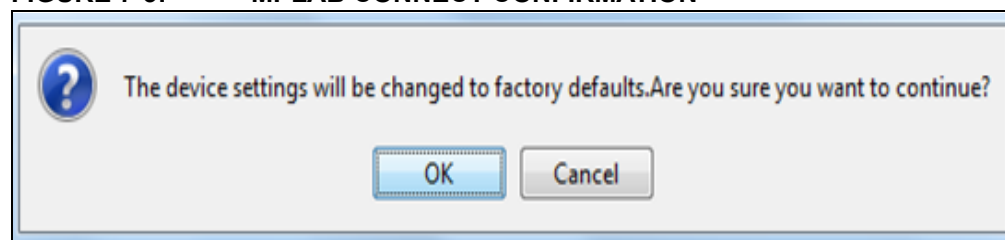
7.1.1.1.3 Configuration Memory

The LAN74xx device has two types of configuration memory: EEPROM and OTP. If EEPROM memory is enabled, all the configuration values are loaded from EEPROM. If OTP memory is enabled, all the configuration values are loaded from OTP memory.

7.1.1.1.4 Restore Factory Defaults

This option restores the device to the original manufacturer settings by erasing all of the stored information. Only EEPROM memory supports the “Restore Factory Defaults” option. Refer to Figure 7-3.

FIGURE 7-3: MPLAB CONNECT CONFIRMATION



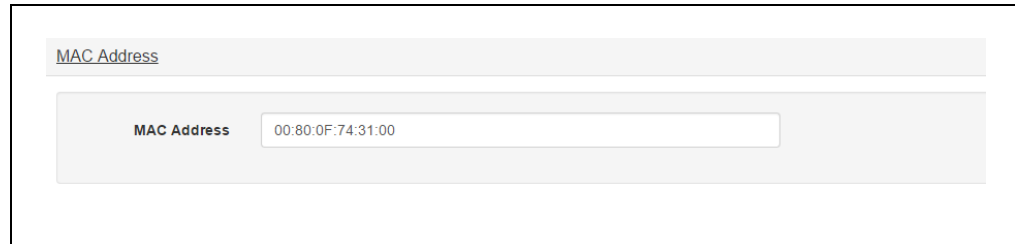
7.1.1.1.5 Erase EEPROM

EEPROM content can be erased using this button.

7.1.1.2 MAC ADDRESS

This is the 6-byte universally unique MAC address the board uses. Bytes are separated by a colon. See [Figure 7-4](#).

FIGURE 7-4: MAC ADDRESS



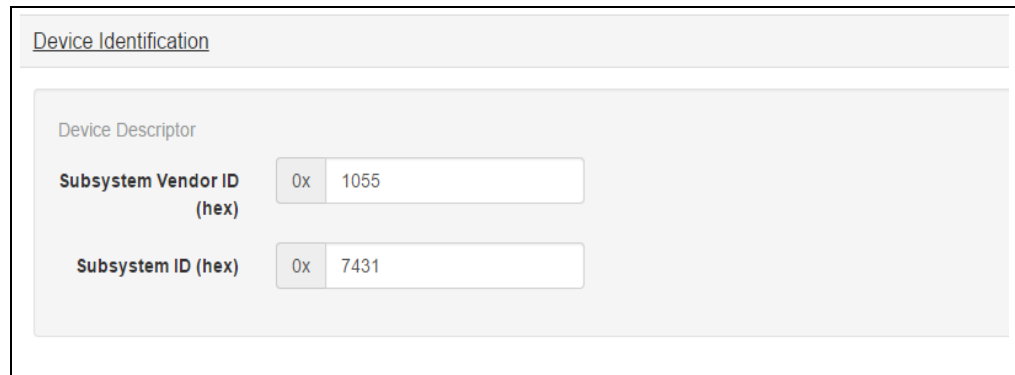
MAC Address

MAC Address 00:80:0F:74:31:00

7.1.1.3 DEVICE IDENTIFICATION

For LAN7430 and LAN7431, the Device Descriptor fields configure Subsystem Vendor ID and Subsystem ID. See [Figure 7-5](#).

FIGURE 7-5: DEVICE IDENTIFICATION



Device Identification

Device Descriptor

Subsystem Vendor ID (hex) 0x 1055

Subsystem ID (hex) 0x 7431

7.1.1.4 POWER MANAGEMENT CAPABILITIES

7.1.1.4.1 AUX Current

This field reports the 3.3 Vaux auxiliary current requirements for the PCI function. The following codes are used:

111 – 375 mA

110 – 320 mA

101 – 270 mA

100 – 220 mA

011 – 160 mA

010 – 100 mA

001 – 55 mA

000 – 0 (self powered)

7.1.1.4.2 PME Support

This field indicates the power states in which the function may generate a PME. A value of 0b for any bit indicates that the function is not capable of signaling PME while in that power state.

bit(11) X XXX1b – PME can be signaled from D0

bit(12) X XX1Xb – PME can be signaled from D1

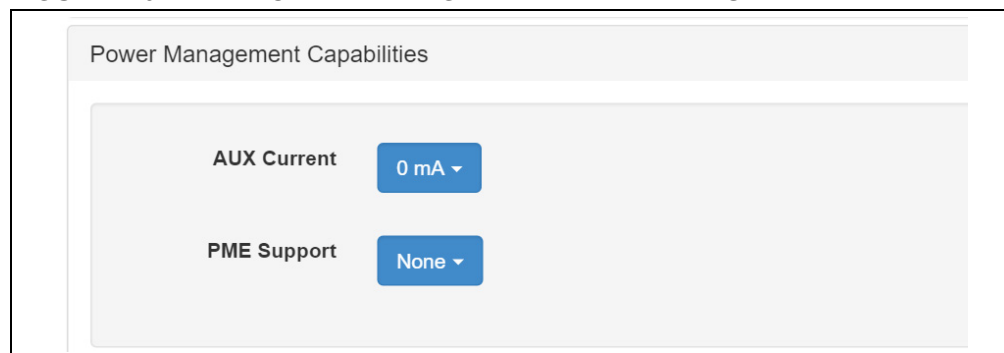
bit(13) X X1XXb – PME can be signaled from D2

bit(14) X 1XXXb – PME can be signaled from D3hot

bit(15) 1 XXXXb – PME can be signaled from D3cold

See [Figure 7-6](#).

FIGURE 7-6: POWER MANAGEMENT CAPABILITIES



7.1.1.5 LED CONFIGURATION

All the four LEDs (LED0, LED1, LED2, and LED3) can be enabled and configured in any of the modes below:

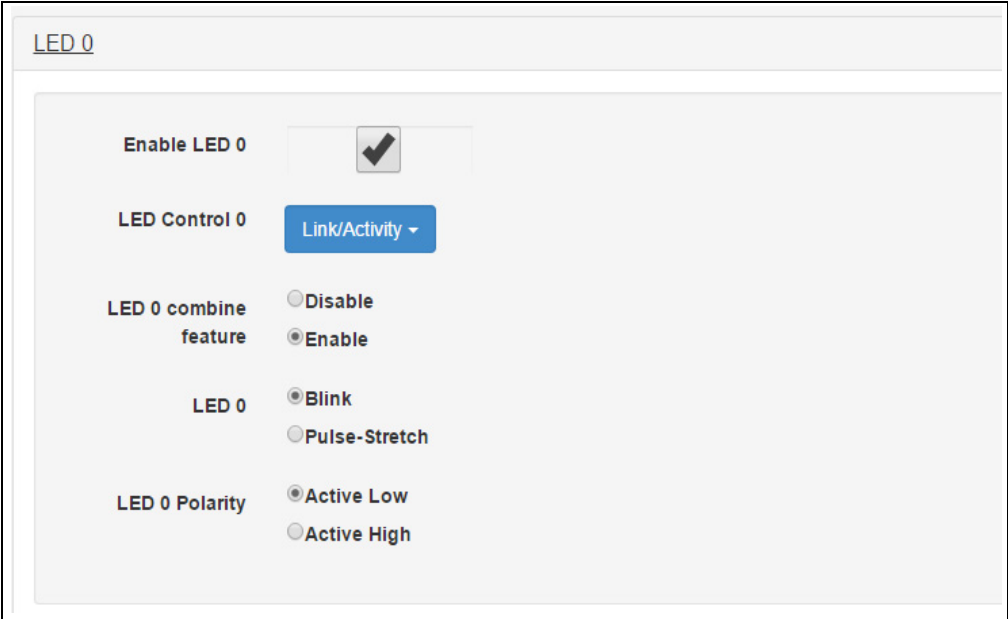
- **Link/Activity:** LED monitors the Link at any speed and its activity.
- **Link1000/Activity:** LED monitors the Link in 1000BASE-T and its activity.
- **Link100/Activity:** LED monitors the Link in 100BASE-TX and its activity.
- **Link10/Activity:** LED monitors the Link in 10BASE-T and its activity.
- **Link100/1000/Activity:** LED monitors the Link in 1000BASE-T and 100BASE-TX and its activity.
- **Link10/1000/Activity:** LED monitors the Link in 1000BASE-T and 10BASE-T and its activity.
- **Link10/100/Activity:** LED monitors the Link in 100BASE-TX and 10BASE-T and its activity.
- **Duplex/Collision:** LED monitors the half-duplex, full-duplex, and their collision status.
- **Collision:** LED monitors the collision in link.
- **Activity:** LED monitors the activity in link.
- **Auto-Negotiation Fault:** LED monitors the auto-negotiation fault in link.
- **Serial Mode:** LED monitors the serial stream.
- **Force LED OFF:** This de-asserts the LED.
- **Force LED ON:** This asserts the LED.

All the four LEDs' behavior can be configured using the fields in [Figure 7-7](#) and [Figure 7-9](#).

FIGURE 7-7: LED CONFIGURATION



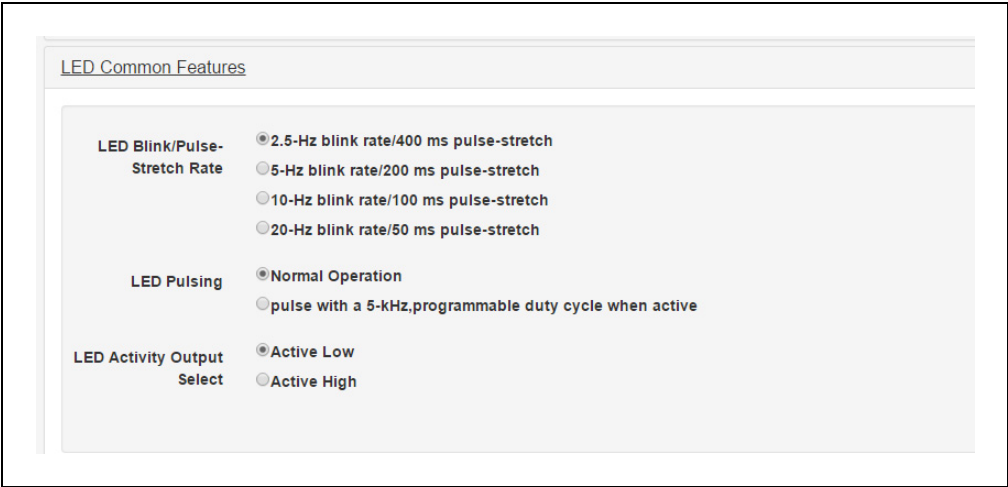
FIGURE 7-8: LED CONFIGURATION FOR LED 0



7.1.1.5.1 LED Common Features

All the four LED behaviors can be configured using the fields in [Figure 7-9](#).

FIGURE 7-9: LED COMMON FEATURES



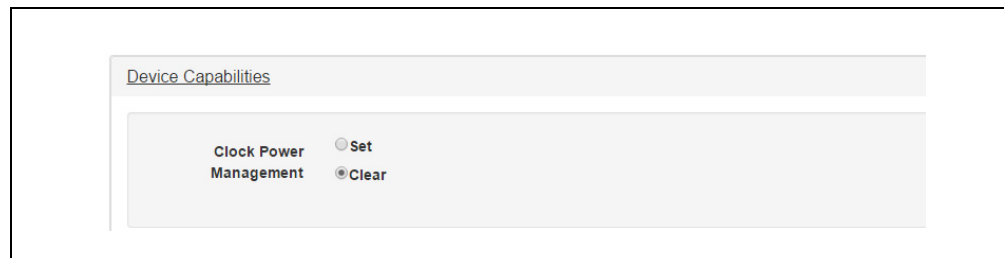
7.1.2 Advanced Features

7.1.2.1 DEVICE CAPABILITIES

7.1.2.1.1 Clock Power Management

Setting this field indicates that the device tolerates the removal of any reference clock(s) via the “clock request” (CLKREQ#) mechanism when the (DBI - W/ Link is in the L1 and L2/L3 Ready Link states. Clearing this field indicates the STKY.) device does not have this capability and that reference clock(s) must not be removed in these Link states. See [Figure 7-10](#).

FIGURE 7-10: DEVICE CAPABILITIES



7.1.2.2 DEVICE CAPABILITIES 2

7.1.2.3 LTR MECHANISM SUPPORT

Setting this field indicates support for the optional Latency Tolerance Reporting (LTR) mechanism. Refer to [Figure 7-11](#).

7.1.2.3.1 OBFF Support

This field indicates whether OBFF is supported or not. It also indicates the signaling mechanism used for supported OBFF. Refer to [Figure 7-11](#).

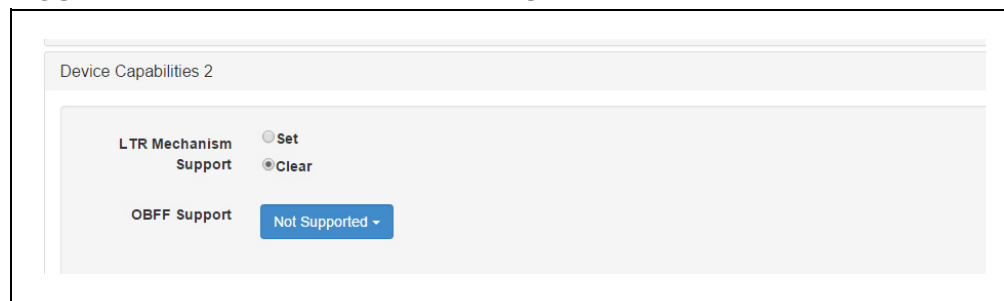
00 – OBFF not supported.

01 – OBFF supported using Message signaling only.

10 – OBFF supported using WAKE# signaling only.

11 – OBFF supported using WAKE# and Message signaling.

FIGURE 7-11: DEVICE CAPABILITIES 2



7.1.2.4 ASPM CONFIGURATION

7.1.2.4.1 ASPM L1 Entry Control

Setting this field indicates that core enters ASPM L1 after a period in which it has been idle. If the option is selected as “L1 entry from L0s” then the core enters ASPM L1 only after an idle period during which both receive and transmit are in L0s.

7.1.2.4.2 ASPM L0s Entrance Latency

ASPM L0s Entrance Latency can be configured with the following options:

- 000 – 1 μ s
- 001 – 2 μ s
- 010 – 3 μ s
- 011 – 4 μ s
- 100 – 5 μ s
- 101 – 6 μ s
- 110 or 111 – 7 μ s

7.1.2.4.3 ASPM L1 Entrance Latency

ASPM L1 Entrance Latency can be configured with the following options:

- 000 – 1 μ s
- 001 – 2 μ s
- 010 – 4 μ s
- 011 – 8 μ s
- 100 – 16 μ s
- 101 – 32 μ s
- 110 or 111 – 64 μ s

A visual representation of the fields is shown in [Figure 7-12](#).

FIGURE 7-12: ASPM CONFIGURATION

The image shows a screenshot of the 'ASPM Configuration' window. It contains three main settings:

- ASPM L1 Entry Control:** Two radio buttons are present. The first, labeled 'L1 entry from L0s', is selected. The second, labeled 'L1 entry directly after Idle', is unselected.
- ASPM L0s Entrance Latency:** A dropdown menu is set to '1 μ s'.
- ASPM L1 Entrance Latency:** A dropdown menu is set to '1 μ s'.

7.1.2.5 L1 PM SUBSTATES CAPABILITIES

7.1.2.5.1 L1 PM Substates Support

When set, this field indicates that this port supports L1 PM Substates.

7.1.2.6 PCI-PM L1.2 SUPPORT

When set, this field indicates that PCI-PM L1.2 is supported.

7.1.2.6.1 PCI-PM L1.1 Support

When set, this field indicates that PCI-PM L1.1 is supported.

7.1.2.6.2 ASPM L1.2

When set, this field indicates that ASPM L1.2 is supported.

7.1.2.6.3 ASPM L1.1

When set, this field indicates that ASPM L1.1 is supported.

Refer to specified fields in [Figure 7-13](#).

FIGURE 7-13: L1 PM SUBSTATES CAPABILITIES

L1 PM Substates Capabilities	
L1 PM Substates Support	<input checked="" type="radio"/> Clear <input type="radio"/> Set
PCI-PM L1.2 Support	<input type="radio"/> Clear <input checked="" type="radio"/> Set
PCI-PM L1.1 Support	<input checked="" type="radio"/> Clear <input type="radio"/> Set
ASPM L1.2	<input type="radio"/> Clear <input checked="" type="radio"/> Set
ASPM L1.1	<input checked="" type="radio"/> Clear <input type="radio"/> Set

7.1.2.7 MAC CONFIGURATION

7.1.2.7.1 Automatic Speed Detection

When set, the MAC ignores the setting of the “MAC Configuration (CFG)” field and automatically determines the speed of operation. The MAC samples the RX_CLK input to accomplish speed detection and reports the last determined speed via the “MAC CFG” field. When reset, the setting of the “MAC CFG” field determines the operational speed.

7.1.2.7.2 MAC Configuration

This field determines the operational speed of the MAC when the Automatic Speed Detection (ASD) bit is reset. When the ASD bit is set, this field is read-only and reports the last determined operational speed.

0 – MII Mode – 10 Mbps

1 – MII Mode – 100 Mbps

2, 3 – RGMII/GMII Mode – 1000 Mbps

7.1.2.7.3 Automatic Duplex Detection

When set, the MAC ignores the setting of the Duplex Mode (DPX) bit and automatically determines the Duplex Operational mode. The MAC uses a PHY LED/signal or DUPLEX pin for the LAN7431 and LAN743F to accomplish mode detection and reports the last determined status via the Duplex Mode (DPX) bit. When reset, the setting of the Duplex Mode (DPX) bit determines Duplex operation.

7.1.2.7.4 Automatic Duplex Polarity

This bit indicates the polarity of the FDUPLEX PHY LED.

0 – DUPLEX asserted low indicates the PHY is in Full-duplex mode.

1 – DUPLEX asserted high indicates the PHY is in Full-duplex mode.

7.1.2.7.5 Duplex Mode

This bit determines the Duplex operational mode of the MAC when the Automatic Duplex Detection (ADD) bit is reset. When the Automatic Duplex Detection (ADD) bit is set, this bit is read-only and reports the last determined Duplex operational mode.

7.1.2.7.6 Energy Efficient Ethernet

When set, the “Energy Efficient Ethernet” operation in the MAC is enabled. When cleared, the “Energy Efficient Ethernet” operation is disabled.

7.1.2.7.7 Energy Efficient Ethernet Tx Clock Stop

When set, the MAC halts the GMII GTX_CLK to the PHY during TX LPI.

This bit is unused in 100 Mbs mode.

7.1.2.7.8 RGMII TXC Delay

This is applicable to LAN7431 SKU only.

0 – RGMII TXC Delay mode disabled

1 – RGMII TXC Delay mode enabled

7.1.2.7.9 RGMII RXC Delay

This is applicable to LAN7431 SKU only.

0 – RGMII RXC Delay mode disabled

1 – RGMII RXC Delay mode enabled

7.1.2.7.10 Reference CLK 25 MHz Out

This is applicable to LAN7431 SKU only.

0 – Disabled

1 – When set, it enables the generation of a 25 MHz reference clock on the REF-CLK_25 pin.

7.1.2.7.11 Generate CLK 125 MHz

This is applicable to LAN7431 SKU only.

1 – When set, the device internally generates a 125 MHz clock for RGMII operation on the TXC pin.

0 – When clear, the device receives a 125 MHz clock from the CLK125 pin.

The MAC Configuration settings are shown in [Figure 7-14](#).

FIGURE 7-14: MAC CONFIGURATION

MAC Configuration

Automatic Speed Detection

☒ Enable
☐ Disable

MAC Configuration

☒ 10 Mbps
☐ 100 Mbps
☐ 1000 Mbps

Automatic Duplex Detection

☒ Enable
☐ Disable

Automatic Duplex Polarity

☐ Asserted Low
☒ Asserted High

Duplex Mode

☒ Half Duplex
☐ Full Duplex

Energy Efficient Ethernet

☒ Enable
☐ Disable

Energy Efficient Ethernet Tx Clock Stop

☐ Enable
☒ Disable

FIGURE 7-15: MAC CONFIGURATION (CONTINUED)

RGMII TXC Delay

☒ disable
☐ enable

RGMII RXC Delay

☒ disable
☐ enable

Reference CLK 25 MHz out

☒ Disable
☐ Enable

Generate CLK 125 MHz

☒ Disable
☐ Enable

7.1.2.8 USER INITIALIZATION TABLE

Based on EEPROM, content application dynamically populates the user initialization table. Application has the provision to edit the values or PCIe register address/CSR register address of the table and it has the option to add new entry with PCIe register/CSR register address can be programmed with desired value directly. See [Figure 7-16](#).

FIGURE 7-16: USER INITIALIZATION TABLE

User Initialization Table ⓘ

Add New Entry

For example, [Figure 7-17](#) shows how to add an entry to write the whole dword of the CSR at offset 0x70 with value 0x12345678.

FIGURE 7-17: USER INITIALIZATION TABLE (CONTINUED)

The screenshot shows the 'User Initialization Table' interface. It has two tabs: 'CSR' (selected) and 'PCIe'. The 'CSR/PCIEcfg Register Address(Hex)' field contains '70'. The 'CSR/PCIEcfg Register Value(Hex)' section has four input fields: 'Byte3(MSB)' with '12', 'Byte 2' with '34', 'Byte 1' with '56', and 'Byte 0(LSB)' with '78'. A close button (X) is in the top right corner.

Individual entries can be removed by clicking the **Click to remove Entry** button. Refer to [Figure 7-18](#).

FIGURE 7-18: USER INITIALIZATION TABLE (CONTINUED)

This screenshot is similar to Figure 7-17, but the 'Byte 0(LSB)' field contains '78'. A red box highlights the 'X' icon in the top right corner, which is labeled 'To remove individual entry'.

All the entries can be removed by clicking the **Remove all Entries** button. See [Figure 7-19](#).

FIGURE 7-19: USER INITIALIZATION TABLE (CONTINUED)

The screenshot shows the 'User Initialization Table' interface with two buttons: 'Add New Entry' and 'Remove All Entries'. The 'Remove All Entries' button is highlighted with a red box. Below the buttons, the 'CSR/PCIEcfg Register Value(Hex)' section shows 'Byte3(MSB)' as '0', 'Byte 2' as '0', 'Byte 1' as '33', and 'Byte 0(LSB)' as '44'.

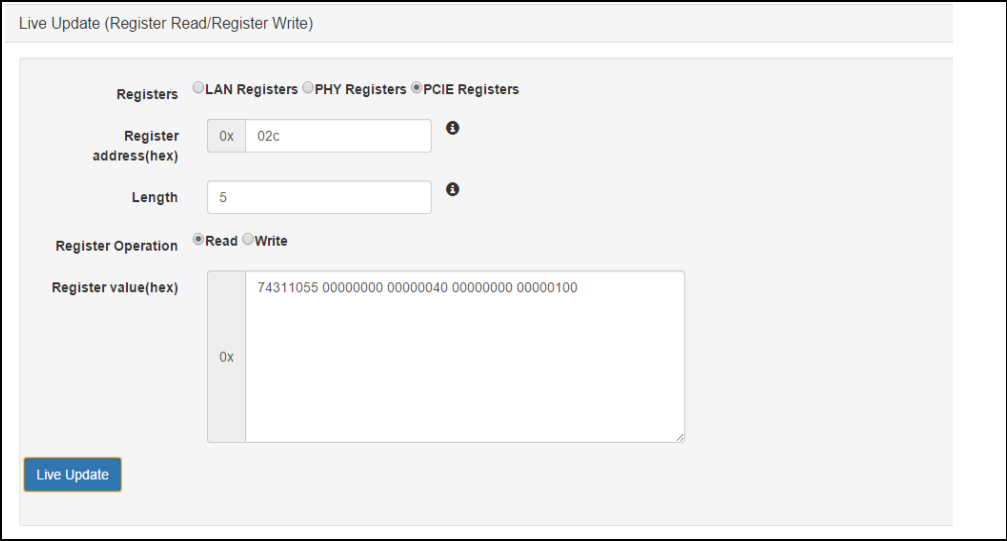
7.1.2.9 LIVE UPDATE

Live Update enables the option of Register Read/Write for LAN registers, PHY registers and PCIe registers. Register Address and the length of bytes to read/write is given in "Register Address" and "Length" fields. See [Figure 7-20](#) and [Figure 7-21](#).

FIGURE 7-20: LIVE UPDATE

The screenshot shows the 'Live Update (Register Read/Register Write)' interface. It has a 'Registers' section with three radio buttons: 'LAN Registers' (selected), 'PHY Registers', and 'PCIe Registers'. Below this, the 'Register address(hex)' field contains '0x 10' and the 'Length' field contains '1'. The 'Register Operation' section has two radio buttons: 'Read' (selected) and 'Write'. The 'Register value(hex)' field contains '00700000'. A 'Live Update' button is at the bottom left.

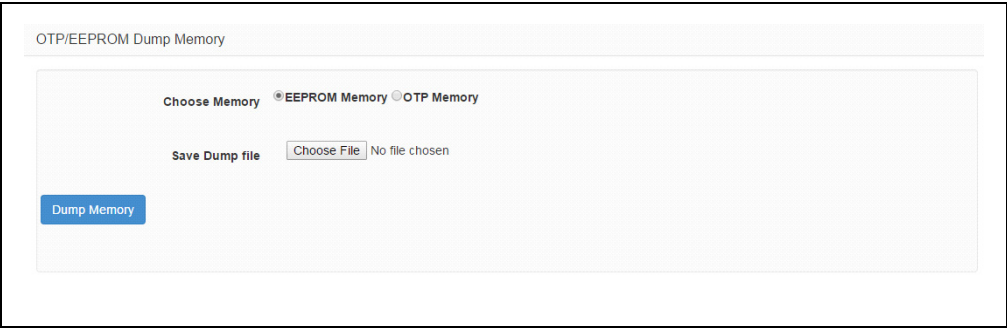
FIGURE 7-21: LIVE UPDATE (CONTINUED)



7.1.2.10 OTP/EEPROM DUMP MEMORY

Dump memory allows the user to dump OTP or EEPROM memory. The OTP or EEPROM memory dump is saved in the selected file with the .bin extension. See [Figure 7-22](#).

FIGURE 7-22: OTP/EEPROM DUMP MEMORY



7.1.3 Special Features

7.1.3.0.1 Ping/External Loopback Demo

Ping Operation/External Loopback test can be done by selecting the **Radio** button and apply. See [Figure 7-23](#).

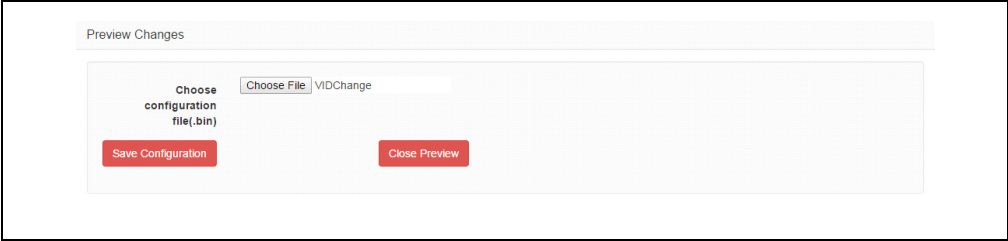
FIGURE 7-23: PING/EXTERNAL LOOPBACK DEMO



7.1.3.0.2 Preview Changes

The Preview Changes option allows the user to generate the configuration file that is going to be programmed in the device. A configuration file with .bin extension is generated. See [Figure 7-24](#).

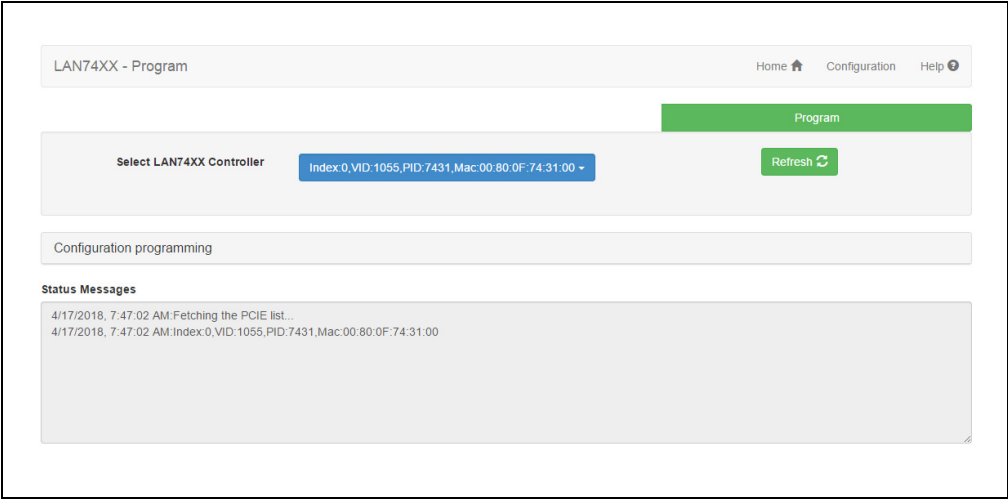
FIGURE 7-24: PREVIEW CHANGES



7.2 PROGRAMMING PAGE

The Programming Page enables the user to program the configuration file in EEPROM or OTP memory. See [Figure 7-25](#).

FIGURE 7-25: PROGRAMMING PAGE



7.2.1 Program Once

Select the device of interest from the **Select the LAN7400 Controller** dropdown menu and choose the memory to be programmed. See [Figure 7-26](#).

FIGURE 7-26: PROGRAM ONCE

7.2.1.1 PROGRAM CONFIGURATION FILE WITH MAC ADDRESS

The configuration file is programmed in either OTP or EEPROM with the given MAC address overwriting the values in the configuration file. Refer to [Example 7-1](#).

EXAMPLE 7-1: EXAMPLE FOR SINGLE PROGRAMMING

MAC Address in Single Programming:

Mac Address: "00:80:0F:74:30:02"

If the above values are given, then the device will be programmed with MAC Address "00:80:0F:74:30:02"

7.3 OFFLINE LAN74XX CONFIGURATION PAGE

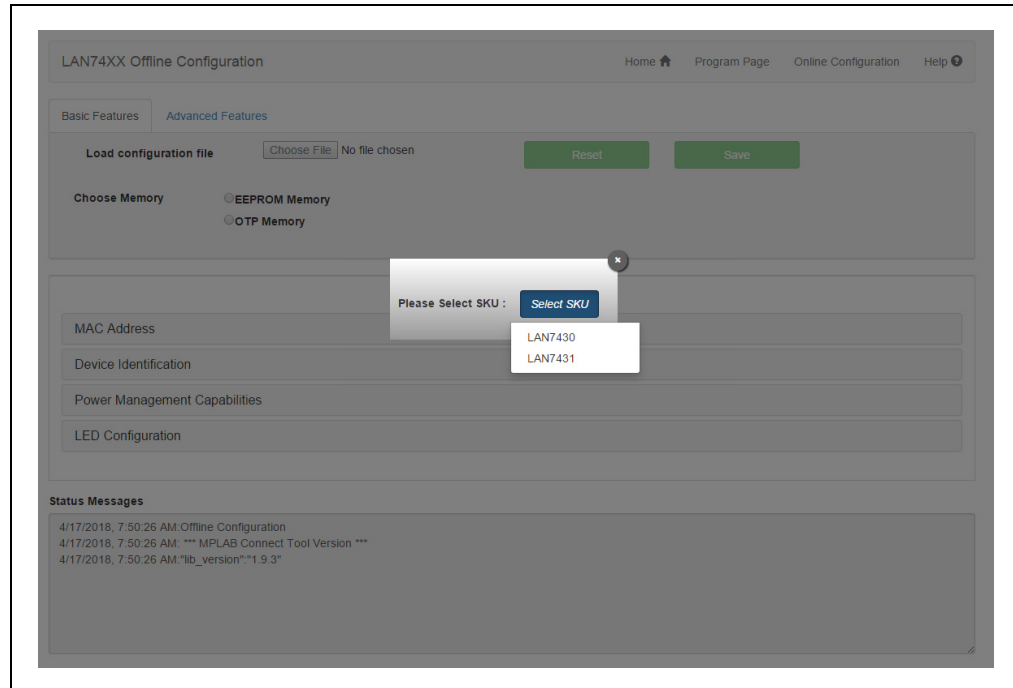
Offline support allows for generation of a configuration file without the hub being connected to the system. Follow the steps below:

1. Select the **Radio** button on the landing page as shown in [Figure 7-27](#).

FIGURE 7-27: SELECTING OFFLINE MODE

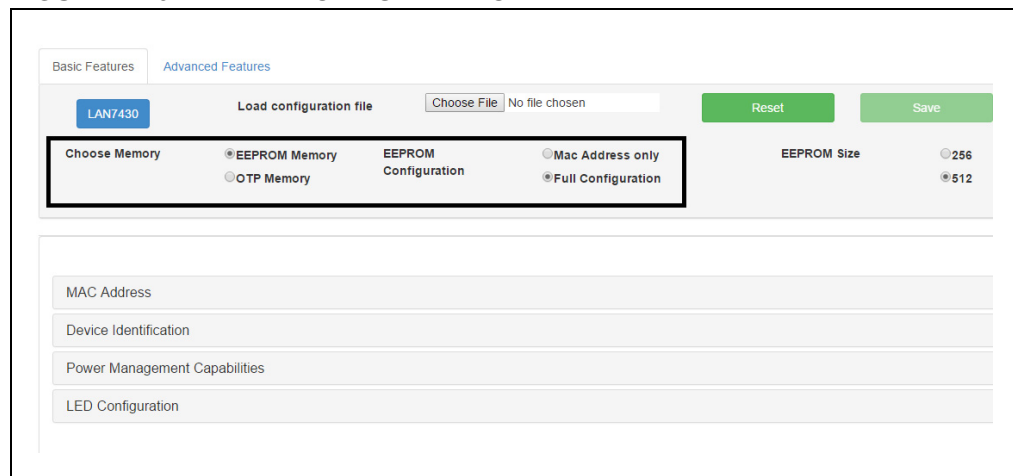
2. Select the part number of interest from the dropdown, as shown in [Figure 7-28](#). Choose the memory type for which the configuration file is to be generated.

FIGURE 7-28: SELECTING THE PART NUMBER



3. If EEPROM is chosen, then the size of EEPROM to be programmed must be selected as shown in [Figure 7-29](#).

FIGURE 7-29: MEMORY SELECTION

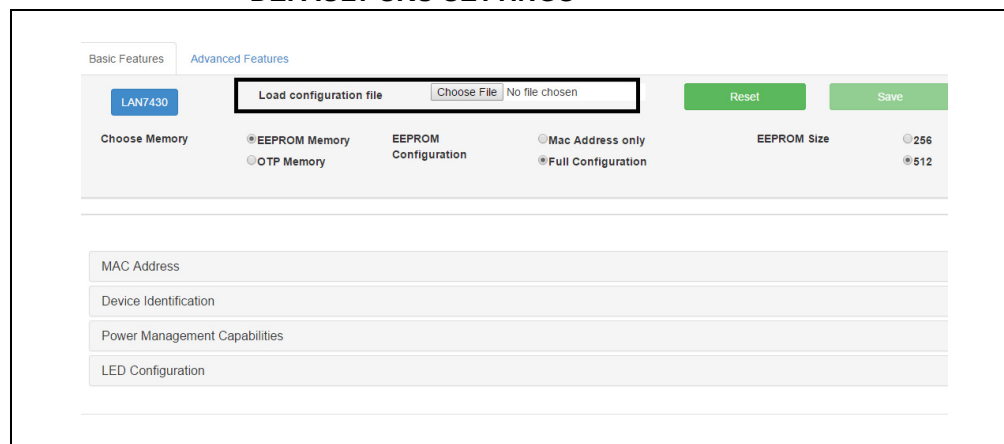


4. If the user wants to generate a configuration file for a device whose configuration is already modified, then that device's dump file should be loaded using the "Load Configuration File" option. This should be done before any changes are performed to the configuration settings. Once changes are made to the default settings, the "Save Configuration File" option becomes active instead of the "Load Configuration File".

The **Save** button is inactive at this time since no changes are made to the configuration settings.

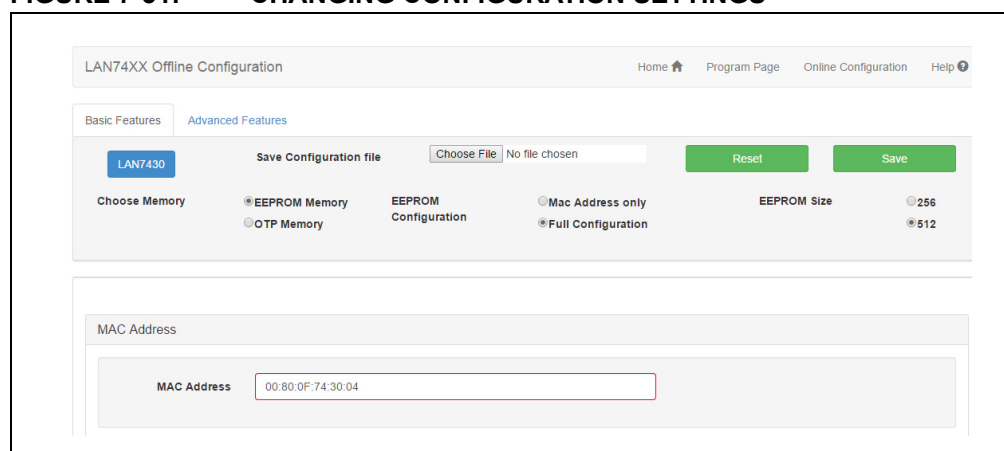
Note: If the user wants to generate a configuration file for a device whose OTP is already programmed once, then the OTP dump file of the device should be loaded using "Load Configuration File" option. See [Figure 7-30](#).

FIGURE 7-30: LOADING A CONFIGURATION FILE ON TOP OF THE DEFAULT SKU SETTINGS



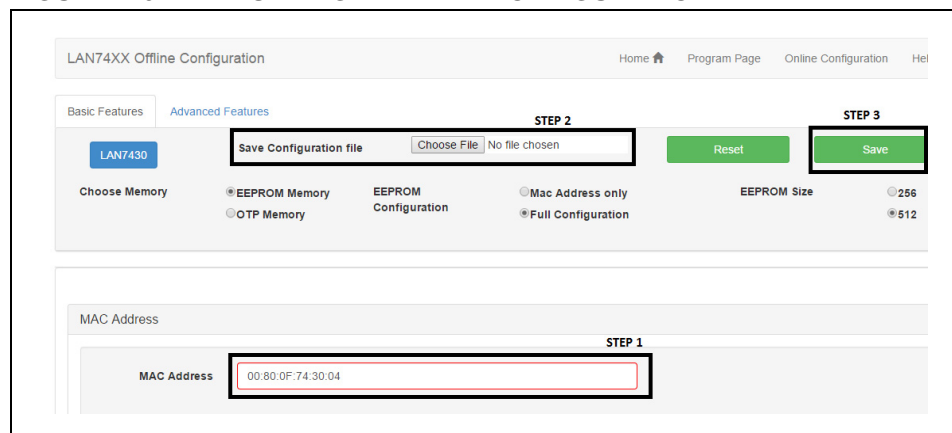
5. Configuration settings of the hub can be changed from the default or default+Load configuration settings as shown in [Figure 7-31](#). The **Save** button becomes active as soon as any changes are done.

FIGURE 7-31: CHANGING CONFIGURATION SETTINGS



6. If the changes are to be reversed, then the **Reset** button can be used to reset to default settings.
7. Once the required changes are made, the user can select the File name and click on the **Save** button. See [Figure 7-32](#).

FIGURE 7-32: SAVING THE NEW CONFIGURATION FILE



Appendix A. Troubleshooting and Error Codes

A.1 TROUBLESHOOTING

1. Check if you have installed the VSM and WinUSB drivers before running this tool using the CLI. Navigate to `MPLABConnect_CLI` folder and run the following command as an administrator from the CMD line. Refer to the *MPLAB® Connect Configurator CLI User's Guide* for more details.

```
>>MPLABConnect.exe /i
```

2. If the MCHP hub is connected to a USB 3.1 Gen1 Host controller, check if power saving is disabled in the hub using the device manager (Refer to [Appendix B. "Disable Power Management"](#)) or connect a USB device (like a mouse or pen drive) to one of the downstream ports to prevent the hub from going to sleep.
3. If still having issues, email the `MPLABConnect.log` file created in the same directory as the tool to your Microchip support contact.

Debugging failures can be done by analyzing the log file generated which contains the error codes as mentioned in [Section A.2 "Error Codes"](#). The `MPLABConnect.log` file is created in the same directory as the Microchip application.

A.2 ERROR CODES

An error code would be displayed when an error occurs. Detailed descriptions of these codes are provided in [Table A-1](#).

TABLE A-1: LIST OF ERROR CODES

Error Code	Description
0x0000	Success; no errors
0x0001	The specified device was not found.
0x0003	Device handle passed to the API is not valid.
0x0004	API of the WinUSB library failed.
0x0005	System reboot is required.
0x0006	Error in installing VSM filter driver
0x0007	Operation successful but requires reboot
0x0008	Bin file size is invalid.
0x0009	Error while reading <code>cfg/bin</code> file
0x0011	Error in installing WinUSB driver
0x0012	Invalid argument
0x0013	Error when VSM filter is not available
0x0014	Error when application does not have administrator rights
0x0015	Error if VSM command failed due to power state of the device
0x0016	Error if firmware and configuration file are programmed at one shot for USB253X/USB(8)4604 hubs
0x0017	Error if device memory reached maximum size
0x0018	Error if OTP programmed content does not match with the input configuration
0x1000	Could not load the binary file

TABLE A-1: LIST OF ERROR CODES (CONTINUED)

Error Code	Description
0x1001	Reading from SPI flash failed.
0x1002	File size did not match.
0x1003	SPI pass through write command failed.
0x1004	SPI pass through Enter command failed.
0x1005	SPI flash could not be detected or was not present.
0x1008	SPI pass through Exit command failed.
0x1009	SPI pass through read command failed.
0x100A	Unsupported SPI flash detected
0x100B	SPI flash read back and compare failed with programmed binary.
0x100E	SRAM programming failed.
0x100F	SPI flash Erase Signature failed.
0x1011	Could not load the Json file
0x1012	Could not load the INI file
0x1013	Flex register field was not programmed.
0x1014	SPI flash access was not supported for the device.
0x2000	Cannot enable I ² C pass-through interface
0x2002	I ² C Transfer failed.
0x3000	Maximum configuration block is already programmed.
0x4000	Communication at the specified baud rate will be error-prone.
0x4001	Cannot set USB2534 UART registers (most likely a command failure)
0x4002	Transmit failed without transmitting any data.
0x4003	Transmit failed after transmitting some data.
0x4004	Receive failed due to buffer overrun, reduce baud rate.
0x4005	Receive failed due to unexpected Rx FIFO status.
0x4006	Receive failed since worker thread creation failed.
0x4007	UART Rx is pending due to Asynchronous mode.
0x4009	UART Receive command failed by the firmware.
0x400B	UART Receive timeout
0x5000	Invalid GPIO pin number
0x6000	Cannot access LAN78XX products
0x6001	LAN78XX adapter busy
0x6002	Requested LAN operation failed.
0x6003	Physical EEPROM is absent and OTP is blank.
0x6004	EEPROM absent and OTP has free space
0x6005	EEPROM absent and no free space in OTP
0x6006	EEPROM present and no free space in OTP
0x6007	EEPROM present and OTP is blank
0x6008	EEPROM present and OTP has invalid signature
0x6009	EEPROM absent and OTP has invalid signature
0x600A	EEPROM absent
0x600B	EEPROM is present and the highest priority goes to EEPROM.
0x7000	Ping Operation failed.
0x7001	Parameter passed to the API is invalid.
0x7002	Cannot restart LAN74XX device. Requires administrator rights.
0xFFFF	Unknown error occurred

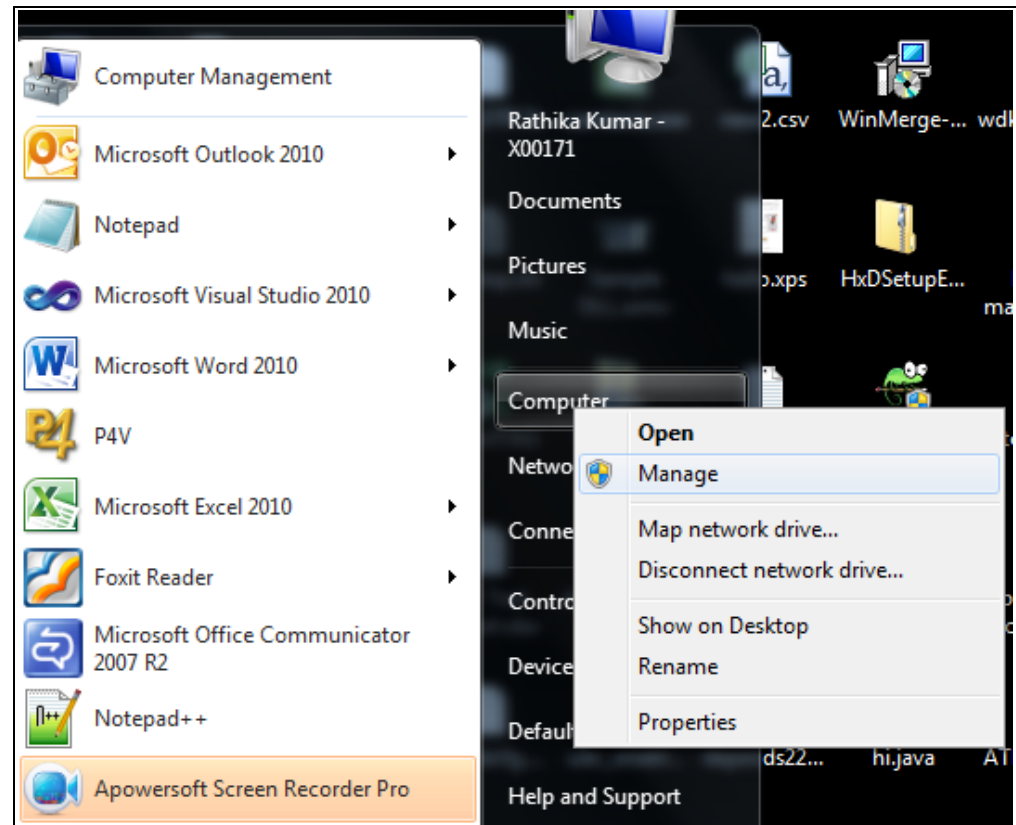
Appendix B. Disable Power Management

B.1 WINDOWS 7

To disable power management in Windows 7:

1. Right-click My Computer.
2. From the context menu, click Manage. See [Figure B-1](#).

FIGURE B-1: MANAGE MENU OPTION



3. Select Device Manager. See [Figure B-2](#).

FIGURE B-2: DEVICE MANAGER

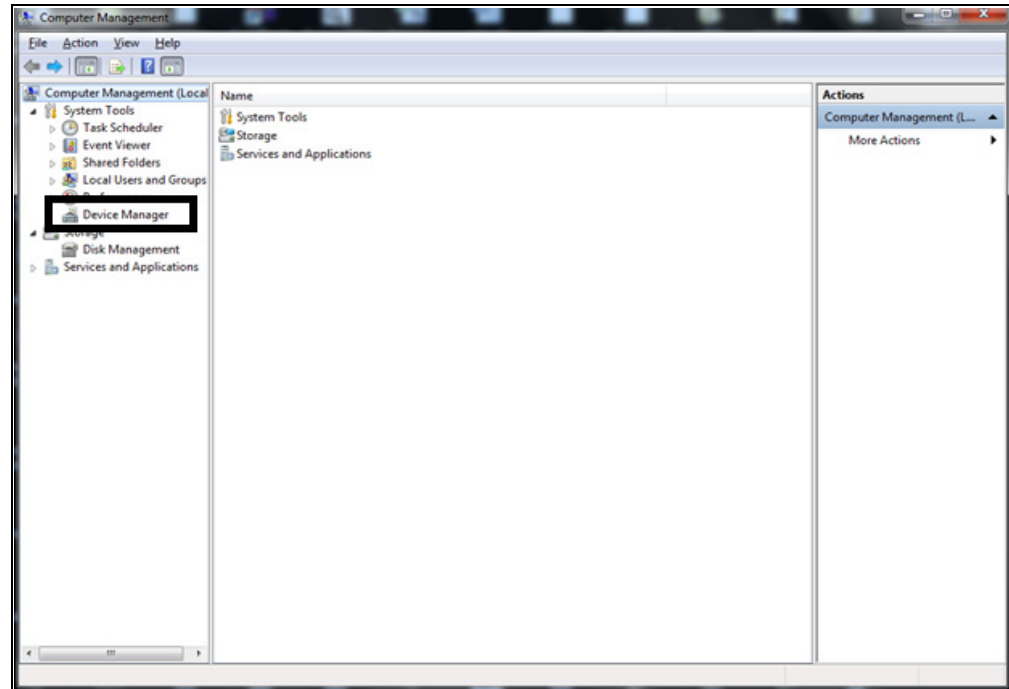
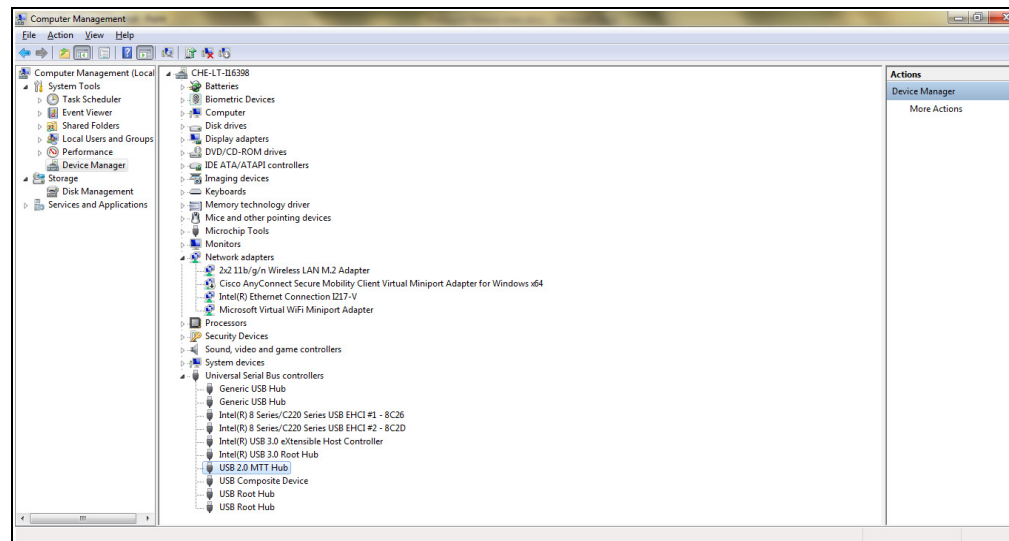


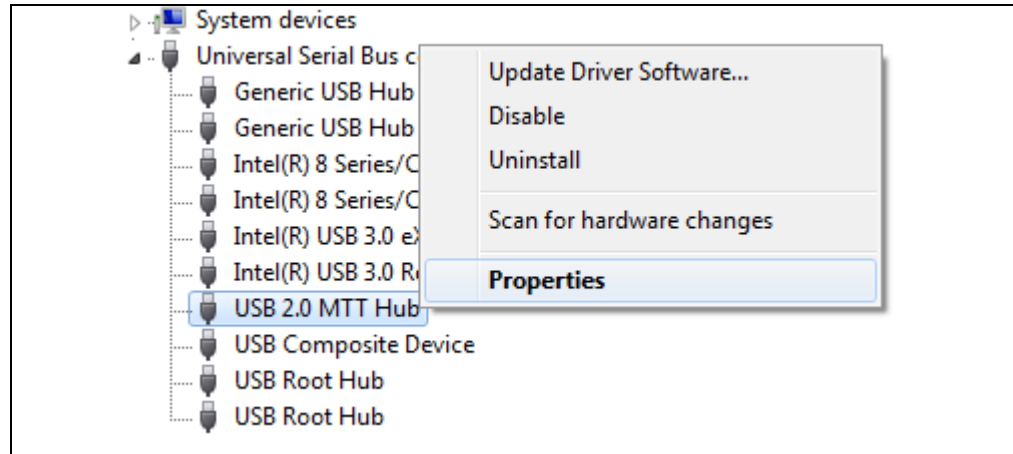
FIGURE B-3: COMPUTER MANAGEMENT



4. Search for the hub based upon the VID and PID.
5. Right-click the hub.
6. Click Properties. See [Figure B-4](#).

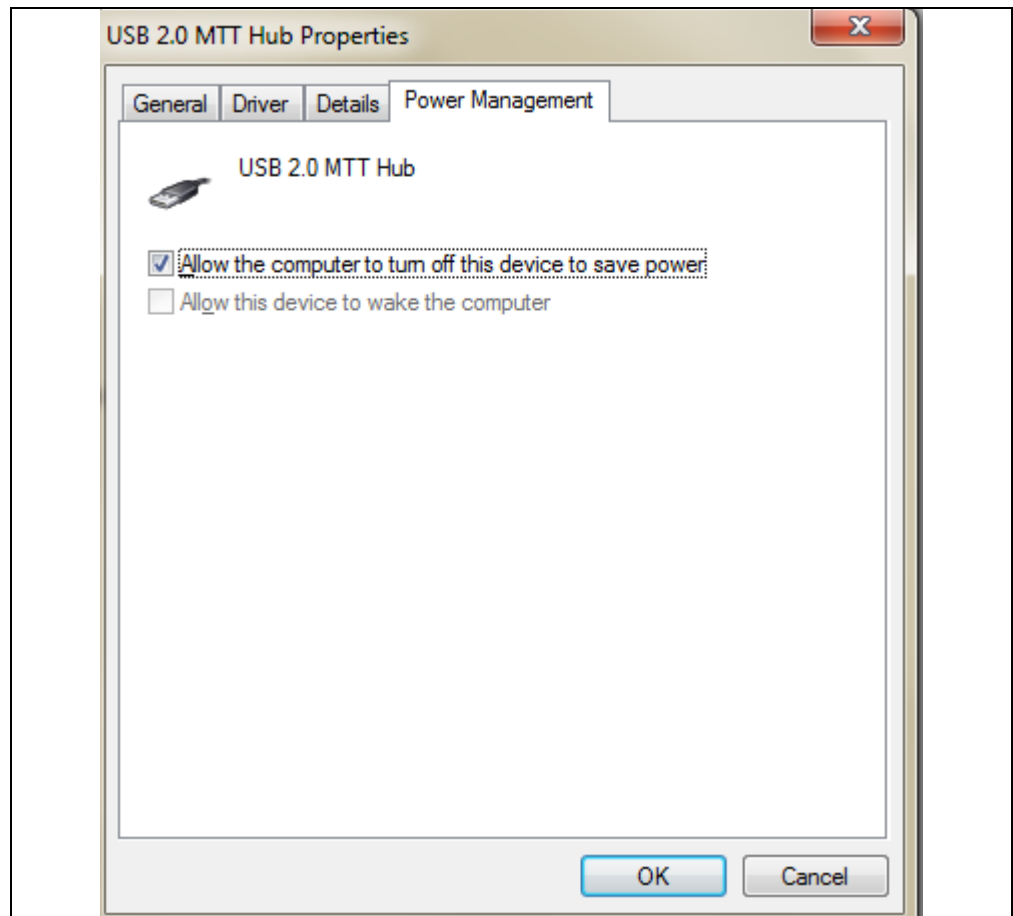
Disable Power Management

FIGURE B-4: PROPERTIES MENU OPTION



7. Select the **Power Management** tab. See [Figure B-5](#).

FIGURE B-5: POWER MANAGEMENT TAB



8. To disable Power Management, uncheck the “Allow the computer to turn this device off to save power” check box.
9. Click **OK**.

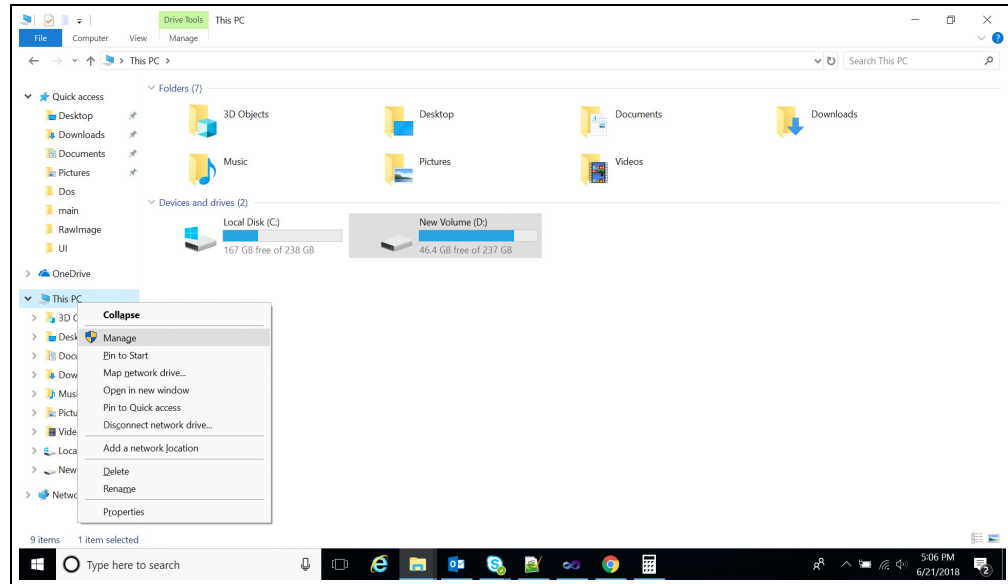
Note: Repeat the above steps for the USB 3.0 MTT hub if the connected hub is USB 3.1 Gen1.

B.2 WINDOWS 8.1

To disable power management in Windows 8.1:

1. Right-click My Computer.
2. From the context menu, click Manage. See [Figure B-6](#).

FIGURE B-6: MANAGE MENU OPTION



3. Repeat Steps 3-8 as in [Windows 7](#).

Appendix C. Protouch/MPLABConnect Interoperability

C.1 INTRODUCTION

The MPLAB® Connect Configurator GUI can be used on a system which has ProTouch 1 (PT1) previously installed

(<http://www.microchip.com/SWLibraryWeb/product.aspx?product=Protouch>).

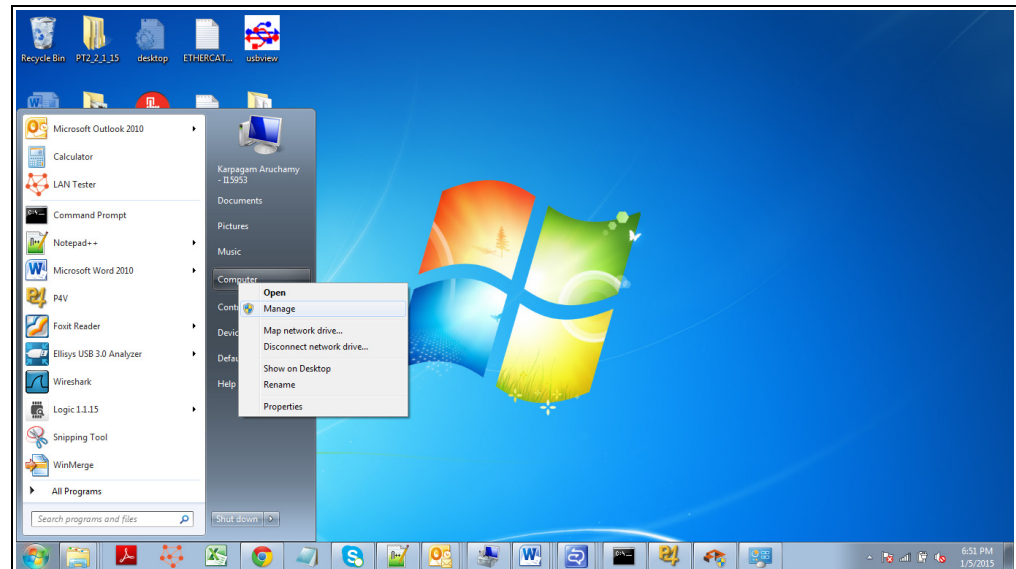
To use the MPLAB Connect Configurator GUI application, the user can install MPLAB Connect Configurator drivers and start using the MPLAB Connect Configurator GUI or CLI.

Note: PT1 and MPLAB Connect Configurator GUI use different drivers. As such, if user wants to use PT1, the drivers have to be restored after installing the MPLAB Connect Configurator GUI.

To restore drivers:

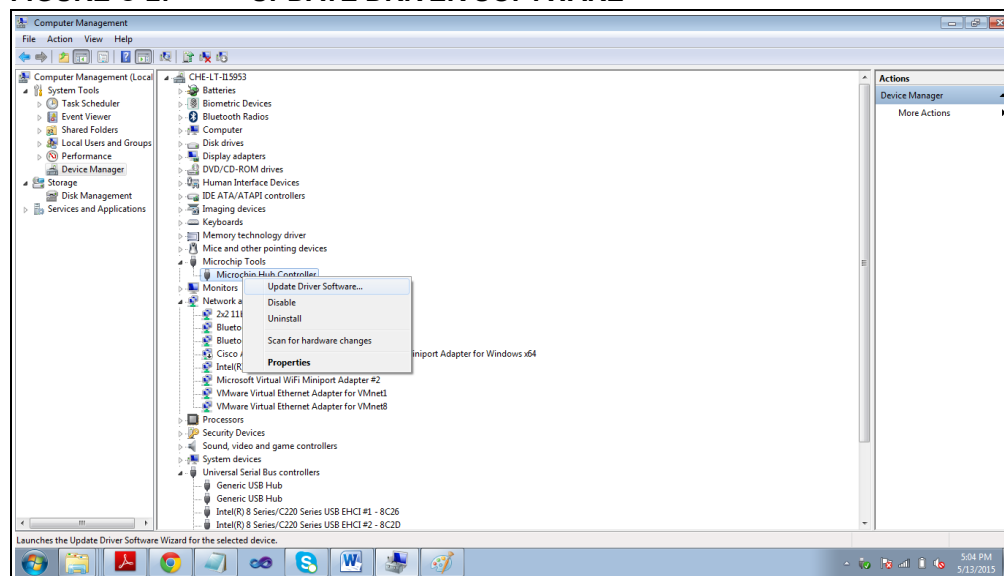
1. Open the PT1 tool after connecting a Microchip hub.
HFC (internal USB device) would be enabled with MPLAB Connect Configurator GUI driver.
2. Manually load the WinUSB driver.
To start, open Device Manager as in [Figure C-1](#).

FIGURE C-1: OPEN DEVICE MANAGER



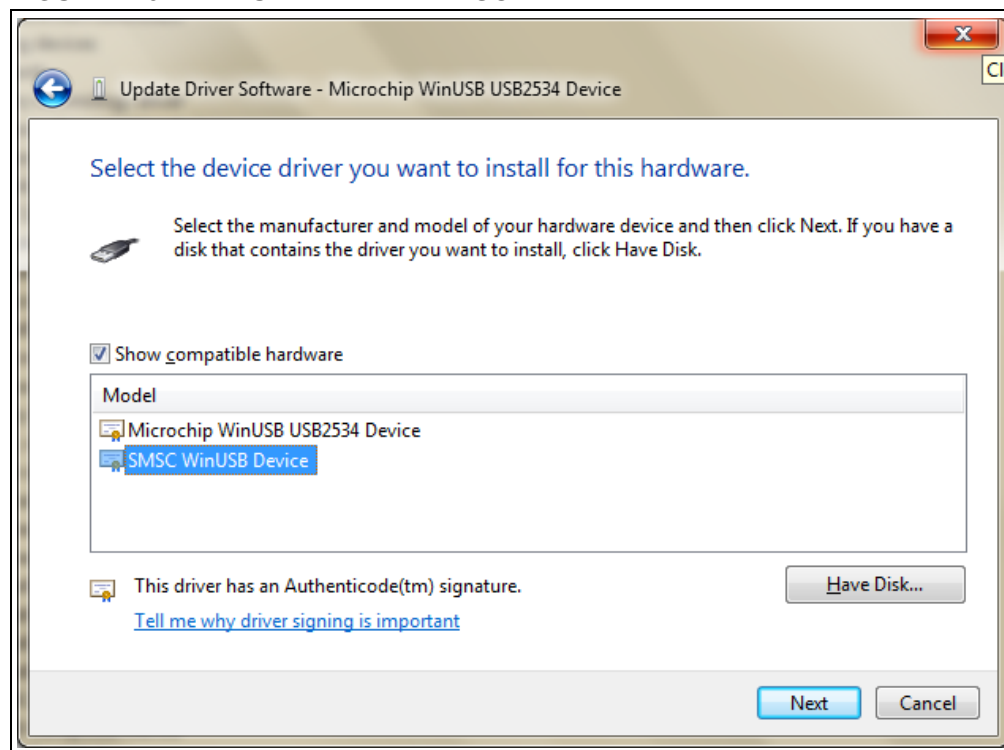
3. Right-click the WinUSB Device and select Update Driver Software as in [Figure C-2](#).

FIGURE C-2: UPDATE DRIVER SOFTWARE



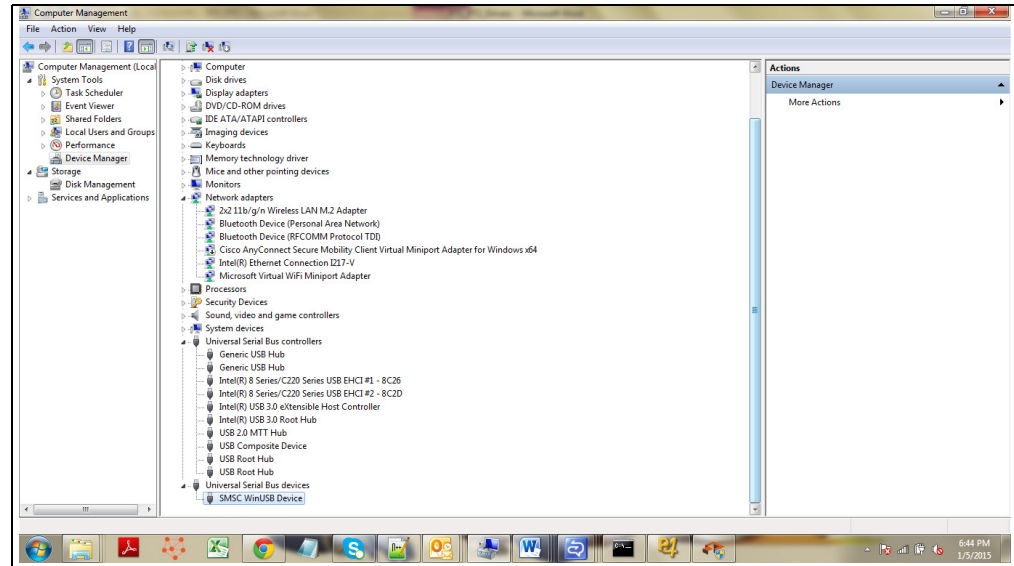
4. Select the SMSC WinUSB Device option, as in [Figure C-3](#).

FIGURE C-3: UPDATE DRIVER SOFTWARE



The WinUSB Device would be loaded with the SMSC WinUSB device driver. See [Figure C-4](#).

FIGURE C-4: COMPUTER MANAGEMENT



The user can now start using the PT1 tool.

5. Install the MPLAB Connect Configurator drivers for running the MPLAB Connect Configurator GUI application.

NOTES:

Appendix D. Verifying MPLABConnect Driver Installation

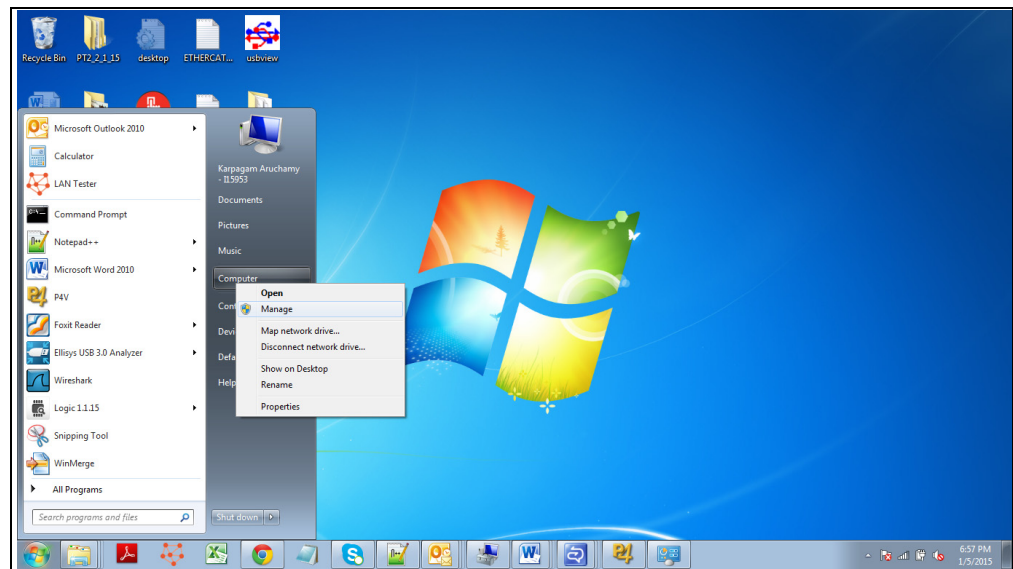
D.1 INTRODUCTION

This section details how to check if the drivers (VSM and WinUSB) were installed correctly.

D.1.1 VSM as Class Filter

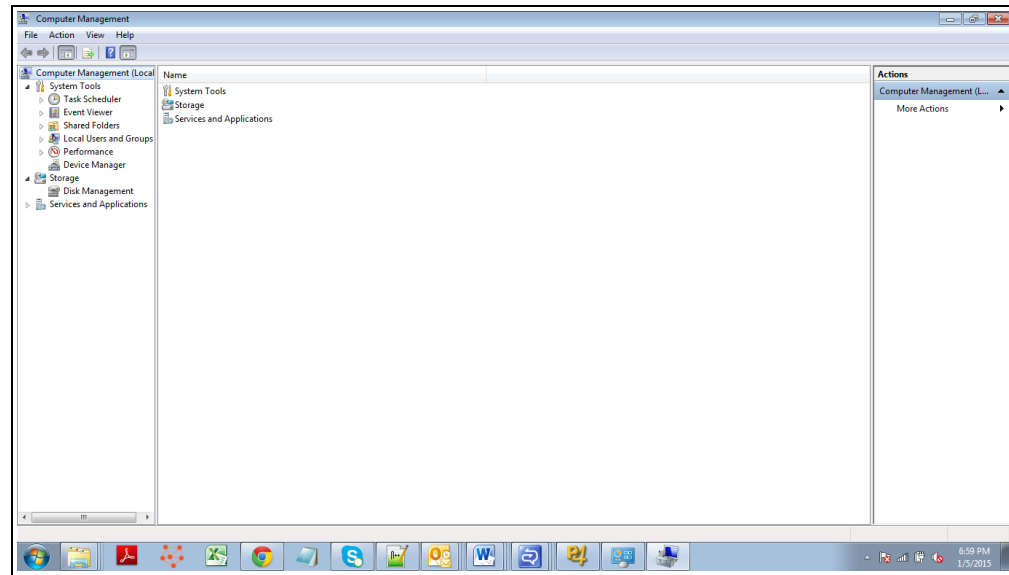
1. Right-click My Computer.
2. Select Manage. See [Figure D-1](#).

FIGURE D-1: MANAGE OPTION



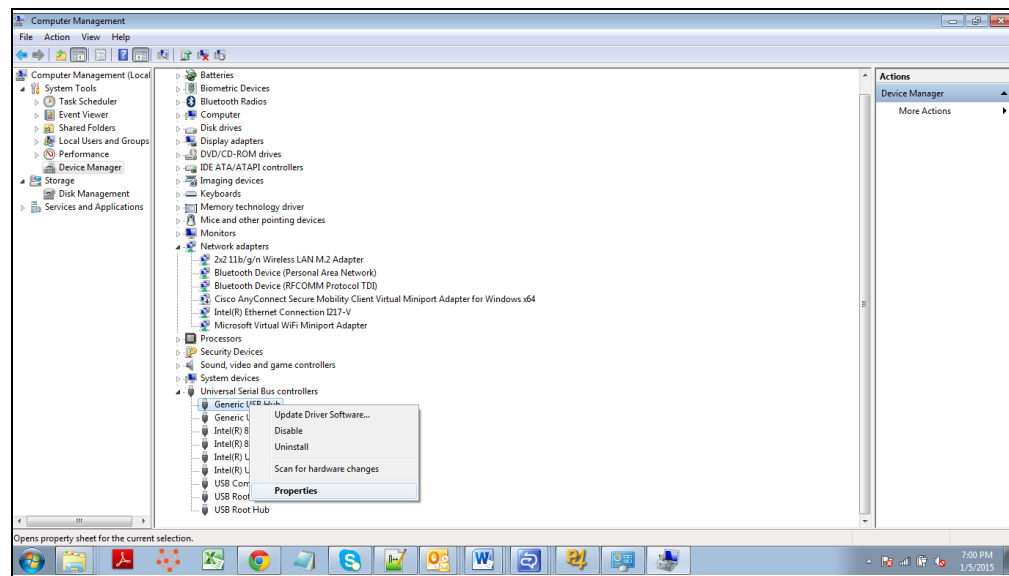
The Computer Management window displays. See [Figure D-2](#).

FIGURE D-2: COMPUTER MANAGEMENT



3. Click Device Manager.
4. Right-click the desired hub.
5. Click Properties. See [Figure D-3](#).

FIGURE D-3: PROPERTIES OPTION

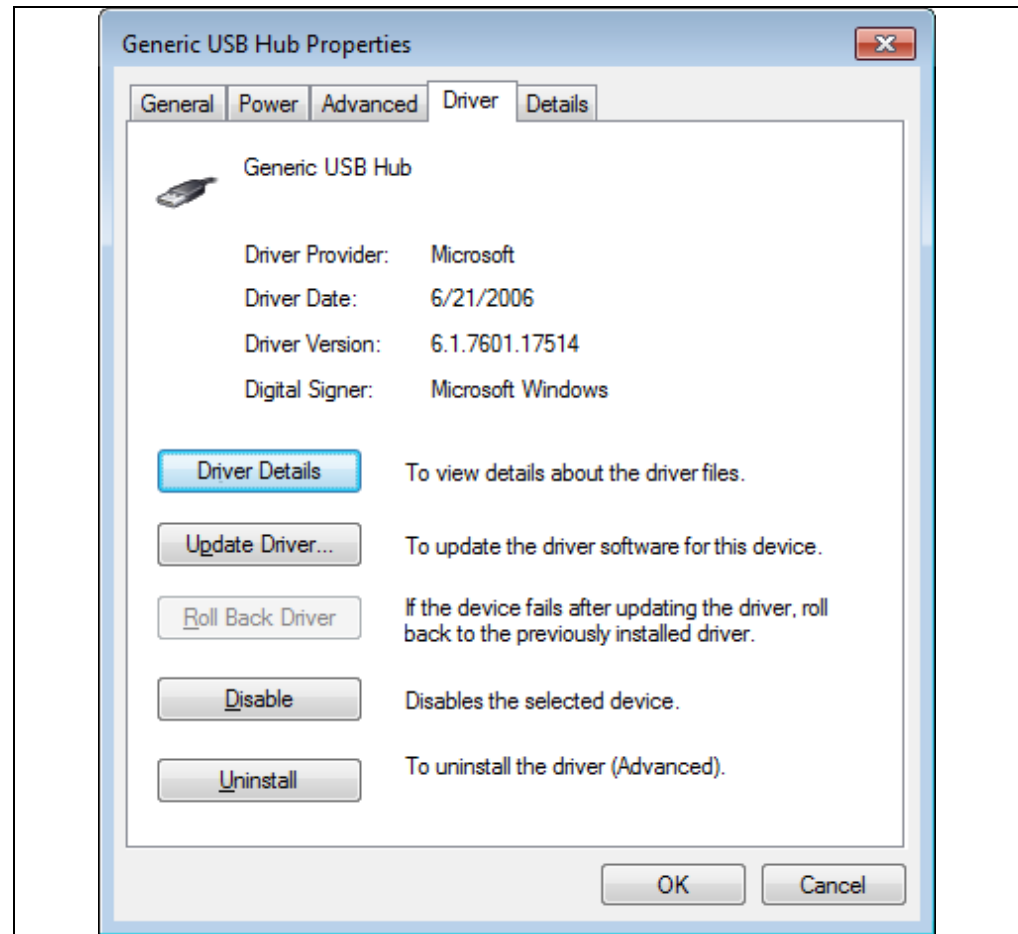


The Hub Properties window displays.

Verifying MPLABConnect Driver Installation

6. Click the **Driver** tab. See [Figure D-4](#).

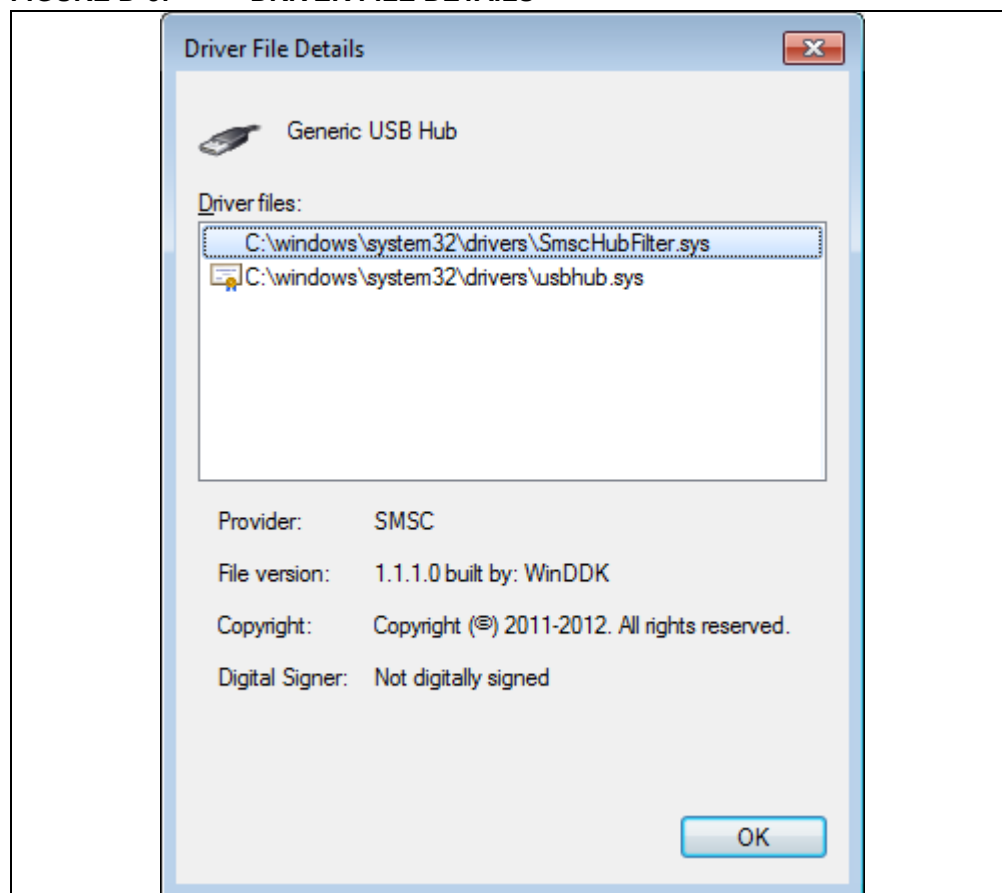
FIGURE D-4: DRIVER TAB



7. Click Driver Details.

The Driver File Details dialog displays. See [Figure D-5](#).

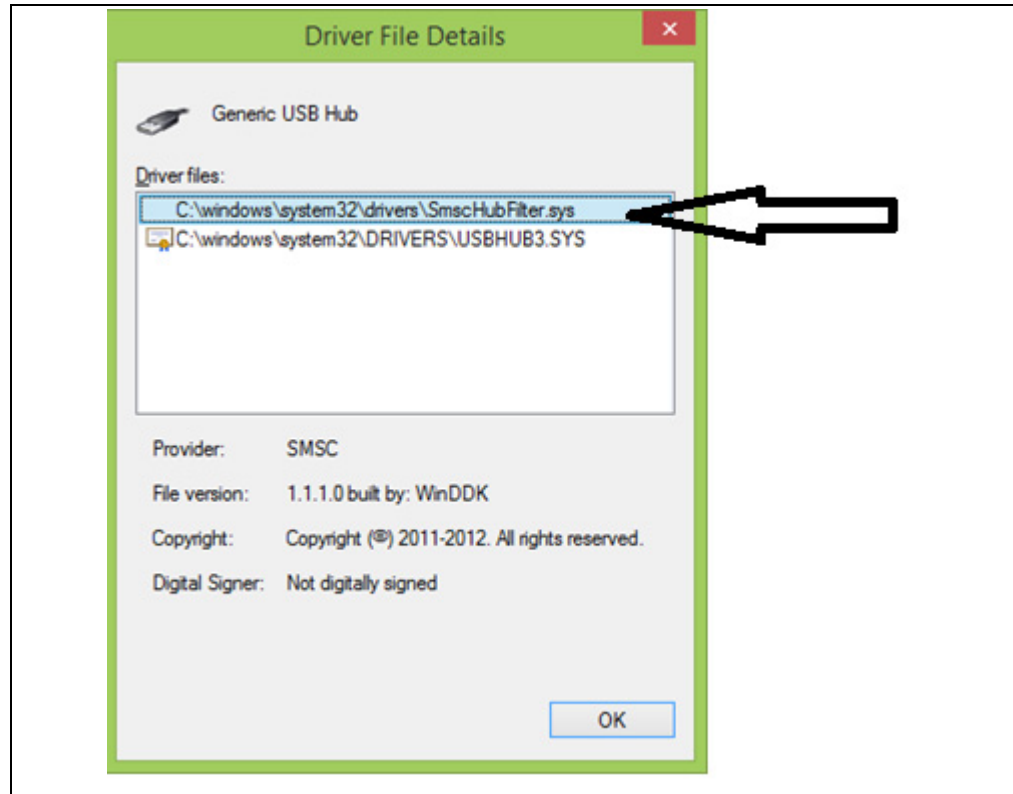
FIGURE D-5: DRIVER FILE DETAILS



Verifying MPLABConnect Driver Installation

If the VSM driver was installed as a class filter, there will be an entry for `Smschubfilter.sys`. See [Figure D-6](#).

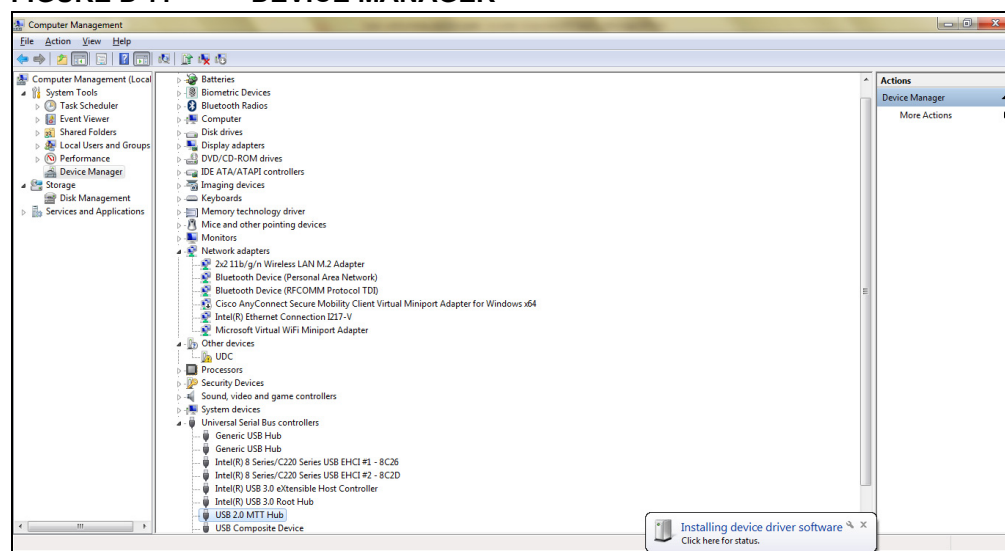
FIGURE D-6: SMSCHUBFILTER.SYS



D.1.2 WinUSB Driver

It may take a few seconds to load the WinUSB driver. While it is loading, its status can be monitored in the device manager. See [Figure D-7](#).

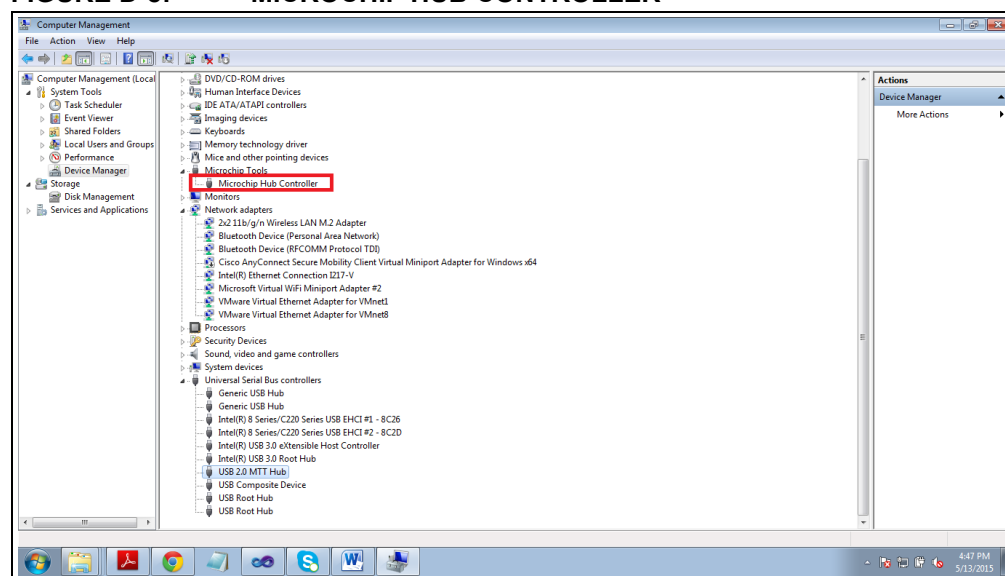
FIGURE D-7: DEVICE MANAGER



To check the installation of the WinUSB driver:

1. The HFC (Internal USB device) will be enabled once the MPLAB® Connect Configurator GUI tool is launched. The Microchip hub should be connected. See [Figure D-8](#).

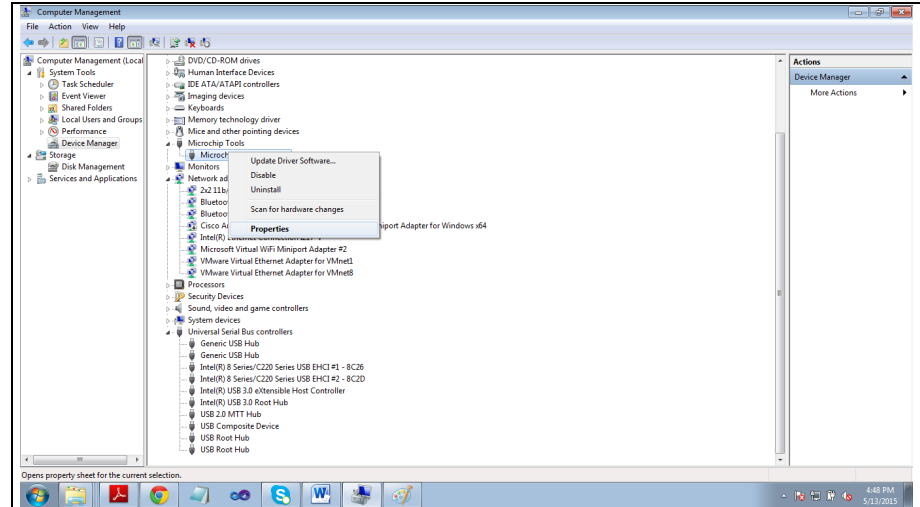
FIGURE D-8: MICROCHIP HUB CONTROLLER



2. Right-click the HFC in the device manager.
3. Click Properties. See [Figure D-9](#).

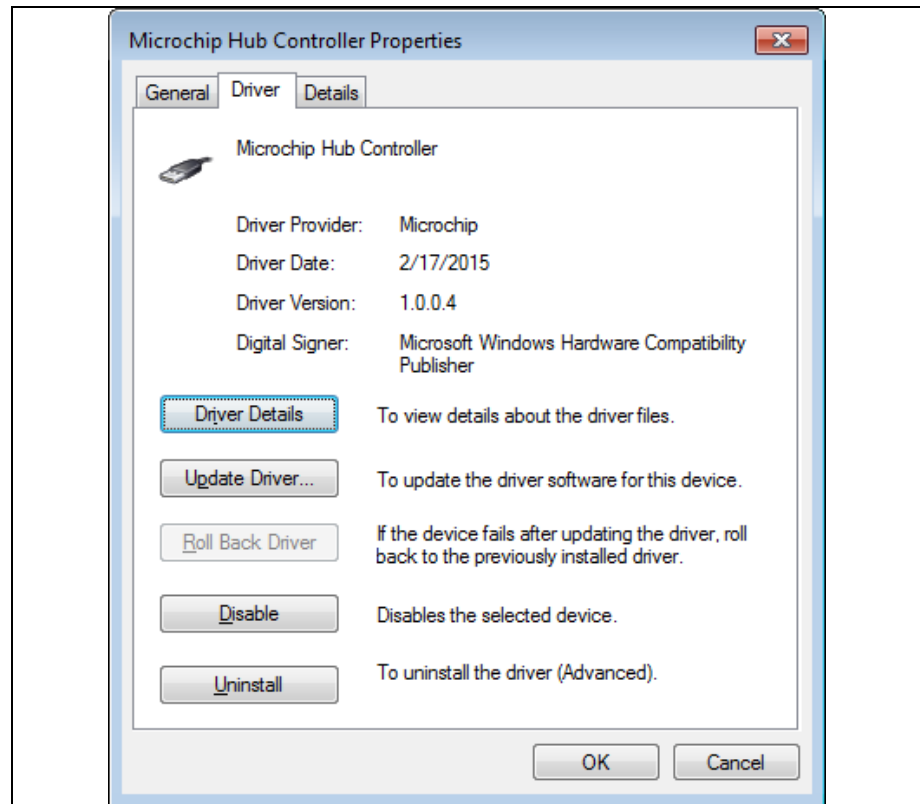
Verifying MPLABConnect Driver Installation

FIGURE D-9: PROPERTIES OPTION



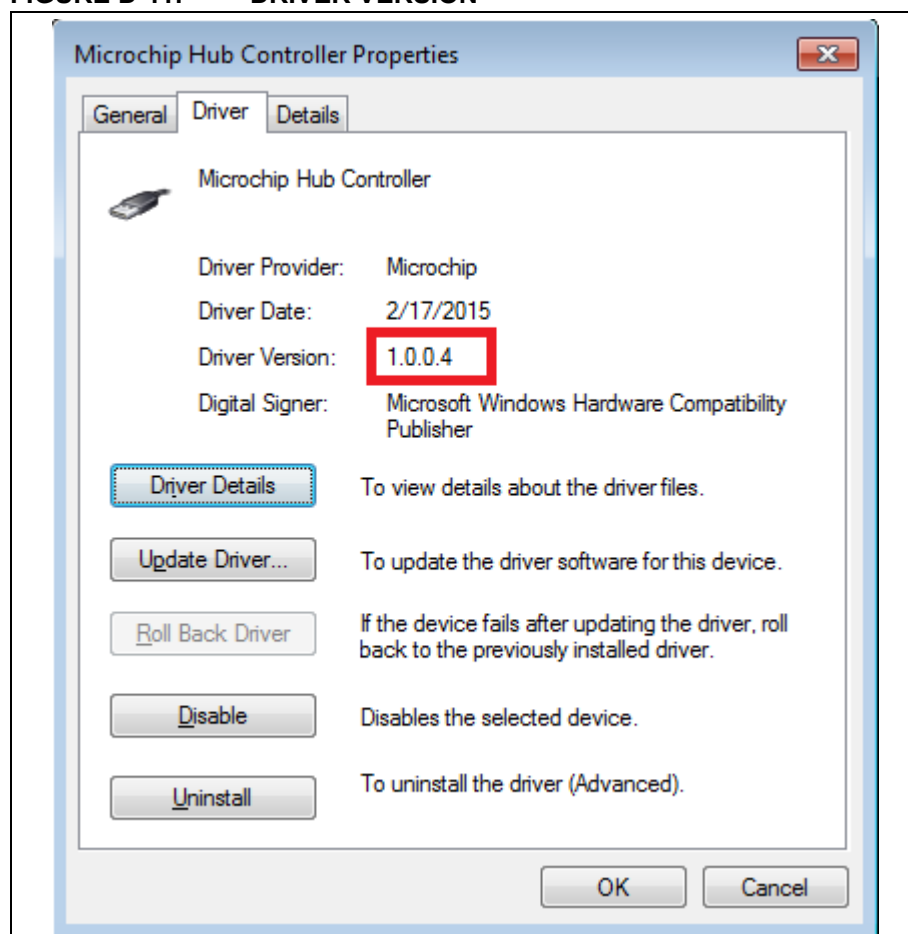
4. The Microchip Hub Controller Properties dialog opens. See [Figure D-10](#).

FIGURE D-10: MICROCHIP HUB CONTROLLER PROPERTIES



The driver version should be 1.0.0.4, as highlighted in [Figure D-11](#).

FIGURE D-11: DRIVER VERSION



Appendix E. Supported Configuration Items

E.1 INTRODUCTION

Supported configuration items for USB Hubs that can be used in .json file are specified in [Table E-1](#) to [Table E-8](#).

TABLE E-1: CONFIGURATION ITEMS COMMON FOR USB253X/USB(8)4604/USB3X13, USB57XX, USB49XX, AND USB70XX FAMILIES

Name	Explanation	Example
vid	16-bit value that uniquely identifies the Vendor ID of the USB2 device	"vid": "0x0424"
pid	16-bit value that the Vendor can assign that uniquely identifies particular Product ID for USB2 device	"pid": "0x2744"
did	16-bit device release number for USB2 device in BCD format	"did": "0x1234"
usbvcd	USB2 Specification Release Number in BCD format	"usbvcd": "2.10"
languageid	USB2 LANGUAGE ID	"languageid": "0x0409"
manufacturer	Manufacturer String of the USB2 HUB	"manufacturer": "Microchip"
product	Product String of the USB2 HUB	"product": "USB2734"
serial	Serial String of the USB2 HUB	"serial": "123456"
hceenable	Force enable 5th endpoint device Value: 1 – Force enable Value: 0 – Default behavior	"hceenable": "1"
enableport1 enableport2 enableport3 enableport4	Enable/Disable USB2.0 downstream ports Value: 1 – Enable Value: 0 – Disable	"enableport1": "1"
swapupstreamdpdm swapport1dpdm swapport2dpdm swapport3dpdm swapport4dpdm	Swaps dp and dm Value: 1 – Swaps D+ and D- Value: 0 – Default behavior	"swapupstreamdpdm": "1"
nrdeviceport4 nrdeviceport3 nrdeviceport2 nrdeviceport1	Configures Port as removable/non-removable Value: 1 – Port Non-removable Value: 0 – Port Removable	"nrdeviceport4": "1"
flexconnect	Enables FlexConnect. Swaps downstream port1 and upstream Value: 1 – Enable Flexconnect Value: 0 – Disable Flexconnect Flexconnect is not supported for USB70xx family hubs.	"flexconnect": "1"
hubpower	Configures hubpower as self-power or bus-power Value: 1 – Self-power Value: 0 – Bus-power	"hubpower": "1"

TABLE E-1: CONFIGURATION ITEMS COMMON FOR USB253X/USB(8)4604/USB3X13, USB57XX, USB49XX, AND USB70XX FAMILIES (CONTINUED)

Name	Explanation	Example
PHYBOOST PHYBOOSTdownstreamport1 PHYBOOSTdownstreamport2 PHYBOOSTdownstreamport3 PHYBOOSTdownstreamport4	Configures PHY Boost Value 0 – Nominal 17.78 mA 1 – 5% below nominal 2 – 10% above nominal 3 – 5% above nominal 4 – 20% above nominal 5 – 15% above nominal 6 – 30% above nominal 7 – 25% above nominal	"PHYBOOSTdownstreamport1": "1"
SQUELCH SQUELCHdownstreamport1 SQUELCHdownstreamport2 SQUELCHdownstreamport3 SQUELCHdownstreamport4	Configures Varisense of hub 0 – Nominal 100 mV 1 – 12.5 mV below nominal 2 – 25 mV below nominal 3 – 37.5 mV below nominal 4 – 50 mV below nominal 5 – 62.5 mV below nominal 6 – 25 mV above nominal 7 – 12.5 mV above nominal	"SQUELCHdownstreamport1": "2"
downstreamBCsettingport1 downstreamBCsettingport2 downstreamBCsettingport3 downstreamBCsettingport4	Configures Battery charging of port 0 – Standard USB port Maximum – 500 mA 1 – BC 1.2-Compliant Port Maximum – 1.5A 2 – Most Devices-supported mode Maximum – 2A	"downstreamBCsettingport1": "1"
upstreamHsRiseFall downstreamport1HsRiseFall downstreamport2HsRiseFall downstreamport3HsRiseFall downstreamport4HsRiseFall	Configures High Speed Rise/Fall time of port 0 – Default 1 – +18% 2 – -18% 3 – -12%	"downstreamport1HsRiseFall": "1"
hce_vid	16-bit value that uniquely identifies the Vendor ID of the HFC device	"hce_vid": "0x0424"
hce_pid	16-bit value that the Vendor can assign that uniquely identifies particular Product for HFC ID Device	"hce_pid": "0x2740"
hce_did	16-bit device release number for HFC device in BCD format	"hce_did": "0x1234"
hce_languageid	Language Id of HFC device	"hce_languageid": "0x0409"
hce_manufacturer	Manufacturer String of HFC device	"hce_manufacturer": "Microchip"
hce_product	Product String of HFC device	"hce_product": "Controller hub"
hce_serial	Serial String of HFC device	"hce_serial": "123456"

TABLE E-2: ADDITIONAL CONFIGURATION ITEMS FOR HSIC ENABLED PORTS OF USB3X13

Name	Explanation	Example
swapupstreamhsic	Swap data and strobe in HSIC enabled port 0 – Enable Swap 1 – Disable Swap	"swapupstreamhsic": "1"
DOI DOIdownstreamport1 DOIdownstreamport2 DOIdownstreamport3	HSIC Driver Output Impedance 0 – 40 ohm 1 – 50 ohm	"DOIdownstreamport1": "0"
SLEW SLEWdownstreamport1 SLEWdownstreamport2 SLEWdownstreamport3	Slew Tune for HSIC enabled ports 0 – Default 1 – +30%	"SLEWdownstreamport1": "0"

Supported Configuration Items

TABLE E-3: ADDITIONAL CONFIGURATION ITEMS FOR USB57XX

Name	Explanation	Example
usb3vid	16-bit value that uniquely identifies the Vendor ID of the USB3.1 Gen1 device	"usb3vid": "0x0424"
usb3pid	16-bit value that the Vendor can assign that uniquely identifies particular Product ID for USB3.1 Gen1 device	"usb3pid": "0x5744"
usb3did	16-bit device release number for USB3.1 Gen1 device in BCD format	"usb3did": "0x5678"
usb3languageid	USB3.1 Gen1 LANGUAGE ID	"usb3languageid": "0x0409"
usb3manufacturer	Manufacturer String of the USB3.1 Gen1 Hub	"usb3manufacturer": "Microchip"
usb3product	Product String of the USB3.1 Gen1 Hub	"usb3product": "USB5734"
usb3serial	Serial String of the USB3.1 Gen1 Hub	"usb3serial": "456789"
cdcenable	Communication Device class Value: 0 – Disable CDC Value: 1 – Enable CDC at interface 1 Value: 2 – Enable CDC at interface 0	"cdcenable": "1"
usb3nrdeviceport1 usb3nrdeviceport2 usb3nrdeviceport3 usb3nrdeviceport4	Configures Port as removable/non-removable for USB3.1 Gen1 Ports Value: 1 – Port Non-removable Value: 0 – Port Removable	usb3nrdeviceport": "1"
usb3enableport1 usb3enableport2 usb3enableport3 usb3enableport4	Enable/Disable 3.1 Gen1 downstream ports Value: 1 – Enable Value: 0 – Disable	"usb3enableport1": "1"

TABLE E-4: ADDITIONAL CONFIGURATION ITEMS FOR USB58XX/USB59XX

Name	Explanation	Example
usb3vid	16-bit value that uniquely identifies the Vendor ID of the USB3.1 Gen1 device	"usb3vid": "0x0424"
usb3pid	16-bit value that the Vendor can assign that uniquely identifies particular Product ID for USB3.1 Gen1 device	"usb3pid": "0x5744"
usb3did	16-bit device release number for USB3.1 Gen1 device in BCD format	"usb3did": "0x5678"
usb3languageid	USB3.1 Gen1 LANGUAGE ID	"usb3languageid": "0x0409"
usb3manufacturer	Manufacturer String of the USB3.1 Gen1 Hub	"usb3manufacturer": "Microchip"
usb3product	Product String of the USB3.1 Gen1 Hub	"usb3product": "USB5734"
usb3serial	Serial String of the USB3.1 Gen1 Hub	"usb3serial": "456789"
cdcenable	Communication Device class Value: 0 – Disable CDC Value: 1 – Enable CDC at interface 1 Value: 2 – Enable CDC at interface 0	"cdcenable": "1"
usb3nrdeviceport1 usb3nrdeviceport2 usb3nrdeviceport3 usb3nrdeviceport4 usb3nrdeviceport5 usb3nrdeviceport6 usb3nrdeviceport7	Configures Port as removable/non-removable for USB3.1 Gen 1 Ports Value: 1 – Port Non-removable Value: 0 – Port Removable	"usb3nrdeviceport": "1"

TABLE E-4: ADDITIONAL CONFIGURATION ITEMS FOR USB58XX/USB59XX (CONTINUED)

Name	Explanation	Example
usb3enableport1 usb3enableport2 usb3enableport3 usb3enableport4 usb3enableport5 usb3enableport6 usb3enableport7	Enable/Disable 3.1 Gen 1 downstream ports Value: 1 – Enable Value: 0 – Disable	"usb3enableport1": "1"
downstreamBCsettingport5 downstreamBCsettingport6 downstreamBCsettingport7	Configures Battery charging of port 0 – Standard USB port Maximum – 500 mA 1 – BC 1.2-Compliant Port Maximum – 1.5A 2 – Most Devices supported mode Maximum – 2A	"downstreamBCsettingport5": "1"
PHYBOOSTdownstreamport5 PHYBOOSTdownstreamport6 PHYBOOSTdownstreamport7	Configures PHY Boost Value 0 – Nominal 17.78 mA 1 – 5% below nominal 2 – 10% above nominal 3 – 5% above nominal 4 – 20% above nominal 5 – 15% above nominal 6 – 30% above nominal 7 – 25% above nominal	"PHYBOOSTdownstreamport5": "1"
SQUELCHdownstreamport5 SQUELCHdownstreamport6 SQUELCHdownstreamport7	Configures Varisense of hub 0 – Nominal 100 mV 1 – 12.5 mV below nominal 2 – 25 mV below nominal 3 – 37.5 mV below nominal 4 – 50 mV below nominal 5 – 62.5 mV below nominal 6 – 25 mV above nominal 7 – 12.5 mV above nominal	"SQUELCHdownstreamport5": "2"
Downstreamport5HsRiseFall Downstreamport6HsRiseFall Downstreamport7HsRiseFall	Configures High-Speed Rise/Fall time of port 0 – Default 1 – +18% 2 – -18% 3 – 12%	"downstreamport5HsRisefall": "1"
Port5splitting Port6splitting	Configures the port 5 splitting. 1 – Enable 0 – Disable	"port5splitting": "1" "port6splitting": "1"

TABLE E-5: CONFIGURATION ITEMS COMMON FOR USB49XX, USB4715 FAMILY

Name	Explanation	Example
Primary_vid	16-bit value that uniquely identifies the Vendor ID of the USB2 device	"Primary_vid": "0x0424"
Primary_pid	16-bit value that the Vendor can assign that uniquely identifies particular Product ID for USB2 device	"Primary_pid": "0x2744"
Primary_did	16-bit device release number for USB2 device in BCD format	"Primary_did": "0x1234"
Primary_usbvcd	USB2 Specification Release Number in BCD format	"Primary_usbvcd": "2.10"
Primary_languageid	USB2 LANGUAGE ID	"Primary_languageid": "0x0409"
Primary_manufacturer	Manufacturer String of the USB2 Hub	"Primary_manufacturer": "Microchip"

Supported Configuration Items

TABLE E-5: CONFIGURATION ITEMS COMMON FOR USB49XX, USB4715 FAMILY (CONTINUED)

Name	Explanation	Example
Primary_product	Product String of the USB2 Hub	"Primary_product": "USB2734"
Primary_serial	Serial String of the USB2 Hub	"Primary_serial": "123456"
Hceenable	Force enable 5 th endpoint device Value: 1 – Force enable Value: 0 – Default behaviour	"hceenable": "1"
enableport1 enableport2 enableport3 enableport4	Enable/Disable USB2.0 downstream ports Value: 1 – Enable Value: 0 – Disable	"enableport1": "1"
Swapupstreamdpdm swapport1dpdm swapport2dpdm swapport3dpdm swapport4dpdm	Swaps dp and dm Value: 1 – Swaps D+ and D– Value: 0 – Default behaviour	"swapupstreamdpdm": "1"
nrdeviceport4 nrdeviceport3 nrdeviceport2 nrdeviceport1	Configures Port as removable/non-removable Value: 1 – Port Non-removable Value: 0 – Port Removable	"nrdeviceport4": "1"
Hubpower	Configures hubpower as self-power or bus-power Value: 1 – Self-power Value: 0 – Bus-power	"hubpower": "1"
PHYBOOST PHYBOOSTdownstreamport1 PHYBOOSTdownstreamport2 PHYBOOSTdownstreamport3 PHYBOOSTdownstreamport4	Configures PHY Boost Value 0 – Nominal 17.78 mA 1 – 5% below nominal 2 – 10% above nominal 3 – 5% above nominal 4 – 20% above nominal 5 – 15% above nominal 6 – 30% above nominal 7 – 25% above nominal	"PHYBOOSTdownstreamport1": "1"
SQUELCH SQUELCHdownstreamport1 SQUELCHdownstreamport2 SQUELCHdownstreamport3 SQUELCHdownstreamport4	Configures Varisense of hub 0 – Nominal 100 mV 1 – 12.5 mV below nominal 2 – 25 mV below nominal 3 – 37.5 mV below nominal 4 – 50 mV below nominal 5 – 62.5 mV below nominal 6 – 25 mV above nominal 7 – 12.5 mV above nominal	"SQUELCHdownstreamport1": "2"
downstreamBCsettingport1 downstreamBCsettingport2 downstreamBCsettingport3 downstreamBCsettingport4	Configures Battery charging of port 0 – Standard USB port Maximum – 500 mA 1 – BC 1.2-Compliant Port Maximum – 1.5A 2 – Most Devices supported mode Maximum – 2A	"downstreamBCsettingport1": "1"
upstreamHsRiseFall downstreamport1HsRiseFall downstreamport2HsRiseFall downstreamport3HsRiseFall downstreamport4HsRiseFall	Configures High Speed Rise/Fall time of port 0 – Default 1 – +18% 2 – –18% 3 – –12%	"downstreamport1HsRisefall": "1"
hce_vid	16-bit value that uniquely identifies the Vendor ID of the HFC device	"hce_vid": "0x0424"
hce_pid	16-bit value that the Vendor can assign that uniquely identifies particular Product for HFC ID Device	"hce_pid": "0x2740"

TABLE E-5: CONFIGURATION ITEMS COMMON FOR USB49XX, USB4715 FAMILY (CONTINUED)

Name	Explanation	Example
hce_did	16-bit device release number for HFC device in BCD format	"hce_did": "0x1234"
hce_languageid	Language Id of HFC device	"hce_languageid": "0x0409"
hce_manufacturer	Manufacturer String of HFC device	"hce_manufacturer": "Microchip"
hce_product	Product String of HFC device	"hce_product": "Controller hub"
hce_serial	Serial String of HFC device	"hce_serial": "123456"
i2senable	Configures the I2S interface 0 – Disable – I2S is completely disabled. 1 – Enable Audio IN – Audio IN is enabled. 2 – Enable Audio OUT – Audio OUT is enabled. 3 – Enable Audio IN and Audio OUT – Audio IN and Audio OUT are enabled.	"i2senable": "1"
i2senablehid	Configures the I2S hid interface 0 – Disable – HID is disabled 1 – Audio IN Control – HID interface controls only Audio IN 2 – Audio OUT Control – HID interface controls only Audio OUT 3 – Audio IN Control and Audio OUT Control- HID interface controls both Audio IN and Audio OUT	"i2senablehid": "2"

TABLE E-6: ADDITIONAL CONFIGURATION ITEMS FOR THE USB49XX FAMILY

Name	Explanation	Example
secondary_vid	16-bit value that uniquely identifies the Vendor ID of the USB2 device	"secondary_vid": "0x0424"
secondary_pid	16-bit value that the Vendor can assign that uniquely identifies particular Product ID for USB2 device	"secondary_pid": "0x2744"
secondary_did	16-bit device release number for USB2 device in BCD format	"secondary_did": "0x1234"
secondary_usbvcd	USB2 Specification Release Number in BCD format	"secondary_usbvcd": "2.10"
secondary_languageid	USB2 LANGUAGE ID	"secondary_languageid": "0x0409"
secondary_manufacturer	Manufacturer String of the USB2 Hub	"secondary_manufacturer": "Microchip"
secondary_product	Product String of the USB2 Hub	"secondary_product": "USB2734"
secondary_serial	Serial String of the USB2 Hub	"secondary_serial": "123456"
swapsecondaryportdpdm	Swaps DP and DM Value: 1 – Swaps D+ and D- Value: 0 – Default behaviour	"swapsecondaryportdpdm": "1"
PHYBOOSTdownstreamsecondaryport	Configures PHY Boost Value 0 – Nominal 17.78 mA 1 – 5% below nominal 2 – 10% above nominal 3 – 5% above nominal 4 – 20% above nominal 5 – 15% above nominal 6 – 30% above nominal 7 – 25% above nominal	"PHYBOOSTdownstreamsecondaryport": "1"

Supported Configuration Items

TABLE E-6: ADDITIONAL CONFIGURATION ITEMS FOR THE USB49XX FAMILY (CONTINUED)

Name	Explanation	Example
SQUELCHdownstreamsecondaryport	Configures Varisense of hub 0 – Nominal 100 mV 1 – 12.5 mV below nominal 2 – 25 mV below nominal 3 – 37.5 mV below nominal 4 – 50 mV below nominal 5 – 62.5 mV below nominal 6 – 25mV above nominal 7 – 12.5 mV above nominal	"SQUELCHdownstreamsecondaryport": "2"
secondaryportHsRiseFall	Configures High Speed Rise/Fall time of port 0 – Default 1 – +18% 2 – -18% 3 – -12%	"secondaryportHsRiseFall": "2"

TABLE E-7: ADDITIONAL CONFIGURATION ITEMS FOR THE USB4715

Name	Explanation	Example
flexconnect	Enables FlexConnect. Swaps downstream port1 and upstream Value: 1 – Enable FlexConnect Value: 0 – Disable FlexConnect	"flexconnect": "1"
flexport	FlexConnect occurred with the selected port and upstream Value: 1 – Port 1 Value: 2 – Port2 Value: 3 – Port3 Value: 3 – Port4	"flexport": "1"

TABLE E-8: ADDITIONAL CONFIGURATION ITEMS FOR USB70XX

Name	Explanation	Example
usb3vid	16-bit value that uniquely identifies the Vendor ID of the USB3.1 Gen1 device	"usb3vid": "0x0424"
usb3pid	16-bit value that the Vendor can assign that uniquely identifies Product ID for USB3.1 Gen1 device	"usb3pid": "0x7002"
usb3did	16-bit device release number for USB3.1 Gen1 device in BCD format	"usb3did": "0x5678"
usb3languageid	USB3.1 Gen1 LANGUAGE ID	"usb3languageid": "0x0409"
usb3manufacturer	Manufacturer String of the USB3.1 Gen1 Hub	"usb3manufacturer": "Microchip"
usb3product	Product String of the USB3.1 Gen1 Hub	"usb3product": "USB5734"
usb3serial	Serial String of the USB3.1 Gen1 Hub	"usb3serial": "456789"
usb3nrdeviceport1 usb3nrdeviceport2 usb3nrdeviceport3 usb3nrdeviceport4 usb3nrdeviceport5	Configures Port as removable/non-removable for USB3.1 Gen 1 Ports Value: 1 – Port Non-removable Value: 0 – Port Removable	"usb3nrdeviceport": "1"

TABLE E-8: ADDITIONAL CONFIGURATION ITEMS FOR USB70XX (CONTINUED)

usb3enableport1 usb3enableport2 usb3enableport3 usb3enableport4 usb3enableport5	Enable/Disable 3.1 Gen 1 Downstream ports Value: 1 – Enable Value: 0 – Disable	"usb3enableport1": "1"
downstreamBCsettingport5 downstreamBCsettingport6	Configures Battery charging of port 0 – Standard USB port Maximum – 500mA 1 – BC 1.2-Compliant Port Maximum – 1.5A 2 – Most Devices supported mode Maximum – 2A	"downstreamBCsettingport5": "1"
PHYBOOSTdownstreamport5 PHYBOOSTdownstreamport6	Configures PHY Boost Value 0 – Nominal 17.78 mA 1 – 5% below nominal 2 – 10% above nominal 3 – 5% above nominal 4 – 20% above nominal 5 – 15% above nominal 6 – 30% above nominal 7 – 25% above nominal	"PHYBOOSTdownstreamport5": "1"
SQUELCHdownstreamport5 SQUELCHdownstreamport6	Configures Varisense of hub 0 – Nominal 100 mV 1 – 12.5 mV below nominal 2 – 25 mV below nominal 3 – 37.5 mV below nominal 4 – 50 mV below nominal 5 – 62.5 mV below nominal 6 – 25 mV above nominal 7 – 12.5 mV above nominal	"SQUELCHdownstreamport5": "2"
downstreamport5HsRiseFall downstreamport6HsRiseFall	Configures High Speed Rise/Fall time of port 0 – Default 1- +18% 2- -18% 3- -12%	"downstreamport5HsRiseFall": "1"
port3splitting port4splitting port5splitting	Configures the port splitting. 1 – Enable 0 – Disable	"port5splitting": "1"
power_ontime	Configures the power on time	"power_ontime": "33"
minihost_vidpidrule	Configures the MiniHost VID PID rule	"minihost_vidpidrule": "1+1+0424+4504+0+1"
minihost_vidpidmaskrule	Configures the Minihost VID PID Mask rule	"minihost_vidpidmaskrule": "1+1+0424+4504+ffffffff+0+2"
minihost_deviceclassrule	Configures the Minihost Device Class rule	"minihost_deviceclassrule": "1+1+00+0+3"

NOTES:

Worldwide Sales and Service

AMERICAS

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

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

Austin, TX
Tel: 512-257-3370

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

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

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

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

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

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

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

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

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

ASIA/PACIFIC

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

China - Xiamen
Tel: 86-592-2388138

China - Zhuhai
Tel: 86-756-3210040

ASIA/PACIFIC

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

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

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

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

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

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

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

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

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

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

Singapore
Tel: 65-6334-8870

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

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

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

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

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

EUROPE

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

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

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

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

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

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

Germany - Karlsruhe
Tel: 49-721-625370

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

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

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

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

Italy - Padova
Tel: 39-049-7625286

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

Norway - Trondheim
Tel: 47-7288-4388

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