# MPLAB® Code Configurator Setup for Software Development on PIC32CX_BZ2/WBZ45x

Pre-requisite: Clone the EA71C53A repo available at https://github.com/MicrochipTech/EA71C53A

1. Install MPLAB x IDE, instructions available in EA71C53A\MPLAB X IDE folder
2. Install XC32 Compiler, instructions available in EA71C53A\Compiler
3. Install Device Family Part Pack, located in EA71C53A\MPLAB X IDE

   Device Family Packs are device description files (.PIC files for PIC® devices, .ATDF files for AVR® and SAM devices), which contain SFR names, memory regions, programming information. Device-dependent source code files (i.e., peripheral header files) are being moved to DFPs. XC8 (AVR target) and XC32 (SAM target) are implemented today. Libraries will be part of the DFP on XC8 (AVR, CSTARTUP) and XC32 – XC16 will store the libraries in the compiler directory.
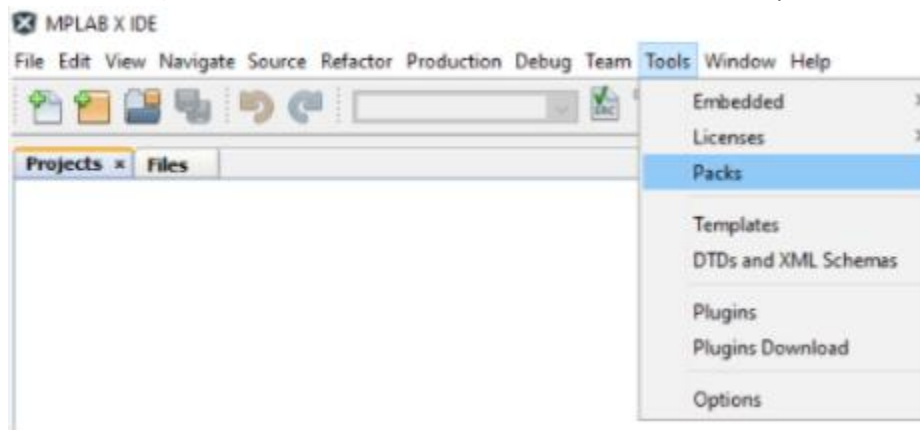
   3.1 Open the MPLAB X IDE and select Tools > Packs
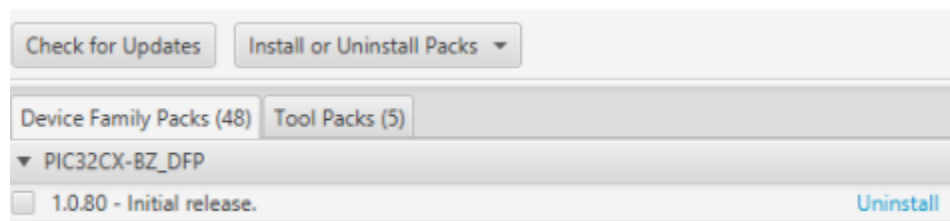
   3.2 Click "Install from Local Source"
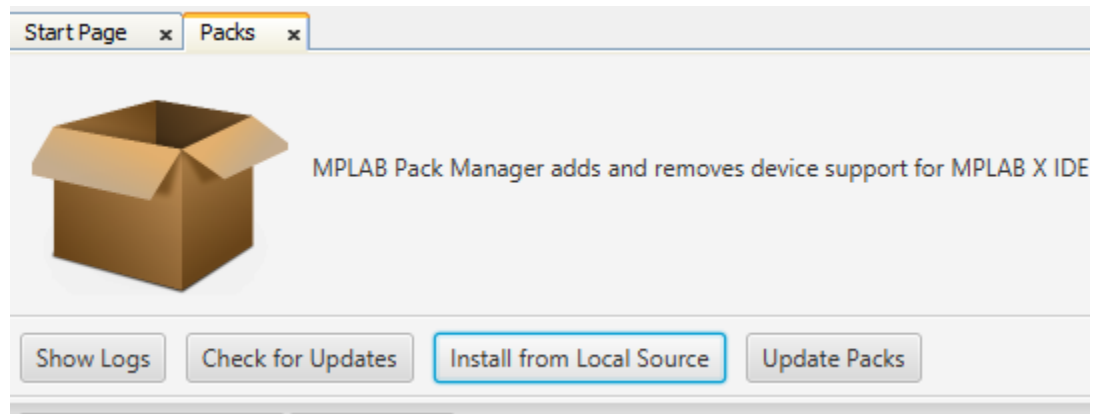
3.3 Locate (EA71C53A\MPLAB X IDE) and **select** the device family pack **Microchip.PIC32CX-BZ_DFP-1.0.xx** in **MPLAB X IDE** directory
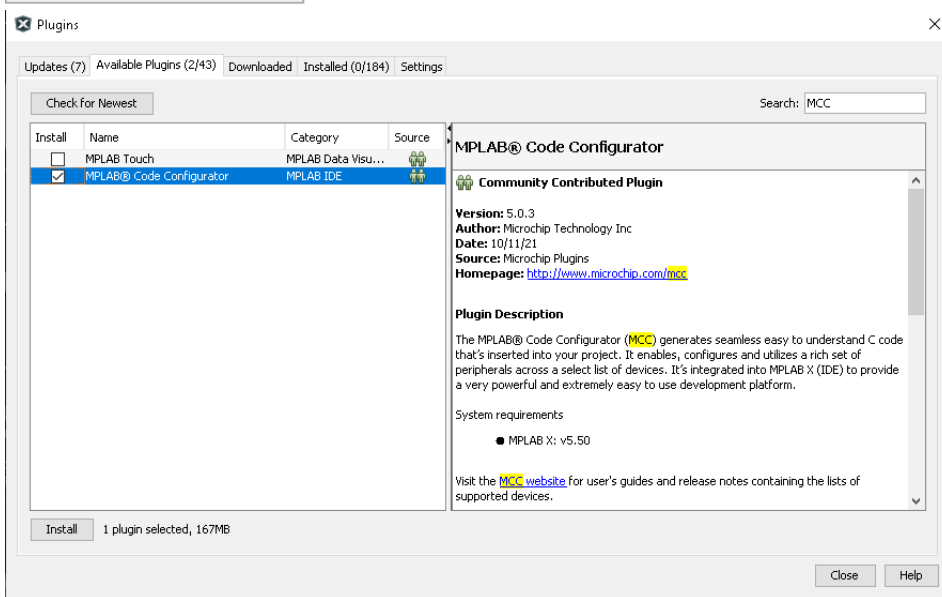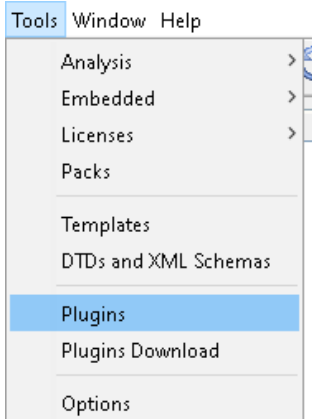

3.4 Verify the installation of device family pack by searching in the window – search for "bz" keyword

Restart MPLAB X IDE

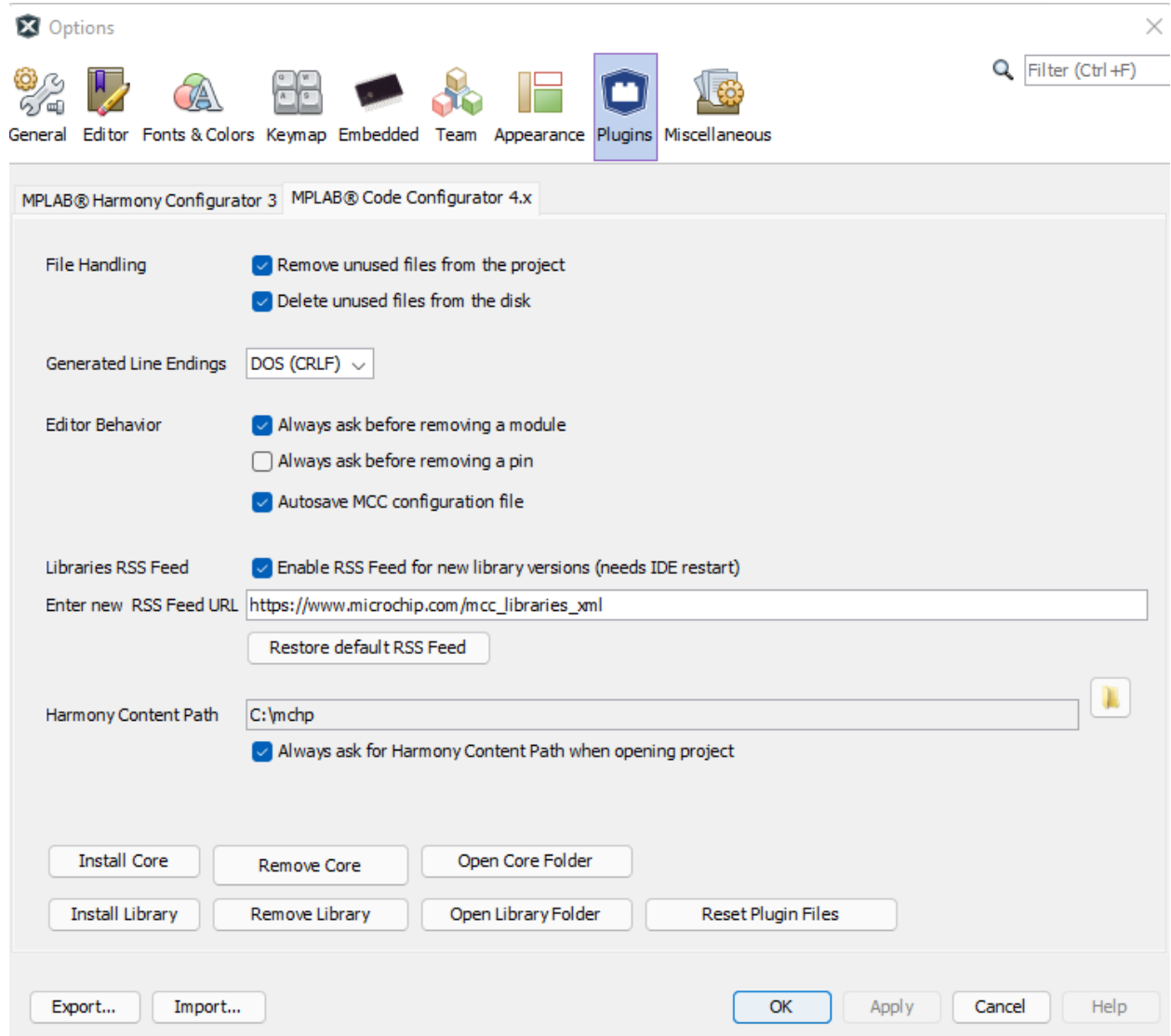4. Install MCC plugin in the IDE, restart the IDE after installation

5. Configure the Plugin Options
Several aspects of the operation of the MCC can be managed by using the "Options" panel (see figure below), which can be invoked by clicking Tools → Options → Plugins → MPLAB Code Configurator in the menu bar of the MPLAB X IDE.
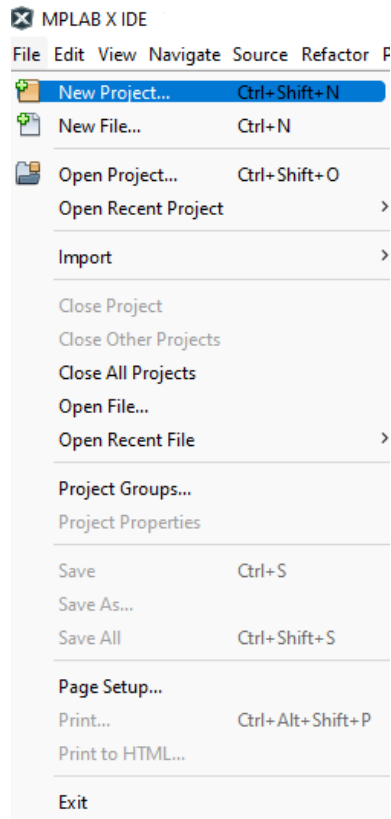
6. Clone the Harmony repositories (required content for SW Development) using MCC Content Manager Wizard
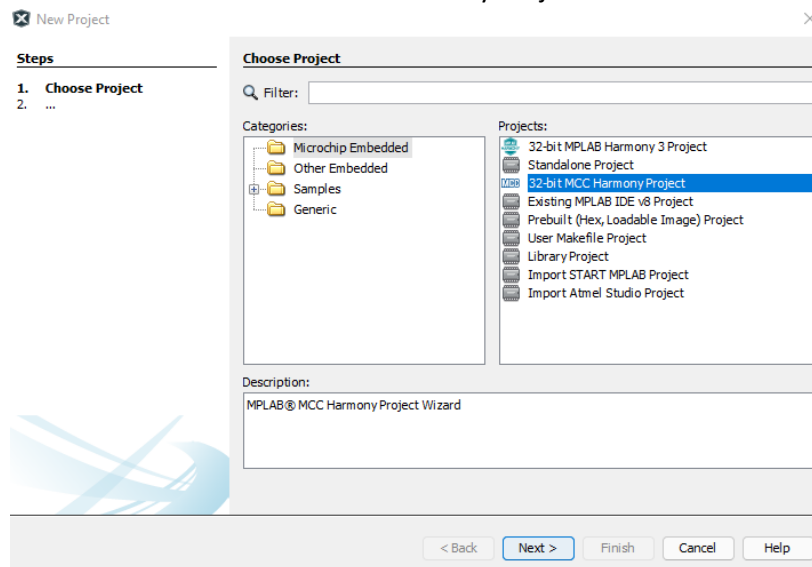
    6.1 Create a new "MCC Harmony" project (In order to clone the Harmony repositories user needs to create an empty project and clone the required repositories)

        6.1.1   Select "New Project"

MPLAB X IDE

| File | Edit | View | Navigate | Source | Refactor | F |

	New Project...	Ctrl+Shift+N
	New File...	Ctrl+N
	Open Project...	Ctrl+Shift+O
	Open Recent Project	>
	Import	>
	Close Project	
	Close Other Projects	
	Close All Projects	
	Open File...	
	Open Recent File	>
	Project Groups...	
	Project Properties	
	Save	Ctrl+S
	Save As...	
	Save All	Ctrl+Shift+S
	Page Setup...	
	Print...	Ctrl+Alt+Shift+P
	Print to HTML...	
	Exit	

6.1.2   Select "32-bit MCC Harmony Project"

New Project

Steps

1. **Choose Project**
2. ...

Choose Project

Filter:

Categories:
- Microchip Embedded
- Other Embedded
- Samples
- Generic

Projects:
- 32-bit MPLAB Harmony 3 Project
- Standalone Project
- 32-bit MCC Harmony Project
- Existing MPLAB IDE v8 Project
- Prebuilt (Hex, Loadable Image) Project
- User Makefile Project
- Library Project
- Import START MPLAB Project
- Import Atmel Studio Project

Description:

MPLAB® MCC Harmony Project Wizard

< Back   Next >   Finish   Cancel   Help

6.1.3   Framework Selection

New Project

Steps

1. Choose Project
2. **Framework Selection**
3. Project Settings
4. Configuration Settings

Manage Framework

Framework Path: C:\mchp

☑ Convert to Relative Path for Configuration

< Back   Next >   Finish   Cancel   Help

6.1.4   Project Settings

New Project

Steps

1. Choose Project
2. Framework Selection
3. **Project Settings**
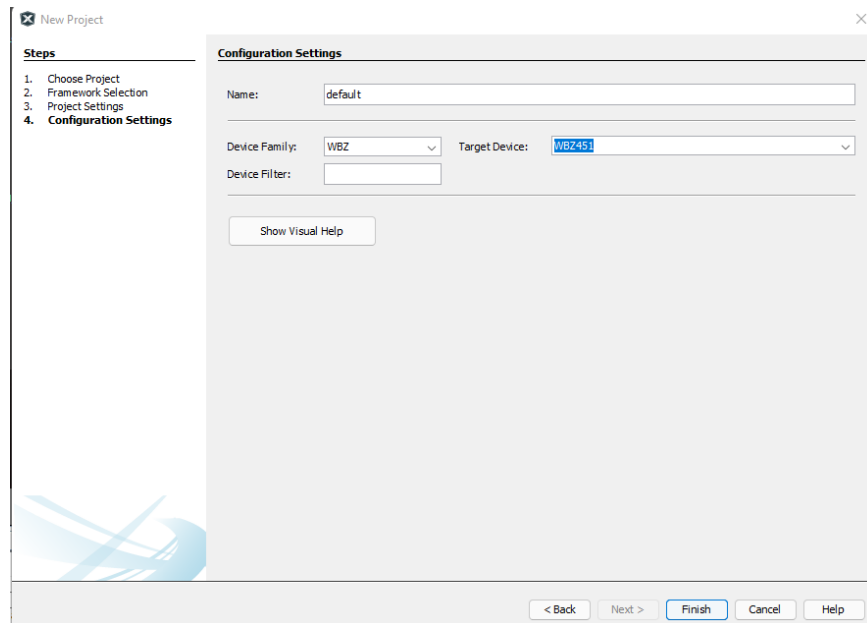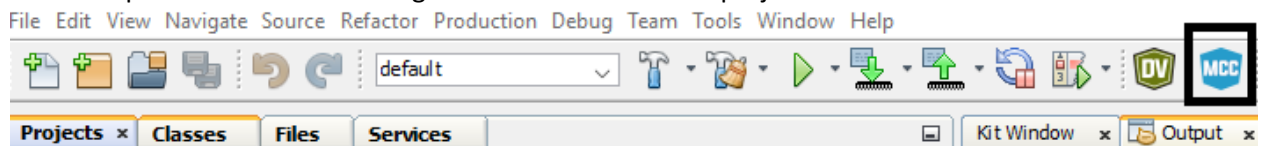4. Configuration Settings

Name and Location

Location: C:\Users\user\HarmonyProjects\MyProject_4

Folder: newProject

Name:

Path: C:\Users\user\HarmonyProjects\MyProject_4\firmware\newProject.X

Show Visual Help

< Back   Next >   Finish   Cancel   Help

6.1.5   Configuration Settings

6.2 Open "MPLAB Code Configurator" after creation of project

MCC content manager window will appear in IDE

6.3 Select "MPLAB Harmony" from the content manager window
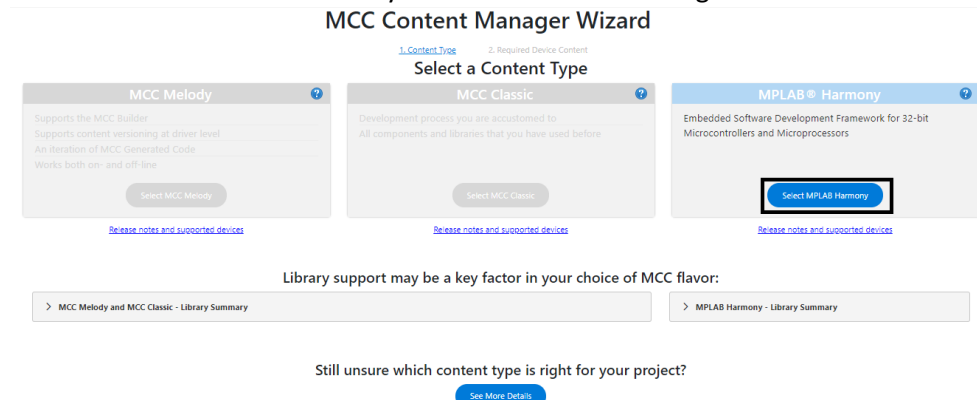
Select "Optional Content" as shown below

Optional Content

Select optional content to be made available in Device Resources for selection

Optional Content		

Component ↑↓	Version	Description
> ☐ Harmony Bootloader		
> ☐ Harmony Chip Support Package		
> ☐ Harmony Networking Stack and Solutions		
> ☐ Harmony USB solutions		
∨ ⊟ Harmony Core		
☑ bsp	3.10.0	
☑ core	3.10.0	
☑ zlib	1.2.11	
∨ ⊟ Harmony Cryptography solutions		
☑ crypto	3.7.5	
☐ crypto_apps_encrypt_decrypt	3.7.1	
☐ crypto_apps_large_hash	3.7.1	
☐ crypto_apps_speed_test	3.7.1	
> ☐ Harmony Aerospace solutions		
> ☐ Harmony Graphics (Aria) solutions		
> ☐ Harmony CryptoAuthLib solutions		
> ☐ arm Mbed OS		
> ☐ Harmony Capacitive Touch solutions		
> ☐ Harmony littlefs solutions		
> ☐ Harmony AWS solutions		
∨ ⊟ Harmony WolfSSL solutions		
☐ wolfMQTT	1.7.1	
☐ wolfssh	1.4.1	
☑ wolfssl	4.7.0	
> ☐ X2C for Harmony		
∨ ☑ CMSIS FreeRTOS		
☑ CMSIS-FreeRTOS	10.3.1	

Ensure all components are selected as displayed below (ignore versions for now). Version selection will be done at a later stage

MCC Content Manager Wizard

1. Content Type   2. Required Device Content   Finish

Required Content

Some required content must be downloaded. The following content will be downloaded when you click on "Finish".
To change content versions later, access the Content Manager from Device Resources.

Component ↑↓	Version ↑↓
csp	3.10.0
mhc	3.8.2
quick_docs	1.4.0
dev_packs	3.10.0
bsp	3.10.0
core	3.10.0
CMSIS-FreeRTOS	10.3.1
crypto	3.7.5
zlib	1.2.11
wolfssl	4.7.0

Select "Finish" The downloading of selected components from harmony repositories will take some minutes.

If all the selected components are cloned successfully, MCC logs in IDE will display this information

If in the process a failure to download a particular component appears, try redownloading again only the component that failed to download.

For example, if user received a prompt from MCC saying "dev_packs" was not download, close the MCC reopen MCC again and start from step 3, with the "dev_packs" as the only missing component for downloading and select "Finish"

MCC Content Manager Wizard

1. Content Type   2. Required Device Content   ⟳ Finish

Required Content

Some required content must be downloaded. The following content will be downloaded when you click on
To change content versions later, access the Content Manager from Device Resources.

Component ↑↓	Version ↑↓
dev_packs	3.10.0

Optional Content

Select optional content to be made available in Device Resources for selection

7. Select/Confirm the Harmony Framework Path

Note: Framework Path selection window appears beneath the Initializing Project pop-up, users should Choose Path as "C:\mchp"

Project Graph Window will appear after choosing the framework path and successful initialization of project

8. Change content versions as mentioned below in the table using the content manager from Device Resources and select "Apply"

Package	Version	Location
MPLAB X IDE XC32	5.50 3.01	IDE folder Compiler folder
MCC Plugin	5.0.3	MPLAB X IDE > Tools > Plugins

csp	3.10.0	MPLAB® Code Configurator Content Manager
core	3.10.0	MPLAB® Code Configurator Content Manager
mhc	3.8.0	MPLAB® Code Configurator Content Manager
dev_packs	3.10.0	MPLAB® Code Configurator Content Manager
bsp	3.10.0	MPLAB® Code Configurator Content Manager
CMSIS-FreeRTOS	10.3.1	MPLAB® Code Configurator Content Manager
crypto	3.7.2	MPLAB® Code Configurator Content Manager
wolfssl	4.7.0	MPLAB® Code Configurator Content Manager
wireless	211211	EA71C53A\H3
PIC32CX-BZ_DFP	1.0.80	EA71C53A \MPLAB X IDE

Wireless_system_pic32cxbz_wbz		EA71C53A \H3\

9. After successfully checking out the right version of content/components, Harmony Framework needs to be selected again and Project will be reinitialized with the new versions of content
    -   Follow instructions mentioned in step 7 -
10. Copy the "clk_pic32cx_bz" folder located in "EA71C53A\" folder to "C:\mchp\csp\peripheral" folder, A prompt will appear warning destination has same file names - Select "Replace the file in the destination".
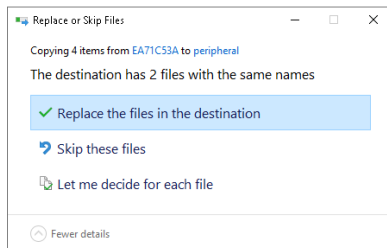
11. Copy the "wireless__" folders (3) located in "EA71C53A\H3" to "C:\mchp"
    Here is an example of how the folder should look like after copying the "wireless__" content
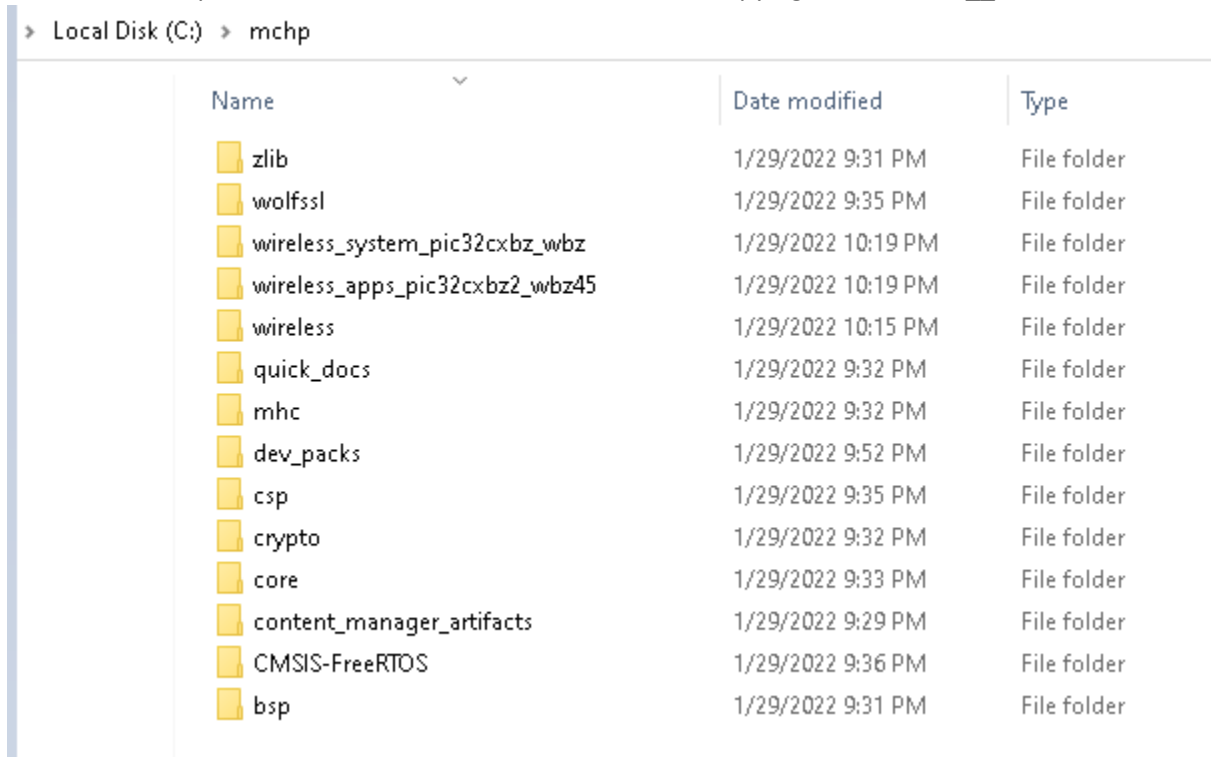
Next Steps

12. Open existing application examples/ develop a new application – visit -
mchp\wireless_apps_pic32cxbz2_wbz45\apps\

13. How to open, build and program an existing application example
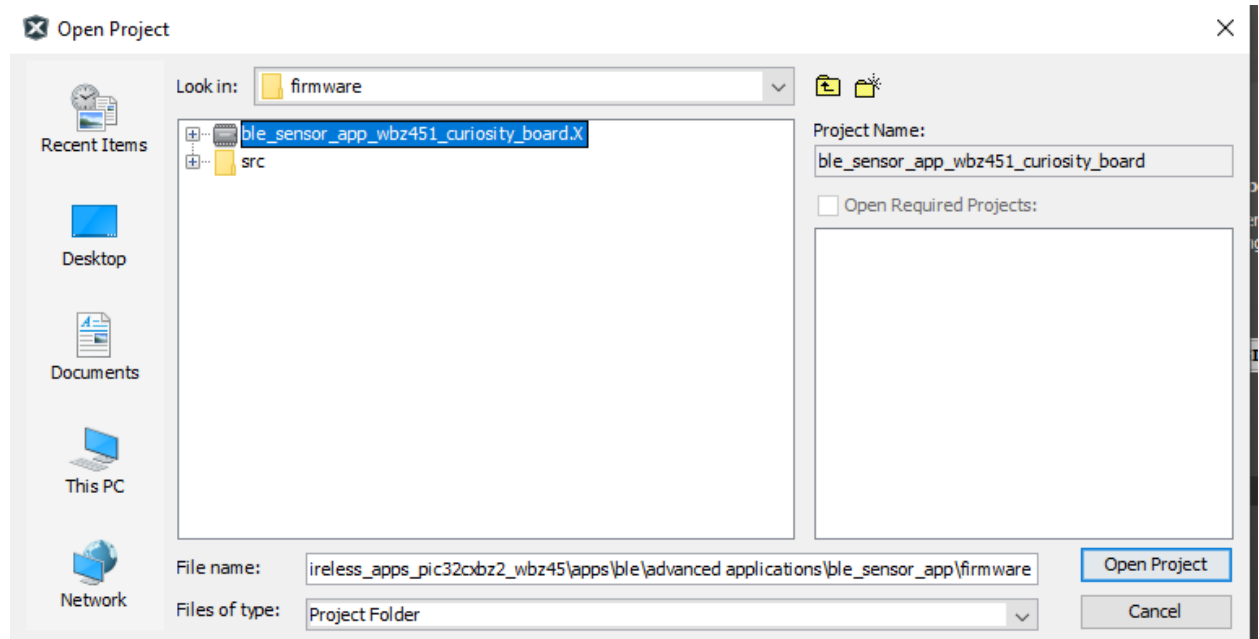**Pre-requisites:** Complete steps 1-11
13.1   Connect Curiosity Board to the PC using usb cable
13.2   Open MPLAB IDE
13.3   Select **File > Open Project**
13.4   Select the project from C:\mchp\wireless_apps_pic32cxbz2_wbz45\apps\ble\advanced
applications\ble_sensor_app\firmware

Information related to the workings of the application example are available in readme.md file
available in the ble_sensor_app folder or Getting Started html
13.5   Open Project Properties
      13.5.1   Select WBZ451 Curiosity Board as hardware tool for programming
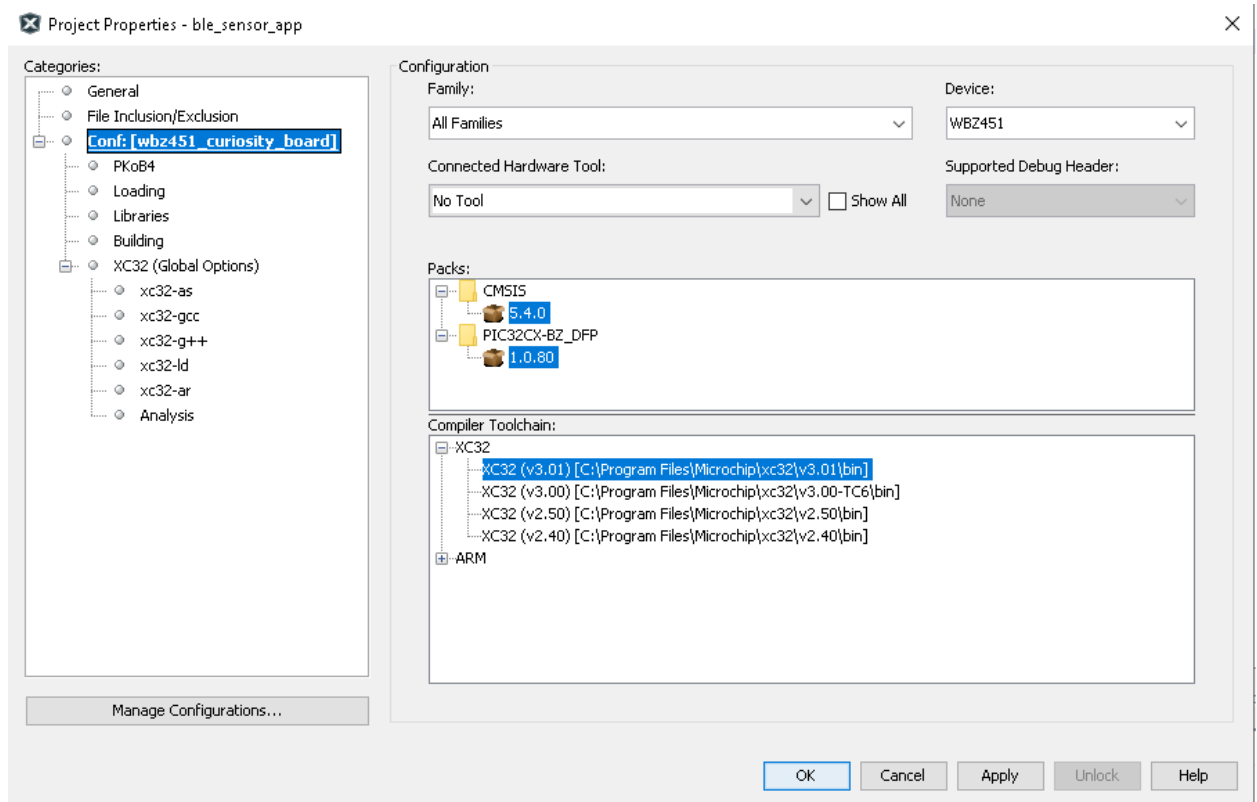   13.5.2   Ensure DFP v1.0.xx is selected and CMSIS v5.4.0
Note: DFP version should match the version mentioned in table 1
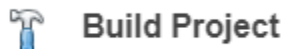   13.5.3   select XC32 v3.01 compiler (in case user has several versions of XC32
            compilers installed)
Note: Compiler version should match the version mentioned in table 1

13.6 Select option **Build Project** in IDE to compile the application example

13.7 Plug the Curiosity Development board to PC using usb cable

13.8 Select option **Run Project** in IDE to program the target – the onboard debugger will program the example application

Note: A smartphone App might be needed to explore the full feature set of Application examples, users can refer to readme.md (markdown reader recommended) available in respective Application Example folder or Getting Started html points to the instructions of the Application example