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

Pre-requisite: Clone the EA71C53A repo available @ **https://github.com/MicrochipTech/EA71C53A.git**

1. Versions of different components and tool chain required for test this package

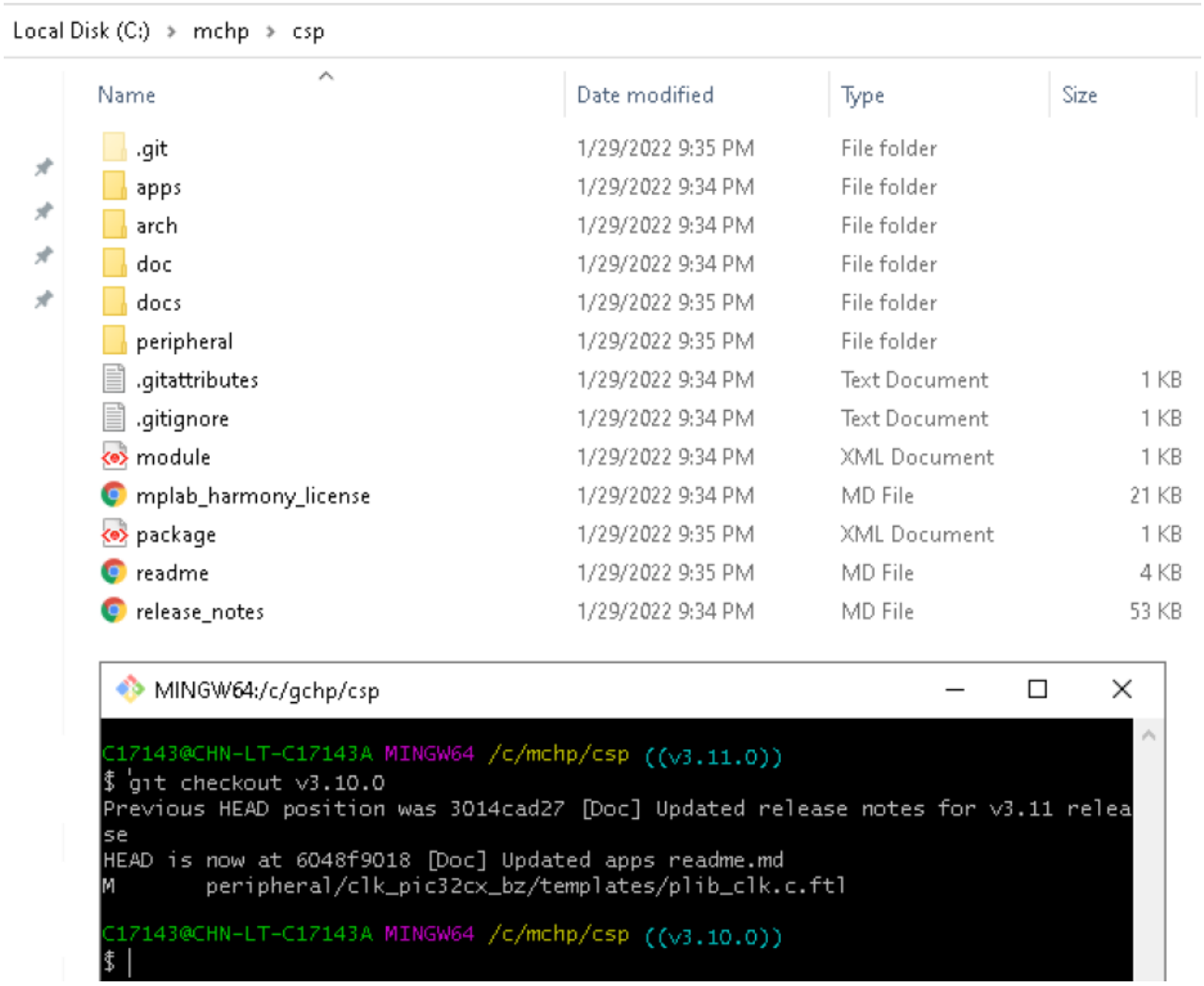
IDE, Compiler, MCC plugin	Version	Location
MPLAB X IDE XC32	6.00 4.10	IDE Compiler
MCC Plugin	5.1.4	MPLAB X IDE > Tools > Plugins

Harmony components to be cloned with MCC Plugin	Version	Location
csp	3.12.0	MPLAB® Code Configurator Content Manager
core	3.10.0	MPLAB® Code Configurator Content Manager
mhc	3.8.5	MPLAB® Code Configurator Content Manager

dev_packs	3.12.0	MPLAB® Code Configurator Content Manager
bsp	3.12.0	MPLAB® Code Configurator Content Manager
CMSIS-FreeRTOS	10.3.1	MPLAB® Code Configurator Content Manager
crypto	3.7.6	MPLAB® Code Configurator Content Manager
wolfssl	4.7.0	MPLAB® Code Configurator Content Manager
PIC32CX-BZ_DFP	1.0.80	EA71C53A \MPLAB X IDE
wireless	222105	EA71C53A\H3
wireless_system_pic32cxbz_wbz	222105	EA71C53A \H3
wireless_apps_pic32cxbz2_wbz45	222105	EA71C53A \H3

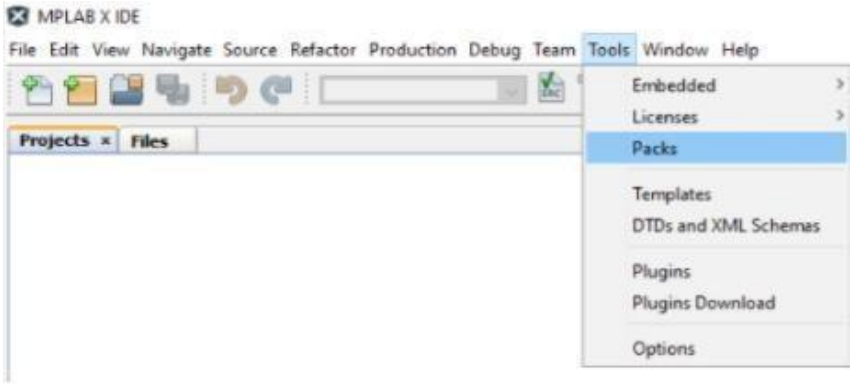
Note: The component versions mentioned above are the versions with which this entire package was tested on by Microchip.

Checking out a particular version of Harmony Component



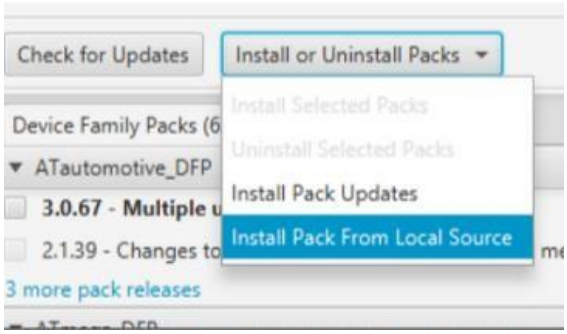
2. Install MPLAB x IDE (**No Apple M1 chip support**, user could use traditional intel based MAC's)
3. Install XC32 Compiler
4. Install Device Family Part Pack, located in EA71C53A\MPLAB X IDE

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.

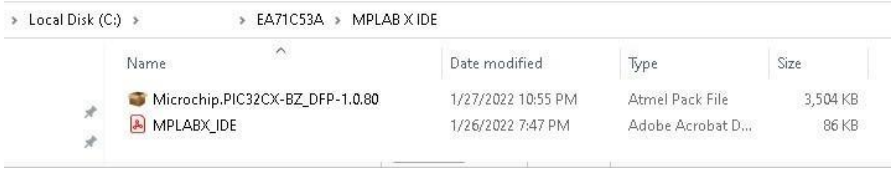


4.1 Open the MPLAB X IDE and select Tools > Packs

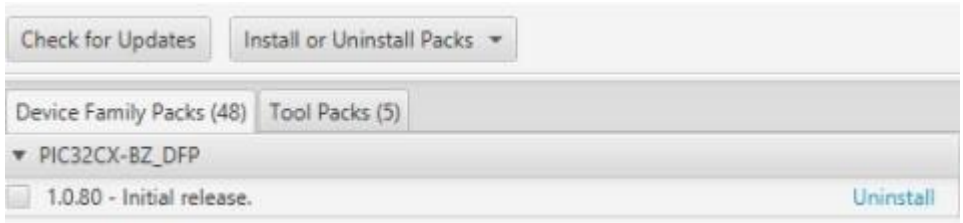
4.2 Click “Install from Local Source”



4.3 **Locate (EA71C53A\MPLAB X IDE)** and **select** the device family pack **Microchip.PIC32CXBZ_DFP-1.0.xx** in **MPLAB X IDE** directory

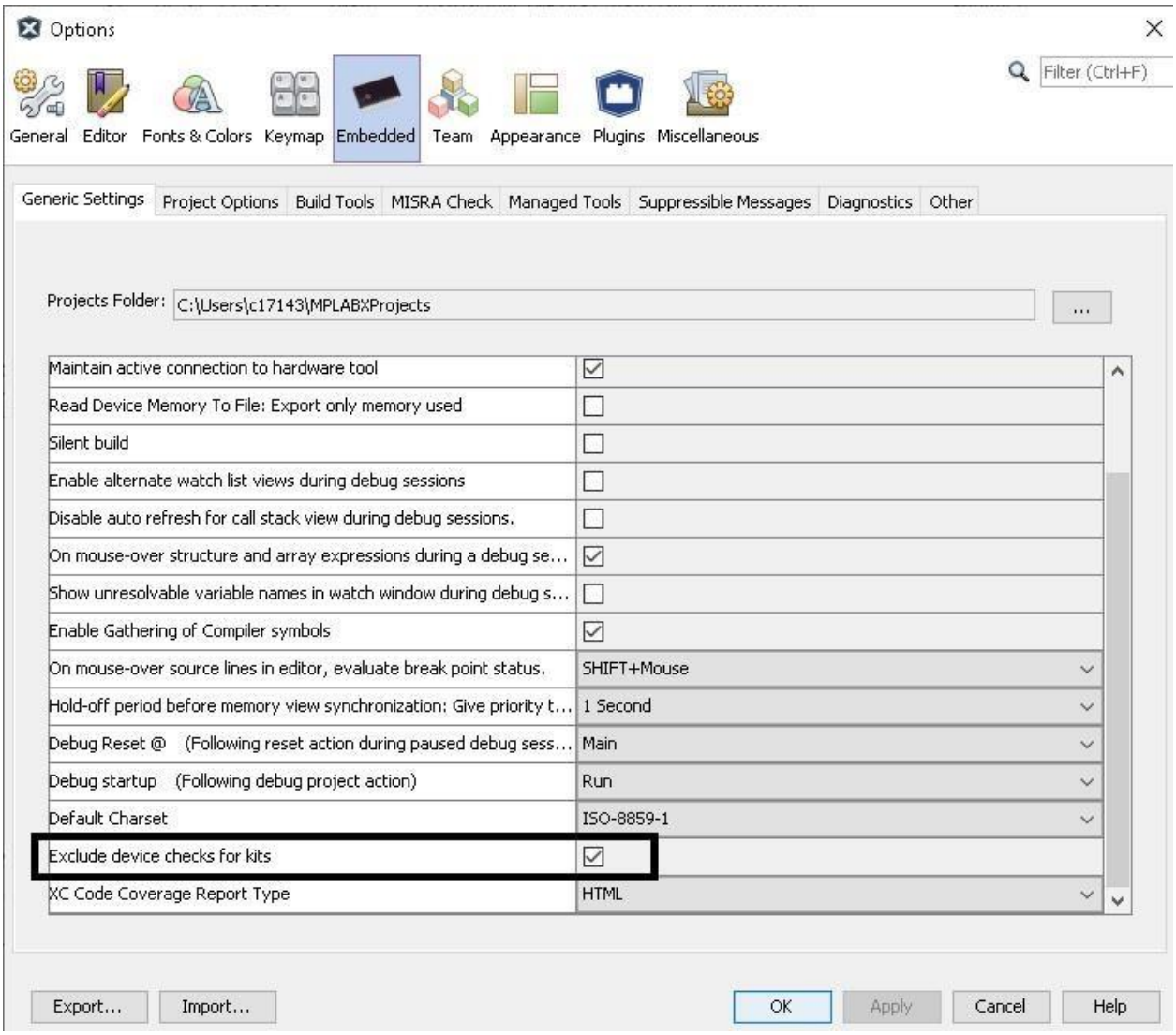


4.4 **Verify** the installation of device family pack by searching in the window – search for “bz” keyword



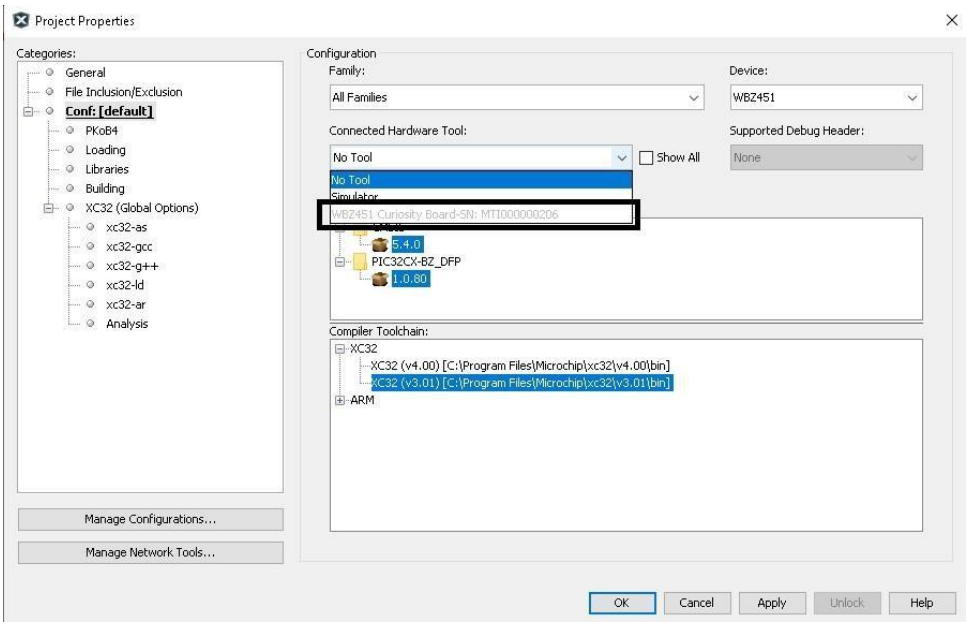
Restart MPLAB X IDE

5. Exclude Device checks for kits (Tools -> Options -> Embedded) , select “ok” once done!

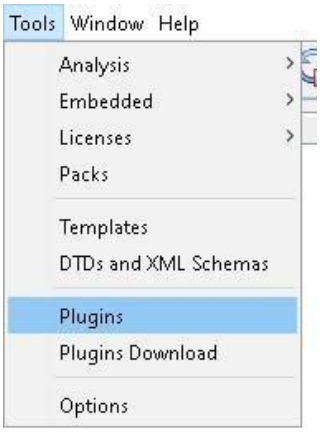


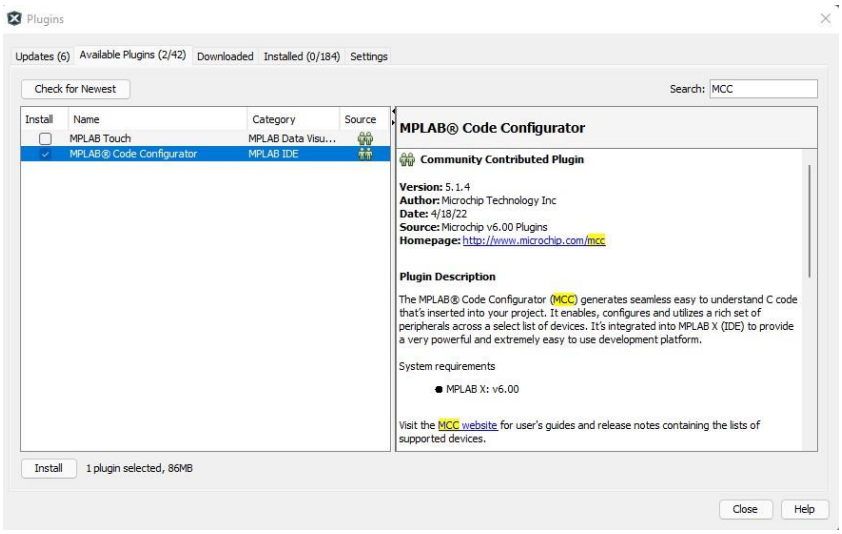
Note: Some preproduction boards might have “PIC32CX1012BZ25048” Soc Name programmed on PKOB4, if the “Exclude device checks for kits” is not enabled, user will not be able to choose the PKOB4 to program the WBZ451 module on the Curiosity Board

User cannot select the Kit (It will appear greyed out) in Programming options if “Exclude Device Checks” is not enabled



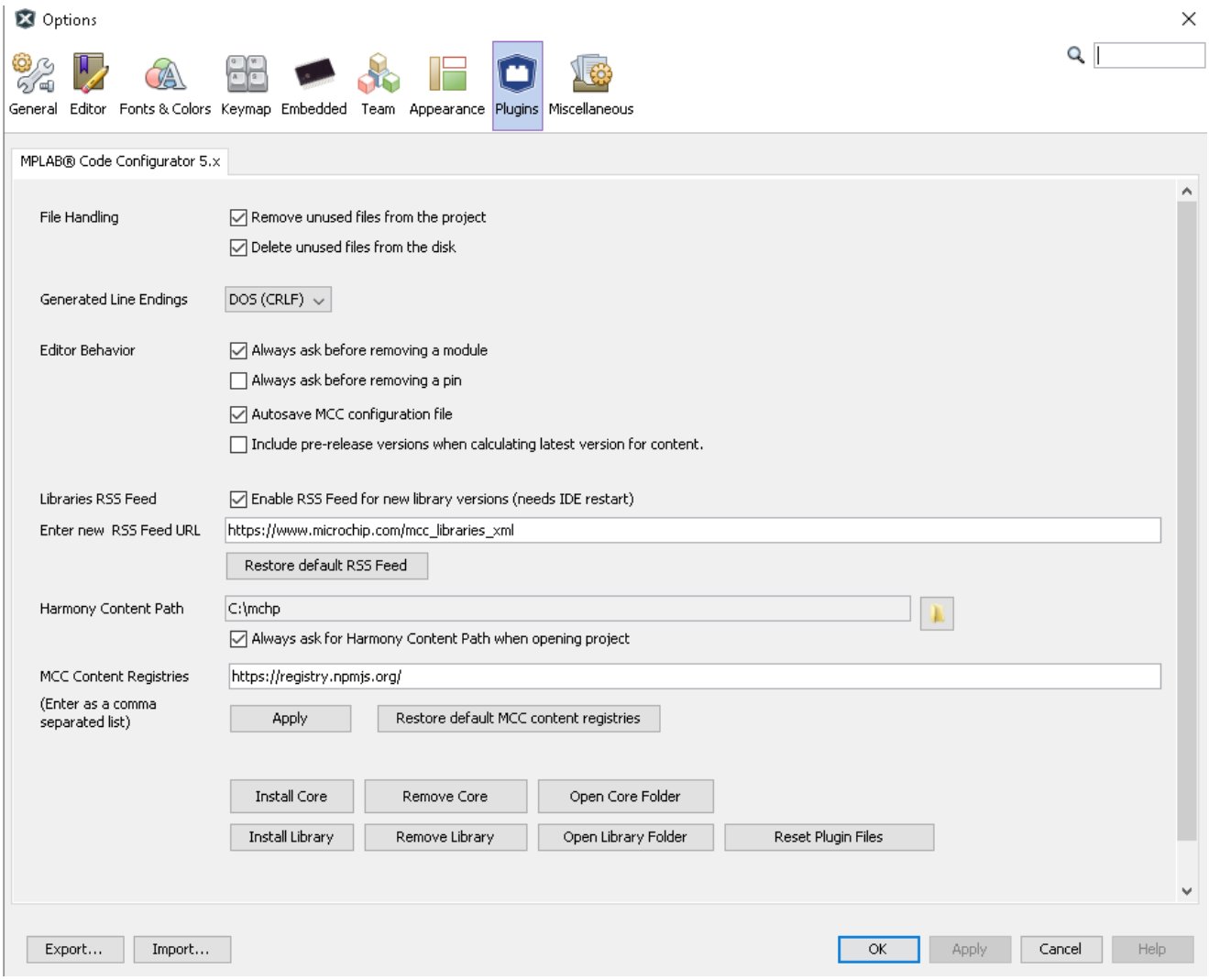
6. Delete the. mcc folder available in the following path – “C:/Users/user_name/”
   Note: “user_name” is the user profile
7. Install MCC plugin in the IDE, restart the IDE after installation (uninstall any older version of MCC, use MCC v5.1.4 or above)





8. 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.



Note: "Harmony Content Path" should be in root directory (maximum 1 folder deep and folder name cannot exceed 4 letters)

For example **-> C:\mchp** or **C:\test** are **acceptable** "Harmony Content Path"

C:\Microchip
C:\test\microchip **are not acceptable** path choices.

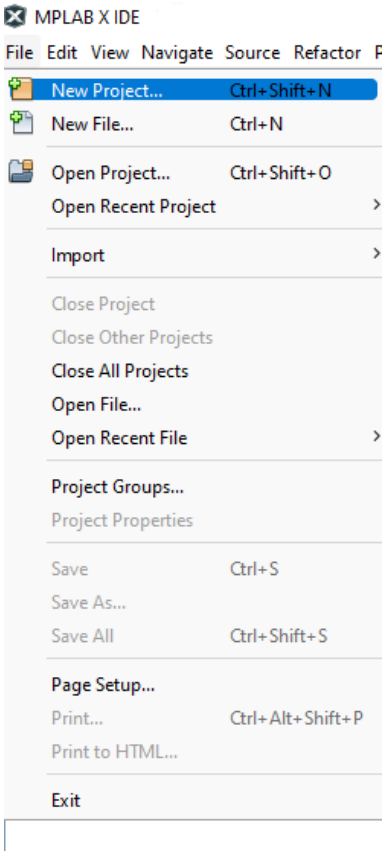
This note is added to avoid a known issue of Maximum File path, see here for more information

9. Clone the Harmony repositories (required content for SW Development) using MCC Content

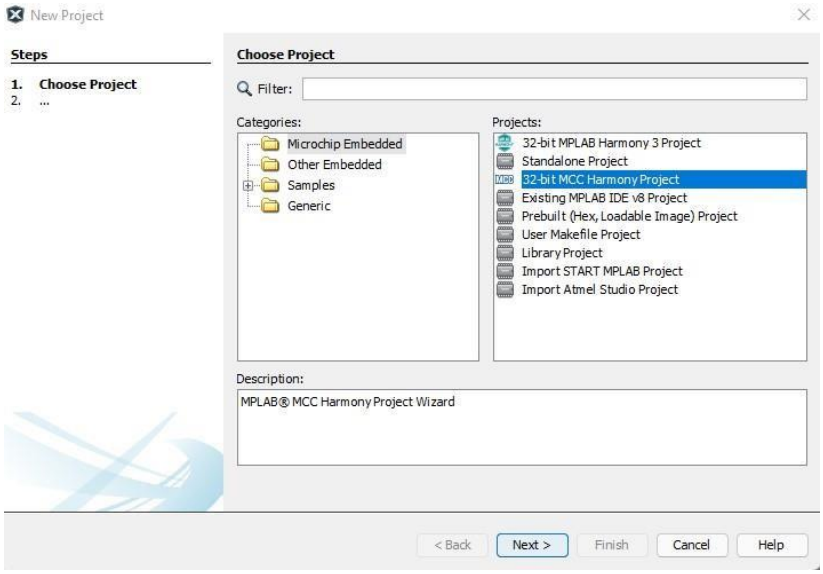
Manager Wizard

9.1 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)

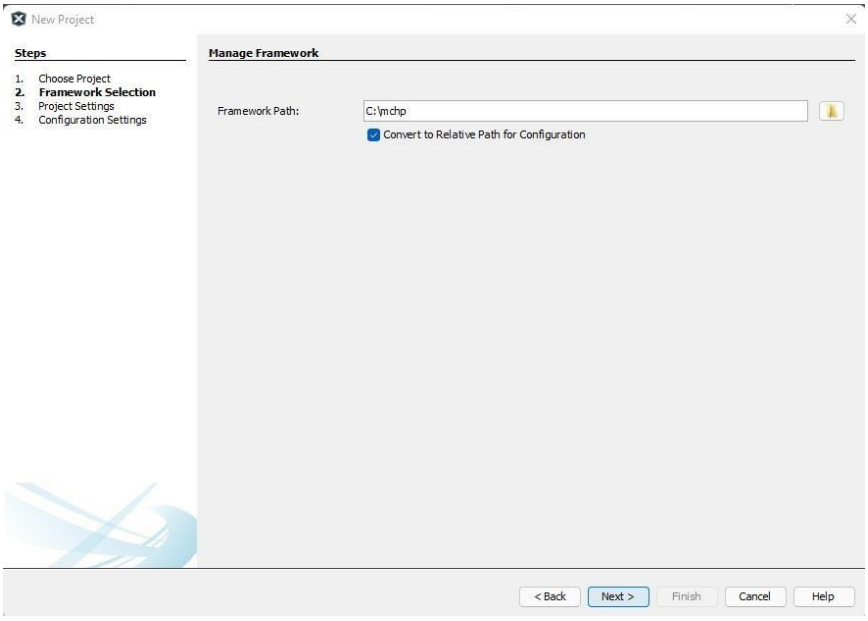
9.1.1 Select “New Project”



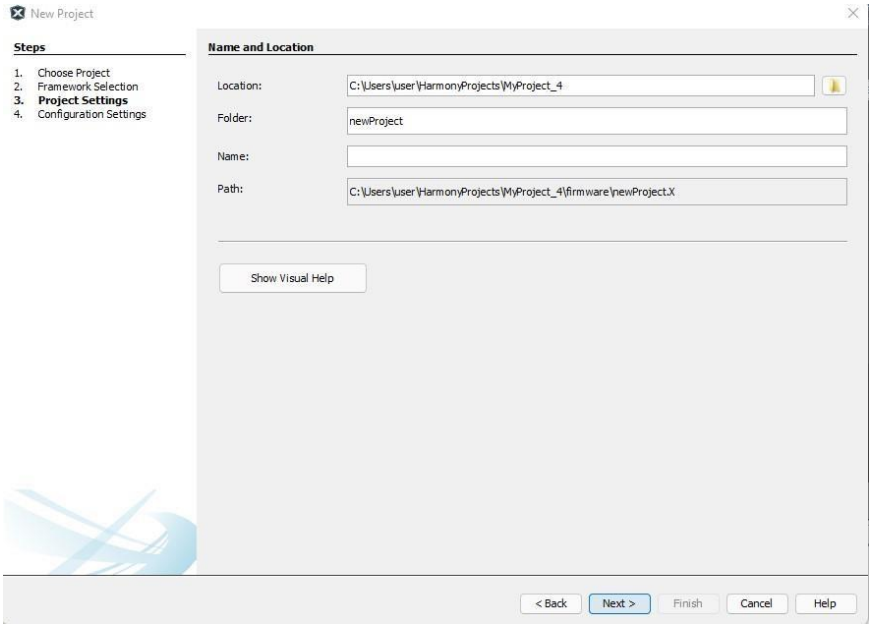
9.1.2 Select “32-bit MCC Harmony Project”



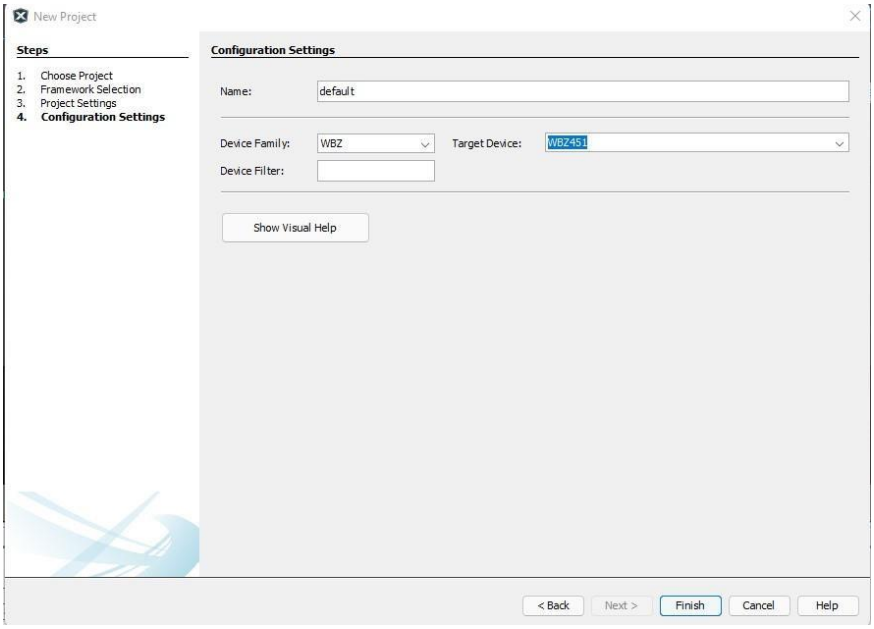
9.1.3 Framework Selection



9.1.4 Project Settings



9.1.5 Configuration Settings, click “Finish”

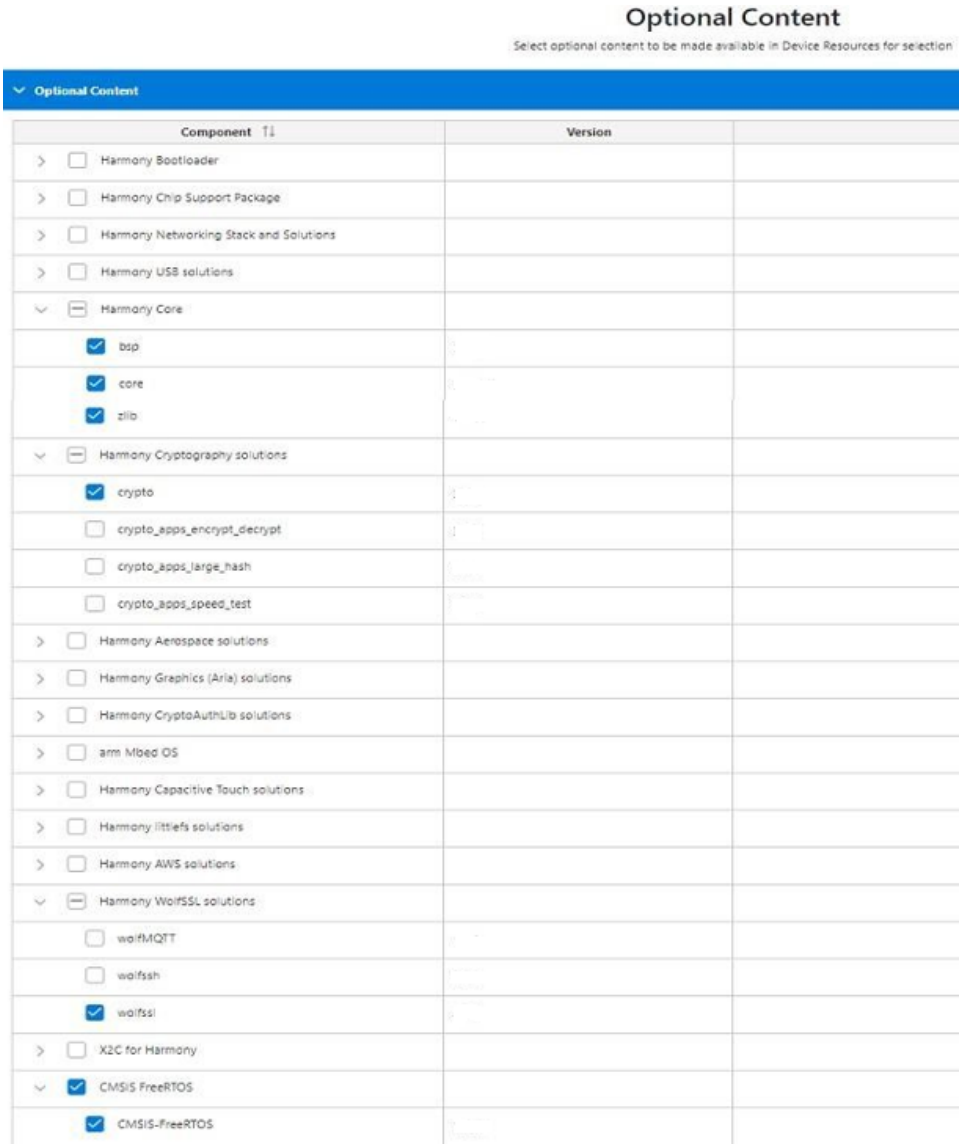


9.2 Open “MPLAB Code Configurator” after creation of project

MCC content manager window will appear in IDE

9.3 Select “MPLAB Harmony” from the content manager window


MCC Content Manager Wizard
1. Content Type
2. Required Device Content
Select a Content Type
MCC Melody
MCC Classic
MPLAB® Harmony
Embedded Software Development Framework for 32-bit Microcontrollers and Microprocessors
Select MPLAB Harmony
Release notes and supported devices
Release notes and supported devices
Release notes and supported devices
Library support may be a key factor in your choice of MCC flavor:
MCC Melody and MCC Classic - Library Summary
MPLAB Harmony - Library Summary
Still unsure which content type is right for your project?
See More Details

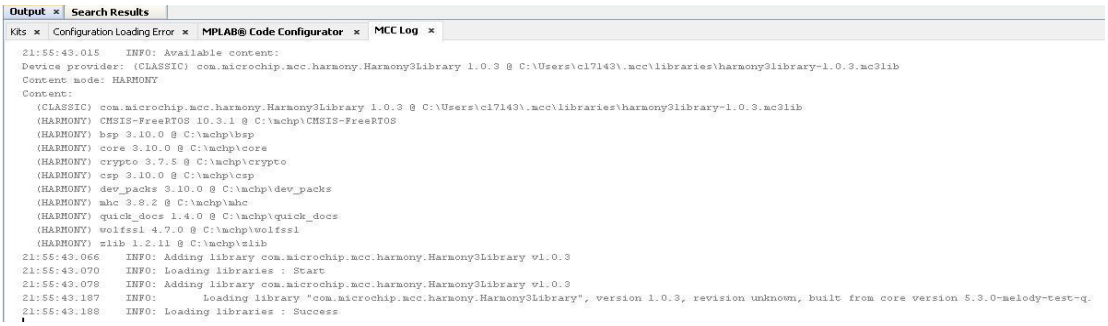


Note: Versions selected in the images may not be the latest,

Ensure all components are selected as displayed below

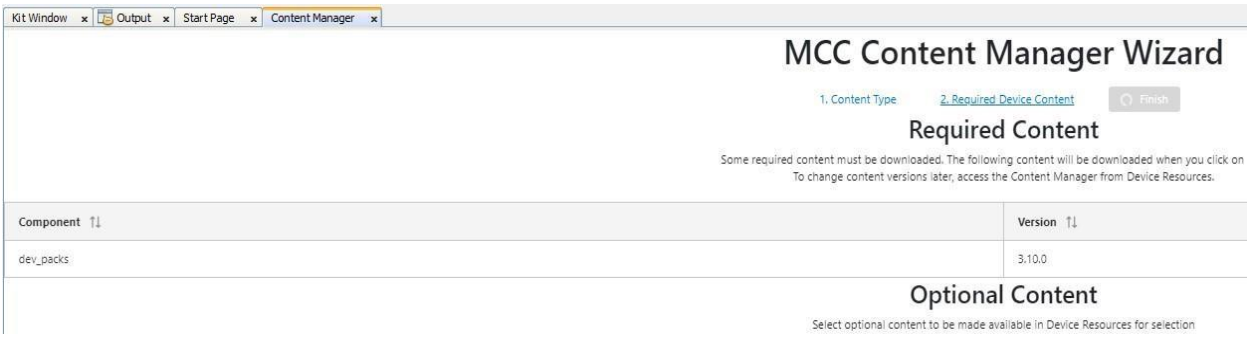
Select “Finish” The downloading of selected components from harmony repositories will take some minutes.

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

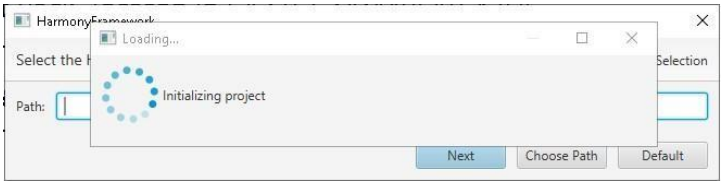


If in the process a failure to download a particular component appears, try redownloading again only the component that failed to download.

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”

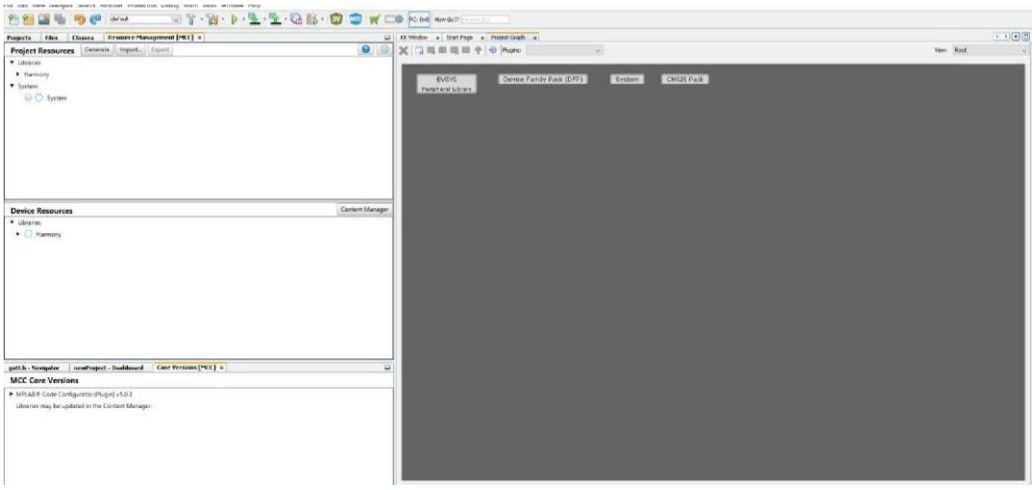


10. Select/Confirm the Harmony Framework Path



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

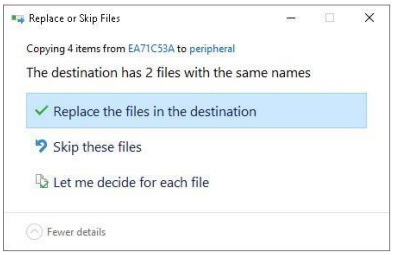
Project Graph Window will appear after choosing the framework path and successful initialization of project



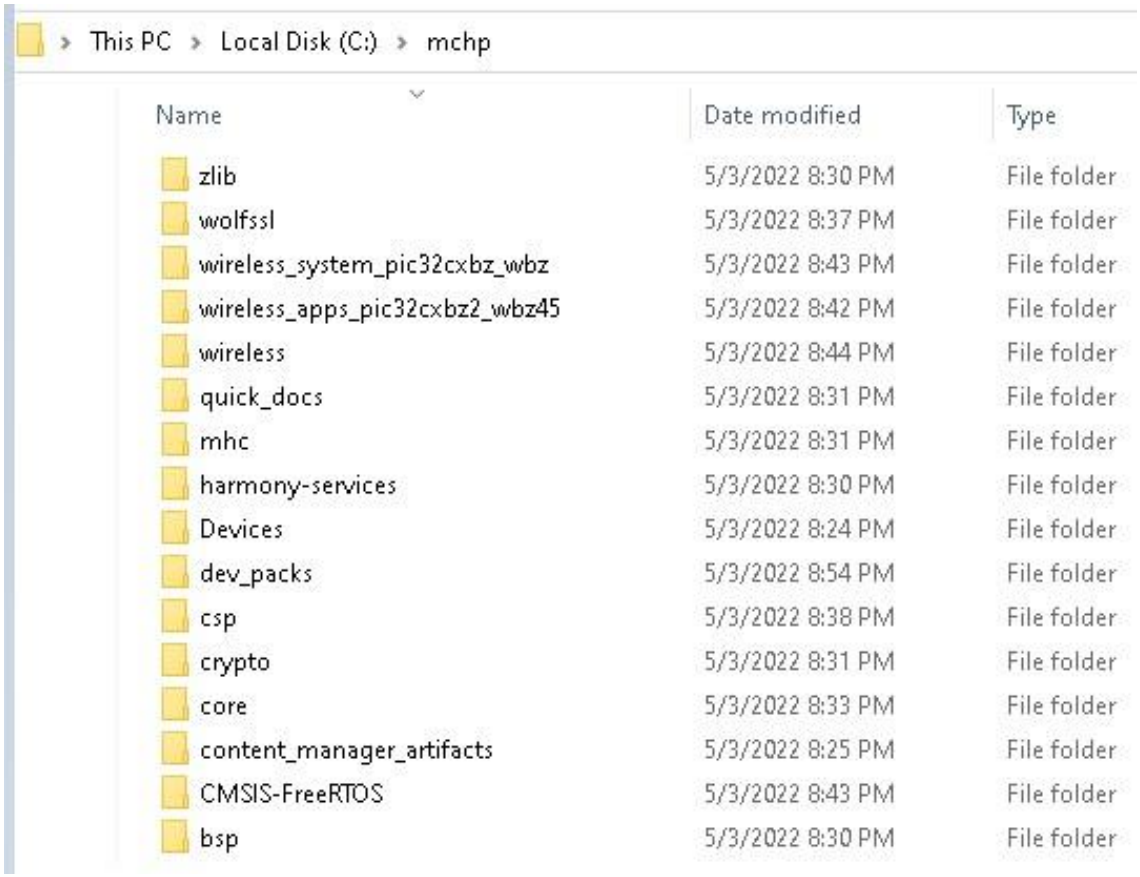
11. Ensure content versions as chosen based on the table below - using the content manager from Device Resources and select “Apply”

Harmony Chip Support Package				
csp	3.12.0	Latest		
Harmony Core				
core	3.10.0	Latest		
Harmony Tools				
mhc	3.8.5	Latest		
bsp	3.12.0	Latest		
Harmony Cryptography solutions				
crypto	3.7.5	Latest		
crypto_apps_encrypt_decrypt	-	Available		
crypto_apps_large_hash	-	Available		
crypto_apps_speed_test	-	Available		
harmony-services	1.1.0	Latest		
MCC Harmony Core				
com.microchip.mcc.harmony.Harmony3Library	1.1.1	Latest		Communicates with the MCC core, providing views and other functionality for MCC Harmony
Harmony Capacitive Touch solutions				
Harmony littlefs solutions				
Harmony AWS solutions				
Harmony TensorFlow Lite for Microcontroller (TFLM) Solutions				
Harmony WolfSSL solutions				
wolfMQTT	-	Available		
wolfssh	-	Available		
wolfssl	4.7.0	Latest		
X2C for Harmony				
CMSIS FreeRTOS				
CMSIS-FreeRTOS	10.3.1	Latest		
zlib	1.2.11	Latest		
dev_packs	3.12.0	Latest		

12. 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".



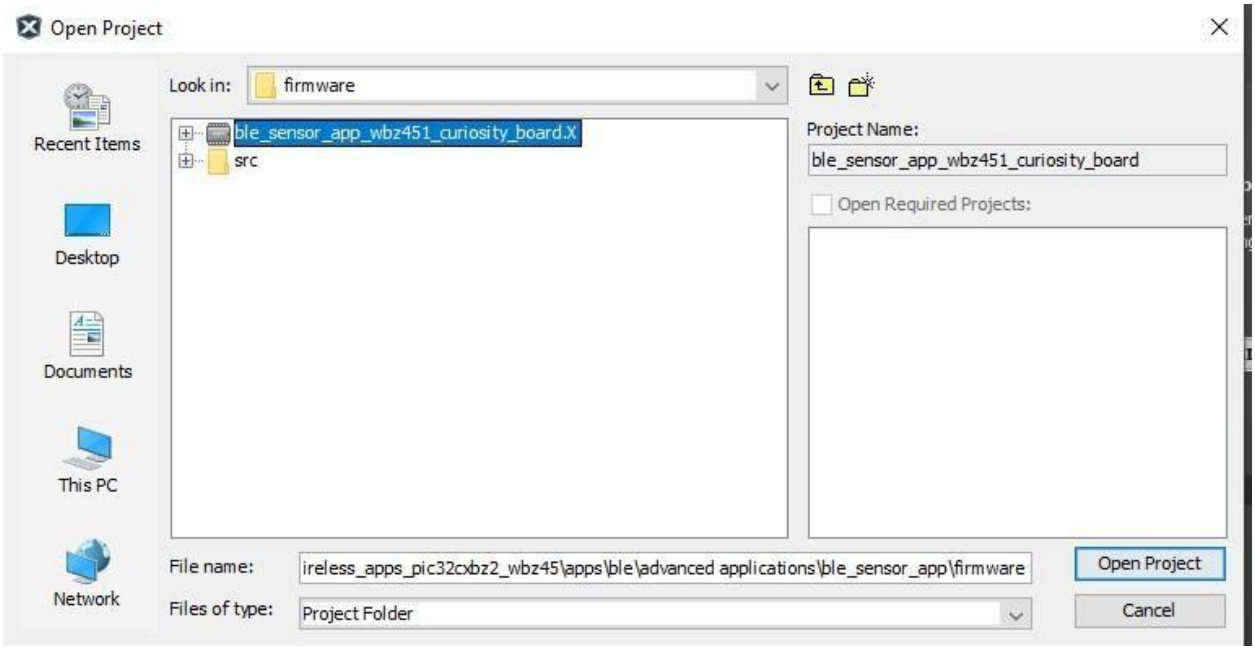
13. Copy all the 3 folders titled “wireless_” located in “EA71C53A\H3” to “C:\mchp”
Here is an example of how the folder should look like after copying the “wireless_” folders



Next Steps

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

15. How to open, build and program an existing application example
    Pre-requisites: Complete steps 1-11
    - 15.1 Connect Curiosity Board to the PC using usb cable
    - 15.2 Open MPLAB IDE
    - 15.3 Select **File > Open Project**
    - 15.4 Select the project from C:\mchp\wireless_apps_pic32cxbz2_wbz45\apps\ble\advanced applications\ble_sensor_app\firmware
        - 15.4.1 If user has multiple projects open, select the ble_sensor project, Right click and set the project as “Set as Main Project”



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

15.5 Open Project Properties

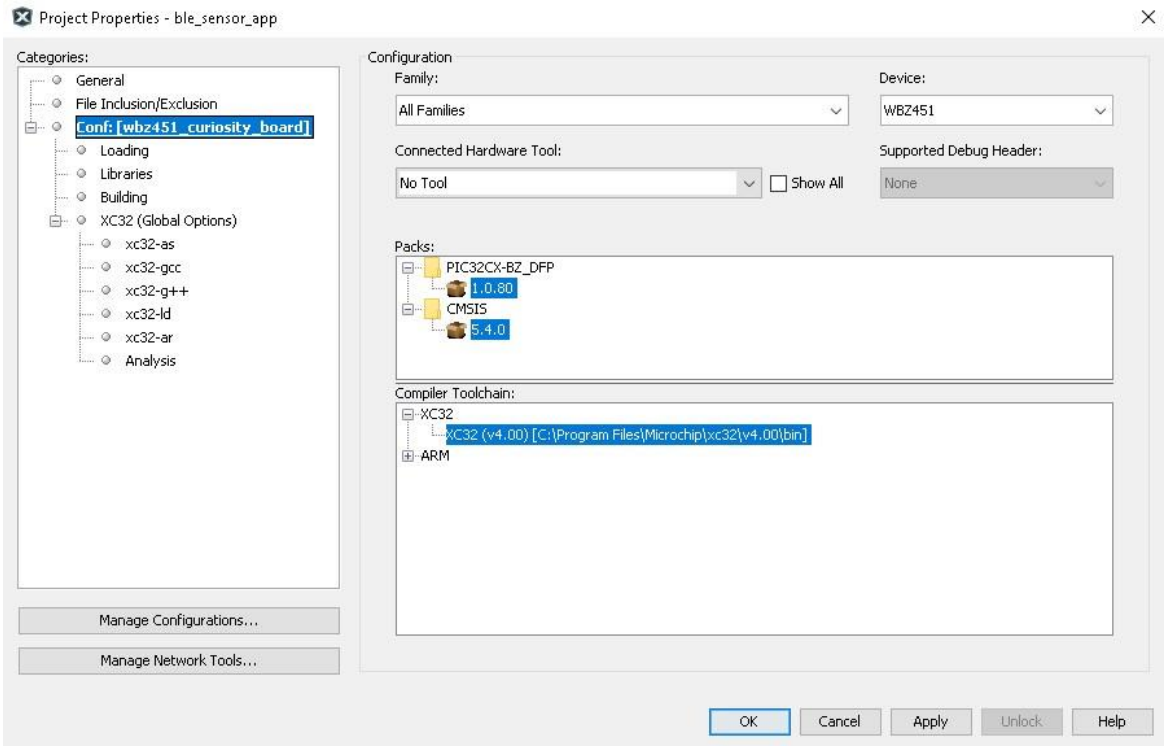
15.5.1 Select WBZ451 Curiosity Board as hardware tool for programming

15.5.2 Ensure DFP v1.0.xx is selected and CMSIS v5.4.0

Note: DFP version should match the version mentioned in table 1

15.5.3 select XC32 v4.00 compiler (in case user has several versions of XC32 compilers installed)

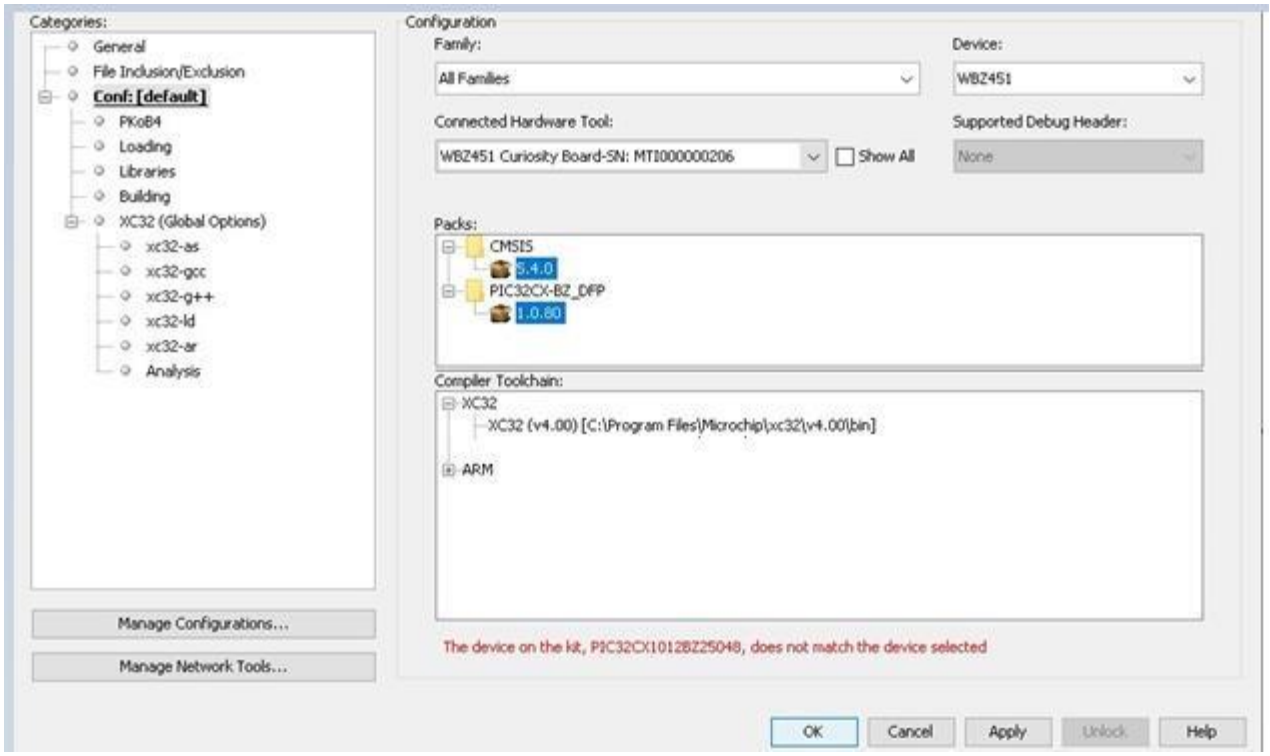
Note: Compiler version should match the version mentioned in table 1



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

15.7 Plug the Curiosity Development board to PC using usb cable

15.8 Select the “Connected Hardware Tool” in properties and Select “Apply”



Note: The message in red can be ignored as the project built for WBZ451 will directly run on the Curiosity board. WBZ451 is an RF module based of PIC32CX1012BZ25048 Soc

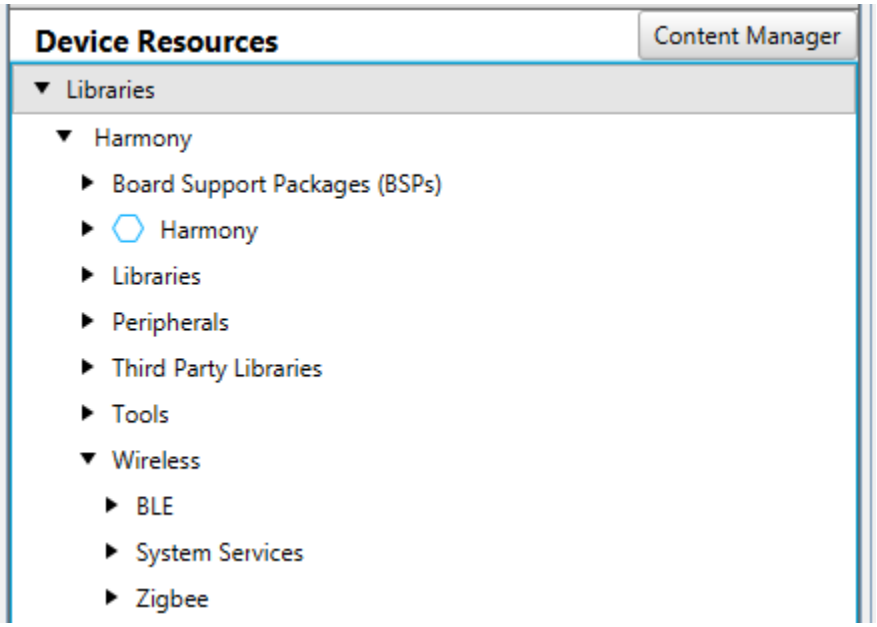
15.9 Select option **Run Project** in IDE to program the target – the onboard debugger will program the example application

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

Troubleshooting

1. If users see warning saying – “Project’s device is not supported by the currently loaded libraries”, uninstall MCC plugin from IDE-> step 6 – delete the. mcc folder -> Step 7- Install the MCC plugin again

2. If users cannot view the BLE stack/ BLE System services in the Device Resources, user's must have missed copying the "wireless_" folders (step 13) to the framework folder



3. If users cannot open the MCC configurator for a precompiled app example as part of this package, user's must have missed copying the "wireless_" folders (step 13) to the framework folder
4. If low power consumption is not observed as documented in the example user guide, user's should check if step 12 (copy "clk_pic32cx_bz")was completed and also should check the known issues.pdf document in /Documentation/ folder