

GW-MPC5748G RDB

SOFTWARE INTEGRATION GUIDE (SWIG)

Ultra-Reliable MCUs for Industrial and Automotive Applications



Contents

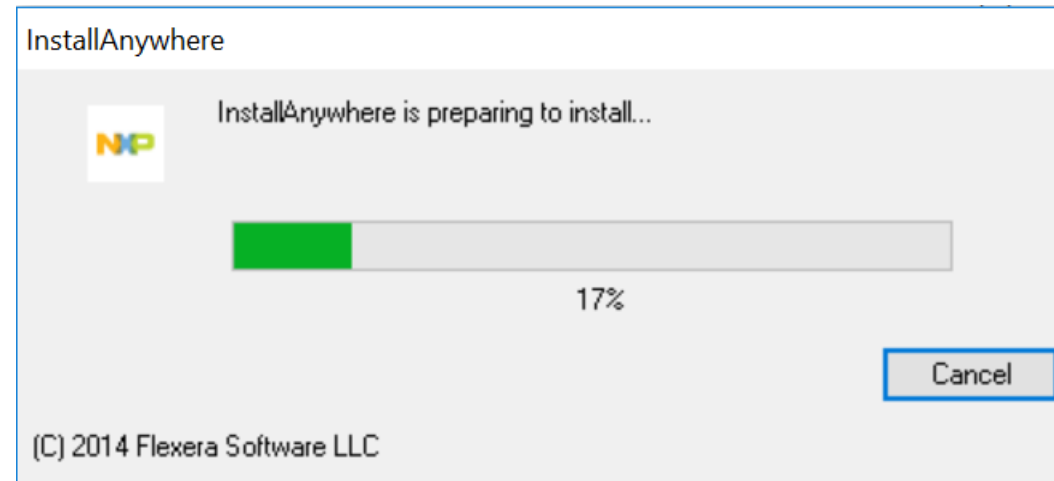
- Installing S32 Design Studio IDE for Power Architecture
- Update SDK
- SDK Overview
- Create a New Project
- Build and Debug Projects
- Import Projects

INSTALLING S32 DESIGN STUDIO IDE FOR POWER ARCHITECTURE



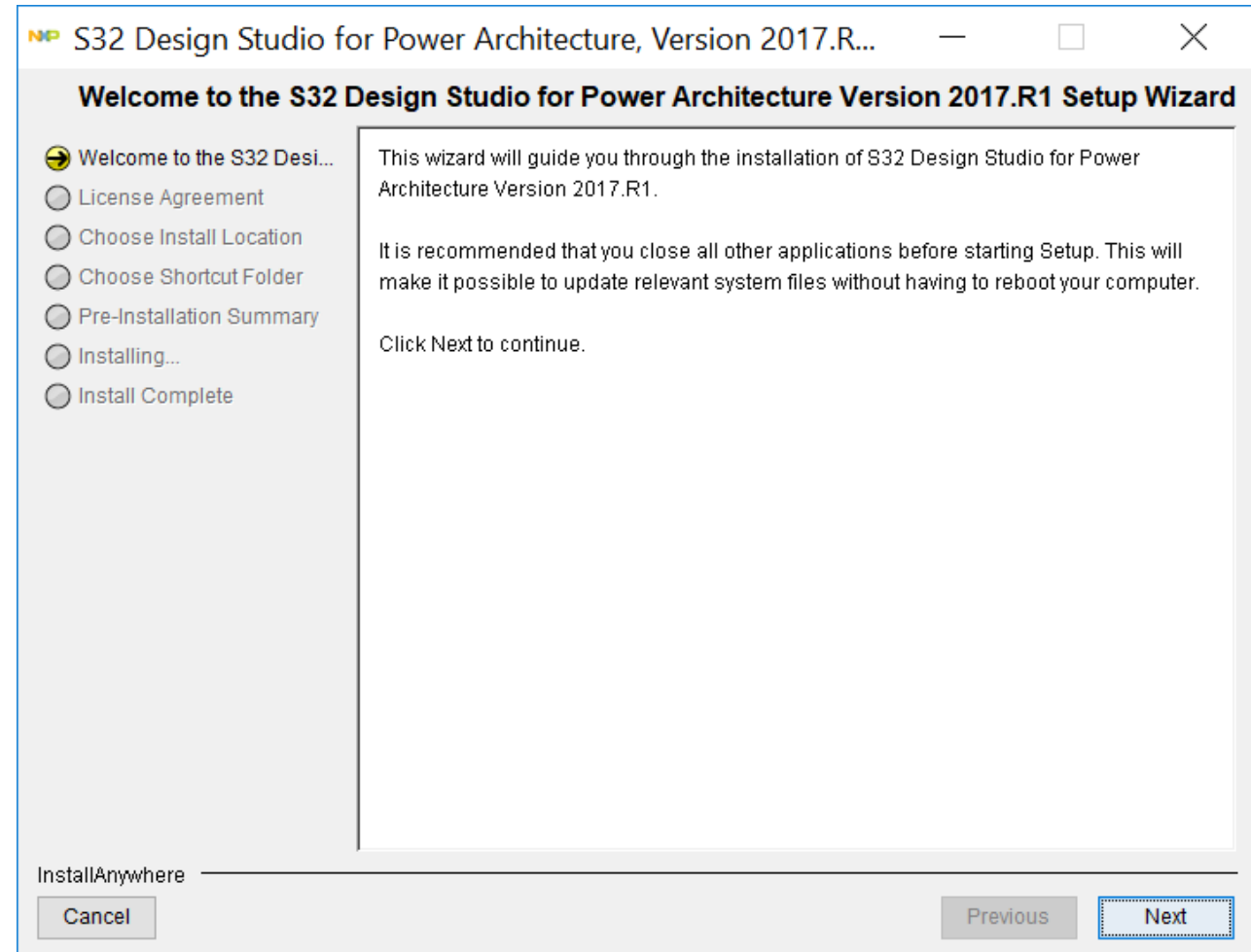
Step-1

- Go to https://www.nxp.com/support/developer-resources/run-time-software/s32-design-studio-ide/s32-design-studio-ide-for-power-architecture-based-mcus:S32DS-PA?tab=Design_Tools_Tab to download latest version of S32 Design Studio IDE for Power Architecture
- Go into download folder, run the installation file
- The “preparing to install” dialogue box will appear



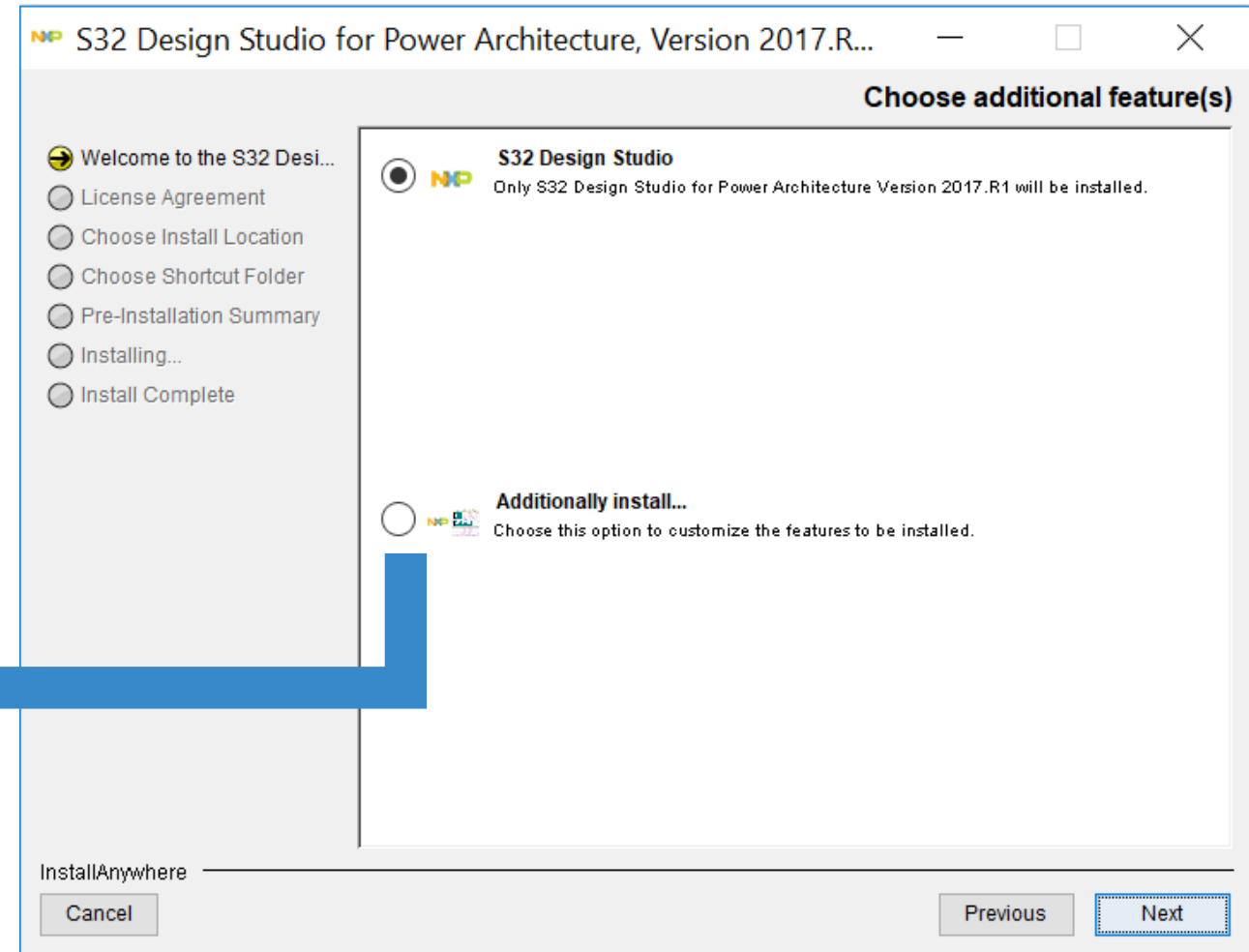
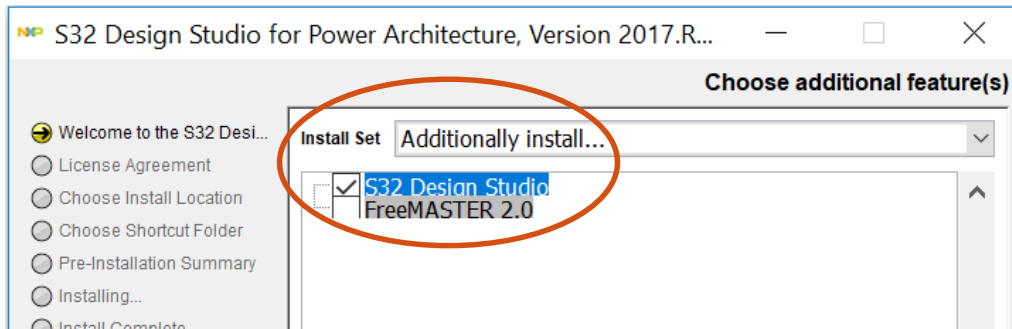
Step-2

- An Installer welcome window will be displayed, click **Next** to continue



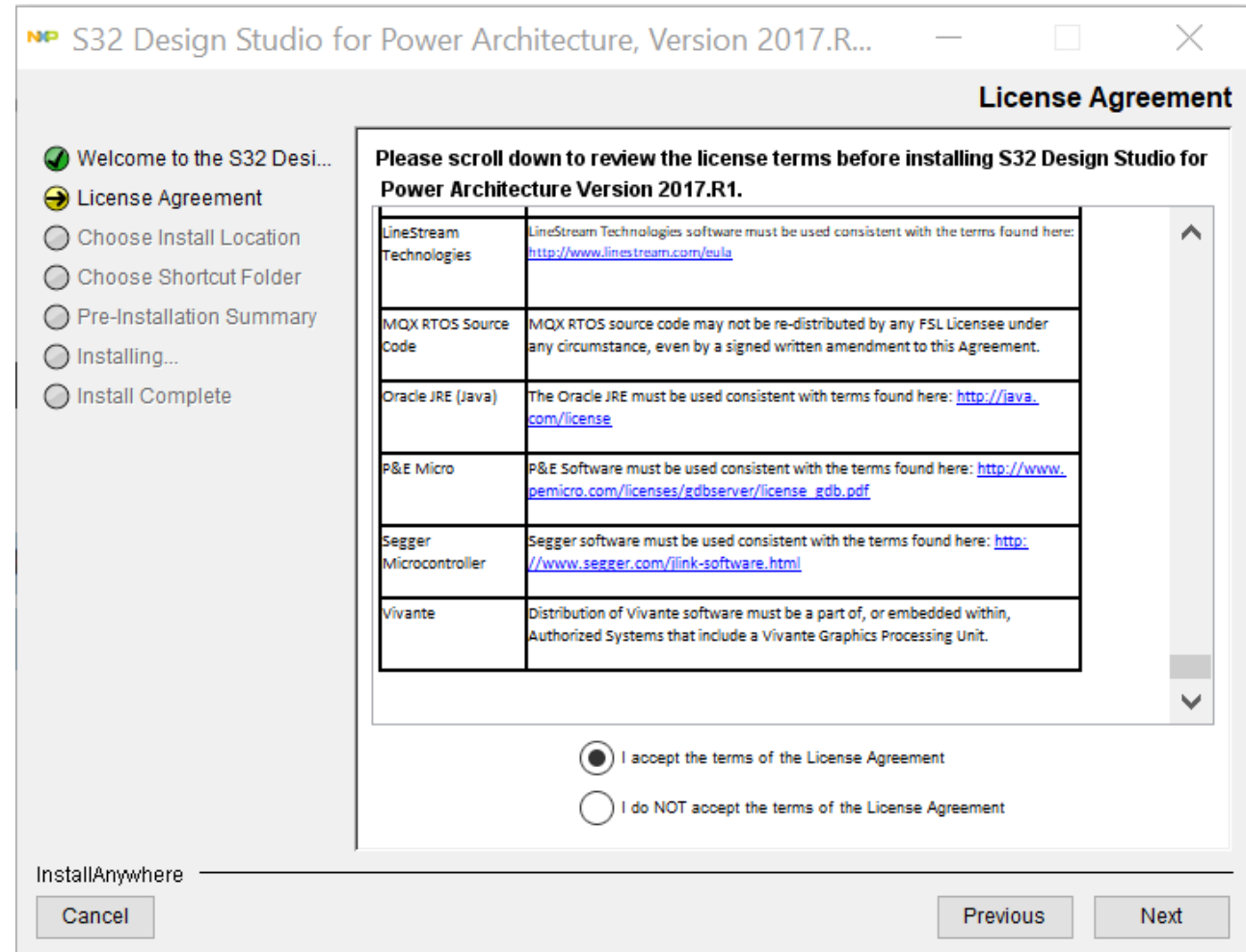
Step-3

- Choose additional features
 - Selecting “S32 Design Studio” option will only install S32 Design Studio
 - Selecting “Additionally install...” will allow you to install other software too
- Click on Next



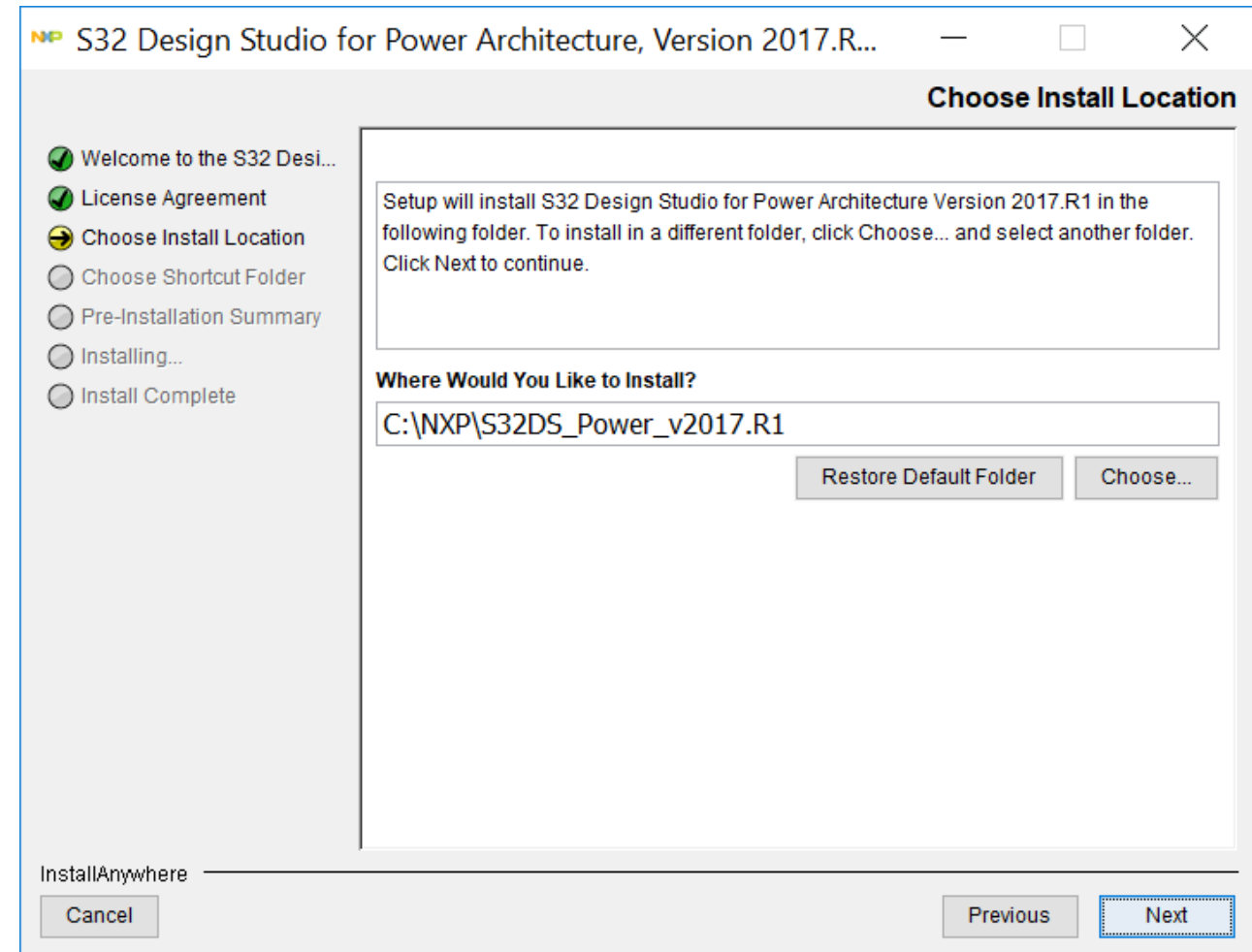
Step-4

- Read and scroll down the license agreement to end
- Select the radio button accepting the license agreement terms and click **Next** to continue



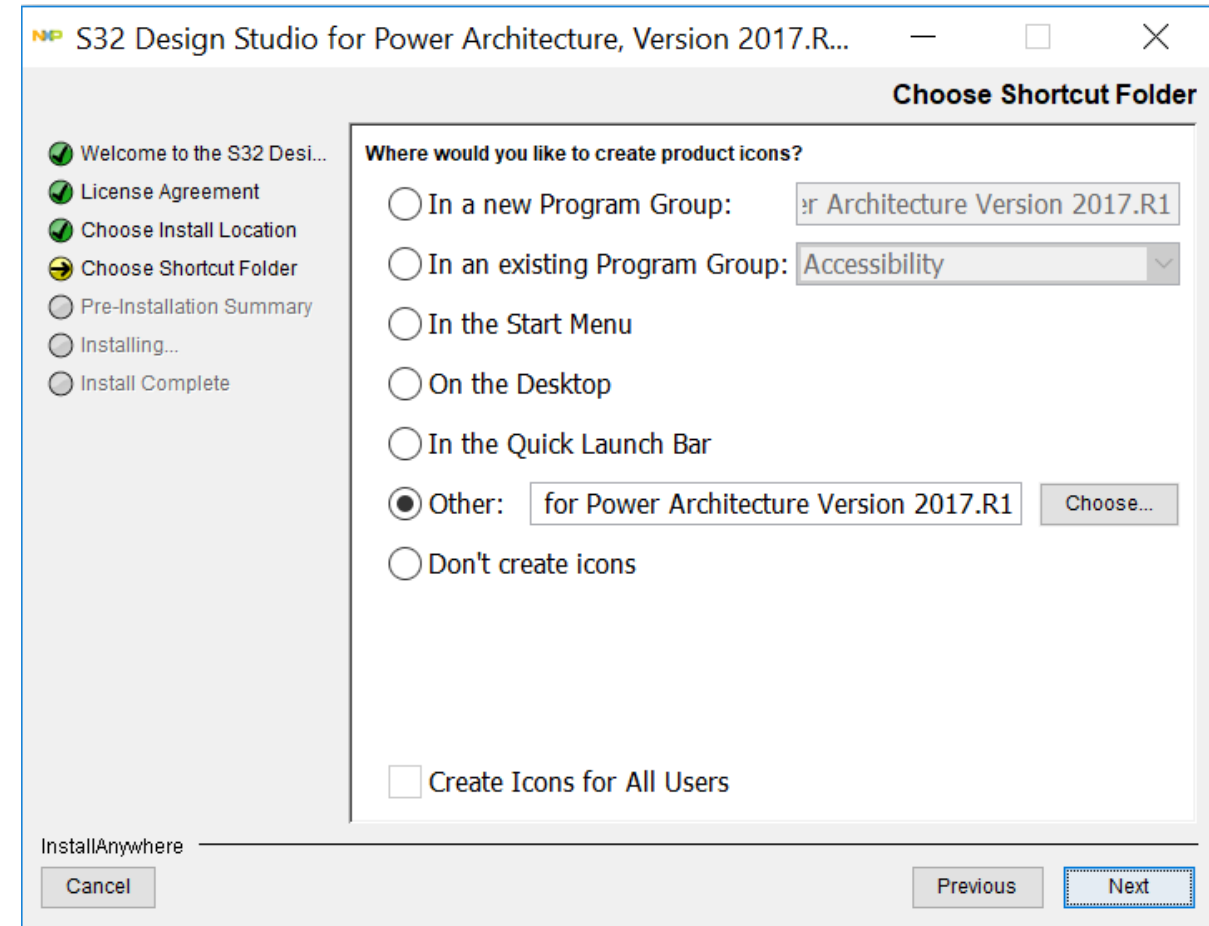
Step-5

- Click **Next** to choose the default installation location (Could be changed, but recommended to install into path without spaces)



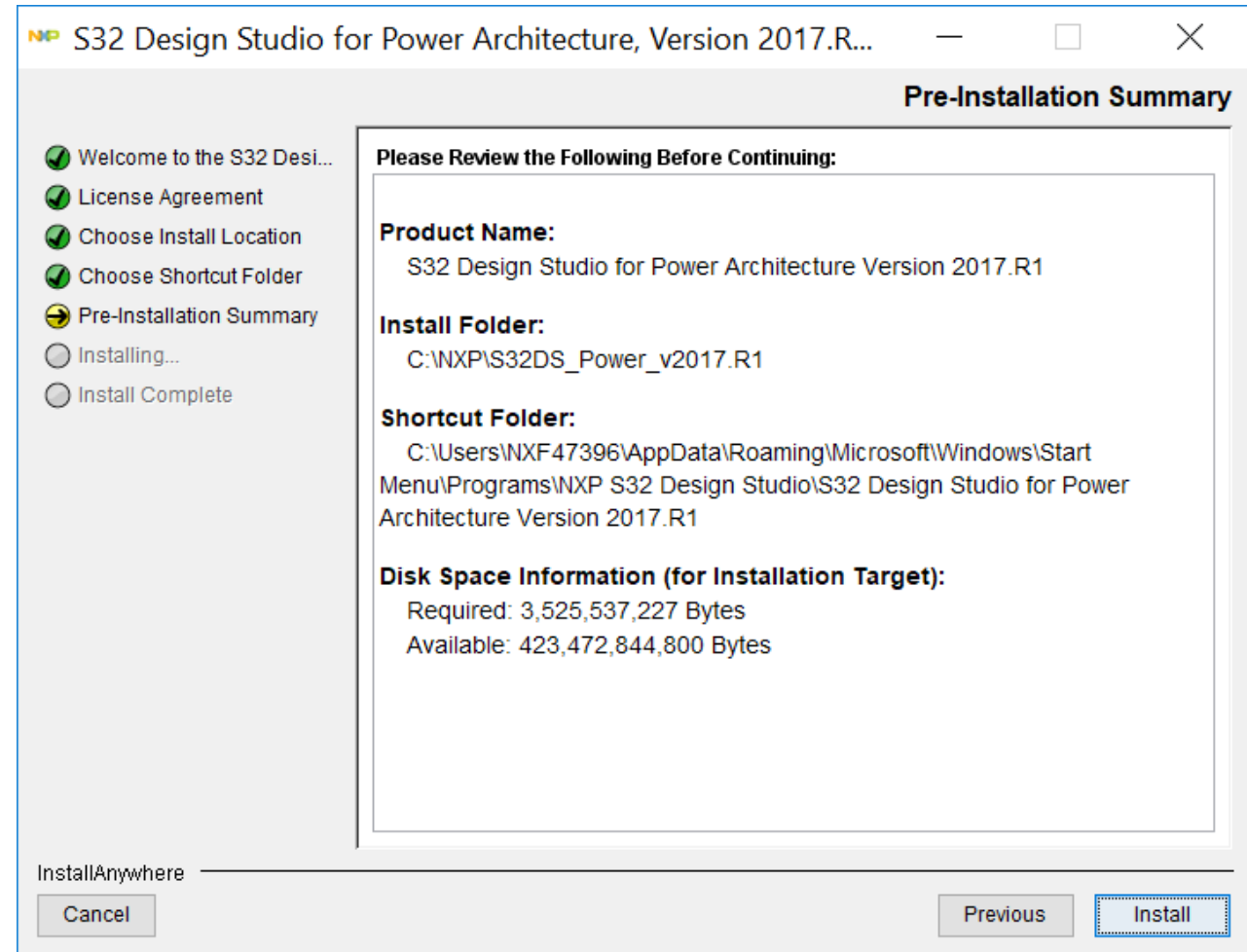
Step-6

- Select path where you want to generate a shortcut and click **Next** to continue



Step-7

- Verify settings on “Pre-installation Summary” tab and click **Install** to start Installation

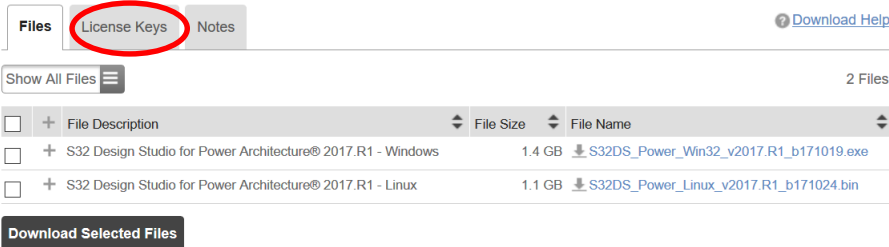


Step-8

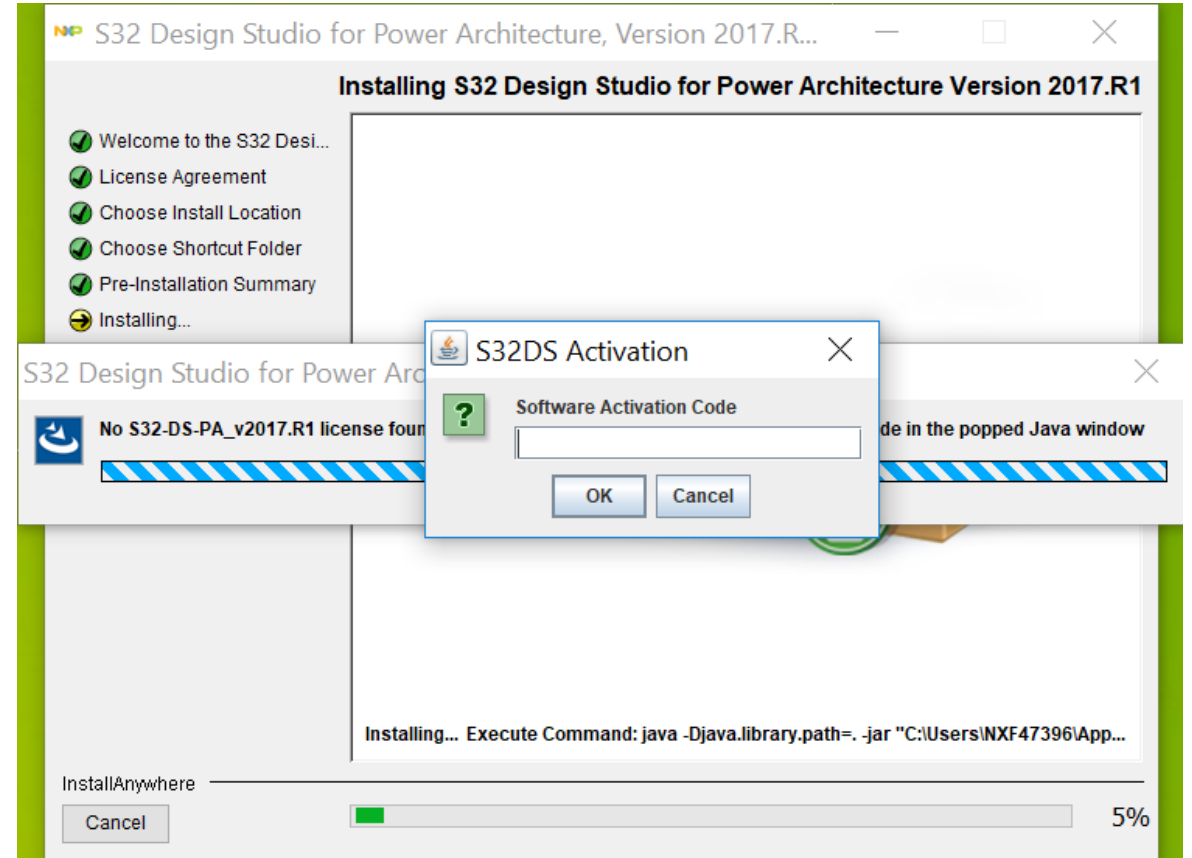
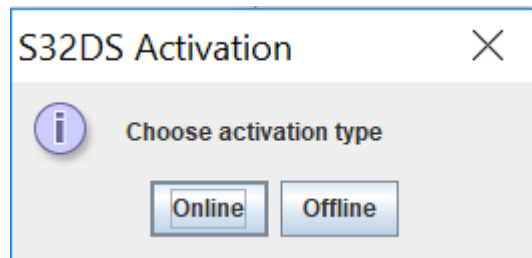
- When asked for Software Activation Code, copy and paste from the download page

Product Download

S32 Design Studio for PA E200

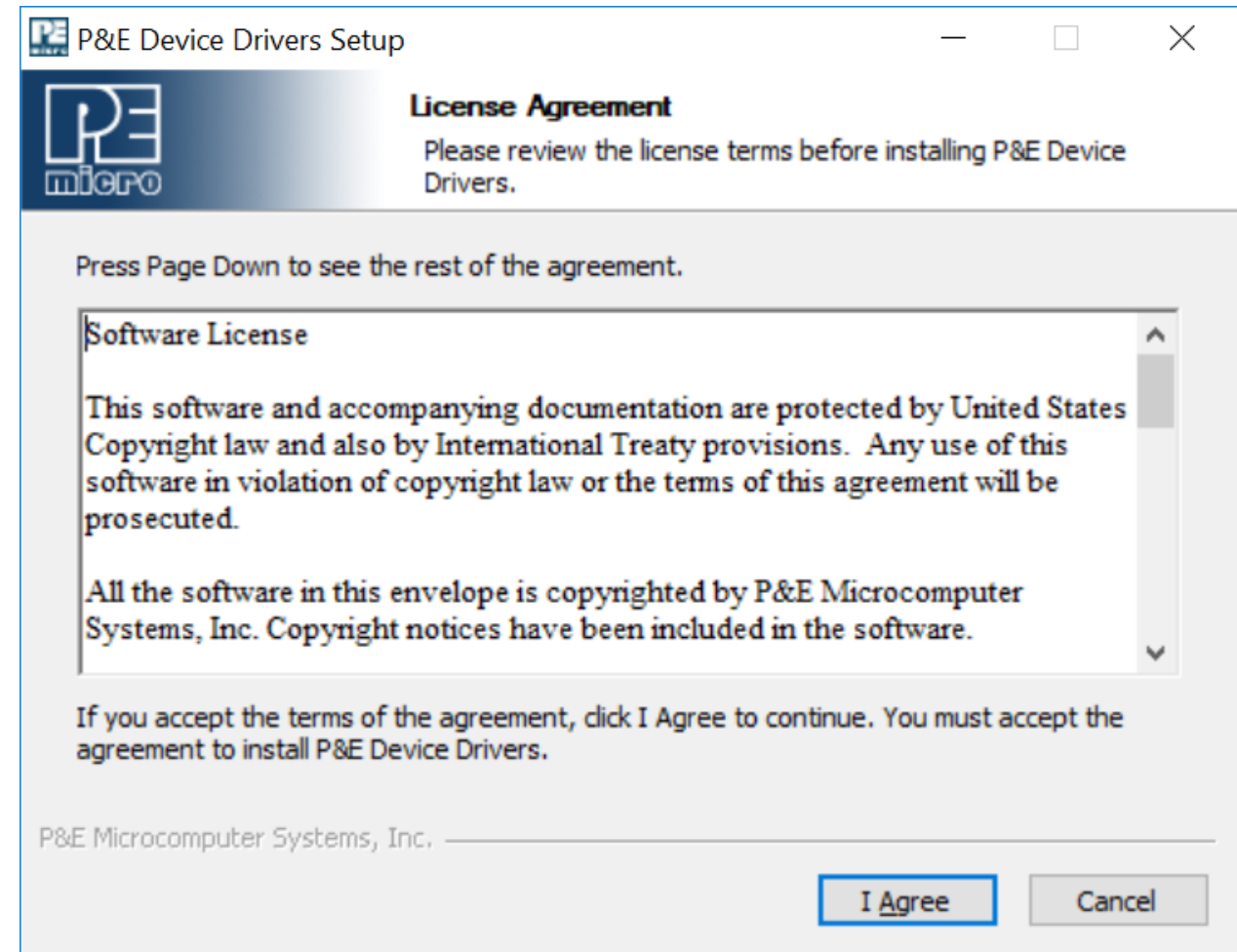


- Click **OK**
- In activation type window, click **Online**



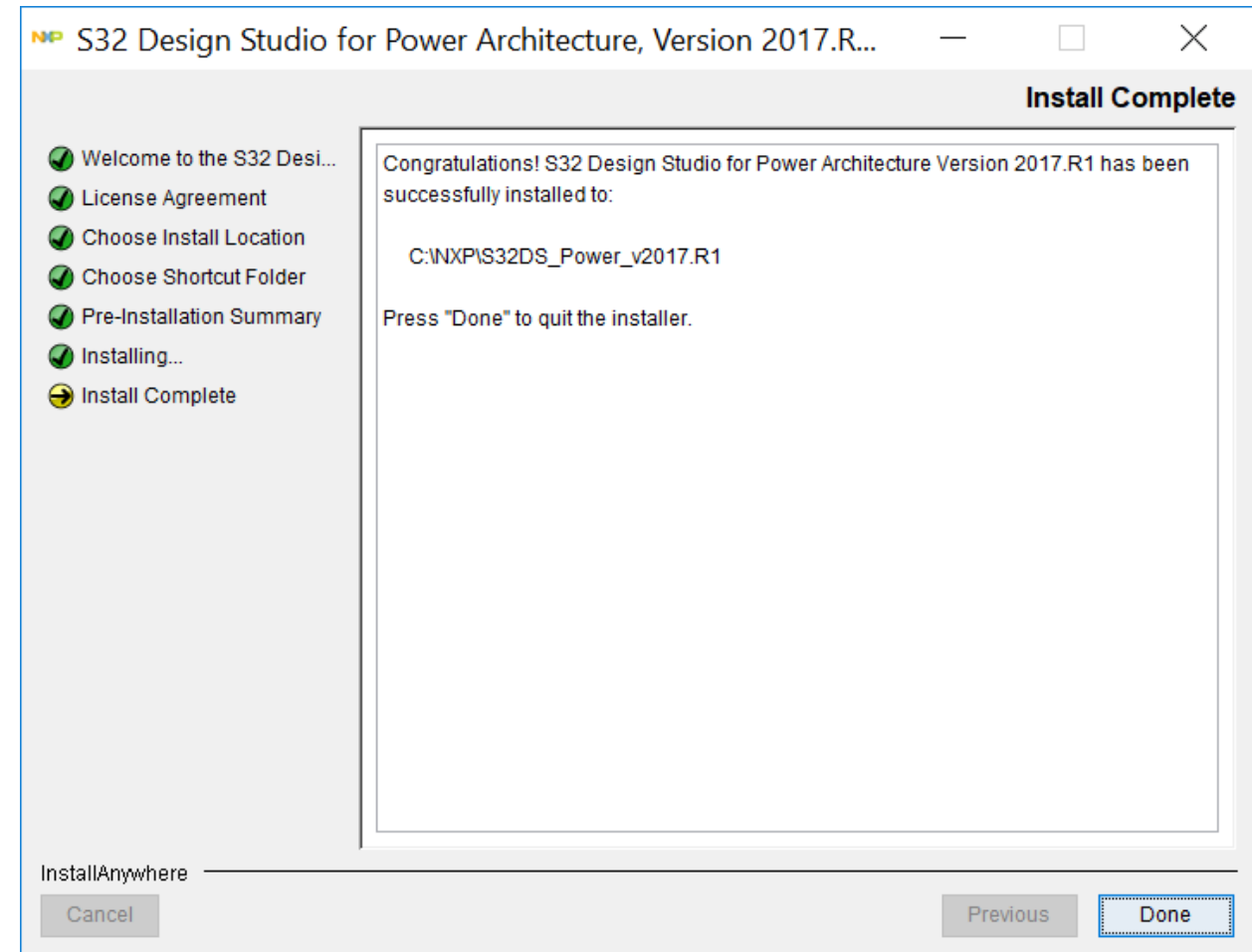
Step-9

- During the Installation it may ask you install P&E Device Drivers
- Read license agreement and click **I agree**
- Then select the Destination Folder and **Click Install**
- When the installation is done, click **Close** to close the P&E Device Driver Setup window



Step-10

- Once the installation is completed click **Done** to exit the installation wizard



UPDATE SDK

Step-1

- Go to https://www.nxp.com/support/developer-resources/run-time-software/s32-design-studio-ide/s32-design-studio-ide-for-power-architecture-based-mcus:S32DS-PA?tab=Design_Tools_Tab to download **SDK RTM 1.0.0**
- If you already have SDK RTM 1.0.0, you can go directly to Step-2 without downloading



S32 Design Studio for Power Architecture 2017.R1 Updates 5 SDK MPC574xx RTM 1.0.0 and Update 6 Service Pack MPC5775x(REV UP5-UP6)

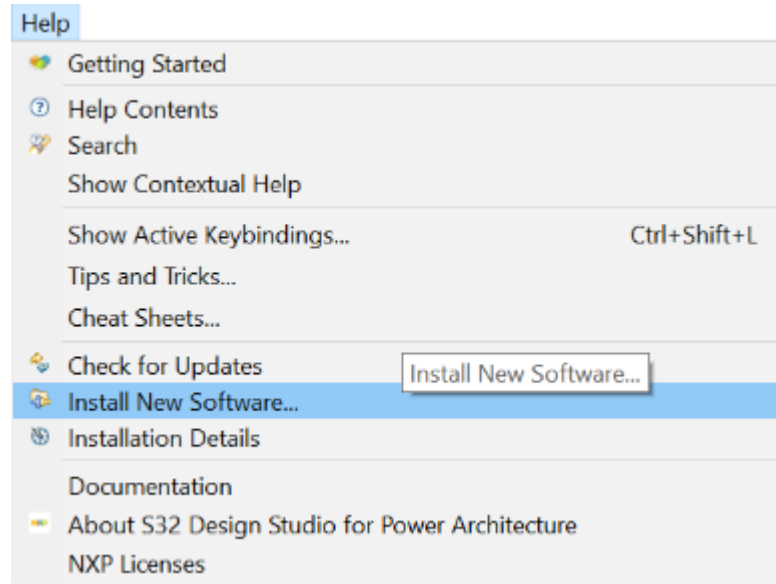
This update adds SDK MPC574xx RTM 1.0.0 and Service Pack to support MPC5775 B/E. This update is cumulative from previous updates and applicable for S32 Design Studio for PA v2017.R1.

 ZIP 1140618 KB S32DS_PA_v2017.R1_UP5_UP6 2018-08-03 14:42:00

[Download](#)

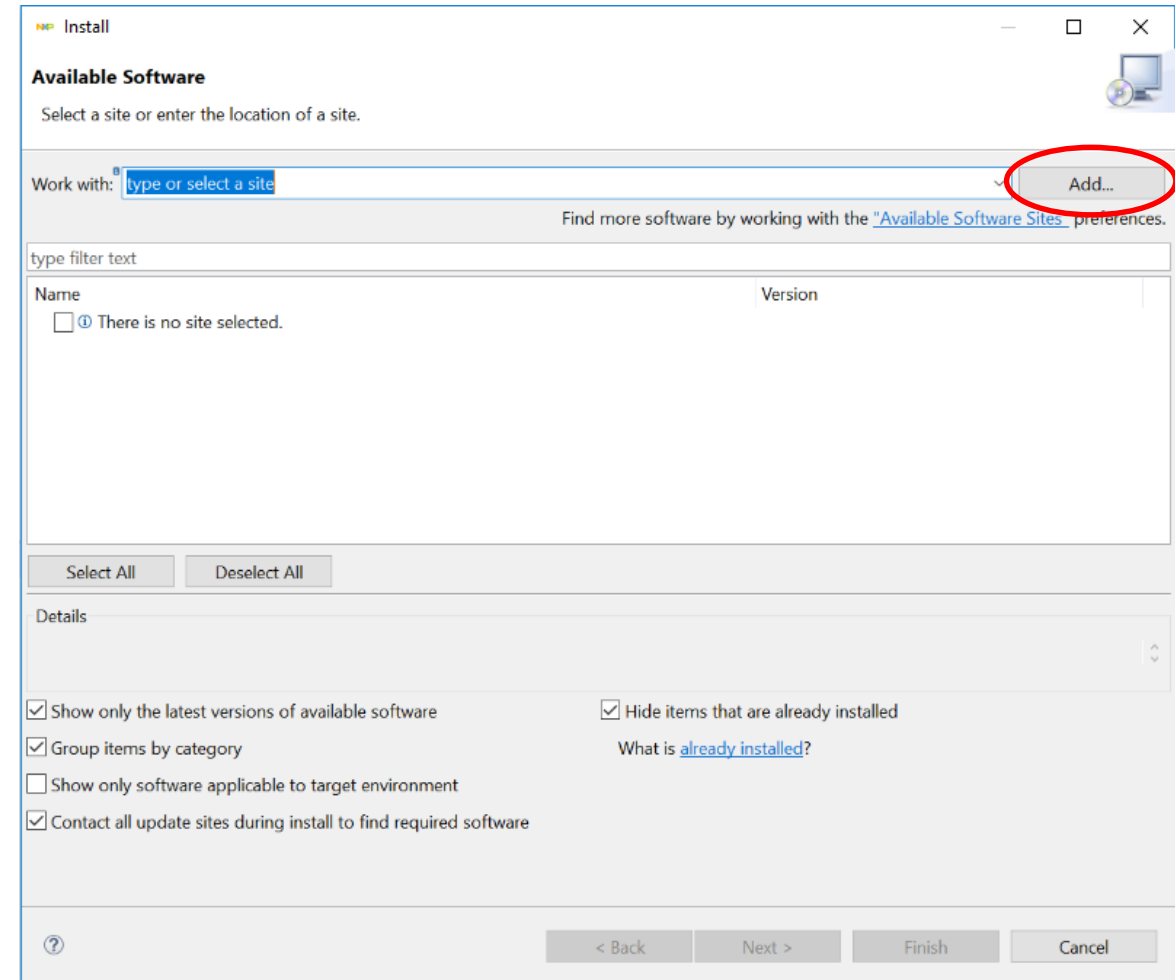
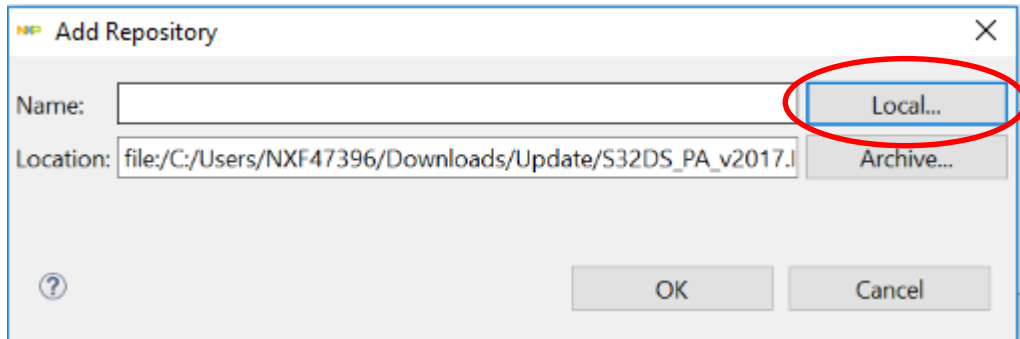
Step-2

- Go to menu bar: Help — Install New Software



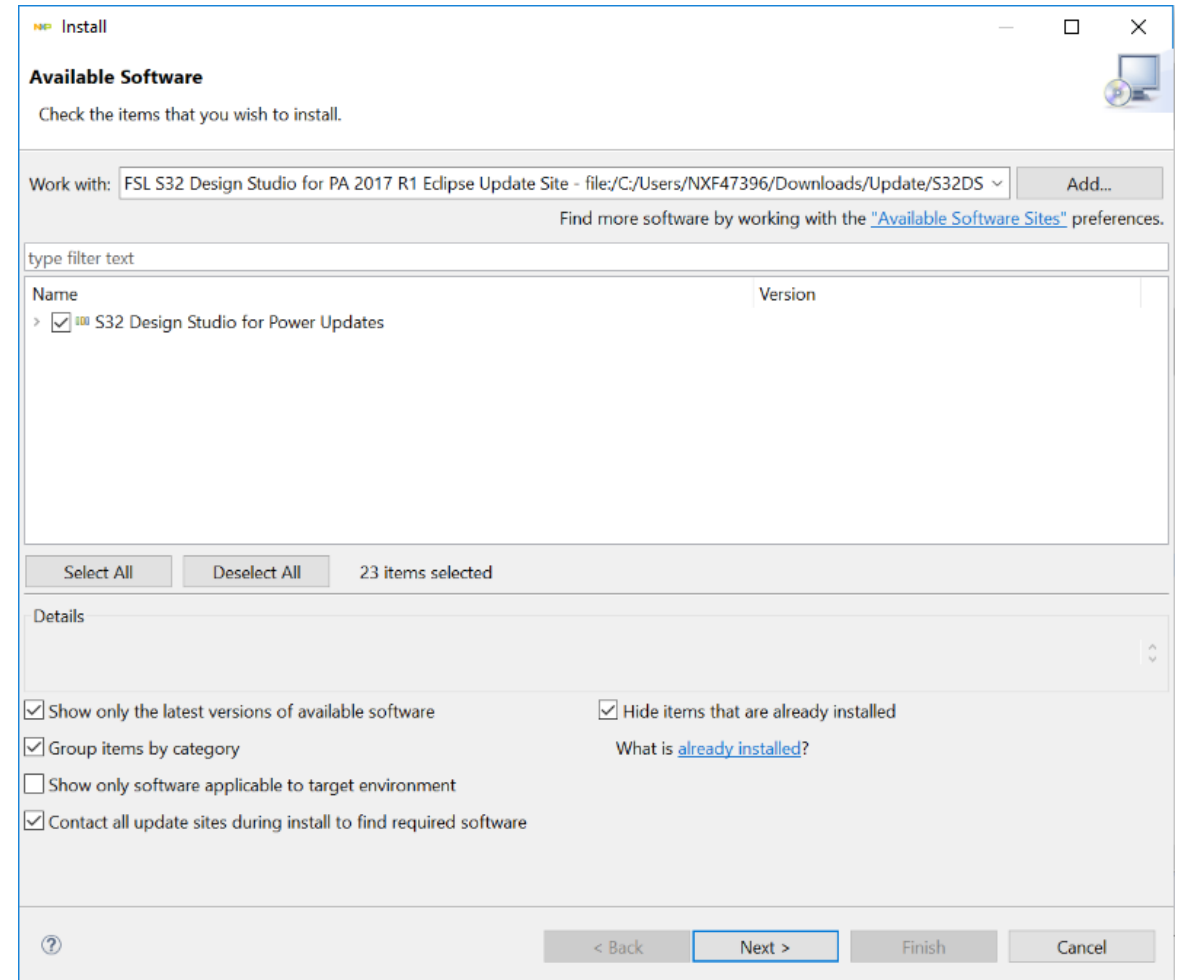
Step-3

- Click “Add...” to find the SDK which has been downloaded in Step-1
- Click “Local...” to add the paths of SDK
- Then click on “OK”



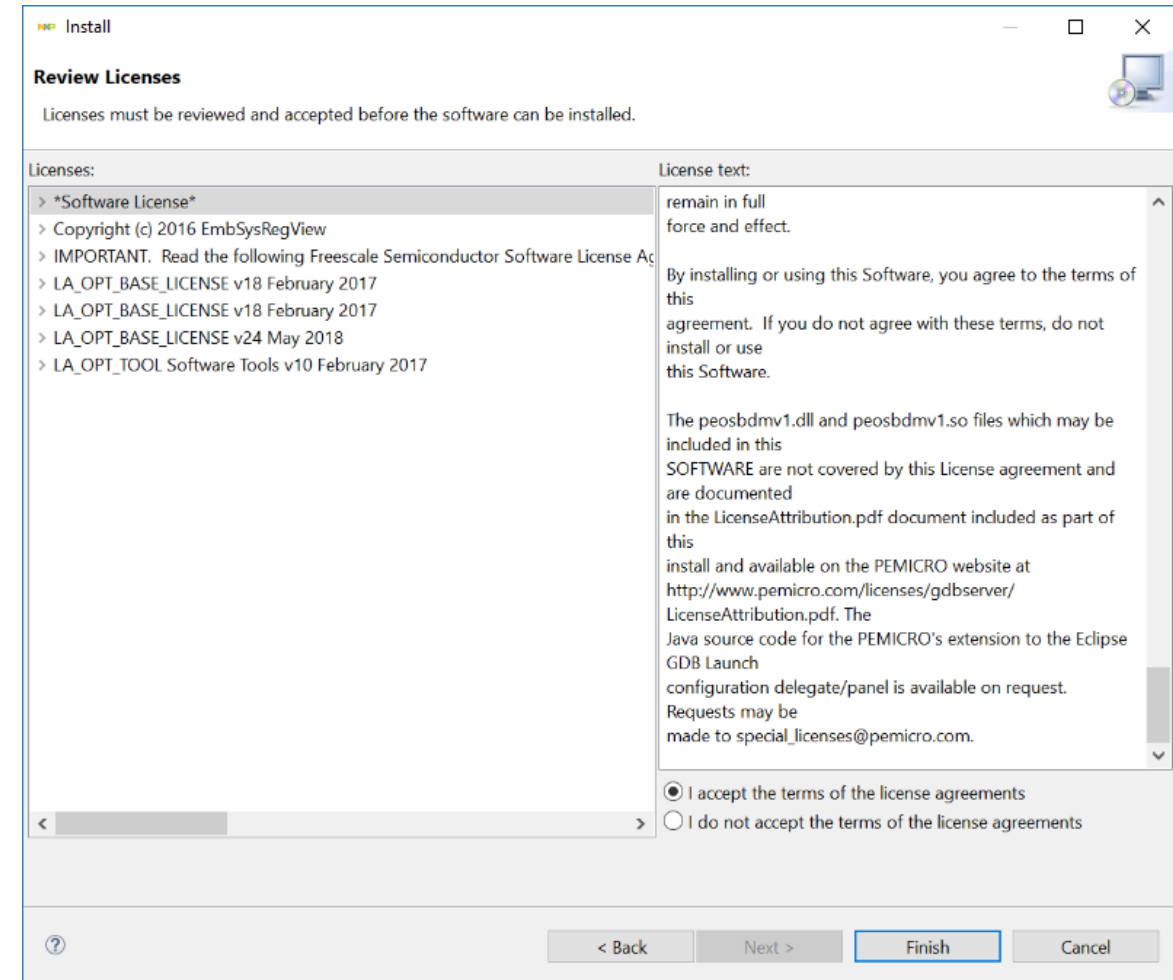
Step-4

- Check the “S32 Design Studio for Power Updates”
- Then click on “Next” to start update



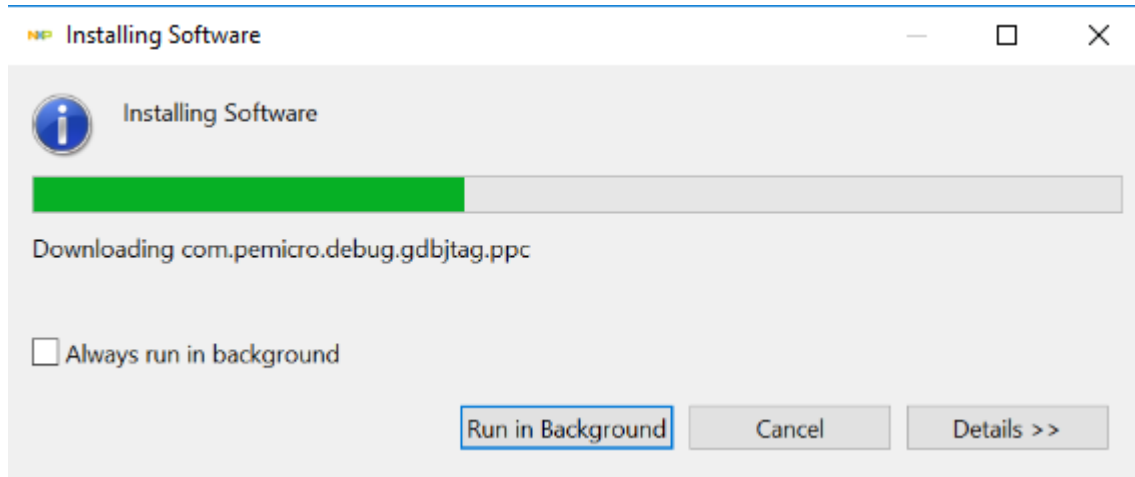
Step-5

- In “Install Details” window, Click on “Next”
- In “Review Licenses” window, read and accept license terms
- Click on “Finish”

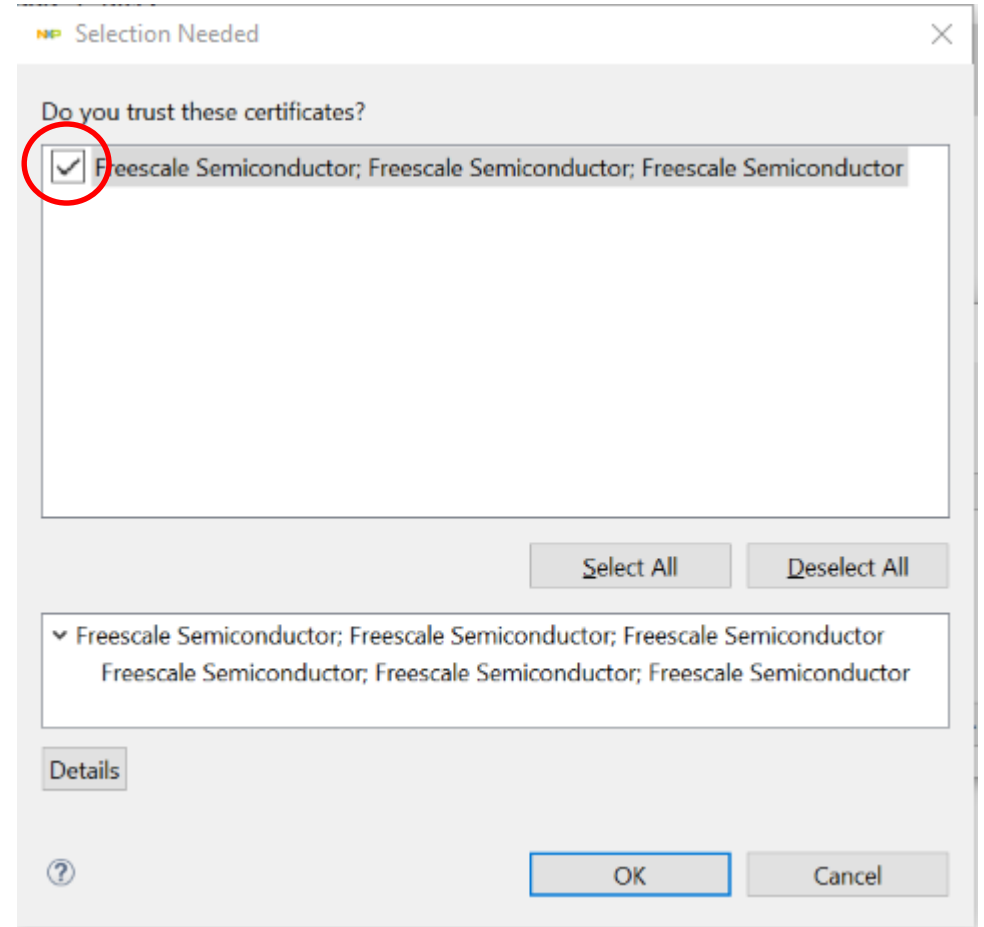


Step-6

- Wait for installation



- When asked “Do you trust these certificates”, check it and click on “OK”
- When finished installation, restart S32DS



SDK OVERVIEW

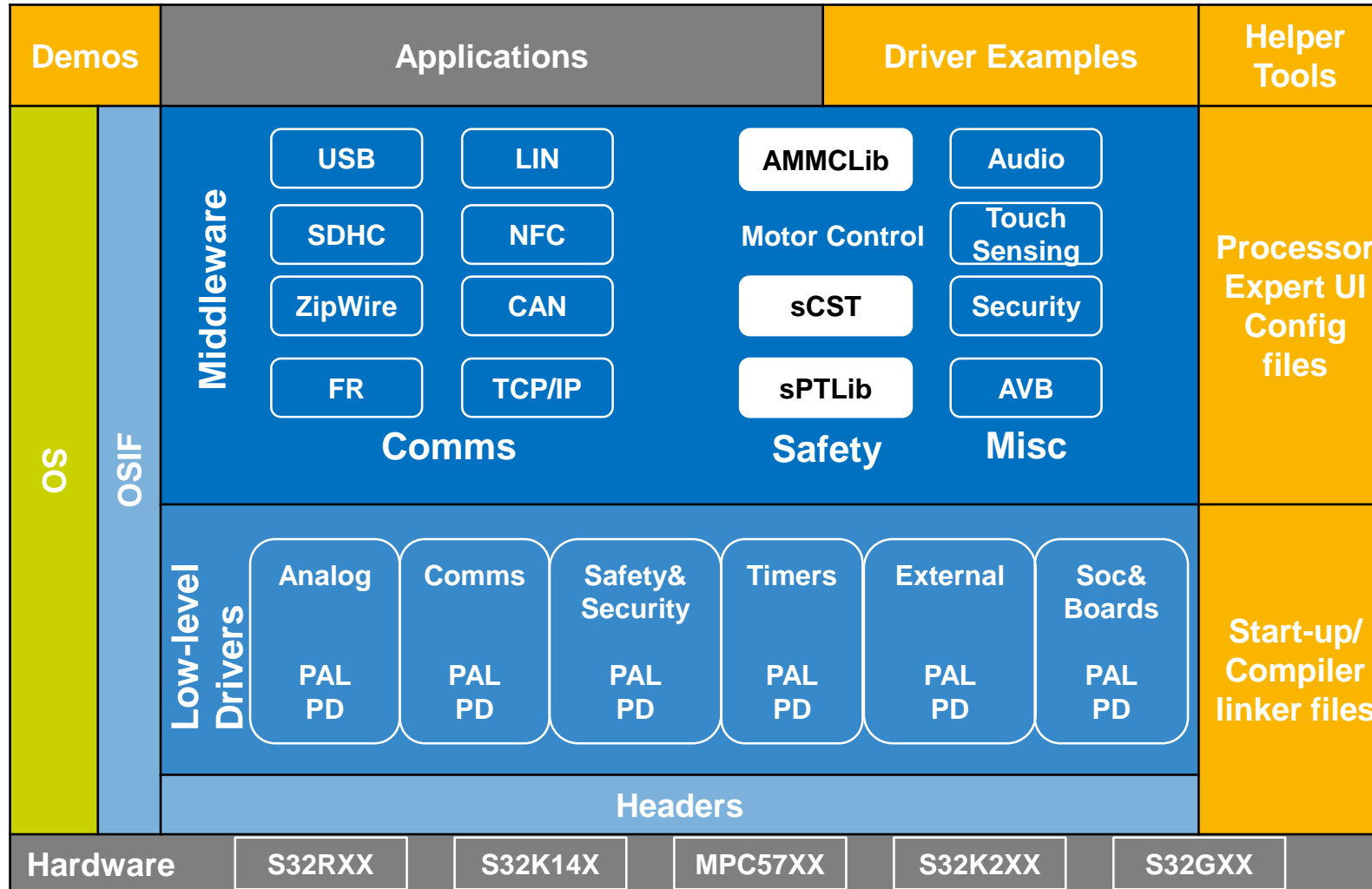


S32 Software Development Kit (SDK)

- Non-Autosar Software package
- **Automotive Grade:** SPICE/CMMI compliant, MISRA 2012
- Graphical-based configuration
- Compatible with Eclipse & other IDEs
- Supports MPC574x family
- Supports multiple toolchains



S32 SDK – Architecture



Features

- Integrated **Non-Autosar SW Production-grade** software
- Graphical-based Configuration
- Layered Software Architecture
- Documented Source Code and Examples
- Integrated with S32 Design Studio and other IDEs
- FreeRTOS integration
- Multiple toolchains supported
- Several examples and demos



S32 SDK – File Structure

S32 SDK

- ▼ SDK
 - ▼ platform
 - ▼ devices
 - > common
 - > MPC5748G
 - > callbacks.h
 - > devassert.h
 - > device_registers.h
 - > startup.c
 - > startup.h
 - > status.h
 - ▼ drivers
 - > inc
 - > src
 - ▼ rtos
 - ▼ osif
 - > osif_baremetal.c
 - > osif.h



File Structure

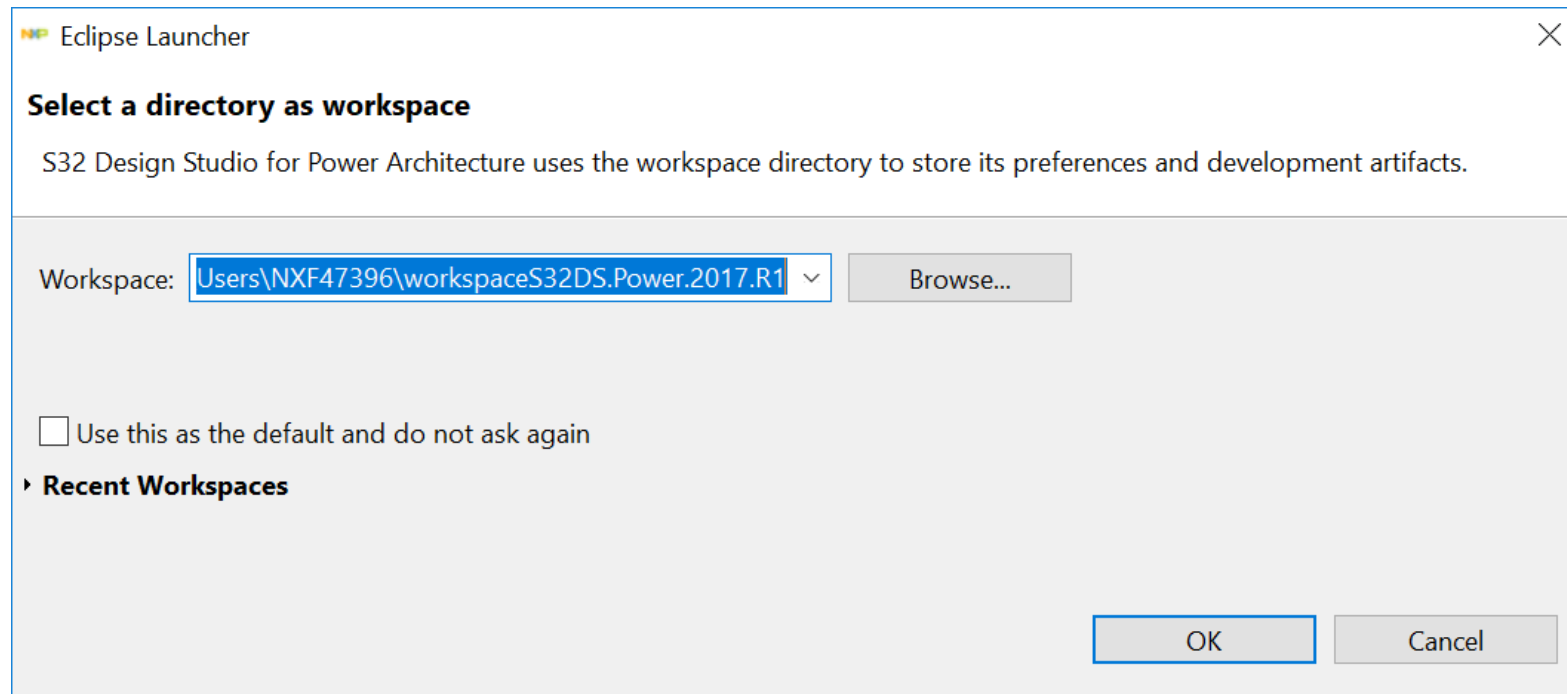
- ▼ SDK
 - ▼ platform
 - > devices ————— Device-specific Files
 - > drivers ————— Peripheral drivers
 - ▼ rtos
 - > osif

CREATE A NEW PROJECT



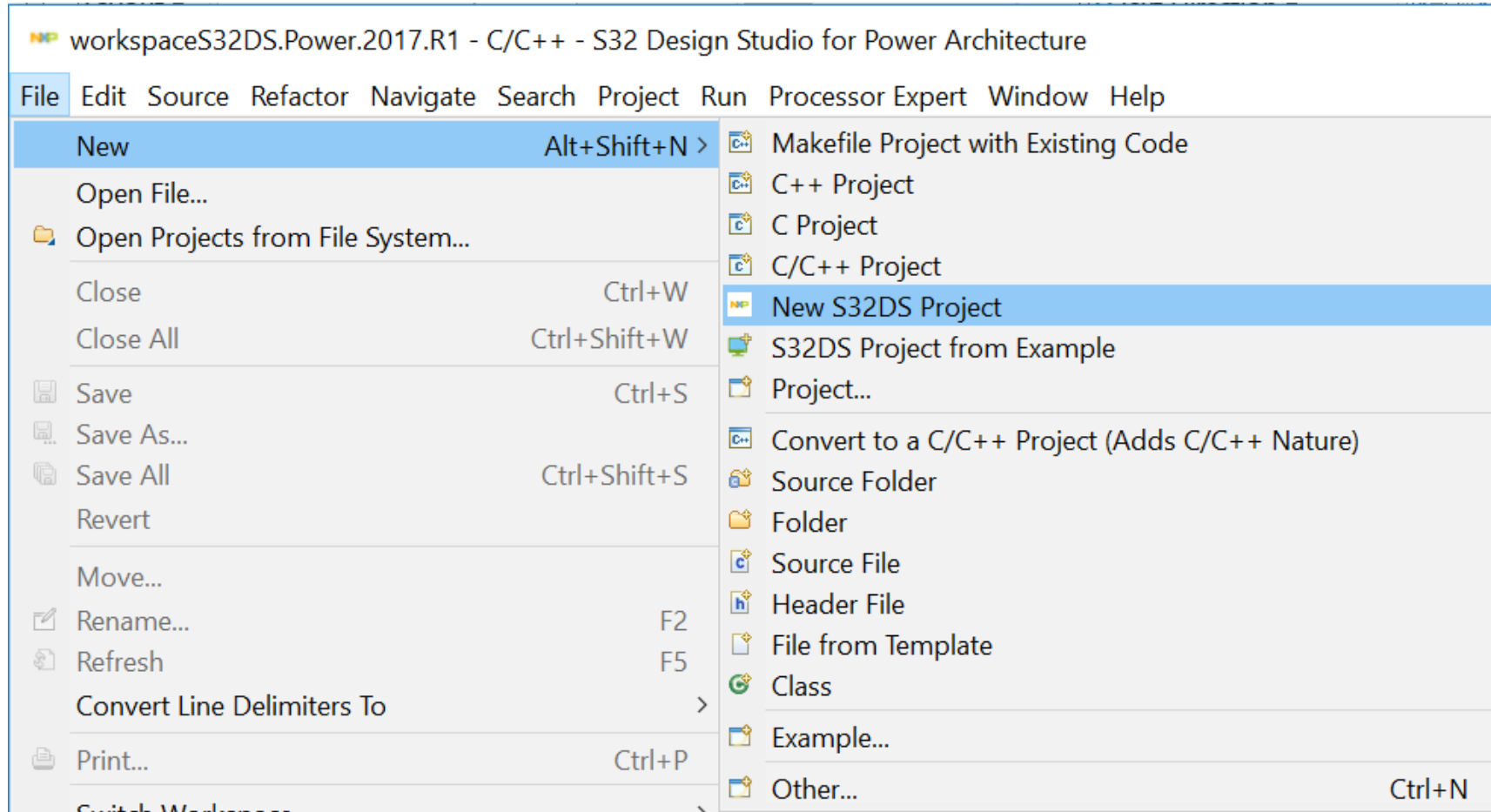
Step-1

- Start program: Click “S32 Design Studio for Power Architecture Version x.x” icon
- Select workspace:
 - Choose default or specify new one
 - Suggestion: Uncheck the box“ Use this as default and do not ask again”
 - Click “ok”



Step-2

- Go to menu bar: File — New — New S32DS Project



Step-3

- Input Project Name:
 - Example: DemoProject
- Select Project Type:
 - Recommended: Use Elf S32DS Project
- Select Controller:
 - Example: MPC5748G
- Click **Next**

New S32DS Project

Create a S32 Design Studio Project

Create New S32DS Project

Project name:
DemoProject

☒ Use default location

Location: C:\Users\NXF47396\workspaceS32DS.Power.2017.R1\DemoProject Browse...

Elf S32DS project Library project

Processors :

type filter text

- Family MPC574xG
 - MPC5746G
 - MPC5747G
 - MPC5748G**
- Family MPC577xK
- Family MPC574xP
- Family MPC574xR
- Familv MPC5777C

ToolChain Selection:

Core Kind	Name	Toolchain
Z4	Boot Z4	Standard S32DS toolchain for E200
Z4	Z4	Standard S32DS toolchain for E200
Z2	Z2	Standard S32DS toolchain for E200

Description :

GCC toolchain is selected

? < Back Next > Finish Cancel

Step-4

- Select Cores
- Select Flash and RAM size
- Select Programming Language
- Select the Library
- Select the Debugger
- Click **Finish**

Recommended: use Default settings
(for beginners)

New S32DS Project for MPC5748G

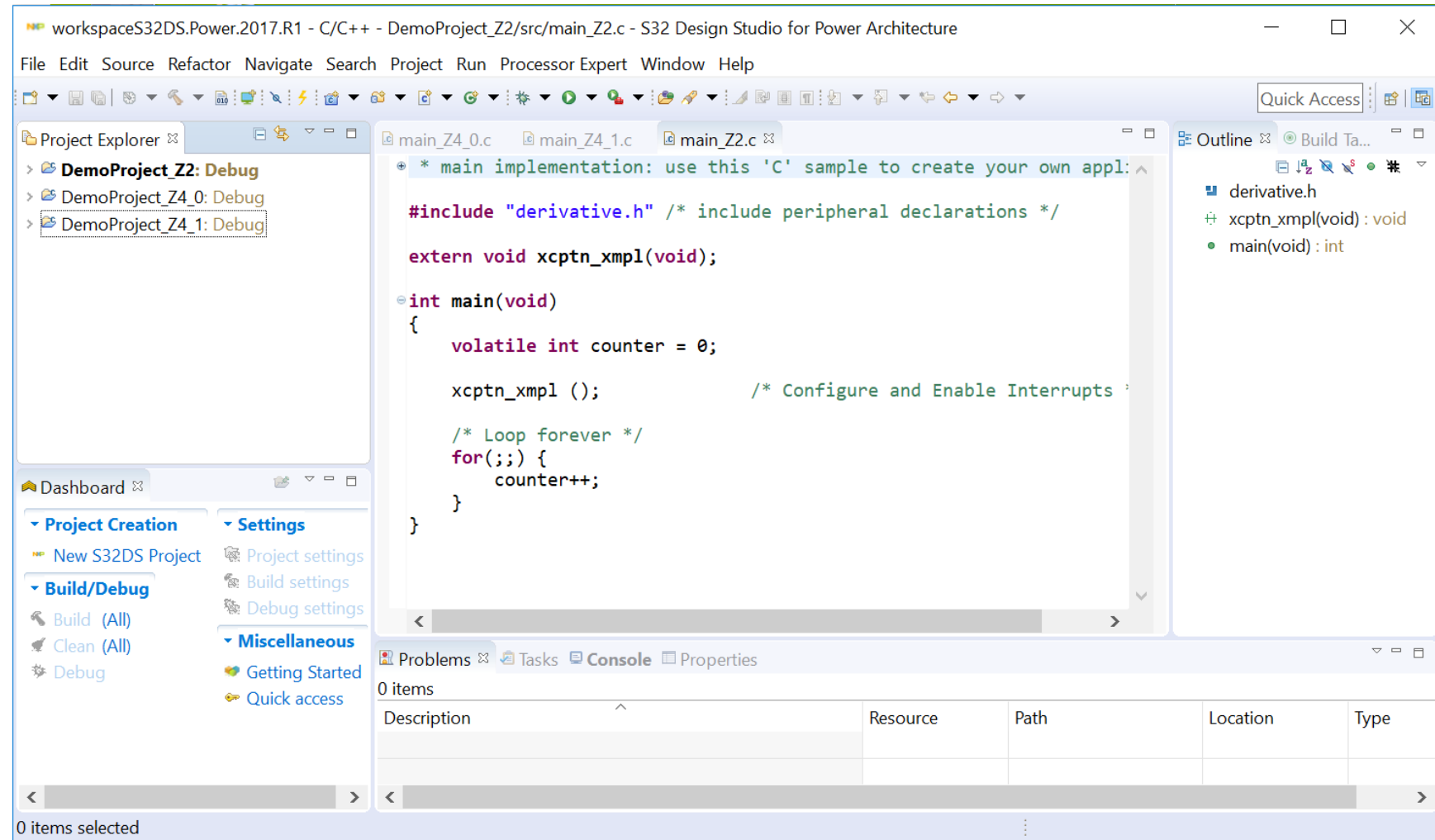
Select required cores and parameters for them.

	DemoProject_Z4_0	DemoProject_Z4_1	DemoProject_Z2
Project Name	DemoProject_Z4_0	DemoProject_Z4_1	DemoProject_Z2
Core	<input checked="" type="checkbox"/> Boot Z4	<input checked="" type="checkbox"/> Z4	<input checked="" type="checkbox"/> Z2
FLASH Start Address	0x1000000	0x11d0000	0x13a0000
FLASH Size, KB	1856	1856	1856
Unused FLASH, KB	64		
RAM Start Address	0x40000000	0x40040000	0x40080000
RAM Size, KB	256	256	256
Unused RAM, KB	0		
Language	C	C	C
Library	EWL	EWL	EWL
SDKs
Debugger	PE Micro GDB server		

Finish

Step-5

- 3 Projects will be created for 3 different cores of MPC5748G

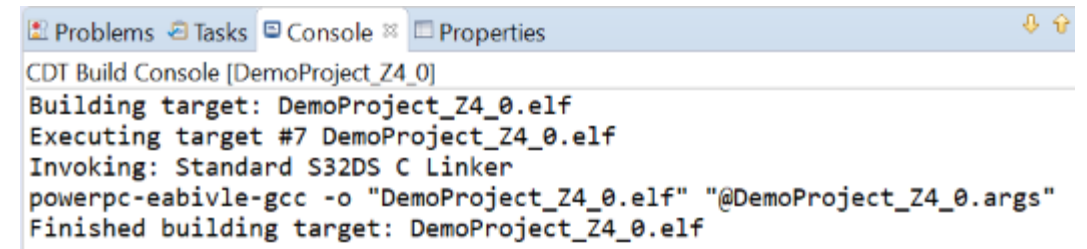
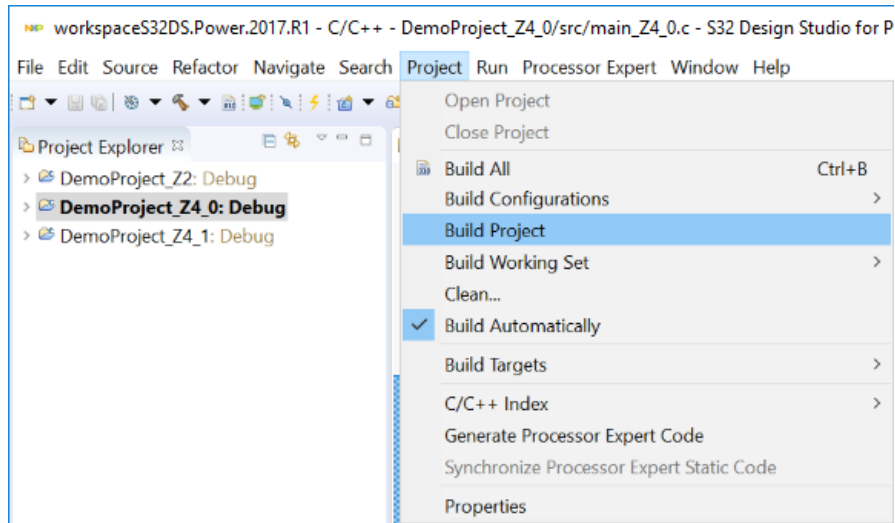


BUILD AND DEBUG PROJECTS




Build a Project

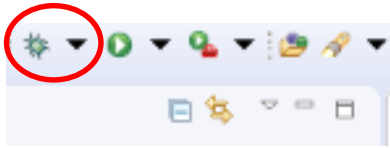
- To build a project follow one of the methods below:
 1. Project — Build Project
- If project is built successfully, following message will be displayed on the **Console** window



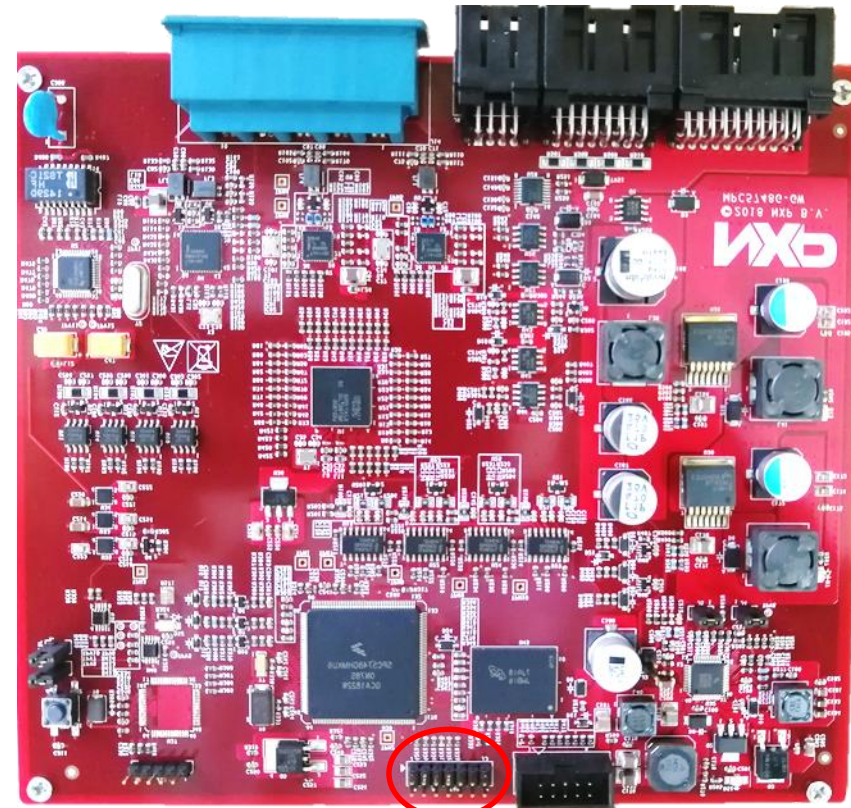
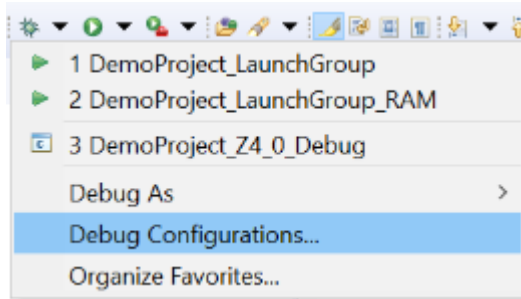
2.  Click on hammer symbol to build that project

Debug a Project

- Connect a **PE Micro** debugger to both, the board and the PC
- Click on arrow in the  icon



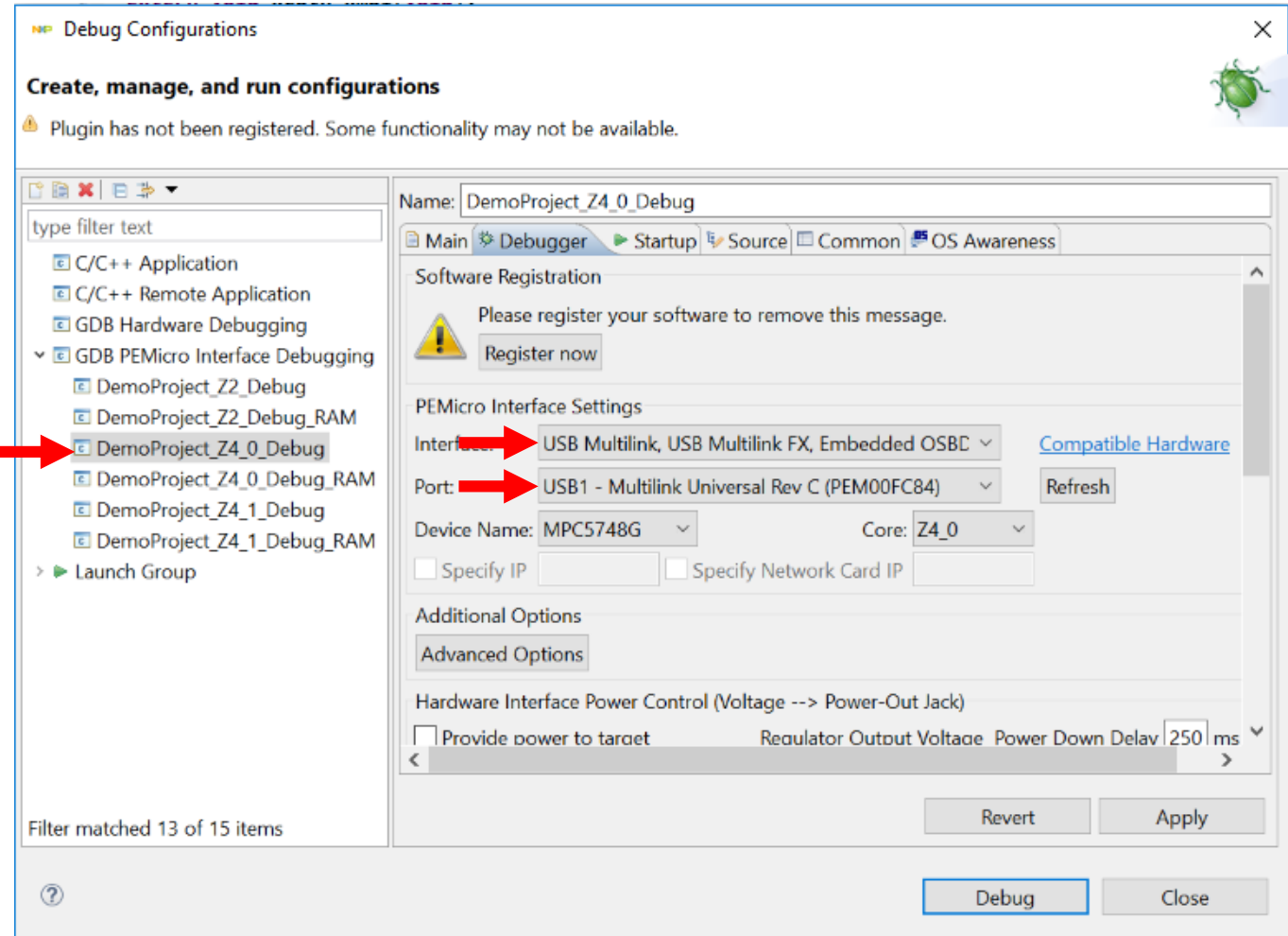
- Then Open **Debug Configurations**




Debugger Interface

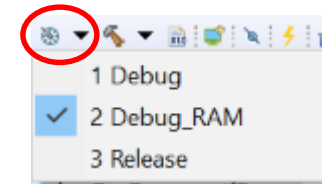
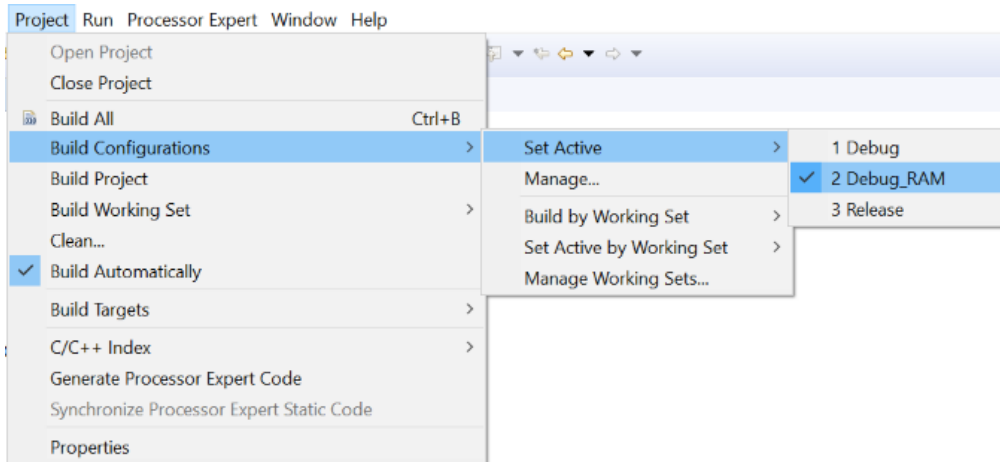
Debug a Project

- Select Project
 - Example:
DemoProject_Z4_0_Debug
- Select Interface:
USB Multilink, USB Multilink FX...
- Select Port:
USB1-Multilink Universal Rev C...
- Click on **Debug** to start debugging



Debug a Project from RAM

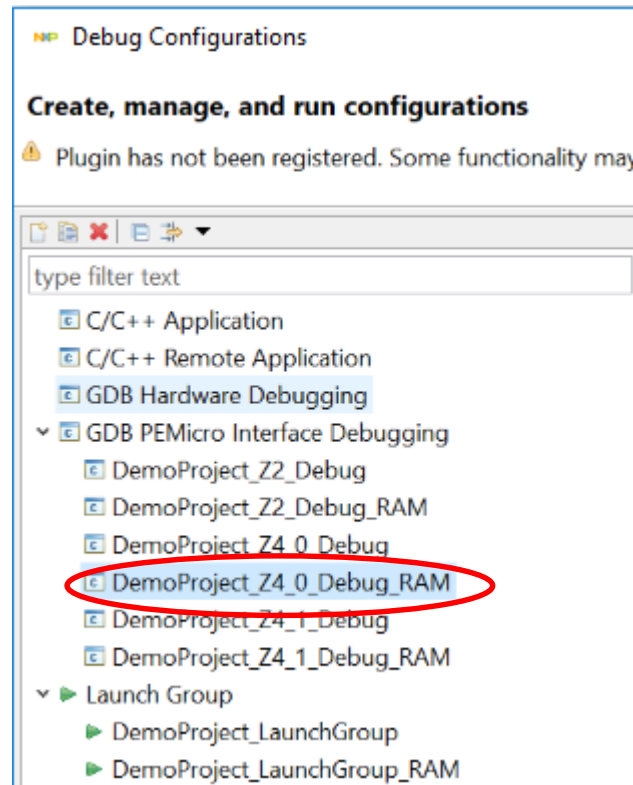
- Step1: Configure a project to debug from RAM, follow one of the methods:
 1. Project — Build Configurations — Set Active — Debug_RAM
 2. Select Debug_RAM by clicking Down Arrow of  icon



- Step2: Follow the steps shown on “[Build a Project](#)” Page

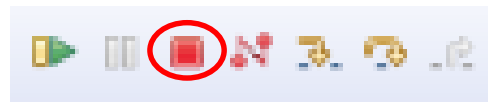
Debug a Project from RAM

- Step3: Follow the steps shown on “[Debug a Project](#)” Page, but need to select the RAM related session while debugging



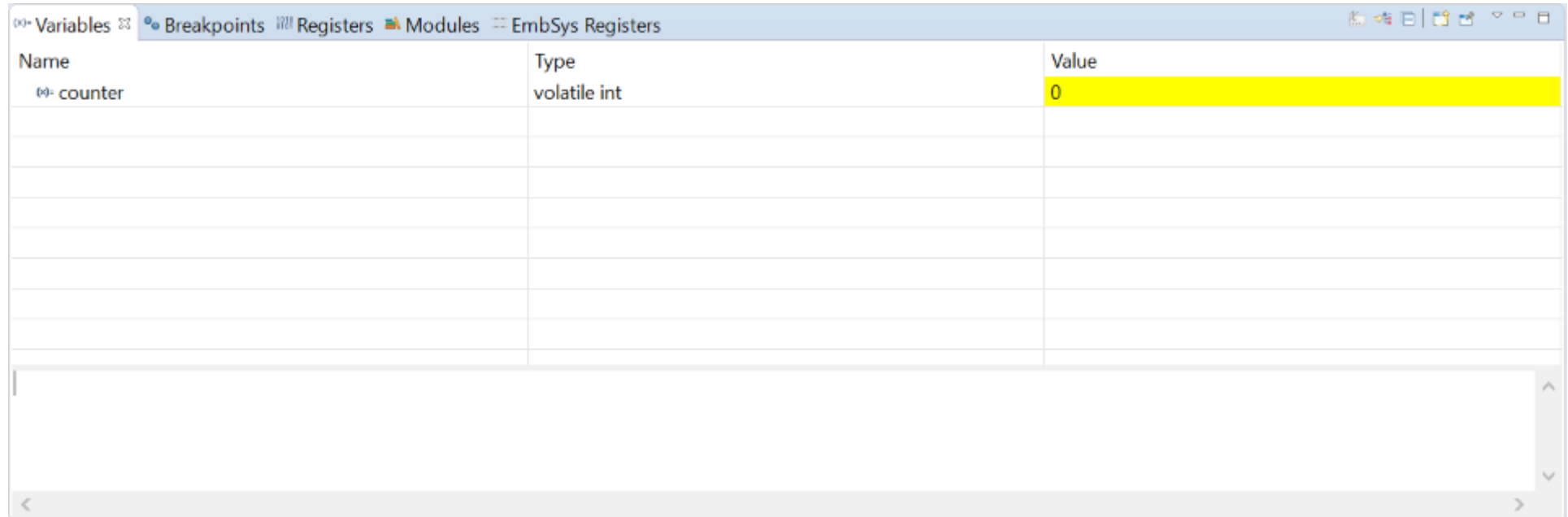
Debug Basics: Step, Run, Suspend, Resume

- Step Into (F5)
- Step Over (F6)
- Run
- Suspend
- Resume (F8)
- Terminate (Ctrl+F2)



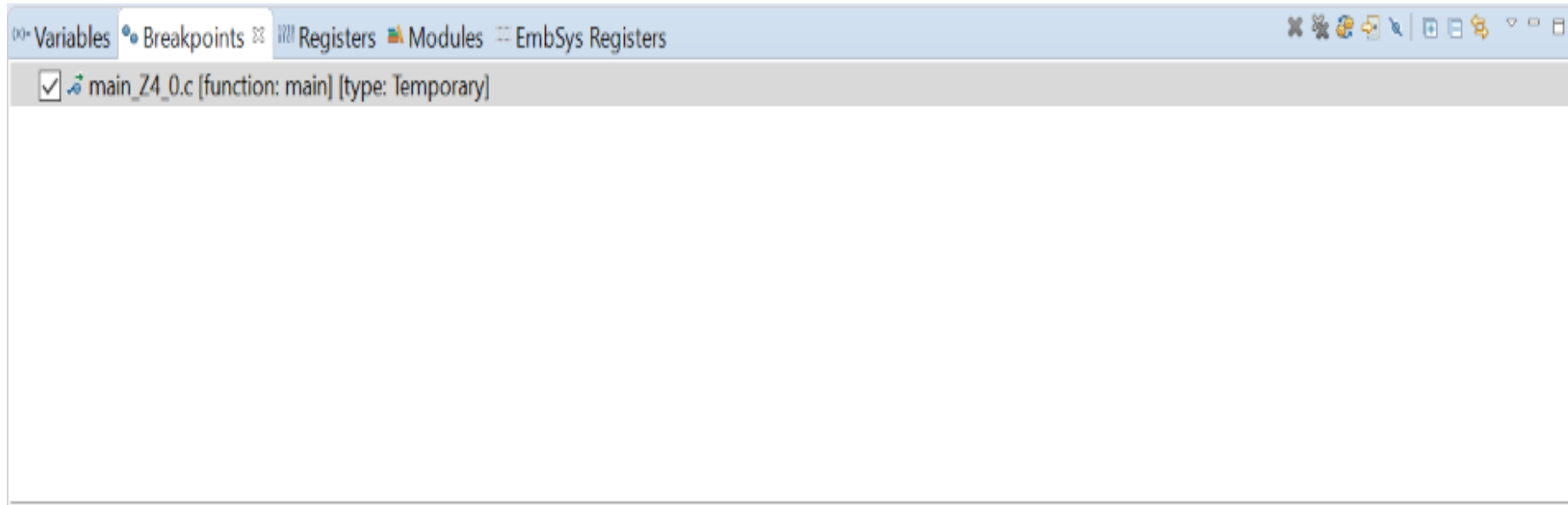
Debug Basics: View & Alter Variables

- View variables in “Variables” tab
- Click on a value to allow typing in a different value



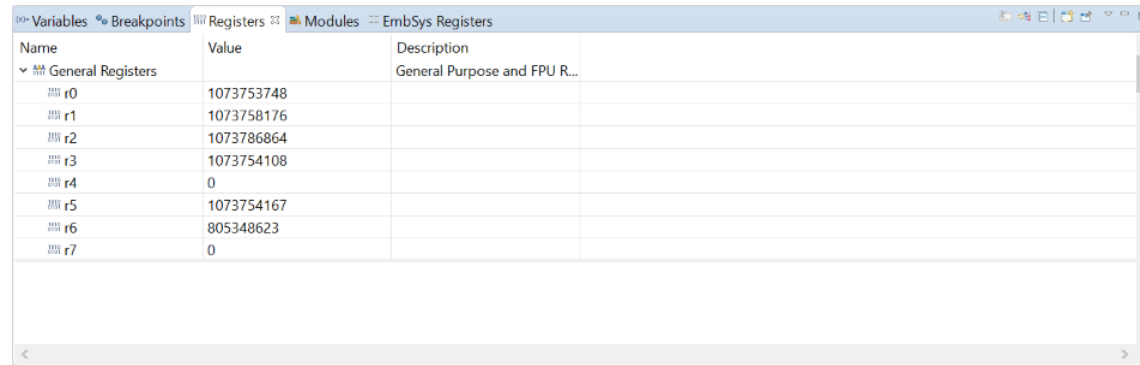
Debug Basics: Disable or Enable Breakpoints(Temporarily)

- When uncheck the breakpoint, it will be temporarily disabled.



Debug Basics: View & Alter Registers

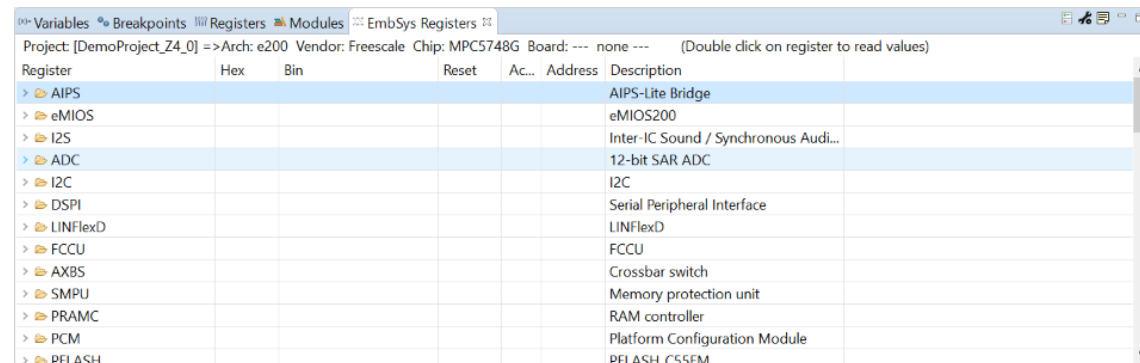
- View CPU registers in the “Registers” tab
- Click on a value to allow typing in a different value



The screenshot shows the 'Registers' tab in a debugger. The 'EmbSys Registers' sub-tab is selected. A table lists general registers r0 through r7. The 'Value' column shows hexadecimal values for each register. The 'Description' column indicates they are 'General Purpose and FPU R...'. The 'r4' register has a value of 0.

Name	Value	Description
General Registers		General Purpose and FPU R...
r0	1073753748	
r1	1073758176	
r2	1073786864	
r3	1073754108	
r4	0	
r5	1073754167	
r6	805348623	
r7	0	


- View peripheral registers in the “EmbSys Registers” tab

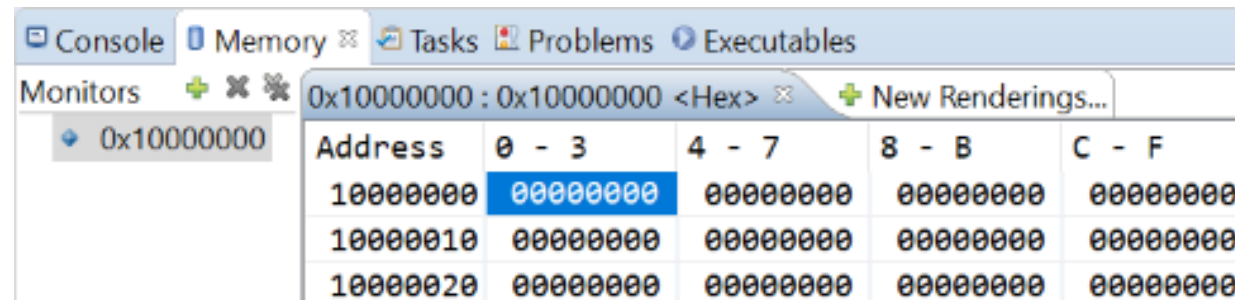
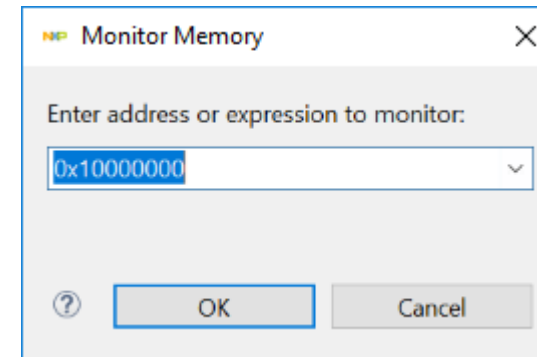


The screenshot shows the 'EmbSys Registers' sub-tab in the debugger. It displays a list of peripheral registers with columns for Register, Hex, Bin, Reset, Ac..., Address, and Description. The 'AIPS' register is highlighted. The project information at the top indicates: Project: [DemoProject_Z4_0] => Arch: e200 Vendor: Freescale Chip: MPC5748G Board: --- none ---.

Register	Hex	Bin	Reset	Ac...	Address	Description
AIPS						AIPS-Lite Bridge
eMIOS						eMIOS200
I2S						Inter-IC Sound / Synchronous Audi...
ADC						12-bit SAR ADC
I2C						I2C
DSPI						Serial Peripheral Interface
LINFlexD						LINFlexD
FCCU						FCCU
AXBS						Crossbar switch
SMPU						Memory protection unit
PRAMC						RAM controller
PCM						Platform Configuration Module
PFIASH						PFIASH C55FM

Debug Basics: View Memory

- Add Memory Monitor
 - Click on  icon
- Enter address to monitor
 - Example: 0x10000000
- View Memory

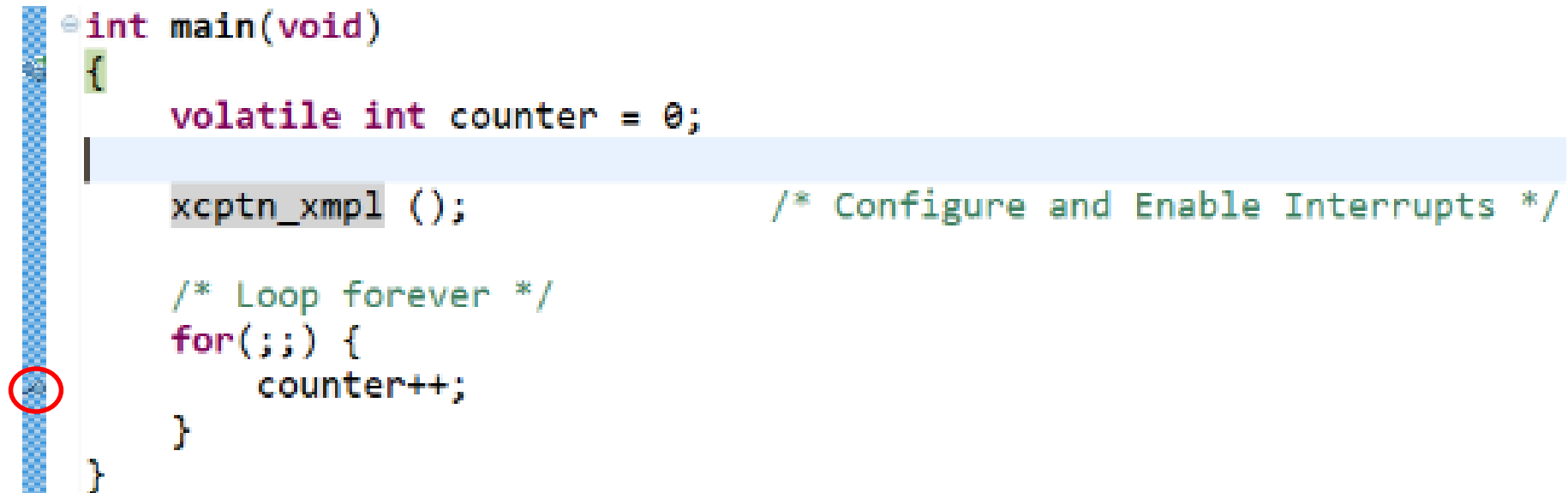


The screenshot shows the 'Memory' tab in a debugger. The 'Monitors' list on the left shows a monitor for address '0x10000000'. The main pane displays the memory view for this address, showing a table of hex values.

Address	0 - 3	4 - 7	8 - B	C - F
10000000	00000000	00000000	00000000	00000000
10000010	00000000	00000000	00000000	00000000
10000020	00000000	00000000	00000000	00000000

Debug Basics: Breakpoint

- Add Breakpoint: Point mouse pointer at circled area and **Double Click** there
 - Light blue dot will pop up that represents debugger breakpoint



```
int main(void)
{
    volatile int counter = 0;

    xcptn_xmpl ();                /* Configure and Enable Interrupts */

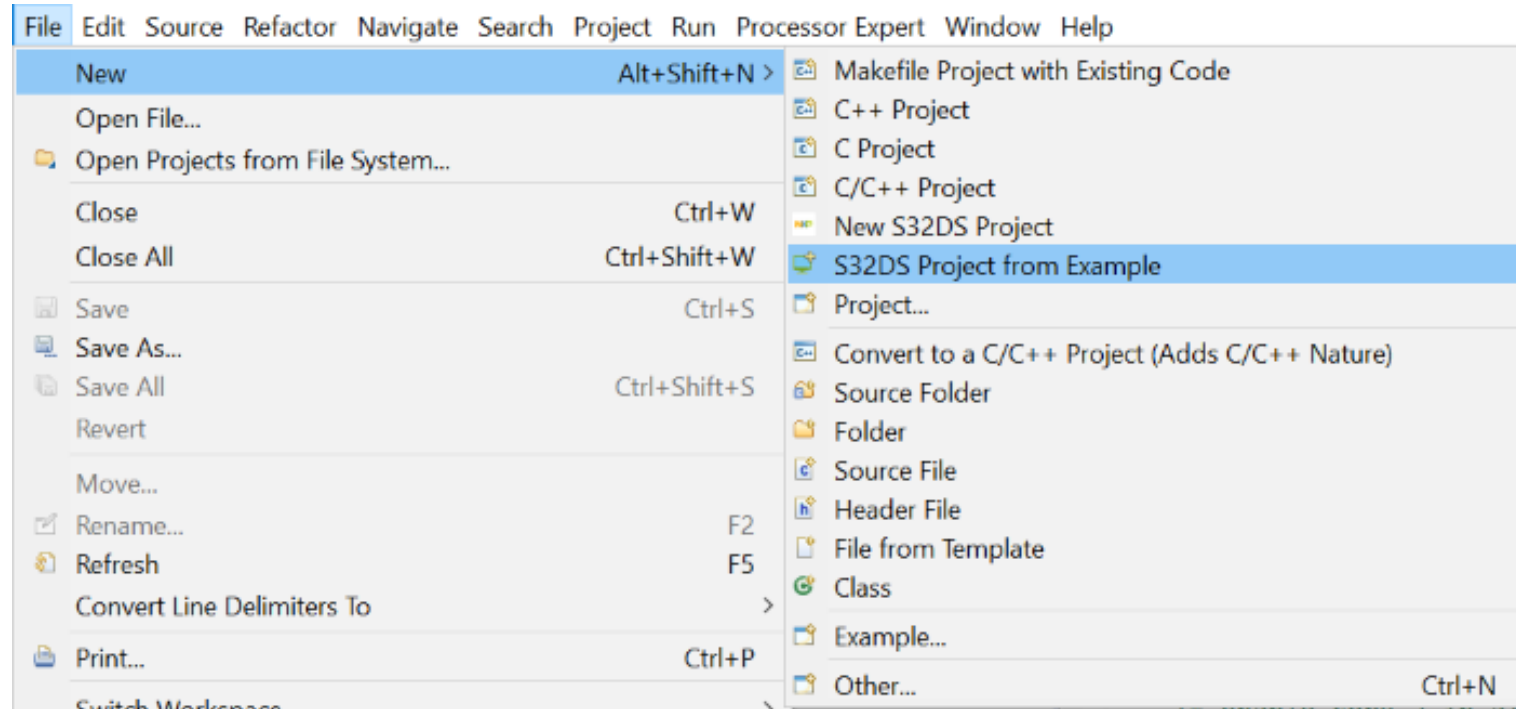
    /* Loop forever */
    for(;;) {
        counter++;
    }
}
```

The image shows a code editor window with a C program. A red circle highlights a small icon in the left margin, next to the line containing the 'for' loop, indicating where a breakpoint has been set. The code includes a main function, a volatile integer counter, a function call xcptn_xmpl, and an infinite loop that increments the counter.

MAKING PROJECTS FROM BUILT-IN EXAMPLE

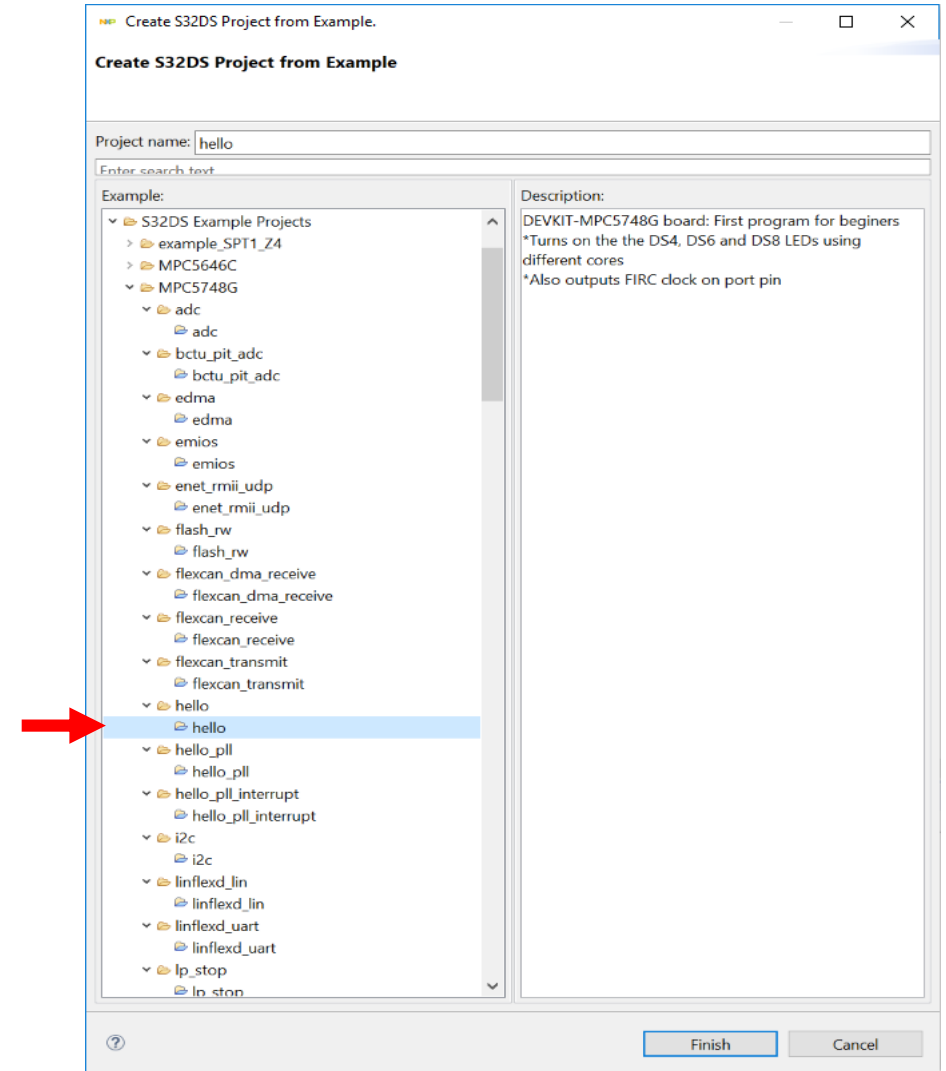
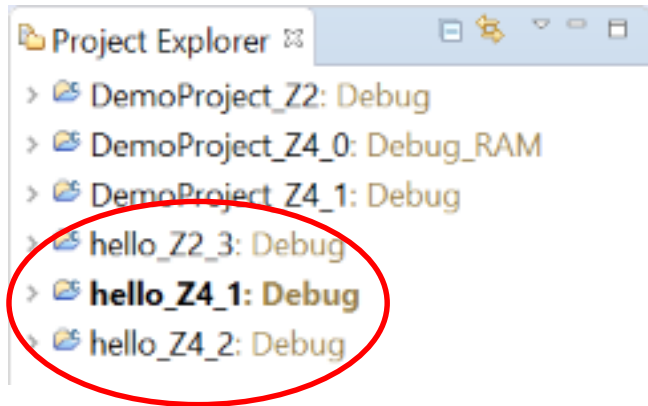
Step-1

- Go to menu bar: File — New — S32DS Project from Example



Step-2

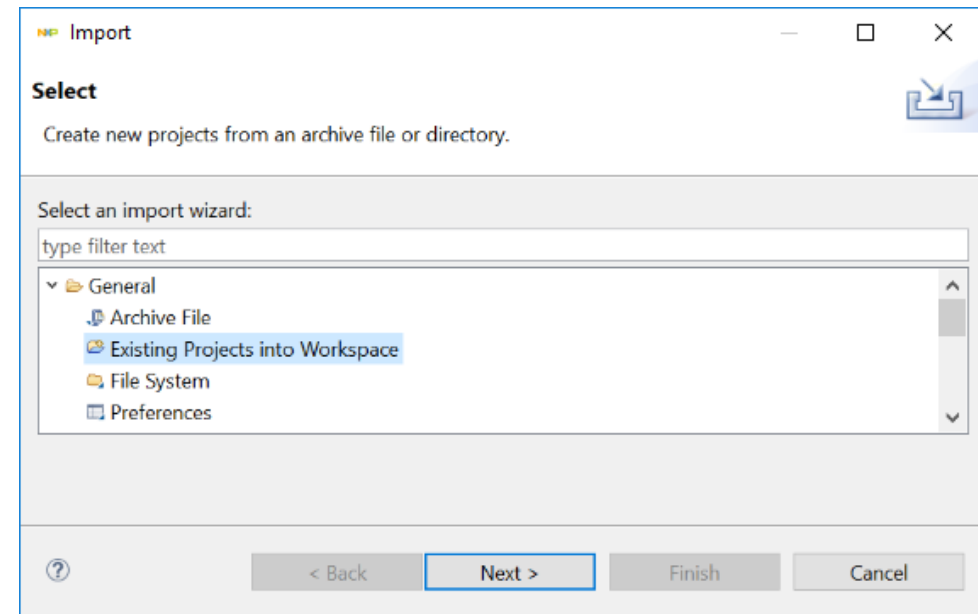
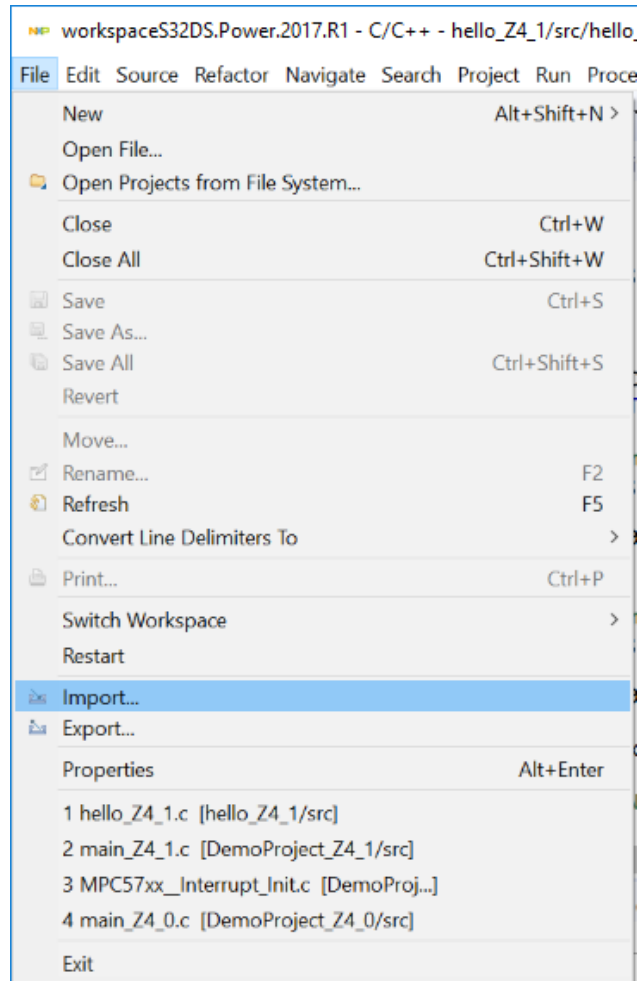
- Select the built-in project of your choice
- Click on “**Finish**”
- Project will be copied to the active workspace as shown below



IMPORTING PROJECTS

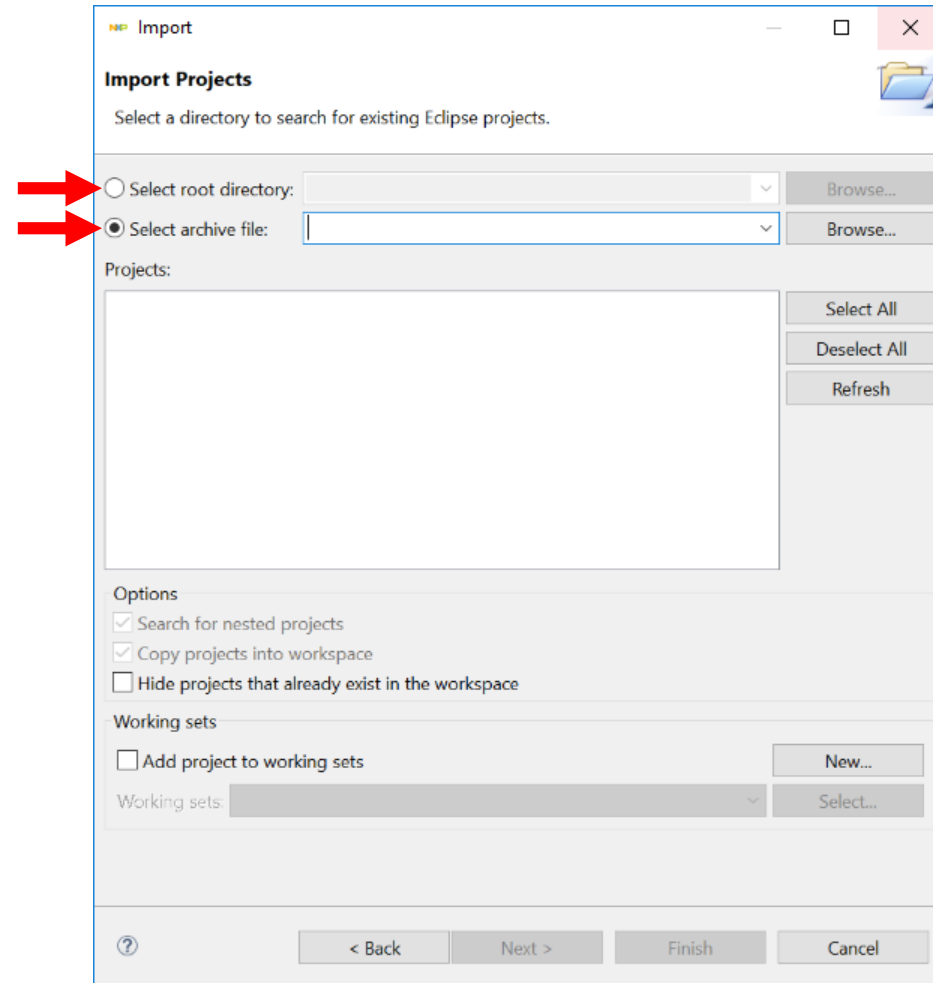
Step-1

- Go to menu bar: File — Import
- Click on: “Existing Projects into Workspace”
— Click “Next”



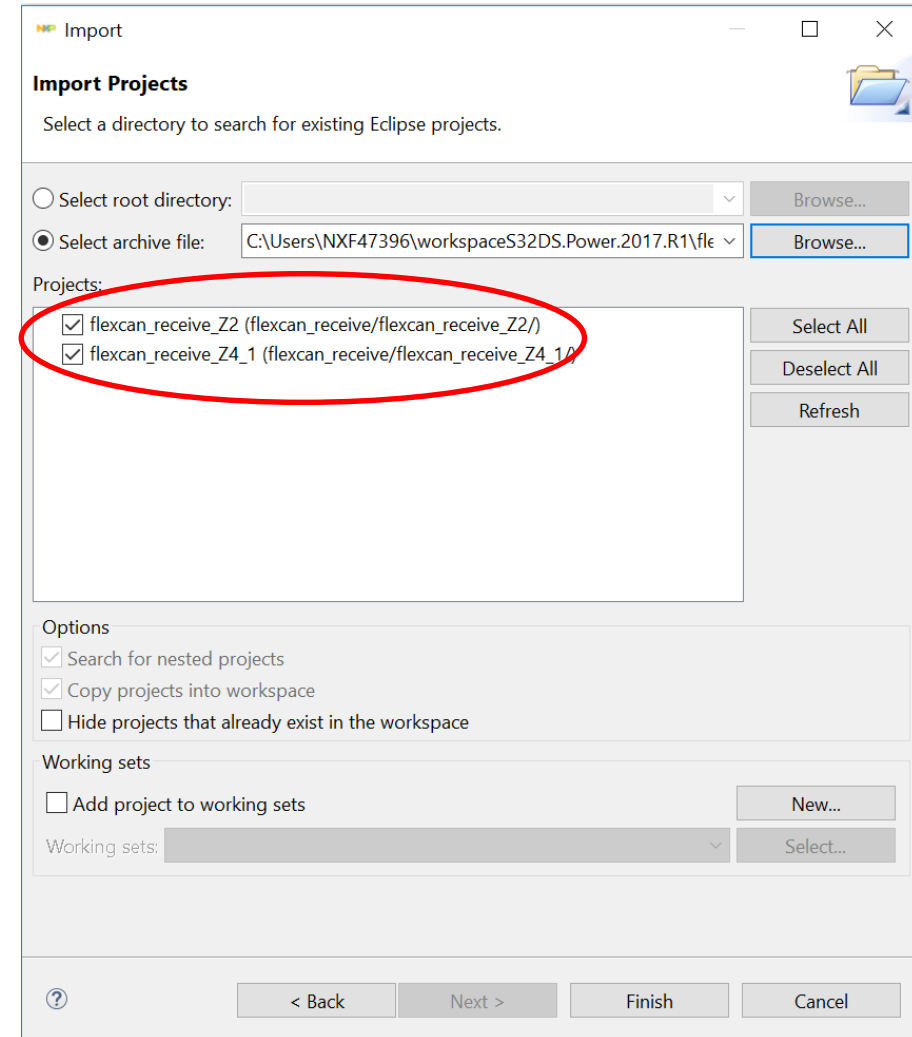
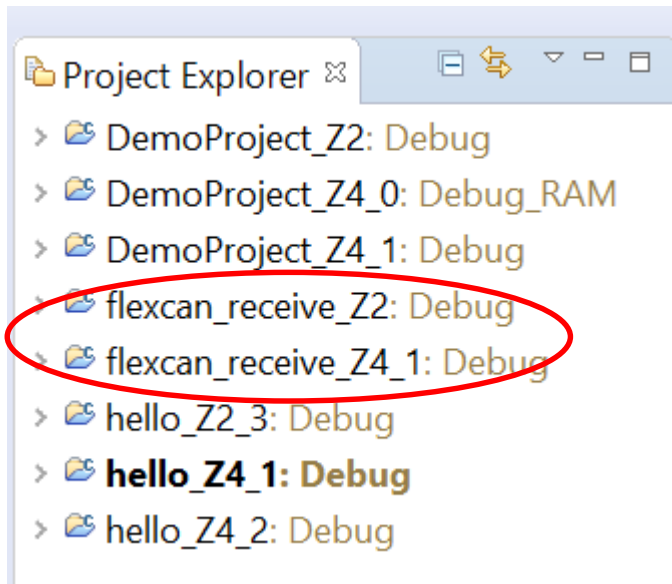
Step-2

- Click on: Browse & Select Example Folder



Step-3

- Select the project
- Click on “**Finish**” to Import a Project into Workspace





SECURE CONNECTIONS
FOR A SMARTER WORLD