CMPE 240

Feb 21 Sun

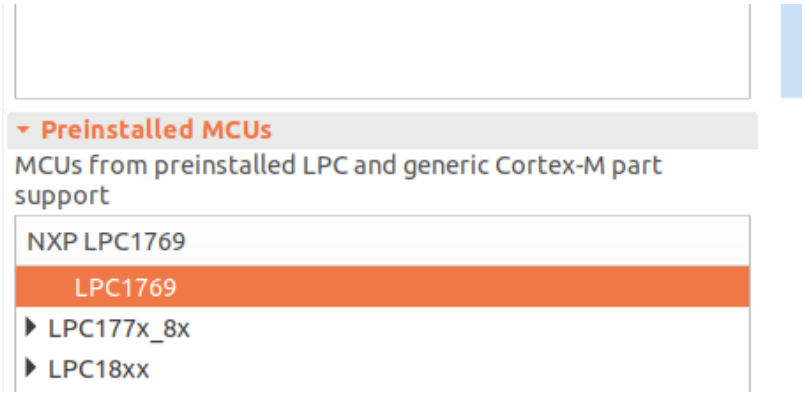
LPC1768/1769 - ARM Cortex M3 Development - MCUXpresso IDE Installation And Adding A Template Project

https://copperhilltech.com/blog/lpc17681769-arm-cortex-m3-development-mcuxpresso-ide-installation-and-adding-a-template-project/

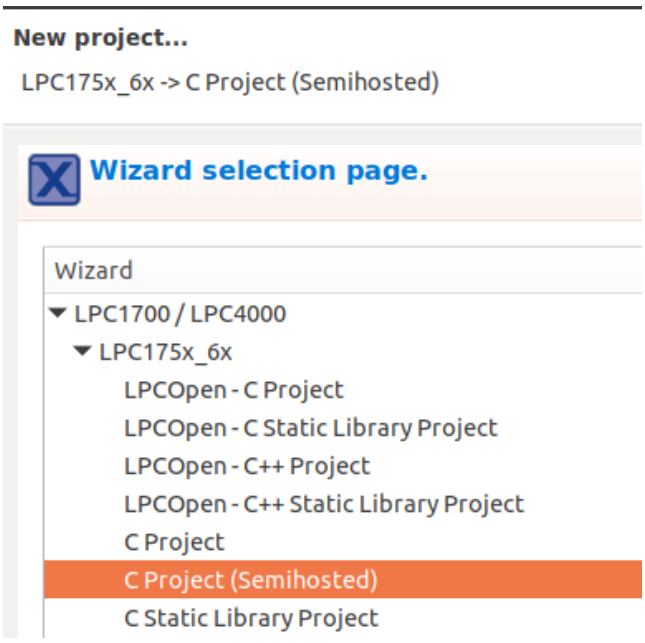
1. After creating MCUXpresso projects with CMSIS, the IDE will automatically add all the necessary startup files(for initializing MCU), headers and a project source file(C/C++) depending on the settings we choose while creating the project.

2. workspace
Please be aware that the IDE allows you to create several workspaces and to switch between them. In fact, we recommend to use this feature to separate projects

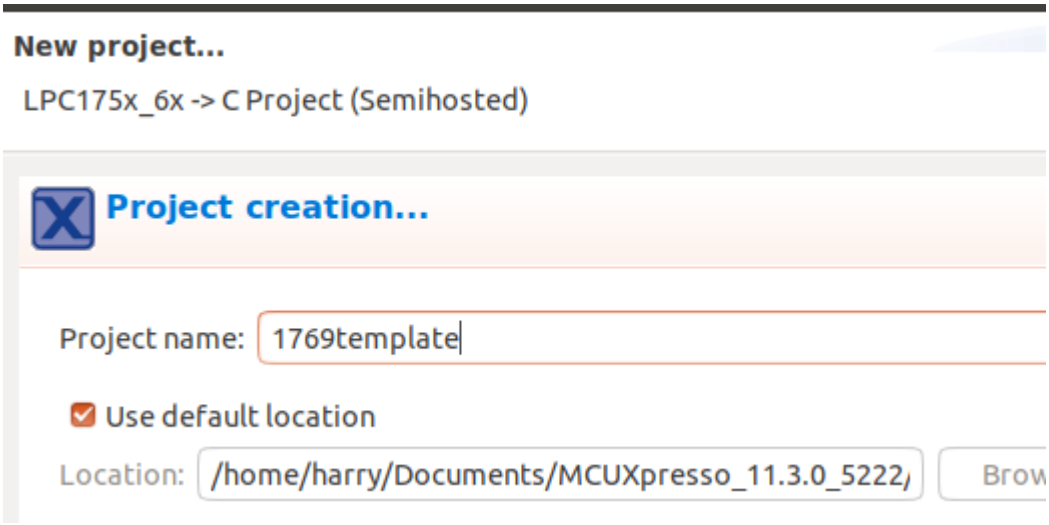
(1) from quick start panel, click to start new project.
(2) from pre-installed MCUs, look for 1769, double click the photo of the board on the right side of the panel on it.



(3) select C (semihost project) when double click on the photo of the board,



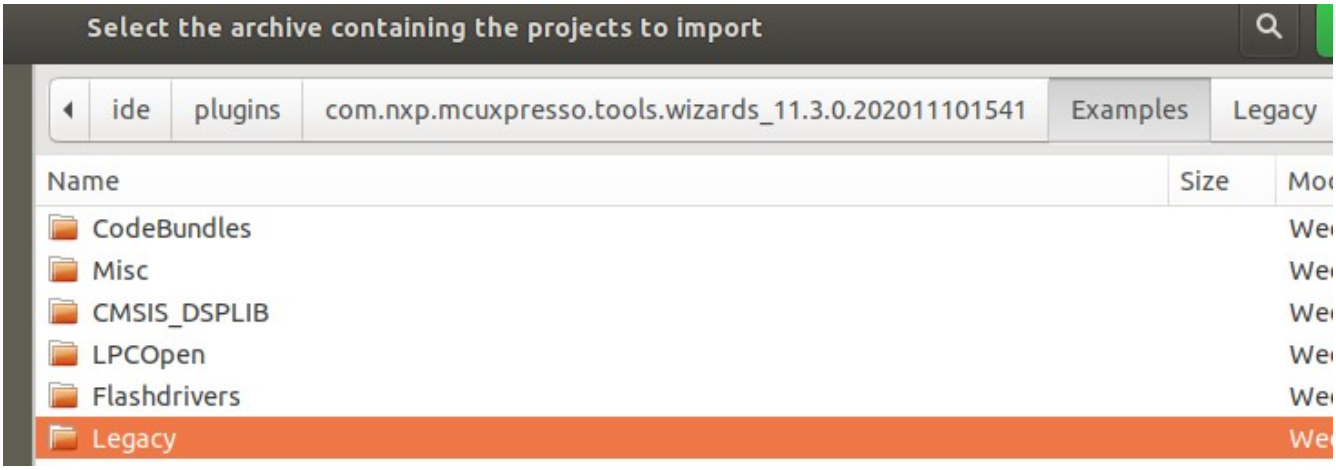
(4) give project name by clicking the next button on the popup window above



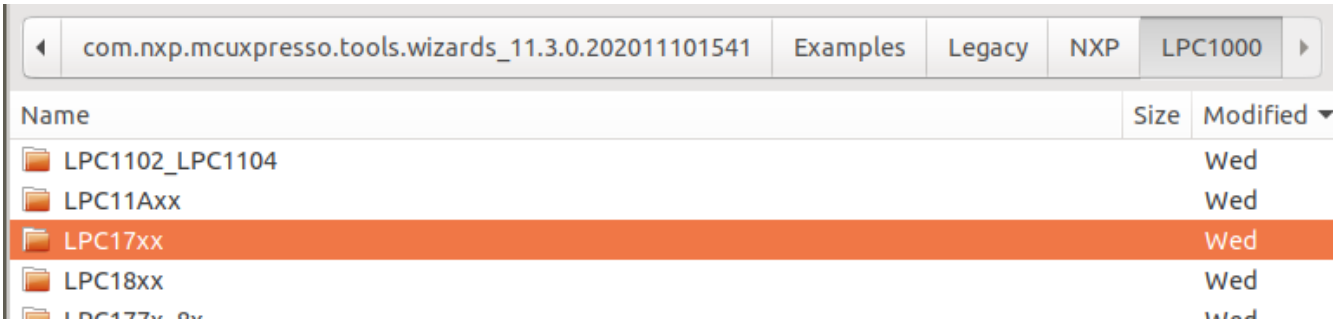
(5) then at the popup window to select import the CMSIS_CORE,

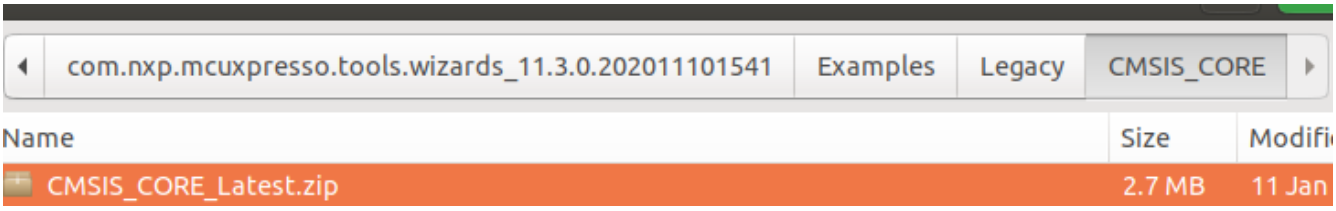
See the new project window popup:

At the drop down menu, then click on “import”, a popup window will appear

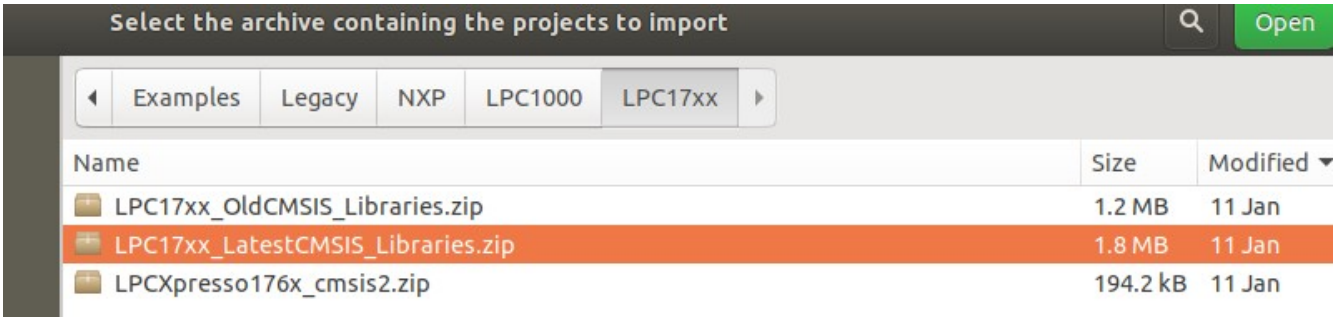


Then click on Legacy, then enter the Legacy folder, then click on Legacy > NXP > LPC1000 > LPC17xx

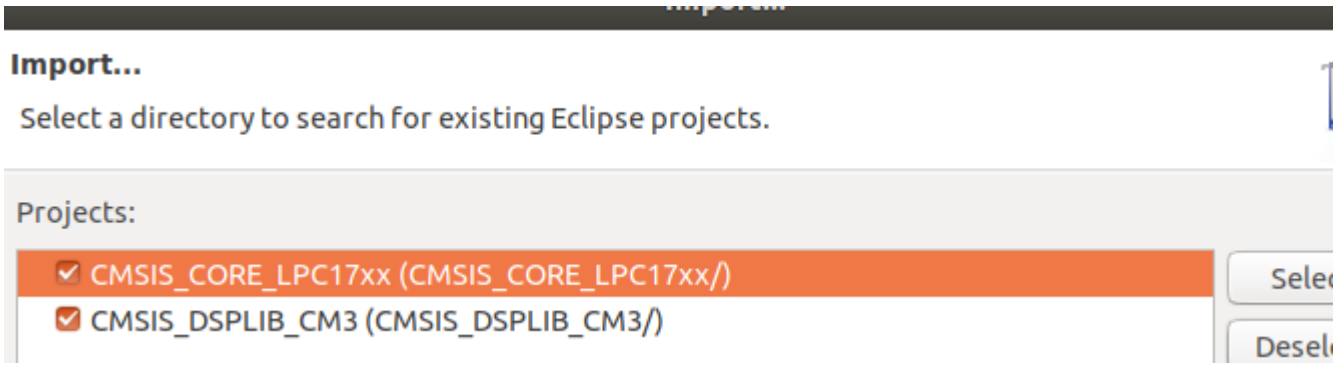




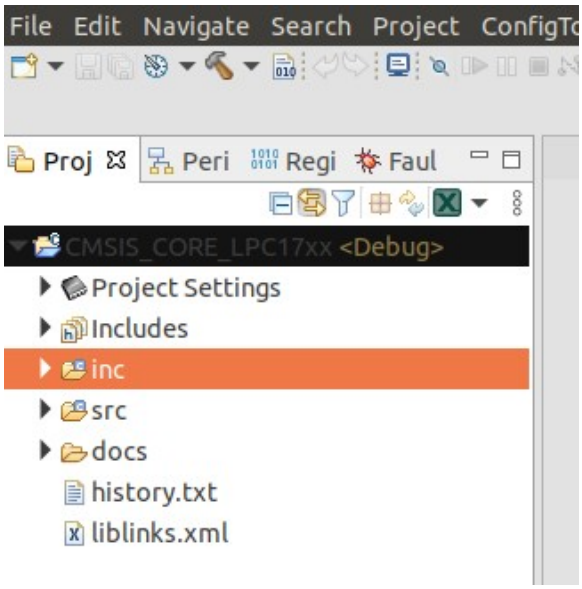
Then select latest for LPC17xx



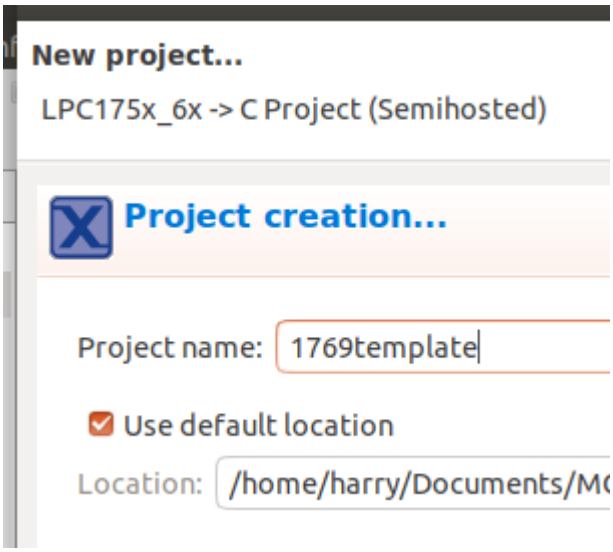
then click NEXT, at the popup window select CORE… as below



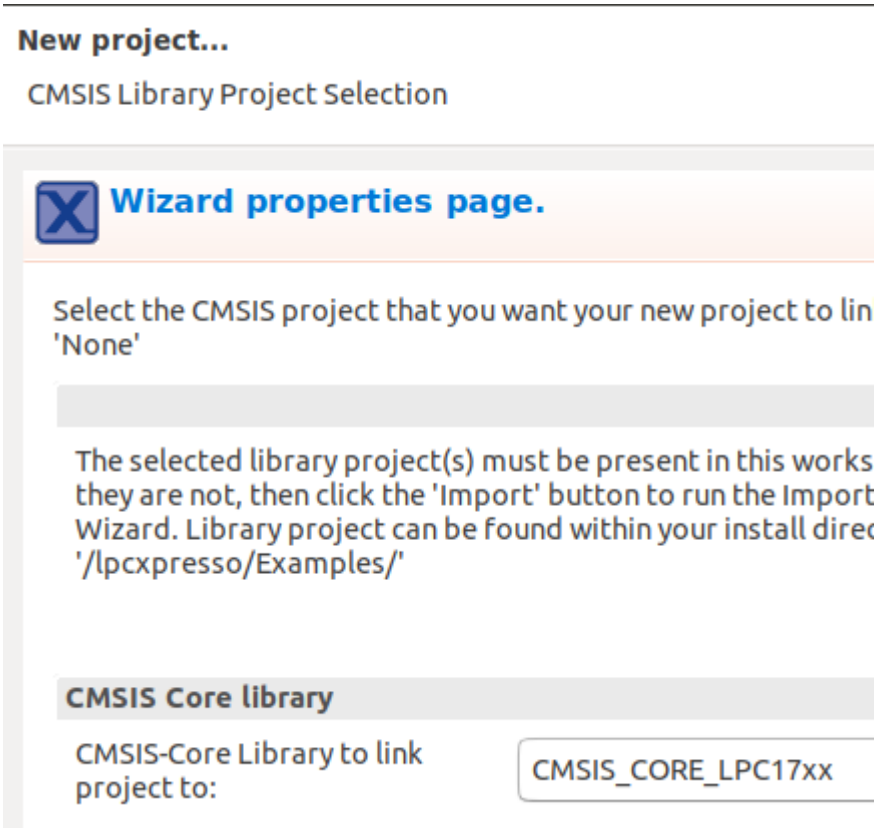
Note Deselect the DSPLIB, then proceed by click FINISH. So you will have the CMSIS_CORE installed as (this part is different from the web tutorial)



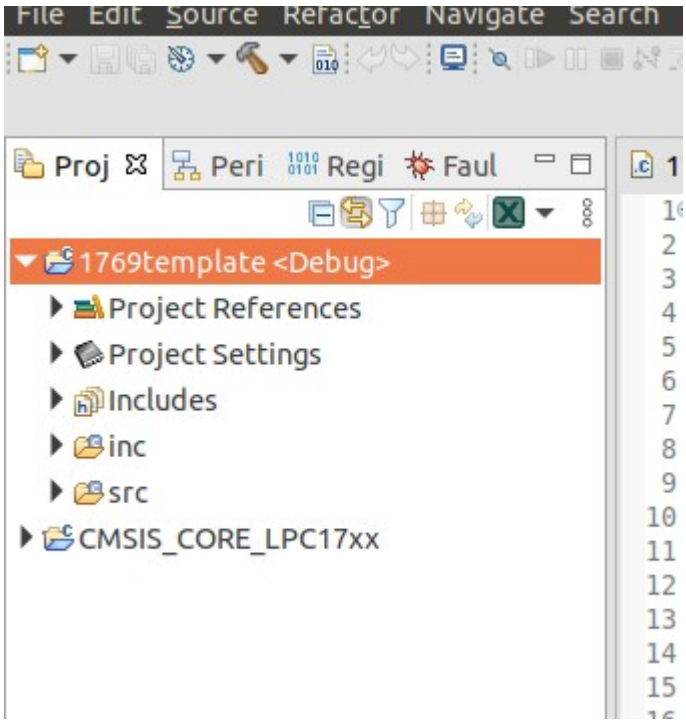
Now, to create your project …
go back to create project, double click on it, at the popup window, enter your preferred project name, in my case 1769template



then click next, the popup window shows “New project … CMSIS Library Project Selection”, click Next, and at the popup window click Next again

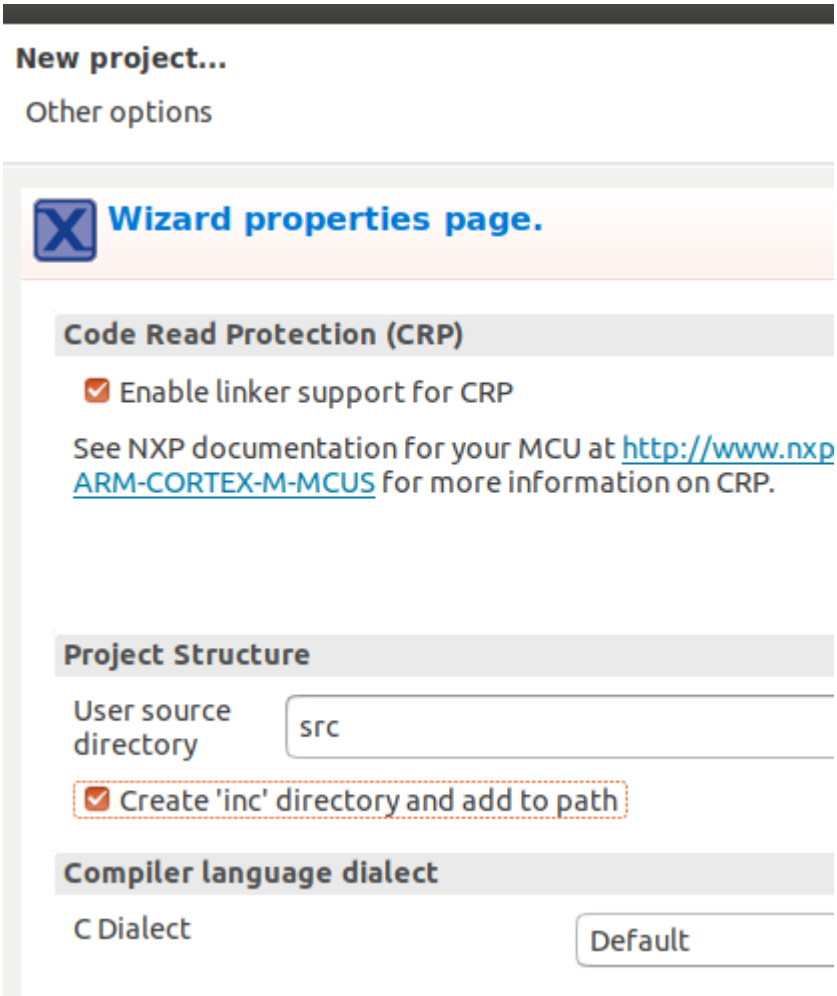


The CMSIS Core library should be shown on the popup window, and the next popup window the DSP LIB should show NO. After click Next, then at the popup window, then select “inc” option, then click NEXT , so your 1769template project is created



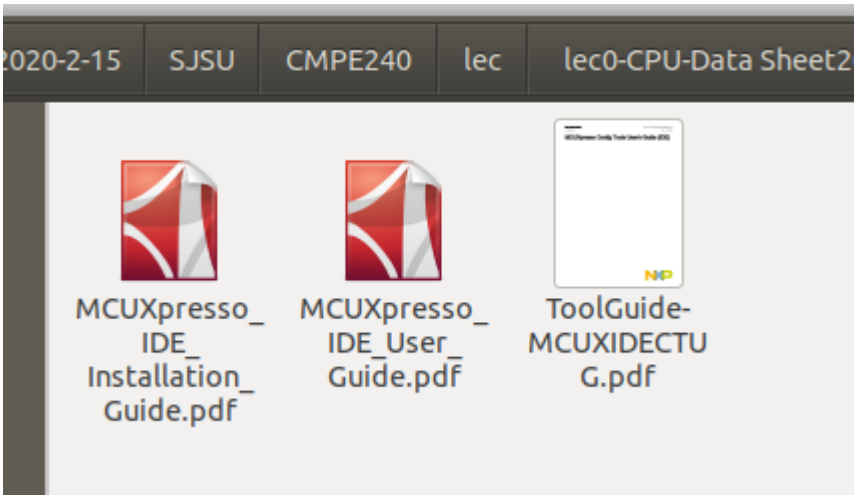
The “hello, the world” example

https://copperhilltech.com/blog/lpc17681769-arm-cortex-m3-development-the-hello-world-application/



Feb 20 Sat

1. Reference document for MCUXpresso



harry@workstation:/media/harry/easystore/backup-2020-2-15/SJSU/CMPE240/lec/lec0-CPU-Data Sheet2015/mcu-xpresso$ tree -L 1

```
.
├── MCUXpresso_IDE_Installation_Guide.pdf
├── MCUXpresso_IDE_User_Guide.pdf
└── ToolGuide-MCUXIDECTUG.pdf
```

0 directories, 3 files

Note:
1.
• Linux – Ubuntu 18.04 LTS and 20.04 LTS
• Only 64-bit versions of Linux are supported. pp. 3, Installation Guide.

2.
MCUXpresso IDE functions and features are under continuous development, it is strongly recommended for user to read both the ReadMe and KnownIssues files inside the product installation directory. pp. 6, installation guide

where is product installation directory?

3.
When using the
(1) new version, use a new workspace – to keep your new “evaluation” world separate from your old “development” world. You can very easily
(2) copy your projects from your existing workspace to your new workspace – for example by checking them out of version control again, or simply using the
(3) IDE’s Quickstart Panel option to “Import project(s) from file system...” and pointing at the root directory of your existing workspace.

4.
MCUXpresso IDE v11.2.x (or later) projects are not backward compatible with earlier versions of MCUXpresso IDE. Also, if an existing project is edited with MCUXpresso IDE v11.2.x, it may no longer be usable with an earlier version of MCUXpresso IDE. pp. 6

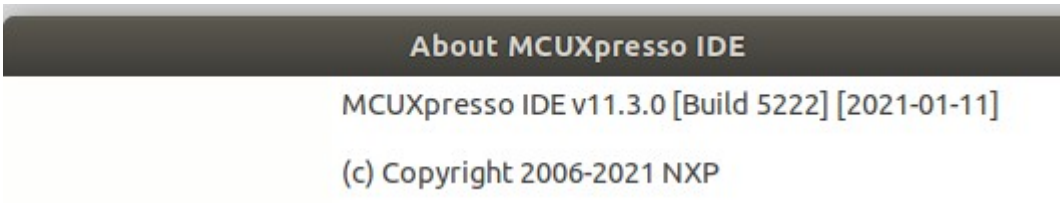
5.
Preferences can also be imported from your existing workspace to your new workspace. To do this run your old IDE installation, and use File -> Export -> General -> Preferences to export. Then in your new IDE installation use File -> Import -> General -> Preferences to pull your preferences in.

6.
If you have installed additional plugins into your original IDE installation, then you can also import these. To do this, from your new installation select File -> Import -> Install -> From Existing Installation and point at the ide directory within your original IDE’s installation directory. Note: that on macOS / Linux – this option will effectively run automatically the first time you run the new version of the IDE. pp. 6

7.
MCUXpresso IDE version 10.2.0 (or later) allows projects to be imported by simply dragging a project folder (or zip archive of projects) directly into the IDE’s (1) Project Explorer view. In addition, it is possible to drag from the Project Explorer view of an older IDE, directly into Project Explorer view of MCUXpresso IDE version 10.2.0. This provides a very simple way of transferring projects into the new IDE. However,

it is recommended that
(2) build configuration and
(3) launch configuration folders
are deleted before (or after) copying. Note: Due to enhancements from MCUXpresso IDE version 10.2, older launch configurations are no longer compatible with this version, failure to delete them will lead to a warning and the launch configuration will then be deleted automatically on the next debug attempt. pp. 6

Note my version



8.
4. Appendix B – Migrating from LPCXpresso IDE version
8.2.x – Hints and Tips pp. 9

9. Version control capability and

Installing Eclipse Plugins
If you install a new version of MCUXpresso IDE on Mac OS X or Linux, then the first time you run the new product you will be offered the opportunity to reinstall previously used plugins (for example, those for version control).

pp. 9

10. Import workspace to the new installation (newer version)
The simplest way to do this is to create a new workspace in the new MCUXpresso IDE version, and then import any projects into this new workspace. How to import projects into a new workspace is detailed in the FAQ
https://community.nxp.com/message/630625

11. Import/export projects
https://community.nxp.com/t5/LPCXpresso-IDE-FAQs/How-to-import-export-projects/m-p/470495

Import/Export projects

importing projects

To import an archive containing one or more projects into the workspace, including the Examples provided with the LPCXpresso IDE:

- To Import projects.

- In the Quickstart Panel, click on Import projects
- If Importing from a zip file, press the Archive Browse button
    - Browse to the zip file, and press OK
- If you are importing projects from a directory, press the Root directory Browse button
    - Browse to the directory containing the projects, and press OK
- A list of importable projects contained in the directory is displayed
- Check/uncheck projects from the list as necessary and Finish

The projects will now be imported.

Labels

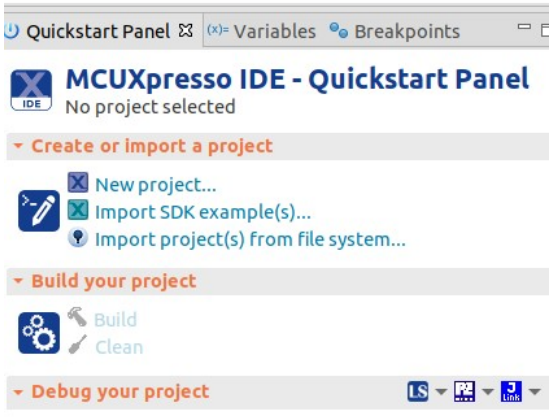
- **IDE usage and settings**

- Tags:
- lpcxpresso
- projects

- All forum topics
- Previous Topic
- Next Topic

0 Replies

My quick start panel



Exporting projects

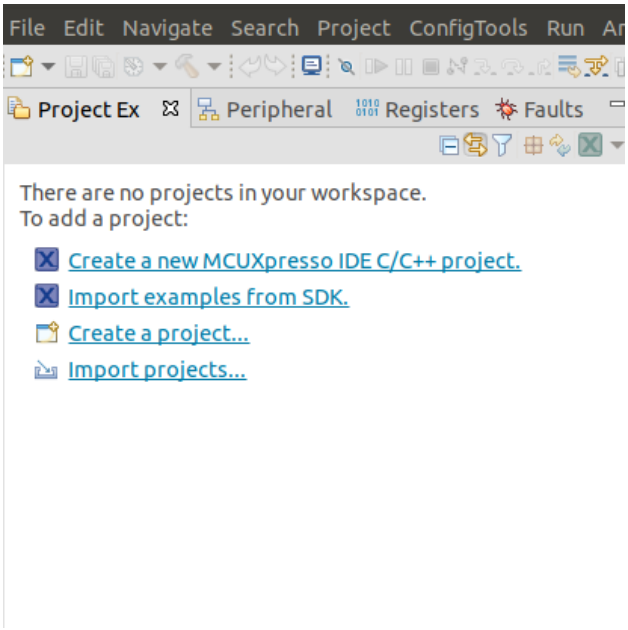
We recommend that before exporting a project, you perform a "Clean" build to remove binary executable and object files from the build. These are not normally required, and this will make the exported archive significantly smaller. (HL: but not for my students or my engineers submission)

To export a project:

- Select the project (or projects) in the Project explorer
    - Use Ctrl-Click to select more than one project
- In the Quickstart Panel, click on:
    - Click on Export project to archive (zip) or
    - Export project and references to archive (zip)
- Enter or Browse the zip file name and then Finish

Your project is now exported

Note project explore panel



12.
Version Control with Eclips
version control system (for example, using
Subversion and the Subclipe Eclipse plugin), then you can simply check your projects out into the new workspace. pp. 9

13.
Do full clean build
do a full, clean build after switching to the new version

14.
Deleting debug launch configuration

deleting any debug launch configurations within your
project that were created by an earlier version. These files are typically named
<projectname> Debug.launch and <projectname> Release.launch.
The easiest way to do this is to right-click on the project in Project Explorer and select Launch Configurations -> Delete Launch Configurations. The IDE will then automatically create a fresh set of launch configurations the next time you start a debug session. Note that you may need to reapply any modifications you made to your launch configurations in your previous version of MCUXpresso IDE.

pp. 13

15. Launch configuration file
https://community.nxp.com/t5/LPCXpresso-IDE-FAQs/Launch-Configuration-Menu/m-p/472277

Within each LPCXpresso project, there will be a "launch configuration file" for each build variant, which is used to store the settings for a debug connection for that build configuration.

These will be created in the root directory of a project the first time that you launch a debug session for that project and are typically called "**<projname> Debug.launch**" and "**<projname> Release.launch**".

pp. 13

"Launch Configurations" entry on the context sensitive menu available from the Project Explorer view.

16.
Start up Code

17.
Linker Scripting

pp. 13

18.
SDU installed MCU is different from other mean installed MCU.

19.
SPIFI Flash Drivers for LPC18xx and LPC43xx

20.
MCUXpresso IDE requires no activation procedure and uses no licenses
pp. 14

MCUXpresso IDE User Guide

Rev. 11.3.0 — 10 January, 2021 User guide

Note:
1.
Change workspace

You may change the Workspace that MCUXpresso IDE uses, via the File -> Switch Workspace option.

pp. 12

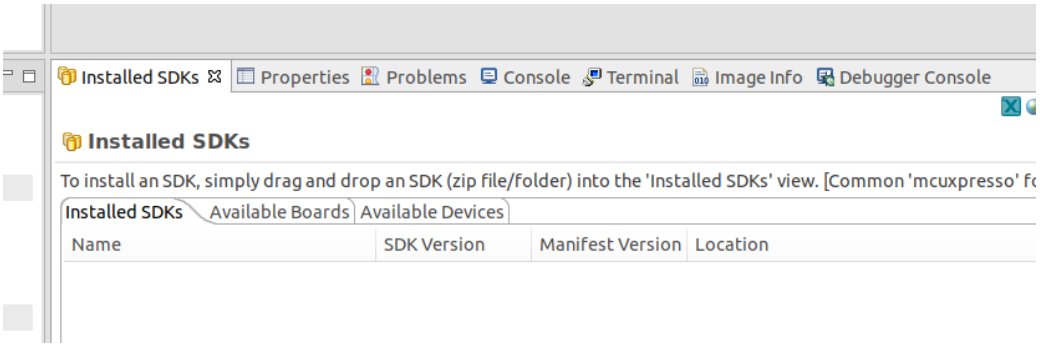
2.
Copy sittings

copy settings (preferences) from an
existing workspace to the new workspace using the various Copy Settings tick box options.

pp. 12

3.
most NXP MCUs is added to the IDE via the installation of an SDK

4.
SDK view
pp. 26



5.
3.5 Major Components of the Develop Perspective
pp. 16

6.
4.2.1 Obtaining and Installing a Plugin SDK
pp. 26

3.5 Major Components of the Develop Perspective

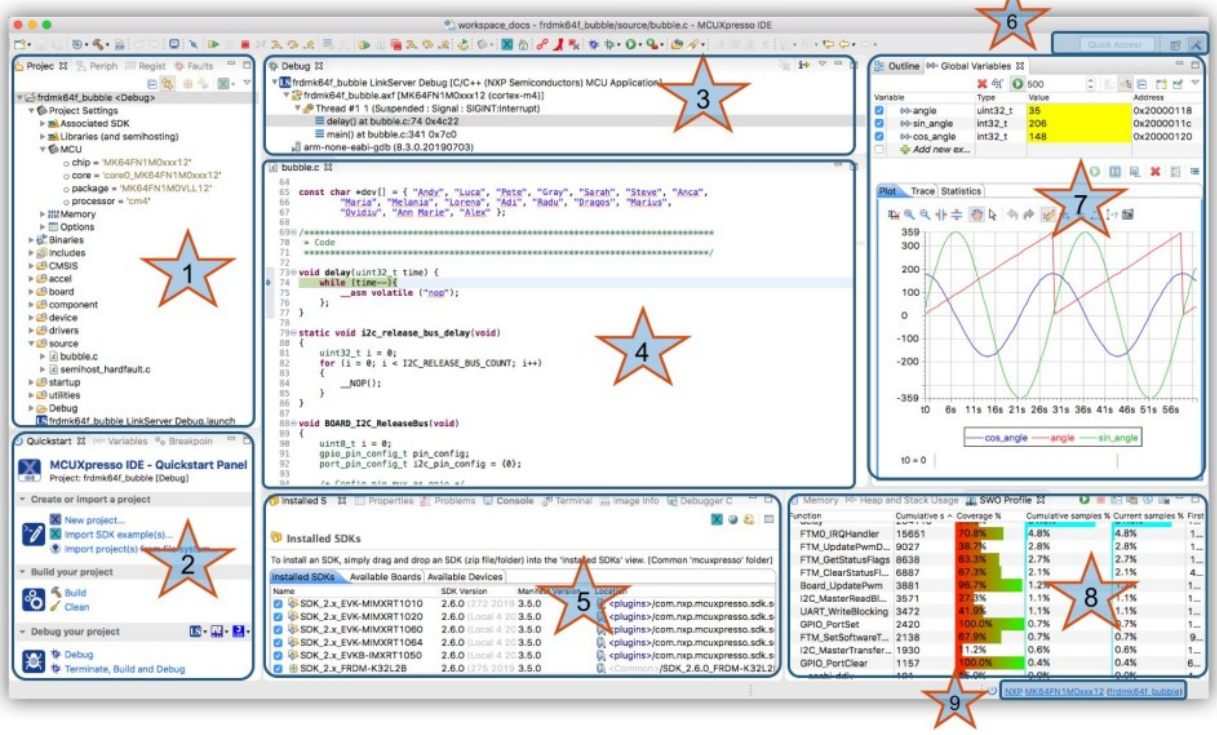
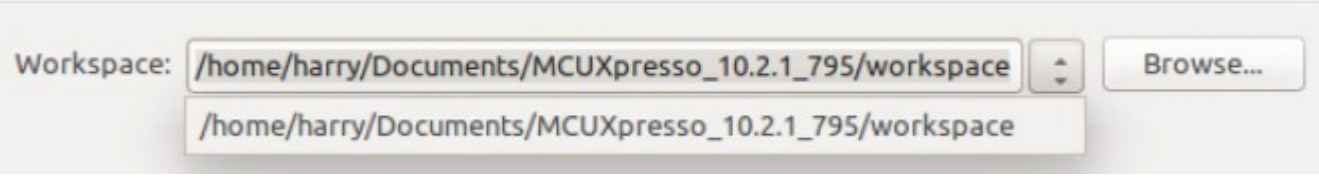


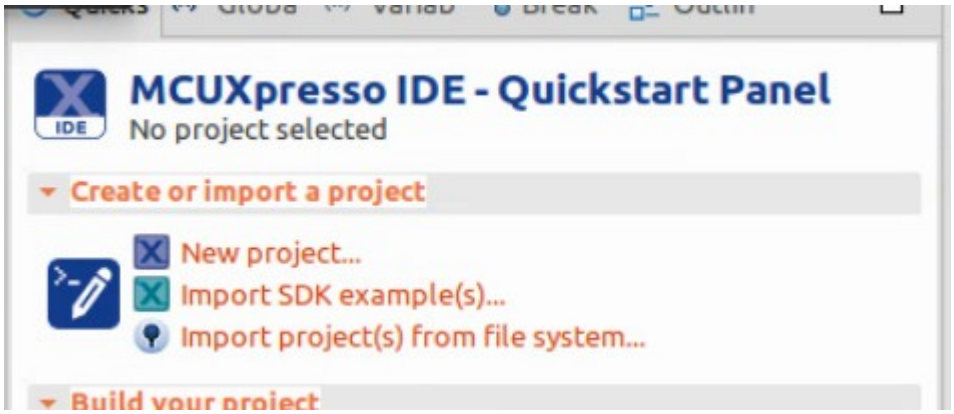
Figure 3.6. Develop Perspective (whilst debugging)

Feb 17 (Wed)

Location of my workspace on dell

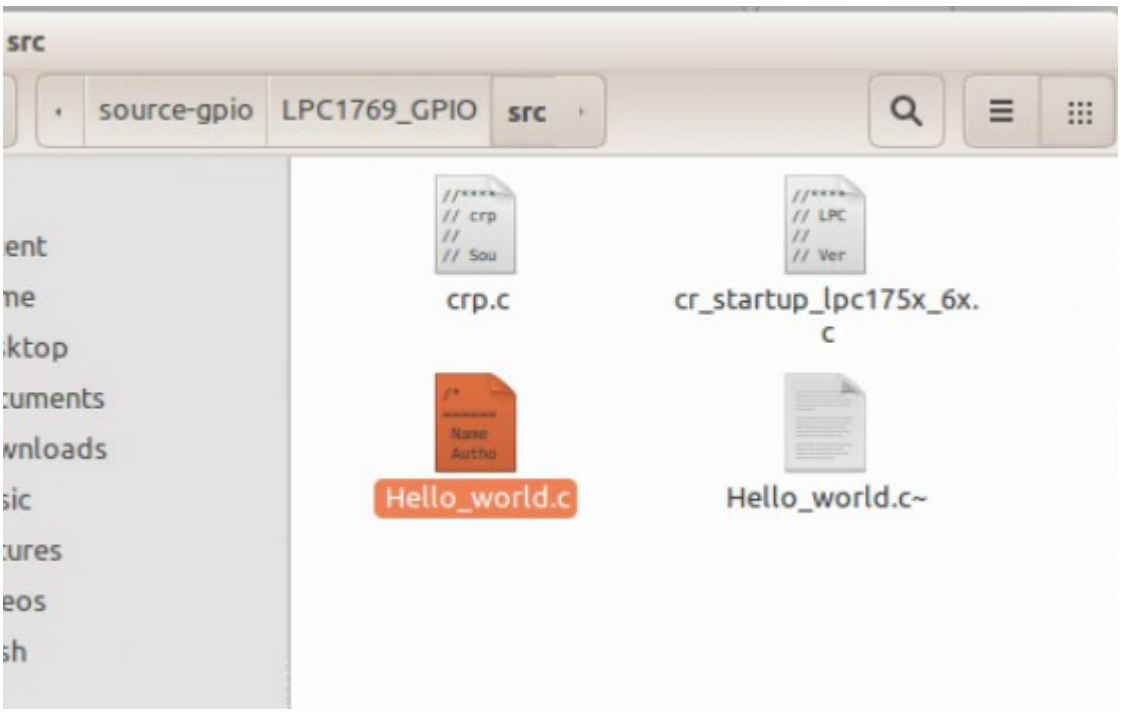


no project selected

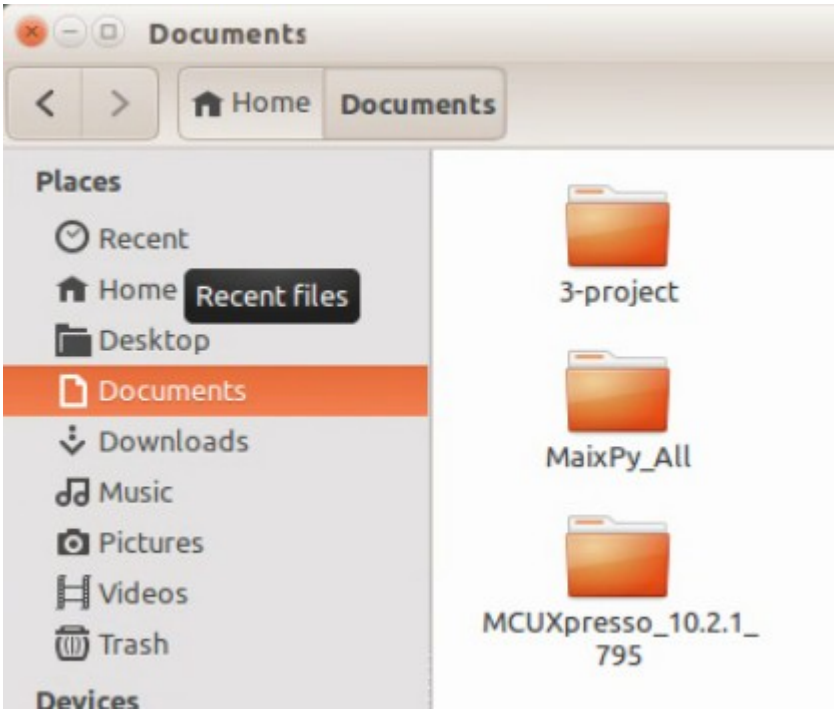


On ARM11 laptop, the expresso

For the GPIO on Dell

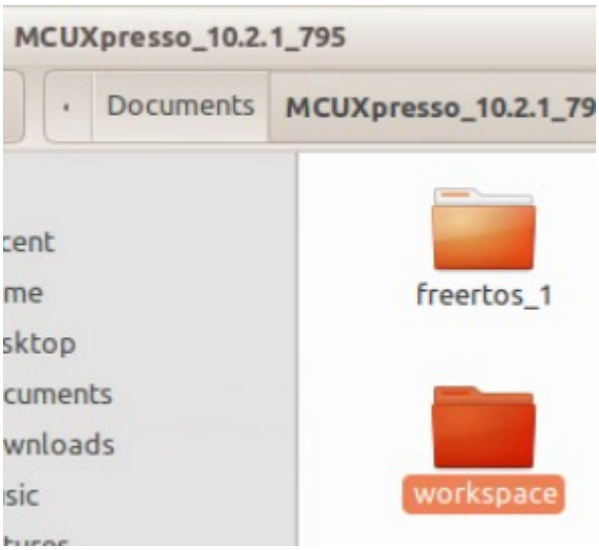


location of mcuxpresso on my dell

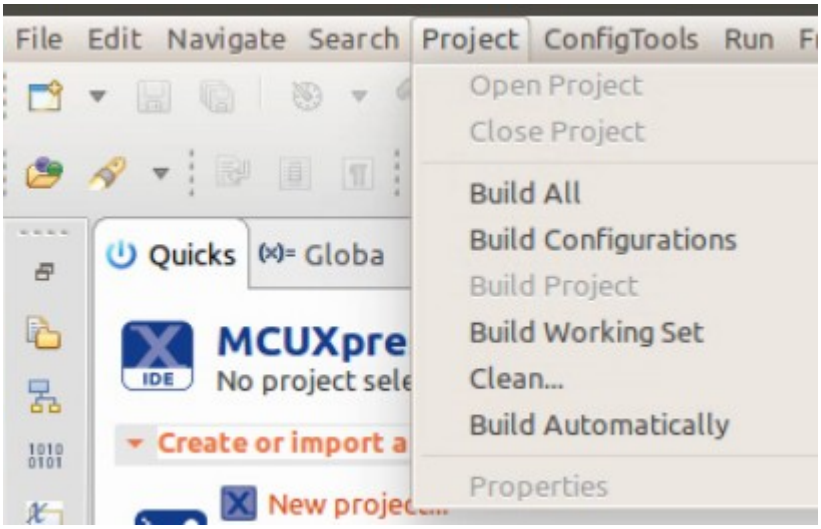


Version: 10.2.1_795

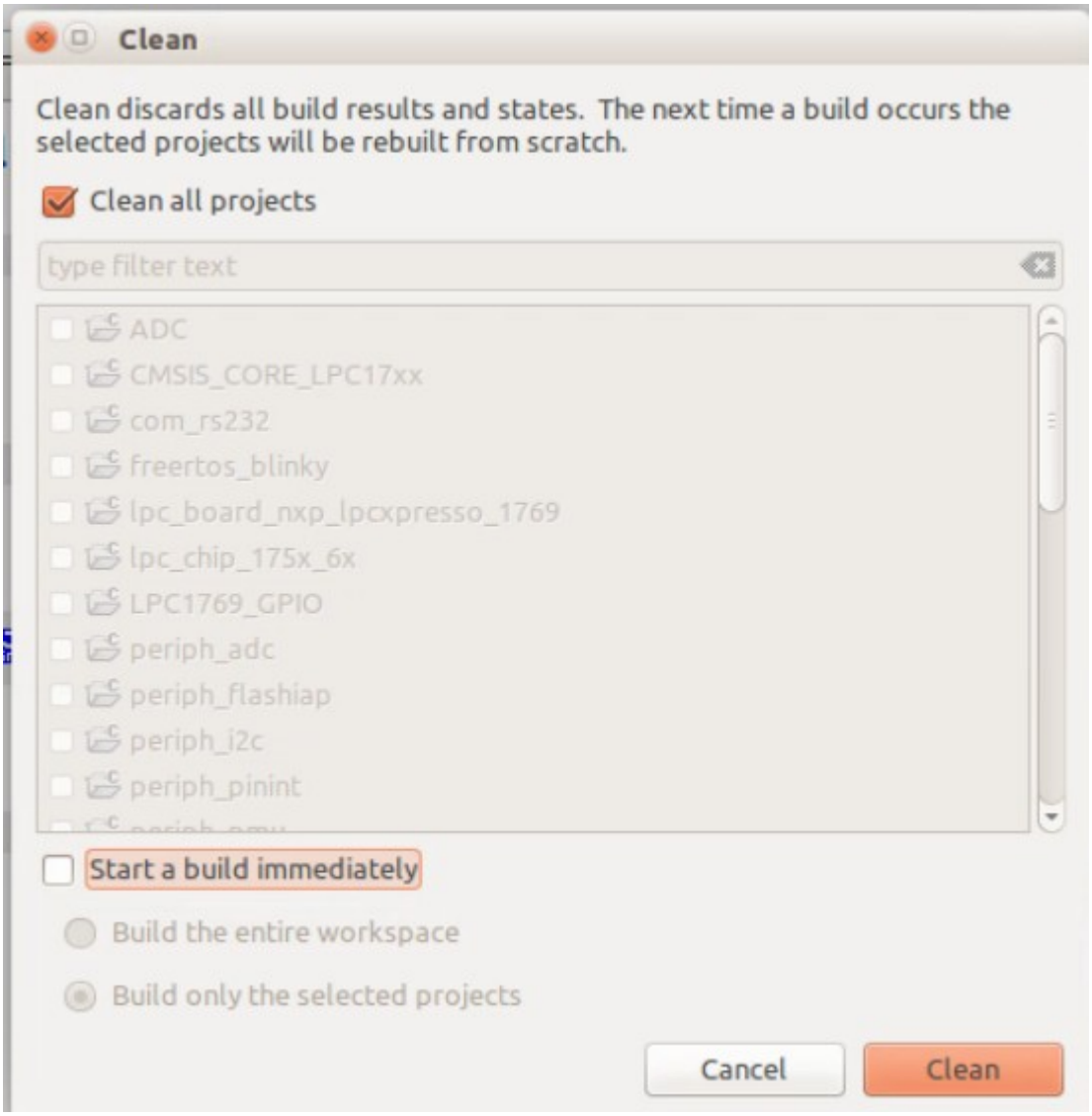
Workspace location



Now to start a existing project
(1) go to project, click on it to select clean

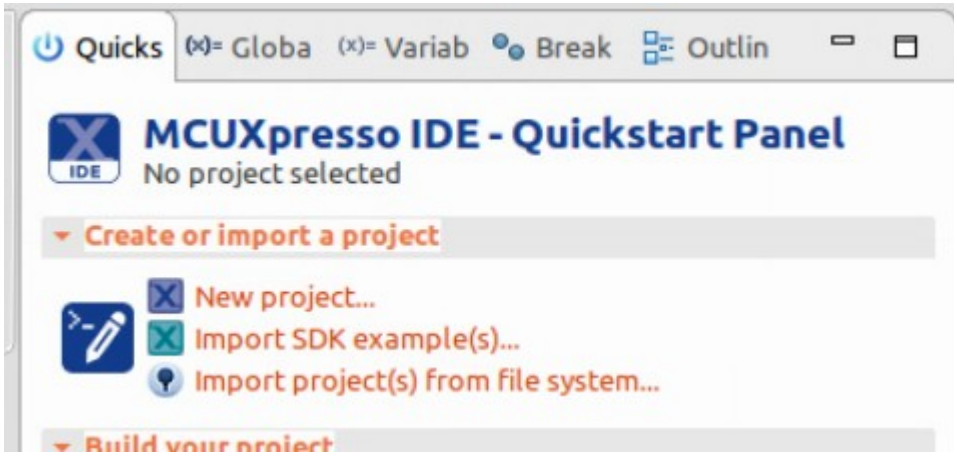


(2) Then at the popup window “clean”, click on clean all projects.

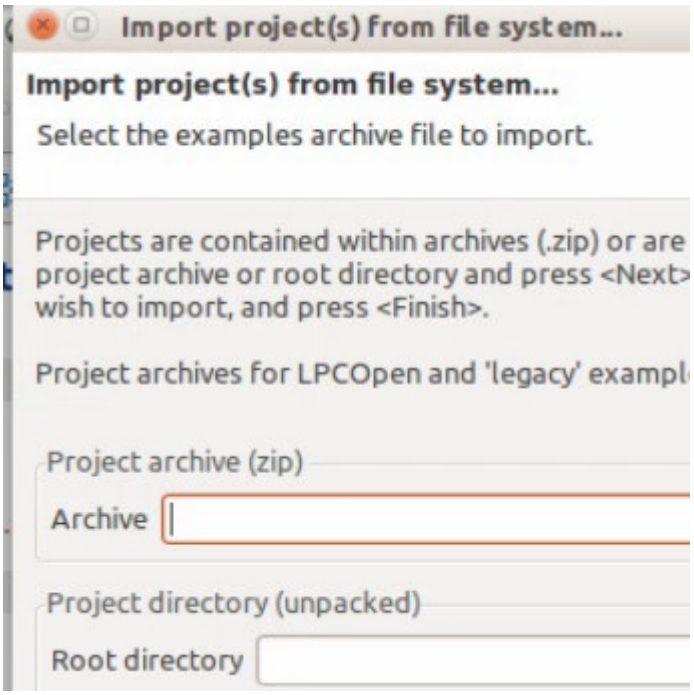


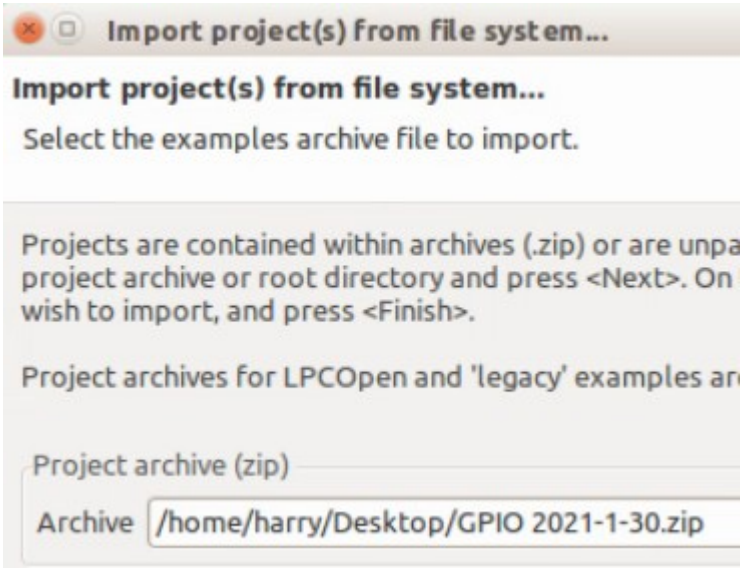
Note deselect “Start a build immediately”.

(3) now at the project panel, select import project



(4) at the popup window select the import option as archive (zip),

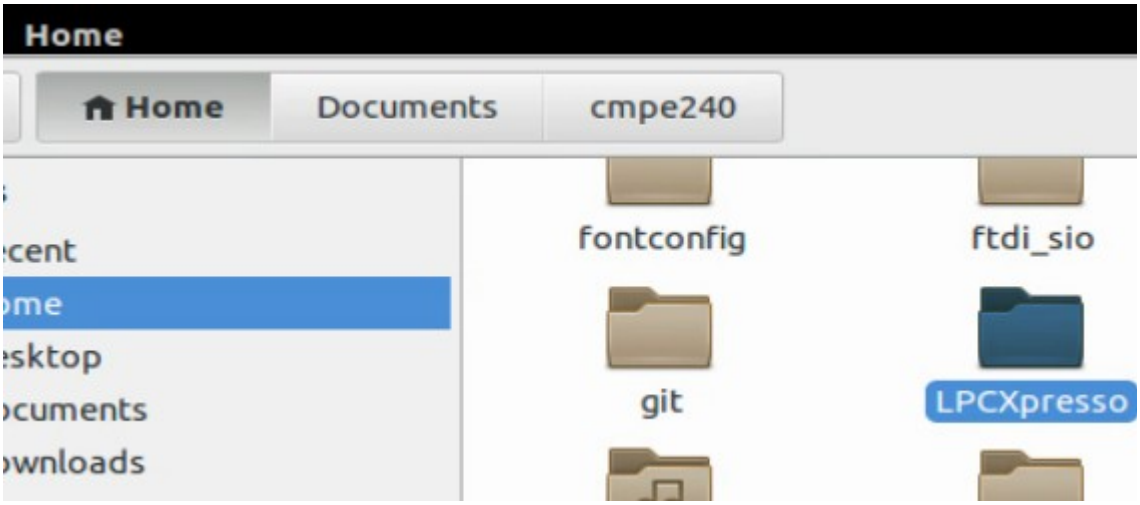




No, the above steps are not working for the project(s) already existing in the workspace. Without cleaning the workspace, we want to select project to work with.

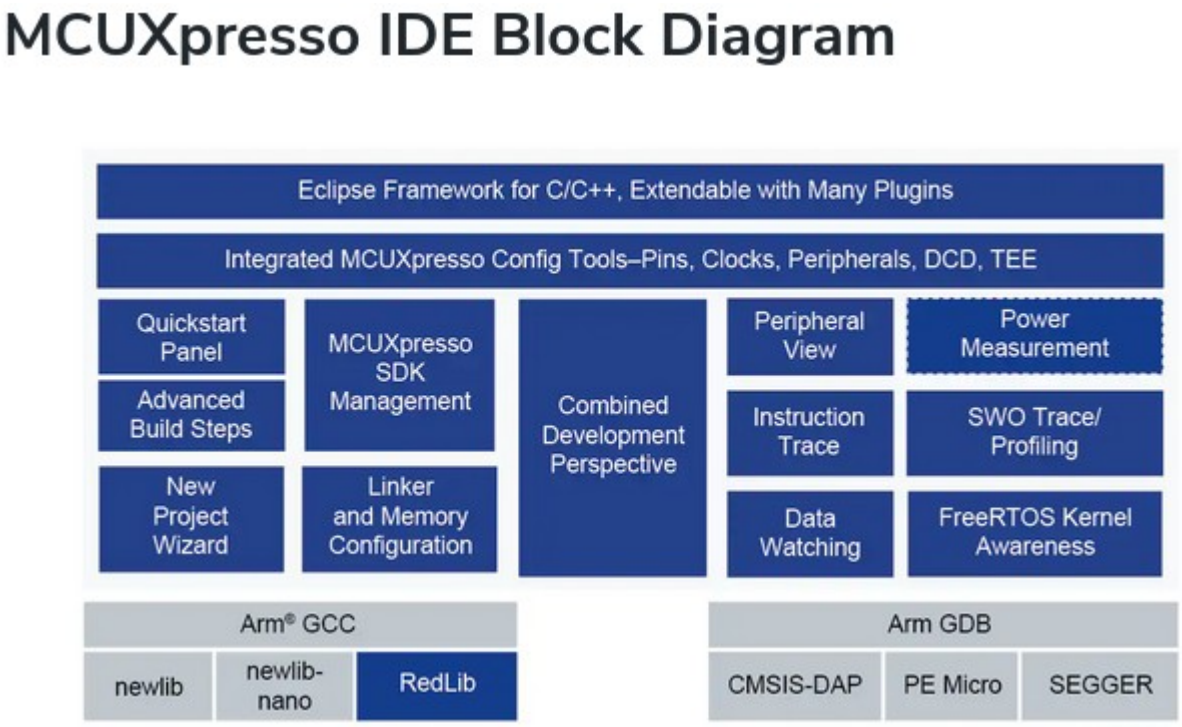
Location of xpresso on my touchscreen laptop

home/LPCXpresso



Down load MCUXpresso

https://www.nxp.com/design/software/development-software/mcuxpresso-software-and-tools-/mcuxpresso-integrated-development-environment-ide:MCUXpresso-IDE



SDK and IDE are not the same per NXP use

System Requirements

- SDK v2.9 requires IDE v11.3.x or later

Supported Devices

- Processors and Microcontrollers
- Wireless Connectivity

Download link
https://www.nxp.com/design/software/development-software/mcuxpresso-software-and-tools-/mcuxpresso-integrated-development-environment-ide:MCUXpresso-IDE?tab=Design_Tools_Tab#nogo

To ensure that you receive this email,please add the following email address to your contact list
helper@nxp.com

https://www.nxp.com/docs/en/quick-reference-guide/MCUXpresso_IDE_Installation_Guide.pdf

Note (1) MCUXpresso IDE: native debug connections via LinkServer (CMSIS-DAP).

(2) SEGGER J-Link Debug probes and PEmicro Debug probes is also installed bydefault.

(3) The installer is supplied as an executable that installs the MCUXpresso IDE components. Theinstaller requires root privileges, although, once it is installed, no special privileges are required