

Import DAVE™ version 4 generated library sources into Atollic TrueSTUDIO® for ARM®

XMC™ microcontrollers
July 2016



Agenda

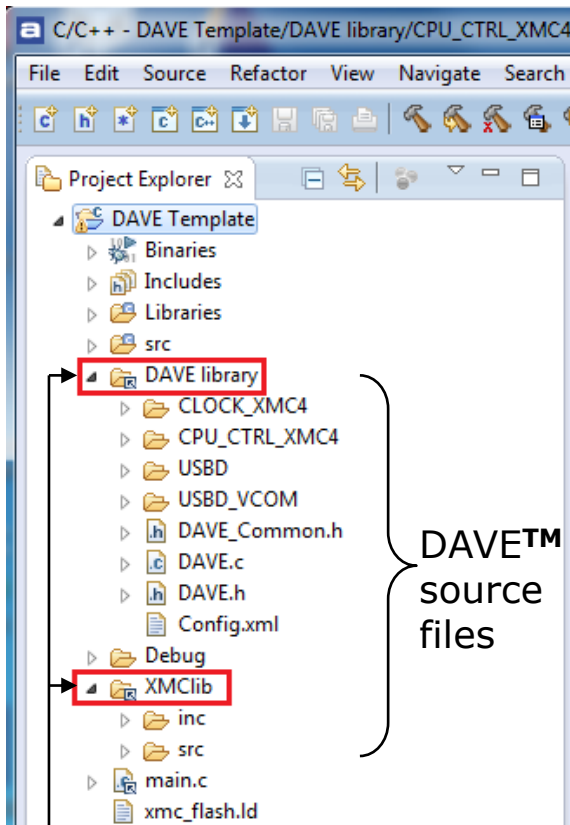
- 1 Purpose and concept
- 2 Create a new Atollic project
- 3 Define new folder in DAVE™
- 4 Link DAVE™ folders
- 5 Add DAVE™ source and include folders
- 6 Tool settings and build the project

Agenda

- 1 Purpose and concept
- 2 Create a new Atollic project
- 3 Define new folder in DAVE™
- 4 Link DAVE™ folders
- 5 Add DAVE™ source and include folders
- 6 Tool settings and build the project

Purpose: Import DAVE™ version 4 generated source files to Atollic

Atollic project outline:

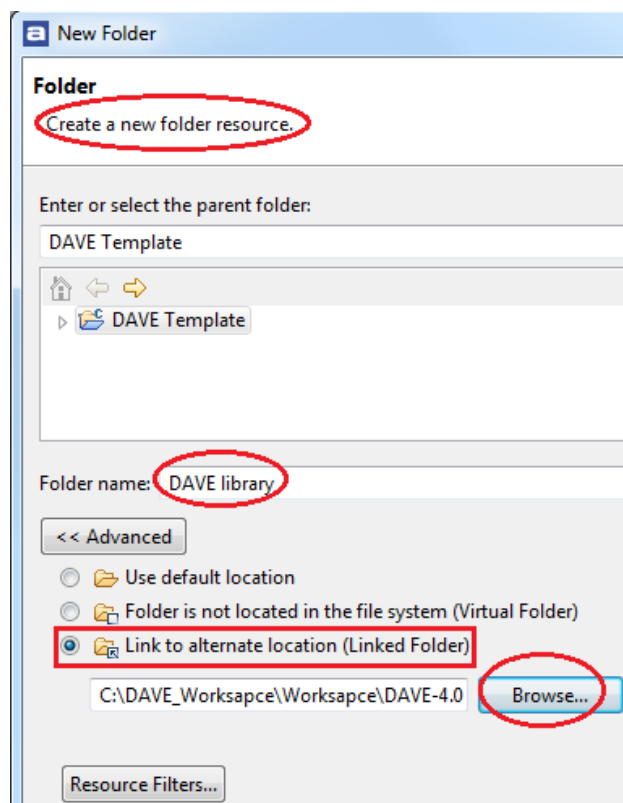


Linked resource: link to DAVE™ generated C sources, header files and XMCLib

- › DAVE™ version 4 is a free eclipse based development platform that can generate application libraries from DAVE™ Apps
- › The generated library sources are included in a DAVE™ version 4 project
- › Atollic TrueSTUDIO® (eclipse) project uses “linked resource” management to link the DAVE™ library source files and header files to the Atollic project
- › The linked resource references only the generated sources, headers and **XMC Lib LLD**. It is expected that the user develops the user code afterwards in Atollic TrueSTUDIO®
- › Not the full DAVE™ version 4 project is ported. To port the full DAVE™ project the existing user code has to be manually referenced (linked)

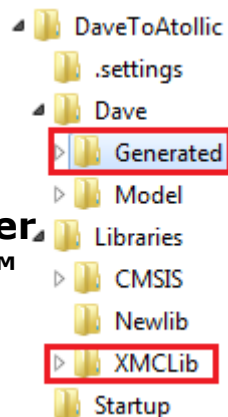
Concept

Define a new folder in Atollic project:

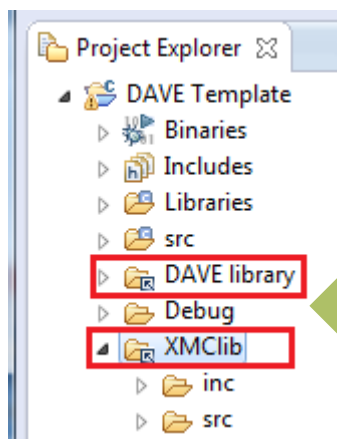


Link the new folder to DAVE™ library source folder

DAVE™ project:



linked



Respective project view in Atollic project

- > Define a new Atollic project or open the existed Atollic project
- > Open two **new** folders (DAVE™ library, XMCLib) in the Atollic project, respectively
- > Select "link to alternative location"
- > Browse to DAVE™ v4 project folder **DAVE™ \Generated** and **Libraries\XMCLib**, respectively
- > Click "Finish"
- > In the Atollic project the linked DAVE™ library source will be shown

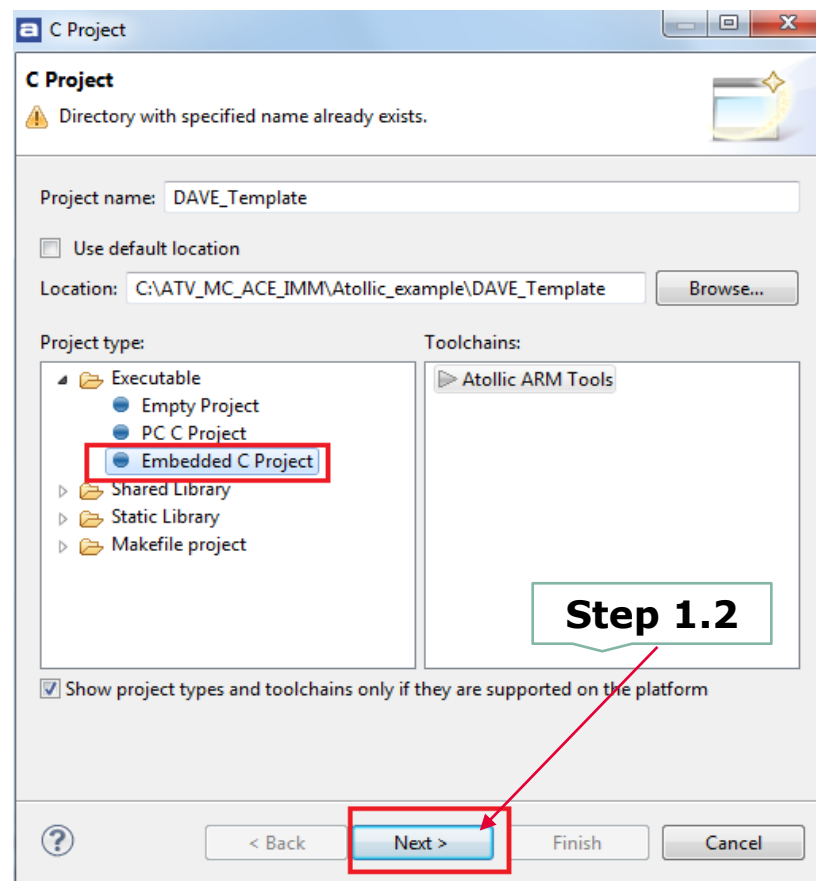
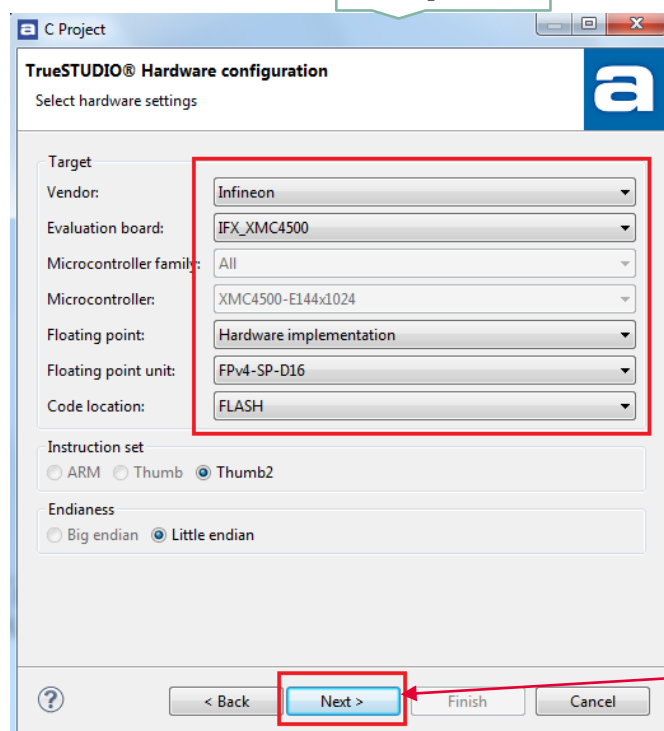
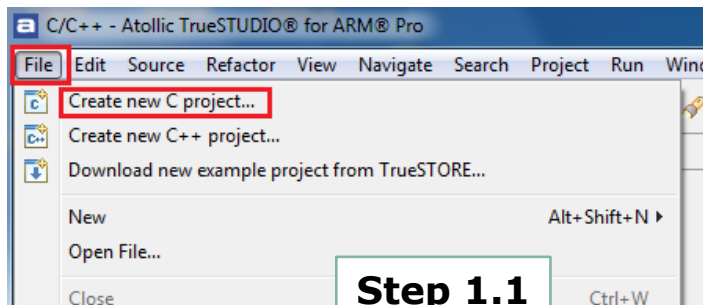
Prerequisites to follow the tutorial

- › Atollic TrueSTUDIO® version 5.2.0 or higher
- › Open the Atollic TrueSTUDIO®
- › Then, do the following step by step.....

Agenda

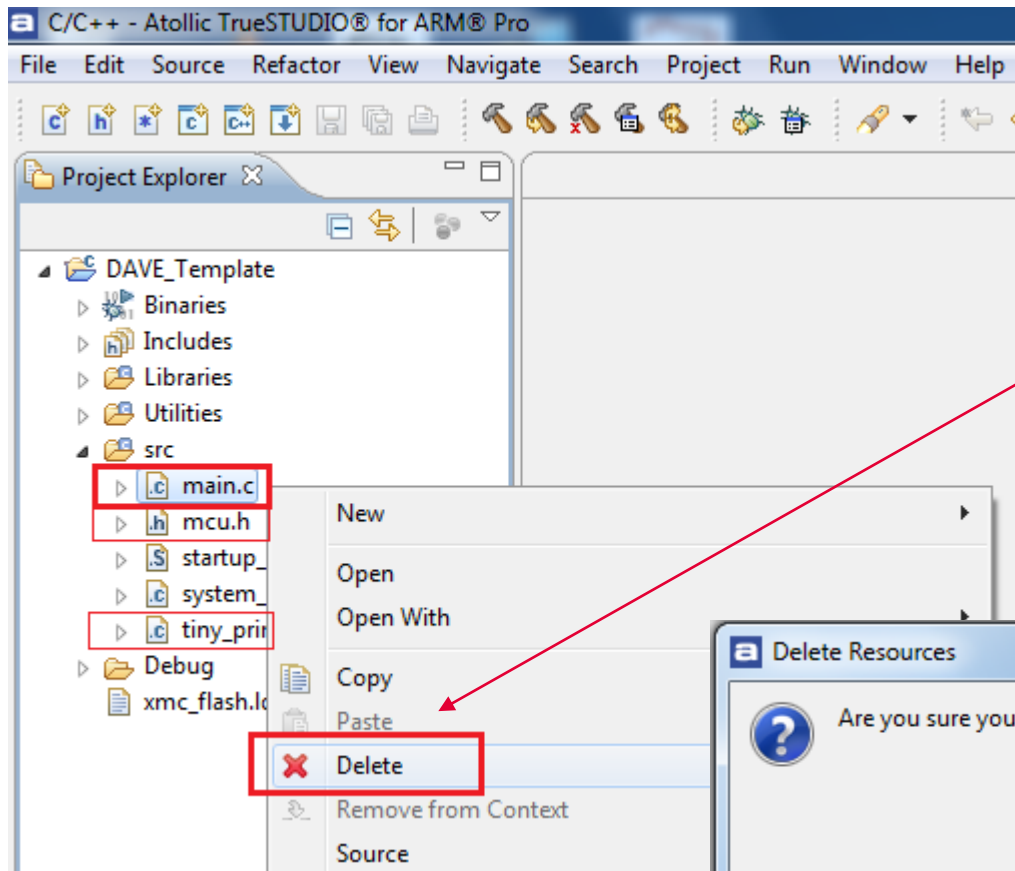
- 1 Purpose and concept
- 2 Create a new Atollic project
- 3 Define new folder in DAVE™
- 4 Link DAVE™ folders
- 5 Add DAVE™ source and include folders
- 6 Tool settings and build the project

Step 1: Create a new Atollic project



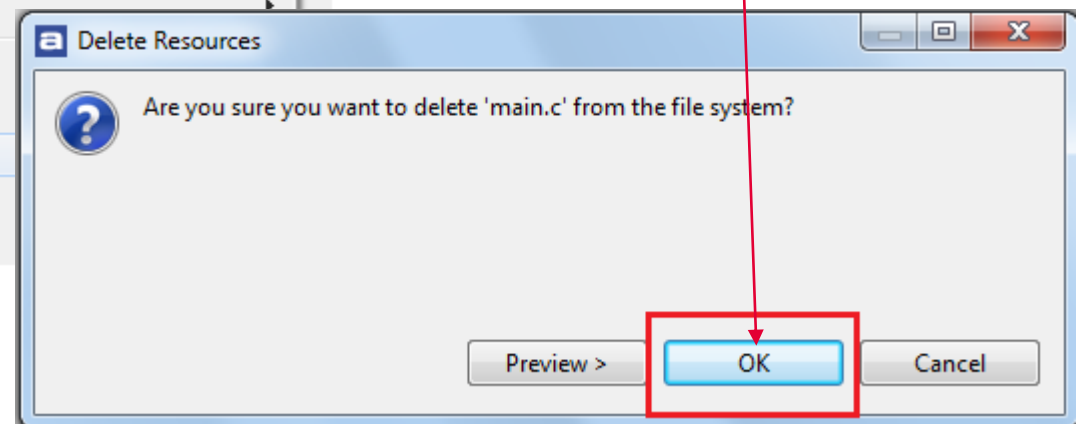
Step 1.3: choose target devices (same target as used in the DAVE™ project)

Step 2: Delete main.c from defined project



Step 2.1:
delete following files from
defined project:

- > **Main.c**
- > **Mcu.h** (only in XMC4xxx project)
- > **Tiny_printf.c**



Agenda

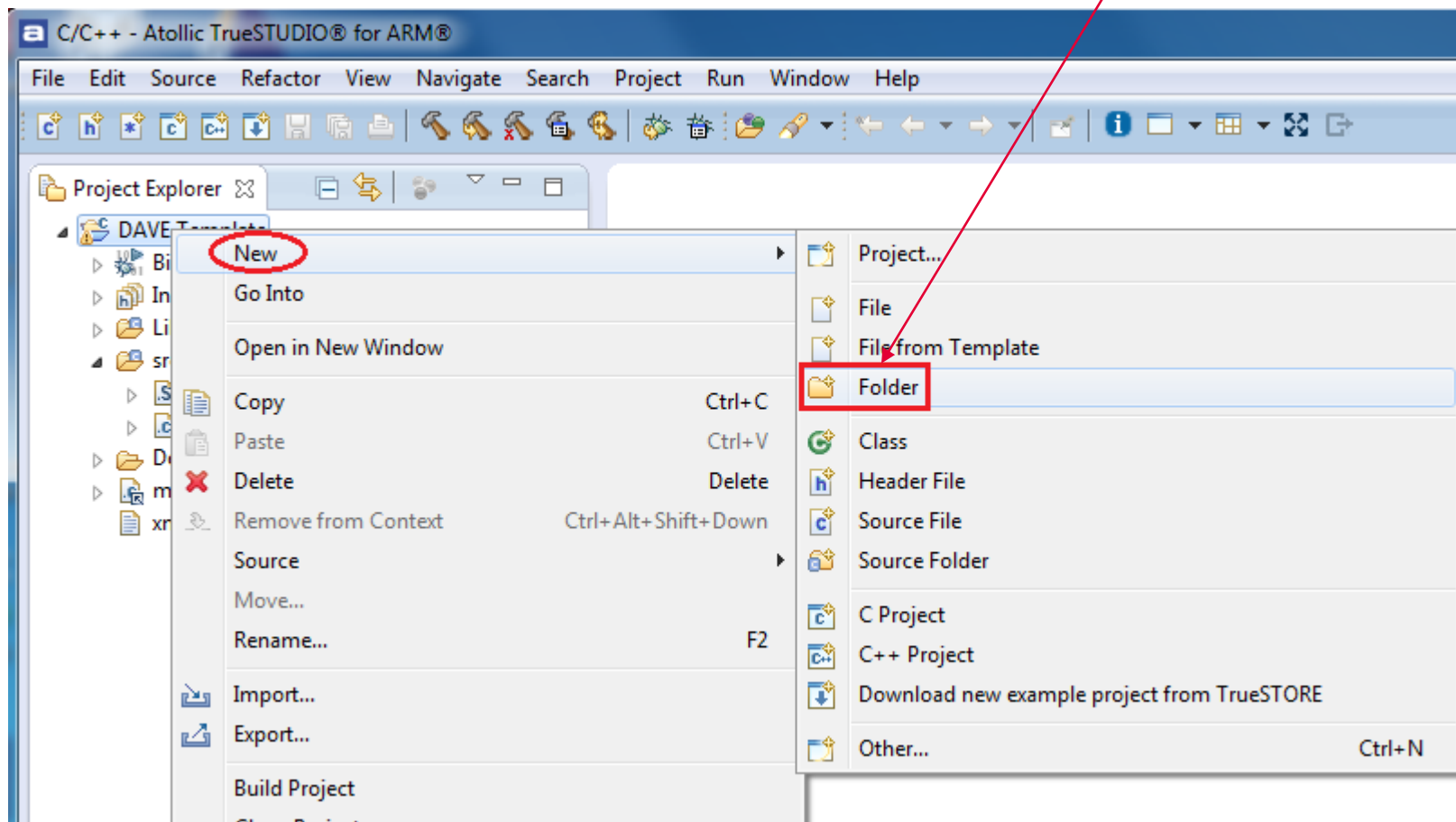
- 1 Purpose and concept
- 2 Create a new Atollic project
- 3 Define new folder in DAVE™
- 4 Link DAVE™ folders
- 5 Add DAVE™ source and include folders
- 6 Tool settings and build the project

Step 3

Define new folders: DAVE™ library and XMClib

Step 3.1:
In the project field, click right mouse

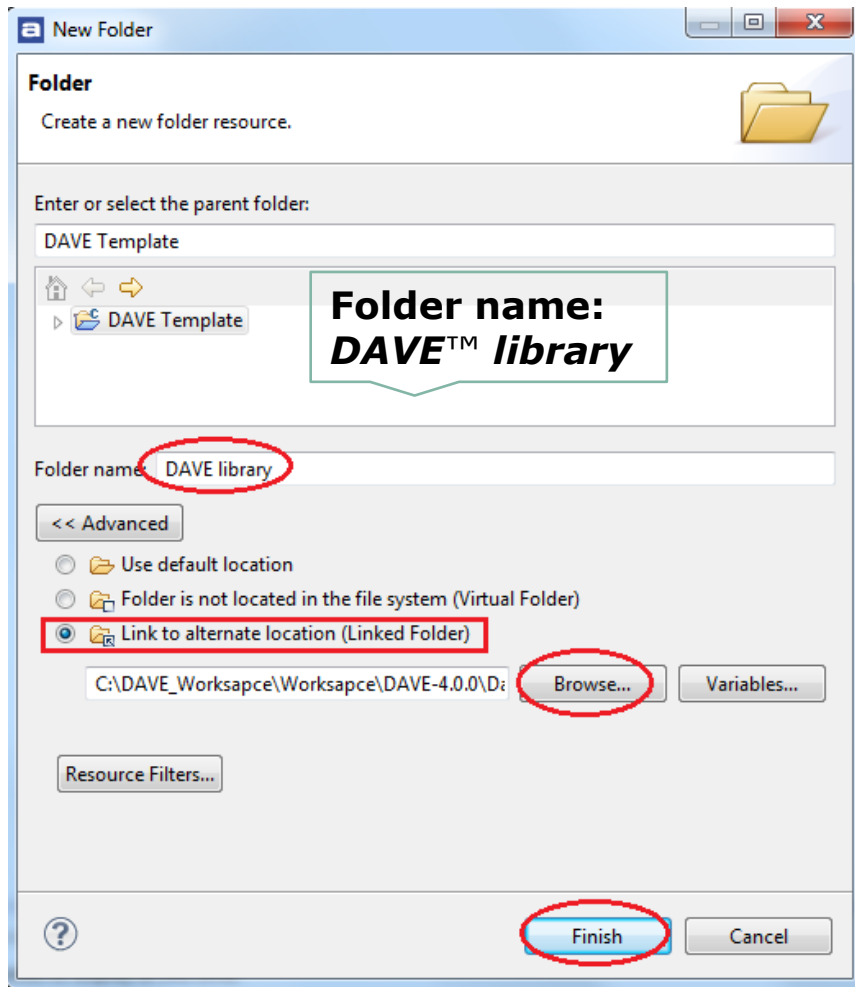
Step 3.2:
Select New->Folder



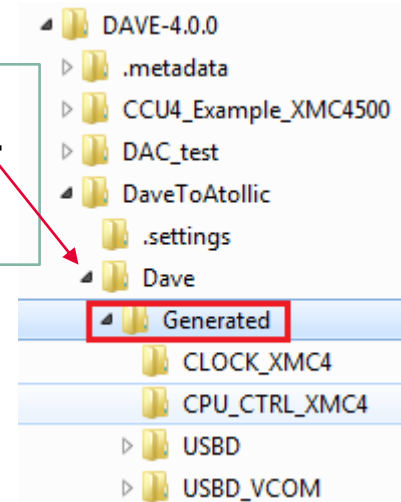
Agenda

- 1 Purpose and concept
- 2 Create a new Atollic project
- 3 Define new folder in DAVE™
- 4 Link DAVE™ folders
- 5 Add DAVE™ source and include folders
- 6 Tool settings and build the project

Step 4: Link DAVE™ source codes into DAVE™ library



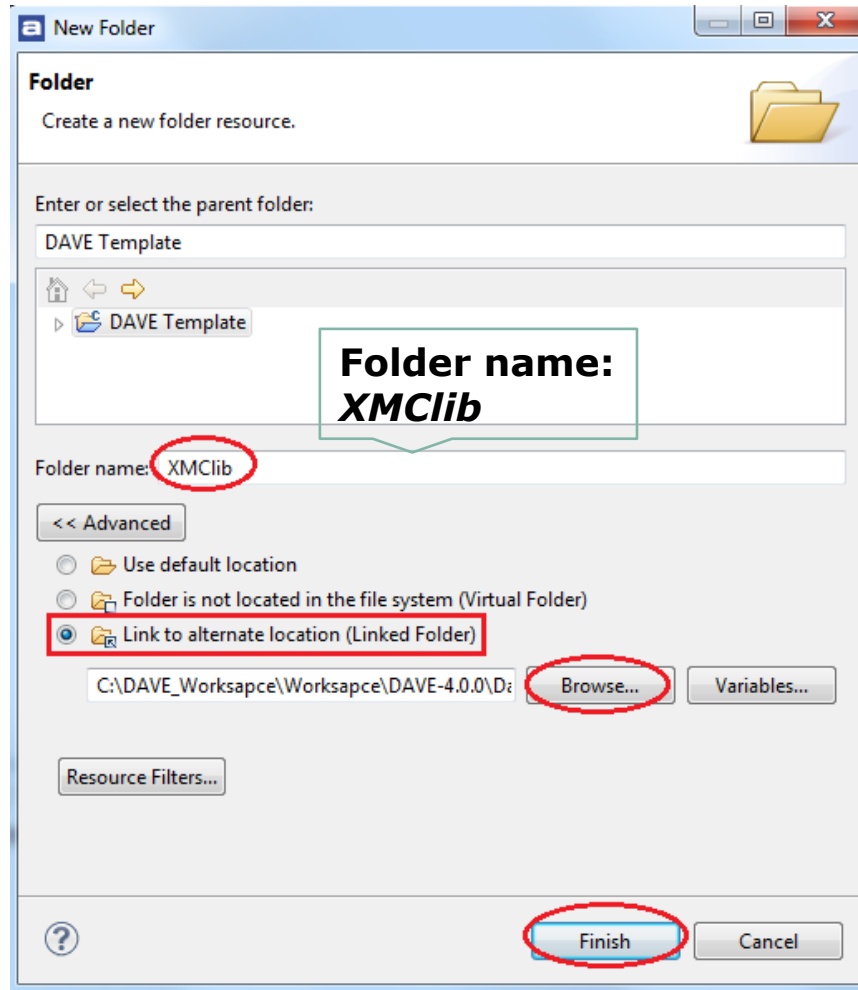
Step 4.1:
**Browse to DAVE™ v4
project folder
../Dave™/Generated**



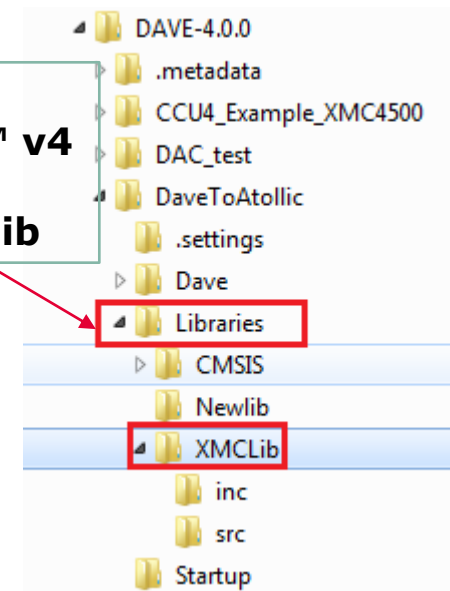
Step 4.2

DAVE™ v4 project folder

Step 5: Link LLD library source codes into XMClib



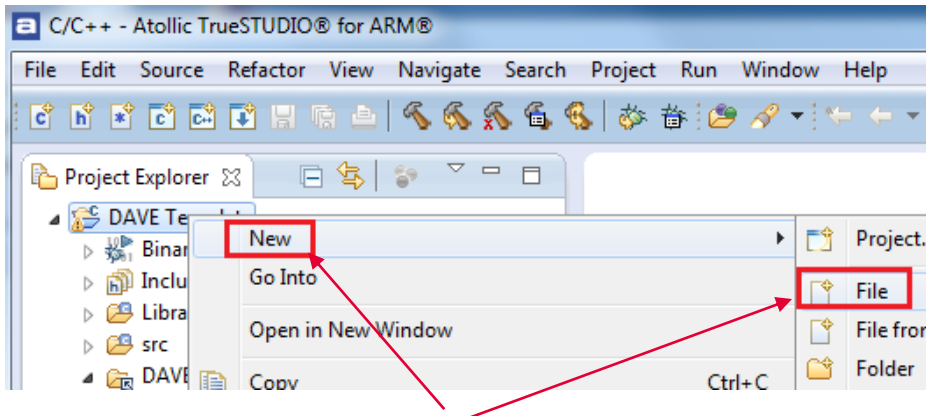
Step 5.1:
**Browse to DAVE™ v4
project folder
../Libraries/XMClib**



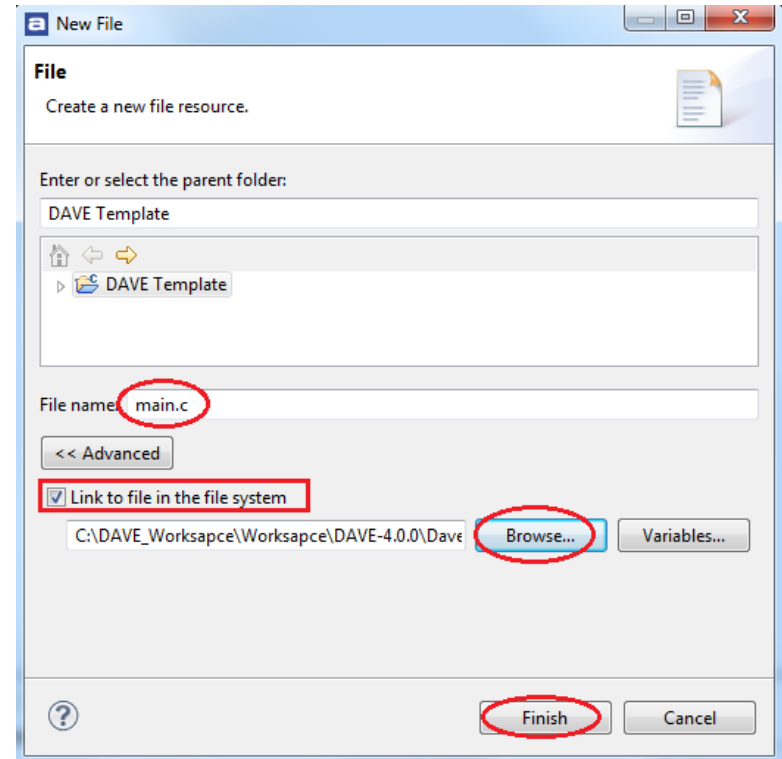
Step 5.2

DAVE™ v4 project folder

Step 6: Link main.c as needed



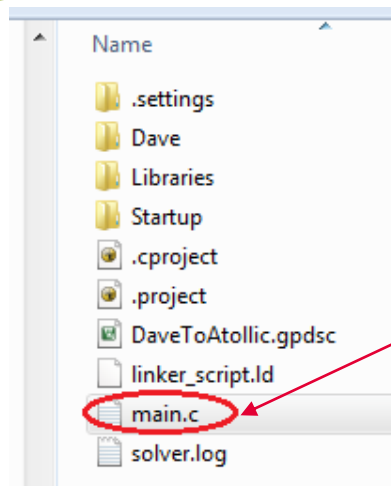
Step 6.1:
In the project field, click right mouse,
select New->File



Step 6.2:
Browse to DAVE™ project
folder, select *main.c*

Step 6.2

**DAVE™
project
folder**

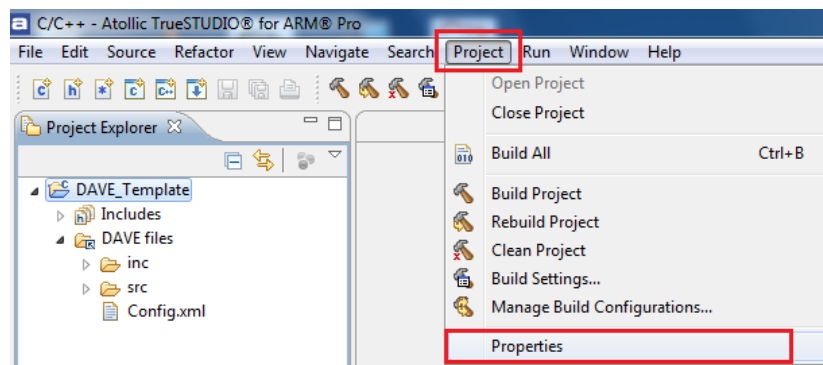


**Or simply copy the main.c from the
DAVE™ project to the Atollic project**

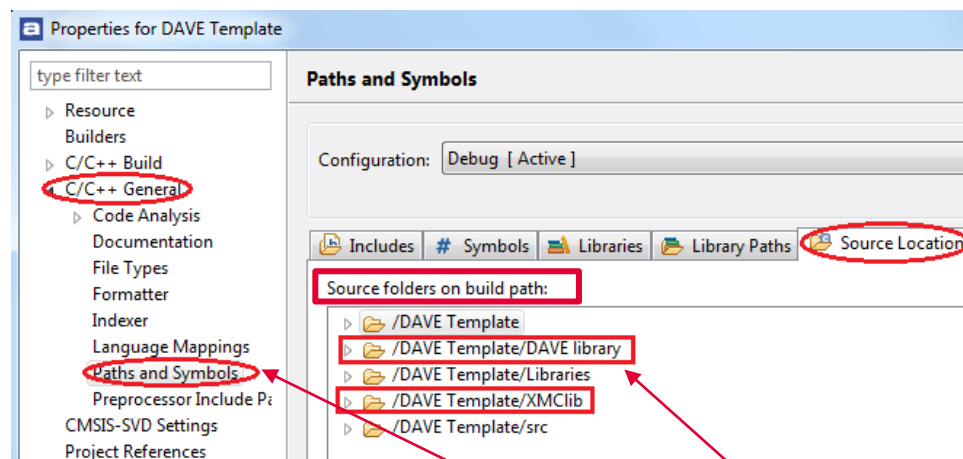
Agenda

- 1 Purpose and concept
- 2 Create a new Atollic project
- 3 Define new folder in DAVE™
- 4 Link DAVE™ folders
- 5 **Add DAVE™ source and include folders**
- 6 Tool settings and build the project

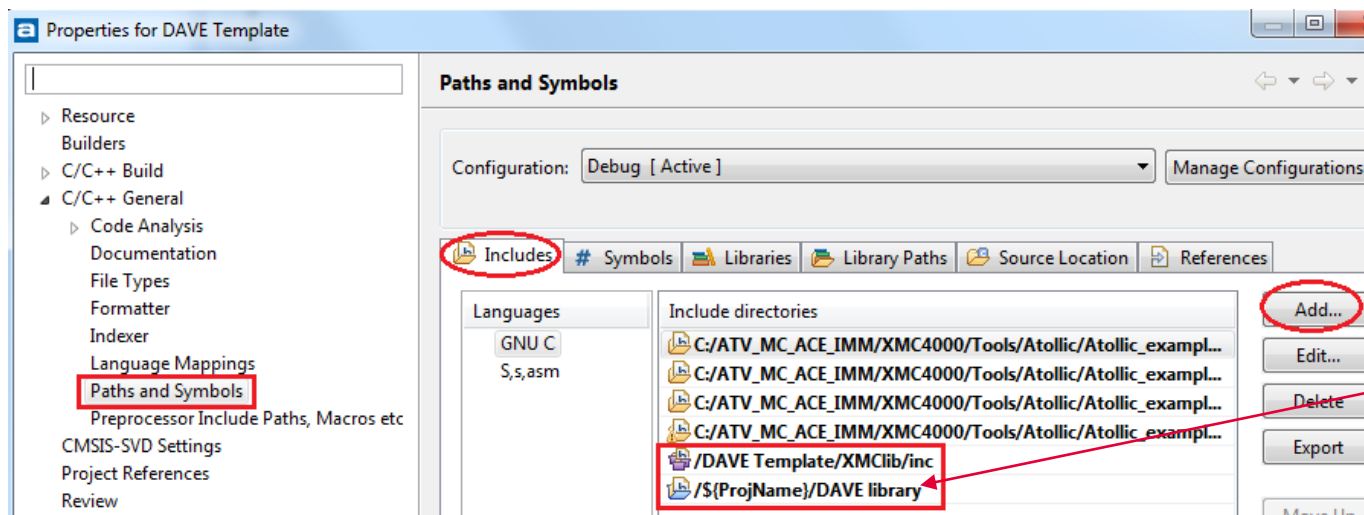
Step 7: Add DAVE™ source and include folders



Step 7.1: Open properties



Step 7.2: Add source Paths ../DAVE™ library /DAVE™ Template



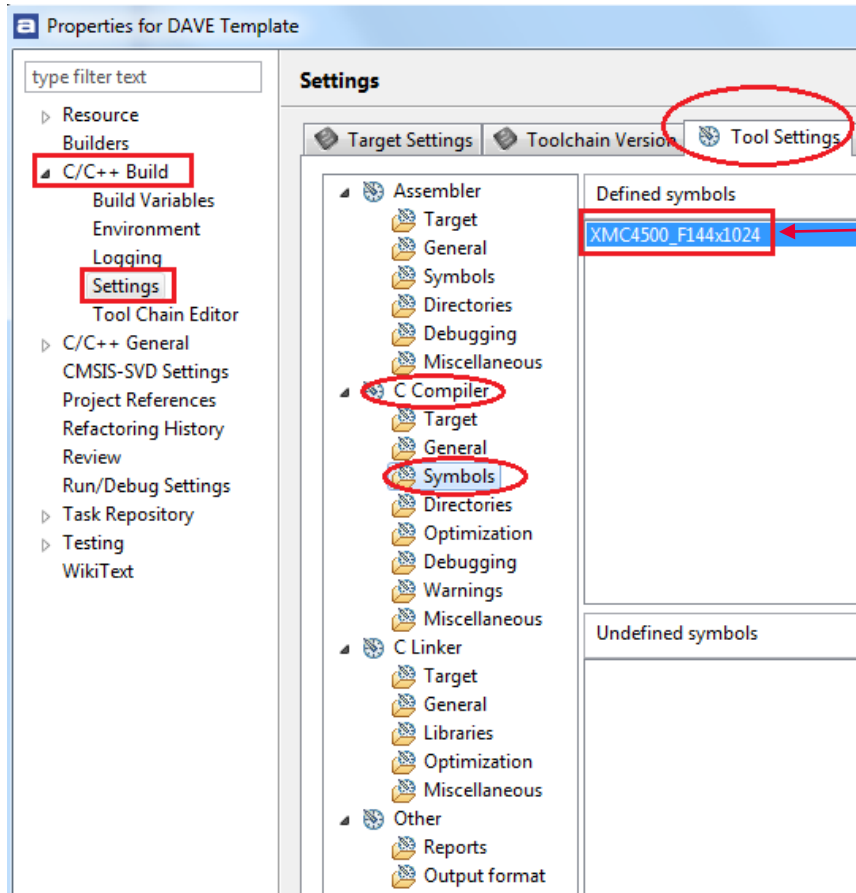
Step 7.3: Add include file Paths: ../XMClib/inc/ ../DAVE™ library/

Agenda

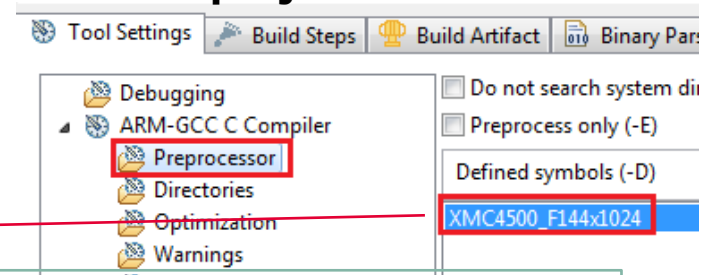
- 1 Purpose and concept
- 2 Create a new Atollic project
- 3 Define new folder in DAVE™
- 4 Link DAVE™ folders
- 5 Add DAVE™ source and include folders
- 6 Tool settings and build the project

Step 8: Tool settings

Atollic project:



DAVE™ project:



Copy symbols from DAVE™ project to Atollic project, if it is not defined.

Please note, other settings might need be changed depending on the imported code and required optimizations.

Necessary changes:

1. Compiler

- > Optimization level
- > Symbols definition
- > FPU

2. Linker

- > Linker script file
- > Libraries needed

Step 9: Build the project

Project Explorer

- DAVE Template
 - Binaries
 - Includes
 - DAVE library
 - CLOCK_XMC4
 - CPU_CTRL_XMC4
 - USB
 - USB_VCOM
 - DAVE_Common.h
 - DAVE.c
 - DAVE.h
 - Config.xml
 - Libraries
 - XMCLib
 - inc
 - src
 - Debug
 - main.c
 - xmc_flash.ld

main.c

```

1  /*
2  * main.c
3  *
4  * Created on: 2015 Mär 06 11:48:15
5  * Author: wangguan
6  */
7
8
9
10
11 #include <XMC4500.h>
12 #include <DAVE.h>          //Declarations from DAVE Code Generation (includes SFR d
13
14 /**
15  * @brief main() - Application entry point
16  *
17  * <b>Details of function</b><br>
18  * This routine is the application entry point. It is invoked by the device startup
19  * invoking the App initialization dispatcher routine - DAVE_Init() and hosting the
20  * code.
21  */
22 int main(void)
23 {
24     DAVE_STATUS_t status;
25
26     status = DAVE_Init();    /* Initialization of DAVE Apps */
27
28     !!!
  
```

Problems Tasks Console Properties Search

CDT Build Console [DAVE Template]

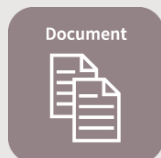
```

arm-atollic-eabi-gcc -c -mthumb -mcpu=cortex-m4 -mfloat-abi=hard -mfpu=fpv4-sp-d16 -std=gr
arm-atollic-eabi-gcc -c -mthumb -mcpu=cortex-m4 -mfloat-abi=hard -mfpu=fpv4-sp-d16 -std=gr
arm-atollic-eabi-gcc src\system_XMC4500.o src\startup_XMC4500.o main.o XMCLib\src\xmc_wdt
C:\Program Files\Atollic\TrueSTUDIO for ARM Pro 5.2.0\ide\jre\bin\java -jar C:\Program Fi
Generate build reports...
Print size information
text  data  bss  dec  hex filename
17420 3724 2216 23360 5b40 DAVE Template.elf
Print size information done
Generate build reports done
  
```

Build project

Support material

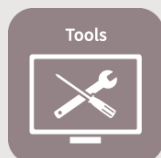
Collaterals and Brochures



- › Product Briefs
- › Selection Guides
- › Application Brochures
- › Presentations
- › Press Releases, Ads

› www.infineon.com/XMC

Technical Material



- › Application Notes
- › Technical Articles
- › Simulation Models
- › Datasheets, MCDS Files
- › PCB Design Data

› www.infineon.com/XMC

› [Kits and Boards](#)

› [DAVE™](#)

› [Software and Tool Ecosystem](#)

Videos



- › Technical Videos
- › Product Information Videos

› [Infineon Media Center](#)

› [XMC Mediathek](#)

Contact



- › Forums
- › Product Support

› [Infineon Forums](#)

› [Technical Assistance Center \(TAC\)](#)

Glossary abbreviations

- › DAVE™ Free development IDE for XMC™
- › LLD Low-Level Diver

Disclaimer

The information given in this training materials is given as a hint for the implementation of the Infineon Technologies component only and shall not be regarded as any description or warranty of a certain functionality, condition or quality of the Infineon Technologies component.

Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind (including without limitation warranties of non-infringement of intellectual property rights of any third party) with respect to any and all information given in this training material.



Part of your life. Part of tomorrow.

