DWM1001-DEV Development Board

Simple Example Code Projects

For nRF52 on DWM1001 module using uVision

Version 1.0

Contents

[1 uVision 3](#_Toc503883878)

[2 Hardware set-up 3](#_Toc503883879)

[3 nRF52 SDK 4](#_Toc503883880)

[4 Folder structure 4](#_Toc503883881)

[4.1 examples 4](#_Toc503883882)

[4.2 Components 5](#_Toc503883883)

[4.3 deca\_driver 5](#_Toc503883884)

[4.4 external 5](#_Toc503883885)

[5 Gotcha 6](#_Toc503883886)

[5.1 J-Link Debug settings. 6](#_Toc503883887)

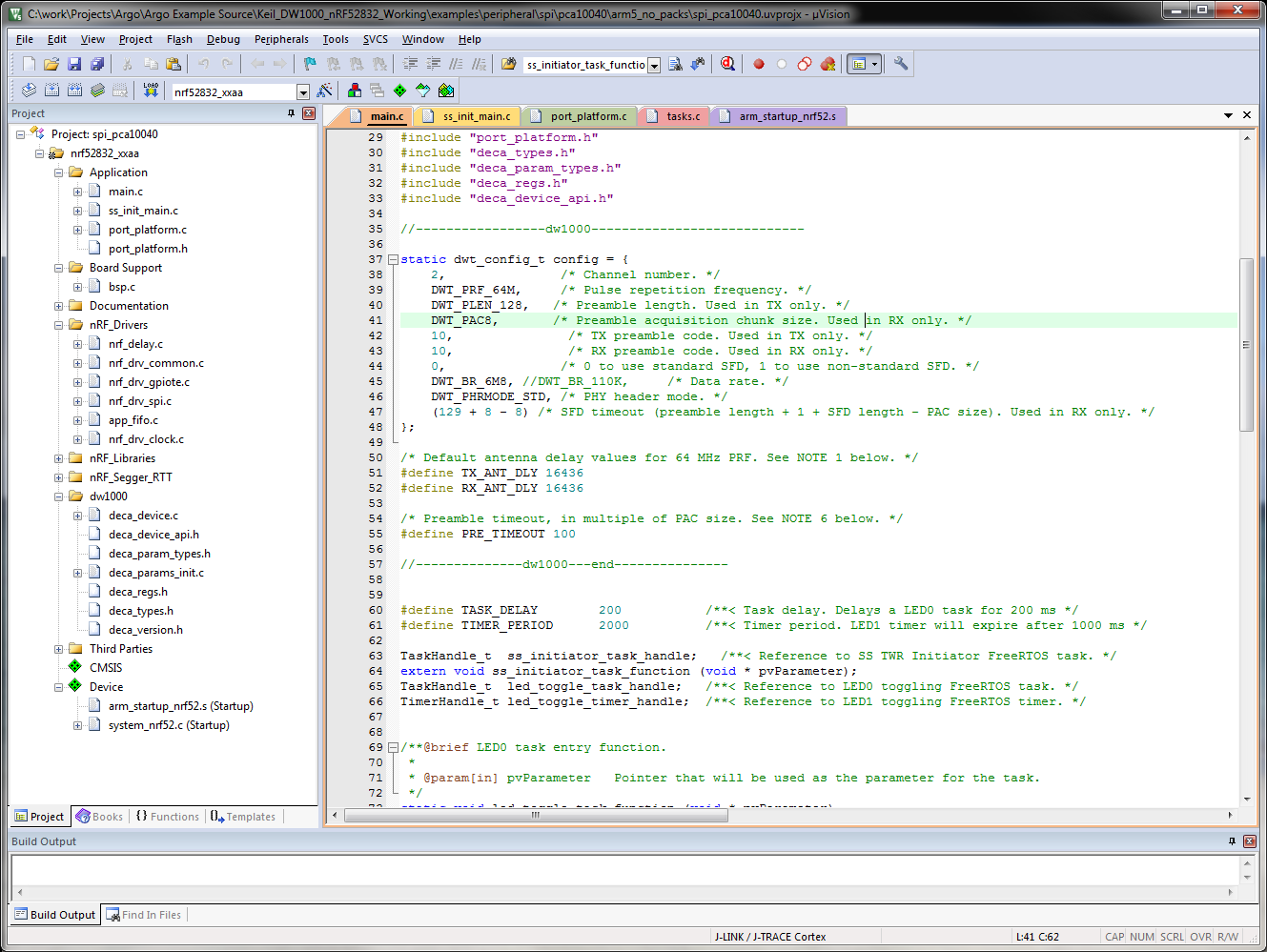
[5.2 Flash erase. 6](#_Toc503883888)

[5.3 "Cortex M4 Error" 7](#_Toc503883889)

[6 Setting up a new project by copying an existing project. 8](#_Toc503883890)

# uVision

The free version of uVision is code limited to 32K flash but it is ideal for getting started with creating your own code.



Download from:

https://www.keil.com/demo/eval/arm.htm

# Hardware set-up

As the DWM1001 board has a built in J-Link the board can be connected directly to the PC USB port. Flash can be downloaded directly and debugging can be carried out without further devices or connections. See the diagram below to see this simple connection.



# nRF52 SDK

The required sections of the nRF SDK are included in the release of simple examples for ARGO. So for this purpose you do not need to download the SDK.

Details from.

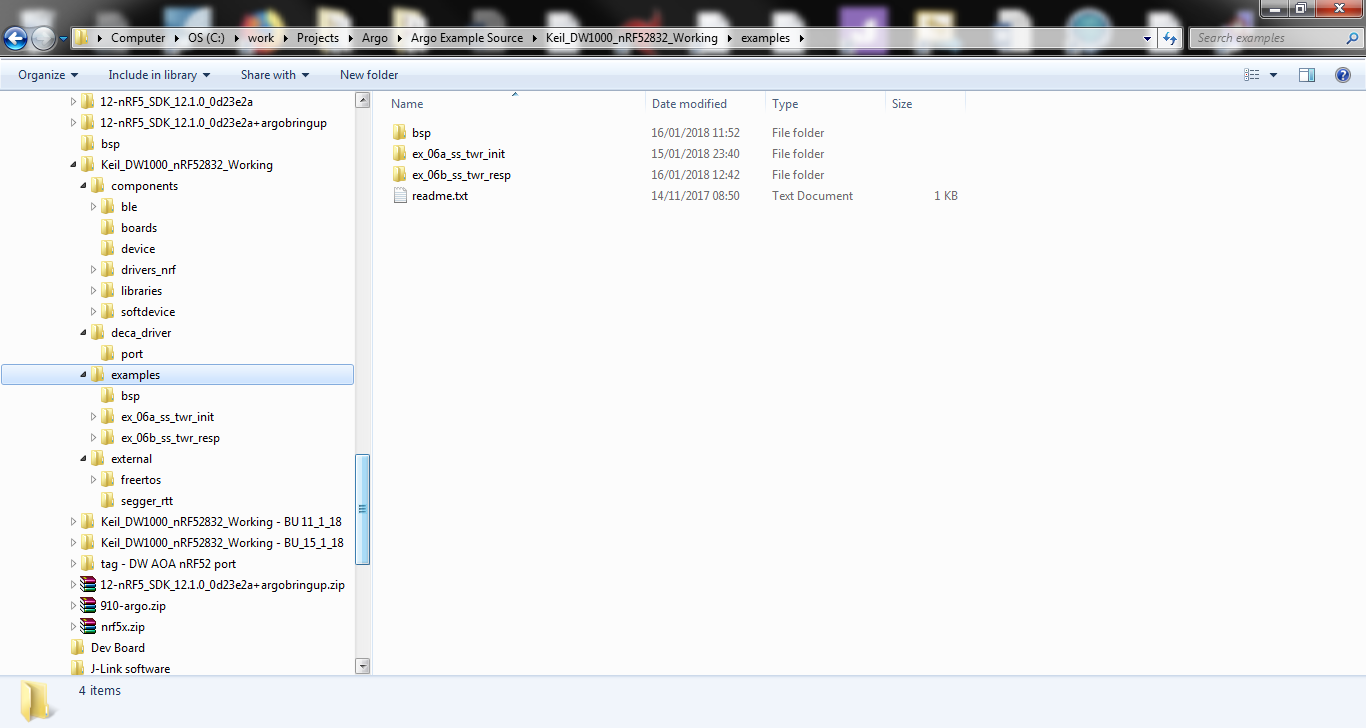
http://infocenter.nordicsemi.com/index.jsp?topic=%2Fcom.nordic.infocenter.nrf52%2Fdita%2Fnrf52%2Fchips%2Fnrf52832\_ps.html

If required download SDK From:

https://www.nordicsemi.com/eng/nordic/Products/nRF52-DK/nRF5-SDK-v12-zip/54283

# Folder structure

The objective here is to create a simple but elegant structure which keeps the example code in a separate folder to the DW1000 specific code and the nRF SDK and any other drivers or development code required to build the projects. The sections below give a short overview of each of the folders in the release. Further project development should follow the same structure.



## examples

This is the folder for the example projects. In the figure above it contains just two example projects

ex\_06a\_ss\_twr\_init

and

ex\_06b\_ss\_twr\_resp

These example projects allow the setup of Single Sided Two Way Ranging between two DWM1001-DEV PCBs.

Each project folder contains the project files and source files specific to this project and the sys\_config.h configuration header file for this project.

This folder also contains the "bsp" folder. This contains board specific defines which originated with the Nordic SDK and which were adapted to match the DWM1001-DEV PCB and the DWM1001 module..

## Components

This folder contains the required sections of the Nordic SDK.

## deca\_driver

The source and header files proving the API for the DW1000 are located in this folder.

## external

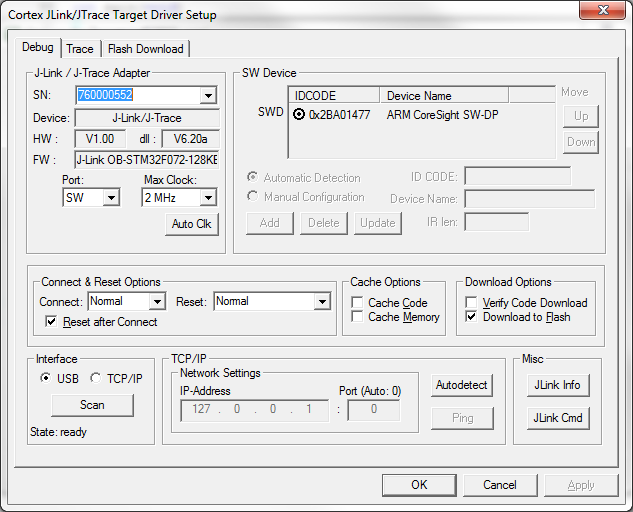
This folder contains code with is not DW1000 or nRF52 specific. The FreeRTOS code is located here.

# Gotcha

The uVision application presented a few problems which I discovered when developing the first of the examples. The settings in this section can be accessed fro the project properties dialog.

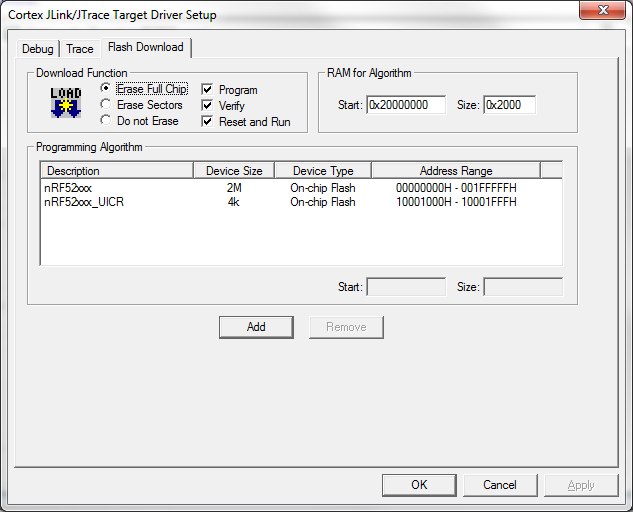
## J-Link Debug settings.

The operation of debug can be sensitive to the Max Clock setting. Best to limit this to 1MHz or 2MHz.



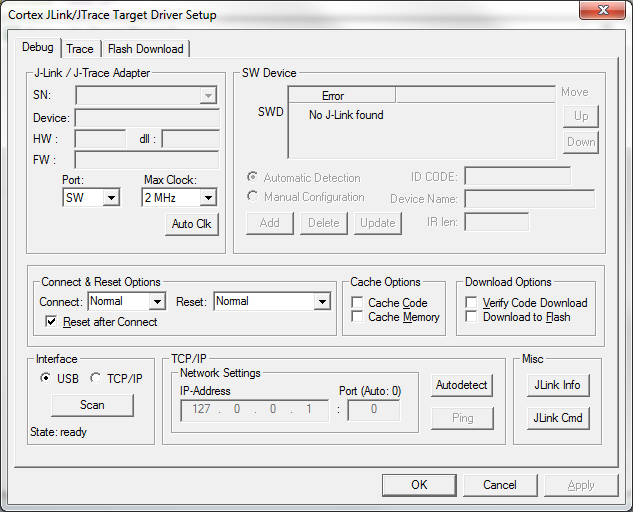
## Flash erase.

This appears to relate to using Argo PCBs which have previously been programmed with the Nordic Soft Device firmware. Select "Erase Full Chip" in the flash download Tab



## "Cortex M4 Error"

In the Debug tab uncheck the "Download to Flash" check box. Otherwise I was getting an unexplained error when starting debug.



# Setting up a new project by copying an existing project.

Instead of starting with a new project from scratch every time you want to try another simple piece of coding, or to try another DW1000 feature, it is possible to use the existing projects by copying the project and renaming it. The method is as follows:

1. Copy the existing project files into the new folder.
2. Rename the following files according to the new project name. .uvproj. .uvopt. .uvgui.

(Please note: handling former uV2-projects only requires renaming the file .uv2)

1. Enter the new project name at: Options For Target, Output tab, Name of Executable.

Before you proceed to add new code or new source files you should build and debug the new project and confirm it is all good before changing it.