ECE 3710 Lab 1 – Fall 2024

Due Date: Week of September 16 at the beginning of your lab section (20 points)

Objectives

Become familiar with the process of building, downloading, debugging, running, and modifying assembly programs on a microcontroller using an IDE.

Overview

In this lab you will run, modify, and debug your first assembly program on the STM32L476 microcontroller.

Preparation

1. Come to lab with your STM32 Nucleo-64 Board and lab kit.

2. Read Chapter 3 (ARM Instruction Set Architecture) in the textbook.

3. Read through the two Keil µVision tutorials from Circuits Today (Keil Debug & Keil Setup) posted on the Lab Supplements page on Canvas

Procedure

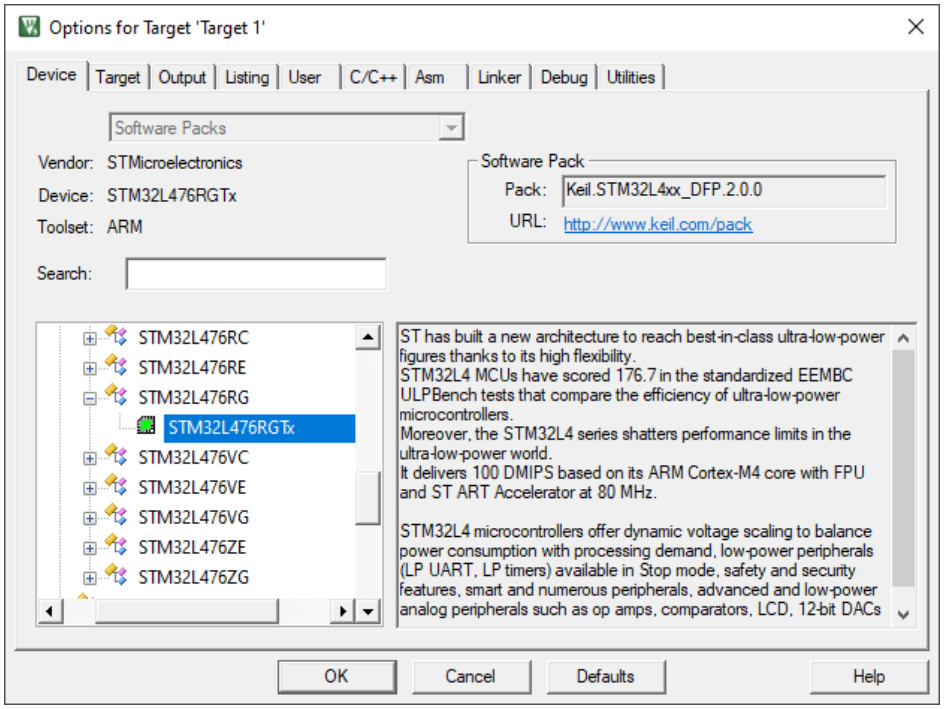
1. Create a new µVision project, build, and run a simple program that turns on the user LED on the STM32 Nucleo-64 Board.

    a. Review Lab 0 to recall the microcontroller pin that drives the green user LED on the Nucleo board.

    b. Start Keil µVision.

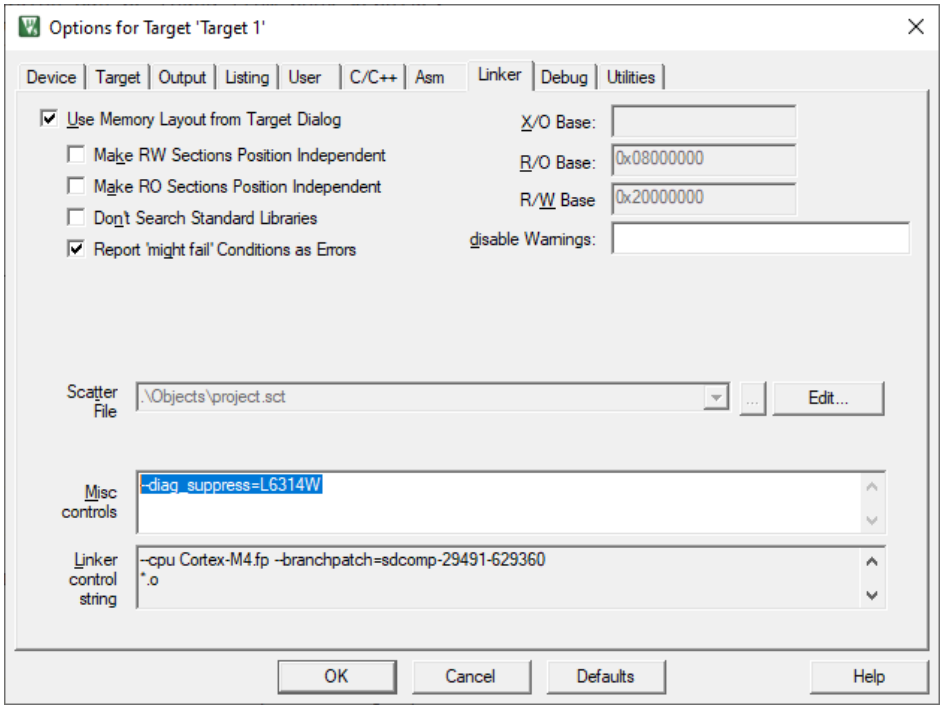
    c. Select **Project -> New µVision Project** and give your project a name and a path that make sense to you.

    d. Select the appropriate microcontroller. Our Nucleo boards use the STMicroelectronics STM32L476RGTx.

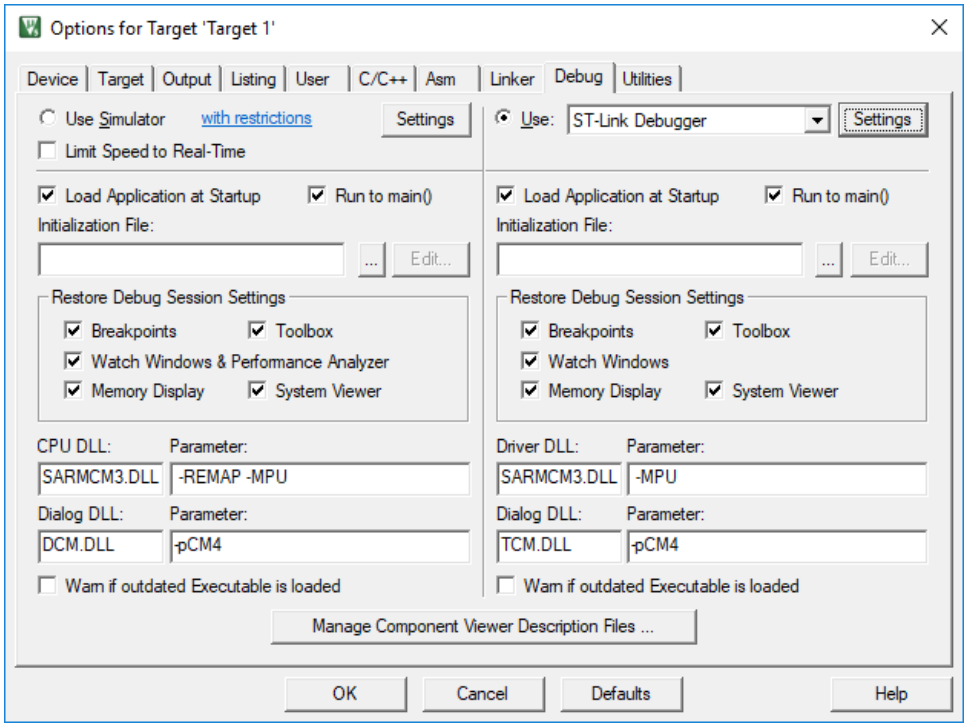


e. Use the default settings for any other prompts (Just click **OK**).

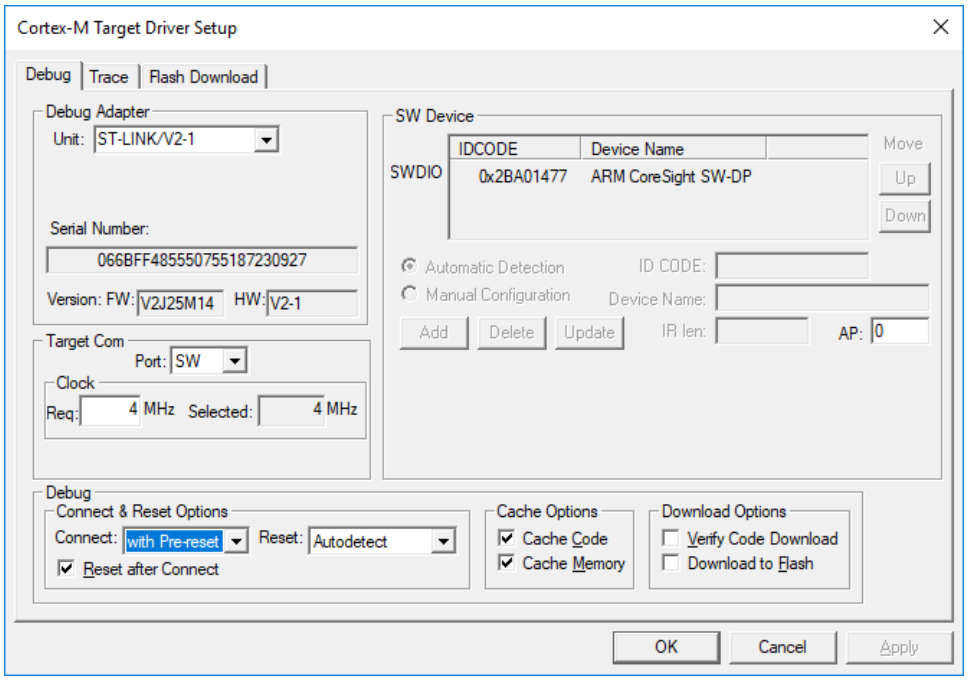
f. Click **Project->Options for Target 'Target 1'**, select the **Linker** tab, and make sure **Use Memory Layout from Target Dialog** is selected. The linker will fail otherwise. Also, paste the string "--diag_suppress=L6314W" into the **Misc controls** text box as shown.



g. Still in the Options dialog box, click on the **Debug** tab, select **Use** in the right-hand column, then select **ST-Link Debugger** from the drop-down menu.

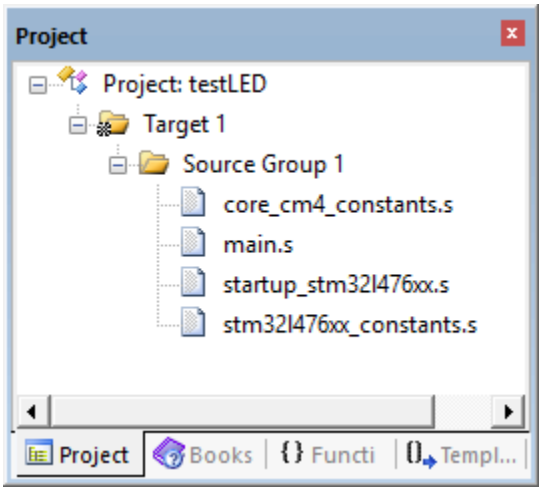


h. Click on **Settings** in the top-right corner, select the **Debug** tab, and select **with Pre-reset** in the drop-down menu of the **Connect & Reset Options** section (bottom left). Then click **OK** until all pop-ups are closed.



i. Copy the 4 assembly files (.s extensions) included in the zip file to your project directory.

j. On the project tab (left side of screen), expand the tree until you see **Source Group 1**. Right click on it, then select **Add existing files to Source Group 1**. Add the 4 files you copied in the previous step. Your Project window should now look similar to this:



k. Open **main.s** by double-clicking. This file contains ARM assembly code.

l. Click on **Project->Build Target**. Your project should build with no errors. Warnings are OK. If you get errors, notify the TA.

m. You will be using the onboard debugger to download code to and receive diagnostics from your board. The software drivers should already be installed on your host PC.

n. Select **Debug -> Start/Stop Debug Session** and your debugger will automatically launch. In the debugger you have access to all of the memory on your device. You can step through the code using **F10/F11** or you can let it run by pressing **F5**. You can set breakpoints by clicking in the left margin next to the line of interest. If the debugger fails to launch, talk to the TA.

o. Select **View -> System Viewer -> GPIO -> GPIOA**. This is the port that controls the green LED on your physical board. In the GPIOA window, look at the DATA register. You will see values change as you step through the code.

p. Press **F5** to run your program if you haven't already. Note the changing values in the GPIOA DATA register. Step through the assembly code and see if you can understand what is happening.

2. Commenting code

a. The provided main.s file is missing comments to explain what the code does. Add a meaningful comment to **every** line of assembly code.

3. Modify your code

    a. Modify the main.s file to continuously turn the green user LED on and off as fast as possible (it will just look dim to the naked eye). Modify your comments appropriately.

    b. Use the logic analyzer to measure the frequency of the blinking LED.

        i. Review the user manual for the logic analyzer, paying special attention to "Trigger Setup" (p. 87-93) in the MDO3000 Series Mixed Domain Oscilloscopes User Manual (refer back to the previous lab if needed).

        ii. ***Make sure you understand how to connect your board to the logic analyzer. Ask your TA if you are unsure BEFORE connecting.***

    c. **Pass-off:** Demonstrate your working blinking user LED and logic analyzer to the TA.

4. Lab Report

    a. Include a screen capture of the LED timing for the final version of your code.

    b. Include the final, well-commented version of main.s.

    c. Follow the rubric on Canvas for tips on how to structure your lab report.