Logo, company name

Description automatically generatedTutorial 1 - Getting started with

Keil MDK and NUC140 board

Objective

This guide provides instructions on how to set up the ARM Keil Microcontroller Development Kit (Keil MDK) on your Personal Computer (PC) and communicate with the Nuvoton Nu-LB-NUC140 embedded system development board (hereafter referred to as the **NUC140** board).

A couple of other notes:

* This guide might not work with other versions of Keil MDK from any other sources.
* It is essential to check the [Keil MDK System Requirements](http://www2.keil.com/system-requirements/) before installing this software on your PC
* You should have basic knowledge of PC and are familiar with installing software on a computer.

If you use a computer in Engineering labs, you can skip the first three steps.

Estimated Time to Complete – 45 minutes

Table of Contents

[Step 1 - Install Keil MDK v5.30 2](#_Toc104971896)

[Step 2 - Install Nuvoton NUC140 Software-pack 4](#_Toc104971897)

[Step 3 – Download Supporting Library 4](#_Toc104971898)

[Step 4 – Open a Keil Project 5](#_Toc104971899)

[Step 5 – Setup Nuvoton Board 7](#_Toc104971900)

[Step 6 – Configure Nulink Debugger 7](#_Toc104971901)

[Step 7 – Download and Test Program 10](#_Toc104971902)

[Homework 12](#_Toc104971903)

# Step 1 - Install Keil MDK v5.30

1. Download [Keil MDK version 5.30](https://drive.google.com/file/d/12nipY6H80kP6BheI-BWbPm5oZthcI1dx/view?usp=sharing) from Google Drive.
2. Run the executable file (i.e., MDK530.EXE) to install the Keil MDK. Following the next couple of screenshots until the end:

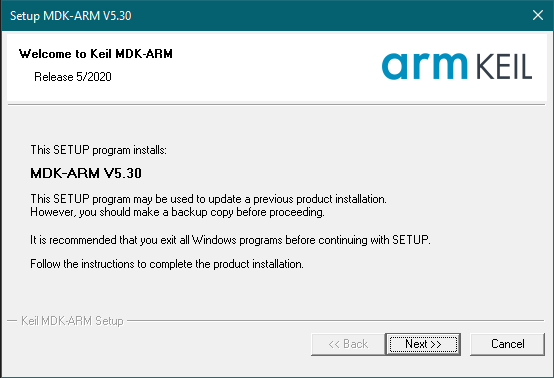


Figure 1: Execute the MDK530.exe file to start the installation process.

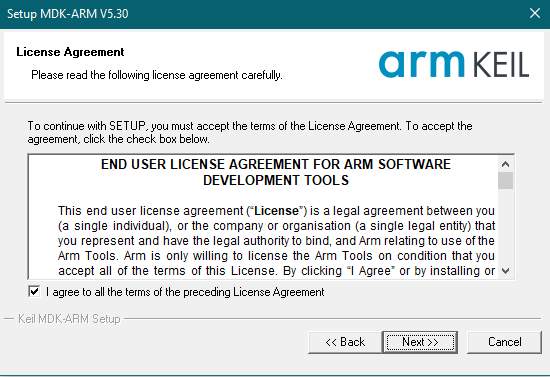


Figure 2: Agree on the License Agreement and click "Next"

Make sure you choose the destination folders as exact in Figure 3.

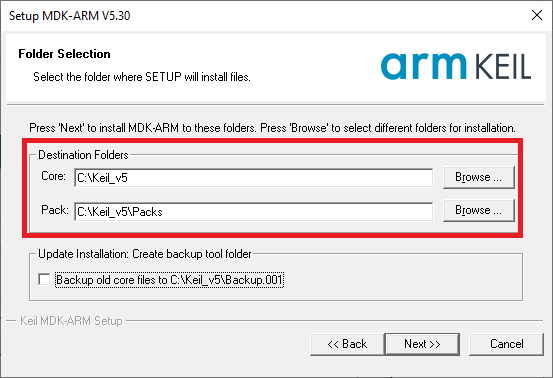


Figure 3: Choose the folder in your PC to install the Core software and device-specific packs

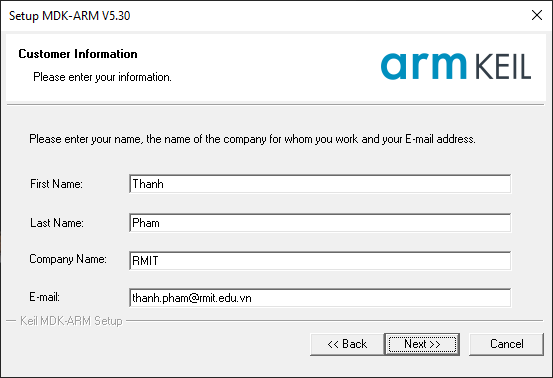


Figure 4: Provide your personal information. Click “Next” to start and complete the Core software installation

# Step 2 - Install Nuvoton NUC140 Software-pack

1. Download the [software](https://drive.google.com/file/d/1XBhDQD1Ya4q3QKr89Wuh2ceKJWi00XIV/view?usp=sharing) pack.
2. Execute the file, and then install. The software pack will pick up the folder where you select earlier.

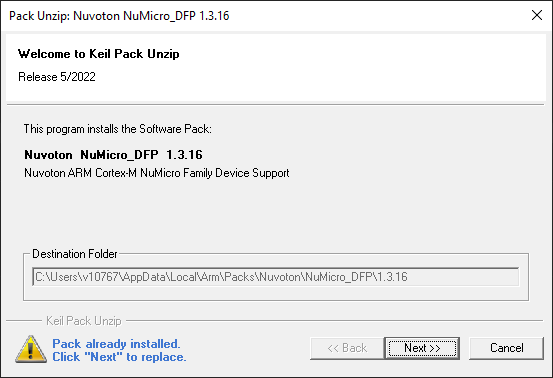


Figure 5: Install NUC140 Software Pack

After this step, we have successfully installed the core and device-specific Packs of Keil MDK required to work with the NUC140 board.

# Step 3 – Download Supporting Library

1. Download and decompress the [achieved library files](https://drive.google.com/file/d/1LkLOLWiUyQ-0J2yU38AAPuGL4EDkUpjm/view?usp=sharing) to the same folder as the Keil MDK earlier. This folder stores setup files for the embedded board in the following tutorial.

A screenshot of a computer

Description automatically generated with medium confidence

Figure 6: Extract the Library folder to the Keil MDK folder.

# Step 4 – Open a Keil Project

1. Launch **Keil uVision5**

Graphical user interface, application, Word

Description automatically generated

Figure 7: Keil main window

1. Download the [Tutorial1\_TestKeil.zip](https://drive.google.com/file/d/1I2Z3WpgMiaBKgSQksw7PlLbEZolYj5fg/view?usp=sharing)
2. Create a Master folder for this course, name it **EEET2481\_2022B**. Please read and strictly follow this advice:

* For each new project, you are strongly recommended to create a sub-folder in your PC to store its data. DO NOT place multiple projects in ONE folder
* Give your project a meaningful name
* In all cases, please avoid using special characters (e.g. whitespace, slashes, asterisk, etc.) in the project folder and project name.

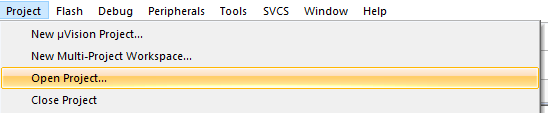
1. Move the zip file to this master folder and then extract here.

Graphical user interface, text, application

Description automatically generated

Figure 8: Folder arrangement

1. In the Keil main window, **Project 🡪 Open Project**, **browse to the project in Tutoria1\_TestKeil**



Graphical user interface, text, application

Description automatically generated

Figure 9: Open new project in uVision 5

1. You should see the project open as follows.

Graphical user interface, text, application

Description automatically generated

Figure 10: Project open in Keil

# Step 5 – Setup Nuvoton Board

1. Next, we will connect the board with the laptop via the USB cable. You should see the power LED on.

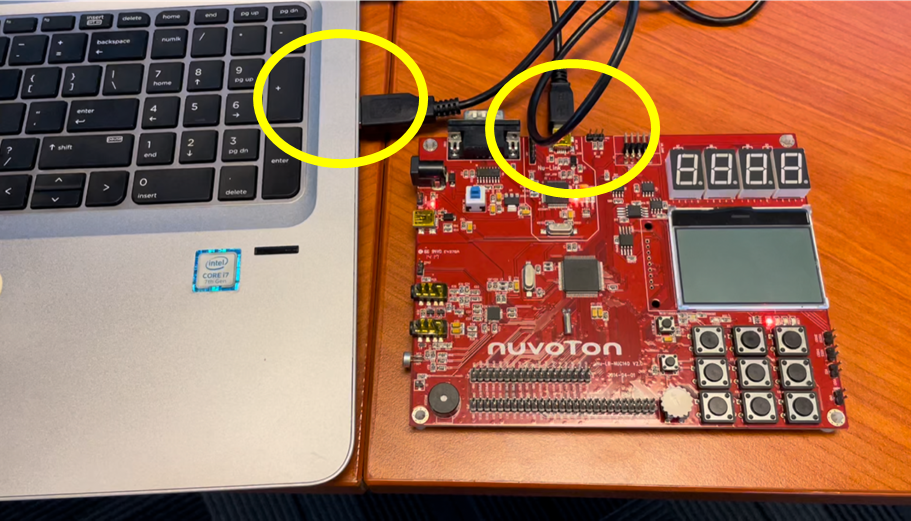


Figure 11: Setup the board

# Step 6 – Configure Nulink Debugger

1. In the project window, right click on **Target 1** and select **Options for Target ‘Target 1**’.

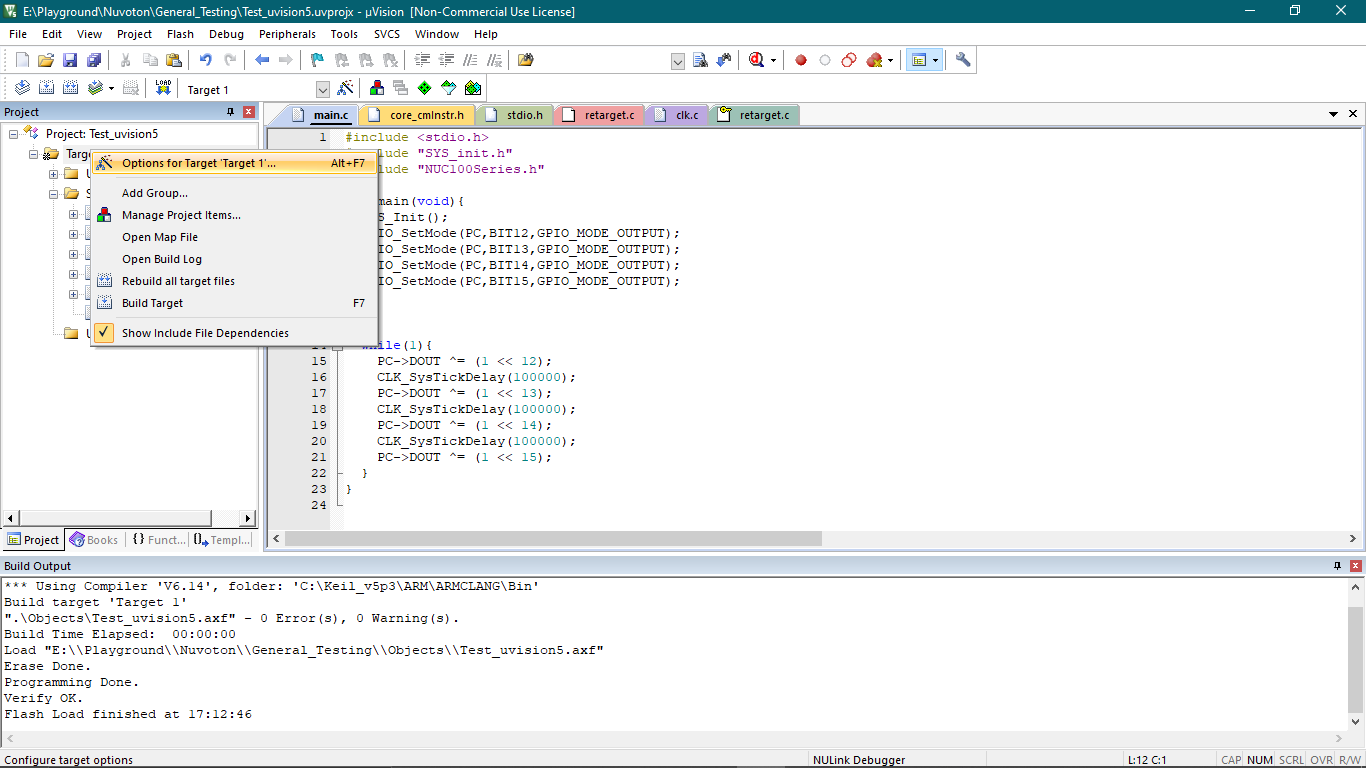


Figure 12: define debugging, search path, etc. options for the embedded software project

1. In the window which has just appeared, the **C/C++(AC6)** tab needs to be selected. This is where we need to include all directives where the header files (.h) are located - this is done in the Include Paths field. If you setup the directories as I recommended earlier then we should have no issues here.

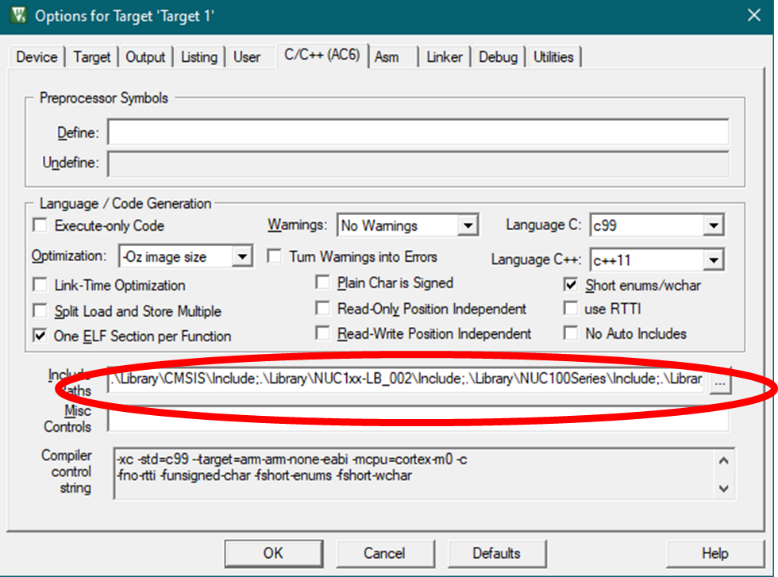


Figure 13: Include Paths

1. Select the **Debug** tab. In the drop-down menu, select **NULink Debugger** (this is the debugger on board the Nu-LB-NUC140). The debugger is a tool which can be used to step through the program code (line by line) which is running on the microcontroller in real-time. As a result, there are various tools which can be used to analyse the various activities occurring on the microcontroller. To see if this is working, click on **Settings**. If this is working correctly, you will be shown a window which displays various details of the Nu-Link driver setup. Also, an indicating LED on board will be shown as well.

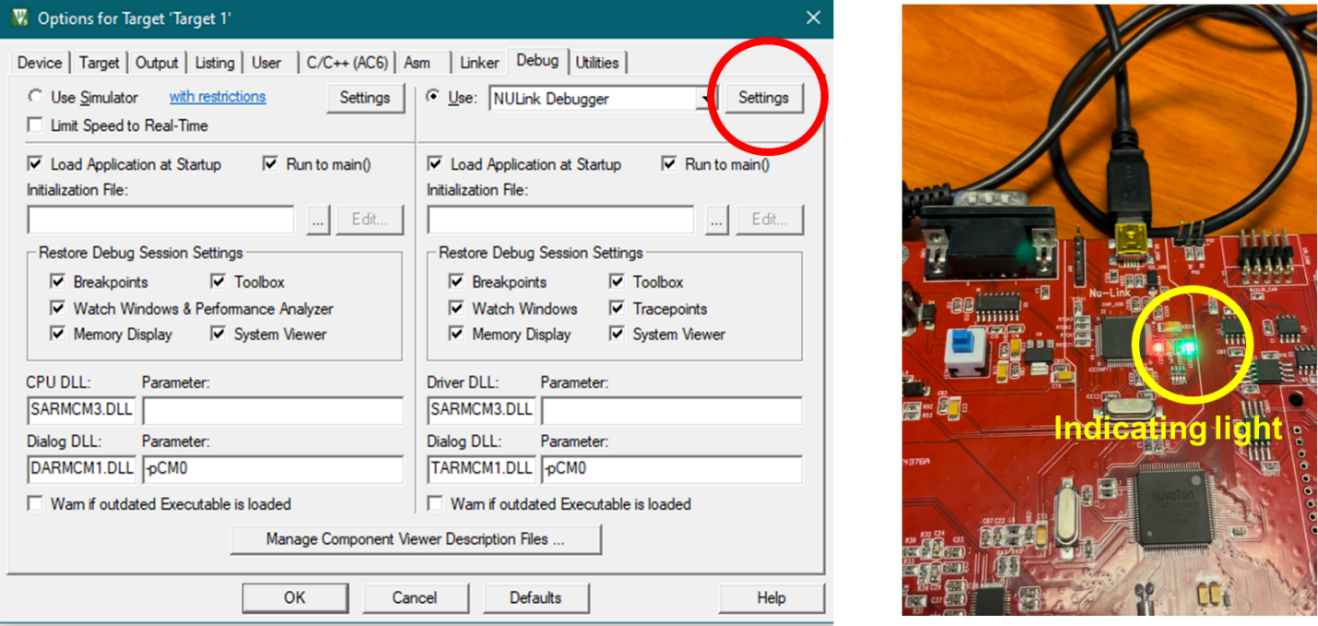


Figure 14: Check Debugger setting

Graphical user interface

Description automatically generated

Figure 15: Debugger information

1. IF you are prompted with a message saying a version mismatch has been detected and the firmware on the microcontroller is out of date, in this case you will need to update the firmware - click OK and follow the instructions (you will need to unplug and plug the board when asked to do so). Once the firmware is updated, click on settings again to view the Nu-Link driver details.

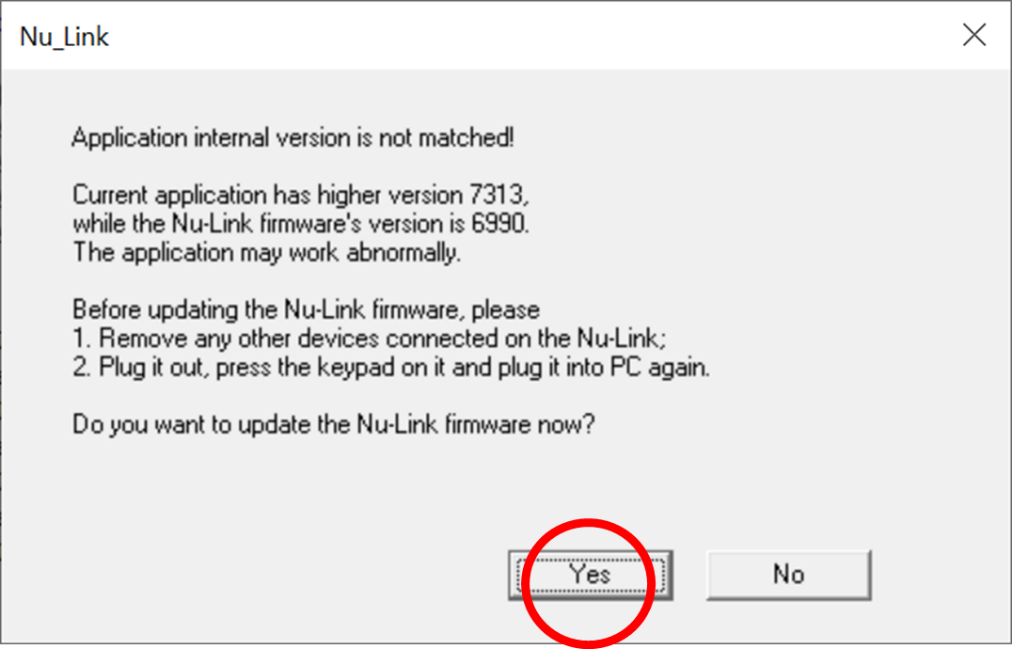


Figure 16: Update firmware for the Debugger if you are being asked.

1. Click **Ok** to close the Option window

# Step 7 – Download and Test Program

1. Now that the program has been written, the project needs to be built. Click on the **Build** button or press F7 to build the project. After the build process has completed, we need to see if it has failed or succeeded. If you see no errors, then we are good for now. We can ignore the warnings.

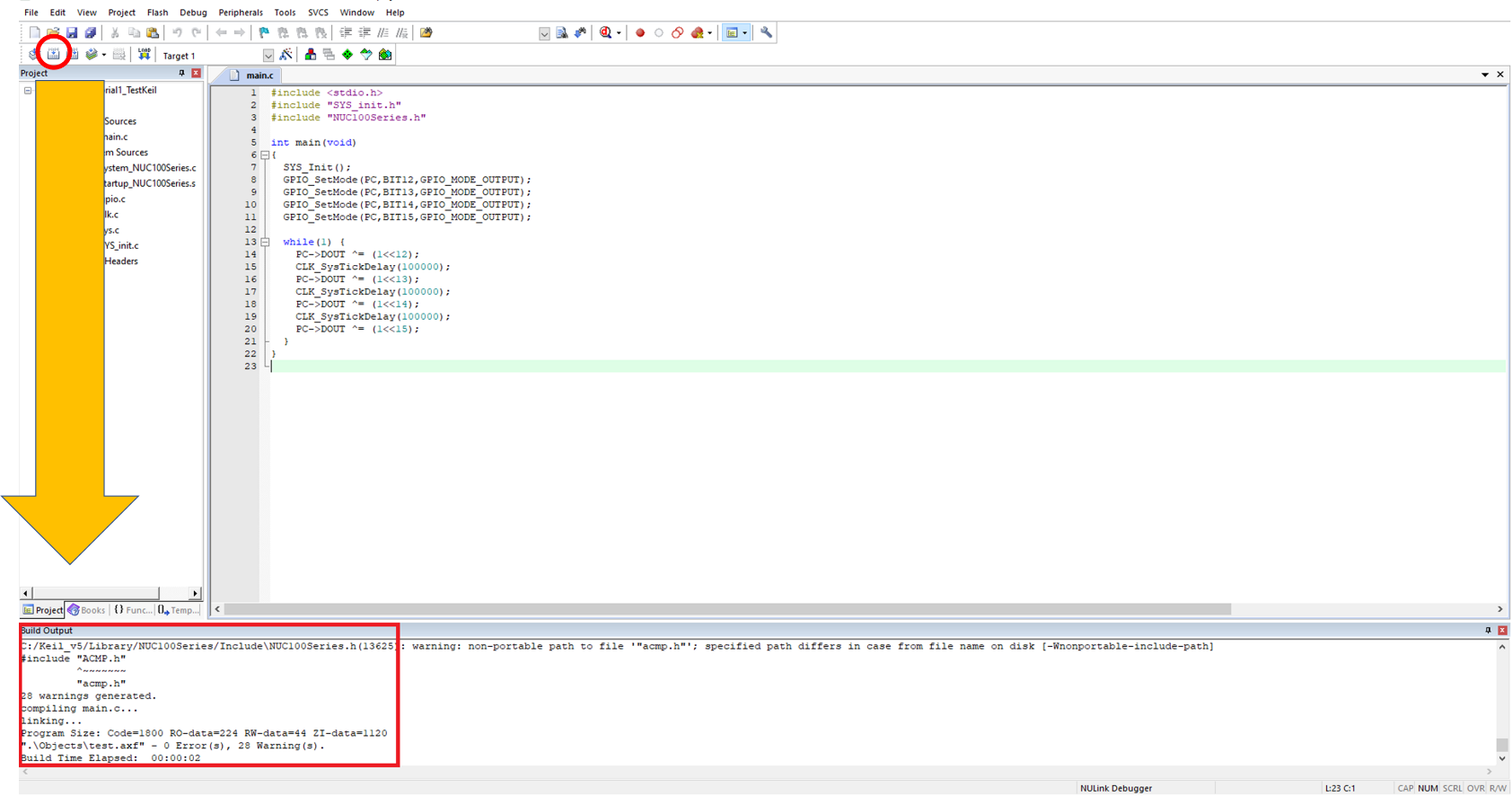


Figure 17: Build the project and Build Output window

1. The success build will generate a machine code version of the embedded program we have written (i.e., either in Binary or Hexadecimal format).
2. To load the embedded program into the microcontroller (i.e., to program the MCU), click the **load** button or press **F8**. You will see a green light on your learning board, this indicates that the board is busy uploading code to the flash memory. Wait until this green light has turned off. You will also be able to track the upload progress in the build output window. In this process, you can observe the load process has a few steps:
   1. Current program stored in flash memory is erased
   2. New program is uploaded into flash
   3. New program is verified with the program stored on the computer (if these are identical, the upload process was successful)

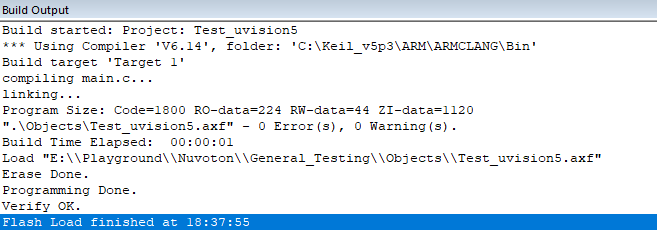


Figure 18: MCU build report

1. To see your program running, you will need to press the reset button on your microcontroller learning board

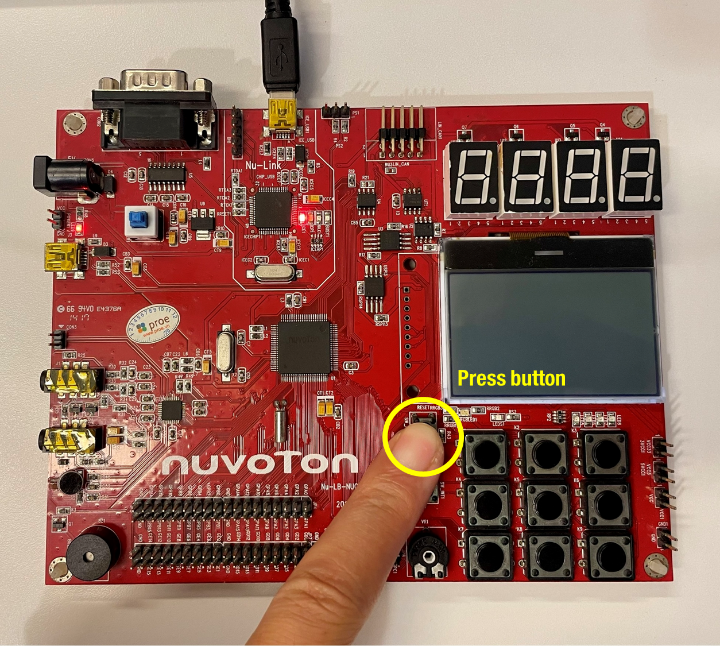
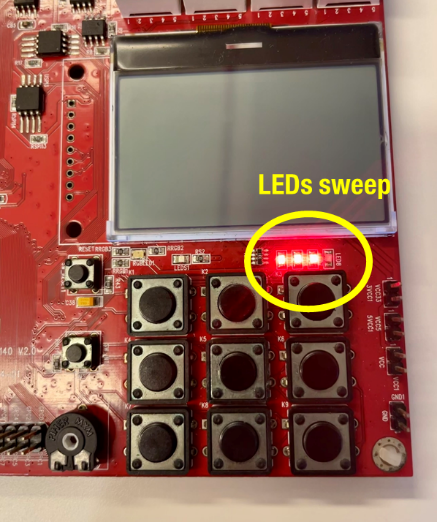
 

Figure 19: press the reset button to start the NUC140 board. See some LEDs flashing!

1. The program earlier is to sweep through 4 LEDs on board with a fixing delay time between each LED. IF you can see the LEDs sweeping on the board, it proves the program has been successfully developed and tested on the hardware. This process will be important for us to apply on incoming tutorials and laboratories. In term of the code, we will discuss in detail, what the existing program does and how to develop one in the next tutorial. You can check out the code as follows. You can double click on the Figure 20 and then copy the code to the IDE if you wish.



Figure 20: Code for the Tutorial1

# Homework

* Complete the all the steps in the Tutorial note if you could not complete in class.
* Inform lecturer if you have any issues.
* You can also check the video version of this Tutorial on Canvas (around 20 minutes long)