

Microcontrollers & Microprocessors Starter Guide

Advanced Computing Lab HCMUT

https://www.fb.com/groups/aclabhcmut

Table of contents

Set Up Your Environment.	2
IDE: STM32CubeIDE.	2
STM32 ST-LINK Utility.	2
Proteus.	2
VSCode and Platformio(Optional).	2
Lotus STM32F103 Dev Board Overview.	3
ST-LINK & STM32F103 MCU.	3
Built-In Leds and Buttons.	4

Set Up Your Environment.

IDE: STM32CubeIDE.

STM32Cube IDE is a graphical tool for:

- Configuring STM32 peripherals and generating initialization code.
- Setup of peripherals, clock trees, and middleware like USB or RTOS.
- Support Debugging.



You need to create a MyST account to use the software.

STM32 ST-LINK Utility.

For the Lotus STM32F103RB board, we need STM32 ST-LINK Utility to update ST-LINK to the latest version.

Proteus.

Proteus is a simulation software used for designing and testing electronic circuits. In this course we will use proteus to test our firmware.

VSCode and Platformio(Optional).

We can also use Visual Studio Code with the PlatformIO extension for coding. However, for beginners, we recommend STM32CubeIDE as it provides a graphical interface for configuring STM32 peripherals and generating initialization code, making development more intuitive.



Template for Lotus F103 Board: https://github.com/ACLAB-HCMUT/Template_LotusF103RB_Platformio

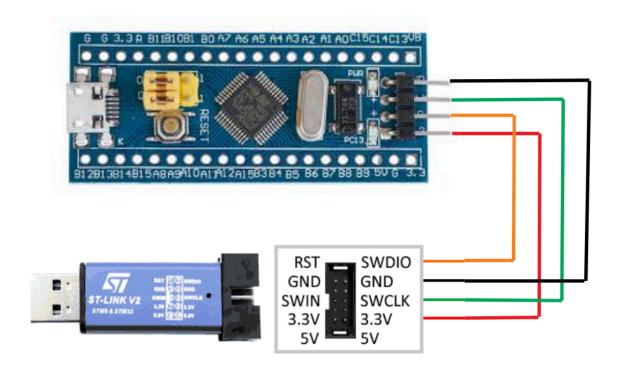
Lotus STM32F103 Dev Board Overview.

ST-LINK & STM32F103 MCU.

The Lotus F103 and the NUCLEO-F103RB have similar flash/debug interface to the picture below:

ST-LINK:

- ST-LINK is a hardware debugger and programmer tool developed by STMicroelectronics.
- It connects to the MCU via JTAG or SWD (Serial Wire Debug) interfaces, enabling developers to load firmware, debug code, and perform memory operations.





ST-LINK is also made using a STM32 MCU with firmware developed by STMicroelectronics.

We connect to ST-LINK via USB interface, ST-LINK then connect to our main MCU via SWD interface.

Built-In Leds and Buttons.