CAN ADAPTOR

CAN Adaptor with STM32F072C8T6

Purpose of the Document

The purpose of this document is to explain the technical specifications and manual for using the USB dongle board powered by STM32F072C8T6 and CAN transceiver.

Document History

Version	Author	Date	Description
A	5G HUB	14.04.2024	Initial Document

Table of Contents

1 Package contents:

1.1 STM32 USB Dongle Package:

- STM32F072C8T6 USB dongle.

1.2 Download

Install STM32CubeProgrammer for Windows.
https://www.st.com/en/development-tools/stm32cubeprog.html

Arduino software can be downloaded from the following website:
https://github.com/5ghub/stm32

To use the board with Arduino IDE and starts running Arduino projects and sketches, install the following software:

Install Arduino IDE for Windows from the following web site
https://www.arduino.cc/en/Main/Software

Download and Install LTE&GNSS modem driver for Windows OS:
https://github.com/5ghub/5G-NB-IoT/tree/master/Driver

Download and Install QNavigator and QCOM tools for Quectel BG95 here:
https://github.com/5ghub/5G-NB-IoT/tree/master/Tools

2 General Description

2.1 Overview

This is a USB dongle powered by STM32 MCU and CAN transceiver. It is used as CAN adaptor for CAN applications. The USB is designed as compact and complete development platform for STMicroelectronics ARM cortex-M0 core-based STM32F072C8T6 microcontroller with CAN transceiver, I2C, SPI, USART, CAN, 12-bit ADC, 12-bit DAC, GP comparators, internal 16KB SRAM and 128KB Flash, USB FS, Touch sensing, SWD debugging support.

The CAN adaptor is a USB board used as a standalone STM32/CAN board or used as a motherboard where other daughter boards can be stacked on it. Daughter boards can be GNSS, Cellular, and other modules.

The USB leverages and utilizes all hardware features of STM32F072C8T6 and ISO1050DWR CAN transceiver. The USB is designed for the evaluation of all the peripherals and the development of user-specific applications. Extension headers are used to easily connect a daughter board to it.

The USB board is a rich hardware board that can be used to stack other daughter boards such as 4G LTE or GNSS wireless technology and enables a variety of smart applications for devices. It enables large number of applications such as wireless POS, smart metering, tracking, smart transportation, smart buildings, smart city, and smart homes, CAN scanners, On-Board vehicle Diagnostics (OBD).

The board is also compatible with Arduino and Arduino software (IDE). Arduino sketches and examples are provided with the kit and additional sketches can be developed and uploaded to the board.

2.2 Key Features

- STM32F072C8T6 MCU
- ISO1050DWR CAN transceiver
- USB full-speed connector
- 5V USB power supply
- I2C, SPI, and UART interface
- CAN2.0A/B compliant interface
- SWD debug support
- Motherboard and daughter board switch button
- Extension connector for daughter board or wrapping board
- Analogue and Digital GPIOs
- Two user-programmable LEDs
- Compact board size of 59 mm x 25mm
- Arduino IDE Compatible
- Works with Windows or Linux
- Ready as a CAN adaptor, CAN applications, smart applications and development (smart home, smart city, smart transportation, smart metering, smart farming, smart waste management, asset tracking, location, navigation, mapping, and timing applications). Application such as Vehicle CAN reader, OBDs, CAN adaptor, Gas Detector, Soil PH Tester, Optical Sensor, Machinery Alarm System, Irrigation Controller, Elevator, Asset Tracking Electronics, Person/Pet Tracking, Water/Gas Metering, Smart Parking System, Fire Hydrant, Smoke Alarm, Trash Bin, Street Lighting

2.3 Overview Diagrams

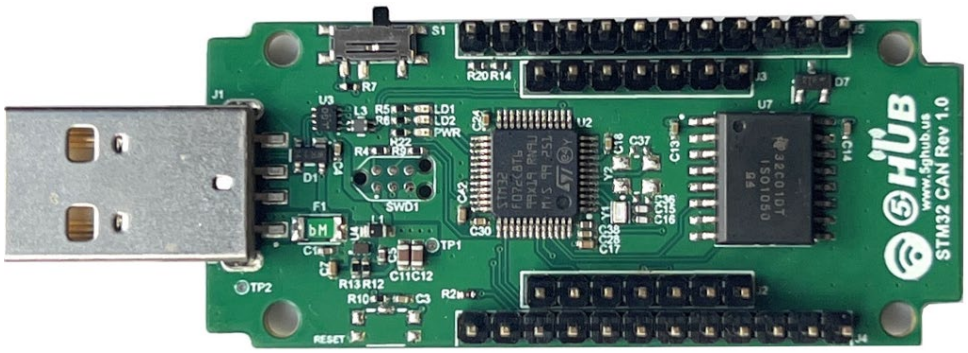


Figure 1. CAN Adator Overview Diagram – Top View

Figure 2. CAN Adator Overview Diagram – Bottom View

2.4 Physical Characteristics

The width and length of the USB modem is 25 mm (width) by 59 mm (length). The board have two screw holes in each corner that allows the board to be attached to a surface or case.

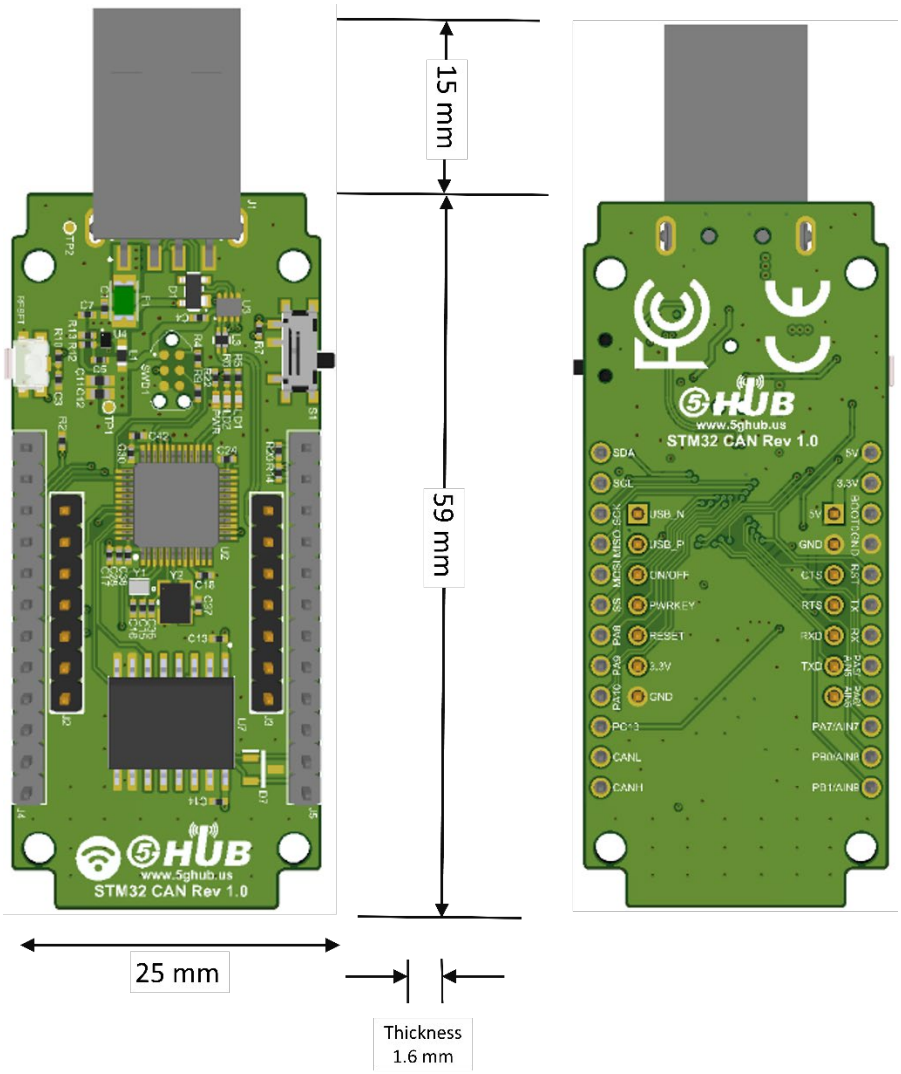


Figure 3. Physical Characteristics.

2.5 Peripherals – Key Components

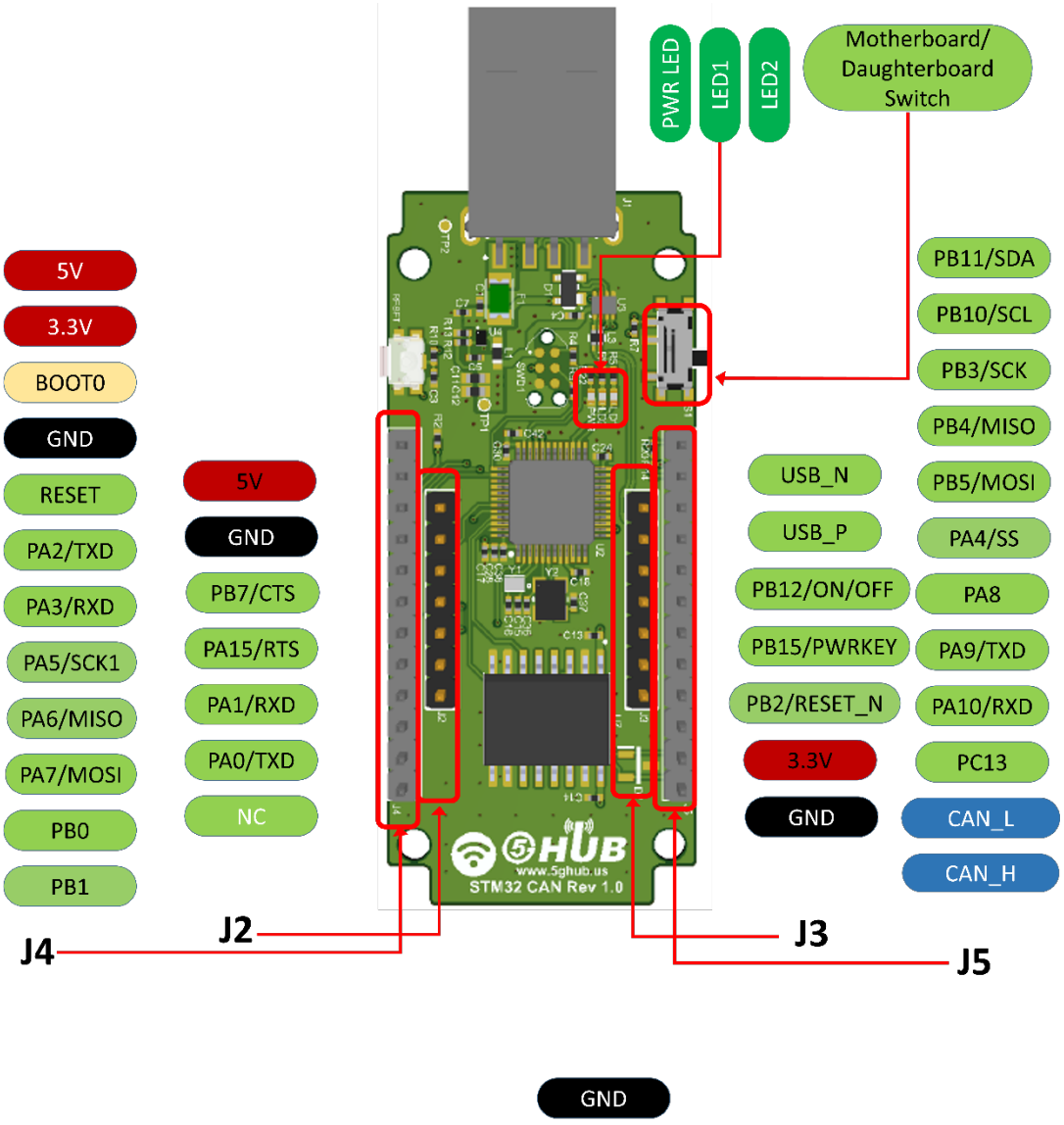


Figure 4. CAN Adaptor Top Side – Key Components

2.6 Hardware Specification

Technical Specification	
Microcontroller (MCU)	STM32F072C8T6, 32-Bit ARM Cortex M0+
Clock Speed	48 MHz
Flash Memory	128 KB
SRAM	16 KB
CAN Transceiver	ISO1050DWR
Dimension	25 mm (width) by 60 mm (length)
Weight	20 grams
Power Supply	USB (5V)
LED	Power LED, LED1, LED2
Interfacing Logic Voltage Level (Operating Voltage)	3.3V
Voltage output	5V, 3.3V
RESET buttons	Not Mounted
USB Switch	1 switch to connect to MCU directly or BG95 directly
General-purpose digital I/O Pins	32 (PA0-PA15, PB0-PB15)
USB	1
I^2C	1
SPI	1
UART	1 (with ISO7816 interface, LIN, IrDA, auto baud rate detection and wakeup feature)
CAN	1
ADC pins	1 (12-bit ADC)
DAC pin	1 (12-bit DAC)
External interrupts	14 (All general-purpose PINs)
PWM pin	6
DC Current per I/O Pin	10 mA
JTAG Debug	Cortex Debug Connector (Single Wire Debug)

2.7 PIN Description

PIN	DIRECTION	Description
USB Connector	I	The USB modem is powered from the USB port (3.8V-5V)
LED1 (USER)	O	LED which can be controlled from MCU (D25). When the pin is HIGH value, the LED is on, when the pin is LOW, it is off
LED2 (USER)	O	LED which can be controlled from MCU (D26). When the pin is HIGH value, the LED is on, when the pin is LOW, it is off
USB Switch	I	1 switch to connect to MCU directly or BG95 directly
3.3V	O	3.3V generated by the on-board regulator. Maximum current drawn is 3A. The regulator also provides power to the MCU and BG95
5V	O	5V generated from the board. The board is supplied with power from USB connector (typical 5V)
GND		Ground
PA5_AIN5	IO	Six analog inputs which can provide up to 12 bits of resolution (i.e. 4096 different values). By default, each input measures from ground to 3.3 volts.
PA6_AIN5	IO	
PA7_AIN5	IO	
PB0_AIN9	IO	
PB1_AIN9	IO	
PA8	IO	
PA9	IO	GPIO
PA10	IO	GPIO
SCL	IO	GPIO
SDA	IO	I^2C. The SCL (clock line). Can be used as GPIO
SCK	IO	I^2C. The SDA (data line). Can be used as GPIO
MISO	IO	SPI Interface. Can be used as GPIO
MOSI	IO	SPI Interface. Can be used as GPIO
SS	IO	SPI Interface. Can be used as GPIO
CAN_L	IO	SPI Interface. Can be used as GPIO
CAN_L	IO	CAN bus interface
Cortex Debug Connector	IO	GPIO
BOOT0	I	Using Single Wire Debug to burn bootloader and debug the board
		Connected to BG95. Force the BG95 to enter emergency download mode

Precaution

The USB modem runs at 3.3V. The maximum voltage that the I/O pins can tolerate is 3.3V. Applying voltages higher than 3.3V to any I/O pin could damage the board

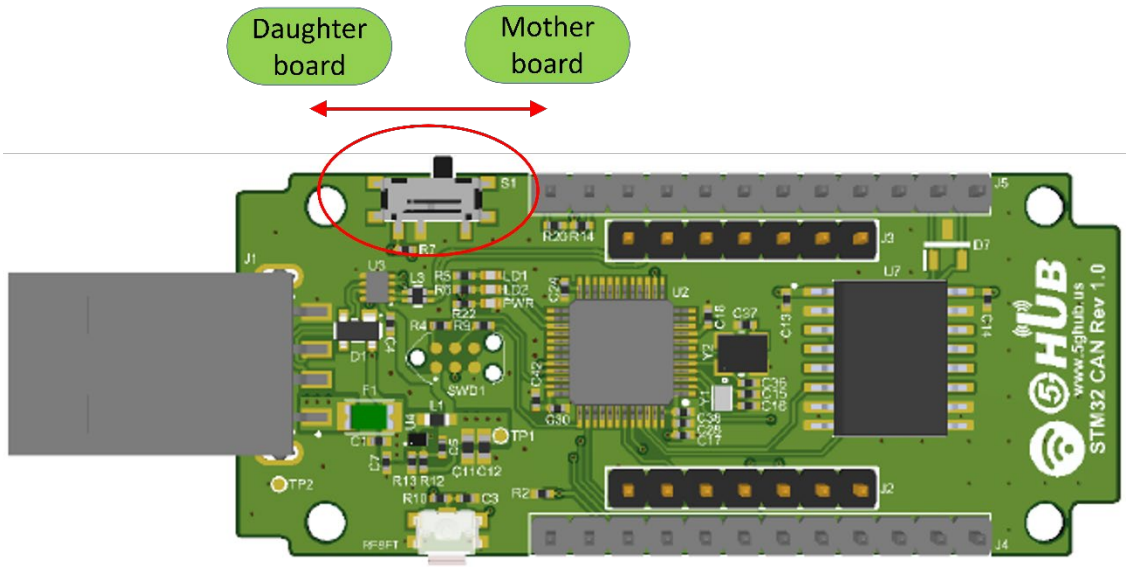
3 Using the Board with Arduino IDE

3.1 Installing the Software

To use the board with Arduino IDE and starts running Arduino projects and sketches, install the following software:

3.2 Setting Up the Board

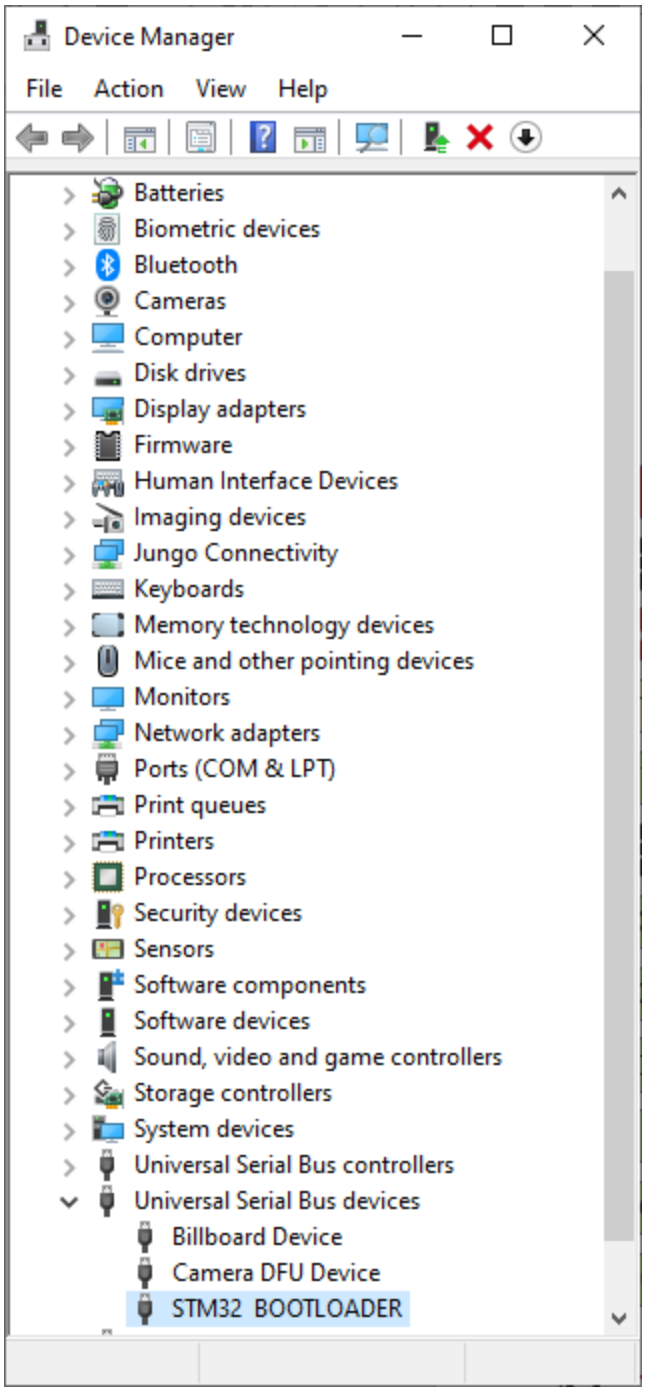
Gently move the switch S1 left or right to make the USB port either connect to Motherboard (STM32) or daughter board.



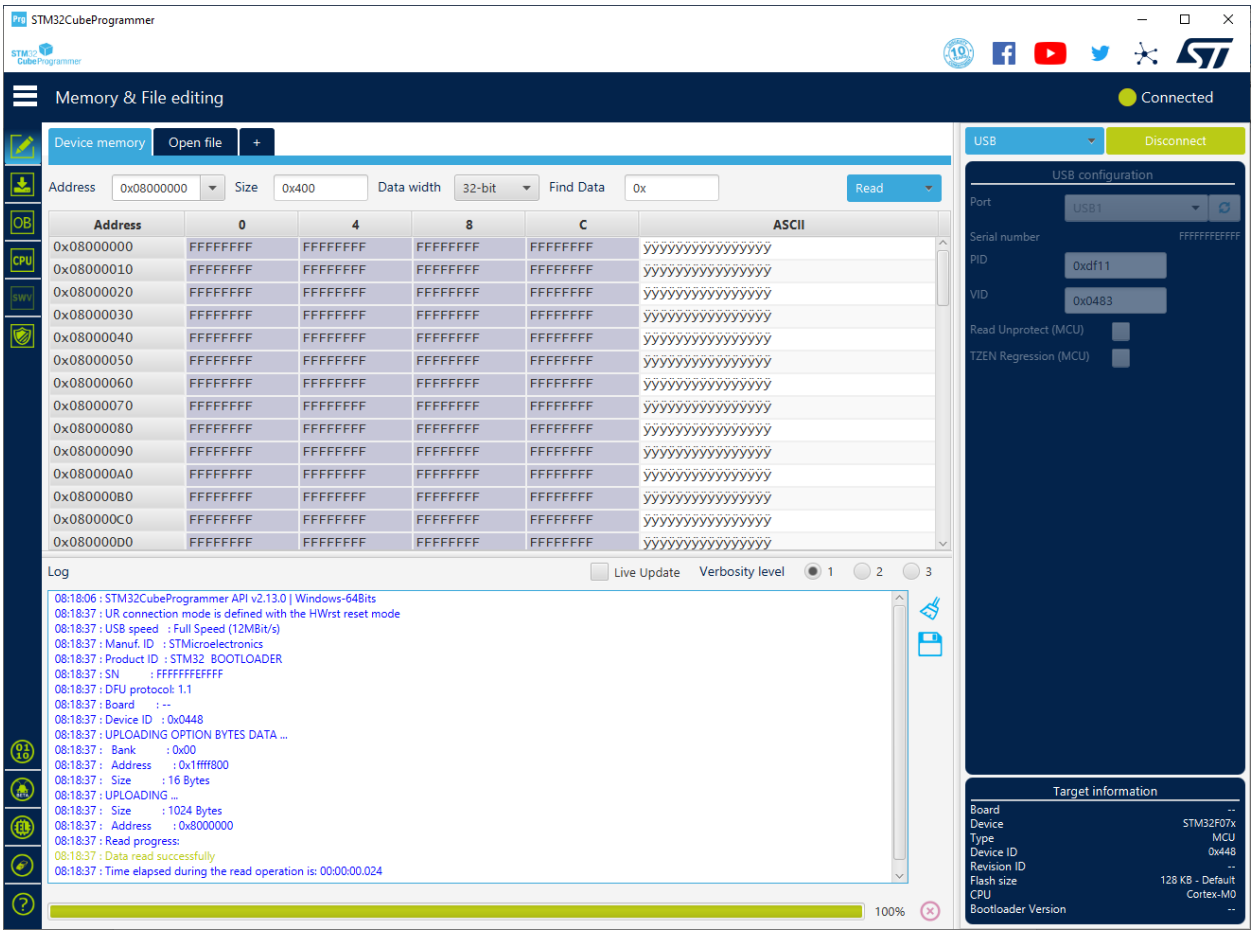
3.3 Setting Up STM32 Bootloader

Make sure to install STM32CubeProgammer. Do the following:

1- Connect a shunt into PIN2 and PIN3 on J4
2- Insert the USB dongle into USB port on a PC.
3- Launch windows device manager, and you shall see the STM32 bootloader.



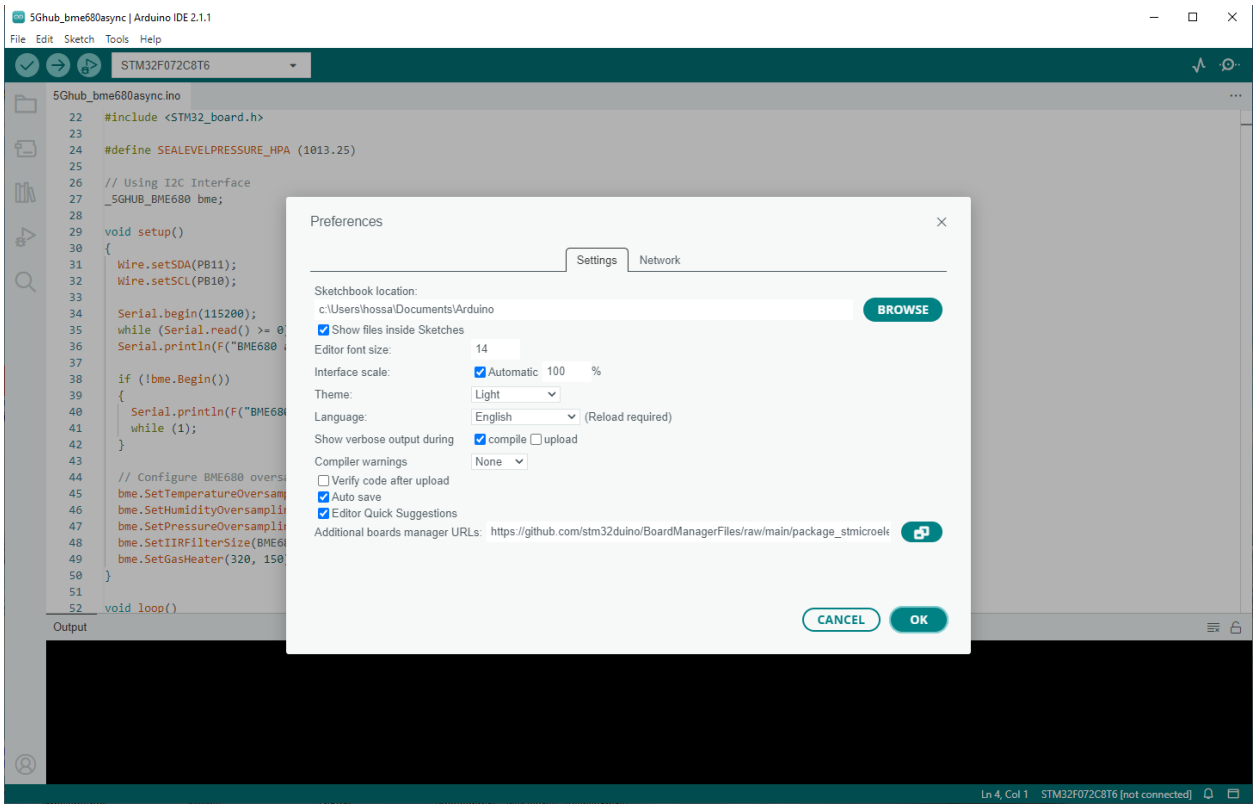
4- Launch the STM32CubeProgammer, select SUB, and click Connect. You shall see the STM32F072 connected.



3.4 Setting Up Arduino IDE

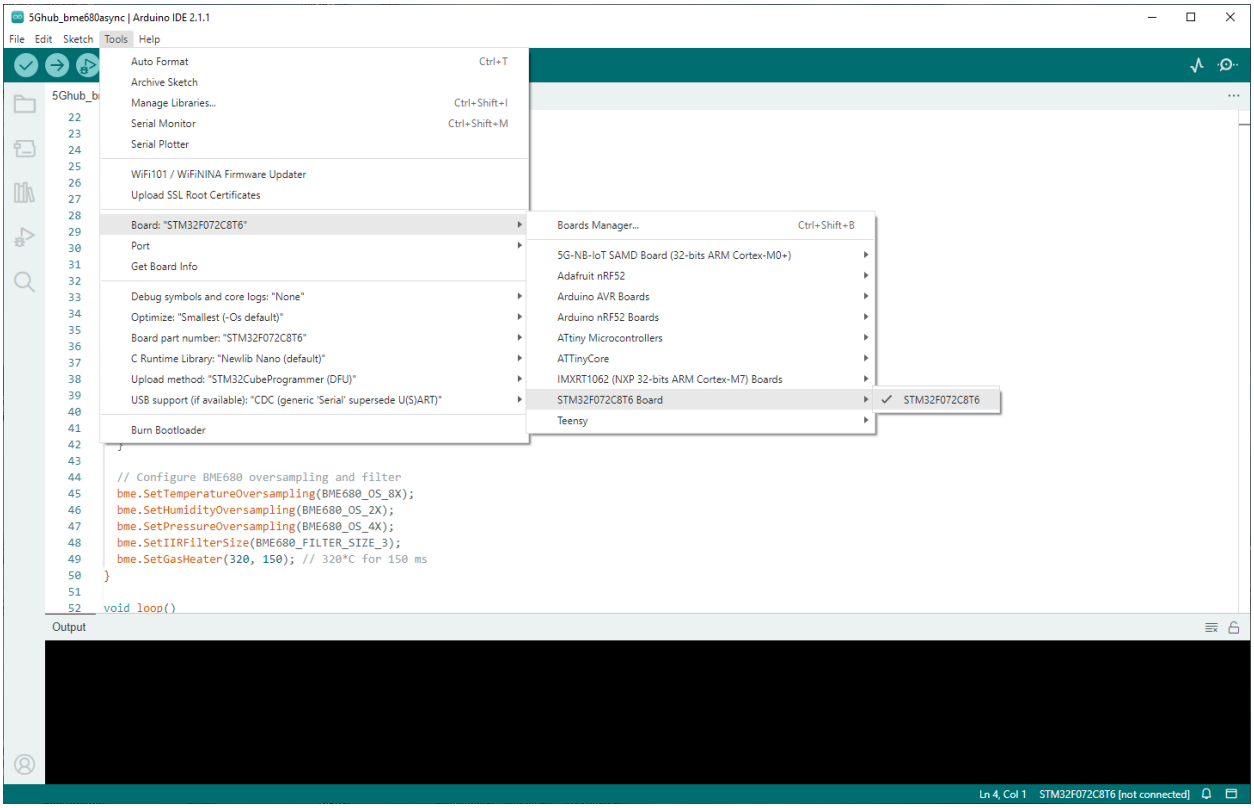
1- Launch Arduino IDE and choose **File->Preferences**. In the Additional Boards Manager URLs, insert the following URL:

https://github.com/stm32duino/BoardManagerFiles/raw/main/package_stmicroelectronics_index.json



2- In Arduino IDE, choose **Tools->Board->Boards Manager**, select and install "**STM32F072C8T6 Board**".

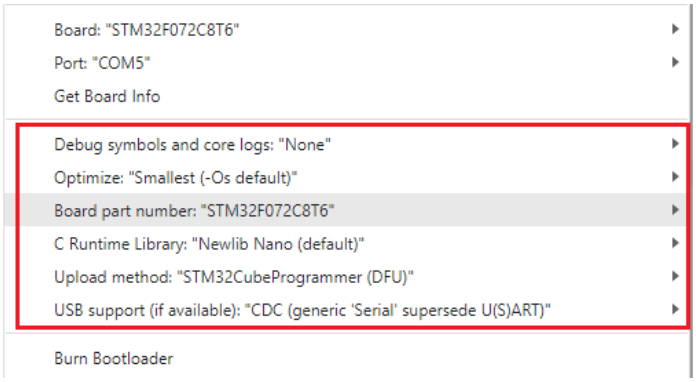
3- Choose "**STM32F072C8T6**"



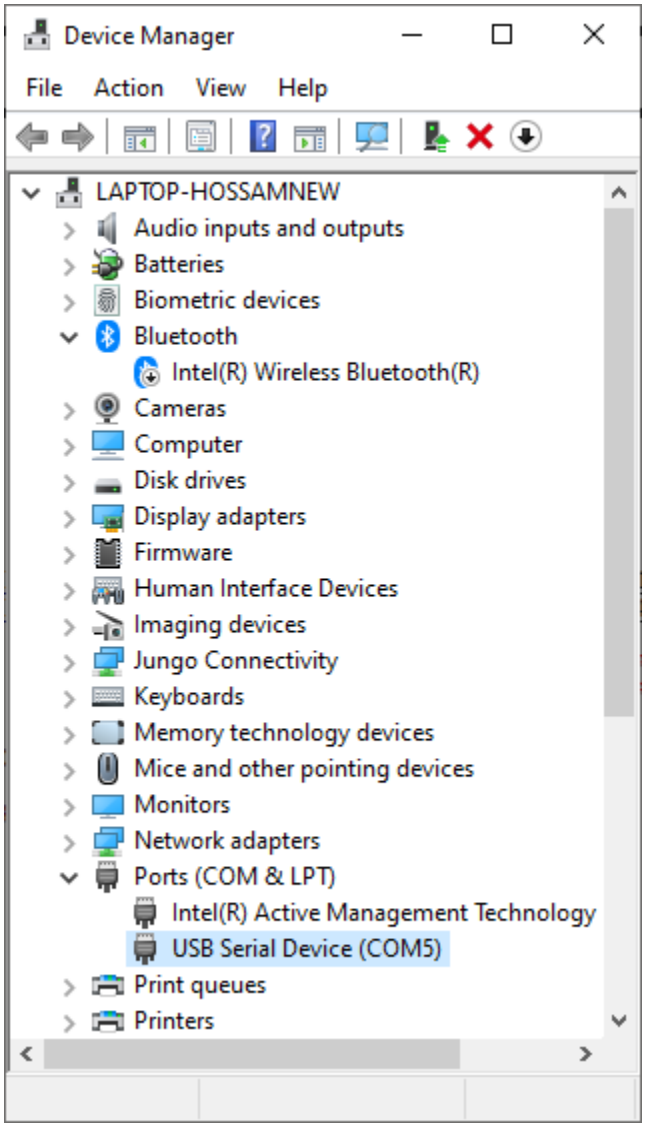
4- In the Arduino IDE, Choose **Sketch->Include Library->Add .Zip** Library and select the file STM32_Arduino.zip

5- You are ready now to use the Arduino IDE and write the first sketch. You can start compiling Arduino sketches and upload the sketch to the USB dongle.

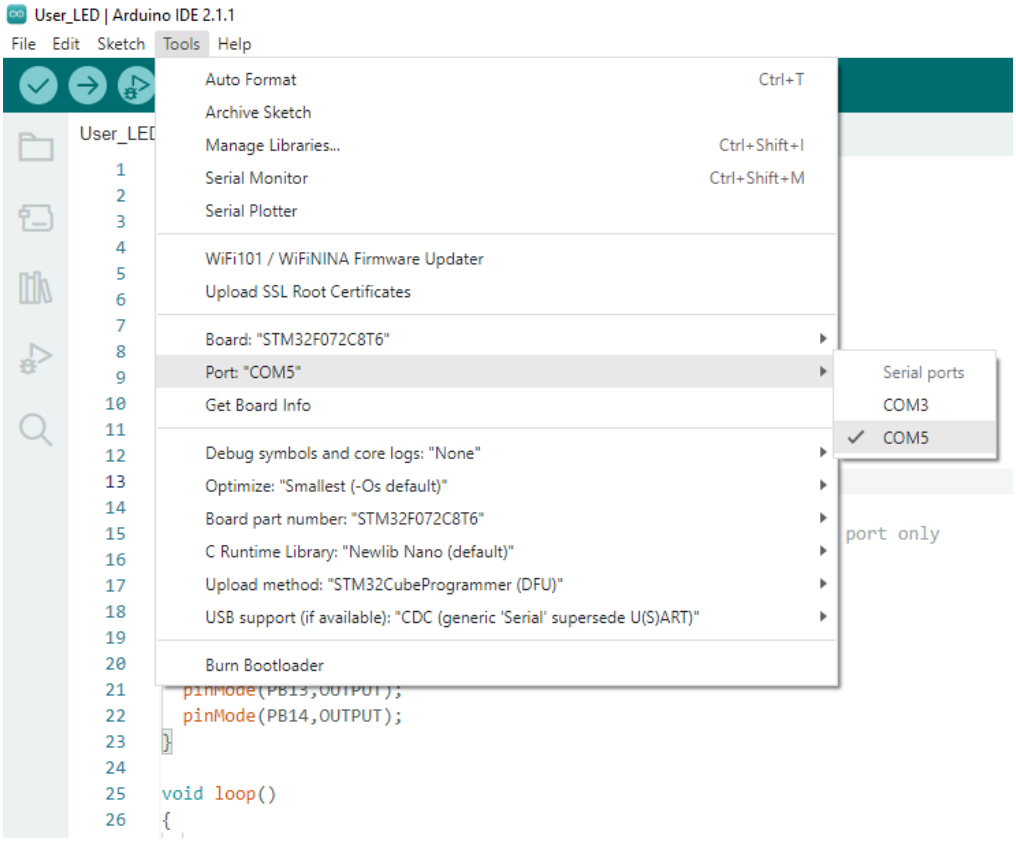
6- When uploading an Arduino sketch, make sure the following setting are selected:



7- After uploading an Arduino sketch, unplug the shunt, and re-insert the USB dongle again into the USB port. Windows will recognize the USB dongle as a new COM port as show here:



8- In Arduino IDE, choose **Port->COM5**.



Now, you can enable Arduino IDE Serial Monitor and see the Arduino sketch running on the board.

NOTE:
When uploading Arduino sketch to the USB dongle, a shunt must be connected between PIN2 and PIN3 of J4 (i.e, the STM32 MCU must be in BOOT mode)

NOTE:
To run the Arduino sketch, unplug the shunt and re-insert the USB dongle into a USB port again.