

STM32 USB

USB Mother board 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.

Document History

Version	on Author Date		Description
Α	5G HUB	14.01.2024	Initial Document
В	B 5G HUB 21.04.2024		Minor update in format

Table of Contents

Purpos	Purpose of the Document2						
Docum	Document History						
1 1.1	Package contents: STM32 USB Dongle Package:	4 4					
1.2	Download	4					
2 2.1	General Description Overview	5 5					
2.2	Key Features	5					
2.3	Overview Diagrams	6					
2.4	Physical Characteristics	6					
2.5	Peripherals – Key Components	8					
2.6	Hardware Specification	9					
2.7	PIN Description1	0					
3 3.1	Using the Board with Arduino IDE1 Installing the Software1	1 1					
3.2	Setting Up the Board1	1					
3.3	Setting Up STM32 Bootloader1	1					
3.4	Setting Up Arduino IDE1	3					

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: <u>https://github.com/5ghub/stm32</u>

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 <u>https://www.arduino.cc/en/Main/Software</u>

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: <u>https://github.com/5ghub/5G-NB-IoT/tree/master/Tools</u>

2 General Description

2.1 Overview

This is a USB dongle powered by STM32 MCU. The USB is designed as compact and complete development platform for STMicroelectronics ARM cortex-M0 core-based STM32F072C8T6 microcontroller with 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 USB is used as a standalone board or used as a motherboard where other daughter boards can be stacked on it. Daughter boards can be GNSS, Cellular, CAN transceiver, and other modules.

The USB leverages and utilizes all hardware features of STM32F072C8T6. 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 for the 4G LTE and 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
- 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 for 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 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. USB Modem Overview Diagram – Top View



Figure 2. USB Modem 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. USB Modem 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				
NB-IoT Module	Quectel BG95-M2 or BG95-M2				
Dimension	25 mm (width) by 60 mm (length)				
Weight	20 grams				
Power Supply	USB (5V)				
LED	Power LED, LED1, LED2				
Interfacing Logic Voltage Level	2.21/				
(Operating Voltage)	3.5%				
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	32 (PAN-PA15 PRN-PR15)				
Pins					
USB	1				
I ² C	1				
SPI	1				
LIAPT	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)					
	О	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					
	0	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	1	1 switch to connect to MCU directly or BG95 directly					
		3.3V generated by the on-board regulator. Maximum current					
3.3V	0	drawn is 3A. The regulator also provides power to the MCU and					
		BG95					
5V	0	5V generated from the board. The board is supplied with power					
	_	from USB connector (typical 5V)					
GND		Ground					
PA5_AIN5	IO						
PA6_AIN5	IO	Six analog inputs which can provide up to 12 bits of resolution					
PA7_AIN5	IO	(i.e. 4096 different values). By default, each input measures					
PB0_AIN9	IO	from ground to 3.3 volts.					
PB1_AIN9	IO						
SCL	IO						
SDA IO		I ² C. The SCL (clock line). Can be used as GPIO					
SCK	IO	I ² C. 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 Tx/Rx	IO	CAN Interface. Can be used as GPIO					
PA8/ PA9/ PA10	IO	GPIO					
Cortex Debug	10	GPIO. Can be used as GPIO					
Connector	10						
BOOTO		Using Single Wire Debug to burn bootloader and debug the					
20010	I	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 PIN4 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.

Prg ST	M32CubeProgrammer									- 🗆 X
SIMC 🖤 😥 😥 🔀 🕎										
≡	Memory & File e	diting							•	Connected
	Device memory O	pen file +						USB	•	Disconnect
	Address 0x0800000	0 🔻 Size	0x400 Data	a width 32-bit	▼ Find Data	0x	Read 🗸 🗸	Port	JSB configurat	ion
OB	Address	0	4	8	с	ASCII			OSBI	
	0x0800000	FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	<u> </u>	_	Serial number		
CPU	0x08000010	FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ		PID	0xdf11	
swv	0x08000020	FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	<u>ŸŸŸŸŸŸŸŸŸŸŸŸŸŸŸ</u>		VID	0x0483	
	0x08000030	FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	<u> </u>		Dead Uneverteet (_
	0x08000040	FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	<u>ŸŸŸŸŸŸŸŸŸŸŸŸŸŸŸ</u> ŸŸ		Read Unprotect (
	0x08000050	FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	<u> </u>		TZEN Regression	(MCU)	
	0x08000060	FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	<u> </u>				
	0x08000070	FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	<u> </u>				
	0x08000080	FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	<u>ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ</u>				
	0x08000090	FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	<u>ŸŸŸŸŸŸŸŸŸŸŸŸŸ</u>				
	0x080000A0	FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	<u> </u>				
	0x080000B0	FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	<u>ŸŸŸŸŸŸŸŸŸŸŸŸŸŸŸ</u>				
	0x080000C0	FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	<u>ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ</u>				
	0x080000D0	FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	уууууууууууууууу	~			
	Log				L	ive Update Verbosity level 🔘	1 2 3			
(Ø8:1806 : STM322CubeProgrammer API v2.13.0 [Windows-648its 08:1837 : UR connection mode is defined with the HWrst reset mode 08:1837 : US8 speed : Full Speed (12M8it/s) 08:1837 : Manuf. ID : STMicroelectronics 08:1837 : None StM32 & BOOTLOADER 08:1837 : DPU protocol 1.1 08:1837 : DPU protocol 0.1 08:1837 : DPU protocol 0.2 08:1837 : DPU protocol 0.2 08:1837 : DPU protocol 0.2 10 08:1837 : DPU protocol 0.2 11 08:1837 : DPU protocol 0.2 11 08:1837 : DPU protocol 0.2 12 12									
	08:18:37 : Size : 16 08:18:37 : UPLOADING	Bytes							Target info <u>rma</u>	tion
	08:18:37 : Size : 10	24 Bytes						Board Device		 STM32F07y
	08:18:37 : Read progress:	0.000000						Type		MCU
\bigcirc	08:18:37 : Data read succe 08:18:37 : Time elapsed d	essfully luring the read opera	ation is: 00:00:00.024				~	Revision ID		
?							100% 😣	CPU Bootloader Versi	on	Cortex-M0

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_in_dex.json_



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

File Edit Sketch Tools Help		1
Addo Format Col+1	~	·Q··
Archive Sketch		
SGhub_b Manage Libraries Ctrl+Shift+1		
22 Serial Monitor Ctrl+Shift+M		_
23 24 Serial Plotter		
25 WEETOT / WEENNA Einemen Hadates		
26 Weinful / Winniko ramwale Opdater		
27 Option 32 Not Centricates		
29 Board: *STM32F072C8T6* Boards Manager Ctrl+Shift+B		
30 Port SG-NB-IoT SAMD Board (32-bits ARM Cortex-M0+)		
31 Get Board Info Adafruit nRF52		
33 Debug symbols and core logs: "None" Arduino AVR Boards		
34 Optimize: "Smallest (-Os default)" Arduino nRF52 Boards		
35 Board part number: "STM32F072C876" ► ATtiny Microcontrollers ►		
36 27 C Runtime Library, "Newlib Nano (default)" ► ATTinyCore ►		
38 Upload method: "STM32CubeProgrammer (DFU)" IMXRT1062 (NXP 32-bits ARM Cortex-M7) Boards		
39 USB support (if available): "CDC (generic 'Serial' supersede U(S)ART)" ► STM32F072C8T6 Board ► ✓ STM32F072C8T6		
40 Teensy		
41 Burn Bootloader 42		
43		
44 // Configure BME680 oversampling and filter		
45 bme.SetlemperatureOversampling(BHE080_0S_BX); 46 bme.Setlempiditworksampling(BHE080_0S_BX);		
47 bme.SetPressureOversampling(BHE680 05,4X);		
48 bme.SetIIRFilterSize(BME680_FILTER_SIZE_3);		
49 bme.SetGasHeater(320, 150); // 320*C for 150 ms		
50 } 51		
52 void loop()		
Output		≣ 6

- 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:

Board: "STM32F072C8T6"	×
Port: "COM5"	Þ
Get Board Info	
Debug symbols and core logs: "None"	×
Optimize: "Smallest (-Os default)"	•
Board part number: "STM32F072C8T6"	Þ
C Runtime Library: "Newlib Nano (default)"	Þ
Upload method: "STM32CubeProgrammer (DFU)"	Þ
USB support (if available): "CDC (generic 'Serial' supersede U(S)ART)"	Þ
Burn Bootloader	

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 PIN4 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.