How to program a microSWIFT V2

Download STM32CubeProgrammer

- Go to https://www.st.com/en/development-tools/stm32cubeprog.html (or just Google STM32CubeProgrammer) and download the correct version for your Operating System.
  - You will need an ST account. Registration is quick and easy.

STM32CubeProg - STM32CubeP

st.com/en/development-tools/stm32cubeprog.html

Careers | Sample & Buy | Support & Community | 日本語 | 中文 | English

life.augmented | Products | Tools & Software | Applications | Solutions | STM32 Developer Zone | About Us

Development Tools > Software Development Tools > STM32 Software Development Tools > STM32 Programmers > STM32CubeProg >

STM32CubeProg ACTIVE

Save to MyST

STM32CubeProgrammer software for all STM32

Get Software | Download databrief

Overview | Documentation | Tools & Software

Click here to bring up downloads list

Click here to create an account

Product overview

Description | All features | Circuit Diagram | Get Software | Featured Products | Featured Videos | Recommended for you

Description

STM32CubeProgrammer (STM32CubeProg) is an all-in-one multi-OS software tool for programming STM32 products.

It provides an easy-to-use and efficient environment for reading, writing and verifying device memory through both the debug interface (JTAG and SWD) and the bootloader interface (UART, USB DFU, I^2C, SPI, and CAN).

STM32CubeProgrammer offers a wide range of features to program STM32 internal memories (such as Flash, RAM, and OTP) as well as external memories

Feedback

Download the appropriate binaries

- All binaries are in the main branch of the microSWIFT-V2-Binaries repo within SASLabGroup organization: https://github.com/SASlabgroup/microSWIFT-V2-Binaries
  - This repo is private – if you are unable to access this repo, speak to someone in SASLabGroup to have your github account added to the organization
- Binaries are file that end in the extension .elf, with the files named based on the deployment they are supporting

microSWIFT-V2-Binaries

Private

Watch 1 Fork 0 Star 1

main 1 branch 0 tags Go to file Add file Code

prbush SIZRS_2 binaries for September 2023 deployment. d237ab1 2 hours ago 2 commits

README.md	Initial commit	2 days ago
sizrs_2.elf	SIZRS_2 binaries for September 2023 deployment.	2 hours ago

Example .elf file

README.md

microSWIFT-V2-Binaries

Compiled binaries for microSWIFT V2 buoys.

About

Compiled binaries for microSWIFT V2 buoys.

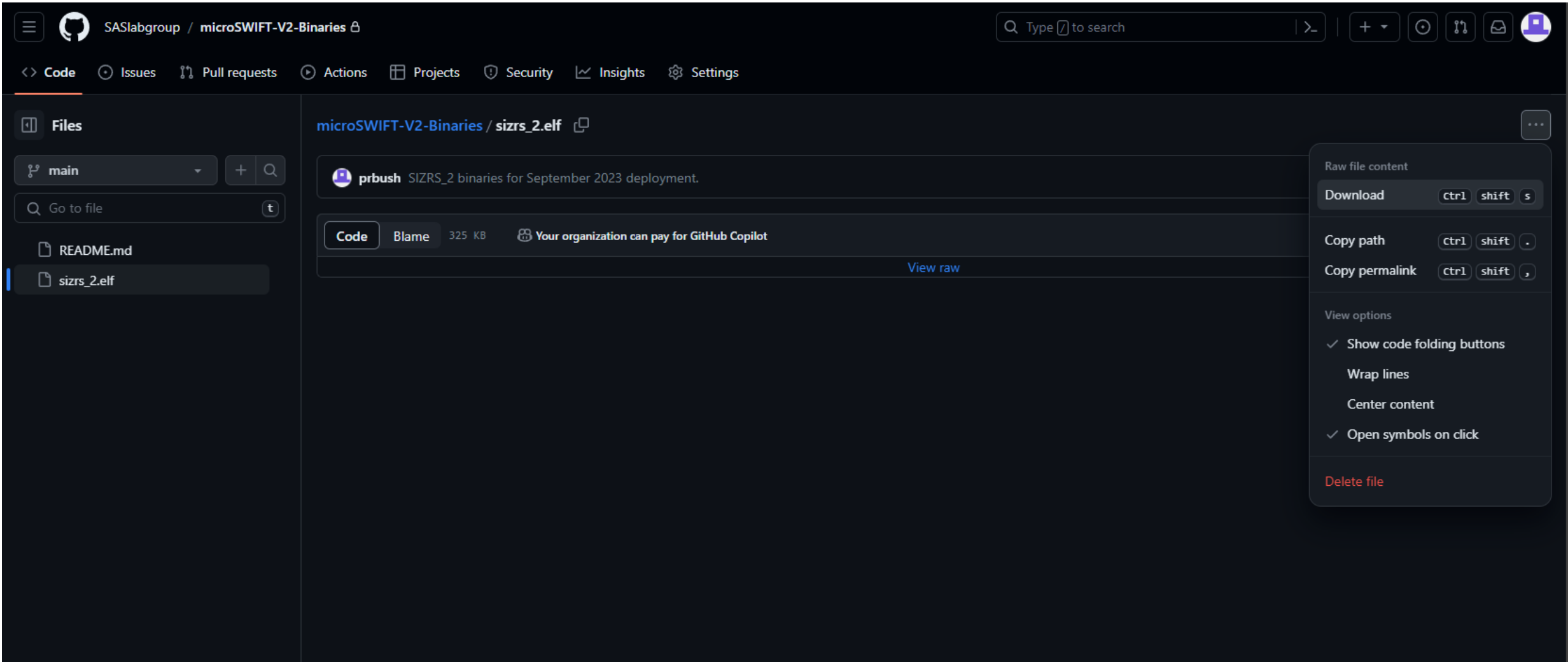
Readme
Activity
1 star
1 watching
0 forks

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package



Clicking on the file brings you to a page with an option to download the file

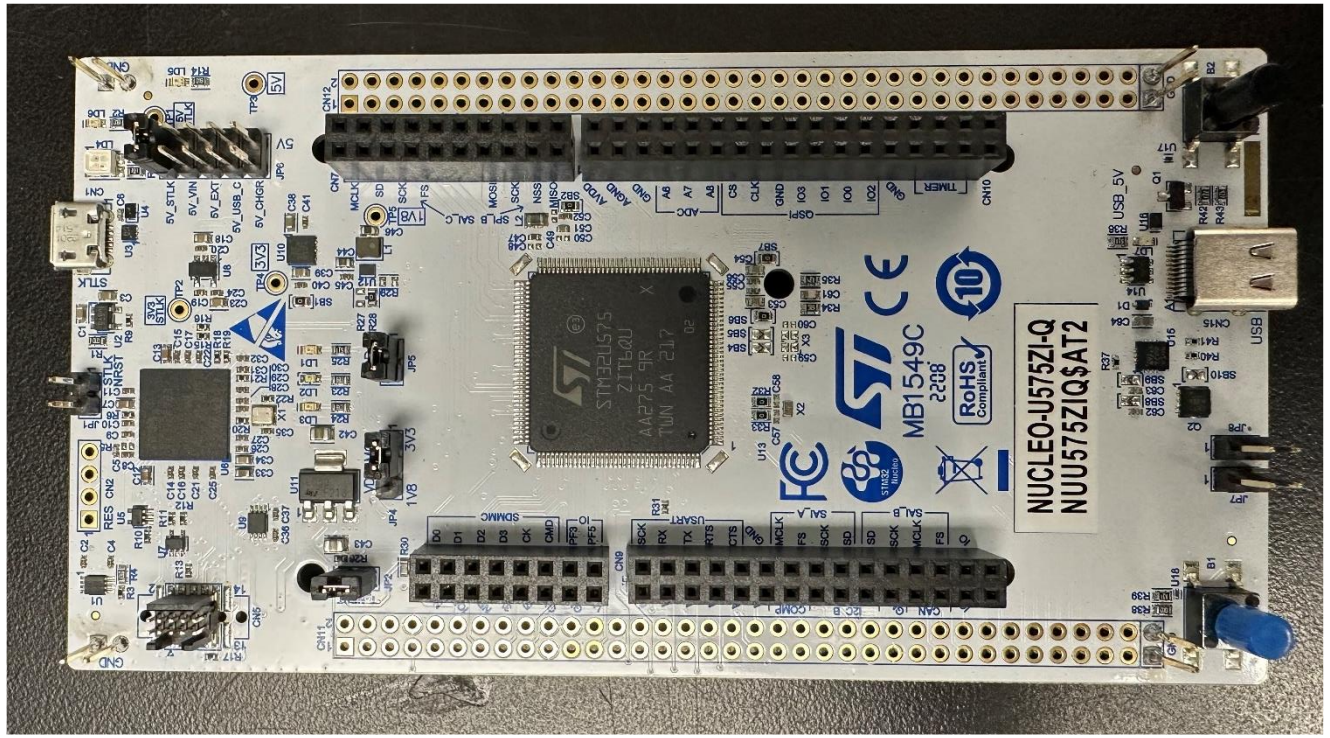
Setting up STM32CubeProgrammer

- There are two options for programming the STM32 Nucleo boards:
  1. Program the board directly using the onboard ST-Link. For this to work, solder bridge SB-1 must be connected. This is a good option with a fresh, straight out of the box board.
  2. Program with external debugger. This is a great option when the microSWIFT V2 has already been assembled.

Setting up with the onboard ST-Link

- When programming with the onboard ST-Link, you simply need to connect a micro USB to the board and program
  - Prior to programming, verify the jumpers on the Nucleo board are in the correct configuration. The configuration straight out of the box is correct for this, as noted in the photo.

Micro USB port for programming



Setting up with an external debugger

- When programming an already assembled microSWIFT V2, use an external debugger and a power source, either battery pack or DC power supply
  - Prior to programming, verify the jumpers on the Nucleo board are in the correct configuration

Pin assignments

(ST-Link → Nucleo):

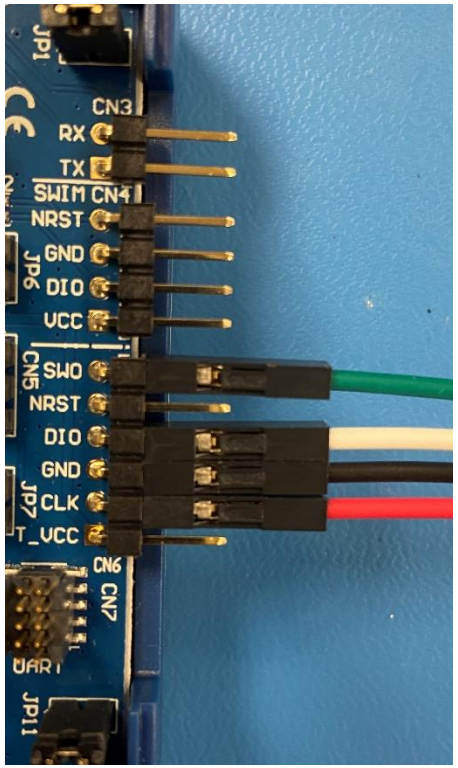
SWO → PB3

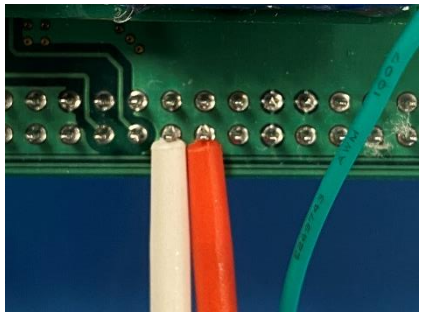
DIO → PA13

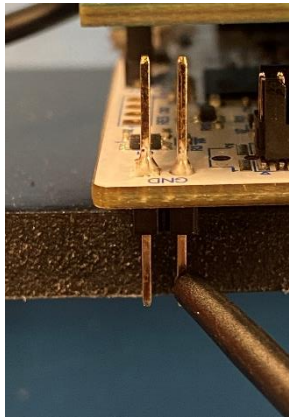
GND → Nucleo Ground

CLK → PA14

Pins are labelled on the bottom of the Nucleo Board







STM32CubeProgramming

- Open STM32CubeProgrammer
- Power on the microSWIFT
- Connect the ST-Link
- Click on the “Erasing and Programming” tab
- Set the filepath for the downloaded binaries
- Check the following boxes:
  - Verify Programming
  - Full Flash Memory checksum
  - Run after programming
- Click on “Start Programming”

STM32CubeProgrammer

Data Information Notice

Erasing & Programming

Connected

Download

File path: C:\Users\Phil\Desktop\nhci_2.elf — Browse

Start address

- [ ] Skip flash erase before programming
- [x] Verify programming
- [x] Full Flash memory checksum
- [x] Run after programming

Start Programming

Automatic Mode

- [ ] Full chip erase
- [x] Download file
- [ ] Option bytes commands: -ob

External Flash programming is not supported with Automatic Mode

Start automatic mode

Erase flash memory | Erase external memory

Erase selected sectors | Full chip erase

Select	Index	Start Address	Size
	0	0x08000000	8K
	1	0x08002000	8K
	2	0x08004000	8K
	3	0x08006000	8K
	4	0x08008000	8K
	5	0x0800A000	8K
	6	0x0800C000	8K
	7	0x0800E000	8K
	8	0x08010000	8K
	9	0x08012000	8K
	10	0x08014000	8K
	11	0x08016000	8K

ST-LINK | Disconnect

ST-LINK configuration

Serial number: 0021003C33...
Port: SWD
Frequency (kHz): 8000
Mode: Normal
Access port: 0
Reset mode: Software reset
Speed: Reliable
Shared: Disabled
Debug in Low Power mode
External loader
Target voltage: 0.00 V
Firmware version: V3J11M3B5S — Firmware upgrade

Target information

Board	STLINK-V3SET
Device	STM32U575/STM32U585
Type	MCU
Device ID	0x482
Revision ID	Rev X
Flash size	2 MB
CPU	Cortex-M33
Bootloader Version	0x92

Log

Live Update | Verbosity level: 1 2 3

```
17:07:24 : Data read successfully
17:07:24 : Time elapsed during the read operation is: 00:00:00.005
17:07:25 : Memory Programming ...
17:07:25 : Opening and parsing file: nhci_2.elf
17:07:25 :   File         : nhci_2.elf
17:07:25 :   Size         : 152.83 KB
17:07:25 :   Address      : 0x08000000
17:07:25 : Erasing memory corresponding to segment 0:
17:07:25 : Erasing internal memory sectors [0 19]
17:07:26 : Download in Progress:
17:07:27 : File download complete
17:07:27 : Time elapsed during download operation: 00:00:01.777
17:07:27 : Verifying ...
17:07:27 : Read progress:
17:07:28 : Download verified successfully
17:07:31 : Memory [0x08000000 : 0x08200000] - Checksum : 0x1E68AABA
17:07:31 : RUNNING Program ...
17:07:31 :   Address:    : 0x08000000
17:07:31 : Application is running, Please Hold on...
17:07:31 : Start operation achieved successfully
```

100%

If it doesn’t work

- Verify that the connections are solid. It is easy for adjacent connections (PA13 and PA14 on Nucleo board) to touch
- Ensure you have powered on and then connected to the MCU
- Check that the ST-Link has a solid green light
- Check that jumpers on the Nucleo are correct