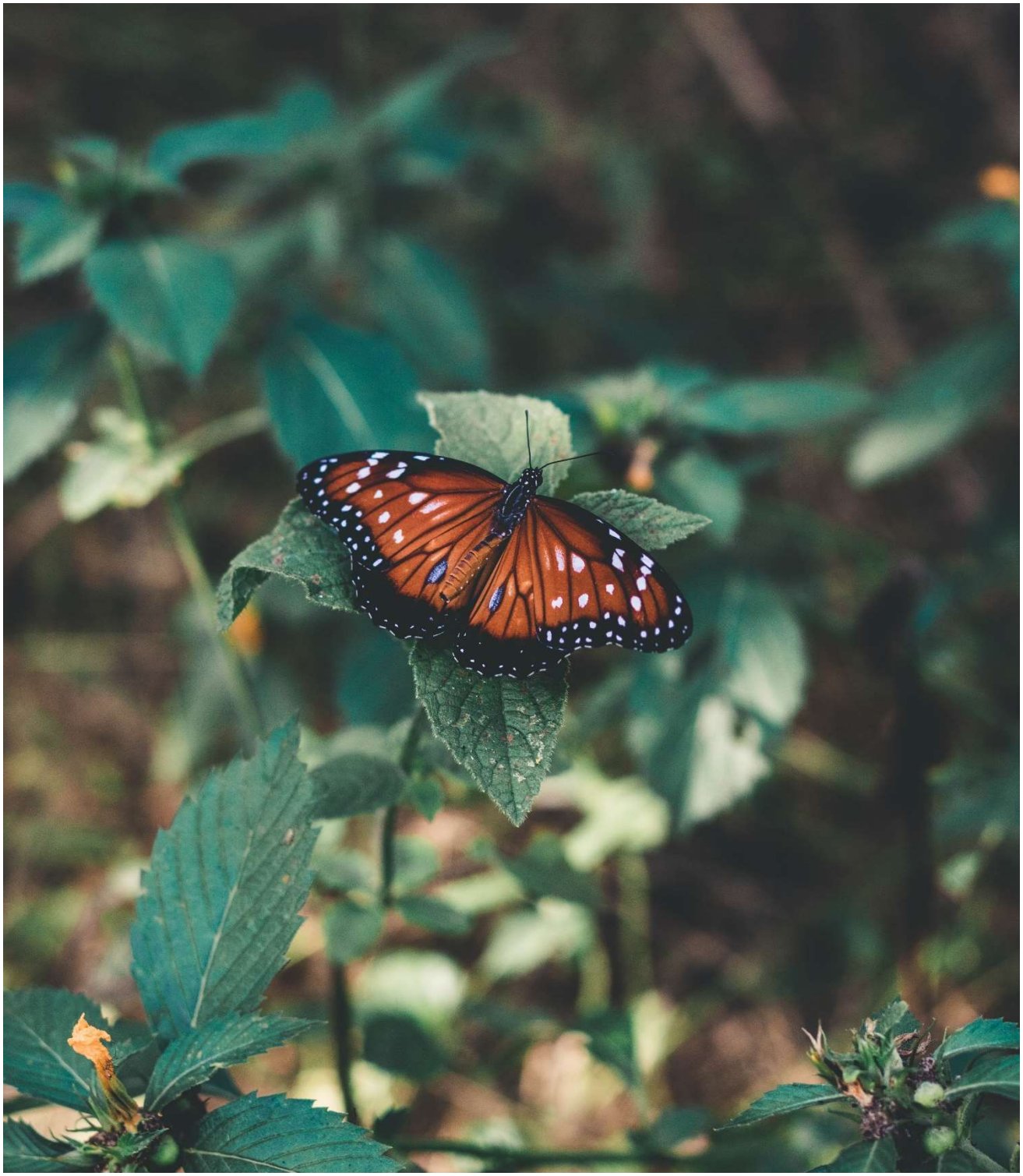
SMH Technologies®

17/03/2025

Interfacing FlashRunner 2.0 with STMICROELECTRONICS STM32

Driver v. 5.40

Moreno Ortolan



SMH Technologies S.r.l.

UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

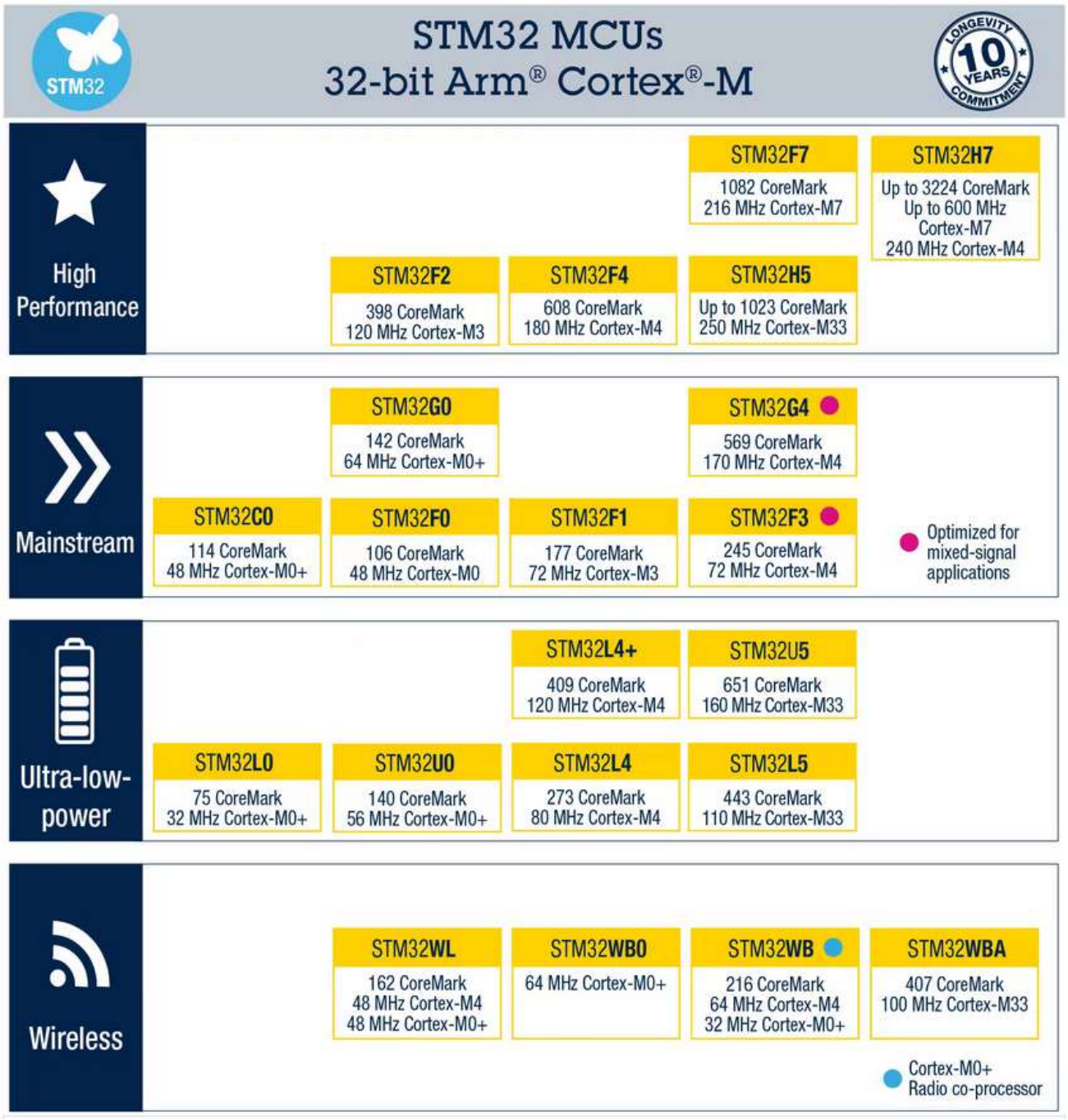
info@smh-tech.com

SMH Technologies®

STM32 Introduction

The STM32 family of 32-bit microcontrollers based on the Arm Cortex®-M processor is designed to offer new degrees of freedom to MCU users. It offers products combining very high performance, real-time capabilities, digital signal processing, low-power / low-voltage operation, and connectivity, while maintaining full integration and ease of development.

The unparalleled range of STM32 microcontrollers, based on an industry-standard core, comes with a vast choice of tools and software to support project development, making this family of products ideal for both small projects and end-to-end platforms.



SMH Technologies S.r.l.

UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

STM32 Protocols and PIN maps

All the STM32 devices support the SWD protocol.

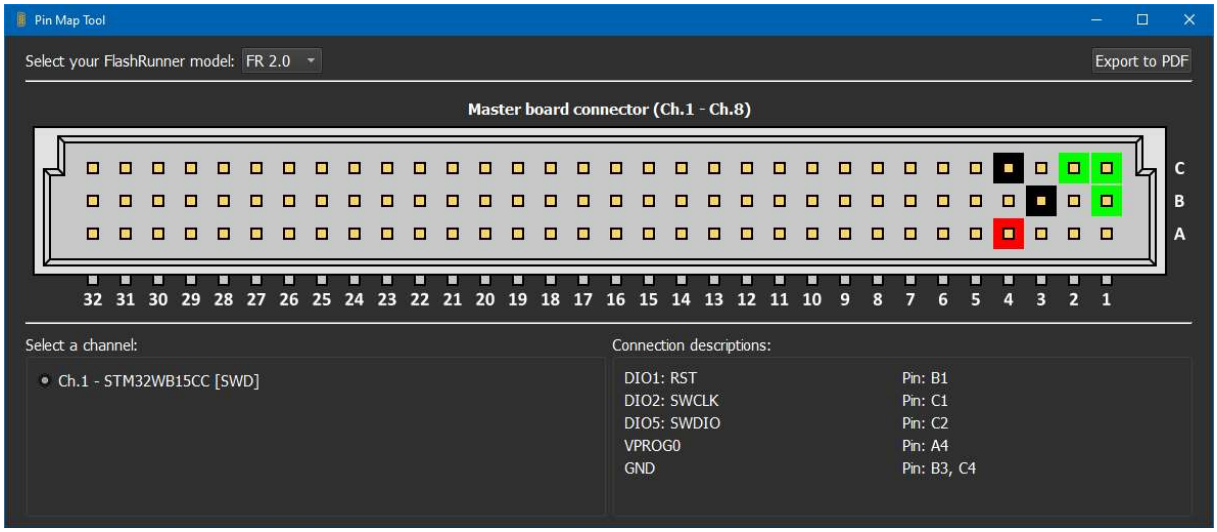
The JTAG protocol is supported for STM32U5x, STM32WBx, STM32WLx, STM32L1x, STM32L4x, STM32L4plus, STM32L5x, STM32G4x, STM32F1x, STM32F2x, STM32F3x, STM32F4x, STM32F7x, STM32H5x and STM32H7x.

You can choose the communication protocol during the Wizard procedure and you can modify it either through the same procedure, or manually in the project via the command:

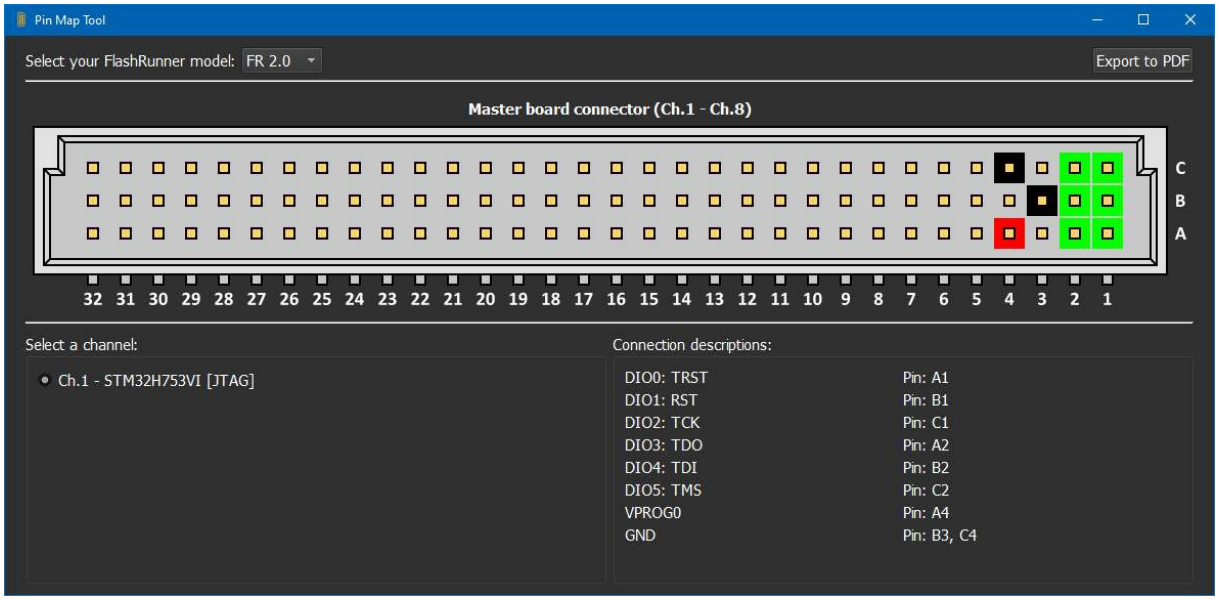
SMH recommend using the SWD protocol because it allows you to use fewer wires for communication.

`#TCSETPAR CMODE <SWD/JTAG>`

SWD PIN MAP



JTAG PIN MAP



HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

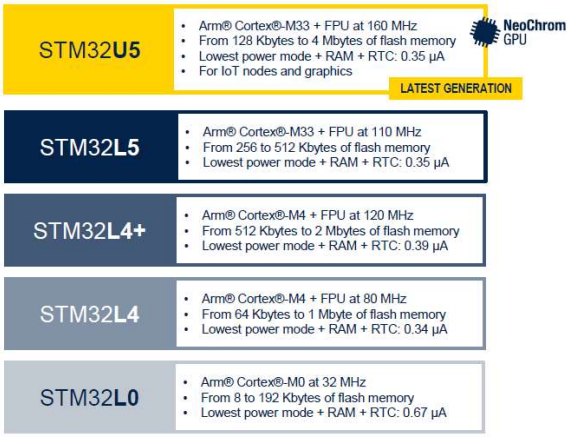
SMH Technologies®

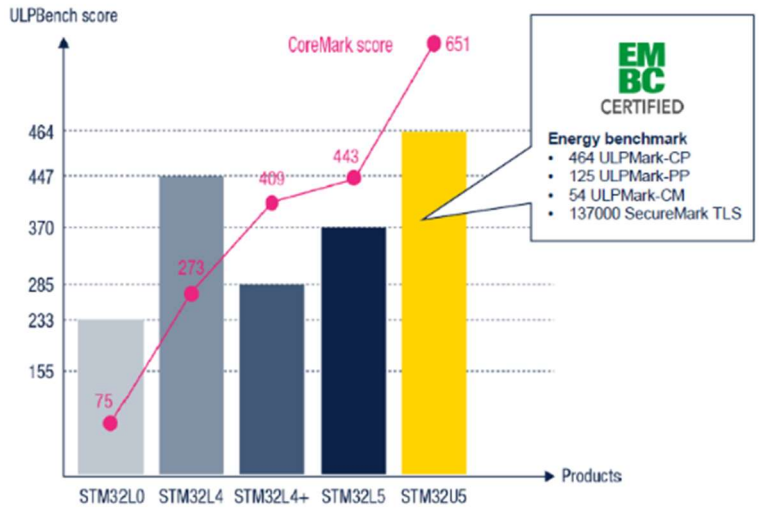
STM32 Families [1437 devices]

The STM32 family of 32-bit microcontrollers based on the Arm® Cortex®-M processor is divided into four groups and each of these groups are divided into various STM32 series:

STM32 Ultra Low Power MCUs [448 devices]

- **STM32L0 Series [99 devices]**
  - STM32L0x0 Value Line
  - STM32L0x1
  - STM32L0x2
  - STM32L0x3
- **STM32L1 Series [87 devices]**
  - STM32L100 Value Line
  - STM32L151/152
  - STM32L162
- **STM32L4 Series [94 devices]**
  - STM32L4x1
  - STM32L4x2
  - STM32L4x3
  - STM32L4x5
  - STM32L4x6
- **STM32L4+ Series [45 devices]**
  - STM32L4P5/Q5
  - STM32L4R5/S5
  - STM32L4R7/S7
  - STM32L4R9/S9
- **STM32L5 Series [17 devices]**
  - STM32L5x2
- **STM32U5 Series [74 devices]**
  - STM32U535/545
  - STM32U575/585
  - STM32U595/5A5
  - STM32U599/5A9
  - STM32U5F7/5G7
  - STM32U5F9/5G9
- **STM32U0 Series [32 devices]**
  - STM32U031x
  - STM32U073x
  - STM32U083x
- **STM32U3 Series [xx devices]**
  - STM32U38x
  - STM32U37x





SMH Technologies S.r.l.

UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

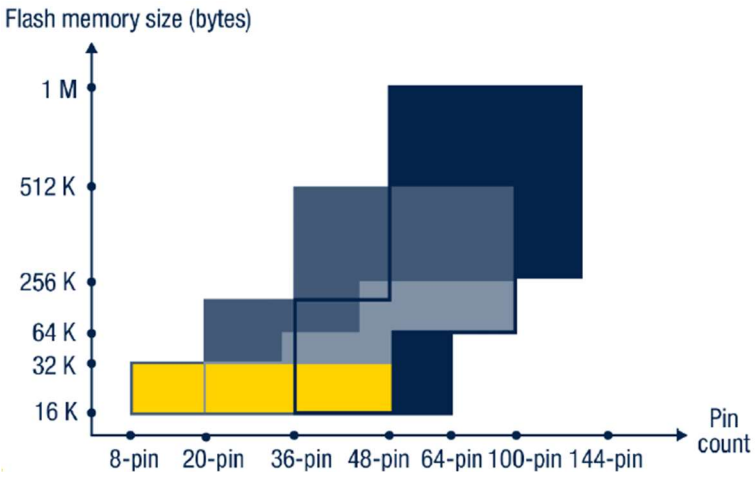
info@smh-tech.com

SMH Technologies®

STM32 Mainstream MCUs [472 devices]

- **STM32C0 Series [53 devices]**
  - STM32C0x1
- **STM32F0 Series [73 devices]**
  - STM32F0x0 Value Line
  - STM32F0x1
  - STM32F0x2
  - STM32F0x8
- **STM32F1 Series [95 devices]**
  - STM32F100 Value Line
  - STM32F101
  - STM32F102
  - STM32F103
  - STM32F105/107
- **STM32F3 Series [69 devices]**
  - STM32F301
  - STM32F302
  - STM32F303
  - STM32F334
  - STM32F373
  - STM32F3x8
- **STM32G0 Series [99 devices]**
  - STM32G0x0 Value Line
  - STM32G0x1
- **STM32G4 Series [83 devices]**
  - STM32G4x1
  - STM32G4x3
  - STM32G4x4

STM32F1	• Arm® Cortex®-M3 72 MHz – 61 DMIPS • Performance and peripherals, user-friendly tools
STM32G0	• Arm® Cortex®-M0+ at 64 MHz – 59 DMIPS • Maximum IO count per package • Advanced function: analog, low-power, control
STM32F0	• Entry-level MCU for cost-sensitive operations • Arm® Cortex®-M0 at 48 MHz – 38 DMIPS
STM32C0	• Arm® Cortex®-M0+ FPU at 48 MHz – 44 DMIPS • Most affordable 32-bit MCU LATEST GENERATION



SMH Technologies S.r.l.

UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

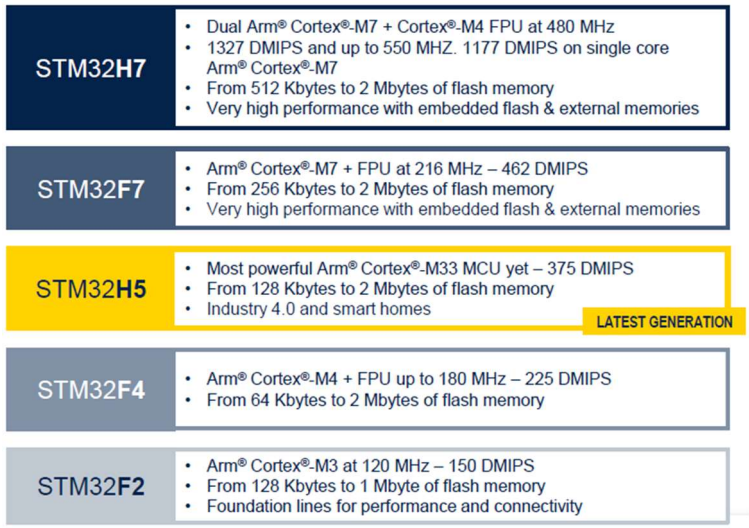
→smh-tech.com

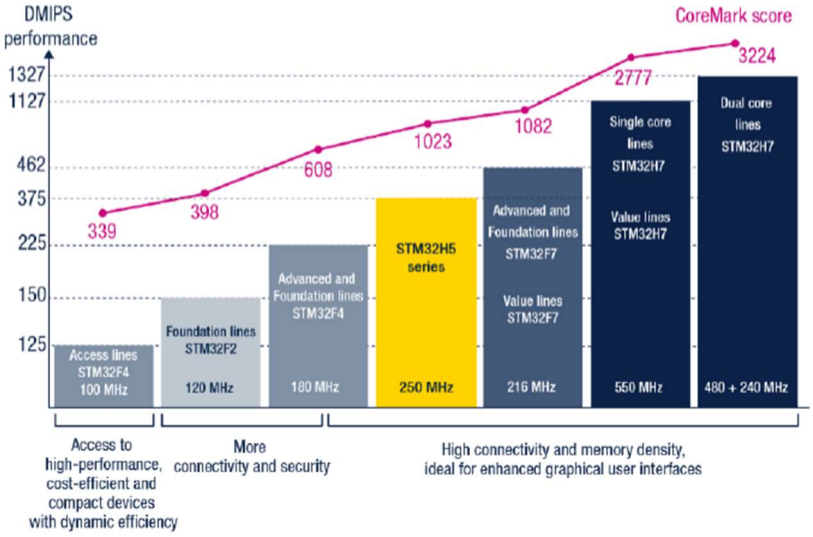
info@smh-tech.com

SMH Technologies®

STM32 High Performance MCUs [451 devices]

- **STM32F2 Series [38 devices]**
  - STM32F2x5
  - STM32F2x7
- **STM32F4 Series [149 devices]**
  - STM32F401
  - STM32F405/415
  - STM32F407/417
  - STM32F410
  - STM32F411
  - STM32F412
  - STM32F413/423
  - STM32F427/437
  - STM32F429/439
  - STM32F446
  - STM32F469/479
- **STM32F7 Series [86 devices]**
  - STM32F7x0 Value Line
  - STM32F7x2
  - STM32F7x3
  - STM32F7x5
  - STM32F7x6
  - STM32F7x7
  - STM32F7x9
- **STM32H5 Series [45 devices]**
  - STM32H503
  - STM32H522/533
  - STM32H562
  - STM32H563/573
- **STM32H7 Series [133 devices]**
  - STM32H723/733
  - STM32H725/735
  - STM32H730 Value Line
  - STM32H742
  - STM32H743/753
  - STM32H745/755
  - STM32H747/757
  - STM32H750 Value Line
  - STM32H7A3/7B3
  - STM32H7B0 Value Line
  - STM32H7R3/S3
  - STM32H7R7/S7





SMH Technologies S.r.l.

UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

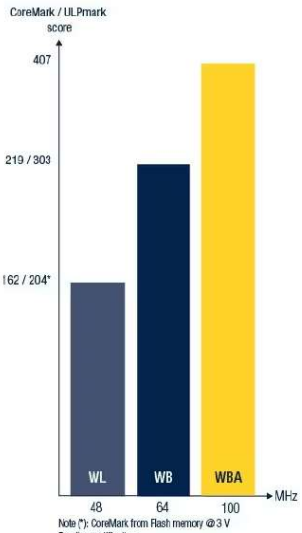
SMH Technologies®

STM32 Wireless MCUs [66 device]

- **STM32WB Series [18 devices]**
  - STM32WBx0 Value Line
  - STM32WBx5
  - STM32WBxM Modules
- **STM32WB0 Series [10 device]**
  - STM32WBx0 Value Line
- **STM32WBA Series [15 devices]**
  - STM32WBA5x
- **STM32WL Series [23 devices]**
  - STM32WL3x
  - STM32WL5x
  - STM32WLEx
  - STM32WLxM Module

The ideal fit for RF designers looking for more than just a radio device

STM32**WBA**	• Arm® Cortex®-M33 w/ **TrustZone®** @ 100 MHz • **1 Mbyte of** flash memory / **128 Kbytes** RAM • Bluetooth® Low Energy 5.3 (long-range, 2 Mbps, adverting extension) • Up to **+10 dBm** output power • Enhanced security Bluetooth 5 — LATEST GENERATION
STM32**WB**	• **Dual core** & security (Arm® Cortex®-M4 /-M0+) • Up to **1 Mbyte flash- memory/ 256 Kbytes RAM** • Bluetooth® Low Energy 5.4, Zigbee R22 & Thread, proprietary, Matter Q4'23 OPENTHREAD, matter, Bluetooth 5, Zigbee 3.0
STM32**WB0**	• Arm® Cortex®--M0+ at 64 MHz • Up to **512 Kbytes of flash memory / 64 Kbytes RAM** • Transceiver frequency: 2.4 GHz • Power outputs: up to 8 dBm • Bluetooth® Low Energy 5.3 Bluetooth 5
STM32**WL**	• World 1st MCU enabling **LoRa®, (G)FSK, (G)MSK, BPSK** • Arm® Cortex®-M4 and -M0+ at 48 MHz supporting RF – 60 DMIPS • Up to **256 Kbytes of flash memory / 64 Kbytes RAM** • Transceiver frequency: 150 to 960 MHz • Dual-power outputs: up to 22 dBm and up to 15 dBm (Embedded PAs) LoRaWAN, M-Bus, sigfox, mioty



SMH Technologies S.r.l.

UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

STM32 Available Commands

STM32 Low Power MCUs

STM32L0 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
EEprom [E]			✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Factory Options [I]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

STM32L0 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

STM32L1 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
EEprom [E]			✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Factory Options [I]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

STM32L1 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

SMH Technologies®

STM32L4 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
OTP Area [T]			✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

Note: Fast Programming and Standard Programming Modes available and selectable via **#TCSETPAR** PROGRAM MODE [F/S] <Fast/Standard>.
Driver version >= **5.27**

STM32L4 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

STM32L4+ Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
OTP Area [T]			✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

Note: Fast Programming and Standard Programming Modes available and selectable via **#TCSETPAR** PROGRAM MODE [F/S] <Fast/Standard>.
Driver version >= **5.27**

STM32L4+ Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

SMH Technologies®

STM32L5 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
OTP Area [T]			✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

STM32L5 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

STM32U0 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
OTP Area [T]			✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

Note: Fast Programming and Standard Programming Modes available and selectable via **#TCSETPAR** PROGRAM MODE [F/S] <Fast/Standard>.
Driver version >= **5.30**

STM32U0 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for RDP regression with Password:

#TPCMD GET_DEVICE_AUTHENTICATION_ID
#TPCMD RDP_PASSWORD_STORE
#TPCMD RDP_PASSWORD_STORE_LOCK
#TPCMD RDP_PASSWORD_SET
#TPCMD RDP_PASSWORD_SET_UNLOCK
#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE
#TPCMD RDP_PASSWORD_REMOVE

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

STM32U3 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
OTP Area [T]			✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

Note: Fast Programming and Standard Programming Modes available and selectable via **#TCSETPAR** PROGRAM MODE [F/S] <Fast/Standard>.
Driver version >= **5.36**

STM32U0 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for RDP regression with Password:

#TPCMD GET_DEVICE_AUTHENTICATION_ID
#TPCMD RDP_PASSWORD_STORE
#TPCMD RDP_PASSWORD_STORE_LOCK
#TPCMD RDP_PASSWORD_SET
#TPCMD RDP_PASSWORD_SET_UNLOCK
#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE
#TPCMD RDP_PASSWORD_REMOVE

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

SMH Technologies®

STM32U5 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
OTP Area [T]			✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

STM32U5 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for RDP regression with Password:

#TPCMD GET_DEVICE_AUTHENTICATION_ID
#TPCMD RDP_PASSWORD_STORE
#TPCMD RDP_PASSWORD_STORE_LOCK
#TPCMD RDP_PASSWORD_SET
#TPCMD RDP_PASSWORD_SET_UNLOCK
#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE
#TPCMD RDP_PASSWORD_REMOVE

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

STM32 Mainstream MCUs

STM32C0 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
OTP Area [T]			✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Engineering Bytes [I]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

Note: Fast Programming and Standard Programming Modes available and selectable via **#TCSETPAR** PROGRAM MODE [F/S] <Fast/Standard>.
Driver version >= **5.11**

STM32C0 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

STM32F0 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

STM32F0 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

STM32F1 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

Note: If RDP is set to Level 1, you cannot Read the Option Bytes Area. RDP value to set level 0 is **0xA5**

STM32F1 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

STM32F3 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

STM32F3 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

STM32G0 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
OTP Area [T]			✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Engineering Bytes [I]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

Note: Fast Programming and Standard Programming Modes available and selectable via **#TCSETPAR** PROGRAM MODE [F/S] <Fast/Standard>.
Driver version >= **4.12**

STM32G0 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

STM32G4 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
OTP Area [T]			✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

Note: Fast Programming and Standard Programming Modes available and selectable via **#TCSETPAR** PROGRAM MODE [F/S] <Fast/Standard>.
Driver version >= **4.12**

STM32G4 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

STM32 High Performance MCUs

STM32F2 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
OTP Area [T]			✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

Note: If the RDP is set to Level 1, you cannot Read the Option Bytes Area

STM32F2 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

STM32F4 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
OTP Area [T]			✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

Note: If the RDP is set to Level 1, you cannot Read the Option Bytes Area

STM32F4 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

STM32F7 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
OTP Area [T]			✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

Note: If the RDP is set to Level 1, you cannot Read the Option Bytes Area

STM32F7 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

SMH Technologies®

STM32H5 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Non-Secure Internal Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
OTP Area [T]			✓	✓	✓	✓	✓	✓
Information Area [I]							✓	✓
System Memory [S]							✓	✓
Secure Internal Flash [G]	✓	✓	✓	✓	✓	✓	✓	✓
OBK Option Bytes Key [K]							✓	✓
Option Bytes [O]				✓	✓		✓	✓
Provisioning [P] (virtual memory)	Virtual Memory – specific commands are available for this memory area							
Regression [R] (virtual memory)	Virtual Memory – specific commands are available for this memory area							
Secure Manager SFI [M] (virtual memory)	Virtual Memory – specific commands are available for this memory area							

STM32H5 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for debug authentication and SFI:

#TPCMD DISCOVERY
#TPCMD PROVISIONING
#TPCMD REGRESSION
#TPCMD SECURE_FIRMWARE_INSTALL

Commands for product state and trust zone management:

#TPCMD GET_PRODUCT_STATE
#TPCMD SET_PRODUCT_STATE
#TPCMD CHECK_PRODUCT_STATE
#TPCMD GET_TRUST_ZONE
#TPCMD SET_TRUST_ZONE
#TPCMD CHECK_TRUST_ZONE

Commands for option bytes management:

#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

SMH Technologies®

STM32H7 Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
OTP Area [T]			✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Information Area [I]							✓	✓
Option Bytes [O]				✓	✓		✓	✓
External Flash [X]	✓	✓	✓	✓	✓	✓	✓	✓
Provisioning [P] (virtual memory)	Virtual Memory – specific commands are available for this memory area (only for STM32H7Rx and STM32H7Sx)							
Regression [R] (virtual memory)	Virtual Memory – specific commands are available for this memory area (only for STM32H7Rx and STM32H7Sx)							

STM32H7 Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for restart the device:

#TPCMD RUN

Commands for debug authentication:

#TPCMD DISCOVERY
#TPCMD PROVISIONING
#TPCMD REGRESSION

Commands for product state and trust zone management:

#TPCMD GET_PRODUCT_STATE
#TPCMD SET_PRODUCT_STATE
#TPCMD CHECK_PRODUCT_STATE

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

SMH Technologies®

STM32 Wireless MCUs

STM32WB Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
Factory Options [I]							✓	✓
System Memory [S]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

STM32WB Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for FUS (Firmware Upgrade Service):

#TPCMD FUS_GET_VERSION
#TPCMD FUS_GET_WIRELESS_STACK_VERSION
#TPCMD FUS_CHECK_VERSION
#TPCMD FUS_CHECK_WIRELESS_STACK_VERSION
#TPCMD FUS_READ_INFORMATIONS
#TPCMD FUS_DELETE_FIRMWARE
#TPCMD FUS_UPDATE
#TPCMD FUS_START
#TPCMD FUS_START_WIRELESS_STACK
#TPCMD FUS_ANTI_ROLLBACK

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

SMH Technologies®

STM32WBA Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
OTP Area [T]			✓	✓	✓	✓	✓	✓
System Memory [S]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

STM32WBA Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections:

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information:

#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for RDP regression with Password:

#TPCMD GET_DEVICE_AUTHENTICATION_ID
#TPCMD RDP_PASSWORD_STORE
#TPCMD RDP_PASSWORD_STORE_LOCK
#TPCMD RDP_PASSWORD_SET
#TPCMD RDP_PASSWORD_SET_UNLOCK
#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE
#TPCMD RDP_PASSWORD_REMOVE

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

STM32WL Series

MEMORY	MASSERASE	ERASE PAGE	BLANKCHECK	PROGRAM	VERIFY READOUT	VERIFY CHECKSUM	READ	DUMP
Flash [F]	✓	✓	✓	✓	✓	✓	✓	✓
Factory Options [I]							✓	✓
System Memory [S]							✓	✓
Option Bytes [O]				✓	✓		✓	✓

STM32WL Additional Commands

Commands for Flash memory:

#TPCMD UNPROTECT
#TPCMD ERASE F

Commands for option bytes and RDP protections (for all STM32WL device except STM32WL3x):

#TPCMD GET_PROTECTION
#TPCMD SET_PROTECTION
#TPCMD CHECK_PROTECTION
#TPCMD RESTORE_OPTION_BYTES
#TPCMD OVERVIEW_OPTION_BYTES
#TPCMD WRITE_OPTION_BYTE
#TPCMD COMPARE_OPTION_BYTE

Commands for device Information (for all STM32WL device except STM32WL3x):

#TPCMD GET_UNIQUE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID

Commands for device Information:

#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_INFORMATIONS

Commands for read device memory:

#TPCMD READ_MEM8
#TPCMD READ_MEM16
#TPCMD READ_MEM32
#TPCMD GET_MEMORY_HASH

Commands for restart the device:

#TPCMD RUN

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

STM32 Driver Commands

STM32 Standard Commands

Here you can find the complete list of all available commands for STM32 driver.

```
Memory type:

F → NON-SECURE FLASH
G → SECURE FLASH
E → EEPROM
O → OPTION BYTES
K → OBK OPTION BYTES KEY
T → OTP AREA
S → SYSTEM MEMORY (Read Only)
I → FACTORY OPTIONS (Read Only)
X → EXTERNAL MEMORY
P → PROVISIONG (Virtual Memory)
R → REGRESSION (Virtual Memory)
M → SECURE MANAGER SFI (Virtual Memory)
```

#TPCMD CONNECT

#TPCMD CONNECT

This function performs the entry and is the first command to be executed when starting the communication with the device.

Example for STM32WB series:

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x24770011, Type: AMBA AHB3 bus.
AP[1] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[0] ROM table base address 0xE00FF000.
CPUID: 0x410FC241.
Implementer Code: 0x41 - [ARM].
Found Cortex M4 revision r0p1.
Cortex M4 Core halted [0.004 s].
CPU2 debug access is disabled.
System and Flash memory are secure from address 0x080CF000.
BOR Level 0. Reset level threshold is ~ 1.7 V.
Device configuration: [0x39FFF1AA]
 * The device's RDP level is 0 [0xAA].
 * System Security ESE enabled.
 * BOR Level 0. Reset level threshold is around 1.7 V.
 * No reset generated when entering the Stop mode.
 * No reset generated when entering the Standby mode.
 * No reset generated when entering the Shutdown mode.
 * Software independent watchdog selected.
 * Independent watchdog counter is running in Stop mode.
 * Independent watchdog counter is running in Standby mode.
 * Software window watchdog selected.
 * SRAM2 parity check disabled.
 * SRAM2 and PKA RAM erased when a system reset occurs.
 * BOOT0 signal is taken from the option bit nBOOT0.
 * Radio automatic gain control trimming [0x1].
PLL enabled using internal HSI oscillator.
Requested Clock is 37.50 MHz.
Generated Clock is 37.50 MHz.
Good samples: 4 [Range 4-7].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 37.50 MHz.
Time for Connect: 0.109 s.
>|
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Example for STM32H5 series:

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 37.50 MHz.
Trying Hot Plug connect procedure.
Good samples: 3 [Range 4-6].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 37.50 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x54770002, Type: AMBA APB2 or APB3 bus.
AP[1] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[1] ROM table base address 0xE00FE000.
CPUID: 0x410FD214.
Implementer Code: 0x41 - [ARM].
Found Cortex M33 revision r0p4.
Program counter value is 0xEFFFFFFE.
Cortex M33 Core halted [0.001 s].
> Device configuration 1: [0x30F0EDF8]
 * The device's product state is Open [0xED].
 * BOR level 1, the threshold level is low (around 2.1 V).
 * Brownout high disabled.
 * IWDG watchdog is controlled by software.
 * WWDG watchdog is controlled by software.
 * No reset generated when entering Stop mode on core domain.
 * No reset generated when entering Standby mode on core domain.
 * High-speed IO at low VDD voltage feature disabled (VDD can exceed 2.7 V).
 * High-speed IO at low VDDIO2 voltage feature disabled (VDDIO2 can exceed 2.7 V).
 * Independent watchdog keep running in system Stop mode.
 * Independent watchdog keep running in Standby mode.
 * Bank1 and Bank2 not swapped.
> Device configuration 2: [0xC3000034]
 * SRAM1 and SRAM3 not erased when a system reset occurs.
 * SRAM2 erased when a system reset occurs.
 * BKPRAM ECC check disabled.
 * SRAM3 ECC check disabled.
 * SRAM2 ECC check enabled.
 * TrustZone is disabled.
 * UBE OEM-iRoT (user flash) selected. In Open Product State this value selects bootloader.
PLL enabled using internal HSI oscillator.
Requested Clock is 37.50 MHz.
Generated Clock is 37.50 MHz.
Good samples: 4 [Range 3-6].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 37.50 MHz.
Time for Connect: 0.427 s.
>|
```

#TPCMD MASSERASE

#TPCMD MASSERASE <F|G|E|X>

F: Masserase command for Non-Secure Flash memory of target device.
G: Masserase command for Secure Flash memory of target device.
E: Masserase command for EEprom memory of target device.
X: Masserase command for External Memory connected to target device.

#TPCMD ERASE

#TPCMD ERASE <F|G|X>

This function performs a page/sector erase of all Non-Secure Flash memory, Secure Flash memory or External memory.

#TPCMD ERASE <F|G|X> <start address> <size>

This function performs a page/sector erase of Non-Secure Flash memory, Secure Flash memory or External memory.
Enter the Start Address and Size in hexadecimal format.

UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

#TPCMD BLANKCHECK

#TPCMD BLANKCHECK <F|G|E|T|X>
Blankcheck is available for Non-Secure Flash, Secure Flash, EEprom, OTP and External memory.
Verify if all memory is erased.

#TPCMD BLANKCHECK <F|G|E|T|X> <start address> <size>
Blankcheck is available for Non-Secure Flash, Secure Flash, EEprom, OTP and External memory.
Verify if selected part of memory is erased.
Enter the Start Address and Size in hexadecimal format.

#TPCMD PROGRAM

#TPCMD PROGRAM <F|G|E|T|O|X>
Program is available for Non-Secure Flash, Secure Flash, EEprom, OTP, Option Bytes and External memory.
Programs all memory of the selected type based on the data in the FRB file.

#TPCMD PROGRAM <F|G|E|T|O|X> <start address> <size>
Program is available for Non-Secure Flash, Secure Flash, EEprom, OTP, Option Bytes and External memory.
Programs selected part of memory of the selected type based on the data in the FRB file.
Enter the Start Address and Size in hexadecimal format.

#TPCMD VERIFY

#TPCMD VERIFY <F|G|E|T|O|X> <R>
R: Readout Mode.
Verify Readout is available for Non-Secure Flash, Secure Flash, EEprom, OTP, Option Bytes and External memory.
Verify all memory of the selected type based on the data in the FRB file.

#TPCMD VERIFY <F|G|E|T|O|X> <R> <start address> <size>
R: Readout Mode.
Verify Readout is available for Non-Secure Flash, Secure Flash, EEprom, OTP, Option Bytes and External memory.
Verify selected part of memory of the selected type based on the data in the FRB file.
Enter the Start Address and Size in hexadecimal format.

#TPCMD VERIFY <F|G|E|T|X> <S>
S: Checksum 32 Bit Mode.
Verify Checksum is available for Non-Secure Flash, Secure Flash, EEprom, OTP and External memory.
Verify all memory of the selected type based on the data in the FRB file.

#TPCMD VERIFY <F|G|E|T|X> <S> <start address> <size>
S: Checksum 32 Bit Mode.
Verify Checksum is available for Non-Secure Flash, Secure Flash, EEprom, OTP and External memory.
Verify selected part of memory based on the data in the FRB file.
Enter the Start Address and Size in hexadecimal format.

#TPCMD READ

#TPCMD READ <F|G|E|T|O|X|S|I>
Read is available for all memories.
Read all memory of selected type.
The result of the read command will be visible into the Terminal.

#TPCMD READ <F|G|E|T|O|X|S|I> <start address> <size>
Read is available for all memories.
Read selected part of memory of the selected type.
The result of the read command will be visible into the Terminal.

UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

#TPCMD DUMP

`#TPCMD DUMP <F|G|E|T|O|X|S|I>`
Dump is available for all memories.
Dump all memory of selected type.
The result of the dump command will be stored in the FlashRunner 2.0 internal memory.

`#TPCMD DUMP <F|G|E|T|O|X|S|I> <start address> <size>`
Dump is available for all memories.
Dump selected part of memory of the selected type.
The result of the dump command will be stored in the FlashRunner 2.0 internal memory.

#TPCMD DISCONNECT

`#TPCMD DISCONNECT`
Disconnect function. Power off and exit.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

STM32 Additional Commands

The additional commands are specific commands that perform particular functions such as, for example, they allow easier management of the Option Bytes, RDP protections, Permission Levels, etc.

They also allow you to perform complex procedures with a simple single command and allow you to obtain important information from the device.

Typically, all additional commands are available in the last section of the Graphical User Interface when creating a project.

Obviously, it is possible to see the specific commands assigned to the specific STM32 series in the chapter Available operations for family.

Additional Commands for RDP Management

These commands are used to modify, check or get the RDP (Readout Protection) level from/to target STM32 device.
They are available for all devices except the STM32H5 series.

For STM32U5 there is the possibility to set the RDP using a specific password for RDP1 and RDP2 regression.
Please go to this chapter STM32U5 - Readout Protection Level regression with password if you want to perform this procedure.

#TPCMD SET_PROTECTION

Syntax: `#TPCMD SET_PROTECTION <RDP value>`

`<RDP value>` RDP level in HEX format [0x00 - 0xFF] (i.e., 0xBB)

Prerequisites: none

Description: This command sets the RDP value into target device
Specific Option Byte is modified only in the appropriate RDP section

Note: **STM32H5**: this command is not available

The RDP level 0 is **0xAA** for all devices except for **STM32F1** where level 0 is **0xA5**
The RDP level 0.5 is **0x55** only when TrustZone is active
The RDP level 2 is **0xCC**
The RDP level 1 is for all values excepted the above

When RDP level 1 is active, programming the RDP to level 0 causes the Flash memory to be mass-erased
When RDP level 2 is active, JTAG/SWD port is forever disabled unless there is a password stored inside the device (this last option is available only for **STM32U5**)

Examples: Correct command execution: 😊

```
---#TPCMD SET_PROTECTION 0xBB
Time for Set Protection: 0.058 s
```

Now if you try to reconnect to target device you can see that RDP value is changed to 1.

```
* The device's RDP level is 1 [0xBB].
```

Wrong command execution for RDP value **0xCC**: ☹

To avoid setting the RDP level 2 by mistake, if the command `#TPCMD SET_PROTECTION` receives the value **0xCC** as input, it will respond with this error:

```
---#TPCMD SET_PROTECTION 0xCC
═══════════════════════════════════════════════════════════ *** WARNING *** ═══════════════════════════
Are you sure to set Readout Protection Level to 0xCC?.
Level 2: Enabled debug/chip read protection.
 - All protections provided by Level 1 are active.
 - Booting from system memory is not allowed anymore.
 - JTAG, SWD, SWV (single-wire viewer) are disabled.
 - User option bytes can no longer be changed.
```


UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

```
 - When booting from Flash memory, accesses to Flash memory and backup SRAM from user code are allowed.
Memory read protection Level 2 is an irreversible operation.
When Level 2 is activated, the level of protection cannot be decreased to Level 0 or Level 1.
The JTAG port is permanently disabled when Level 2 is active (acting as a JTAG fuse).
As a consequence, boundary scan cannot be performed.
If you are sure to set RDP to 0xCC, please use this command syntax: \" #TPCMD SET_PROTECTION 0xCC Y \".
```

Correct command execution for RDP value **0xCC**: 😊

If you are sure to set RDP to 0xCC, please use this command syntax: `#TPCMD SET_PROTECTION 0xCC Y`

```
---#TPCMD SET_PROTECTION 0xCC Y
Time for Set Protection: 0.058 s
```

#TPCMD GET_PROTECTION

Syntax: `#TPCMD GET_PROTECTION`

Prerequisites: none

Description: This command gets and return the RDP value from target device

Note: **STM32H5**: this command is not available

Examples: Correct command execution: 😊

```
---#TPCMD GET_PROTECTION
RDP level: 0xAA.
Readout Protection Level 0.
Time for Get Protection: 0.001 s
```

#TPCMD CHECK_PROTECTION

Syntax: `#TPCMD CHECK_PROTECTION <RDP level>`

`<RDP level>` RDP level in decimal format [0, 0.5, 1, 2]

Prerequisites: none

Description: This command checks the RDP value from target device and fails if inserted value is different from device RDP value

Note: **STM32H5**: this command is not available

Examples: Correct command execution: 😊

```
---#TPCMD CHECK_PROTECTION 0
Time for Check Protection: 0.001 s
```

Wrong command execution: ☹

```
---#TPCMD CHECK_PROTECTION 1
Request Readout Protection Level is 1.
The device's Readout Protection Level is 0.
Error!
```

Additional Commands for Product State Management

These commands are used to modify, check or get the Product State level from/to target STM32 device.
They are available for **STM32H5** series.

#TPCMD SET_PRODUCT_STATE

Syntax: `#TPCMD SET_PRODUCT_STATE <Product State>`

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

<Product State> Product State can be:
- OPEN
- PROVISIONING
- IROT_PROVISIONED
- TRUST_ZONE_CLOSED (If the device support Trust Zone)
- CLOSED
- LOCKED

<Product State> Product State can also be:
- 0xED (OPEN)
- 0x17 (PROVISIONING)
- 0x2E (IROT_PROVISIONED)
- 0xC6 (TRUST_ZONE_CLOSED (If the device support Trust Zone))
- 0x72 (CLOSED)
- 0x5C (LOCKED)

Prerequisites: none

Description: This command sets the Product State level into target device
Specific Option Byte is modified only in the appropriate Product State section

Note: Available only for **STM32H5** devices
This command is available from **libstm32.so** version **5.33**

If you want to perform Debug Authentication, please refer to this chapter.

If you set the Product State to Locked or if you set the Product State to Trust Zone Closed or Closed without setting the Debug Authentication Password or Certificate it's not possible anymore to return to Product State OPEN using Regression.

Examples: Correct command execution: 😊

```
---#TPCMD SET_PRODUCT_STATE PROVISIONING
Time for Set Product State: 0.132 s
```

#TPCMD GET_PRODUCT_STATE

Syntax: `#TPCMD GET_PRODUCT_STATE`

Prerequisites: none

Description: This command gets and return the Product State value from target device

Note: Available only for **STM32H5** devices
This command is available from **libstm32.so** version **5.33**

Examples: Correct command execution: 😊

```
---#TPCMD GET_PRODUCT_STATE
The device's product state is Provisioning [0x17].
Time for Get Product State: 0.001 s
```

#TPCMD CHECK_PRODUCT_STATE

Syntax: `#TPCMD CHECK_PRODUCT_STATE` <Product State>

<Product State> Product State can be:
- OPEN
- PROVISIONING
- IROT_PROVISIONED
- TRUST_ZONE_CLOSED (If the device support Trust Zone)
- CLOSED
- LOCKED

<Product State> Product State can also be:
- 0xED (OPEN)
- 0x17 (PROVISIONING)

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale
Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255
D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021
→smh-tech.com
info@smh-tech.com

0x2E (IROT_PROVISIONED)
0xC6 (TRUST_ZONE_CLOSED (If the device support Trust Zone))
0x72 (CLOSED)
0x5C (LOCKED)

Prerequisites: none

Description: This command checks the Product State value from target device and fails if inserted value is different from device Product State value

Note: Available only for **STM32H5** devices
This command is available from **libstm32.so** version **5.33**

Examples: Correct command execution: 😊

```
---#TPCMD CHECK_PRODUCT_STATE PROVISIONING
Time for Check Product State: 0.002 s.
```

Wrong command execution: ☹

```
---#TPCMD CHECK_PRODUCT_STATE CLOSED
Request product state is 0x72.
The device's product state is 0x17.
Error!
```

Additional Commands for Trust Zone Management

These commands are used to modify, check or get the Trust Zone level from/to target STM32 device.
They are available for **STM32H5** series.

#TPCMD SET_TRUST_ZONE

Syntax: **#TPCMD** SET_TRUST_ZONE <Trust Zone value>

<Trust Zone value> ENABLED or DISABLED

Prerequisites: none

Description: This command sets the Trust Zone value into target device
Specific Option Byte is modified only in the appropriate Trust Zone section

Note: Available only for **STM32H5** devices
This command is available from **libstm32.so** version **5.33**

Examples: Correct command execution: 😊

```
---#TPCMD SET_TRUST_ZONE ENABLED
Time for Set Trust Zone: 0.138 s.
```

#TPCMD GET_TRUST_ZONE

Syntax: **#TPCMD** GET_TRUST_ZONE

Prerequisites: none

Description: This command gets and return the Trust Zone value from target device

Note: Available only for **STM32H5** devices
This command is available from **libstm32.so** version **5.33**

Examples: Correct command execution: 😊

```
---#TPCMD GET_TRUST_ZONE
Trust Zone value is 0xB4 - Trust Zone is enabled.
Time for Get Trust Zone: 0.001 s.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

#TPCMD CHECK_TRUST_ZONE

Syntax: `#TPCMD CHECK_TRUST_ZONE <Trust Zone value>`

<Trust Zone value> ENABLED or DISABLED

Prerequisites: none

Description: This command checks the Trust Zone value from target device and fails if inserted value is different from device Trust Zone value

Note: Available only for **STM32H5** devices
This command is available from **libstm32.so** version **5.33**

Examples: Correct command execution: 😊

```
---#TPCMD CHECK_TRUST_ZONE ENABLED
Time for Check Trust Zone: 0.001 s.
```

Wrong command execution: ☹

```
---#TPCMD CHECK_TRUST_ZONE DISABLED
Device Trust Zone is enabled.
Error!
```

Additional Commands for Option Bytes Management

These commands are used to change, verify or get option bytes from/to target STM32 device.
They are available for all STM32 devices.

#TPCMD RESTORE_OPTION_BYTES

Syntax: `#TPCMD RESTORE_OPTION_BYTES`

Prerequisites: none

Description: This command is used to restore Option Bytes to a default value, when they are in a not good condition, due for example to an incorrect device configuration

With this command, as you can guess from the name itself, it is possible to return the Option Bytes to the default state (when this operation is possible)

The Default value of the Option Bytes is often the one present when the device leaves the factory

More precisely, in the various Reference Manuals you can find the various default values for each STM32 family and subfamily

Note: Please be careful when Readout Protection Level is set to 1. If you use the `#TPCMD RESTORE_OPTION_BYTES` you automatically perform a masserase of the entire Flash memory

Examples: Correct command execution: 😊

```
---#TPCMD RESTORE_OPTION_BYTES
Time for Restore Option Bytes: 0.023 s
```

#TPCMD OVERVIEW_OPTION_BYTES

Syntax: `#TPCMD OVERVIEW_OPTION_BYTES`

Prerequisites: none

Description: Reads all option bytes present in the device and represents these values in a table in the Real Time Log.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Note: none

Examples: Correct command execution: 😊

```
---#TPCMD OVERVIEW_OPTION_BYTES
  0x5200201C
  0001|0100|0001|0110|1010|1010|1111|0000
  Hex: 0x1416AAF0
```

Bits:	Name:	Value:
[31]	SWAP_BANK_OP	0
[30]	OPTCHANGEERR	0
[29]	IO_HSLV	0
[21]	SECURITY	0
[20:19]	ST_RAM_SIZE	2
[18]	IWDG_FZ_SDBY	1
[17]	IWDG_FZ_STOP	1
[15:08]	RDP	AA
[07]	NRST_STDY_D1	1
[06]	NRST_STOP_D1	1
[04]	IWDG1_SW	1
[03:02]	BOR_LEV	0

```
  0x52002028
  0000|0000|0000|0000|0000|0000|1111|1111
  Hex: 0x000000FF
```

Bits:	Name:	Value:
[31]	DMEP1	0
[27:16]	PROT_AREA_END1	000
[11:00]	PROT_AREA_START1	0FF

```
  0x52002030
  1000|0000|0000|0000|0000|0000|1111|1111
  Hex: 0x800000FF
```

Bits:	Name:	Value:
[31]	DMES1	1
[27:16]	SEC_AREA_END1	000
[11:00]	SEC_AREA_START1	0FF

```
  0x52002038
  0000|0000|0000|0000|0000|0000|1111|1111
  Hex: 0x000000FF
```

Bits:	Name:	Value:
[07:00]	WRPSn1	FF

```
  ...
Time for Overview Option Bytes: 0.009 s
```

#TPCMD WRITE_OPTION_BYTE

Syntax: **#TPCMD** WRITE_OPTION_BYTE <Address> <Value> <Reload Option Bytes>

<Address>	Address in HEX format (i.e., 0x52002020)
<Value>	Value in HEX format (i.e., 0x1416AAF0)
<Reload Option Bytes>	Optional parameter to reload or not the Option Bytes (TRUE/FALSE), default TRUE

Prerequisites: none

Description: Writes the Option Byte to the selected address (Address) with the entered value (Value).
The reserved bits of the Option Bytes are not changed.
Reload Option Bytes parameter is optional ad is available from **libstm32.so** version **5.35**

Note: Be careful not to set the Readout Protection Level to 0xCC (level 2) by mistake.

Examples: Correct command execution: 😊

```
---#TPCMD WRITE_OPTION_BYTE 0x52002020 0x1416AAF0 TRUE
Time for Write Option Byte: 0.004 s
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

#TPCMD COMPARE_OPTION_BYTE

Syntax: `#TPCMD COMPARE_OPTION_BYTE <Address> <Value>`

<Address>	Address in HEX format (i.e., 0x52002020)
<Value>	Value in HEX format (i.e., 0x1416AAF0)

Prerequisites: none

Description: Writes the Option Byte to the selected address (Address) with the entered value (Value).
The reserved bits of the Option Bytes are not changed.

Note: Be careful not to set the Readout Protection Level to 0xCC (level 2) by mistake.

Examples: Correct command execution: 😊

```
---#TPCMD COMPARE_OPTION_BYTE 0x52002020 0x1416AAD0
Time for Compare Option Byte: 0.001 s
```

Wrong command execution: ☹

```
---#TPCMD COMPARE_OPTION_BYTE 0x52002020 0x1416BBD0
Address: 0x5200201C, Option Byte Read: 0x1416AAD0, Expected: 0x1416BBD0.
Error!
```

Additional Commands for Device Informations

These commands are used to get specific information from target STM32 device.
All of these commands print into Terminal and into Real Time Log.
They are available for all STM32 devices.

#TPCMD GET_UNIQUE_ID

Syntax: `#TPCMD GET_UNIQUE_ID`

Prerequisites: none

Description: Get Unique ID from target STM32 and print it into Terminal and Real Time Log

Note: This command is available from **libstm32.so** version **5.13**

Examples: Correct command execution: 😊

```
---#TPCMD GET_UNIQUE_ID
Unique ID: 323938353038510400480038.
Time for Get Unique ID: 0.001 s
```

#TPCMD GET_FLASH_SIZE

Syntax: `#TPCMD GET_FLASH_SIZE`

Prerequisites: none

Description: Get Flash Size from target STM32 and print it into Terminal and Real Time Log

Note: This command is available from **libstm32.so** version **5.13**

Examples: Correct command execution: 😊

```
---#TPCMD GET_FLASH_SIZE
Flash Size: 0x0080 - 128 KB.
Time for Get Flash Size: 0.001 s
```


UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

#TPCMD GET_PACKAGE_ID

Syntax: `#TPCMD GET_PACKAGE_ID`

Prerequisites: none

Description: Get Package ID from target STM32 and print it into Terminal and Real Time Log

Note: This command is available from **libstm32.so** version **5.13**

Examples: Correct command execution: 😊

```
---#TPCMD GET_PACKAGE_ID
Package Type: 0x00.
Time for Get Package ID: 0.001 s
```

#TPCMD GET_DEVICE_ID

Syntax: `#TPCMD GET_DEVICE_ID`

Prerequisites: none

Description: Get Device ID from target STM32 and print it into Terminal and Real Time Log

Note: This command is available from **libstm32.so** version **5.13**

Examples: Correct command execution: 😊

```
---#TPCMD GET_DEVICE_ID
Device ID: 0x450.
Time for Get Device ID: 0.001 s
```

#TPCMD GET_REVISION_ID

Syntax: `#TPCMD GET_REVISION_ID`

Prerequisites: none

Description: Get Revision ID from target STM32 and print it into Terminal and Real Time Log

Note: This command is available from **libstm32.so** version **5.13**

Examples: Correct command execution: 😊

```
---#TPCMD GET_REVISION_ID
Revision ID: 0x1003 - Rev Y.
Time for Get Revision ID: 0.001 s
```

#TPCMD GET_DEVICE_INFORMATIONS

Syntax: `#TPCMD GET_DEVICE_INFORMATIONS`

Prerequisites: none

Description: Get Device Informations from target STM32 and print it into Terminal and Real Time Log

Note: This command prints into Terminal and Real Time Log from **libstm32.so** version **5.13**
Prior to this version this command only prints to the Real Time Log

Examples: Correct command execution: 😊

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
---#TPCMD GET_DEVICE_INFORMATIONS
Unique ID: 323938353038510400480038.
X and Y Wafer: 0x00480038.
Wafer number: 0x04.
Lot number: 0x32393835303851.
Flash Size: 0x0080 - 128 KB.
Package Type: 0x00.
Device ID: 0x450.
Revision ID: 0x1003 - Rev Y.
Time for Get Device Informations: 0.002 s
```

Additional Commands for Device Execution

#TPCMD RUN

Syntax: `#TPCMD RUN <Time [s]>`

`<Time [s]>` Time in seconds (i.e., 2 s). This time is an optional parameter.

Prerequisites: none

Description: Move the Reset line up and down quickly if no parameter `<Time [s]>` is inserted.
`#TPCMD RUN <Time [s]>` instead moves the Reset line down and high, waits for the entered time.
This command typically can be used to execute the firmware programmed in the device.

Additional Commands for Flash Memory

These commands are used to perform some specific operation into STM32 Flash memory.

#TPCMD UNPROTECT

Syntax: `#TPCMD UNPROTECT`

Prerequisites: none

Description: Remove the Flash write protected sectors and perform a Masserase F of all Flash memory
If there are no write protected sectors in the flash memory, then a standard Masserase F is performed
Available for all STM32 Devices

Note: This command is available from **libstm32.so** version **4.07**

Examples: Correct command execution: 😊

```
---#TPCMD UNPROTECT
Some sectors of Flash Memory are Write Protected.
Removing Write Protected sectors protection.
Readout Protection level is 1. Option Bytes Masserase.
Time for Unprotect: 7.324 s
```

#TPCMD ERASE F [Address] [Size]

Syntax: `#TPCMD ERASE F <Address> <Size>`

`<Address>` Address in HEX format (i.e., 0x08000000)
`<Size>` Size in HEX format (i.e., 0x2000)

Prerequisites: none

Description: Erase all memory with Page/Sector erase.
With this command, a Page/Sector Erase of the device FLASH memory will be performed, whether this memory is made up of a single bank or two banks.

Typically running the Page Erase of the entire Flash memory takes much longer than running the Masserase command.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

If the Readout Protection Level (RDP) isn't set at level 0, an error will be returned.

Available for all STM32 Devices

Note: Erase command isn't available for Option Bytes Area and OTP Area

Examples: Correct command execution: 😊

```
---#TPCMD ERASE F 0x08000000 0x2000
Time for Erase F: 0.863 s.
```

Additional Commands for Read Device Memory

These commands are used to read data from target STM32 device.
All of these commands print into Terminal and into Real Time Log.
They are available for all STM32 devices.

#TPCMD READ_MEM8 [Address] [Byte Count]

Syntax: `#TPCMD READ_MEM8 <Address> <Byte Count>`

<Address> Address in HEX format (i.e., 0x52002020)
<Byte Count> Byte count in decimal format (i.e., 8 -> eight bytes)

Prerequisites: none

Description: Read memory byte per byte from target STM32 and print it into Terminal and Real Time Log

Note: This command prints into Terminal and Real Time Log from **libstm32.so** version **4.16**

Examples: Correct command execution: 😊

```
---#TPCMD READ_MEM8 0x52002020 8
Read[0x52002020]: 0xF0
Read[0x52002021]: 0xAA
Read[0x52002022]: 0x16
Read[0x52002023]: 0x14
Read[0x52002024]: 0x00
Read[0x52002025]: 0x00
Read[0x52002026]: 0x00
Read[0x52002027]: 0x00
Time for Read Mem: 0.002 s
```

#TPCMD READ_MEM16 [Address] [16-bit Word Count]

Syntax: `#TPCMD READ_MEM16 <Address> <16-bit Word Count>`

<Address> Address in HEX format (i.e., 0x52002020)
<16-bit Word Count> 16-bit Word count in decimal format (i.e., 4 -> four 16-bit words)

Prerequisites: none

Description: Read memory 16-bit word per 16-bit word from target STM32 and print it into Terminal and Real Time Log

Note: This command prints into Terminal and Real Time Log from **libstm32.so** version **4.16**

Examples: Correct command execution: 😊

```
---#TPCMD READ_MEM16 0x52002020 4
Read[0x52002020]: 0xAAF0
Read[0x52002022]: 0x1416
Read[0x52002024]: 0x0000
Read[0x52002026]: 0x0000
Time for Read Mem: 0.002 s
```


UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

#TPCMD READ_MEM32 [Address] [32-bit Word Count]

Syntax: `#TPCMD READ_MEM32 <Address> <32-bit Word Count>`

<Address> Address in HEX format (i.e., 0x52002020)
<32-bit Word Count> 32-bit Word count in decimal format (i.e., 2 -> two 32-bit words)

Prerequisites: none

Description: Read memory 32-bit word per 32-bit word from target STM32 and print it into Terminal and Real Time Log

Note: This command prints into Terminal and Real Time Log from **libstm32.so** version **4.16**

Examples: Correct command execution: 😊

```
---#TPCMD READ_MEM32 0x52002020 2
Read[0x52002020]: 0x1416AAF0
Read[0x52002024]: 0x00000000
Time for Read Mem: 0.002 s
```

#TPCMD GET_MEMORY_HASH [HASH Type] [Memory Type] [Address] [Size]

Syntax: `#TPCMD GET_MEMORY_HASH <HASH Type> <Memory Type> <Address> <Size>`

<HASH Type> SHA512, SHA384, SHA256, SHA224 or MD5
<Memory Type> Memory Type (i.e., F, E, T, O, X, S, I)
<Address> Address in HEX format (i.e., 0x08000000) [optional parameter]
<Size> Size in HEX format (i.e., 0x02000) [optional parameter]

Prerequisites: none

Description: Read selected memory and calculate the HASH

Note: This command is available from **libstm32.so** version **5.32**
This command prints into Terminal and Real Time Log

Examples: Correct command execution: 😊

Real Time Log:

```
---TPCMD GET_MEMORY_HASH SHA512 F 0x08000000 0x2000
[0x08000000 - 0x08001FFF] SHA512:
7ECFE0D83E52E1325414A37FBD0D60D683F8378089A920EDBA41983127CB9989B5538E0F6A5A9AACD6597F53B12C9CCCAA648D87
Time for Get Memory HASH F: 0.020 s.
>|
```

Terminal:

```
01|MEMORY HASH:
01|Memory: [0x08000000 - 0x08001FFF] SHA512:
7ECFE0D83E52E1325414A37FBD0D60D683F8378089A920EDBA41983127CB9989B5538E0F6A5A9AACD6597F53B12C9CCCAA648D87
```


UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

STM32 Driver Parameters

The additional parameters are used to configure some specific option inside STM32 driver.

STM32 Standard Additional Parameters:

#TCSETPAR ENTRY_CLOCK

Syntax: `#TCSETPAR ENTRY_CLOCK <Frequency>`

`<Frequency>` Accepted parameters 4000000, 2000000, 1000000, 500000, 100000 Hz

Description: Set the JTAG/SWD frequency used into Connect procedure before raising the PLL of the device, if device PLL is available

Note: Default value 4.00 MHz

#TCSETPAR PLL_ENABLED

Syntax: `#TCSETPAR PLL_ENABLED <Value>`

`<Value>` Accepted parameters YES / NO

Description: Enable the PLL of the device at the highest possible frequency if it's available using HSI oscillator

Note: Default value YES

#TCSETPAR RESET_HARDWARE

Syntax: `#TCSETPAR RESET_HARDWARE <Value>`

`<Value>` Accepted parameters YES / NO

Description: Use Hardware reset (DIO1) into Connect procedure during halt Cortex Core
Please leave this parameter to NO except when it is strictly necessary
Usually, the Software Reset is enough to proceed with the reset of the device and to continue with the programming procedure

Note: Default value NO

#TCSETPAR CONNECT_MODE

Syntax: `#TCSETPAR CONNECT_MODE <Value>`

`<Value>` Accepted parameters:
HOT_PLUG
RESET_IMPULSE
UNDER_RESET
NORMAL

Description: Select the specific entry mode procedure used in the Connect command.
This is useful when you need to communicate with the device using a specific entry mode.
If this parameter is not set, the default entry mode will be Hot Plug.

Note: Available from driver version **5.38**

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

#TCSETPAR SAMPLING_POINT

Syntax: `#TCSETPAR SAMPLING_POINT <Value>`

`<Value>` Accepted values are in the range 1-15

Description: Use this parameter to permanently set the sampling point of the FPGA
It is recommended to leave this parameter with the default value

Note: Default value 17

#TCSETPAR HASH_TYPE

Syntax: `#TCSETPAR HASH_TYPE <Value>`

`<Value>` Accepted parameters are SHA512, SHA384, SHA256, SHA224 and MD5

Description: Available from driver version **5.32**
Adds the selected Hash calculation within the Program and Verify commands based on the FRB data.

Here you can see some examples of how this parameter works:

Consider setting the parameter like this (for other types of HASH we have the same behaviour):

```
#TCSETPAR HASH_TYPE SHA512
```

FRB source file with no holes (like a binary files):

In this example we are programming an FRB of size **8 KiB** in a **128 KiB** Flash memory.

```
---#TPCMD PROGRAM F
[0x08000000 - 0x08001FFF] SHA512:
7ECFE0D83E52E1325414A37FBD0D60D683F8378089A920EDBA41983127CB9989B5538E0F6A5A9AACD6597F53B12C9CCCAA648D87
Time for Program F: 0.077 s.
>|
---TPCMD VERIFY F R
[0x08000000 - 0x08001FFF] SHA512:
7ECFE0D83E52E1325414A37FBD0D60D683F8378089A920EDBA41983127CB9989B5538E0F6A5A9AACD6597F53B12C9CCCAA648D87
Time for Verify Readout F: 0.020 s.
>|
```

You can also get the same output from the Terminal:

```
#1|RUN STM32.prj
01|PROGRAM:
01|Source: [0x08000000 - 0x08001FFF] SHA512:
7ECFE0D83E52E1325414A37FBD0D60D683F8378089A920EDBA41983127CB9989B5538E0F6A5A9AACD6597F53B12C9CCCAA648D87
01|VERIFY:
01|Target: [0x08000000 - 0x08001FFF] SHA512:
7ECFE0D83E52E1325414A37FBD0D60D683F8378089A920EDBA41983127CB9989B5538E0F6A5A9AACD6597F53B12C9CCCAA648D87
01|#1|RUN STM32.prj
01|>
```

The HASH calculation is performed only on the FRB data (including any fills bytes added to align with the memory granularity).

During the Program the hash calculation is performed on the data transferred to the device.
During the Verify the hash calculation is performed on the data read from the Flash memory at the addresses defined by the FRB file.

So, it is clear that if you run GET_MEMORY_HASH on all the Flash memory, the value obtained will be different from the one above because the rest of the memory (where it is not programmed) has a value of 0xFF.

Instead, if you execute the GET_MEMORY_HASH on the section covered by the FRB then the value obtained will be the same.

UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

```
---TPCMD GET_MEMORY_HASH SHA512
[0x08000000 - 0x0801FFFF] SHA512:
FB535CA92520FD7D46CF1096B643BC39FF9F89CF9061CFA0A72956675D38E000664F74BE7A769F0EADD59A1BE382DD3212317CE3
Time for Get Memory HASH F: 0.077 s.
>|
---TPCMD GET_MEMORY_HASH SHA512 F 0x08000000 0x2000
[0x08000000 - 0x08001FFF] SHA512:
7ECFE0D83E52E1325414A37FBD0D60D683F8378089A920EDBA41983127CB9989B5538E0F6A5A9AACD6597F53B12C9CCCAA648D87
Time for Get Memory HASH F: 0.020 s.
>|
```

FRB source file with holes (like a hex-srec file with discontinued blocks in the same memory):

In this example we are programming an FRB with two blocks in a **128 KiB** Flash memory.

Data Type	File Name	Type	Block Start Address	Block End Address	Block Encoding	Memory Area	Target Start Address	Target End Address	Target Addressing
DATA	stm32.frb	FRB	0x00000000	0x000003FF	BYTE	[F] - Flash	0x08000000	0x080003FF	BYTE
DATA	stm32.frb	FRB	0x00000000	0x000007FF	BYTE	[F] - Flash	0x08004000	0x080047FF	BYTE

1st block of data is from address **0x8000000** with size **1KB**.
2nd block of data is from address **0x8004000** with size **2KB**.

The two blocks of data are discontinued and the hole in the middle is not filled by FlashRunner with the Flash memory blank value because the hole is larger than the Flash memory programming page size.
So basically, the FlashRunner will program the 1st block and then will program the 2nd block, but FlashRunner will not program 0xFF (blank value of Flash memory) in the memory section between these two blocks.

In this case, if you set the `#TCSETPAR HASH_TYPE SHA512` you can see the following output:

```
---#TPCMD PROGRAM F
[0x08000000 - 0x080047FF] SHA512:
B3AA97E4AF64C333D881B92FB7405B6452E50529179826C87D0BDED9EA8EA54F607B6287D606F470DDD8ED3DBE63771972E9F334
Time for Program F: 0.054 s.
>|
---TPCMD VERIFY F R
[0x08000000 - 0x080047FF] SHA512:
B3AA97E4AF64C333D881B92FB7405B6452E50529179826C87D0BDED9EA8EA54F607B6287D606F470DDD8ED3DBE63771972E9F334
Time for Verify Readout F: 0.012 s.
>|
```

You need to pay attention to two points:

1. The addresses displayed are [0x08000000 - 0x080047FF] because they are the first programmed address and the last programmed address.
   So, the fact that there is a hole in the FRB file and therefore not all the bytes within this address range have been programmed is not shown.

2. The hash value is not calculated on the data within the address [0x08000000 - 0x080047FF] but on the data present in the FRB.
   So, to be clear, in this case we have two blocks of data, so the hash is calculated from 0x08000000 to 0x08000FFF (1st block) and from 0x08004000 to 0x080047FF (2nd block).

So, if you calculate the HASH of the memory in the range [0x08000000 - 0x080047FF] you will get a different value than the one obtained above.

```
---TPCMD GET_MEMORY_HASH SHA512 0x08000000 0x4800
[0x08000000 - 0x080047FF] SHA512:
C5FC0CE55E5D202F54D187A394C406F356249E9E783F203475B222CDD70ABA8BE3B916ED12B0A3A71352666976531A0DF80B6EF9
Time for Get Memory HASH F: 0.077 s.
>|
```

Note: By default, this parameter is not set

#TCSETPAR CONNECT_UNDER_RESET [Obsolete]

Syntax: `#TCSETPAR CONNECT_UNDER_RESET <Value>`

`<Value>` Accepted parameters YES / NO

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Description: Available from driver version 4.07 to driver version 4.10
Removed with driver version 4.11
Perform a Hardware reset (DIO1) before the Connect procedure

Note: Default value NO

STM32 Additional Parameters for External Memory management:

```
#TCSETPAR PIN_QSPI_BANK1_NCS
#TCSETPAR PIN_QSPI_BANK1_IO0
#TCSETPAR PIN_QSPI_BANK1_IO1
#TCSETPAR PIN_QSPI_BANK1_IO2
#TCSETPAR PIN_QSPI_BANK1_IO3

#TCSETPAR PIN_QSPI_BANK2_NCS
#TCSETPAR PIN_QSPI_BANK2_IO0
#TCSETPAR PIN_QSPI_BANK2_IO1
#TCSETPAR PIN_QSPI_BANK2_IO2
#TCSETPAR PIN_QSPI_BANK2_IO3

#TCSETPAR PIN_QSPI_CLOCK

#TCSETPAR QSPI_BANK
#TCSETPAR QSPI_PROTOCOL
#TCSETPAR QSPI_DDR_ENABLE

#TCSETPAR QSPI_OSCILLATOR_
#TCSETPAR QSPI_PLLM [Value]
#TCSETPAR QSPI_PLLN [Value]
#TCSETPAR QSPI_PLLP [Value]
#TCSETPAR QSPI_PLLQ [Value]
#TCSETPAR QSPI_PLLR [Value]

#TCSETPAR QSPI_SYSCLK_DIV [Value]
#TCSETPAR QSPI_AHBCLK_DIV [Value]
#TCSETPAR QSPI_APB1CLK_DIV [Value]
#TCSETPAR QSPI_APB2CLK_DIV [Value]
#TCSETPAR QSPI_APB3CLK_DIV [Value]
#TCSETPAR QSPI_APB4CLK_DIV [Value]

#TCSETPAR QSPI_SPI_PRESCALER [Value]
```

For a detailed description of these Additional Parameters please refer to STM32 – H7 – Program an External Flash Memory.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

STM32 Readout Protection Level Regression with Password

Readout Protection Level regression with password is available only for STM32U5 Series Arm®-based 32-bit MCUs and STM32WBA Series Arm®-based 32-bit MCUs.

For all other informations please refer to specific STMicroelectronics STM32U5 and STM32WBA reference manual.

Device life cycle managed by readout protection (RDP) transitions

It is easy to move from level 0 or level 0.5 to level 1 by changing the value of the RDP byte to any value (except 0xCC).
By programming the 0xCC value in the RDP byte, it is possible to go to level 2 either directly from level 0 or from level 0.5 or from level 1.
Once in level 2, it is no longer possible to modify the readout protection level unless an OEM2 key is Provisioned.

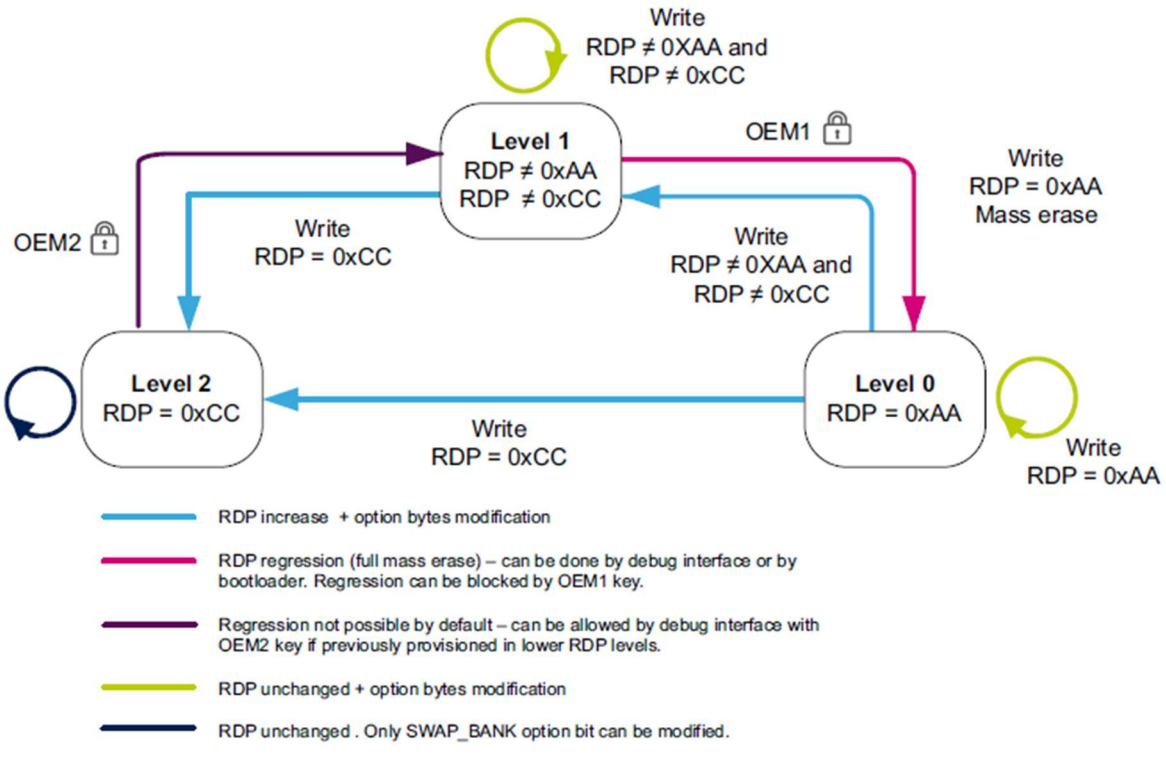
When the RDP is reprogrammed to the value 0xAA to move from level 1 to level 0, a masserase of the flash main memory and all SRAMs is performed. The OTP area is not erased.

At RDP level 0.5, it is not possible to request RDP level 0.
Instead, an RDP increase to level 1 followed by an RDP regression to level 0 is required.

When the RDP is programmed to the value 0x55 to move from level 1 to level 0.5, a partial masserase of the flash main memory is performed. The OTP area is not erased.
The RDP level 0.5 and partial nonsecure erase are only available when TrustZone is active.

> *Note: Full masserase is performed only when level 1 is active and level 0 requested. When the protection level is increased (0 to 0.5, 0 to 1, 0.5 to 1, 1 to 2, 0 to 2 or 0.5 to 2), there is no masserase.*

RDP level transition scheme when TrustZone is disabled (TZEN = 0)



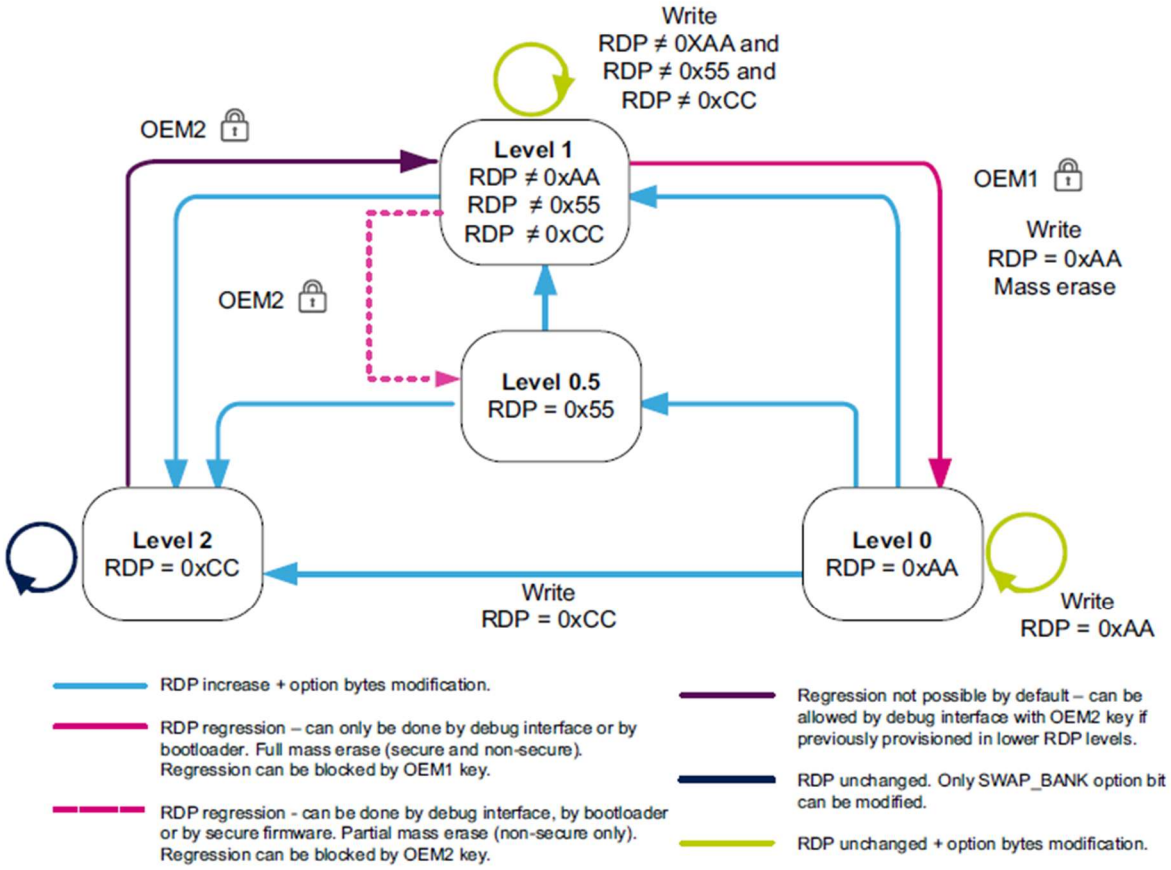
HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

RDP level transition scheme when TrustZone is enabled (TZEN = 1)



OEM1/OEM2 lock activation

Two 64-bit keys (OEM1KEY and OEM2KEY) can be defined in order to lock the RDP regression.

OEM1KEYcan be modified:

- in readout protection level 0
- in readout protection level 0.5 or 1 if OEM1LOCK is 0

OEM2KEYcan be modified:

- in readout protection level 0 or 0.5
- in readout protection level 1 if OEM2LOCK is 0

FlashRunner 2 commands for RDP regression with password

```
#TPCMD RDP_PASSWORD_STORE <RDP level> <PW 1st 32 bit> <PW 2nd 32 bit>
#TPCMD RDP_PASSWORD_STORE_LOCK <RDP level> <PW 1st 32 bit> <PW 2nd 32 bit>
#TPCMD RDP_PASSWORD_SET <RDP level> <PW 1st 32 bit> <PW 2nd 32 bit>
#TPCMD RDP_PASSWORD_SET_UNLOCK <RDP level> <PW 1st 32 bit> <PW 2nd 32 bit>
#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE <RDP level> <PW 1st 32 bit> <PW 2nd 32 bit>
#TPCMD RDP_PASSWORD_REMOVE <RDP level>
```

*Note: These commands are available from **libstm32.so** version **5.15**.*

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

OEM1/OEM2 lock activation with FlashRunner 2

First of all, please check the status of your device performing a connect command.
We assume that we start from a virgin device or a device with standard Option Bytes configuration.

Here an example of the connect command:

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
IDCODE: 0x0BE12477.
Designer: 0x23B, Part Number: 0xBE12, Version: 0x0.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x14770015, Type: AMBA AHB5 bus.
AP[0] ROM table base address 0xE00FE000.
CPUID: 0x410FD214.
Implementer Code: 0x41 - [ARM].
Found Cortex M33 revision r0p4.
Program counter value is 0xEFFFFFFE.
Cortex M33 Core halted [0.001 s].
Device configuration: [0x1FEFF8AA]
 * The device's RDP level is 0 [0xAA].
 * OEM1 RDP key read during the is virgin. OEM1 RDP lock mechanism is not active.
 * OEM2 RDP key read during the is virgin. OEM2 RDP lock mechanism is not active.
 * BOR Level 0. Reset level threshold is around 1.7 V.
 * No reset generated when entering the Stop mode.
 * No reset generated when entering the Standby mode.
 * No reset generated when entering the Shutdown mode.
 * SRAM1, SRAM3 and SRAM4 not erased when a system reset occurs.
 * Software independent watchdog selected.
 * Independent watchdog counter is running in Stop mode.
 * Independent watchdog counter is running in Standby mode.
 * Software window watchdog selected.
 * Bank 1 and bank 2 address are not swapped.
 * Dual-bank Flash with contiguous addresses.
 * Backup RAM ECC detection and correction disabled.
 * SRAM3 ECC detection and correction disabled.
 * SRAM2 ECC detection and correction disabled.
 * SRAM2 is not erased when a system reset occurs.
 * BOOT0 signal is taken from PH3/BOOT0 pin.
 * USB power delivery dead-battery disabled/TDI pull-up activated.
 * High-speed IO at low VDD voltage feature disabled (VDD can exceed 2.5 V).
 * High-speed IO at low VDDIO2 voltage feature disabled (VDDIO2 can exceed 2.5 V).
 * Global TrustZone security disabled.
PLL enabled using internal HSI oscillator.
Requested Clock is 5.00 MHz.
Generated Clock is 5.00 MHz.
Good samples: 16 [Range 0-15].
IDCODE: 0x0BE12477.
Designer: 0x23B, Part Number: 0xBE12, Version: 0x0.
ID-Code read correctly at 5.00 MHz.
Time for Connect: 0.108 s.
```

In the log, the three most important lines for this chapter are highlighted in green.
Note that the RDP level is 0 (value 0xAA) and no passwords have been entered for regression from RDP1 or RDP2 level.

Before starting to describe the commands remember that:

There are 2 password keys - OEM1KEY, OEM2KEY, both 64-bit, write only, unreadable.

- OEM1KEY is used to manage the RDP level regression from Level 1 to Level 0.
- OEM2KEY is used to manage the RDP level regression from Level 2 to Level 1 and Level 1 to Level 0.5.

To unlock and for RDP regression, the correct key must be shifted through JTAG / SWD pins during reset.
Unlocking the device with a password is possible only once per power cycle.
Password key-based RDP regressions are available through the debug interface or via the system bootloader.

Two 64-bit keys are defined in embedded flash to independently protect secure (OEM1) and non-secure (OEM2) application codes, as shown.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Available commands for RDP regression with password

#TPCMD RDP_PASSWORD_STORE

Syntax: `#TPCMD RDP_PASSWORD_STORE <RDP level> <PW 1st 32 bit> <PW 2nd 32 bit>`

<RDP level>	RDP level can be 1 or 2
<PW 1st 32 bit>	First 32-bit password with HEX format (i.e., 0xDEADBEEF)
<PW 2nd 32 bit>	Second 32-bit password with HEX format (i.e., 0xC0FFEE00)

Prerequisites:

OEM1KEY can be modified:
- ✓ in readout protection level 0
- ✓ in readout protection level 0.5 or 1 if OEM1LOCK is 0

OEM2KEY can be modified:
- ✓ in readout protection level 0 or 0.5
- ✓ in readout protection level 1 if OEM2LOCK is 0

Description: This command stores the RDP1 or RDP2 password inside the device
The RDP level is not changed

Note: This command is available from **libstm32.so** version **5.15**
Suggested **libstm32** driver version for this command is **5.29** or higher

Examples: Correct command execution: 😊

```
---#TPCMD RDP_PASSWORD_STORE 1 0xDEADBEEF 0xC0FFEE00
Try to set the device RDP level 1 password.
 * RDP level 1 password set correctly
Time for RDP Password Store: 0.077 s
```

```
---#TPCMD RDP_PASSWORD_STORE 2 0xDEADBEEF 0xC0FFEE00
Try to set the device RDP level 2 password.
 * RDP level 2 password set correctly
Time for RDP Password Store: 0.077 s
```

Now if you try to reconnect to target device you can see that RDP value is not changed but the OEM1 and/or OEM2 RDP lock mechanism are active.

```
* The device's RDP level is 0 [0xAA].
* OEM1 RDP key read is not virgin. OEM1 RDP lock mechanism is active.
* OEM2 RDP key read is not virgin. OEM2 RDP lock mechanism is active.
```

Now if you use the #TPCMD SET_PROTECTION [Value] you can modify the RDP value with the password/s already stored inside the device.

Wrong command execution: ☹

If RDP level of the device is set to 1 and you try to execute the following command, you receive an error:

```
---#TPCMD RDP_PASSWORD_STORE 1 0xDEADBEEF 0xC0FFEE00
Unable to set RDP to level 1 with password.
OEM1lock bit set. To set RDP level 1 with password the RDP level of the device must be 0.
Error!
```

```
---#TPCMD RDP_PASSWORD_STORE 2 0xDEADBEEF 0xC0FFEE00
Unable to set RDP to level 2 with password.
OEM2lock bit set. To set RDP level 2 with password the RDP level of the device must be 0 or 0.5
Error!
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

#TPCMD RDP_PASSWORD_STORE_LOCK

Syntax: `#TPCMD RDP_PASSWORD_STORE_LOCK <RDP level> <PW 1st 32 bit> <PW 2nd 32 bit>`

<RDP level>	RDP level can be 1 or 2
<PW 1st 32 bit>	First 32-bit password with HEX format (i.e., 0xDEADBEEF)
<PW 2nd 32 bit>	Second 32-bit password with HEX format (i.e., 0xC0FFEE00)

Prerequisites:

OEM1KEY can be modified:
- ✓ in readout protection level 0
- ✓ in readout protection level 0.5 or 1 if OEM1LOCK is 0

OEM2KEY can be modified:
- ✓ in readout protection level 0 or 0.5
- ✓ in readout protection level 1 if OEM2LOCK is 0

Description: This command stores the RDP1 or RDP2 password inside the device
The RDP level is changed to level 1 or 2

Note: This command is available from **libstm32.so** version **5.15**
Suggested **libstm32** driver version for this command is **5.29** or higher
Please be careful when setting RDP to level 2! If something goes wrong the JTAG/SWD debug port will become unavailable forever

Examples: Correct command execution: 😊

RDP level 1:

```
---#TPCMD RDP_PASSWORD_STORE_LOCK 1 0xDEADBEEF 0xC0FFEE00
Try to set the device RDP password.
 * RDP level 1 password set correctly.
Try to change Readout Protection Level to 1.
 * Readout Protection Level set correctly to 1.
Time for RDP Password Store and Lock: 0.156 s
```

Now if you try to reconnect to target device you can see that RDP value is changed to 1 and the OEM1 RDP lock mechanism is active.

```
* The device's RDP level is 1 [0xBB].
* OEM1 RDP key read during the is not virgin. OEM1 RDP lock mechanism is active.
* OEM2 RDP key read during the is virgin. OEM2 RDP lock mechanism is not active.
```

RDP level 2:

```
---#TPCMD RDP_PASSWORD_STORE_LOCK 2 0xDEADBEEF 0xC0FFEE00
Try to set the device RDP password.
 * RDP level 2 password set correctly.
Try to change Readout Protection Level to 2.
 * Readout Protection Level set correctly to 2.
Time for RDP Password Store and Lock: 0.144 s
```

Now if you try to reconnect to target device you can see that RDP value is changed to 2 and the OEM2 RDP lock mechanism is active.

```
Protocol selected SWD.
Entry Clock is 1.00 MHz.
Trying Hot Plug connect procedure.
IDCODE: 0x0BE12477.
Designer: 0x23B, Part Number: 0xBE12, Version: 0x0.
ID-Code read correctly at 1.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x14770015, Type: AMBA AHB5 bus.
AP[0] ROM table base address 0xE00FE000.
Your device could be locked by RDP level 2 with password.
If you want to unlock it, try to use the following command:
 * RDP_PASSWORD_SET_UNLOCK to unlock the device and move the RDP level to 1.
```

Wrong command execution: ☹

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

```
---#TPCMD RDP_PASSWORD_STORE_LOCK 1 0xDEADBEEF 0xC0FFEE00
Unable to set RDP to level 1 with password.
OEM1lock bit is set. To set RDP level 1 with password the RDP level of the device must be 0.
Error!
```

```
---#TPCMD RDP_PASSWORD_STORE_LOCK 2 0xDEADBEEF 0xC0FFEE00
Unable to set RDP to level 2 with password.
OEM2lock bit set. To set RDP level 2 with password the RDP level of the device must be 0 or 0.5
Error!
```

#TPCMD RDP_PASSWORD_SET

Syntax: **#TPCMD** RDP_PASSWORD_SET <RDP level> <PW 1st 32 bit> <PW 2nd 32 bit>

<RDP level>	RDP level can be 1 or 2
<PW 1st 32 bit>	First 32-bit password with HEX format (i.e., 0xDEADBEEF)
<PW 2nd 32 bit>	Second 32-bit password with HEX format (i.e., 0xC0FFEE00)

Prerequisites: OEM1KEY is set to 1 for RDP level 1 (Password RDP 1 is already stored inside the device)
OEM2KEY is set to 1 for RDP level 2 (Password RDP 2 is already stored inside the device)
Typically, you've already run one of these commands before this command: **#TPCMD** RDP_PASSWORD_STORE or **#TPCMD** RDP_PASSWORD_STORE_LOCK

Description: This command sets the RDP1 or RDP2 temporary password to prepare the device to be unlocked
If you turn your device off and on, this temporary password is lost
For RDP1 level the device RDP level is not changed.
For RDP2 level this command automatically unlocks the device and move the RDP level to 1 with value 0xFF

Note: This command is available from **libstm32.so** version **5.15**
Suggested **libstm32** driver version for this command is **5.29** or higher

Examples: Correct command execution: 😊

```
---#TPCMD RDP_PASSWORD_SET 1 0xDEADBEEF 0xC0FFEE00
Try to set RPD level 1 password to unlock the device.
 * Set RDP level 1 password correctly to unlock device.
Time for RDP Password Set: 0.001 s
```

```
---#TPCMD RDP_PASSWORD_SET 2 0xDEADBEEF 0xC0FFEE00
Try to set RDP level 2 password to unlock the device.
 * Set RDP level 2 password correctly to unlock device.
Power off and on the device now!
 * Reconnect to the device now.
Time for RDP Password Set: 0.134 s
```

Skipped command execution: 😐

```
--- #TPCMD RDP_PASSWORD_SET 1 0xDEADBEEF 0xC0FFEE00
RDP level 1 is not set into target device. Skipped procedure.
Time for RDP Password Set: 0.001 s
```

```
--- #TPCMD RDP_PASSWORD_SET 2 0xDEADBEEF 0xC0FFEE00
RDP level 2 is not set into target device. Skipped procedure.
Time for RDP Password Set: 0.001 s
```

```
--- #TPCMD RDP_PASSWORD_SET 1 0xDEADBEEF 0xC0FFEE00
RDP level 1 is set into target device but RDP level is not 1. Skipped procedure.
Time for RDP Password Set: 0.001 s
```

```
--- #TPCMD RDP_PASSWORD_SET 2 0xDEADBEEF 0xC0FFEE00
RDP level 2 is set into target device but RDP level is not 2. Skipped procedure.
Time for RDP Password Set: 0.001 s
```


UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

#TPCMD RDP_PASSWORD_SET_UNLOCK

Syntax: `#TPCMD RDP_PASSWORD_SET_UNLOCK <RDP level> <PW 1st 32 bit> <PW 2nd 32 bit>`

<RDP level>	RDP level can be 1 or 2
<PW 1st 32 bit>	First 32-bit password with HEX format (i.e., 0xDEADBEEF)
<PW 2nd 32 bit>	Second 32-bit password with HEX format (i.e., 0xC0FFEE00)

Prerequisites: OEM1KEY is set to 1 for RDP level 1 (Password RDP 1 is already stored inside the device)
OEM2KEY is set to 1 for RDP level 2 (Password RDP 2 is already stored inside the device)
Typically, you've already run one of these commands before this command: `#TPCMD RDP_PASSWORD_STORE` or `#TPCMD RDP_PASSWORD_STORE_LOCK`

Description: This command sets the RDP1 or RDP2 temporary password to prepare the device to be unlocked
The RDP level is decreased by 1 automatically

Note: This command is available from **libstm32.so** version **5.15**
Suggested **libstm32** driver version for this command is **5.29** or higher

Examples: Correct command execution: 😊

RDP level 2:

```
---#TPCMD RDP_PASSWORD_SET_UNLOCK 2 0xDEADBEEF 0xC0FFEE00
Try to set RDP level 2 password to unlock the device.
 * Set RDP level 2 password correctly to unlock device.
Power off and on the device now!
 * Reconnect to the device now.
Try to change Readout Protection Level to 1.
 * Readout Protection Level set correctly to 1.
Time for RDP Password Set and Unlock: 0.258 s
```

Now if you try to reconnect to target device you can see that RDP value is changed to 1 but the OEM1 RDP and OEM2 RDP lock mechanism are active because passwords are already stored inside the device.

```
* The device's RDP level is 1 [0xBB].
* OEM1 RDP key read is not virgin. OEM1 RDP lock mechanism is active.
* OEM2 RDP key read is not virgin. OEM2 RDP lock mechanism is active.
```

RDP level 1:

```
---#TPCMD RDP_PASSWORD_SET_UNLOCK 1 0xDEADBEEF 0xC0FFEE00
Try to set RPD level 1 password to unlock the device.
 * Set RDP level 1 password correctly to unlock device.
Try to change Readout Protection Level to 0.
 * Readout Protection Level set correctly to 0.
Time for RDP Password Set and Unlock: 0.058 s
```

Now if you try to reconnect to target device you can see that RDP value is changed to 0 but the OEM1 RDP lock mechanism is active because password is already stored inside the device.

```
* The device's RDP level is 0 [0xAA].
* OEM1 RDP key read is not virgin. OEM1 RDP lock mechanism is active.
* OEM2 RDP key read is virgin. OEM2 RDP lock mechanism is not active.
```

Skipped command execution: 😐

```
---#TPCMD RDP_PASSWORD_SET_UNLOCK 1 0xDEADBEEF 0xC0FFEE00
RDP level 1 password is not set into target device. Skipped procedure.
Time for RDP Password Set and Unlock: 0.001 s
```

```
---#TPCMD RDP_PASSWORD_SET_UNLOCK 1 0xDEADBEEF 0xC0FFEE00
RDP level 1 password is set into target device but RDP level is not 1. Skipped procedure.
Time for RDP Password Set and Unlock: 0.001 s
```

```
---#TPCMD RDP_PASSWORD_SET_UNLOCK 2 0xDEADBEEF 0xC0FFEE00
RDP level 2 password is not set into target device. Skipped procedure.
Time for RDP Password Set and Unlock: 0.001 s
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

```
---#TPCMD RDP_PASSWORD_SET_UNLOCK 1 0xDEADBEEF 0xC0FFEE00
RDP level 2 password is not set into target device. Skipped procedure.
Time for RDP Password Set and Unlock: 0.001 s
```

Wrong command execution: ☹ Wrong Password!

```
---#TPCMD RDP_PASSWORD_SET_UNLOCK 1 0xC0FFEE00 0xC0FFEE00
Try to set RDP level 1 password to unlock the device.
 * Set RDP level 1 password correctly to unlock device.
Try to change Readout Protection Level to 0.
OPTWERR: Option write error.
This bit is set by hardware when the options bytes are written with an invalid configuration.
Error!
```

#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE

Syntax: **#TPCMD** RDP_PASSWORD_SET_UNLOCK_REMOVE <RDP level> <PW 1st 32 bit> <PW 2nd 32 bit>

<RDP level>	RDP level can be 1 or 2
<PW 1st 32 bit>	First 32-bit password with HEX format (i.e., 0xDEADBEEF)
<PW 2nd 32 bit>	Second 32-bit password with HEX format (i.e., 0xC0FFEE00)

Prerequisites: OEM1KEY is set to 1 for RDP level 1 (Password RDP 1 is already stored inside the device)
OEM2KEY is set to 1 for RDP level 2 (Password RDP 2 is already stored inside the device)
Typically, you've already run one of these commands before this command: **#TPCMD** RDP_PASSWORD_STORE or **#TPCMD** RDP_PASSWORD_STORE_LOCK

Description: This command sets the RDP1 or RDP2 temporary password to prepare the device to be unlocked
The RDP level is decreased by 1 automatically
RDP1 password is removed into the device, instead to remove RDP2 password you need to move the device to RDP level 0 or 0.5

Note: This command is available from **libstm32.so** version **5.15**
Suggested **libstm32** driver version for this command is **5.29** or higher

Examples: Correct command execution: ☺

RDP level 2:

```
---#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 2 0xDEADBEEF 0xC0FFEE00
Try to set RDP level 2 password to unlock the device.
 * Set RDP level 2 password correctly to unlock device.
Power off and on the device now!
 * Reconnect to the device now.
Try to change Readout Protection Level to 1.
 * Readout Protection Level set correctly to 1.
To remove the RDP level 2 password you need to set the device RDP level to 0 or 0.5.
Time for RDP Password Set, Unlock and Remove: 0.337 s
```

Now if you try to reconnect to target device you can see that RDP value is changed to 1 but the OEM1 RDP and OEM2 RDP lock mechanism are active because passwords are already stored inside the device.

```
* The device's RDP level is 1 [0xBB].
* OEM1 RDP key read is not virgin. OEM1 RDP lock mechanism is active.
* OEM2 RDP key read is not virgin. OEM2 RDP lock mechanism is active.
```

RDP level 1:

```
---#TPCMD RDP_PASSWORD_SET_UNLOCK 1 0xDEADBEEF 0xC0FFEE00
Try to set RPD level 1 password to unlock the device.
 * Set RDP level 1 password correctly to unlock device.
Try to change Readout Protection Level to 0.
 * Readout Protection Level set correctly to 0.
Try to remove the device RDP level 1 password.
* RDP level 1 password removed correctly.
Time for RDP Password Set, Unlock and Remove: 0.058 s
```

Now if you try to reconnect to target device you can see that RDP value is changed to 0 and the OEM1 RDP lock mechanism is not active.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
* The device's RDP level is 0 [0xAA].
* OEM1 RDP key read during the is virgin. OEM1 RDP lock mechanism is not active.
* OEM2 RDP key read is not virgin. OEM2 RDP lock mechanism is active.
```

Skipped command execution: 😐

```
---#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 1 0xDEADBEEF 0xC0FFEE00
RDP level 1 password is not set into target device. Skipped procedure.
Time for RDP Password Set, Unlock and Remove: 0.001 s
```

```
---#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 1 0xDEADBEEF 0xC0FFEE00
RDP level 1 password is set into target device but RDP level is not 1. Skipped procedure.
Time for RDP Password Set, Unlock and Remove: 0.001 s
```

```
---#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 2 0xDEADBEEF 0xC0FFEE00
RDP level 2 password is not set into target device. Skipped procedure.
Time for RDP Password Set, Unlock and Remove: 0.001 s
```

```
---#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 2 0xDEADBEEF 0xC0FFEE00
RDP level 2 password is set into target device but RDP level is not 2. Skipped procedure.
Time for RDP Password Set, Unlock and Remove: 0.001 s
```

Wrong command execution: ☹ Wrong Password!

```
---#TPCMD RDP_PASSWORD_SET_UNLOCK 1 0xC0FFEE00 0xC0FFEE00
Try to set RDP level 1 password to unlock the device.
 * Set RDP level 1 password correctly to unlock device.
Try to change Readout Protection Level to 0.
OPTWERR: Option write error.
This bit is set by hardware when the options bytes are written with an invalid configuration.
Error!
```

#TPCMD RDP_PASSWORD_REMOVE

Syntax: `#TPCMD RDP_PASSWORD_REMOVE <RDP level>`

`<RDP level>` RDP level can be 1 or 2

Prerequisites: Device RDP level is 0 or 0.5
OEM1KEY is set to 1 for RDP level 1
OEM2KEY is set to 1 for RDP level 2

Description: This command removes the RDP1 or RDP2 password inside the device.
The new RDP1 or RDP2 password is set to **0xFFFFFFFF - 0xFFFFFFFF**

Note: This command is available from **libstm32.so** version **5.15**.
Suggested **libstm32** driver version for this command is **5.29** or higher.

Examples: Correct command execution: 😊

```
---#TPCMD RDP_PASSWORD_REMOVE 1
Try to remove the device RDP level 1 password.
 * RDP level 1 password removed correctly.
Time for RDP Password Remove: 0.058 s
```


```
---#TPCMD RDP_PASSWORD_REMOVE 2
Try to remove the device RDP level 2 password.
 * RDP level 2 password removed correctly.
Time for RDP Password Remove: 0.058 s
```

Now if you try to reconnect to target device you can see that RDP value is 0 and the OEM1 RDP and OEM2 RDP lock mechanism are not active.

```
* The device's RDP level is 0 [0xAA].
* OEM1 RDP key read is virgin. OEM1 RDP lock mechanism is not active.
* OEM2 RDP key read is virgin. OEM2 RDP lock mechanism is not active.
```

Skipped command execution: 😐

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
--- #TPCMD RDP_PASSWORD_REMOVE 1
RDP level 1 password is not set into target device. Skipped procedure.
Time for RDP Password Remove: 0.001 s
```

```
--- #TPCMD RDP_PASSWORD_REMOVE 2
RDP level 2 password is not set into target device. Skipped procedure.
Time for RDP Password Remove: 0.001 s
```

Wrong command execution: ☹

```
--- #TPCMD RDP_PASSWORD_REMOVE 1
Unable to remove RDP level 1 password.
To remove RDP level 1 password the RDP level of the device must be 0.
Error!
```

```
--- #TPCMD RDP_PASSWORD_REMOVE 2
Unable to remove RDP level 2 password.
To remove RDP level 2 password the RDP level of the device must be 0.
Error!
```

Examples for RDP regression with password

RDP regression with password using STM32U5

Project commands:

```
#TPSTART
#TPCMD CONNECT
#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 2 0xC0FFEE00 0xDEADBEEF
#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 1 0xDEADBEEF 0xC0FFEE00
#TPCMD RDP_PASSWORD_REMOVE 1
#TPCMD RDP_PASSWORD_REMOVE 2
#TPCMD RDP_PASSWORD_STORE_LOCK 1 0xDEADBEEF 0xC0FFEE00
#TPCMD RDP_PASSWORD_STORE_LOCK 2 0xC0FFEE00 0xDEADBEEF
#TPCMD DISCONNECT
#TPCMD CONNECT
#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 2 0xC0FFEE00 0xDEADBEEF
#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 1 0xDEADBEEF 0xC0FFEE00
#TPCMD RDP_PASSWORD_REMOVE 2
#TPCMD RDP_PASSWORD_REMOVE 1
#TPCMD DISCONNECT
#TPEND
```

Real Time log:

```
---#TPSTART
Load SWD FPGA version 0x00001215.
>|
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 1.00 MHz.
Trying Hot Plug connect procedure.
IDCODE: 0x0BE12477.
Designer: 0x23B, Part Number: 0xBE12, Version: 0x0.
ID-Code read correctly at 1.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x14770015, Type: AMBA AHB5 bus.
AP[0] ROM table base address 0xE00FE000.
CPUID: 0x410FD214.
Implementer Code: 0x41 - [ARM].
Found Cortex M33 revision r0p4.
Program counter value is 0x0BF95C74.
Cortex M33 Core halted [0.002 s].
Device configuration: [0x91A9F9AA]
 * The device's Readout Protection level is 0 [0xAA].
 * OEM1 RDP key read is not virgin. OEM1 RDP lock mechanism is active.
 * OEM2 RDP key read is not virgin. OEM2 RDP lock mechanism is active.
 * BOR level 1. Reset level threshold is around 2.0 V.
 * No reset generated when entering the Stop mode.
 * No reset generated when entering the Standby mode.
 * No reset generated when entering the Shutdown mode.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
 * SRAM1, SRAM3 and SRAM4 not erased when a system reset occurs.
 * Software independent watchdog selected.
 * Independent watchdog counter is frozen in Stop mode.
 * Independent watchdog counter is frozen in Standby mode.
 * Software window watchdog selected.
 * Bank 1 and bank 2 address are not swapped.
 * Dual-bank Flash with contiguous addresses.
 * Backup RAM ECC detection and correction enable.
 * SRAM3 ECC detection and correction disabled.
 * SRAM2 ECC detection and correction disabled.
 * SRAM2 erased when a system reset occurs.
 * BOOT0 signal is taken from the option bit nBOOT0.
 * USB power delivery dead-battery disabled/TDI pull-up activated.
 * High-speed IO at low VDD voltage feature disabled (VDD can exceed 2.5 V).
 * High-speed IO at low VDDIO2 voltage feature disabled (VDDIO2 can exceed 2.5 V).
 * Global TrustZone security enabled.
PLL enabled using internal HSI oscillator.
Requested Clock is 5.00 MHz.
Generated Clock is 5.00 MHz.
Good samples: 16 [Range 0-15].
IDCODE: 0x0BE12477.
Designer: 0x23B, Part Number: 0xBE12, Version: 0x0.
ID-Code read correctly at 5.00 MHz.
Time for Connect: 0.517 s.
>|
---#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 2 0xC0FFEE00 0xDEADBEEF
RDP level 2 password is set into target device but RDP level is not 2. Skipped procedure.
Time for RDP Password Set, Unlock and Remove: 0.001 s.
>|
---#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 1 0xDEADBEEF 0xC0FFEE00
RDP level 1 password is set into target device but RDP level is not 1. Skipped procedure.
Time for RDP Password Set, Unlock and Remove: 0.002 s.
>|
---#TPCMD RDP_PASSWORD_REMOVE 1
Try to remove the device RDP level 1 password.
 * RDP level 1 password removed correctly.
Time for RDP Password Remove: 0.077 s.
>|
---#TPCMD RDP_PASSWORD_REMOVE 2
Try to remove the device RDP level 2 password.
 * RDP level 2 password removed correctly.
Time for RDP Password Remove: 0.077 s.
>|
---#TPCMD RDP_PASSWORD_STORE_LOCK 1 0xDEADBEEF 0xC0FFEE00
Try to set the device RDP level 1 password.
 * RDP level 1 password set correctly.
Try to change Readout Protection level to 1.
 * Readout Protection level set correctly to 1.
Time for RDP Password Store and Lock: 0.156 s.
>|
---#TPCMD RDP_PASSWORD_STORE_LOCK 2 0xC0FFEE00 0xDEADBEEF
Try to set the device RDP level 2 password.
 * RDP level 2 password set correctly.
Try to change Readout Protection level to 2.
 * Readout Protection level set correctly to 2.
Time for RDP Password Store and Lock: 0.144 s.
>|
---#TPCMD DISCONNECT
>|
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 1.00 MHz.
Trying Hot Plug connect procedure.
IDCODE: 0x0BE12477.
Designer: 0x23B, Part Number: 0xBE12, Version: 0x0.
ID-Code read correctly at 1.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x14770015, Type: AMBA AHB5 bus.
AP[0] ROM table base address 0xE00FE000.
Your device could be locked by RDP level 2 with password.
If you want to unlock it, try to use the following command:
 * RDP_PASSWORD_SET_UNLOCK to unlock the device and move the RDP level to 1.
Time for Connect: 0.606 s.
>|
---#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 2 0xC0FFEE00 0xDEADBEEF
Try to set RDP level 2 password to unlock the device.
 * Set RDP level 2 password correctly to unlock device.
Power off and on the device now!
 * Reconnect to the device now.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
Try to change Readout Protection level to 1.
 * Readout Protection level set correctly to 1.
To remove the RDP level 2 password you need to set the device RDP level to 0 or 0.5.
Time for RDP Password Set, Unlock and Remove: 0.811 s.
>|
---#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 1 0xDEADBEEF 0xC0FFEE00
Try to set RDP level 1 password to unlock the device.
 * Set RDP level 1 password correctly to unlock device.
Try to change Readout Protection level to 0.
 * Readout Protection level set correctly to 0.
Try to remove the device RDP level 1 password.
 * RDP level 1 password removed correctly.
Time for RDP Password Set, Unlock and Remove: 0.374 s.
>|
---#TPCMD RDP_PASSWORD_REMOVE 2
Try to remove the device RDP level 2 password.
 * RDP level 2 password removed correctly.
Time for RDP Password Remove: 0.077 s.
>|
---#TPCMD RDP_PASSWORD_REMOVE 1
RDP level 1 password is not set into target device. Skipped procedure.
Time for RDP Password Remove: 0.002 s.
>|
---#TPCMD DISCONNECT
>|
```

RDP regression with password using STM32WBA

Project commands:

```
#TPSTART
#TPCMD CONNECT
#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 2 0xC0FFEE00 0xDEADBEEF
#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 1 0xDEADBEEF 0xC0FFEE00
#TPCMD RDP_PASSWORD_REMOVE 1
#TPCMD RDP_PASSWORD_REMOVE 2
#TPCMD RDP_PASSWORD_STORE_LOCK 1 0xDEADBEEF 0xC0FFEE00
#TPCMD RDP_PASSWORD_STORE_LOCK 2 0xC0FFEE00 0xDEADBEEF
#TPCMD DISCONNECT
#TPCMD CONNECT
#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 2 0xC0FFEE00 0xDEADBEEF
#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 1 0xDEADBEEF 0xC0FFEE00
#TPCMD RDP_PASSWORD_REMOVE 2
#TPCMD RDP_PASSWORD_REMOVE 1
#TPCMD DISCONNECT
#TPEND
```

Real Time log:

```
---#TPSTART
Load SWD FPGA version 0x00001215.
>|
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
IDCODE: 0x0BE12477.
Designer: 0x23B, Part Number: 0xBE12, Version: 0x0.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x14770015, Type: AMBA AHB5 bus.
AP[1] IDR: 0x14770015, Type: AMBA AHB5 bus.
AP[1] ROM table base address 0xE00FE000.
CPUID: 0x410FD214.
Implementer Code: 0x41 - [ARM].
Found Cortex M33 revision r0p4.
Program counter value is 0xEFFFFFFE.
Cortex M33 Core halted [0.001 s].
Device configuration: [0x7FFFF8AA]
 * The device's Readout Protection level is 0 [0xAA].
 * OEM1 RDP key read is virgin. OEM1 RDP lock mechanism is not active.
 * OEM2 RDP key read is virgin. OEM2 RDP lock mechanism is not active.
 * BOR level 0. Reset level threshold is around 1.7 V.
 * No reset generated when entering the Stop mode.
 * No reset generated when entering the Standby mode.
 * SRAM1 not erased when a system reset occurs.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
 * Software independent watchdog selected.
 * Independent watchdog counter is running in Stop mode.
 * Independent watchdog counter is running in Standby mode.
 * Software window watchdog selected.
 * SRAM2 parity check disabled.
 * SRAM2 not erased when a system reset occurs.
 * BOOT0 taken from PH3/BOOT0 pin.
 * NBOOT0 option bit set to 1.
 * Global TrustZone security disabled.
PLL enabled using internal HSI oscillator.
Requested Clock is 37.50 MHz.
Generated Clock is 37.50 MHz.
Good samples: 3 [Range 6-8].
IDCODE: 0x0BE12477.
Designer: 0x23B, Part Number: 0xBE12, Version: 0x0.
ID-Code read correctly at 37.50 MHz.
Time for Connect: 0.106 s.
>|
---#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 2 0xC0FFEE00 0xDEADBEEF
RDP level 2 password is not set into target device. Skipped procedure.
Time for RDP Password Set, Unlock and Remove: 0.001 s.
>|
---#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 1 0xDEADBEEF 0xC0FFEE00
RDP level 1 password is not set into target device. Skipped procedure.
Time for RDP Password Set, Unlock and Remove: 0.001 s.
>|
---#TPCMD RDP_PASSWORD_REMOVE 1
RDP level 1 password is not set into target device. Skipped procedure.
Time for RDP Password Remove: 0.001 s.
>|
---#TPCMD RDP_PASSWORD_REMOVE 2
RDP level 2 password is not set into target device. Skipped procedure.
Time for RDP Password Remove: 0.001 s.
>|
---#TPCMD RDP_PASSWORD_STORE_LOCK 1 0xDEADBEEF 0xC0FFEE00
Try to set the device RDP level 1 password.
 * RDP level 1 password set correctly.
Try to change Readout Protection level to 1.
 * Readout Protection level set correctly to 1.
Time for RDP Password Store and Lock: 0.112 s.
>|
---#TPCMD RDP_PASSWORD_STORE_LOCK 2 0xC0FFEE00 0xDEADBEEF
Try to set the device RDP level 2 password.
 * RDP level 2 password set correctly.
Try to change Readout Protection level to 2.
 * Readout Protection level set correctly to 2.
Time for RDP Password Store and Lock: 0.111 s.
>|
---#TPCMD DISCONNECT
>|
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
IDCODE: 0x0BE12477.
Designer: 0x23B, Part Number: 0xBE12, Version: 0x0.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x14770015, Type: AMBA AHB5 bus.
AP[1] IDR: 0x14770015, Type: AMBA AHB5 bus.
AP[1] ROM table base address 0xE00FE000.
Your device could be locked by RDP level 2 with password.
If you want to unlock it, try to use the following command:
 * RDP_PASSWORD_SET_UNLOCK to unlock the device and move the RDP level to 1.
Time for Connect: 0.205 s.
>|
---#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 2 0xC0FFEE00 0xDEADBEEF
Try to set RDP level 2 password to unlock the device.
 * Set RDP level 2 password correctly to unlock device.
Power off and on the device now!
 * Reconnect to the device now.
Try to change Readout Protection level to 1.
 * Readout Protection level set correctly to 1.
To remove the RDP level 2 password you need to set the device RDP level to 0 or 0.5.
Time for RDP Password Set, Unlock and Remove: 0.382 s.
>|
---#TPCMD RDP_PASSWORD_SET_UNLOCK_REMOVE 1 0xDEADBEEF 0xC0FFEE00
Try to set RDP level 1 password to unlock the device.
 * Set RDP level 1 password correctly to unlock device.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
Try to change Readout Protection level to 0.
 * Readout Protection level set correctly to 0.
Try to remove the device RDP level 1 password.
 * RDP level 1 password removed correctly.
Time for RDP Password Set, Unlock and Remove: 0.221 s.
>|
---#TPCMD RDP_PASSWORD_REMOVE 2
Try to remove the device RDP level 2 password.
 * RDP level 2 password removed correctly.
Time for RDP Password Remove: 0.057 s.
>|
---#TPCMD RDP_PASSWORD_REMOVE 1
RDP level 1 password is not set into target device. Skipped procedure.
Time for RDP Password Remove: 0.001 s.
>|
---#TPCMD DISCONNECT
>|
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

STM32 Firmware Upgrade Service

Firmware Upgrade Services Introduction

This chapter describes the firmware upgrade services (FUS) available on STM32WB Series microcontrollers.

These services are provided by STMicroelectronics code located in a secure portion of the embedded Flash memory, and is used by any code running on Cortex®-M4 with a user Flash memory or through embedded bootloader commands (also running on Cortex®-M4).

Firmware Upgrade Services Definition

FUS (firmware upgrade services) is a firmware running on STM32WB Cortex®-M0+ and offering following features:

1. Install, upgrade or delete STM32WB Cortex®-M0+ wireless stack:
    a. Only encrypted and signed by STMicroelectronics
    b. Optionally, additionally double signed by customer if needed

2. FUS self-upgrade:
    a. Only encrypted and signed by STMicroelectronics
    b. Optionally, additionally double signed by customer if needed

3. Customer authentication key management
    a. Used for images double signature
    b. Install, update and lock the customer authentication key

4. User key management:
    a. Store customer keys
        i. Master key
        ii. Simple clear key
        iii. Encrypted key (by master key)
        iv. In secure area accessible only by Cortex®-M0+ code.
    b. Write stored key (simple or encrypted) into AES1 (advanced encryption standard) in secure mode (the Cortex®-M4 cannot access the key)
    c. Lock a stored key to prevent its usage until next system reset
    d. Unload a previously loaded key from AES to prevent its usage by other applications
    e. Key width: 128 or 256 bits
    f. Up to 100 user keys (encrypted by master key or clear) and one user master key

5. Communication with Cortex®-M4 (user code or bootloader):
    a. Through IPCC commands and response model (same as wireless stack model)
    b. Commands already supported by STM32WB bootloader (in ROM).

Firmware Upgrade Services Versions

Version	Description
V0.5.3	Default version programmed in production for all STM32WB5xx devices. Must be upgraded to V1.0.1 on STM32WB5xG devices or to V1.0.2 on STM32WB5xE/5xC devices. This version is not available for download on www.st.com and cannot be installed by users.
V1.0.1	First official release available on www.st.com and dedicated to STM32WB5xG devices only (1-MBytes Flash memory size) This version must not be installed on STM32WB5xE/5xC devices, otherwise the device enters a locked state and no further updates are possible.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

V1.0.2	First official release available on www.st.com and dedicated to STM32WB5xE/5xC devices (512-KBytes and 256-KBytes Flash memory size) Use the V1.0.2 on the STM32WB5xG devices if the devices present FUS V0.5.3. If an STM32WB5xG device has FUS V1.0.1, then there is no need to upgrade to V1.0.2, since it does not bring any new feature/change vs. V1.0.1. In case FUS V1.0.2 installation is started by user on an STM32WB5xG device with FUS V1.0.1, FUS returns FUS_STATE_IMG_NOT_AUTHENTIC error and discard the upgrade.
V1.1.0	FUS update to support following features: • Add FUS_ACTIVATE_ANTIROLLBACK command that allows activating Anti-rollback on wireless stack by user. User can activate this feature in order to prevent any installation of older wireless stack. • Replace Safe boot by V1.1.0 version (replace full chip lock by factory reset) • Add factory reset in case of Flash ECC, Flash corruption or Option Bytes corruption error. Factory reset means erase of wireless stack if present and reboot on FUS and full erase of other user sectors. FUS V1.1.0 can be installed only on devices containing V1.0.1 or V1.0.2 FUS. In case a device has V0.5.3 installed, user must first install V1.0.2 then install V1.1.0. When installing FUS V1.1.0 over an FUS V0.5.3 results in FUS_STATE_IMAGE_NOT_AUTHENTIC error and discarding the upgrade.
V1.1.1	FUS update to support STM32WB5xx 640KB sales-types. This version is not available on www.st.com and cannot be used for upgrade. This version is fully compatible with V1.1.0 and does not present any difference except management of new 640 KB sales type.
V1.1.2	FUS update to: • Optimize Flash usage: this allows the installation of a stack, maintaining one sector separation below a previously installed stack (instead of stack size space constraint explained in Section 2 Wireless stack image operations) • Security enhancements In order to upgrade from FUS V1.1.0 to FUS V1.1.2, the Anti-rollback must first be activated. Before activating Anti rollback, a wireless stack installed must be present. Upgrading from V1.1.0 to V1.2.0 is possible with no constraints and no addition operations from user.
V1.2.0	**FUS update to:** • **Includes V1.1.2 FUS updates in production** • **Allows direct update from FUS V1.1.0 to FUS V1.2.0 without activating the Anti-rollback.** • **Allows direct update from FUS V0.5.3 to FUS V1.2.0 (without installing intermediate FUS versions)** • **Security updates** **Upgrading from FUS V1.1.0 or any other FUS version, to FUS V1.2.0 is possible without constraints and no interaction from the user.**

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Firmware Upgrade Services Compatibility

Upgrade		To					
		V0.5.3	V1.0.2	V1.1.0	V1.1.1	V1.1.2	V1.2.0
From	V0.5.3	X	√	X	X	X	√
	V1.0.2	X	X	√	X	√	√
	V1.1.0	X	X	**X**	X	(1)	√
	V1.1.1	X	X	X	X	√	√
	V1.1.2	X	X	X	X	√	√
	V1.2.0	X	X	X	X	X	√

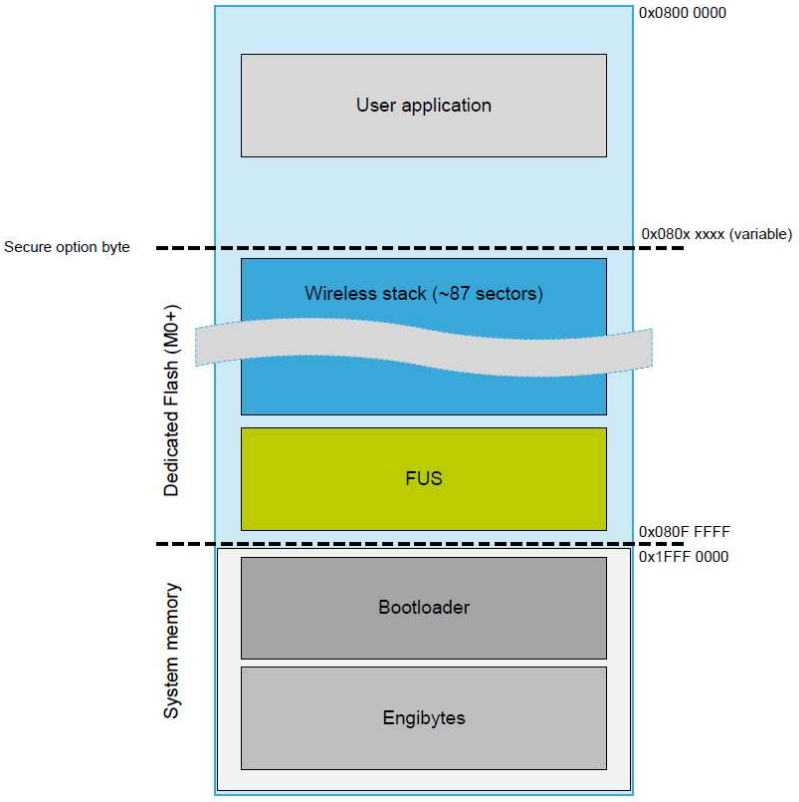
Legend:

- X: Cannot upgrade
- √: Upgradable
- **X**: Must not upgrade, otherwise encryption keys are lost
- (1): Upgradable but a BLE stack needs to be installed first and enable Anti-rollback

Firmware Upgrade Services Memory Mapping

The FUS has a dedicated space in Flash memory that depends on the FUS size. It also uses a dedicated space in SRAM2a and SRAM2b, and a shared space in SRAM2a (shared tables). The size of the dedicated space in Flash memory, SRAM2a and SRAM2b is defined by Option Bytes.
For more information, refer to the product reference manual.

The dedicated Flash memory and SRAM areas are shared with the wireless stack if it is installed. But at a given time, either FUS or wireless stack is running on Cortex®-M0+.



HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Firmware Upgrade Services FUS_Operator

FUS_Operator Overview

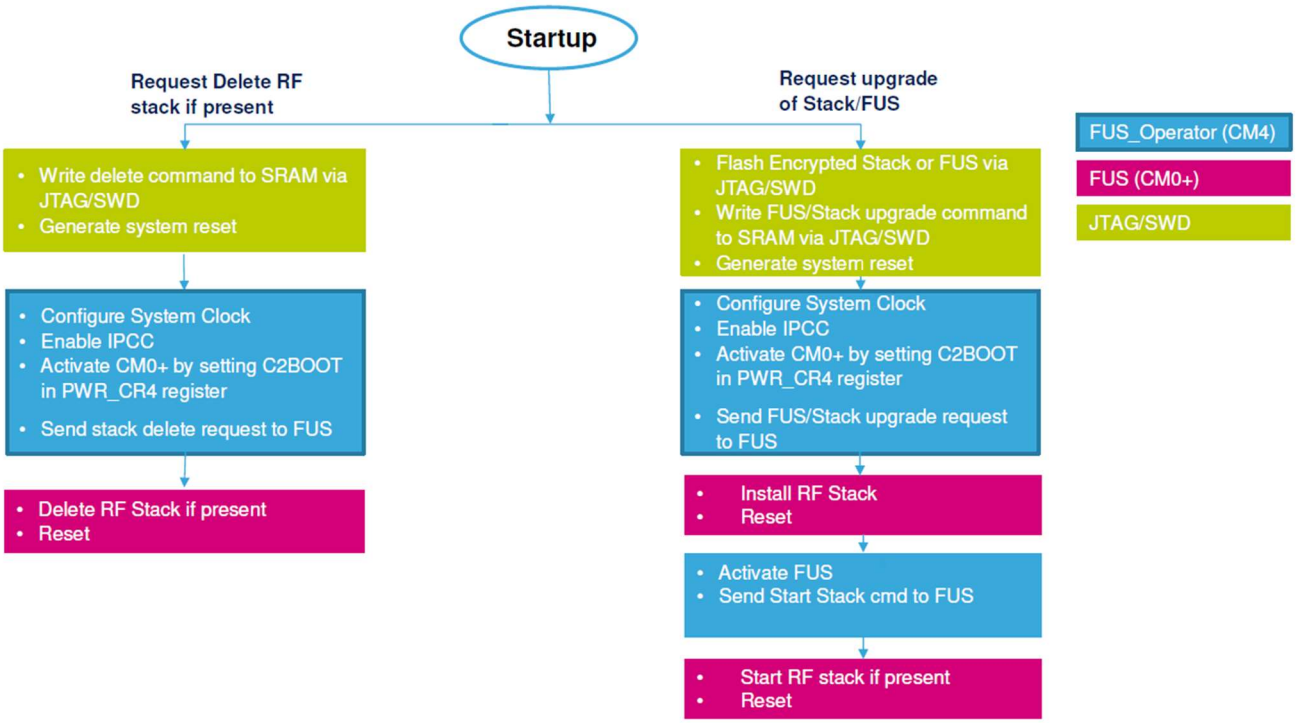
Let's start with an overview of FUS_Operator:

- The FUS_Operator is a firmware running on CM4
- Loaded into user main flash via SWD/JTAG interface
- Allow wireless stack and FUS upgrade without passing through bootloader
- FUS_Operator communicates with FUS through IPCC

FUS_Operator Version History

Version	Description
V2.2	Dedicated for boards equipped with external HSE crystal • Uses HSE as system clock source. • Frequency of HSE crystal is 32MHz • CM4 is running at 32Mhz and CM0+ at 32MHz
V2.8	Dedicated for boards with no HSE crystal • PLL used as system clock source • MSI is used as PLL clock source • CM4 is running at 64Mhz and CM0+ at 32MHz

FUS_Operator Working Model



HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

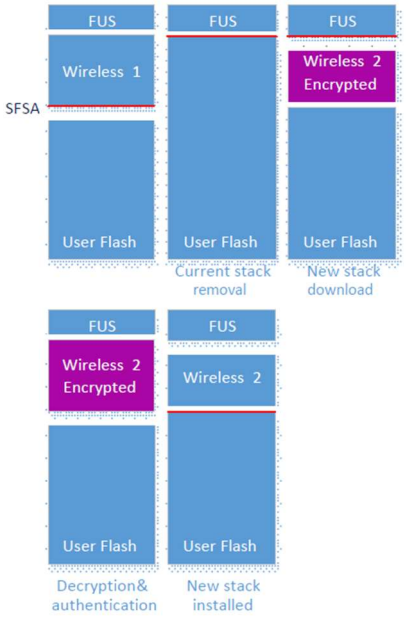
→smh-tech.com
info@smh-tech.com

Firmware Upgrade Services Wireless Stack Update

Update procedure when stack is not present

Update procedure:

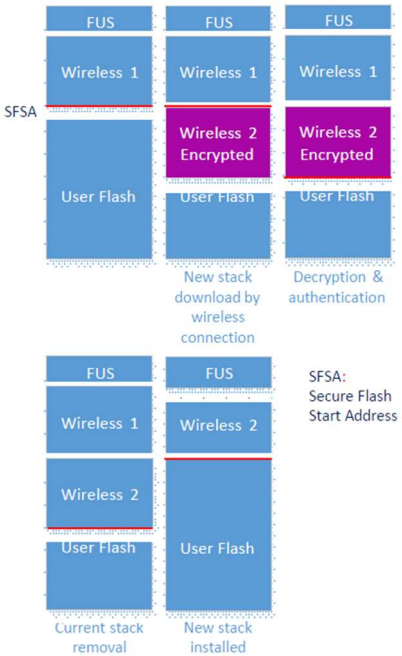
1. Current stack is removed by FUS command
    a. Space of current stack is freed for next download
    b. Wireless connection is no longer possible
2. Encrypted stack is downloaded in available user area
3. Protection is extended to wireless stack
4. Encrypted stack is decrypted and authenticated by FUS
5. Security option bytes are set by FUS according to new stack header
6. CPU2 ready to execute new wireless stack



Update procedure when stack is already present

Update procedure:

1. Download of new wireless stack requires
    a. A FUS_Operator in user flash
    b. Current stack active (wireless 1)
    c. Enough user memory available to store new stack (wireless 2)
2. Protection is extended to new stack
3. Encrypted stack is decrypted and authenticated by FUS
4. After authentication and integrity checks, current stack is removed and replaced by the new one
5. Security option bytes are set by FUS according to new stack header
6. CPU2 ready to execute new wireless stack



HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

SMH Technologies®

Firmware Upgrade Service FlashRunner 2 Commands

```
#TPCMD FUS_READ_INFORMATIONS
#TPCMD FUS_START
#TPCMD FUS_DELETE_FIRMWARE
#TPCMD FUS_UPDATE
#TPCMD FUS_START_WIRELESS_STACK
#TPCMD FUS_ANTI_ROLLBACK
```

Note: These commands are available from **libstm32.so** version **4.22**

```
#TPCMD FUS_CHECK_VERSION
```

Note: This command is available from **libstm32.so** version **5.19**

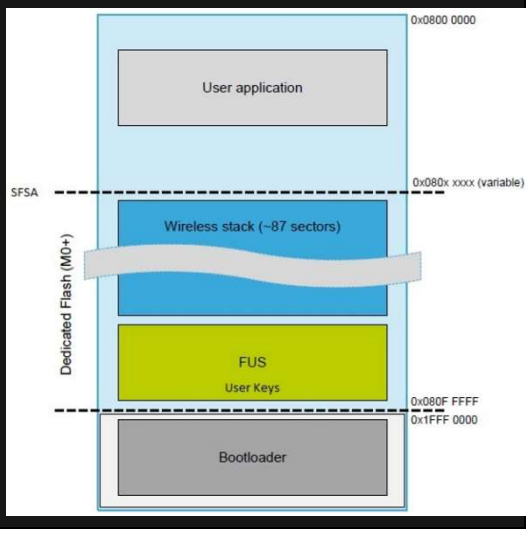
```
#TPCMD FUS_GET_VERSION
#TPCMD FUS_GET_WIRELESS_STACK_VERSION
#TPCMD FUS_CHECK_WIRELESS_STACK_VERSION
```

Note: These commands are available from **libstm32.so** version **5.29**

Software Update procedure with FlashRunner 2

If you connect to a virgin board, probably you can see from the log that the system and flash memory are protected by the address 0x080F4000 (SFSA value depends from target STM32WB device).

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x24770011, Type: AMBA AHB3 bus.
AP[1] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[0] ROM table base address 0xE00FF000.
CPUID: 0x410FC241.
Implementer Code: 0x41 - [ARM].
Found Cortex M4 revision r0p1.
Cortex M4 Core halted [0.020 s].
The device's RDP level is 0.
System security ESE is enabled.
CPU2 debug access is disabled.
System and Flash memory are secure from address 0x080F4000.
BOR Level 0. Reset level threshold is ~ 1.7 V.
PLL Enabled.
Requested Clock is 37.50 MHz.
Generated Clock is 37.50 MHz.
Good samples: 4 [Range 4-7].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 37.50 MHz.
Time for Connect: 0.126 s.
```



This value is obtained through the option bytes SFSA value.
For example, using the additional command `#TPCMD OVERVIEW_OPTION_BYTES` you can see in the real time log the value of the SFSA and you can calculate the protected System and Flash section.
Here the interesting part of the log taken by `#TPCMD OVERVIEW_OPTION_BYTES`

```
0x1FFF8070
  1111|1111|1111|1111|1111|1110|1111|0100
  Hex: 0xFFFFFEF4
```

Bits:	Name:	Value:
[12]	DDS	1
[08]	FSD	0
[07:00]	SFSA	F4

So SFSA is 0xF4, then you need to use this formula:

System + Flash memory secure = Flash Base Address + (SFSA x 0x1000).

So, you obtain: *0x08000000 + (0xF4 x 0x1000) =* ***0x080F4000***

Obviously, all of this is done automatically by the STM32 driver.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

#TPCMD FUS_GET_VERSION

Syntax: `#TPCMD FUS_GET_VERSION`

Prerequisites: FUS operator installed at base Flash address 0x08000000
Target STM32 Device ID know if FUS Operator is not already installed
Option Bytes correctly configured or Option Bytes in a standard configuration
If Flash memory is already programmed at address 0x08000000 but is not present the FUS Operator you need to perform a `#TPCMD MASSERASE F` command
FUS running (`#TPCMD FUS_START` executed)

Description: This command returns the version of the FUS loaded into the STM32 target device.
This command prints also into the Terminal.

Note: This command is available from **libstm32.so** version **5.29**

Examples: Command example:

Correct command execution: 😊

Real Time Log:

```
---#TPCMD FUS_GET_VERSION
FUS version loaded into device is v1.2.0.0.
Time for FUS Get Version: 0.234 s.
```

Terminal:

```
FUS v. 0x01020000
```

Skipped command execution: 😐

Real Time Log:

```
---#TPCMD FUS_GET_VERSION
FUS operator is not loaded.
Can't read the FUS version.
Time for FUS Get Version: 0.004 s.
```

Terminal:

```
FUS version unknown
```

#TPCMD FUS_GET_WIRELESS_STACK_VERSION

Syntax: `#TPCMD FUS_GET_ WIRELESS_STACK_VERSION`

Prerequisites: FUS operator installed at base Flash address 0x08000000
Target STM32 Device ID know if FUS Operator is not already installed
Wireless Stack installed
Option Bytes correctly configured or Option Bytes in a standard configuration
If Flash memory is already programmed at address 0x08000000 but is not present the FUS Operator you need to perform a `#TPCMD MASSERASE F` command
FUS running (`#TPCMD FUS_START` executed)

Description: This command returns the version of the Wireless Stack loaded into the STM32 target device.
This command prints also into the Terminal.

Note: This command is available from **libstm32.so** version **5.29**

Examples: Command example:

Correct command execution: 😊

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Real Time Log:

```
---TPCMD FUS_GET_WIRELESS_STACK_VERSION
Wireless Stack version loaded into device is v1.15.0.3.
Time for FUS Get Wireless Stack Version: 0.234 s.
```

Terminal:

```
Wireless Stack v. 0x010F0003
```

Skipped command execution: 😐

Real Time Log:

```
---#TPCMD FUS_GET_WIRELESS_STACK_VERSION
Wireless Stack is not installed.
Can't read the Wireless Stack version.
Time for FUS Get Wireless Stack Version: 0.234 s.
```

Terminal:

```
Wireless Stack version unknown
```

#TPCMD FUS_CHECK_VERSION

Syntax: `#TPCMD FUS_CHECK_VERSION <Operation Mode> <Version>`

<Operation Mode> Operation mode can be <, <=, ==, >= or >
< Version> FUS version separated by dots (i.e., 0.5.3, 1.2.0)

Prerequisites: FUS operator installed at base Flash address 0x08000000
Target STM32 Device ID know if FUS Operator is not already installed
Option Bytes correctly configured or Option Bytes in a standard configuration
If Flash memory is already programmed at address 0x08000000 but is not present the FUS Operator you need to perform a `#TPCMD MASSERASE F` command
FUS running (`#TPCMD FUS_START` executed)

Description: This command check if inserted FUS version is <, <=, ==, >= or > compared to the FUS version into the device

Note: This command is available from **libstm32.so** version **5.19**

Examples: Command example:

```
; ******** Start FUS using FUS Operator (Install FUS Operator into Flash Memory).
#TPCMD FUS_START

; ******** Load FUS FRB for upgrade from 0.5.3 to 1.2.0.
#TPSETSRC FUS_fw_for_fus_0_5_3_V1_2_0.frb

; ******** Check if FUS installed is 0.5.3 -> If yes, continue. If not, go to #THEN and load the FUS FRB for direct upgrade to 1.2.0
#IFERR TPCMD FUS_CHECK_VERSION == 0.5.3
; **** Load FUS FRB for upgrade from 0.5.3 to 1.2.0.
#THEN TPSETSRC FUS_fw_V1_2_0.frb

; ******** Check if FUS installed is 1.2.0 -> If yes, continue without Update the FUS. Otherwise, go to #THEN and update the FUS.
#IFERR TPCMD FUS_CHECK_VERSION >= 1.2.0
; **** Update the FUS
#THEN TPCMD FUS_UPDATE
; **** Re-Start the FUS using FUS Operator. [This is not mandatory but strongly suggested].
#THEN TPCMD FUS_START
```

Correct command execution: 😊

```
---#TPSETSRC FUS_fw_for_fus_0_5_3_V1_2_.frb
FRB CRC32 = 0xCEA0D597
>|
---#IFERR TPCMD FUS_CHECK_VERSION == 0.5.3
FUS version inserted is 0x00050300.
FUS version read from device is 0x01020000.
0800A64D!|
---#THEN TPSETSRC FUS_fw_V1_2_0.frb
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
FRB CRC32 = 0x5B1FF505
>|
---#IFERR TPCMD FUS_CHECK_VERSION >= 1.2.0
FUS version inserted is 0x01020000.
FUS version read from device is 0x01020000.
Time for FUS Check Version: 0.001 s
>|
```

Wrong command execution: ☹

```
---#IFERR TPCMD FUS_CHECK_VERSION == 0.5.3
FUS operator is not loaded.
Can't read the Wireless Stack version.
```

#TPCMD FUS_CHECK_WIRELESS_STACK_VERSION

Syntax: `#TPCMD FUS_CHECK_WIRELESS_STACK_VERSION <Operation Mode> <Version>`

<Operation Mode> Operation mode can be <, <=, ==, >= or >
< Version> FUS version separated by dots (i.e., 1.15.0.3)

Prerequisites: FUS operator installed at base Flash address 0x08000000
Target STM32 Device ID know if FUS Operator is not already installed
Option Bytes correctly configured or Option Bytes in a standard configuration
Wireless Stack not running
If Flash memory is already programmed at address 0x08000000 but is not present the FUS Operator you need to perform a `#TPCMD MASSERASE F` command
FUS running (`#TPCMD FUS_START` executed)

Description: This command check if inserted Wireless Stack version is <, <=, ==, >= or > compared to the Wireless Stack version into the device

Note: This command is available from **libstm32.so** version **5.29**

Examples: Command example:

```
; ********* Check if Wireless Stack installed is 1.15.0.3 -> If yes, continue.
;           If not, go to #THEN and load the Wireless Stack FRB for direct upgrade to 1.15.0.3
#IFERR TPCMD FUS_CHECK_WIRELESS_STACK_VERSION == 1.15.0.3

; ********* Delete Wireless Stack installed using FUS Operator.
#THEN TPCMD FUS_DELETE_FIRMWARE

; ********* Load BLE_Stack FRB version 1.15.0.3 to upgrade Wireless Stack.
#THEN TPSETSRC BLE_Stack.frb

; ********* Update the Wireless Stack using FUS Operator.
#THEN TPCMD FUS_UPDATE
```

Correct command execution: ☺

```
---#IFERR TPCMD FUS_CHECK_WIRELESS_STACK_VERSION == 1.15.0.3
Wireless Stack is not installed.
Can't read the Wireless Stack version.
0800A6A5!|
---#THEN TPCMD FUS_DELETE_FIRMWARE
Wireless Stack is not installed.
No need to delete Stack.
Time for FUS Delete Firmware: 0.004 s.
>|
---#THEN TPSETSRC BLE_Stack.frb
FRB CRC32 = 0x76A6BD89
>|
---#THEN TPCMD FUS_UPDATE
```

Wrong command execution: ☹

```
---#IFERR TPCMD FUS_CHECK_WIRELESS_STACK_VERSION == 1.15.0.3
Wireless Stack is running.
Can't read the Wireless Stack version.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale
Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255
D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021
→smh-tech.com
info@smh-tech.com

```
0800A6A5!|
---#THEN TPCMD FUS_DELETE_FIRMWARE
0800A6CB!|
```

#TPCMD FUS_READ_INFORMATIONS

Syntax: `#TPCMD FUS_READ_INFORMATIONS`

Prerequisites: None

Description: This command shows all relevant information about FUS, Wireless stack and FUS operator. FUS operator is needed to see the FUS and Wireless Stack versions loaded into the device.

Note: This command is available from **libstm32.so** version **4.22**
From **libstm32.so** version **5.18** this command prints into both Real Time Log and Terminal

Examples: Correct command execution with no FUS_Operator but with Wireless Stack installed: 😊

```
---#TPCMD FUS_READ_INFORMATIONS
FUS Reading informations.
Reading FUS status. [0xFEFF0600].
 * FUS is running.
 * Wireless stack is installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
Reading FUS version.
 * FUS version: v1.2.0.0.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v1.15.0.3.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Read Informations: 0.119 s.
```

Correct command execution with older FUS version (need to be updated): 😊

```
---#TPCMD FUS_READ_INFORMATIONS
FUS Reading informations.
 * FUS version: v1.1.1.0.
 * FUS must be updated using #TPCMD FUS_UPDATE before download stack.
Reading FUS status. [0x00000100]
 * FUS is running.
 * Wireless stack is not installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: IDLE
 * FUS Error: no error.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Read Informations: 0.002 s
```

Correct command execution with latest FUS version: 😊

```
---#TPCMD FUS_READ_INFORMATIONS
FUS Reading informations.
 * FUS version: v1.2.0.0.
Reading FUS status. [0x00000100]
 * FUS is running.
 * Wireless stack is not installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: IDLE
 * FUS Error: no error.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

#TPCMD FUS_START

Syntax: `#TPCMD FUS_START`

Prerequisites: FUS operator installed at base Flash address 0x08000000
Target STM32 Device ID know if FUS Operator is not already installed
Option Bytes correctly configured or Option Bytes in a standard configuration
If Flash memory is already programmed at address 0x08000000 but is not present the FUS_Operator you need to perform a `#TPCMD MASSERASE F` command

Description: This command starts the FUS and install the FUS_Operator if it's missing

Note: This command is available from **libstm32.so** version **4.22**

Examples: Correct command execution: 😊

If FUS Operator is not installed:

First of all, you need to know the Device ID.
To get the device ID just execute the command `#TPCMD GET_DEVICE_INFORMATIONS` or `#TPCMD GET_DEVICE_ID`.

```
---#TPCMD GET_DEVICE_INFORMATIONS
X and Y Wafer: 0x002E0041.
Wafer number: 0x0C.
Lot number: 0x2035374D575650.
Flash Size: 0x0100.
Package Type: 0x0A.
Device ID: 0x495.
Revision ID: 0x2001 - Rev A
Time for Get Device Information: 0.002 s
```

```
---#TPCMD GET_DEVICE_ID
Device ID: 0x495.
Time for Get Device ID: 0.001 s
```

In this case you need to create an FRB using *0x495_FUS_Operator.bin* source file and you need to set it at the base address of Flash memory (0x08000000).

STMicroelectronics provides three files named 0xXXX_FUS_Operator.bin differ in the initial value (0x494, 0x495 and 0x496) which is the Device ID value.

Here an example using 0x495_FUS_Operator.bin to create the correct FRB file.

Inside the STM32WBx reference manual you can find the relation between Revision ID and STM32WB series:

Revision ID	STM32WB series
0x494	STM32WB10CC
0x494	STM32WB15CC
0x495	STM32WB50CG/30CE
0x495	STM32WB55xx/35xx

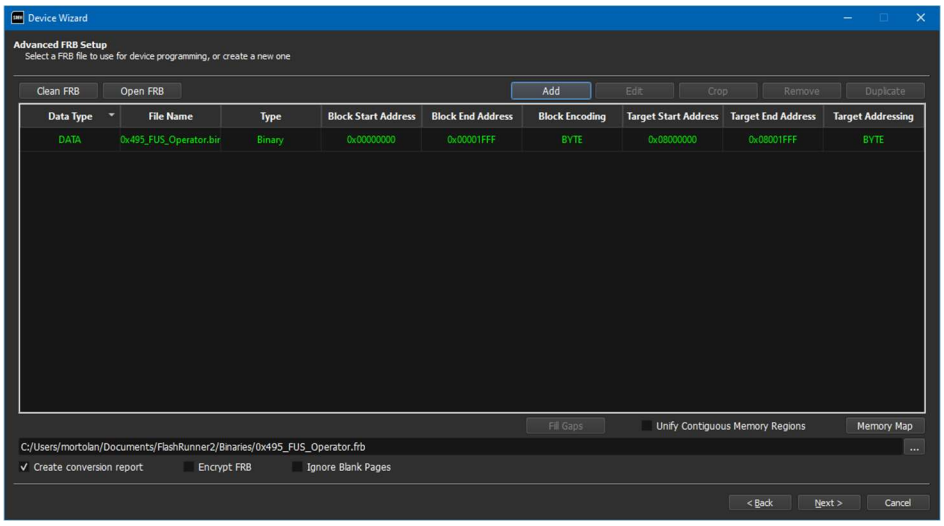
To create the FRB, open Advanced FRB Manager tool into Workbench and click on New FRB > Add > Binary and then choose from your local path the file 0x495_FUS_Operator.bin.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com



Remember to put the 0x495_FUS_Operator.bin file at Flash address 0x08000000.
Now you can use this FRB file to program the FUS_Operator.

Now you can see the standard execution starting from an empty device:

```
---#TPCMD FUS_START
Starting Firmware Service Routine Start.
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 1.
 * Option Bytes configuration isn't correct.
 * Cortex M4 Core must boot from Flash Memory.
 * Updating Option Bytes.
Option Bytes configuration is correct.
Clearing FUS status variables.
Missing FUS Operator firmware into Flash memory.
Try to open FUS Operator firmware FRB file.
FRB Headers collected.
Perform erase from address 0x08000000 to 0x08001FFF.
Erasing Flash pages from 0 to 1 [4KB each].
> Flash memory sectors erased [0.045 s].
Perform program from address 0x08000000 to 0x08001FFF.
> FUS Operator programmed into Flash memory [0.113 s].
Perform verify from address 0x08000000 to 0x08001FFF.
> FUS Operator verified from Flash memory [0.006 s].
Send FUS start command.
Generate a System Reset.
Started FUS start procedure. Multiple resets can occur.
Do not power off the board during start process!
> Operation is completed [0.127 s].
FUS Reading information.
 * FUS version: v1.1.1.0.
 * FUS must be updated using #TPCMD FUS_UPDATE before download stack.
Reading FUS status. [0x00000500]
 * FUS is running.
 * Wireless stack is installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: IDLE
 * FUS Error: no error.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v1.11.0.1.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Start: 0.526 s
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

#TPCMD FUS_DELETE_FIRMWARE

Syntax: `#TPCMD FUS_DELETE_FIRMWARE`

Prerequisites: FUS operator installed at base Flash address 0x08000000
Target STM32 Device ID know if FUS Operator is not already installed
Option Bytes correctly configured or Option Bytes in a standard configuration
If Flash memory is already programmed at address 0x08000000 but is not present the FUS Operator you need to perform a `#TPCMD MASSERASE F` command
FUS running (`#TPCMD FUS_START` executed)

Description: This command removes an installed Wireless Stack If there is not a Wireless Stack already installed, this command will do nothing.

Note: This command is available from **libstm32.so** version **4.22**

Examples: Correct command execution: 😊

Now you can see the standard execution from a device with FUS Operator installed and with FUS running:

```
---#TPCMD FUS_DELETE_FIRMWARE
Starting FUS Delete Firmware (Delete Stack).
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS delete stack command.
Generate a System Reset.
Started FUS Delete Firmware procedure. Multiple resets can occur.
Do not power off the board during erase process!
Operation is currently ongoing...
> Operation is completed [5.115 s].
FUS Reading informations.
 * FUS version: v1.1.1.0.
 * FUS must be updated using #TPCMD FUS_UPDATE before download stack.
Reading FUS status. [0x00000102]
 * FUS is running.
 * Wireless stack is not installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: IDLE
 * FUS Error: no error.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Delete Firmware: 5.118 s
```

Wrong command execution: ☹

```
---#TPCMD FUS_DELETE_FIRMWARE
Starting FUS Delete Firmware (Delete Stack).
FUS (Firmware Service Routine) is not running.
Please perform the #TPCMD FUS_START command before using this command.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

#TPCMD FUS_UPDATE (Update the FUS)

Syntax: `#TPCMD FUS_UPDATE`

Prerequisites: FUS operator installed at base Flash address 0x08000000
Target STM32 Device ID know if FUS Operator is not already installed
Option Bytes correctly configured or Option Bytes in a standard configuration
If Flash memory is already programmed at address 0x08000000 but is not present the FUS_Operator you need to perform a `#TPCMD MASSERASE F` command
FUS running (`#TPCMD FUS_START` executed)

Description: This command updates the FUS installed into the device
This command checks if the source file is valid

Note: This command is available from **libstm32.so** version **4.22**

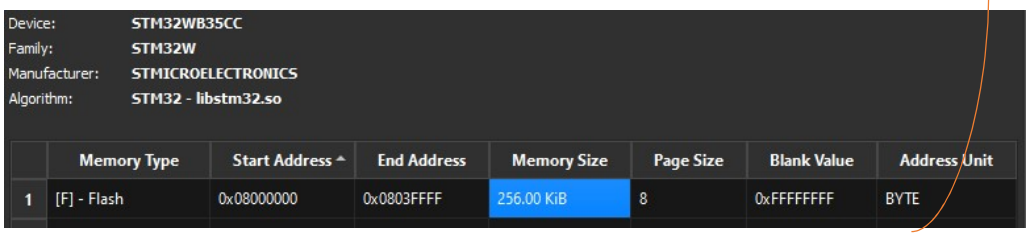
Examples: Correct command execution: 😊

To create properly the correct FRB for FUS update, you need to follow this table

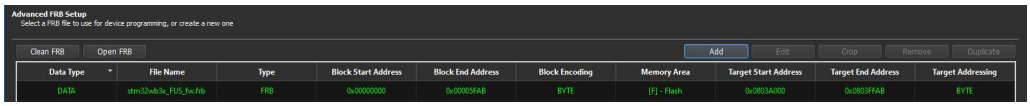
Device (Flash Size)	stm32wb3x_FUS_fw_for_fus_0_5_3.bin V1.2.0	stm32wb3x_FUS_fw.bin V1.2.0
STM32WB3x(256K)	0x0803A000	0x0803A000
STM32WB3x(512K)	0x0807A000	0x0807A000
STM32WB5xxC(256K)	0x0803A000	0x0803A000
STM32WB5xxE(512K)	0x0807A000	0x0807A000
STM32WB5xxY(640K)	0x0809A000	0x0809A000
STM32WB5xxG(1M)	0x080EC000	0x080EC000

(Please ask to STMicroelectronics if there are some updates of this table)

Here an example using an STM32WB3xx(256K). Flash size is 256 KB.



Select ***stm32wb3x_FUS_fw.bin V1.2.0*** and move this binary at start address **0x0803A000**.



Create now your FRB file and add it into your project.

Now you can see the standard execution from a device with FUS_Operator installed and running:

UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

```
---#TPCMD FUS_UPDATE
Starting Firmware Service Routine Update.
> Installed FUS version is: v1.1.1.0.
Starting FUS Delete Firmware (Delete Stack).
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS delete stack command.
Generate a System Reset.
Started FUS Delete Firmware procedure. Multiple resets can occur.
Do not power off the board during erase process!
Operation is currently ongoing...
> Operation is completed [0.231 s].
FRB CRC32 check passed.
FRB Headers collected.
Analysing FUS Upgrade FRB file:
 * File start address: 0x0803A000, size: 0x00005FAC.
 * SFSA value is 0xF4: 0x080F4000.
FRB File is correct for FUS update procedure.
Clearing FUS status variables.
Perform erase from address 0x0803A000 to 0x0803FFFF.
Erasing Flash pages from 58 to 63 [4KB each].
> Flash memory sectors erased [0.134 s].
Perform program from address 0x0803A000 to 0x0803FFFF.
> FUS Update programmed into Flash memory [0.294 s].
Perform verify from address 0x0803A000 to 0x0803FFFF.
> FUS Update verified from Flash memory [0.013 s].
Send FUS upgrade stack command.
Generate a System Reset.
Started FUS upgrade procedure. Multiple resets can occur.
Do not power off the board during update process!
Operation is currently ongoing...
> Operation is completed [3.187 s].
FUS Reading informations.
 * FUS version: v1.2.0.0.
Reading FUS status. [0x00000101]
 * FUS is running.
 * Wireless stack is not installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * FW upgrade have been requested.
 * FUS State: IDLE
 * FUS Error: no error.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Update: 3.892 s
```

Wrong command execution: ☹

```
---#TPCMD FUS_UPDATE
Starting Firmware Service Routine Update.
FUS (Firmware Service Routine) is not running.
Please perform the #TPCMD FUS_START command before using this command.
```

Wrong command execution: ☹

```
---#TPCMD FUS_UPDATE
Starting Firmware Service Routine Update.
> Installed FUS version is: v1.2.0.0.
The source file is not a FUS or a WIRELESS STACK file. Please, check the file.
0800A6DE!|
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

#TPCMD FUS_UPDATE (Update the Wireless Stack)

Syntax: `#TPCMD FUS_UPDATE`

Prerequisites: FUS operator installed at base Flash address 0x08000000
Target STM32 Device ID know if FUS Operator is not already installed
Option Bytes correctly configured or Option Bytes in a standard configuration
If Flash memory is already programmed at address 0x08000000 but is not present the FUS_Operator you need to perform a `#TPCMD MASSERASE F` command
FUS running (`#TPCMD FUS_START` executed)

Description: This command updates the FUS installed into the device
This command checks if the source file is valid

Note: This command is available from **libstm32.so** version **4.22**

Examples: Correct command execution: 😊

To create properly the correct FRB for Wireless Stack update, you need to follow this table:

Wireless Coprocessor Binary	STM32WB3x(512K)	STM32WB3x(256K)	Version
stm32wb3x_BLE_HCILayer_extended_fw.bin	0x08068000	0x08028000	V1.13.3
stm32wb3x_BLE_HCILayer_fw.bin	0x08068000	0x08028000	V1.13.3
stm32wb3x_BLE_HCI_AdvScan_fw.bin	0x08077000	0x08037000	V1.13.3
stm32wb3x_BLE_LLD_fw.bin	0x08078000	0x08038000	V1.12.0
stm32wb3x_BLE_Mac_802_15_4_fw.bin	0x08040000	Na	V1.13.3
stm32wb3x_BLE_Stack_basic_fw.bin	0x0805C000	0x0801C000	V1.13.3
stm32wb3x_BLE_Stack_full_extended_fw.bin	0x08053000	0x08013000	V1.13.3
stm32wb3x_BLE_Stack_full_fw.bin	0x0805C000	0x0801C000	V1.13.3
stm32wb3x_BLE_Stack_light_fw.bin	0x08063000	0x08023000	V1.13.3
stm32wb3x_Mac_802_15_4_fw.bin	0x0806F000	0x0802F000	V1.13.0
stm32wb3x_Phy_802_15_4_fw.bin	0x0806A000	0x0802A000	V1.13.0
stm32wb3x_Thread_FTD_fw.bin	0x08023000	NA	V1.13.0
stm32wb3x_Thread_MTD_fw.bin	0x08036000	NA	V1.13.0
stm32wb3x_Thread_RCP_fw.bin	0x08066000	0x08026000	V1.13.0
stm32wb3x_Zigbee_FFD_fw.bin	0x08030000	NA	V1.13.2
stm32wb3x_Zigbee_RFD_fw.bin	0x0803F000	NA	V1.13.2
stm32wb3x_Safeboot_fw.bin	0x0807E000	0x0803E000	V2.00.0

(Please ask to STMicroelectronics if there are some updates of this table V1.13.3 / 17-Mar-2022)

Wireless Coprocessor Binary	STM32WB5 xxG(1M)	STM32WB 5xxY(640k)	STM32WB 5xxE(512K)	STM32WB 5xxC(256K)	Version
stm32wb5x_BLE_HCILayer_extended_fw.bin	0x080DC000	0x08088000	0x08068000	0x08028000	V1.13.3
stm32wb5x_BLE_HCILayer_fw.bin	0x080DC000	0x08088000	0x08068000	0x08028000	V1.13.3
stm32wb5x_BLE_HCI_AdvScan_fw.bin	0x080EB000	0x08097000	0x08077000	0x08037000	V1.13.3
stm32wb5x_BLE_LLD_fw.bin	0x080ED000	0x08099000	0x08079000	0x08039000	V1.12.0
stm32wb5x_BLE_Mac_802_15_4_fw.bin	0x080B1000	0x0805D000	0x0803D000	NA	V1.13.3
stm32wb5x_BLE_Stack_basic_fw.bin	0x080D0000	0x0807C000	0x0805C000	0x0801C000	V1.13.3
stm32wb5x_BLE_Stack_full_extended_fw.bin	0x080C7000	0x08073000	0x08053000	0x08013000	V1.13.3
stm32wb5x_BLE_Stack_full_fw.bin	0x080D0000	0x0807C000	0x0805C000	0x0801C000	V1.13.3
stm32wb5x_BLE_Stack_light_fw.bin	0x080D7000	0x08083000	0x08063000	0x08023000	V1.13.3
stm32wb5x_BLE_Thread_dynamic_fw.bin	0x0806D000	0x08019000	NA	NA	V1.13.1
stm32wb5x_BLE_Thread_static_fw.bin	0x0806F000	0x0801B000	NA	NA	V1.13.0
stm32wb5x_BLE_Zigbee_FFD_dynamic_fw.bin	0x08071000	0x0801D000	NA	NA	V1.13.3
stm32wb5x_BLE_Zigbee_FFD_static_fw.bin	0x08073000	0x0801F000	NA	NA	V1.13.2
stm32wb5x_BLE_Zigbee_RFD_dynamic_fw.bin	0x08080000	0x0802C000	0x0800C000	NA	V1.13.3
stm32wb5x_BLE_Zigbee_RFD_static_fw.bin	0x08081000	0x0802D000	0x0800D000	NA	V1.13.2
stm32wb5x_Mac_802_15_4_fw.bin	0x080E3000	0x0808F000	0x0806F000	0x0802F000	V1.13.0
stm32wb5x_Phy_802_15_4_fw.bin	0x080DE000	0x0808A000	0x0806A000	0x0802A000	V1.13.0
stm32wb5x_Thread_FTD_fw.bin	0x08097000	0x08043000	0x08023000	NA	V1.13.0
stm32wb5x_Thread_MTD_fw.bin	0x080AA000	0x08056000	0x08036000	NA	V1.13.0
stm32wb5x_Thread_RCP_fw.bin	0x080DA000	0x08086000	0x08066000	0x08026000	V1.13.0
stm32wb5x_Zigbee_FFD_fw.bin	0x080A4000	0x08050000	0x08030000	NA	V1.13.2
stm32wb5x_Zigbee_RFD_fw.bin	0x080B2000	0x0805E000	0x0803E000	NA	V1.13.2
stm32wb5x_Safeboot_fw.bin	0x080F0000	0x0809F000	0x0807F000	0x0803F000	V2.00.0

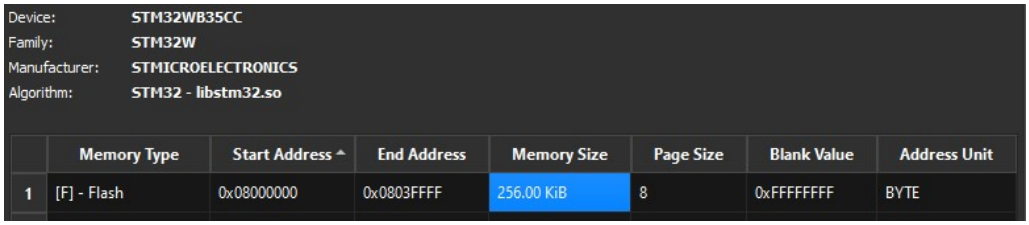
(Please ask to STMicroelectronics if there are some updates of this table V1.13.3 / 17-Mar-2022)

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale
Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255
D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021
→smh-tech.com
info@smh-tech.com

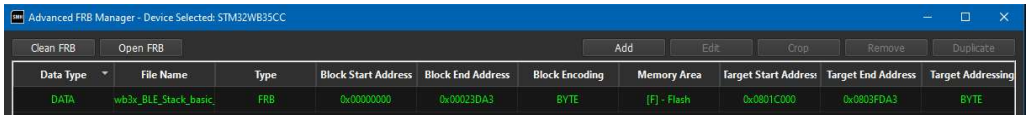
Wireless Coprocessor Binary	stm32wb1x(320K)	Version
stm32wb1x_BLE_HCILayer_fw.bin	0x0802F000	V1.13.0
stm32wb1x_BLE_HCI_AdvScan_fw.bin	0x0803E000	V1.13.0
stm32wb1x_BLE_LLD_fw.bin	0x0803F800	V1.12.0
stm32wb1x_BLE_Stack_basic_fw.bin	0x08023000	V1.13.0
stm32wb1x_BLE_Stack_full_fw.bin	0x08019000	V1.13.0
stm32wb1x_BLE_Stack_light_fw.bin	0x08029000	V1.13.0
stm32wb1x_Safeboot_fw.bin	0x08044000	V2.00.0

(Please ask to STMicroelectronics if there are some updates of this table V1.13.0 / 03-Nov-2021)

Here an example using an STM32WB3xx(256K). Flash size is 256 KB.



Select ***stm32wb3x_BLE_Stack_basic_fw.bin V1.13.3*** and move this binary at start address **0x0801C000** based on above table.



Now you can see the standard execution from a device with FUS_Operator installed and running:

```
---#TPCMD FUS_UPDATE
Starting Firmware Service Routine Update.
> Installed FUS version is: v1.1.1.0.
Starting FUS Delete Firmware (Delete Stack).
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS delete stack command.
Generate a System Reset.
Started FUS Delete Firmware procedure. Multiple resets can occur.
Do not power off the board during erase process!
Operation is currently ongoing...
> Operation is completed [0.231 s].
FRB CRC32 check passed.
FRB Headers collected.
Analysing FUS Upgrade FRB file:
 * File start address: 0x0801C000, size: 0x00023DA4.
 * SFSA value is 0xF4: 0x080F4000.
FRB File is correct for FUS update procedure.
Clearing FUS status variables.
Perform erase from address 0x0801C000 to 0x0803FFFF.
Erasing Flash pages from 23 to 63 [4KB each].
> Flash memory sectors erased [0.796 s].
Perform program from address 0x0801C000 to 0x0803FFFF.
> FUS Update programmed into Flash memory [1.608 s].
Perform verify from address 0x0801C000 to 0x0803FFFF.
> FUS Update verified from Flash memory [0.062 s].
Send FUS upgrade stack command.
Generate a System Reset.
Started FUS upgrade procedure. Multiple resets can occur.
Do not power off the board during update process!
Operation is currently ongoing...
> Operation is completed [5.646 s].
FUS Reading informations.
 * FUS version: v1.2.0.0.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

```
Reading FUS status. [0x00000101]
 * FUS is running.
 * Wireless stack is installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * FW upgrade have been requested.
 * FUS State: IDLE
 * FUS Error: no error.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Update: 6.643 s
```

Wrong command execution: ☹

```
---#TPCMD FUS_UPDATE
Starting Firmware Service Routine Update.
FUS (Firmware Service Routine) is not running.
Please perform the #TPCMD FUS_START command before using this command.
```

Wrong command execution: ☹

```
---#TPCMD FUS_UPDATE
Starting Firmware Service Routine Update.
> Installed FUS version is: v1.2.0.0.
The source file is not a FUS or a WIRELESS STACK file. Please, check the file.
0800A6DE!|
```

#TPCMD FUS_START_WIRELESS_STACK

Syntax: `#TPCMD FUS_START_WIRELESS_STACK`

Prerequisites: FUS operator installed at base Flash address 0x08000000
Option Bytes correctly configured or Option Bytes in a standard configuration
FUS running (`#TPCMD FUS_START` executed)
Stack installed.

Description: This command starts an installed Wireless stack

Note: This command is available from **libstm32.so** version **4.22**

Examples: Correct command execution: ☺

Now you can see the standard execution from a device with FUS_Operator installed and with FUS running:

```
---#TPCMD FUS_START_WIRELESS_STACK
Starting Firmware Service Routine Start Wireless Stack.
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS start Wireless Stack command.
Generate a System Reset.
Started FUS Start procedure. Multiple resets can occur.
Do not power off the board during erase process!
Operation is currently ongoing...
> Operation is completed [0.345 s].
FUS Reading informations.
 * FUS version: v1.2.0.
Reading FUS status. [0x00000102]
 * FUS is running.
 * Wireless stack is installed.
 * Stack BLE is running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: IDLE
 * FUS Error: no error.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Delete Firmware: 5.118 s
```

Wrong command execution (FUS not running): ☹

```
---#TPCMD FUS_START_WIRELESS_STACK
Starting FUS Delete Firmware (Delete Stack).
FUS (Firmware Service Routine) is not running.
Please perform the #TPCMD FUS_START command before using this command.
```

Wrong command execution (No Wireless Stack installed): ☹

```
---#TPCMD FUS_START_WIRELESS_STACK
Starting Firmware Service Routine Start Wireless Stack.
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS start Wireless Stack command.
Generate a System Reset.
Started FUS Start procedure. Multiple resets can occur.
Do not power off the board during erase process!
0800A5B3!|
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Firmware Upgrade Services Examples

1 - FUS STM32WB55RE example Commands

Here you can see a complete example of FUS procedure from **STM32 driver version 5.29**.

```
#TPCMD CONNECT
#TPCMD GET_DEVICE_INFORMATIONS

; ********* Masserase Flash only if Flash Memory is not blank.
#IFERR TPCMD BLANKCHECK F
#THEN TPCMD MASSERASE F
#THEN TPCMD BLANKCHECK F

; ---------------------------------------- FUS Operator ----------------------------------------------------
---

; ********* Load FUS Operator FRB file.
#TPSETSRC 0x495_FUS_Operator.frb
; ********* Start FUS using FUS Operator (Install FUS Operator into Flash Memory).
#TPCMD FUS_START

; ********* Get FUS informations from the device [This is not mandatory].
#TPCMD FUS_GET_VERSION
#TPCMD FUS_GET_WIRELESS_STACK_VERSION
#TPCMD FUS_READ_INFORMATIONS

; ---------------------------------------- FUS -------------------------------------------------------------
---

; ********* Load FUS FRB for upgrade from 0.5.3 to 1.2.0.
#TPSETSRC FUS_fw_for_fus_0_5_3_V1_2_0.frb
; ********* Check if FUS installed is 0.5.3 -> If yes, continue. If not, go to #THEN and load the FUS FRB for
direct upgrade to 1.2.0.
#IFERR TPCMD FUS_CHECK_VERSION == 0.5.3
; **** Load FUS FRB for upgrade from 0.5.3 to 1.2.0.
#THEN TPSETSRC FUS_fw_V1_2_0.frb
; ********* Check if FUS installed is 1.2.0 -> If yes, continue without Update the FUS. Otherwise, go to #THEN
and update the FUS.
#IFERR TPCMD FUS_CHECK_VERSION >= 1.2.0
; **** Update the FUS.
#THEN TPCMD FUS_UPDATE
; **** Re-Start the FUS using FUS Operator. [This is not mandatory but strongly suggested].
#THEN TPCMD FUS_START

; ---------------------------------------- Wireless Stack --------------------------------------------------
---

; ********* Check if Wireless Stack installed is >= 1.15.0.3 -> If yes, continue. If not, go to #THEN and load
the FRB for the upgrade.
#IFERR TPCMD FUS_CHECK_WIRELESS_STACK_VERSION >= 1.15.0.3
; ********* Delete Wireless Stack installed using FUS Operator.
#THEN TPCMD FUS_DELETE_FIRMWARE
; ********* Load BLE_Stack FRB version 1.15.0.3 to upgrade Wireless Stack.
#THEN TPSETSRC BLE_Stack.frb
; ********* Update the Wireless Stack using FUS Operator.
#THEN TPCMD FUS_UPDATE
; ********* Re-Start the FUS using FUS Operator. [This is not mandatory but strongly suggested].
#THEN TPCMD FUS_START
; ********* Start the Wireless Stack using FUS Operator. [This is not mandatory but strongly suggested].
#THEN TPCMD FUS_START_WIRELESS_STACK

; ---------------------------------------- Flash and Option Bytes ------------------------------------------
---

; ********* Load Flash FRB Firmware.
#TPSETSRC Flash.frb

; ********* Erase Flash Memory used by FUS Operator and program Flash FRB file.
#TPCMD ERASE F 0x08000000 0x4000
#TPCMD BLANKCHECK F
#TPCMD PROGRAM F
#TPCMD VERIFY F R
#TPCMD VERIFY F S

; ********* Program Option Bytes.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
#TPCMD PROGRAM O
#TPCMD VERIFY O R

#TPCMD DISCONNECT
```

1 - FUS STM32WB55RE example Real Time Log

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x24770011, Type: AMBA AHB3 bus.
AP[1] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[0] ROM table base address 0xE00FF000.
CPUID: 0x410FC241.
Implementer Code: 0x41 - [ARM].
Found Cortex M4 revision r0p1.
Cortex M4 Core halted [0.002 s].
CPU2 debug access is disabled.
System and Flash memory are secure from address 0x080F4000.
BOR level 0. Reset level threshold is ~ 1.7 V.
Device configuration: [0x39FFF1AA]
 * The device's Readout Protection level is 0 [0xAA].
 * System Security ESE enabled.
 * BOR level 0. Reset level threshold is around 1.7 V.
 * No reset generated when entering the Stop mode.
 * No reset generated when entering the Standby mode.
 * No reset generated when entering the Shutdown mode.
 * Software independent watchdog selected.
 * Independent watchdog counter is running in Stop mode.
 * Independent watchdog counter is running in Standby mode.
 * Software window watchdog selected.
 * SRAM2 parity check disabled.
 * SRAM2 and PKA RAM erased when a system reset occurs.
 * BOOT0 signal is taken from the option bit nBOOT0.
 * Radio automatic gain control trimming [0x1].
PLL enabled using internal HSI oscillator.
Requested Clock is 37.50 MHz.
Generated Clock is 37.50 MHz.
Good samples: 4 [Range 4-7].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 37.50 MHz.
Time for Connect: 0.127 s.
>|
---#TPCMD GET_DEVICE_INFORMATIONS
Unique ID: 203631553630501300510003F.
X and Y Wafer: 0x0051003F.
Wafer number: 0x13.
Lot number: 0x20363155363050.
Flash Size: 0x0200 - 512 KB.
Package Type: 0x13.
Device ID: 0x495.
Revision ID: 0x2003 - Rev X.
Time for Get Device Information: 0.002 s.
>|
---#IFERR TPCMD BLANKCHECK F
Flash memory is not blank.
0800A2E8!|
---#THEN TPCMD MASSERASE F
Readout Protection level is 0. Standard Masserase.
A Flash memory masserase by the CPU1 is ignored and a bus error is generated.
Performing a page erase of the entire non-secure Flash memory.
Erasing Flash pages from 0 to 127 [4KB each].
Time for Masserase F: 2.821 s.
>|
---#THEN TPCMD BLANKCHECK F
Time for Blankcheck F: 0.033 s.
>|
---#TPSETSRC 0x495_FUS_Operator.frb
FRB CRC32 = 0x847BE93D
>|
---#TPCMD FUS_START
Starting Firmware Service Routine Start.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
FUS operator is not loaded.
Try to open FUS Operator firmware FRB file.
Perform erase from address 0x08000000 to 0x08001FFF.
Erasing Flash pages from 0 to 1 [4KB each].
> Flash memory sectors erased [0.046 s].
Perform program from address 0x08000000 to 0x08001FFF.
> FUS Operator programmed into Flash memory [0.093 s].
Perform verify from address 0x08000000 to 0x08001FFF.
> FUS Operator verified from Flash memory [0.006 s].
Send FUS start command.
Generate a System Reset.
Started FUS start procedure. Multiple resets can occur.
Do not power off the board during start process!
> Operation is completed [0.001 s].
FUS Reading informations.
Reading FUS status. [0x00000100].
 * FUS is running.
 * Wireless stack is not installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: IDLE.
 * FUS Error: no error.
Reading FUS version.
 * FUS version: v1.2.0.0.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Start: 0.348 s.
>|
---#TPCMD FUS_GET_VERSION
FUS version loaded into device is v1.2.0.0.
Time for FUS Get Version: 0.008 s.
>|
---#TPCMD FUS_GET_WIRELESS_STACK_VERSION
Wireless Stack is not installed.
Can't read the Wireless Stack version.
Time for FUS Get Wireless Stack Version: 0.008 s.
>|
---#TPCMD FUS_READ_INFORMATIONS
FUS Reading informations.
Reading FUS status. [0x00000100].
 * FUS is running.
 * Wireless stack is not installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: IDLE.
 * FUS Error: no error.
Reading FUS version.
 * FUS version: v1.2.0.0.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Read Informations: 0.007 s.
>|
---#TPSETSRC FUS_fw_for_fus_0_5_3_V1_2_0.frb
FRB CRC32 = 0xFE0CFC68
>|
---#IFERR TPCMD FUS_CHECK_VERSION == 0.5.3
FUS version inserted is 0x00050300.
FUS version read from device is 0x01020000.
0800A679!|
---#THEN TPSETSRC FUS_fw_V1_2_0.frb
FRB CRC32 = 0x6BB3DCFA
>|
---#IFERR TPCMD FUS_CHECK_VERSION >= 1.2.0
FUS version inserted is 0x01020000.
FUS version read from device is 0x01020000.
Time for FUS Check Version: 0.008 s.
>|
---#IFERR TPCMD FUS_CHECK_WIRELESS_STACK_VERSION == 1.15.0.3
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
Wireless Stack is not installed.
Can't read the Wireless Stack version.
0800A6A5!|
---#THEN TPCMD FUS_DELETE_FIRMWARE
Wireless Stack is not installed.
No need to delete Stack.
Time for FUS Delete Firmware: 0.004 s.
>|
---#THEN TPSETSRC BLE_Stack.frb
FRB CRC32 = 0x76A6BD89
>|
---#THEN TPCMD FUS_UPDATE
Starting Firmware Service Routine Update.
> Installed FUS version is: v1.2.0.0.
Wireless Stack is not installed.
No need to delete Stack.
FRB Headers collected.
Analysing FUS Wireless Stack Upgrade FRB file:
 * Source file Wireless Stack version is v1.15.0.3.
 * File start address: 0x0805A000, size: 0x000250B4.
 * SFSA value is 0xF4: 0x080F4000.
FRB File is correct for FUS update procedure.
Clearing FUS status variables.
Perform erase from address 0x0805A000 to 0x0807FFFF.
Erasing Flash pages from 90 to 127 [4KB each].
> Flash memory sectors erased [0.838 s].
Perform program from address 0x0805A000 to 0x0807FFFF.
> FUS Update programmed into Flash memory [1.655 s].
Perform verify from address 0x0805A000 to 0x0807FFFF.
> FUS Update verified from Flash memory [0.064 s].
Send FUS upgrade stack command.
Generate a System Reset.
Started FUS upgrade procedure. Multiple resets can occur.
Do not power off the board during update process!
Operation is currently ongoing...
> Operation is completed [8.223 s].
FUS Reading informations.
Reading FUS status. [0xFEFF0601].
 * FUS is not running.
 * Wireless stack is installed.
 * Stack BLE is running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * FW upgrade have been requested.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Update: 10.931 s.
>|
---#THEN TPCMD FUS_START
Starting Firmware Service Routine Start.
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS start command.
Generate a System Reset.
Started FUS start procedure. Multiple resets can occur.
Do not power off the board during start process!
> Operation is completed [0.129 s].
FUS Reading informations.
Reading FUS status. [0x00000500].
 * FUS is running.
 * Wireless stack is installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: IDLE.
 * FUS Error: no error.
Reading FUS version.
 * FUS version: v1.2.0.0.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v1.15.0.3.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Start: 0.175 s.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
>|
---#THEN TPCMD FUS_START_WIRELESS_STACK
Starting Firmware Service Routine Start Wireless Stack.
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS start Wireless Stack command.
Generate a System Reset.
Started FUS Start Wireless Stack procedure. Multiple resets can occur.
Do not power off the board during start wireless stack process!
> Operation is completed [0.229 s].
FUS Reading informations.
Reading FUS status. [0xFEFF0600].
 * FUS is not running.
 * Wireless stack is installed.
 * Stack BLE is running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Start Wireless Stack: 0.282 s.
>|
---#TPSETSRC Flash.frb
FRB CRC32 = 0xC44BBB74
>|
---#TPCMD ERASE F 0x08000000 0x4000
Erasing Flash pages from 0 to 3 [4KB each].
Time for Erase F: 0.090 s.
>|
---#TPCMD BLANKCHECK F
Cannot read Flash memory beyond address 0x0805A000.
Perform blankcheck from address 0x08000000 to 0x08059FFF.
Time for Blankcheck F: 0.044 s.
>|
---#TPCMD PROGRAM F
FRB Headers collected.
Cannot write Flash memory beyond address 0x0805A000.
Perform program from address 0x08000000 to 0x08059FFF.
Time for Program F: 2.953 s.
>|
---#TPCMD VERIFY F R
Cannot verify Flash memory beyond address 0x0805A000.
Perform verify readout from address 0x08000000 to 0x08059FFF.
Time for Verify Readout F: 0.115 s.
>|
---#TPCMD VERIFY F S
Cannot verify Flash memory beyond address 0x0805A000.
Perform verify checksum 32bit from address 0x08000000 to 0x08059FFF.
Time for Verify Checksum 32bit F: 0.030 s.
>|
---#TPCMD PROGRAM O
Time for Program O: 0.078 s.
>|
---#TPCMD VERIFY O R
Time for Verify Readout O: 0.003 s.
>|
---#TPCMD DISCONNECT
>|
```

1 - FUS STM32WB55RE example Programming Times

Operation	Timings FlashRunner 2.0
Time for Connect	0.127 s
Time for Blankcheck Flash	0.005 s
Time for Conditional Masserase Flash	2.821s
Time for Conditional Blankcheck Flash	0.033 s

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

0x495_FUS_Operator.frb	-
Time for FUS Start	0.348 s
Time for FUS Check Version	0.001 s
Time for Delete Firmware	4.345 s
BLE_Stack.frb	-
Time for FUS Update	10.931 s
Time for FUS Start	0.175 s
Time for FUS Wireless Stack	0.282 s
Time for FUS Read Informations	0.003 s
Flash.frb (262KB)	-
Time for Erase F	0.090 s
Time for Blankcheck F	0.044 s
Time for Program F	2.953 s
Time for Verify F R	0.115 s
Time for Verify F S	0.030 s
Time for Program O	0.078 s
Time for Verify O	0.003 s
Cycle Time	**18.156s**

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

2 - FUS STM32WB15CC example Commands

Here you can see a complete example of FUS procedure from **STM32 driver version 5.25 (Legacy)**.

```
#TPCMD CONNECT

; ********* Masserase Flash only if Flash Memory is not blank.
#IFERR TPCMD BLANKCHECK F
#THEN TPCMD MASSERASE F
#THEN TPCMD BLANKCHECK F

; ********* Load FUS Operator FRB file.
#TPSETSRC 0x494_FUS_Operator.frb

; ********* Start FUS using FUS Operator (Install FUS Operator into Flash Memory).
#TPCMD FUS_START

; ********* Check if FUS installed is 1.2.2.
#TPCMD FUS_CHECK_VERSION == 1.2.2

; ********* Delete Wireless Stack installed using FUS Operator.
#TPCMD FUS_DELETE_FIRMWARE

; ********* Load BLE_Stack FRB version 1.2 to upgrade Wireless Stack.
#TPSETSRC stm32wb1x_BLE_Stack_full_extended_fw.frb

; ********* Update the Wireless Stack using FUS Operator.
#TPCMD FUS_UPDATE

; ********* Re-Start the FUS using FUS Operator. [This is not mandatory but strongly suggested].
#TPCMD FUS_START

; ********* Start the Wireless Stack using FUS Operator. [This is not mandatory but strongly suggested].
#TPCMD FUS_START_WIRELESS_STACK

; ********* Get device FUS status and print status into Real Time Log and Terminal. [This is not mandatory but strongly suggested].
#TPCMD FUS_READ_INFORMATIONS

#TPCMD DISCONNECT
```

2 - FUS STM32WB15CC example Real Time Log

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x24770011, Type: AMBA AHB3 bus.
AP[1] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[0] ROM table base address 0xE00FF000.
CPUID: 0x410FC241.
Implementer Code: 0x41 - [ARM].
Found Cortex M4 revision r0p1.
Cortex M4 Core halted [0.004 s].
CPU2 debug access is disabled.
System and Flash memory are secure from address 0x08018800.
BOR Level 0. Reset level threshold is ~ 1.7 V.
Device configuration: [0x3BFFF1AA]
 * The device's RDP level is 0 [0xAA].
 * System Security ESE enabled.
 * BOR Level 0. Reset level threshold is around 1.7 V.
 * No reset generated when entering the Stop mode.
 * No reset generated when entering the Standby mode.
 * No reset generated when entering the Shutdown mode.
 * Internal resets drives NRST pin low until it is seen as low level.
 * Software independent watchdog selected.
 * Independent watchdog counter is running in Stop mode.
 * Independent watchdog counter is running in Standby mode.
 * Software window watchdog selected.
 * Bidirectional reset: NRST pin configured in reset input/output mode (default mode).
 * SRAM2 parity check disabled.
 * SRAM2 and non-secure PKA RAM not erased when a system reset occurs.
 * BOOT0 signal is taken from the option bit nBOOT0.
 * Radio automatic gain control trimming [0x1].
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
PLL enabled using internal HSI oscillator.
Requested Clock is 37.50 MHz.
Generated Clock is 37.50 MHz.
Good samples: 3 [Range 5-7].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 37.50 MHz.
Time for Connect: 0.111 s.
>|
---#IFERR TPCMD BLANKCHECK F
Cannot read Flash memory beyond address 0x08018800.
Perform blankcheck from address 0x08000000 to 0x080187FF.
Flash memory is not blank.
0800A2E2!|
---#THEN TPCMD MASSERASE F
Readout Protection level is 0. Standard Masserase.
A Flash memory masserase by the CPU1 is ignored and a bus error is generated.
Performing a page erase of the entire non-secure Flash memory.
System and Flash memory are secure from address 0x08018800.
Erasing Flash pages from 0 to 48 [2KB each].
Time for Masserase F: 1.082 s.
>|
---#THEN TPCMD BLANKCHECK F
Cannot read Flash memory beyond address 0x08018800.
Perform blankcheck from address 0x08000000 to 0x080187FF.
Time for Blankcheck F: 0.007 s.
>|
---#TPSETSRC 0x494_FUS_Operator.frb
FRB CRC32 = 0x0100E200
>|
---#TPCMD FUS_START
Starting Firmware Service Routine Start.
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Missing FUS Operator firmware into Flash memory.
Try to open FUS Operator firmware FRB file.
FRB Headers collected.
Perform erase from address 0x08000000 to 0x08001FFF.
Erasing Flash pages from 0 to 3 [2KB each].
> Flash memory sectors erased [0.089 s].
Perform program from address 0x08000000 to 0x08001FFF.
> FUS Operator programmed into Flash memory [0.091 s].
Perform verify from address 0x08000000 to 0x08001FFF.
> FUS Operator verified from Flash memory [0.004 s].
Send FUS start command.
Generate a System Reset.
Started FUS start procedure. Multiple reset can occur.
Do not power off the board during start process!
> Operation is completed [0.130 s].
FUS Reading informations.
 * FUS version: v1.2.2.0.
Reading FUS status. [0x00000500].
 * FUS is running.
 * Wireless stack is installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: IDLE.
 * FUS Error: no error.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v1.17.1.1.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Start: 0.492 s
>|
---#TPCMD FUS_CHECK_VERSION == 1.2.2
FUS version inserted is 0x01020200.
FUS version read from device is 0x01020200.
Time for FUS Check Version: 0.001 s
>|
---#TPCMD FUS_DELETE_FIRMWARE
Starting FUS Delete Firmware (Delete Stack).
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS delete stack command.
Generate a System Reset.
Started FUS Delete Firmware procedure. Multiple reset can occur.
Do not power off the board during erase process!
Operation is currently ongoing...
> Operation is completed [4.326 s].
FUS Reading informations.
 * FUS version: v1.2.2.0.
Reading FUS status. [0x00000102].
 * FUS is running.
 * Wireless stack is not installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: IDLE.
 * FUS Error: no error.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Delete Firmware: 4.345 s
>|
---#TPSETSRC stm32wb1x_BLE_Stack_full_extended_fw.frb
FRB CRC32 = 0xC79D0797
>|
---#TPCMD FUS_UPDATE
Starting Firmware Service Routine Update.
> Installed FUS version is: v1.2.2.0.
Starting FUS Delete Firmware (Delete Stack).
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS delete stack command.
Generate a System Reset.
Started FUS Delete Firmware procedure. Multiple reset can occur.
Do not power off the board during erase process!
Operation is currently ongoing...
> Operation is completed [0.320 s].
FRB Headers collected.
Analysing FUS Upgrade FRB file:
 * File start address: 0x08018800, size: 0x0002D34C.
 * SFSA value is 0x8C: 0x08046000.
FRB File is correct for FUS update procedure.
Clearing FUS status variables.
Perform erase from address 0x08018800 to 0x08045FFF.
Erasing Flash pages from 49 to 139 [2KB each].
> Flash memory sectors erased [2.007 s].
Perform program from address 0x08018800 to 0x08045FFF.
> FUS Update programmed into Flash memory [1.994 s].
Perform verify from address 0x08018800 to 0x08045FFF.
> FUS Update verified from Flash memory [0.075 s].
Send FUS upgrade stack command.
Generate a System Reset.
Started FUS upgrade procedure. Multiple reset can occur.
Do not power off the board during update process!
Operation is currently ongoing...
> Operation is completed [20.527 s].
FUS Reading informations.
 * FUS version unknown.
Reading FUS status. [0xFEFF0601].
 * FUS is not running.
 * Wireless stack is installed.
 * Stack BLE is running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * FW upgrade have been requested.
 * FUS State: Error.
 * FUS Error: not running.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v4.28.0.91.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
Time for FUS Update: 25.118 s
>|
---#TPCMD FUS_START
Starting Firmware Service Routine Start.
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS start command.
Generate a System Reset.
Started FUS start procedure. Multiple reset can occur.
Do not power off the board during start process!
> Operation is completed [0.130 s].
FUS Reading informations.
 * FUS version: v1.2.2.0.
Reading FUS status. [0x00000500].
 * FUS is running.
 * Wireless stack is installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: IDLE.
 * FUS Error: no error.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v1.17.1.1.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Start: 0.148 s
>|
---#TPCMD FUS_START_WIRELESS_STACK
Starting Firmware Service Routine Start Wireless Stack.
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS start Wireless Stack command.
Generate a System Reset.
Started FUS Start procedure. Multiple reset can occur.
Do not power off the board during start wireless stack process!
> Operation is completed [0.320 s].
FUS Reading informations.
 * FUS version unknown.
Reading FUS status. [0xFEFF0600].
 * FUS is not running.
 * Wireless stack is installed.
 * Stack BLE is running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: Error.
 * FUS Error: not running.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v4.28.0.91.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Start Wireless Stack: 0.338 s
>|
---#TPCMD FUS_READ_INFORMATIONS
FUS Reading informations.
 * FUS version unknown.
Reading FUS status. [0xFEFF0600].
 * FUS is not running.
 * Wireless stack is installed.
 * Stack BLE is running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: Error.
 * FUS Error: not running.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v4.28.0.91.
Reading FUS operator version.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
 * FUS operator version: v3.1.0.
Time for FUS Read Informations: 0.003 s
>|
---#TPCMD DISCONNECT
>|
```

2 - FUS STM32WB15CC example Programming Times

Operation	Timings FlashRunner 2.0
Time for Connect	0.111 s
Time for Blankcheck Flash	0.005 s
Time for Conditional Masserase Flash	1.082 a
Time for Conditional Blankcheck Flash	0.007 s
0x494_FUS_Operator.frb	-
Time for FUS Start	0.492 s
Time for FUS Check Version	0.001 s
Time for Delete Firmware	4.345 s
stm32wb1x_BLE_Stack_full_extended_fw.frb	-
Time for FUS Update	25.118 s
Time for FUS Start	0.148 s
Time for FUS Wireless Stack	0.338 s
Time for FUS Read Informations	0.003 s
Cycle Time	**00:31.694 s**

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

3 - FUS STM32WB55CG example Commands

Here you can see a complete example of FUS procedure from **STM32 driver version 5.25 (Legacy)**.

```
#TPCMD CONNECT

; ********* Masserase Flash only if Flash Memory is not blank.
#IFERR TPCMD BLANKCHECK F
#THEN TPCMD MASSERASE F
#THEN TPCMD BLANKCHECK F

; ********* Load FUS Operator FRB file.
#TPSETSRC 0x495_FUS_Operator.frb

; ********* Start FUS using FUS Operator (Install FUS Operator into Flash Memory).
#TPCMD FUS_START

; ********* Load FUS FRB for upgrade from 0.5.3 to 1.2.0.
#TPSETSRC FUS_fw_for_fus_0_5_3_V1_2_0.frb

; ********* Check if FUS installed is 0.5.3 -> If yes, continue. If not, go to #THEN and load the FUS FRB for direct upgrade to
1.2.0
#IFERR TPCMD FUS_CHECK_VERSION == 0.5.3
; **** Load FUS FRB for upgrade from 0.5.3 to 1.2.0.
#THEN TPSETSRC FUS_fw_V1_2_0.frb

; ********* Check if FUS installed is 1.2.0 -> If yes, continue without Update the FUS. Otherwise, go to #THEN and update the FUS.
#IFERR TPCMD FUS_CHECK_VERSION >= 1.2.0
; **** Update the FUS
#THEN TPCMD FUS_UPDATE
; **** Re-Start the FUS using FUS Operator. [This is not mandatory but strongly suggested].
#THEN TPCMD FUS_START

; ********* Delete Wireless Stack installed using FUS Operator.
#TPCMD FUS_DELETE_FIRMWARE

; ********* Load BLE Stack FRB version 1.2 to upgrade Wireless Stack.
#TPSETSRC BLE_Stack.frb

; ********* Update the Wireless Stack using FUS Operator.
#TPCMD FUS_UPDATE

; ********* Re-Start the FUS using FUS Operator. [This is not mandatory but strongly suggested].
#TPCMD FUS_START

; ********* Start the Wireless Stack using FUS Operator. [This is not mandatory but strongly suggested].
#TPCMD FUS_START_WIRELESS_STACK

; ********* Get device FUS status and print status into Real Time Log and Terminal. [This is not mandatory but strongly suggested].
#TPCMD FUS_READ_INFORMATIONS

; ********** Load Flash FRB Firmware.
#TPSETSRC Flash.frb

; ********* Erase Flash Memory used by FUS Operator and program Flash FRB file.
#TPCMD ERASE F 0x08000000 0x2000
#TPCMD BLANKCHECK F
#TPCMD PROGRAM F
#TPCMD VERIFY F R
#TPCMD VERIFY F S

#TPCMD DISCONNECT
```

3 - FUS STM32WB55CG example Real Time Log

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x24770011, Type: AMBA AHB3 bus.
AP[1] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[0] ROM table base address 0xE00FF000.
CPUID: 0x410FC241.
Implementer Code: 0x41 - [ARM].
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
Found Cortex M4 revision r0p1.
Cortex M4 Core halted [0.004 s].
CPU2 debug access is disabled.
System and Flash memory are secure from address 0x080CF000.
BOR Level 0. Reset level threshold is ~ 1.7 V.
Device configuration: [0x39FFF1AA]
 * The device's RDP level is 0 [0xAA].
 * System Security ESE enabled.
 * BOR Level 0. Reset level threshold is around 1.7 V.
 * No reset generated when entering the Stop mode.
 * No reset generated when entering the Standby mode.
 * No reset generated when entering the Shutdown mode.
 * Software independent watchdog selected.
 * Independent watchdog counter is running in Stop mode.
 * Independent watchdog counter is running in Standby mode.
 * Software window watchdog selected.
 * SRAM2 parity check disabled.
 * SRAM2 and PKA RAM erased when a system reset occurs.
 * BOOT0 signal is taken from the option bit nBOOT0.
 * Radio automatic gain control trimming [0x1].
PLL enabled using internal HSI oscillator.
Requested Clock is 37.50 MHz.
Generated Clock is 37.50 MHz.
Good samples: 4 [Range 4-7].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 37.50 MHz.
Time for Connect: 0.111 s.
>|
---#IFERR TPCMD BLANKCHECK F
Cannot read Flash memory beyond address 0x080CF000.
Perform blankcheck from address 0x08000000 to 0x080CEFFF.
Time for Blankcheck F: 0.073 s.
>|
---#TPSETSRC 0x495_FUS_Operator.frb
FRB CRC32 = 0x847BE93D
>|
---#TPCMD FUS_START
Starting Firmware Service Routine Start.
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Missing FUS Operator firmware into Flash memory.
Try to open FUS Operator firmware FRB file.
FRB Headers collected.
Perform erase from address 0x08000000 to 0x08001FFF.
Erasing Flash pages from 0 to 1 [4KB each].
> Flash memory sectors erased [0.046 s].
Perform program from address 0x08000000 to 0x08001FFF.
> FUS Operator programmed into Flash memory [0.093 s].
Perform verify from address 0x08000000 to 0x08001FFF.
> FUS Operator verified from Flash memory [0.006 s].
Send FUS start command.
Generate a System Reset.
Started FUS start procedure. Multiple reset can occur.
Do not power off the board during start process!
> Operation is completed [0.131 s].
FUS Reading informations.
 * FUS version: v1.2.0.0.
Reading FUS status. [0x00000500].
 * FUS is running.
 * Wireless stack is installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: IDLE.
 * FUS Error: no error.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v1.14.0.5.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Start: 0.451 s
>|
---#TPSETSRC FUS_fw_for_fus_0_5_3_V1_2_0.frb
FRB CRC32 = 0xCEA0D597
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
>|
---#IFERR TPCMD FUS_CHECK_VERSION == 0.5.3
FUS version inserted is 0x00050300.
FUS version read from device is 0x01020000.
0800A64D!|
---#THEN TPSETSRC FUS_fw_V1_2_0.frb
FRB CRC32 = 0x5B1FF505
>|
---#IFERR TPCMD FUS_CHECK_VERSION >= 1.2.0
FUS version inserted is 0x01020000.
FUS version read from device is 0x01020000.
Time for FUS Check Version: 0.001 s
>|
---#TPCMD FUS_DELETE_FIRMWARE
Starting FUS Delete Firmware (Delete Stack).
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS delete stack command.
Generate a System Reset.
Started FUS Delete Firmware procedure. Multiple reset can occur.
Do not power off the board during erase process!
Operation is currently ongoing...
> Operation is completed [1.045 s].
FUS Reading informations.
 * FUS version: v1.2.0.0.
Reading FUS status. [0x00000102].
 * FUS is running.
 * Wireless stack is not installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: IDLE.
 * FUS Error: no error.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Delete Firmware: 1.062 s
>|
---#TPSETSRC BLE_Stack.frb
FRB CRC32 = 0xA88D2AA7
>|
---#TPCMD FUS_UPDATE
Starting Firmware Service Routine Update.
> Installed FUS version is: v1.2.0.0.
Starting FUS Delete Firmware (Delete Stack).
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS delete stack command.
Generate a System Reset.
Started FUS Delete Firmware procedure. Multiple reset can occur.
Do not power off the board during erase process!
Operation is currently ongoing...
> Operation is completed [0.232 s].
FRB Headers collected.
Analysing FUS Upgrade FRB file:
 * File start address: 0x080CF000, size: 0x00024A98.
 * SFSA value is 0xF4: 0x080F4000.
FRB File is correct for FUS update procedure.
Clearing FUS status variables.
Perform erase from address 0x080CF000 to 0x080F3FFF.
Erasing Flash pages from 207 to 243 [4KB each].
> Flash memory sectors erased [0.815 s].
Perform program from address 0x080CF000 to 0x080F3FFF.
> FUS Update programmed into Flash memory [1.637 s].
Perform verify from address 0x080CF000 to 0x080F3FFF.
> FUS Update verified from Flash memory [0.064 s].
Send FUS upgrade stack command.
Generate a System Reset.
Started FUS upgrade procedure. Multiple reset can occur.
Do not power off the board during update process!
Operation is currently ongoing...
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
> Operation is completed [8.017 s].
FUS Reading informations.
 * FUS version unknown.
Reading FUS status. [0xFEFF0601].
 * FUS is not running.
 * Wireless stack is installed.
 * Stack BLE is running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * FW upgrade have been requested.
 * FUS State: Error.
 * FUS Error: not running.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v17.22.0.37.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Update: 10.904 s
>|
---#TPCMD FUS_START
Starting Firmware Service Routine Start.
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS start command.
Generate a System Reset.
Started FUS start procedure. Multiple reset can occur.
Do not power off the board during start process!
> Operation is completed [0.131 s].
FUS Reading informations.
 * FUS version: v1.2.0.0.
Reading FUS status. [0x00000500].
 * FUS is running.
 * Wireless stack is installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: IDLE.
 * FUS Error: no error.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v1.14.0.5.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Start: 0.149 s
>|
---#TPCMD FUS_START_WIRELESS_STACK
Starting Firmware Service Routine Start Wireless Stack.
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS start Wireless Stack command.
Generate a System Reset.
Started FUS Start procedure. Multiple reset can occur.
Do not power off the board during start wireless stack process!
> Operation is completed [0.232 s].
FUS Reading informations.
 * FUS version unknown.
Reading FUS status. [0xFEFF0600].
 * FUS is not running.
 * Wireless stack is installed.
 * Stack BLE is running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: Error.
 * FUS Error: not running.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v17.22.0.37.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Start Wireless Stack: 0.250 s
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
>|
---#TPCMD FUS_READ_INFORMATIONS
FUS Reading informations.
 * FUS version unknown.
Reading FUS status. [0xFEFF0600].
 * FUS is not running.
 * Wireless stack is installed.
 * Stack BLE is running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: Error.
 * FUS Error: not running.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v17.22.0.37.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Read Informations: 0.003 s
>|
---#TPSETSRC Flash.frb
FRB CRC32 = 0xF4E7BD62
>|
---#TPCMD ERASE F 0x08000000 0x2000
Erasing Flash pages from 0 to 1 [4KB each].
Time for Erase F: 0.045 s.
>|
---#TPCMD BLANKCHECK F
Cannot read Flash memory beyond address 0x080CF000.
Perform blankcheck from address 0x08000000 to 0x080CEFFF.
Time for Blankcheck F: 0.073 s.
>|
---#TPCMD PROGRAM F
FRB Headers collected.
Cannot write Flash memory beyond address 0x080CF000.
Perform program from address 0x08000000 to 0x080CEFFF.
Time for Program F: 3.006 s.
>|
---#TPCMD VERIFY F R
Cannot verify Flash memory beyond address 0x080CF000.
Perform verify readout from address 0x08000000 to 0x080CEFFF.
Time for Verify Readout F: 0.116 s.
>|
---#TPCMD VERIFY F S
Cannot verify Flash memory beyond address 0x080CF000.
Perform verify checksum 32bit from address 0x08000000 to 0x080CEFFF.
Time for Verify Checksum 32bit F: 0.029 s.
>|
---#TPCMD DISCONNECT
>|
```

3 - FUS STM32WB55CG example Programming Times

Operation	Timings FlashRunner 2.0
Time for Connect	0.111 s
Time for Blankcheck Flash	0.073 s
Time for Conditional Masserase Flash	Not executed
Time for Conditional Blankcheck Flash	Not executed
0x495_FUS_Operator.frb	-
Time for FUS Start	0.451 s
FUS_fw_for_fus_0_5_3_V1_2_0.frb	-
Time for FUS Check Version	0.001 s

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

FUS_fw_V1_2_0.frb	-
Time for FUS Check Version	0.001 s
Time for Conditional FUS Update	Not executed
Time for Conditional FUS Start	Not executed
Time for Delete Firmware	1.062 s
BLE_stack.frb	-
Time for FUS Update	10.904 s
Time for FUS Start	0.149 s
Time for FUS Wireless Stack	0.250 s
Time for FUS Read Informations	0.002 s
Flash.frb	-
Time for Erase Flash 0x08000000 0x2000	0.045 s
Time for Blankcheck Flash	0.073 s
Time for Program Flash	3.006 s
Time for Verify Readout Flash	0.116 s
Time for Verify Checksum Flash	0.029 s
Cycle Time	**00:16.324 s**

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

4 - FUS STM32WB55CG example Commands

Here you can see a complete example of FUS procedure from **STM32 driver version 5.18 (Legacy)**.

```
#TPCMD CONNECT
#TPCMD MASSERASE F
#TPCMD BLANKCHECK F
#TPSETSRC 0x495_FUS_Operator.frb
#TPCMD FUS_START
#TPCMD FUS_DELETE_FIRMWARE
#TPSETSRC wireless_stack.frb
#TPCMD FUS_UPDATE
#TPCMD FUS_START
#TPCMD FUS_START_WIRELESS_STACK
#TPCMD FUS_READ_INFORMATIONS
#TPSETSRC flash_firmware.frb
#TPCMD MASSERASE F
#TPCMD BLANKCHECK F
#TPCMD PROGRAM F
#TPCMD VERIFY F R
#TPCMD VERIFY F S
```

4 - FUS STM32WB55CG example Real Time Log

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x24770011, Type: AMBA AHB3 bus.
AP[1] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[0] ROM table base address 0xE00FF000.
CPUID: 0x410FC241.
Implementer Code: 0x41 - [ARM].
Found Cortex M4 revision r0p1.
Cortex M4 Core halted [0.004 s].
CPU2 debug access is disabled.
System and Flash memory are secure from address 0x080CF000.
BOR Level 0. Reset level threshold is ~ 1.7 V.
Device configuration: [0x39FFF1AA]
 * The device's RDP level is 0 [0xAA].
 * System Security ESE enabled.
 * BOR Level 0. Reset level threshold is around 1.7 V.
 * No reset generated when entering the Stop mode.
 * No reset generated when entering the Standby mode.
 * No reset generated when entering the Shutdown mode.
 * Software independent watchdog selected.
 * Independent watchdog counter is running in Stop mode.
 * Independent watchdog counter is running in Standby mode.
 * Software window watchdog selected.
 * SRAM2 parity check disabled.
 * SRAM2 and PKA RAM erased when a system reset occurs.
 * BOOT0 signal is taken from the option bit nBOOT0.
 * Radio automatic gain control trimming [0x1].
PLL enabled using internal HSI oscillator.
Requested Clock is 37.50 MHz.
Generated Clock is 37.50 MHz.
Good samples: 4 [Range 4-7].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 37.50 MHz.
Time for Connect: 0.109 s.
>|
---#TPCMD MASSERASE F
Readout Protection level is 0. Standard Masserase.
A Flash memory masserase by the CPU1 is ignored and a bus error is generated.
Performing a page erase of the entire non-secure Flash memory.
System and Flash memory are secure from address 0x080CF000.
Erasing Flash pages from 0 to 206 [4KB each].
Time for Masserase F: 4.553 s.
>|
---#TPCMD BLANKCHECK F
Cannot read Flash memory beyond address 0x080CF000.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
Perform blankcheck from address 0x08000000 to 0x080CEFFF.
Time for Blankcheck F: 0.072 s.
>|
---#TPSETSRC 0x495_FUS_Operator.frb
FRB CRC32 = 0x847BE93D
>|
---#TPCMD FUS_START
Starting Firmware Service Routine Start.
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Missing FUS Operator firmware into Flash memory.
Try to open FUS Operator firmware FRB file.
FRB Headers collected.
Perform erase from address 0x08000000 to 0x08001FFF.
Erasing Flash pages from 0 to 1 [4KB each].
> Flash memory sectors erased [0.045 s].
Perform program from address 0x08000000 to 0x08001FFF.
> FUS Operator programmed into Flash memory [0.093 s].
Perform verify from address 0x08000000 to 0x08001FFF.
> FUS Operator verified from Flash memory [0.006 s].
Send FUS start command.
Generate a System Reset.
Started FUS start procedure. Multiple reset can occur.
Do not power off the board during start process!
> Operation is completed [0.131 s].
FUS Reading informations.
 * FUS version: v1.2.0.0.
Reading FUS status. [0x00000500]
 * FUS is running.
 * Wireless stack is installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: IDLE
 * FUS Error: no error.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v1.14.0.5.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Start: 0.437 s
>|
---#TPCMD FUS_DELETE_FIRMWARE
Starting FUS Delete Firmware (Delete Stack).
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS delete stack command.
Generate a System Reset.
Started FUS Delete Firmware procedure. Multiple reset can occur.
Do not power off the board during erase process!
Operation is currently ongoing...
> Operation is completed [1.044 s].
FUS Reading informations.
 * FUS version: v1.2.0.0.
Reading FUS status. [0x00000102]
 * FUS is running.
 * Wireless stack is not installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: IDLE
 * FUS Error: no error.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Delete Firmware: 1.050 s
>|
---#TPSETSRC wireless_stack.frb
FRB CRC32 = 0xA88D2AA7
>|
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
---#TPCMD FUS_UPDATE
Starting Firmware Service Routine Update.
> Installed FUS version is: v1.2.0.0.
Starting FUS Delete Firmware (Delete Stack).
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS delete stack command.
Generate a System Reset.
Started FUS Delete Firmware procedure. Multiple reset can occur.
Do not power off the board during erase process!
Operation is currently ongoing...
> Operation is completed [0.232 s].
FRB Headers collected.
Analysing FUS Upgrade FRB file:
 * File start address: 0x080CF000, size: 0x00024A98.
 * SFSA value is 0xF4: 0x080F4000.
FRB File is correct for FUS update procedure.
Clearing FUS status variables.
Perform erase from address 0x080CF000 to 0x080F3FFF.
Erasing Flash pages from 207 to 243 [4KB each].
> Flash memory sectors erased [0.815 s].
Perform program from address 0x080CF000 to 0x080F3FFF.
> FUS Update programmed into Flash memory [1.636 s].
Perform verify from address 0x080CF000 to 0x080F3FFF.
> FUS Update verified from Flash memory [0.063 s].
Send FUS upgrade stack command.
Generate a System Reset.
Started FUS upgrade procedure. Multiple reset can occur.
Do not power off the board during update process!
Operation is currently ongoing...
> Operation is completed [8.039 s].
FUS Reading informations.
 * FUS version unknown.
Reading FUS status. [0xFEFF0601]
 * FUS is not running.
 * Wireless stack is installed.
 * Stack BLE is running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * FW upgrade have been requested.
 * FUS State: Error.
 * FUS Error: not running.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v17.22.0.37.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Update: 10.900 s
>|
---#TPCMD FUS_START
Starting Firmware Service Routine Start.
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS start command.
Generate a System Reset.
Started FUS start procedure. Multiple reset can occur.
Do not power off the board during start process!
> Operation is completed [0.131 s].
FUS Reading informations.
 * FUS version: v1.2.0.0.
Reading FUS status. [0x00000500]
 * FUS is running.
 * Wireless stack is installed.
 * Stack BLE is not running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: IDLE
 * FUS Error: no error.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v1.14.0.5.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Start: 0.136 s
>|
---#TPCMD FUS_START_WIRELESS_STACK
Starting Firmware Service Routine Start Wireless Stack.
Checking Option Bytes.
 * Option Byte nBOOT0 value is 1.
 * Option Byte nBOOT1 value is 1.
 * Option Byte nSWBOOT0 value is 0.
Option Bytes configuration is correct.
Clearing FUS status variables.
Send FUS start Wireless Stack command.
Generate a System Reset.
Started FUS Start procedure. Multiple reset can occur.
Do not power off the board during start wireless stack process!
> Operation is completed [0.232 s].
FUS Reading informations.
 * FUS version unknown.
Reading FUS status. [0xFEFF0600]
 * FUS is not running.
 * Wireless stack is installed.
 * Stack BLE is running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: Error.
 * FUS Error: not running.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v17.22.0.37.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Start Wireless Stack: 0.238 s
>|
---#TPCMD FUS_READ_INFORMATIONS
FUS Reading informations.
 * FUS version unknown.
Reading FUS status. [0xFEFF0600]
 * FUS is not running.
 * Wireless stack is installed.
 * Stack BLE is running.
 * No FUS/Wireless Stack delete is ongoing.
 * No FUS/Wireless Stack upgrade is ongoing.
 * No FW erase has been requested.
 * No FW upgrade have been requested.
 * FUS State: Error.
 * FUS Error: not running.
Reading Wireless Stack version.
 * FUS Wireless Stack version: v17.22.0.37.
Reading FUS operator version.
 * FUS operator version: v3.1.0.
Time for FUS Read Informations: 0.002 s
>|
---#TPSETSRC flash_firmware.frb
FRB CRC32 = 0xF4E7BD62
>|
---#TPCMD MASSERASE F
Readout Protection level is 0. Standard Masserase.
A Flash memory masserase by the CPU1 is ignored and a bus error is generated.
Performing a page erase of the entire non-secure Flash memory.
System and Flash memory are secure from address 0x080CF000.
Erasing Flash pages from 0 to 206 [4KB each].
Time for Masserase F: 4.553 s.
>|
---#TPCMD BLANKCHECK F
Cannot read Flash memory beyond address 0x080CF000.
Perform blankcheck from address 0x08000000 to 0x080CEFFF.
Time for Blankcheck F: 0.073 s.
>|
---#TPCMD PROGRAM F
FRB Headers collected.
Cannot write Flash memory beyond address 0x080CF000.
Perform program from address 0x08000000 to 0x080CEFFF.
Time for Program F: 3.008 s.
>|
---#TPCMD VERIFY F R
Cannot verify Flash memory beyond address 0x080CF000.
Perform verify readout from address 0x08000000 to 0x080CEFFF.
Time for Verify Readout F: 0.117 s.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
>|
---#TPCMD VERIFY F S
Cannot verify Flash memory beyond address 0x080CF000.
Perform verify checksum 32bit from address 0x08000000 to 0x080CEFFF.
Time for Verify Checksum 32bit F: 0.029 s.
>|
---#TPCMD DISCONNECT
>|
```

4 - FUS STM32WB55CG example Programming Times

Operation	Timings FlashRunner 2.0
Time for Connect	0.109 s
Time for Masserase Flash	4.553 s
Time for Blankcheck Flash	0.072 s
0x495_FUS_Operator.frb	-
Time for FUS Start	0.437 s
Time for FUS Delete Firmware	1.050 s
wireless_stack.frb	-
Time for FUS Update	10.900 s
Time for FUS Start	0.136 s
Time for FUS Wireless Stack	0.238 s
Time for FUS Read Informations	0.002 s
flash_firmware.frb	-
Time for Masserase Flash	4.553 s
Time for Blankcheck Flash	0.073 s
Time for Program Flash	3.008 s
Time for Verify Readout Flash	0.117 s
Time for Verify Checksum Flash	0.029 s
Cycle Time	**00:25.323 s**

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

STM32 Debug Authentication

Debug Authentication is available only for STM32H5 Series Arm®-based 32-bit MCUs.

This chapter is divided into various subchapters listed below:

1. **Security features on STM32H5 MCUs**
   a. Secure Boot for STM32H5

2. **Product Lifecycle Introduction**
   a. Lifecycle Product State Open
   b. Lifecycle Product State Provisioning
   c. Lifecycle Product State iROT-Provisioned
   d. Lifecycle Product State TZ-Closed
   e. Lifecycle Product State Closed/Locked

3. **Debug Authentication Introduction**
   a. Debug Authentication Password
   b. Debug Authentication Certificate
   c. Debug Authentication Full Regression
   d. Debug Authentication Partial Regression

4. **Debug Authentication Virtual Memories**
   a. [P] - Provisioning Virtual Memory
   b. [R] - Regression Virtual Memory
   c. [M] – Secure Manager SFI Virtual Memory

5. **Debug Authentication Commands**
   a. #TPCMD DISCOVERY
   b. #TPCMD PROVISIONING
   c. #TPCMD REGRESSION
   d. #TPCMD SECURE_FIRMWARE_INSTALL

6. **Debug Authentication Flows**
   a. Debug Authentication flow for OTP based devices
   b. Debug Authentication flow for non-crypto devices
   c. Debug Authentication flow for crypto devices

7. **Debug Authentication Examples**
   a. Debug Authentication STM32H503xx OTP Password Hash
   b. Debug Authentication STM32H5xx non-crypto devices
   c. Debug Authentication STM32H5xx crypto devices
   d. Debug Authentication STM32H5xx crypto devices Secure Firmware Install

SMH Technologies®

Security features on STM32H5 MCUs

These chapters are based on ST STM32 Wiki, and it contains an overview of the security features available on STM32H5 MCUs. The table below contains detailed information based on the different product lines.

Security features embedded on:	STM32H503	STM32H523	STM32H533	STM32H563	STM32H573
Secure Boot and Firmware Update	YES	YES	YES	YES	YES
SBSFU legacy	NO	NO	NO	NO	NO
SBSFU by MCUboot	YES	YES	YES	YES	YES
STiRoT	NO	NO	YES	NO	YES
Isolation	YES	YES	YES	YES	YES
HDP	YES	YES	YES	YES	YES
TF-M	NO	NO	NO	YES	YES
Secure manager	NO	NO	NO	NO	YES
IP protection	YES	YES	YES	YES	YES
Secure provisioning	NO	NO	YES	NO	YES
Initial attestation	NO	NO	NO	NO	YES
SMAK	NO	NO	NO	NO	YES
SMDK	NO	NO	NO	NO	YES
Cryptography	YES	YES	YES	YES	YES
ST crypto lib	YES	YES	YES	YES	YES
Crypto libraries	YES	YES	YES	YES	YES
Silicon device life cycle	YES	YES	YES	YES	YES
Legacy RDP	NO	NO	NO	NO	NO
Product state	YES	YES	YES	YES	YES
Debug authentication	YES	YES	YES	YES	YES
Secure manufacturing	YES	YES	YES	YES	YES
SFI	NO	YES	YES	YES	YES
SFIx	NO	NO	NO	YES	YES
Provisioning	YES	YES	YES	YES	YES
Secure storage	NO	YES	YES	YES	YES

STM32H503 — Adapted for parts without TrustZone® support. SBSFU by MCU boot can also be used on STM32H503 product lines. However, there are some limitations compared to larger devices.

STM32H523 — SBSFU variant with TrustZone® support, based on the PSA TF-M MCU boot model. This is the tier of Secure Boot and field upgrade intended for the STM32H523 and STM32H563 product lines.
The Secure Boot starts in the user flash memory on a secure address range, starting from 0x0C000000, where it is assumed that there is an OEMiRoT or another chosen secure software.

STM32H533 — The crypto parts support the option to use STiRoT, which is an initial step in the secure boot solution with the Secure manager.

STM32H563

STM32H573

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Secure Boot for STM32H5

The secure boot is the first firmware executed after a reset and verifies the integrity of the user application(s) (check if not modified) before executing it.

It also ensures a secure firmware installation and firmware update.
The integrity and authentication (identity) of the user application image(s) are verified before installation.
It also activates the security mechanisms.

This is called the Root of Trust (RoT)

Secure Boot stages

The secure boot can be done in one or two stages.

The first stage is done by an immutable code (**iRoT**).
It can't be modified and must include only the mandatory services.

In the second stage (optional) (**uRoT**), more complex operations can be done.
It can be updated in case of future required improvements (e.g. attack robustness improvements).

First stage: Two Immutable Root of Trust (iRoT) are possible for the STM32H5.

- **STiRoT**: Immutable code, provided by ST and integrated into the system flash of the devices supporting embedded cryptographic accelerator.
- **OEMiRoT**: An OEM can develop his boot code including a similar feature to STiRoT. Flash write protection is used to ensure this code cannot be modified (immutable).

Second optional stage: Two updatable Root of Trust (uRoT) are possible for the STM32H5.

- **STuRoT**: installable service provided by ST, includes module installation/update capability
- **OEMuRoT**: developed by 3rd party OEM and installed through the embedded bootloader

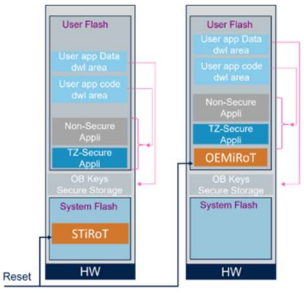
As mentioned above, the uRoT has the advantage to be updatable for later modifications or improvements

One secure boot stage:

Two possible single-stage boots are possible:

- STiRoT using the immutable firmware provided by ST and natively embedded in the device
- OEMiRoT immutable firmware developed and installed by an OEM.

Two possible use cases for STiRoT or OEMiRoT:



Case 1:
A new firmware image (encrypted and signed image containing the secure and the non-secure applications) or a new data image (encrypted and signed image) needs to be installed.

- The iRoT verifies the integrity and authentication of the image (placed in a user flash download area).
- For a user firmware: the image is decrypted and installed in the defined execution user flash area.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

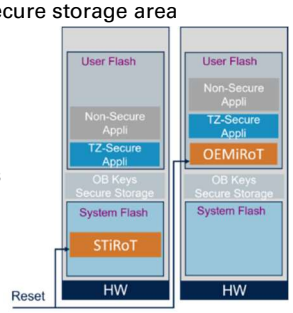
Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

- For user data, if applicable, the image is decrypted and stored after re-encryption in a secure storage area

Case 2:
A secure user application and if applicable non-secure user application is already installed in the user flash.



- The iRoT verifies the integrity before the user firmware execution, so it ensures that this firmware has not been modified.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

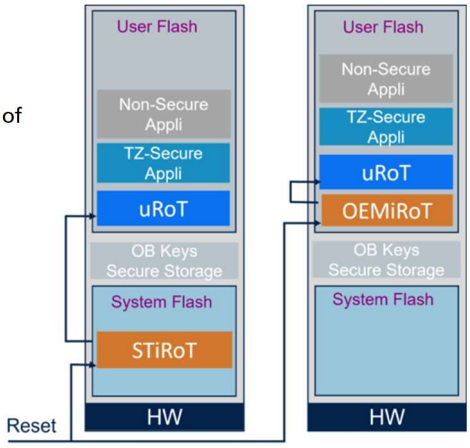
→smh-tech.com
info@smh-tech.com

Two secure boot stages:

For two secure boot stages first the iRoT is executed followed by the uRoT.

Role of the iRoT (STiRoT or OEMiRoT):

- The iRoT ensures the secure installation of the uRoT through verification of the integrity and authenticity of the uRoT firmware.
- The iRoT verifies the integrity of the uRoT before executing the uRoT firmware.
- The iRoT is only insuring the secure execution of the next boot stage (uRoT)



Role of the uRoT (STuRoT or OEMuRoT)

- The uRoT verifies the integrity and authentication of the user application firmware images (secure and non-secure) and the user data image (if applicable) before installing it.
- The uRoT verifies the integrity of the user application firmware (secure and non-secure) before allowing the execution.

The different possible boot paths

For STM32H5 family we have different flow paths:

The STM32H57x / STM32H53x is supporting TrustZone® and full hardware cryptography, so all boot paths are possible with this device.

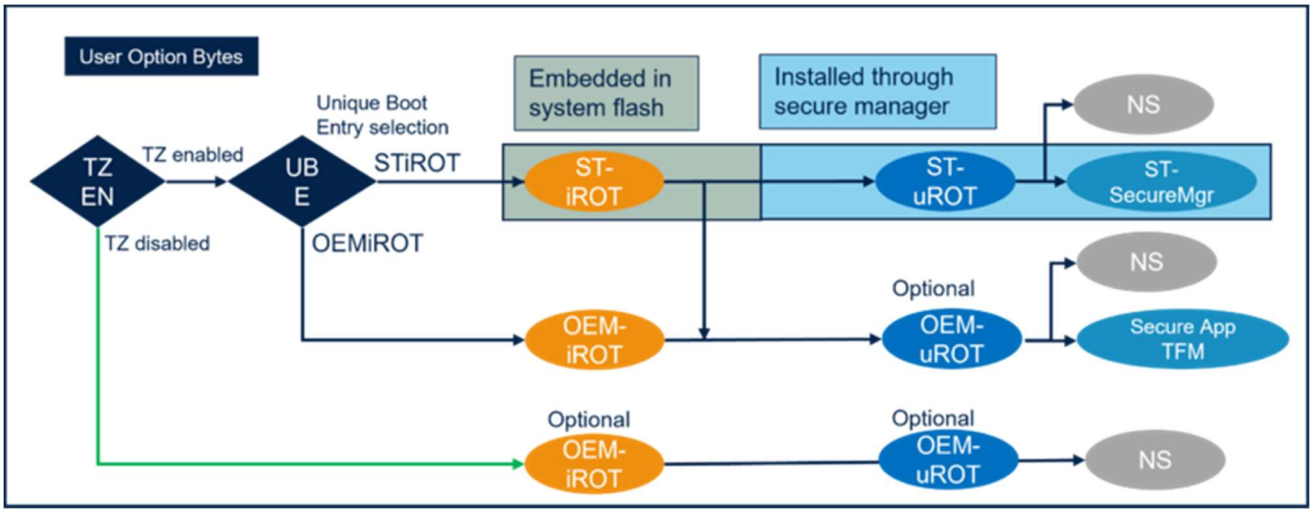
The STM32H56x* / STM32H52x* is supporting TrustZone® but limited hardware cryptography (hash + PKA verif).
The STiRoT (ST immutable Root of Trust), and the secure manager are not supported.

The STM32H503 is not supporting TrustZone®, and has limited hardware cryptography (hash) limiting to a single boot.

* The STM32H56x does not support the full PKA (Public Key Accelerator), but only the embedded Verification Algorithm (DSA and ECDSA signature verification).

The boot path is selected through option bytes programming (TZEN and UBE) as show in the next figure.
When the boot path is selected through FlashRunner 2.0 command, the related option bytes are programmed during the provisioning procedure.
This procedure is done automatically.

The user does not need to take care of which option byte needs to be programmed.



HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

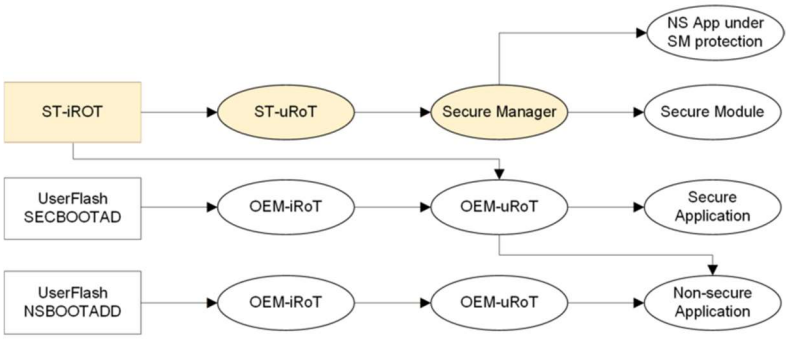
→smh-tech.com
info@smh-tech.com

Boot paths for STM32H53x and STM32H57x

With TrustZone® enabled, the device starts the Secure Boot procedure.

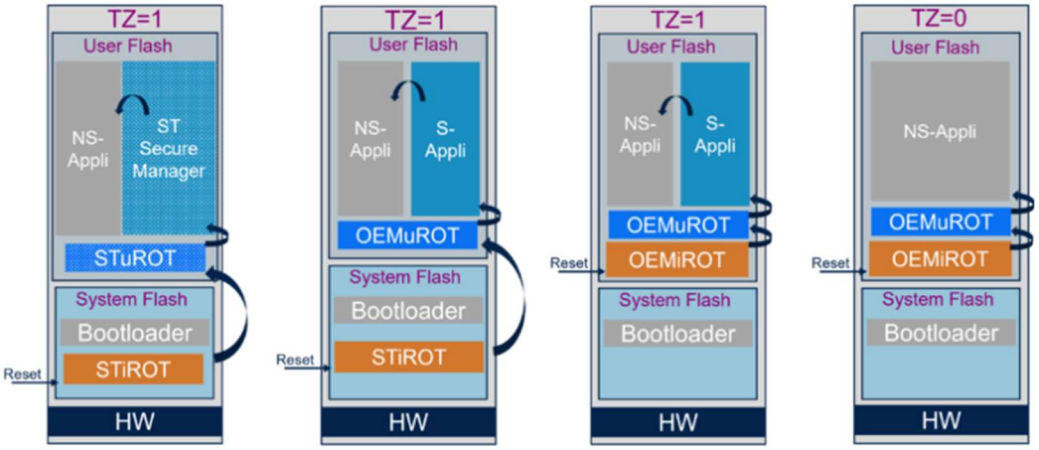
In the case of STM32H533 and STM32H573, they evaluate the Unique Boot Entry (UBE) option byte setting, to determine whether STiRoT is the selected boot option.

From the STiRoT, the Secure Boot also continues in the user flash memory, by either using the STMicroelectronics installable services, like Secure Manager, or customized code.
From the Secure Boot chain point of view, the latter is an OEMuROT.



Depending on the selected boot address, there are different options for Secure Boot progression.
The upper path leading to the Secure Manager is the path targeting PSA level 3 security certification.
This is only available on the STM32H573 crypto product line.

The figure below gives another view of some possible boot paths for the STM32H53 and STM32H57 series.

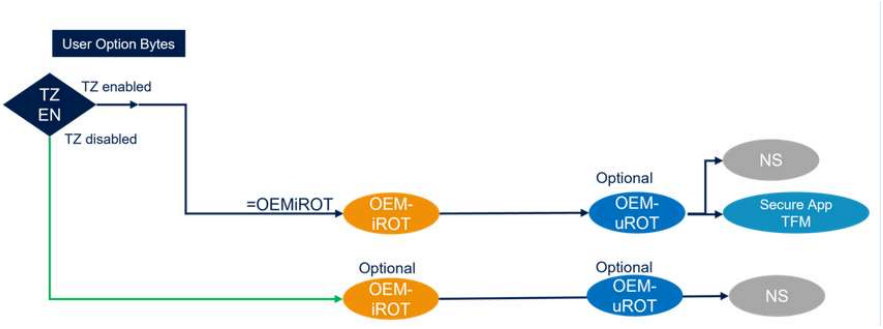


Boot paths for STM32H52x and STM32H56x

The STM32H52x and STM32H56x is not supporting the STiRoT and STuRoT because the cryptographic hardware accelerator is not supported.

Without a hardware cryptographic accelerator, authentication and integrity verification are ensured using the middleware cryptographic library.

When TrustZone® is enabled (TZEN) the OEMiRoT is executed after reset. The figure shows also the optional Updatable Root of Trust (uRoT), which is a second possible boot stage



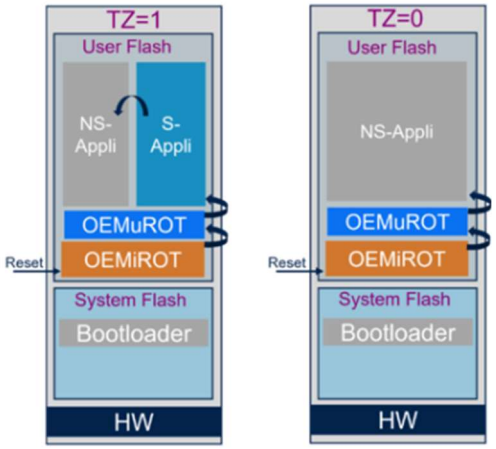
HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

The figure below gives another view of some possible boot paths for the STM32H52 and STM32H56 series.



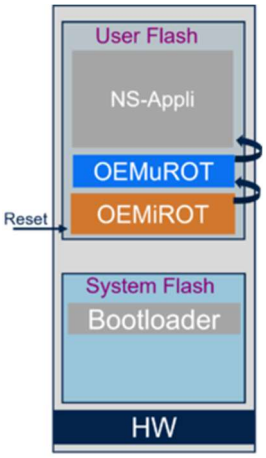
Boot paths for STM32H50x

The STM32H50x supports temporal isolation (HDPL) (as for the other STM32H5).

The figure below shows that with STM32H50x the boot can be done through up to three isolated stages.
For STM32H50x, the TrustZone® disabled is the only supported boot path.

A secure user application is not supported since TrustZone® is disabled.
With TrustZone® disabled it's possible to have only a non-secure user application (OEMiRoT and OEMuRoT are optional).

The figure below gives another view of the only one boot path for the STM32H50 series.



HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Product Lifecycle Introduction

The product life cycle defines the non-volatile product states of the platform that come with different levels of protection.
It is used when considering the product in its different usages:

- Under development (fully open)
- States related to manufacturing (limited debug)
- Final state when configuration is finalized (no more debug access)

As an example, in the development, the debug is fully available to ease the developer life.

When starting to provision official (final product configuration) and sensible assets, consider using a provisioning state where the debug is limited to protect the firmware and the data under provision.

The product state can consider different states depending on the number of sub-systems. Typically, when TrustZone is supported, third-party and non-secure application by another third-party can handle TrustZone sub-system.

It can be implemented by hardware control, and can be completed thanks to software controls.
STM32 microcontrollers products support two kind of hardware life cycle control depending on product families:

- Based on RDP (Read Out Protection): RDP0, RDP0.5, RDP1, RDP2.
- Based on PRODUCT_STATEs: Open, Provisioning, iROT-Provisioned, TZ-Closed, Closed/Locked.

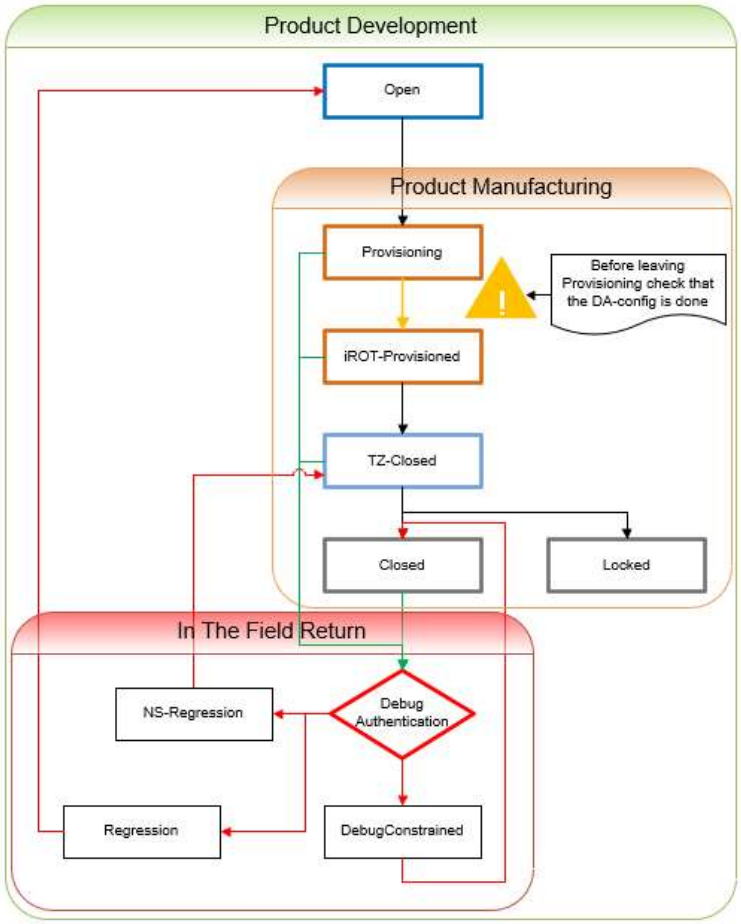
A new product lifecycle is being introduced with the STM32H5 family, to bring added flexibility to product manufacturing and maintenance.

The new product lifecycle includes the following phases:

- Development phase, offering full debug capabilities to the developer
- Provisioning phase, the main asset areas are protected (no longer accessible)
- Final phase, the product is in the field
- Maintenance phase, including field return management

During the product life, the solution must guarantee that the ROT (Root of Trust) and user assets are never disclosed.
This must be true for all four phases stated above.
The new product lifecycle lists all the product states:

When TrustZone® (TZ) is not proposed, the TZ-Closed does not exist.



HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

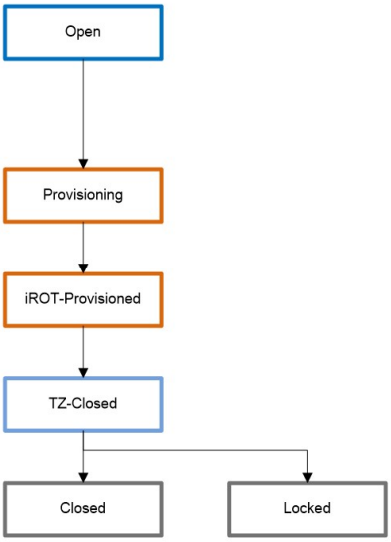
Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Lifecycle Product State Open

The open state corresponds to the default state of the product when it is delivered.
This state is ideal to use while under development, as the debug is fully available.
In this state, the BOOT path configuration can be done using TZEN and UBE option bytes, and most of the features can be used as HDPL, TrustZone, and so on.



Default state when product delivered (fully open)

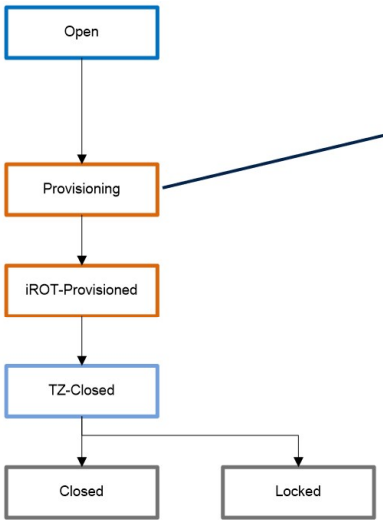
This state is useful for development
- Debug is fully open
- Boot configuration can be SET
  - TZEN = Enabled/Disabled
  - UBE = ST-iROT (only H57x)/OEM-iROT
- HDPL Isolation available
- Trust Zone isolation available
- Bootloader can be used to provision FW, OBK, …

- WARNING:
  - RHUK not set (similar to all products)
  - Means provisioning in this state will not be valid for others

Lifecycle Product State Provisioning

The provisioning state allows the provisioning of the device, while taking care of the asset protection.
It is recommended to manage the product provisioning in this state, in order to benefit of:

- the asset protection => in this state, the debug is limited to HDPL3-NonSecure.
- the encryption of the assets done using a valid DHUK to encrypt the OBKeys or other information, when the SAES is available. In fact, the RHUK is not differentiated in the open state, whereas it is unique for each product in all of the other states.
- the bootloader and JTAG/SWD availability for HDPL3-NS.



WARNING: Debug Authentication only possible if DA-config provisioned
Take care to provision DA-config

State to use for provisioning
Initial or/and Full provisioning

This state is to be used to Provision the device
- RHUK is unique per Device from this state ➔ Provisioning MUST be done in this state.
- Debug is only available to HDPL3-NS
- Boot configuration can be SET
  - TZEN = Enabled/Disabled
  - UBE = ST-iROT (only H57x)/OEM-iROT
- HDPL Isolation available
- Trust Zone isolation available
- Bootloader can be used to provision FW, OBK, …
- SFI can be launch in this state, no more in the next

- WARNING:
  - The DA-config provisioning MUST be done in this state

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

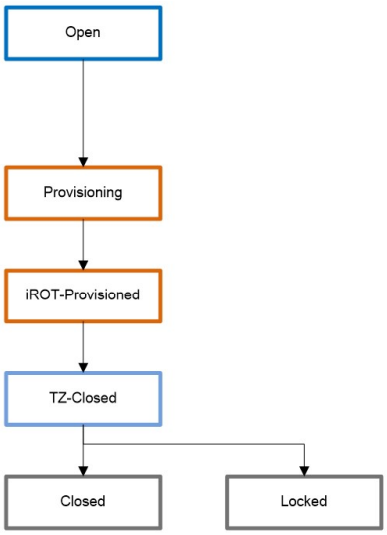
Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Lifecycle Product State iROT-Provisioned

The iROT-Provisioned state corresponds to an intermediate state of the product.
The iROT code and data are provisioned.
From this state, the next level can be updated with firmware update mechanism.

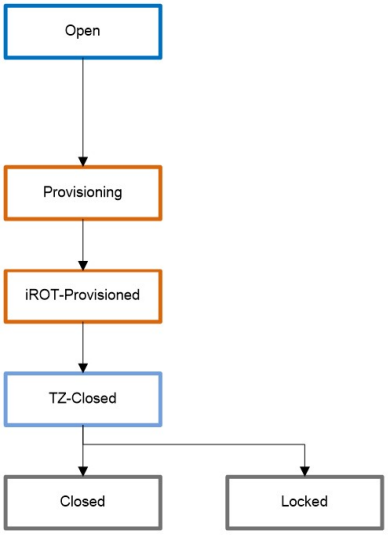


ST-iROT or OEM-iROT setup done

This state is to be used when almost 2 OEMs are used and one of the OEM manage the iROT
- In this state, the iROT must allow the update of the next level(s) : Code & Data
- Debug is only available to HDPL3-NS
- iROT can launch Bootloader if next level not OK
- Debug Authentication can be launch

Lifecycle Product State TZ-Closed

The TZ-Closed state corresponds to an intermediate state of the product.
All the secure firmware is installed, the non-secure application can be developed, or loaded in a second instance.
The iROT + uROT (optional) + Secure-OS code and data are provisioned.
From this state, the non-secure application can be updated using the firmware update mechanism, or directly programmed through the flash loader (embedded in the IDE).



This state is to be used when almost iROT and [uROT+SecureOS] are provisioned
- In this state, the uROT is supposed to allow the update of the next level(s) : Code & Data
- Debug is only available to HDPL3-NS
- uROT can launch Bootloader if next level not OK
- Debug Authentication can be launch

Secure OS is installed

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

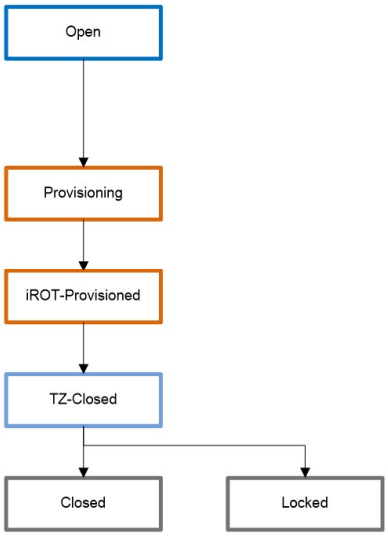
Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Lifecycle Product State Closed/Locked

The closed state corresponds to the final state of the product.
All debug accesses are closed, and only accessible through debug authentication.
From closed state, the debug authentication can be used to launch partial or full regressions, or to open the debug.
The locked state corresponds to the final state of the product. All debug accesses are locked.



This state is to be used when the product is fully provisioned

- The Debug is fully closed
- Closed state allows to use Debug Authentication
- Locked state do not allow any Debug reopening nor regression management.

The product is fully provisioned

Debug Authentication Introduction

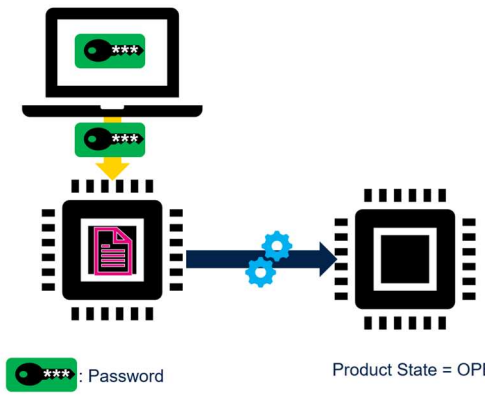
The user leverages on the debug authentication security feature to either:

- Perform secure regression testing on the OPEN or TZ-CLOSED product states, erasing user data in user flash memory, SRAM and OBKeys.
- Safely re-open debug access on the STM32.

Debug authentication can be used when the STM32 is in product state PROVISIONING up to product state CLOSED.
Debug authentication only grants its services when user enters a legitimate user password or a certificate authentication.
The STM32 lifecycle management drives the STM32 debug authentication method, password or certificate.
The user sends a password or certificate to the STM32 via a limited debug access port that does not grant access to STM32 hardware resources such as the CPU registers, memory, peripheral registers, and so on.
When a debug access on a closed device is opened, a new debug access port is also opened, granting access to the STM32 hardware resources.

Debug Authentication Password

In order to access the debug authentication feature, the host sends the debug authentication password to the STM32.
When the STM32 receives the password, it verifies that it corresponds to the one that is provisioned.



HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

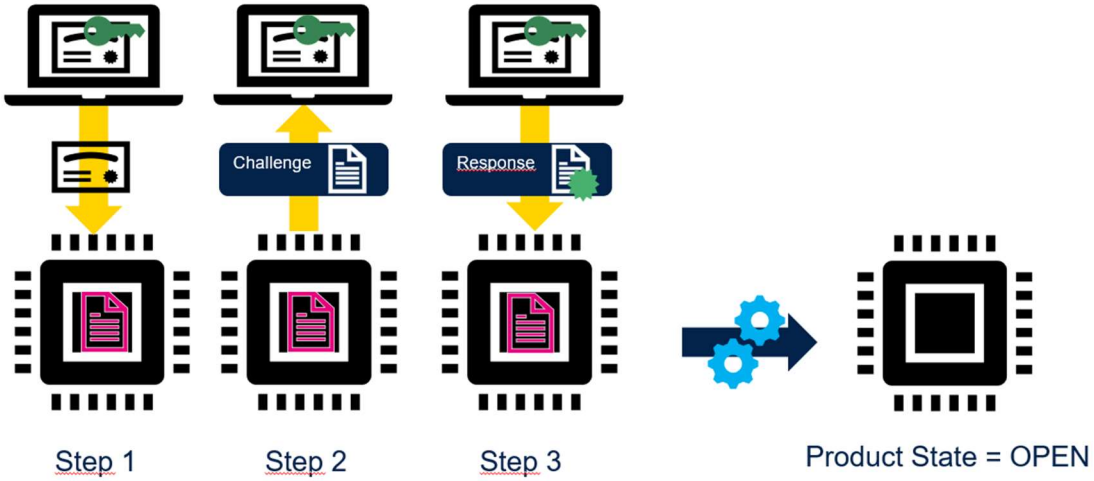
→smh-tech.com
info@smh-tech.com

Debug Authentication Certificate

When the user accesses the debug authentication feature (regression or debug re-opening), he sends first a certificate and a debug authentication action request to the STM32.

When the STM32 receives the certificate, it verifies that:

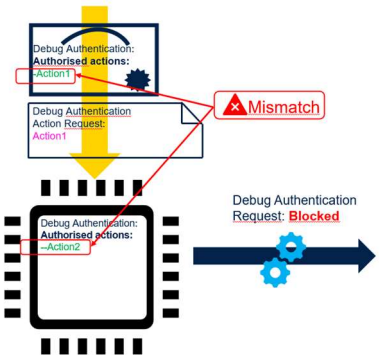
- Certificate fits the one that is provisioned.
- The authorized actions sent with the certificate match the ones provisioned.
- The action request matches the authorized action list carried by the certificate.
- The STM32 starts the challenge-response procedure (Step 2 and Step 3): the STM32 verifies that the host owns the debug authentication private key before performing the requested action (regression or debug re-opening). The certificate carries the requested action.



: Debug Authentication Certificate

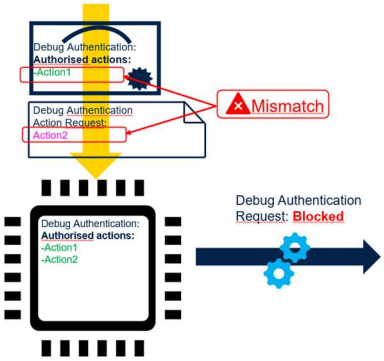
: Debug Authentication Private Key

The next figure illustrates how the STM32 blocks the debug authentication action request when authorized actions in certificate do not match the provisioned authorized actions.



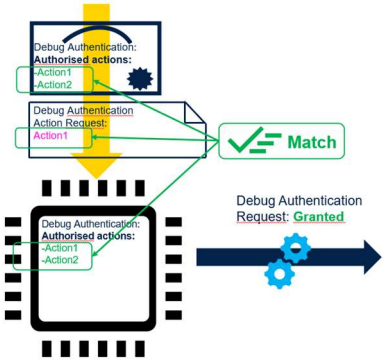
: Debug Authentication Certificate

The figure below illustrates how the STM32 blocks the debug authentication action request because the action request does not fit with certificate authorized actions.



: Debug Authentication Certificate

The figure below illustrates that the certificate authorized actions fit the provisioned authorized action and the action request fits the certificate authorized actions, then STM32 grants debug authentication action request.



: Debug Authentication Certificate

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

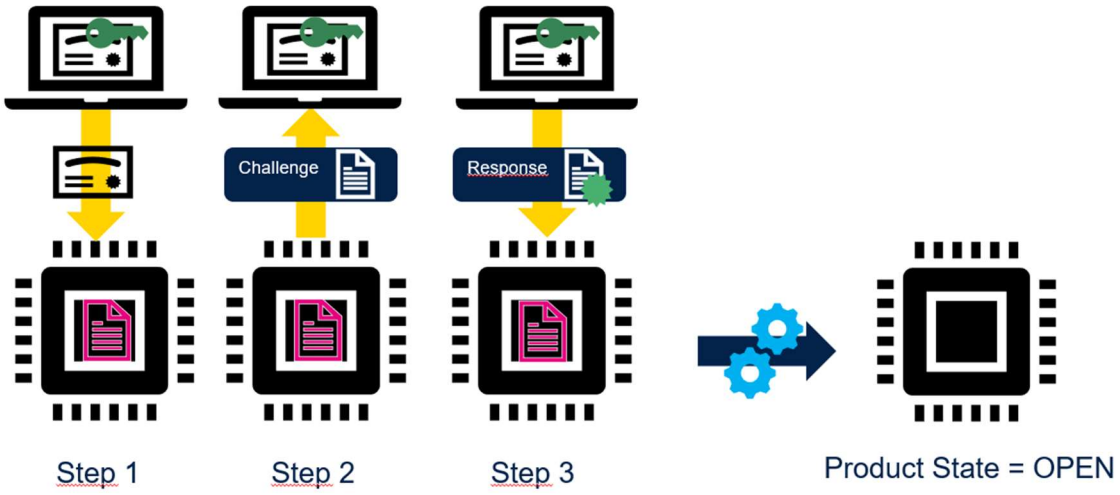
D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

SMH Technologies®

Debug Authentication Full Regression

Debug authentication erases the whole of the user flash memory, SRAM and OBKEys[1]. Full regression erases the debug authentication provisioned data, which is the certificate and authorized actions, hence the user must provision debug authentication data once again. After full regression, the STM32 is in the OPEN product state.

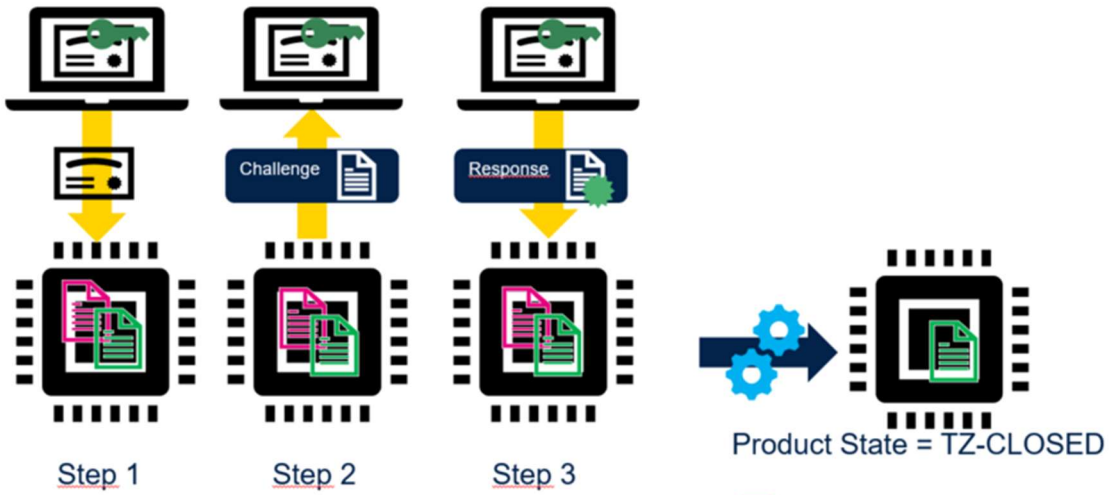


: Debug Authentication Certificate

: Debug Authentication Private Key

Debug Authentication Partial Regression

Debug authentication erases non-secure user flash memory, non-secure SRAM and non-secure OBKEys1. After partial regression, the STM32 is in the TZ-CLOSED product state.



: Debug Authentication Certificate

: Debug Authentication Private Key

: Non-secure firmware

: Secure firmware

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Debug Authentication Virtual Memories

To allow the user to execute the Debug Authentication procedure in a very simple way, we add three different virtual memories.

```
[P] - Provisioning (virtual memory)
[R] - Regression (virtual memory)
[M] - Secure Manager SFI (virtual memory)
```

Here you can see these virtual memories and how these are divided internally.

[P] - Provisioning Virtual Memory

This virtual memory area is a virtual space where user must put the data to be used for Debug Authentication Provisiong procedure.
This area is virtually divided into sub-section to provide to FlashRunner different files to be used during the Provision.

[P] - PROVISIONING			
VIRTUAL SECTION NAME	VIRTUAL SECTION START ADDRESS	VIRTUAL SECTION END ADDRESS	VIRTUAL SECTION SIZE
PASSWORD	0xF0000000	0xF0001FFF	8 KiB
CERTIFICATE	0xF0002000	0xF0003FFF	8 KiB
iROT CONFIG	0xF0004000	0xF0005FFF	8 KiB
iROT DATA	0xF0006000	0xF0007FFF	8 KiB
uROT CONFIG	0xF0008000	0xF0009FFF	8 KiB
uROT DATA	0xF000A000	0xF000BFFF	8 KiB

[R] – Regression Virtual Memory

This virtual memory area is a virtual space where user must put the data to be used for Debug Authentication Regression procedure.
This area is virtually divided into sub-section to provide to FlashRunner different files to be used during the Regression.

[R] - REGRESSION			
VIRTUAL SECTION NAME	VIRTUAL SECTION START ADDRESS	VIRTUAL SECTION END ADDRESS	VIRTUAL SECTION SIZE
PASSWORD	0xF0010000	0xF0011FFF	8 KiB
PRIVATE KEY	0xF0012000	0xF0013FFF	8 KiB
CERTIFICATE CHAIN	0xF0014000	0xF0015FFF	8 KiB

[M] – Secure Manager SFI Virtual Memory

This virtual memory area is a virtual space where user must put the data to be used for Debug Authentication Secure Firmware Install SFI procedure.
This area is virtually divided into sub-section to provide to FlashRunner different files to be used during the SFI.

[M] - SECURE MANAGER SFI			
VIRTUAL SECTION NAME	VIRTUAL SECTION START ADDRESS	VIRTUAL SECTION END ADDRESS	VIRTUAL SECTION SIZE
RSSe LIBRARY	0xF0020000	0xF005FFFF	256 KiB
GLOBAL LICENSE	0xF0060000	0xF009FFFF	256 KiB
SFI FILE	0xF00A0000	0xF049FFFF	4096 KiB

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Debug Authentication Commands

To allow the user to execute the Debug Authentication procedure in a very simple way, we add four different commands.

#TPCMD DISCOVERY

Syntax: `#TPCMD DISCOVERY`

Parameters: none

Prerequisites: `#TPCMD CONNECT` executed

Description: Perform the Debug Authentication Discovery command to get device informations useful for other Debug Authentication commands

Note: This command is available from **libstm32.so** version **5.33**

Examples: Correct command execution: 😊

Real Time Log:

```
---#TPCMD DISCOVERY
Receiving Discovery response from STM32H5 device:
 * PSA auth version:                v1.0
 * Vendor ID:                       STMicroelectronics
 * SoC Class:                       0x484 - STM32H5xxxx
 * SoC ID:                          0x00630052 0x33325116 0x38363236 0x00000000
 * HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
 * PSA lifecycle:                   ST_LIFECYCLE_PROVISIONING
 * SDA version:                     v2.4.0
 * Token Formats:                   0x0200
 * Certificate Formats:             0x0201
 * Cryptosystems:                   Ecdsa-P256 SHA256
 * ST provisioning integrity status: 0xEAEAEAEA
 * ST HDPL1 status:                 0xFFFFFFFF
 * ST HDPL2 status:                 0x00000000
 * ST HDPL3 status:                 0xFFFFFFFF
> Received Discovery response correctly.
Available Permission for selected device:
 * [1] ==> Full Regression
 * [2] ==> To Trust Zone Regression
 * [3] ==> Level 3 Intrusive Debug
 * [4] ==> Level 2 Intrusive Debug
 * [5] ==> Level 1 Intrusive Debug
 * [6] ==> Level 3 Intrusive Non Secure Debug
 * [7] ==> Level 2 Intrusive Non Secure Debug
 * [8] ==> Level 1 Intrusive Non Secure Debug
> Completed Discovery command.
Time for Discovery: 0.346 s.
>|
```

Terminal:

```
01|PSA auth version:                v1.0
01|Vendor ID:                       STMicroelectronics
01|SoC Class:                       0x484 - STM32H5xxxx
01|SoC ID:                          0x00630052 0x33325116 0x38363236 0x00000000
01|HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
01|PSA lifecycle:                   ST_LIFECYCLE_PROVISIONING
01|SDA version:                     v2.4.0
01|Token Formats:                   0x0200
01|Certificate Formats:             0x0201
01|Cryptosystems:                   Ecdsa-P256 SHA256
01|ST provisioning integrity status: 0xEAEAEAEA
01|ST HDPL1 status:                 0xFFFFFFFF
01|ST HDPL2 status:                 0x00000000
01|ST HDPL3 status:                 0xFFFFFFFF
01|Available Permission for selected device:
01|[1] ==> Full Regression
01|[2] ==> To Trust Zone Regression
01|[3] ==> Level 3 Intrusive Debug
01|[4] ==> Level 2 Intrusive Debug
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
01|[5] ==> Level 1 Intrusive Debug
01|[6] ==> Level 3 Intrusive Non Secure Debug
01|[7] ==> Level 2 Intrusive Non Secure Debug
01|[8] ==> Level 1 Intrusive Non Secure Debug
```

#TPCMD PROVISIONING

Syntax: `#TPCMD PROVISIONING <Trust Zone mode> <Program Flash mode> <iROT path> <Finish Product State value>`

Parameters:

<Trust Zone mode>
- TRUST_ZONE_ENABLED
- TRUST_ZONE_DISABLED

<Program Flash mode>
- PROGRAM_FLASH
- SKIP_PROGRAM_FLASH

<iROT path>
- STANDARD
- OEMiROT
- OEMiROT_uROT
- STiROT
- STiROT_uROT

<Finish Product State value>
- PROVISIONING
- IROT_PROVISIONED
- TRUST_ZONE_CLOSED
- CLOSED
- LOCKED

Prerequisites: `#TPCMD CONNECT` executed
Device Product State accepted are OPEN, PROVISIONING, IROT_PROVISIONED or TRUST_ZONE_CLOSED

Description: This command performs the entire Debug Authentication Provisioning procedure.

Note: This command is available from **libstm32.so** version **5.33**

Examples: Correct command execution: 😊

```
---#TPCMD PROVISIONING TRUST_ZONE_ENABLED PROGRAM_FLASH STiROT CLOSED
Provisioning request:
 * Trust Zone requested is enabled.
 * Program Flash is executed before moving to final state.
 * Immutable Root of Trust requested is STiROT.
 * Finish Product State requested is Locked.
> Provisioning command set correctly.
Check device configuration:
 * Device: STM32H573ZI.
 * Flash Size: 2048 KB.
 * Product State: 0xED - Open.
 * Trust Zone: 0xC3 - Disabled.
 * SFSP Silicium Version: v2.5.0.
> Device Configuration is correct.
Set Trust Zone to enabled:
> Trust Zone is enabled correctly.
Set Boot UBE to ST-iRot:
> Boot UBE is already ST-iRot.
Program the Firmware inside Non-secure Flash memory:
 * Time for Masserase Non-secure Flash: 0.413 s.
 * Time for Blankcheck Non-secure Flash: 0.035 s.
 * Time for Program Non-secure Flash: 0.270 s.
 * Time for Verify Readout Non-secure Flash: 0.056 s.
 * Time for Verify Checksum 32bit Non-secure Flash: 0.009 s.
> Firmware programmed correctly into Non-secure Flash memory.
Set Product State to 0x17 - Provisioning:
> Product State set to Provisioning correctly.
Provision the Certificate OBK file:
 * Process the OBK file:
   * Destination address: 0x0FFD0100.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
    * Size: 0x00000060.
    * Do Encryption: 1.
  * Send the OBK file:
> Certificate OBK file provisioned successfully.
Provision the STiROT Config OBK file:
  * Process the OBK file:
    * Destination address: 0x0FFD0200.
    * Size: 0x00000100.
    * Do Encryption: 1.
  * Send the OBK file:
> STiROT Config OBK file provisioned successfully.
Provision the STiROT Data OBK file:
  * Process the OBK file:
    * Destination address: 0x0FFD0400.
    * Size: 0x00000060.
    * Do Encryption: 1.
  * Send the OBK file:
> STiROT Data OBK file provisioned successfully.
Final Product State requested is 0x72 - Closed:
> Final Product State set correctly.
Time for Provisioning: 1.552 s.
>|
```

#TPCMD REGRESSION

Syntax: `#TPCMD REGRESSION <Permission request>`

Parameters: <Permission request>

- FULL_REGRESSION
- TO_TRUST_ZONE
- LEVEL_3_INTRUSIVE_DEBUG
- LEVEL_2_INTRUSIVE_DEBUG
- LEVEL_1_INTRUSIVE_DEBUG
- LEVEL_3_INTRUSIVE_NON_SECURE_DEBUG
- LEVEL_2_INTRUSIVE_NON_SECURE_DEBUG
- LEVEL_1_INTRUSIVE_NON_SECURE_DEBUG

Prerequisites: `#TPCMD CONNECT` executed
Device Product State is not OPEN

Description: This command performs the entire Debug Authentication Regression procedure.

Note: This command is available from **libstm32.so** version **5.33**

Examples: Correct command execution: 😊

```
---#TPCMD REGRESSION FULL_REGRESSION
ST provisioning integrity status: 0xEAEAEAEA.
Import Private Key from FRB file:
  * Importing EC P-256 key.
> Private Key imported correctly.
Import Trust of Chain from FRB file:
  * Found 3 certificates.
> Trust of Chain imported correctly.
Sending Challenge request to the device:
> Received Challenge correctly.
Sign the Token based on Challenge received:
> Signing Token completed successfully.
Send the data to the device to perform Authentication:
  * Sending Certificate n. 1.
  * Sending Certificate n. 2.
  * Sending Certificate n. 3.
  * Sending Authentication Token.
> Authentication completed successfully.
Wait until device reset and reboot:
> Reset detected and device reboots now.
Receiving Discovery response from STM32H5 device:
  * PSA auth version:                v1.0
  * Vendor ID:                       STMicroelectronics
  * SoC Class:                       0x484 - STM32H5xxxx
  * SoC ID:                          0x00630052 0x33325116 0x38363236 0x00000000
  * HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
  * PSA lifecycle:                   ST_LIFECYCLE_OPEN
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
 * SDA version:                      v2.4.0
 * Token Formats:                    0x0200
 * Certificate Formats:              0x0201
 * Cryptosystems:                    ST Password
 * ST provisioning integrity status: 0xF5F5F5F5
 * ST HDPL1 status:                  0xFFFFFFFF
 * ST HDPL2 status:                  0xFFFFFFFF
 * ST HDPL3 status:                  0xFFFFFFFF
> Completed Discovery command.
Time for Regression: 2.488 s.
>|
```

#TPCMD SECURE_FIRMWARE_INSTALL

Syntax: `#TPCMD SECURE_FIRMWARE_INSTALL`

Parameters: none

Prerequisites: `#TPCMD CONNECT` executed
Device Product State is OPEN or PROVISIONING

Description: This command performs the entire Debug Authentication SFI procedure.

Note: This command is available from **libstm32.so** version **5.33**

Examples: Correct command execution: 😊

```
---#TPCMD SECURE_FIRMWARE_INSTALL
Parse SFI file from FRB to check configuration:
> SFI file contains supported areas.
Check current device Product State:
> Device is in a valid life state.
Set correct Option Bytes configuration:
> Option Bytes programmed correctly.
Read device Descriptor:
 * Descriptor version: 01.00.00.
 * Start address of the available SRAM for host: 0x20054000.
 * End address of the available SRAM for host: 0x2008FFFF.
> Device Descriptor read successfully.
Load the RSSe library into STM32 device:
 * Import RSSe library from FRB file.
 * Send RSSe library into the device.
 * Verify RSSe library sent to the device.
> RSSe library programmed correctly.
Start RSSe library service:
 * RSSe library service installed.
 * Reset the device to have RSSe ready.
 * Wait until device reboots.
 * Installed RSSe version 2.0.1.
> RSSe library service is ready.
Processing license:
 * Import License from FRB file.
 * Send License into the device.
 * Verify License sent to the device.
> License programmed correctly.
Execute Process License payload.
> Succeed to execute Process License payload.
Import SFI file from FRB:
 * Import SFI from FRB file.
 * Imported file has a valid SFI image header.
 * Protocol version is 2.
 * Total areas number is 13.
> SFI file imported correctly.
Processing SFI Image Header.
> Completed processing SFI Image Header after 0.001 s.
Processing SFI Area n. 1 marked as H.
> Completed processing SFI Area H after 0.946 s.
Processing SFI Area n. 2 marked as O.
> Completed processing SFI Area O after 0.002 s.
Processing SFI Area n. 3 marked as O.
> Completed processing SFI Area O after 0.005 s.
Processing SFI Area n. 4 marked as O.
> Completed processing SFI Area O after 0.008 s.
Processing SFI Area n. 5 marked as O.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
> Completed processing SFI Area O after 0.003 s.
Processing SFI Area n. 6 marked as S.
> Completed processing SFI Area S after 1.136 s.
Processing SFI Area n. 7 marked as S.
> Completed processing SFI Area S after 0.826 s.
Processing SFI Area n. 8 marked as S.
> Completed processing SFI Area S after 0.736 s.
Processing SFI Area n. 9 marked as S.
> Completed processing SFI Area S after 0.166 s.
Processing SFI Area n. 10 marked as F.
> Completed processing SFI Area F after 0.090 s.
Processing SFI Area n. 11 marked as F.
> Completed processing SFI Area F after 0.005 s.
Processing SFI Area n. 12 marked as F.
> Completed processing SFI Area F after 0.009 s.
Processing SFI Area n. 13 marked as C.
> Completed processing SFI Area C after 0.001 s.
Device reboots now.
> SFI process completed successfully.
Time for Secure Firmware Install: 6.496 s.
>|
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Debug Authentication Flow

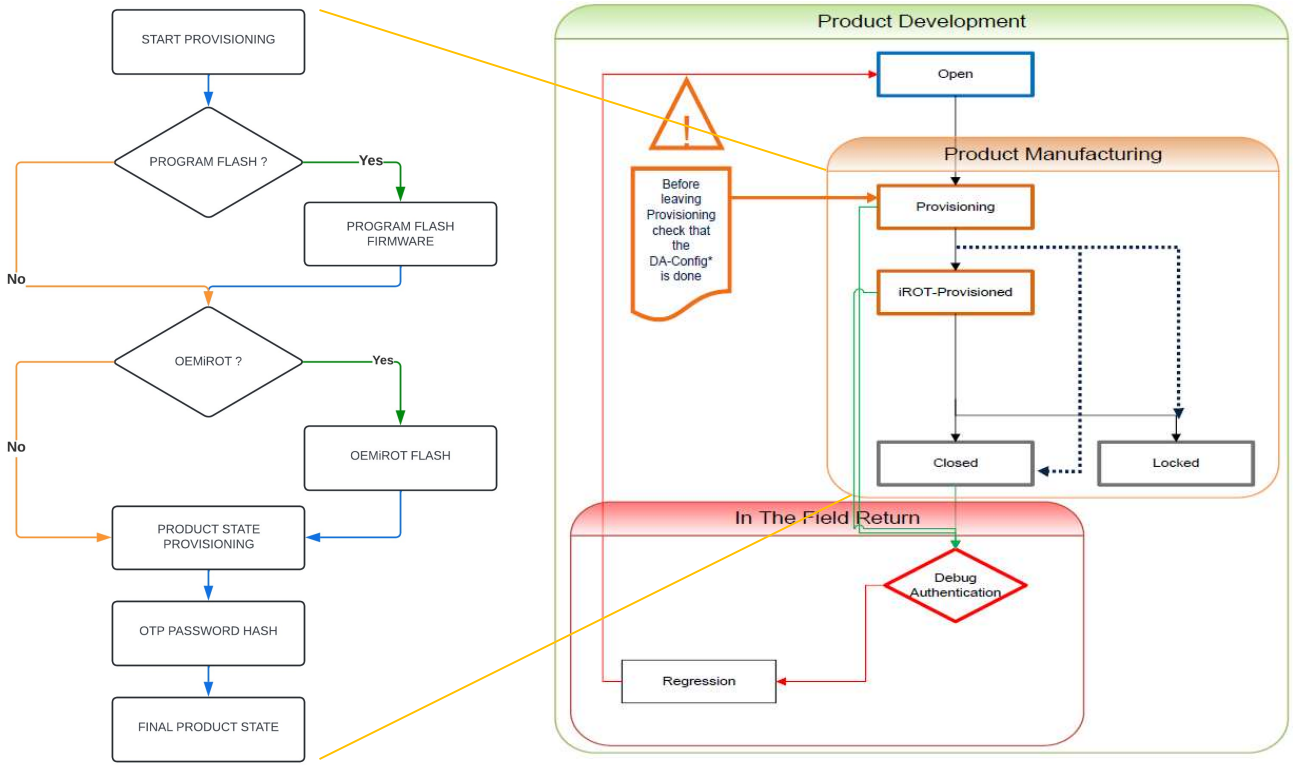
There are a lot of different flow for Debug Authentication based on target device and customer settings.

First of all, we need to categorize the STM32H5 devices:

STM32H5 CLASSIFICATION			
DEVICE FAMILY	OTP BASED	NON-CRYPTO DEVICE	CRYPTO DEVICE
STM32H503xx	✓		
STM32H523xx		✓	
STM32H533xx			✓
STM32H562xx		✓	
STM32H563xx		✓	
STM32H573xx			✓

Debug Authentication flow for OTP based devices

This graph describes all the possible provisiong flow for OTP based devices:



HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

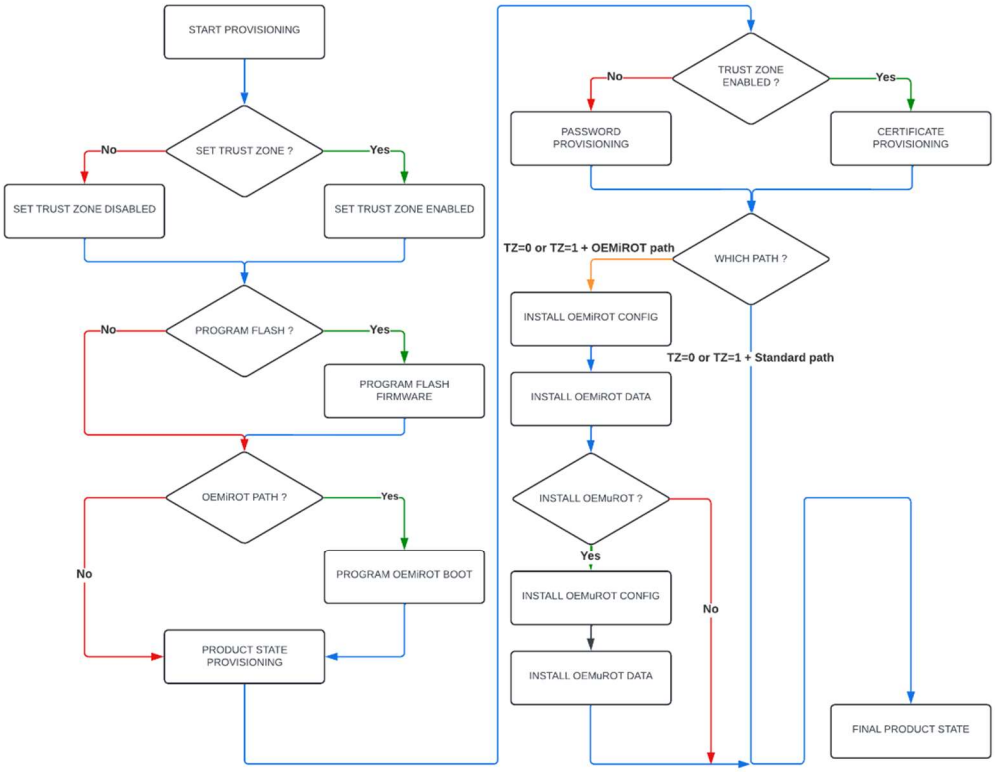
Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Debug Authentication flow for non-crypto devices

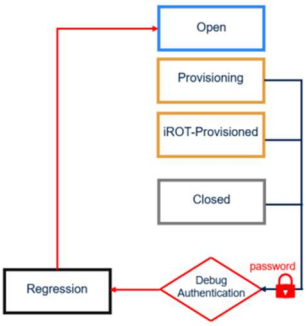
This graph describes all the possible provisiong flow for STM32H5 non-crypto devices based.



Trust Zone Disabled:

Provisioning with password management:

- Provisioning data is located at the beginning of the OBKeys-HDPL1 area.
- Provisioning data is the HASH (SHA256) of the password.
- Only Full Regression is available.



HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

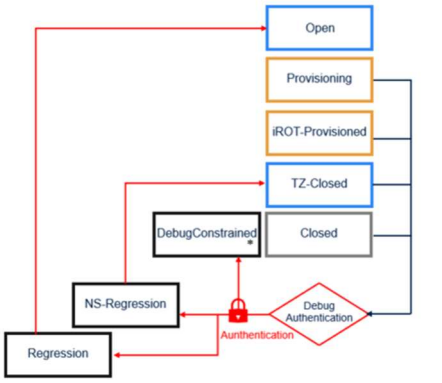
D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Trust Zone Enabled:

Provisioning with certificate management:

- Provisioning data is located at the beginning of the OBKeys -HDPL1 area.
- Provisioning data contain:
  - HASH (SHA256) of the root Certificate Public Key.
  - SOC_PERMISSION: 16 bits defining the permissions authorized by default.



** **Debug constrained** corresponds to the reopen of the debug of code secure or non-secure without performing any regression*

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

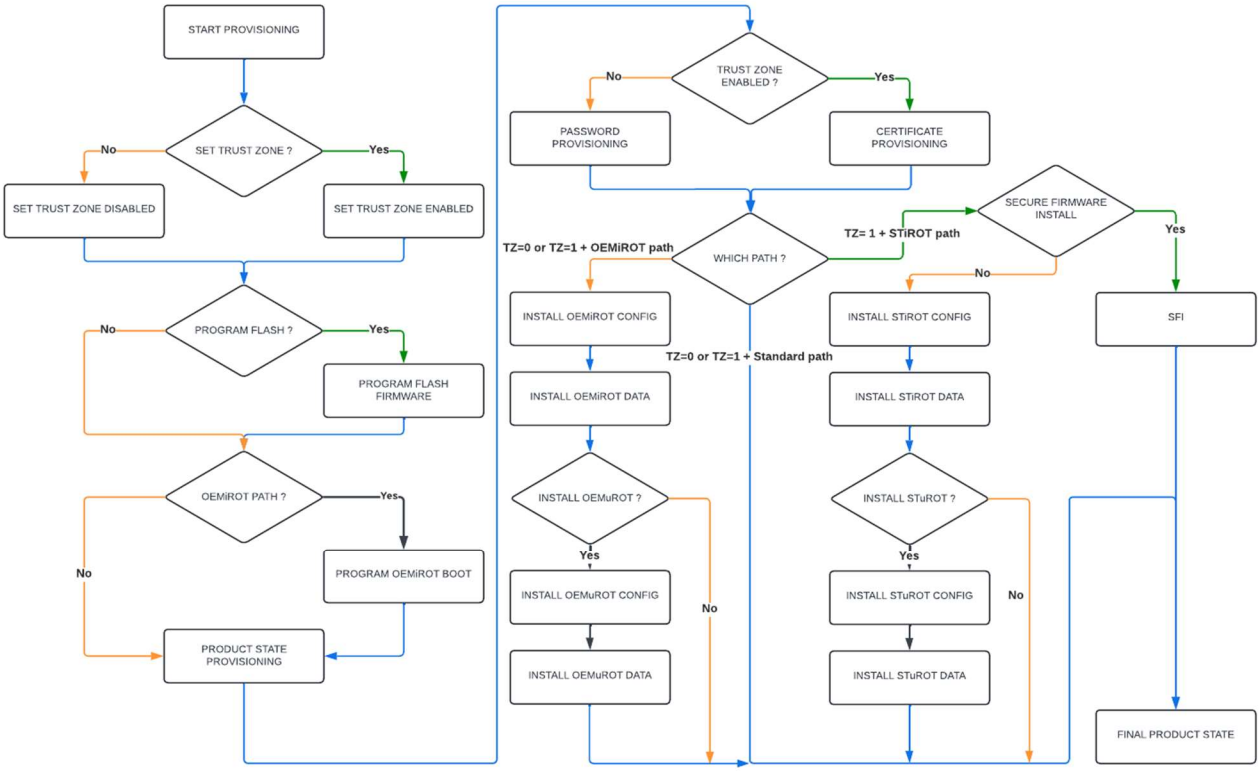
D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Debug Authentication flow for crypto devices

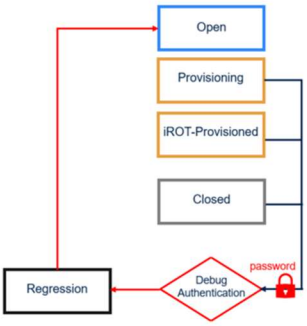
This graph describes all the possible provisiong flow for STM32H5 crypto devices based.



Trust Zone Disabled:

Provisioning with password management:

- Provisioning data is located at the beginning of the OBKeys-HDPL1 area.
- Provisioning data is the HASH (SHA256) of the password.
- Only Full Regression is available.



HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

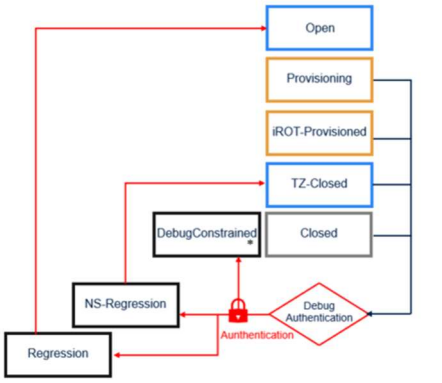
D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Trust Zone Enabled:

Provisioning with certificate management:

- Provisioning data is located at the beginning of the OBKeys -HDPL1 area.
- Provisioning data contain:
  - HASH (SHA256) of the root Certificate Public Key.
  - SOC_PERMISSION: 16 bits defining the permissions authorized by default.



** **Debug constrained** corresponds to the reopen of the debug of code secure or non-secure without performing any regression*

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

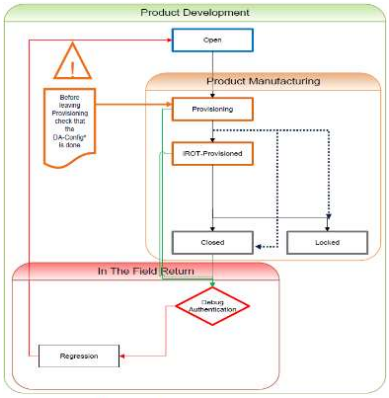
→smh-tech.com

info@smh-tech.com

Debug Authentication Examples

Here you can find a lot of examples on how to use the Debug Authentication procedure in different flow and cases.

Debug Authentication STM32H503xx OTP Password Hash

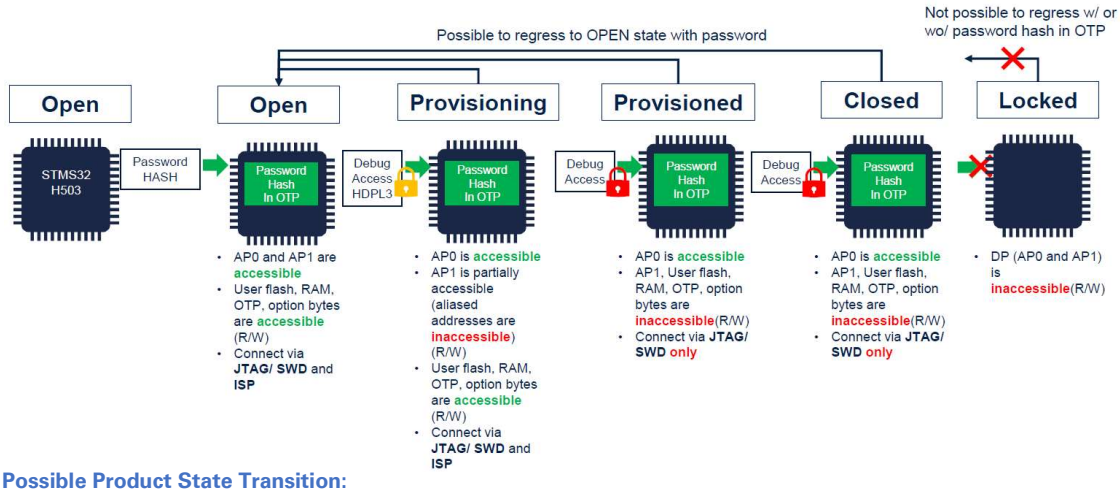


Device configuration and setup:

STM32H503RB
No Trust Zone Available
Password for Debug Authentication
Password Hash into OTP memory
Default Option Bytes

Flow for Debug Authentication:

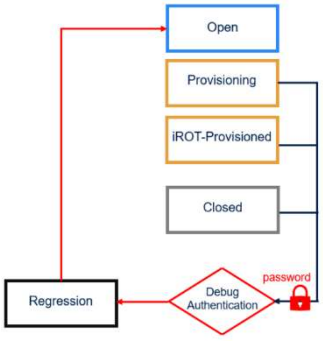
OPEN -> PROVISIONING -> iROT-PROVISIONED -> END STATE -> REGRESSION



Possible Product State Transition:

Is only allowed to transit from Open to other possible product states, but not the other way around.

From	To					
	Open	Provisioning	Provisioned	Closed	Locked	Regression (open)
Open	-	OK	OK	X	X	X
Provisioning	X	-	OK	OK	OK	OK (HDPL1)
Provisioned	X	X	-	OK	OK	OK (HDPL1)
Closed	X	X	X	-	X	OK (HDPL1)
Locked	X	X	X	X	-	X



Prepare data for Debug Authentication using Workbench and FRB files:

To perform Debug Authentication with OTP HASP password you need to use the following two files:

board_password.bin

Offset	+0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +a +b +c +d +e +f	Equivalent ASCII characters
00000000	ff f3 a9 bc dd 37 36 3d 70 3c 1c 4f 95 12 53 36	7 6 = p < O S 6
00000010	86 15 78 68 f0 d4 f1 6a 0f 02 d0 f1 da 24 f9 a2	x h j $

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

This file contains the Password Hash and need to be programmed ad address 0x08FFF000.
The Hash is a **SHA256** hash of user_password.bin file.

Please note that OTP is **ONE TIME PROGRAMMABLE**, cannot be changed anymore.

This file is necessary for **Provisioning**.

password.bin

```
Offset    +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +a +b +c +d +e +f   Equivalent ASCII characters
00000000  00 00 00 80 0c 00 00 00 00 01 02 03 04 05 06 07
00000010  08 09 0a 0b
```

This file is necessary for **Regression**.

FRB file generation for Provisioning and Regression:

Now you need to import these two files inside the Workbench to create a correct FRB file for Debug Authentication:

Import board_password.bin as a binary (.bin extension) and edit the address position to 0x08FFF000

[P] - PROVISIONING			
VIRTUAL SECTION NAME	VIRTUAL SECTION START ADDRESS	VIRTUAL SECTION END ADDRESS	VIRTUAL SECTION SIZE
PASSWORD	0xF0000000	0xF0001FFF	-
CERTIFICATE	0xF0002000	0xF0003FFF	-
iROT CONFIG	0xF0004000	0xF0005FFF	-
iROT DATA	0xF0006000	0xF0007FFF	-
uROT CONFIG	0xF0008000	0xF0009FFF	-
uROT DATA	0xF000A000	0xF000BFFF	-

Import password.bin as a binary (.bin extension) and edit the address position to 0xF0010000

[R] - REGRESSION			
VIRTUAL SECTION NAME	VIRTUAL SECTION START ADDRESS	VIRTUAL SECTION END ADDRESS	VIRTUAL SECTION SIZE
PASSWORD	0xF0010000	0xF0011FFF	password.bin
PRIVATE KEY	0xF0012000	0xF0013FFF	-
CERTIFICATE CHAIN	0xF0014000	0xF0015FFF	-

If you want you can also add the **Flash Firmware** inside **Non-secure Flash** memory.

Here your FRB file from Advanced FRB Manager:

File Name	Memory Area	Target Start Address	Target End Address
128KB.bin	[F] - Flash	0x08000000	0x0801FFFF
board_password.bin	[T] - OTP Area	0x08FFF000	0x08FFF01F
password.bin	[R] - Regression (virtual memory)	0xF0010000	0xF0010013

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Possible Flow using Provisiong and Regression:

First of all, we need to understand the possible flow for Provisioning and Regression for STM32H503xx devices.
Using the `#TPCMD PROVISIONING` command, you can choose one of these possible flows:

Some possible flow for Provisiong:

Example 1:

```
Trust Zone Disabled   -> TRUST_ZONE_DISABLED
Skip Program Flash    -> SKIP_PROGRAM_FLASH
Skip OEMiROT          -> STANDARD
End State             -> CLOSED

#TPCMD PROVISIONING TRUST_ZONE_DISABLED SKIP_PROGRAM_FLASH STANDARD CLOSED
```

Example 2:

```
Trust Zone Disabled   -> TRUST_ZONE_DISABLED
Program Flash         -> PROGRAM_FLASH
Skip OEMiROT          -> STANDARD
End State             -> CLOSED

#TPCMD PROVISIONING TRUST_ZONE_DISABLED PROGRAM_FLASH STANDARD CLOSED
```

Example 3:

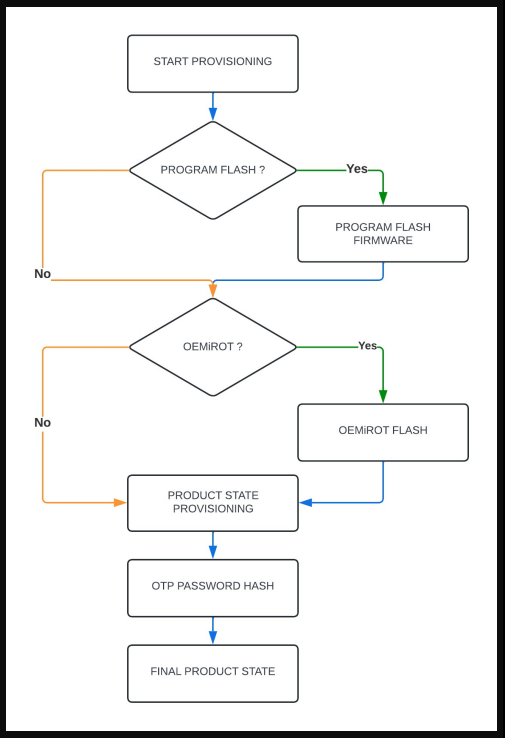
```
Trust Zone Disabled   -> TRUST_ZONE_DISABLED
Program Flash         -> PROGRAM_FLASH
OEMiROT               -> OEMiROT
End State             -> CLOSED

#TPCMD PROVISIONING TRUST_ZONE_DISABLED PROGRAM_FLASH OEMiROT CLOSED
```

Example 4:

```
Trust Zone Disabled   -> TRUST_ZONE_DISABLED
Program Flash         -> PROGRAM_FLASH
OEMiROT               -> OEMiROT
End State             -> LOCKED

#TPCMD PROVISIONING TRUST_ZONE_DISABLED PROGRAM_FLASH OEMiROT LOCKED
```



Using the `#TPCMD REGRESSION` command, you can choose one of these possible flows:

There is only one possible flow for Regression:

Example 1:

```
Regression Mode -> FULL_REGRESSION

#TPCMD REGRESSION FULL_REGRESSION
```

Example using Provisiong and Regression:

Now here we can show to you an example of Provisioning and Regression.
In this case we decide to select example n.2 for Provisioning and the example n.1 for Regression.

Project commands:

```
#TPSETSRC STM32H503xx_Debug_Authentication.frb
#TPSTART
#TPCMD CONNECT
#TPCMD PROVISIONING TRUST_ZONE_DISABLED PROGRAM_FLASH STANDARD CLOSED
#TPCMD DISCOVERY
#TPCMD REGRESSION FULL_REGRESSION
#TPCMD DISCONNECT
#TPEND
```


UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Project execution:

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
Good samples: 16 [Range 0-15].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x54770002, Type: AMBA APB2 or APB3 bus.
AP[1] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[1] ROM table base address 0xE00FE000.
CPUID: 0x410FD214.
Implementer Code: 0x41 - [ARM].
Found Cortex M33 revision r0p4.
Program counter value is 0xEFFFFFFE.
Cortex M33 Core halted [0.001 s].
> Device configuration 1: [0x2D30EDF8]
 * The device's product state is Open [0xED].
 * BOR level 1, the threshold level is low (around 2.1 V).
 * Brownout high disabled.
 * IWDG watchdog is controlled by software.
 * WWDG watchdog is controlled by software.
 * No reset generated when entering Stop mode on core domain.
 * No reset generated when entering Standby mode on core domain.
 * High-speed IO at low VDD voltage feature disabled (VDD can exceed 2.7 V).
 * High-speed IO at low VDDIO2 voltage feature disabled (VDDIO2 can exceed 2.7 V).
 * Independent watchdog keep running in system Stop mode.
 * Independent watchdog keep running in Standby mode.
 * Bank1 and Bank2 not swapped.
> Device configuration 2: [0xC3000010]
 * SRAM2 erased when a system reset occurs.
 * BKPRAM ECC check disabled.
 * SRAM2 ECC check enabled.
 * SRAM1 erased when a system reset occurs.
 * SRAM1 ECC check enabled.
 * TrustZone is not available.
PLL enabled using internal HSI oscillator.
Requested Clock is 37.50 MHz.
Generated Clock is 37.50 MHz.
Good samples: 4 [Range 4-7].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 37.50 MHz.
Time for Connect: 0.439 s.
>|
---#TPCMD PROVISIONING TRUST_ZONE_DISABLED PROGRAM_FLASH STANDARD CLOSED
Provisioning request:
 * Trust Zone requested is disabled.
 * Program Flash is executed before moving to final state.
 * Immutable Root of Trust requested is STANDARD.
 * Finish Product State requested is Locked.
> Provisioning command set correctly.
Check device configuration:
 * Device: STM32H503RB.
 * Flash Size: 128 KB.
 * Product State: 0xED - Open.
 * Trust Zone: 0xC3 - Disabled.
 * SFSP Silicium Version: v1.4.0.
 * OTP memory is not blank. Password Hash is equal to FRB file.
> Device Configuration is correct.
Program the Firmware inside Non-secure Flash memory:
 * Time for Masserase Non-secure Flash: 0.028 s.
 * Time for Blankcheck Non-secure Flash: 0.002 s.
 * Time for Program Non-secure Flash: 0.274 s.
 * Time for Verify Readout Non-secure Flash: 0.057 s.
 * Time for Verify Checksum 32bit Non-secure Flash: 0.024 s.
> Firmware programmed correctly into Non-secure Flash memory.
Set Product State to 0x17 - Provisioning:
> Product State set to Provisioning correctly.
Send Password Hash into OTP memory:
> Password Hash is already programmed into OTP memory.
Final Product State requested is 0x72 - Closed:
> Final Product State set correctly.
Time for Provisioning: 0.963 s.
>|
```

Program Flash memory

Program OTP Hash Password into OTP memory

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
---#TPCMD DISCOVERY
Receiving Discovery response from STM32H5 device:
 * PSA auth version:                v1.0
 * Vendor ID:                       STMicroelectronics
 * SoC Class:                       0x474 - STM32H503xx                              ->  Product Id: STM32H503x
 * SoC ID:                          0x0032006F 0x30325101 0x35343637 0x00000000      ->  Unique Identifier
 * HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
 * PSA lifecycle:                   ST_LIFECYCLE_CLOSED                              ->  End Product State
 * SDA version:                     v1.2.0
 * Token Formats:                   0x0200
 * Certificate Formats:             0x0201
 * Cryptosystems:                   ST Password                                      ->  Password Mode
 * ST provisioning integrity status: 0xFFFFFFFF (Forced value, do not consider it)
 * ST HDPL1 status:                 0x11111111 (Forced value, do not consider it)
 * ST HDPL2 status:                 0x22222222 (Forced value, do not consider it)
 * ST HDPL3 status:                 0x33333333 (Forced value, do not consider it)
> Received Discovery response correctly.
Available Permission for selected device:
 * [1] ==> Full Regression
> Completed Discovery command.
Time for Discovery: 0.330 s.
>|
---#TPCMD REGRESSION FULL_REGRESSION
Opening Password from FRB file:
 * Found 1 password.
> Password imported correctly.
Sending Challenge request to the device:
> Received Challenge correctly.
Send the data to the device to perform Authentication:          Full Regression
 * Sending Password n. 1.
> Authentication successful.
Wait until device reset and reboot:
> Reset detected and device reboots now.
Receiving Discovery response from STM32H5 device:
 * PSA auth version:                v1.0
 * Vendor ID:                       STMicroelectronics
 * SoC Class:                       0x474 - STM32H503xx
 * SoC ID:                          0x0032006F 0x30325101 0x35343637 0x00000000
 * HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
 * PSA lifecycle:                   ST_LIFECYCLE_OPEN                                ->  New Product State
 * SDA version:                     v1.2.0
 * Token Formats:                   0x0200
 * Certificate Formats:             0x0201
 * Cryptosystems:                   ST Password
 * ST provisioning integrity status: 0xFFFFFFFF (Forced value, do not consider it)
 * ST HDPL1 status:                 0x11111111 (Forced value, do not consider it)
 * ST HDPL2 status:                 0x22222222 (Forced value, do not consider it)
 * ST HDPL3 status:                 0x33333333 (Forced value, do not consider it)
> Completed Discovery command.
Time for Regression: 0.710 s.
>|
```

If you perform only the Provisioning step without Regression you leave the device protected and if you try to reconnect to the device, you can see the following log:

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
Good samples: 16 [Range 0-15].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
The device probably is locked with Password or Certificate.
Current Product State is ST_LIFECYCLE_CLOSED.
Time for Connect: 0.433 s.
>|
---#TPCMD DISCOVERY
Receiving Discovery response from STM32H5 device:
 * PSA auth version:                v1.0
 * Vendor ID:                       STMicroelectronics
 * SoC Class:                       0x474 - STM32H503xx
 * SoC ID:                          0x0032006F 0x30325101 0x35343637 0x00000000
 * HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
 * PSA lifecycle:                   ST_LIFECYCLE_CLOSED
 * SDA version:                     v1.2.0
 * Token Formats:                   0x0200
 * Certificate Formats:             0x0201
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
 * Cryptosystems:                    ST Password
 * ST provisioning integrity status: 0xFFFFFFFF (Forced value, do not consider it)
 * ST HDPL1 status:                  0x11111111 (Forced value, do not consider it)
 * ST HDPL2 status:                  0x22222222 (Forced value, do not consider it)
 * ST HDPL3 status:                  0x33333333 (Forced value, do not consider it)
> Received Discovery response correctly.
Available Permission for selected device:
 * [1] ==> Full Regression
> Completed Discovery command.
Time for Discovery: 0.331 s.
>|
---#TPCMD PROVISIONING TRUST_ZONE_DISABLED PROGRAM_FLASH STANDARD CLOSED
Product State is 0x72.
Provisioning cannot be executed in current state. Skipped operation.
Time for Provisioning: 0.001 s.
>|
```

Instead, if you perform Regression, you are able to return to OPEN Product State:

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
Good samples: 16 [Range 0-15].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
The device probably is locked with Password or Certificate.
Current Product State is ST_LIFECYCLE_CLOSED.
Time for Connect: 0.432 s.
>|
---#TPCMD REGRESSION FULL_REGRESSION
Opening Password from FRB file:
 * Found 1 password.
> Password imported correctly.
Sending Challenge request to the device:
> Received Challenge correctly.
Send the data to the device to perform Authentication:
 * Sending Password n. 1.
> Authentication successful.
Wait until device reset and reboot:
> Reset detected and device reboots now.
Receiving Discovery response from STM32H5 device:
 * PSA auth version:                v1.0
 * Vendor ID:                       STMicroelectronics
 * SoC Class:                       0x474 - STM32H503xx
 * SoC ID:                          0x0032006F 0x30325101 0x35343637 0x00000000
 * HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
 * PSA lifecycle:                   ST_LIFECYCLE_OPEN
 * SDA version:                     v1.2.0
 * Token Formats:                   0x0200
 * Certificate Formats:             0x0201
 * Cryptosystems:                   ST Password
 * ST provisioning integrity status: 0xFFFFFFFF (Forced value, do not consider it)
 * ST HDPL1 status:                 0x11111111 (Forced value, do not consider it)
 * ST HDPL2 status:                 0x22222222 (Forced value, do not consider it)
 * ST HDPL3 status:                 0x33333333 (Forced value, do not consider it)
> Completed Discovery command.
Time for Regression: 0.708 s.
>|
```

As you can see from Regression command at the end the device Product State is OPEN so the device is not protected anymore.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Debug Authentication STM32H5xx non-crypto devices

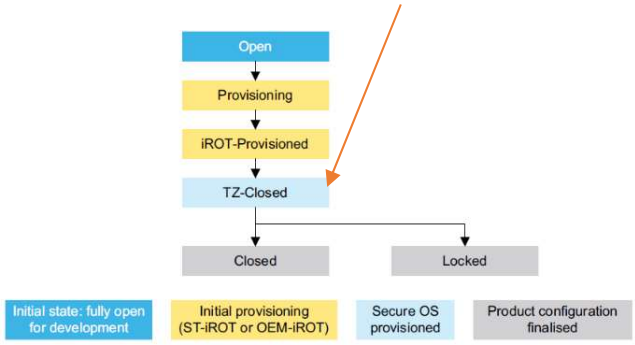
Device configuration and setup:

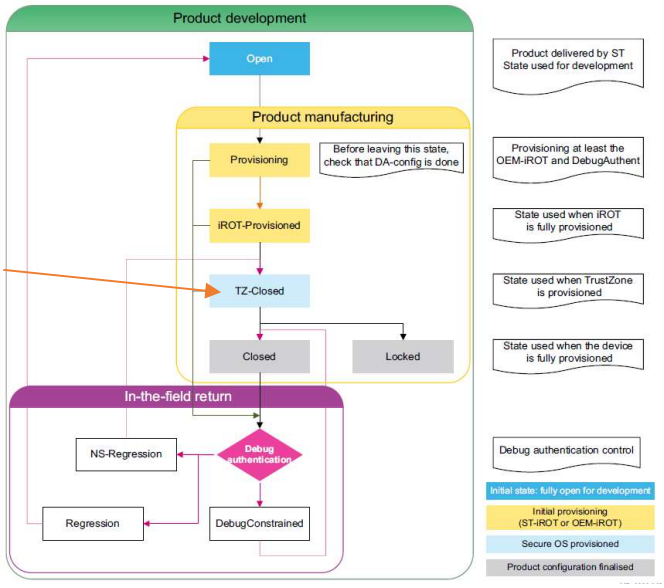
STM32H56x or STM32H52x

Trust Zone is available, can be Enabled or Disabled

Trust Zone Disabled -> **Password** for Debug Authentication
Trust Zone Enabled -> **Certificate** for Debug Authentication

Trust Zone Closed state is available only if Trust Zone is enabled





There is a **limitation** for STM32H5**6x** devices embedding SFSP2.5.0 (or previous SFSP).
On these devices, when the product state is set to CLOSED, the SOC ID cannot be readout using the #TPCMD DISCOVERY command.

Unless the SOC ID is stored before closing the device, it will not be possible to retrieve this identifier needed to generate a certificate chain specific to this device.

The SFSP version can be readout at the address: 0x0BF960CC, for SFSP2.5.0 the value is: 0x02050000.

Flow for Debug Authentication:

OPEN -> PROVISIONING -> iROT-PROVISIONED -> END STATE -> REGRESSION

Possible Product State Transition:

Is only allowed to transit from Open to other possible product states, but not the other way around.

From	To(1)							
	Open	Provisioning	iROT-Provisioned	TZ-Closed(2)	Closed	Locked	Regression	NS-Regression(2)
Open	-	OK	OK	X	X	X	X	X
Provisioning	X	-	OK	OK	OK	OK	OK (HDPL1)	X
iROT-Provisioned	X	X	-	OK	OK	OK	OK (HDPL1)	X
TZ-Closed	X	X	X	-	OK	OK	OK (HDPL1)	X
Closed	X	X	X	X	-	X	OK (HDPL1)	OK (HDPL1)
Locked	X	X	X	X	X	-	X	X
Regression	OK (HDPL0)	X	X	X	X	X	-	X
NS-Regression	X	X	X	OK (HDPL0)	X	X	X	-

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Prepare data for Debug Authentication using Workbench and FRB files:

To perform Debug Authentication with Password or Certificate (depending on Trust Zone configuration) you need to use the following files:

For Trust Zone Disabled -> Password Mode:

DA_ConfigWithPassword.obk

Offset	+0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +a +b +c +d +e +f	Equivalent ASCII characters
00000000	00 01 fd 0f 60 00 00 00 00 00 00 00 5e 9b d6 5f	` ^ _
00000010	e4 8e a1 7e 40 ae 12 6f 76 e7 c1 01 3a 73 bf 35	~ @ o v : s 5
00000020	c9 5d 48 97 5c 5d 15 75 a7 19 84 42 18 4a a4 6d	] H \ ] u B J m
00000030	81 34 11 72 7d a0 dc 9e 64 18 6b b9 90 72 89 b5	4 r } d k r
00000040	aa b4 b3 20 d2 6f ff 5e a4 5d 8e 3d 77 50 00 00	o ^] = w P
00000050	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
00000060	00 00 00 00 00 00 00 00 00 00 00 00	

This file is necessary for **Provisioning**.

password.bin

Offset	+0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +a +b +c +d +e +f	Equivalent ASCII characters
00000000	00 00 00 80 10 00 00 00 30 31 32 33 34 35 36 37	0 1 2 3 4 5 6 7
00000010	38 39 30 31 32 33 34 35	8 9 0 1 2 3 4 5

This file is necessary for **Regression**.

For Trust Zone Enabled -> Certificate Mode:

DA_Config.obk

Offset	+0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +a +b +c +d +e +f	Equivalent ASCII characters
00000000	00 01 fd 0f 60 00 00 00 00 00 00 00 98 9c 08 e9	`
00000010	69 34 b8 b0 c7 54 e7 b3 87 04 dc ca 96 11 55 18	i 4 T U
00000020	38 f2 f4 29 d3 0f c0 56 98 6a 75 7f cc 05 b0 13	8) V j u
00000030	17 77 5c 86 5a 3f 98 7d 17 78 c1 2d 7e 51 78 51	w \ Z ? } x - ~ Q x Q
00000040	a0 f4 2d d9 2d fa 6c 1d 85 9b 25 fe 77 50 00 00	- - l % w P
00000050	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
00000060	00 00 00 00 00 00 00 00 00 00 00 00	

This file is necessary for **Provisioning**.

key_3_leaf.pem

```
-----BEGIN PRIVATE KEY-----
MIGHAgEAMBMGByqGSM49AgEGCCqGSM49AwEHBG0wawIBAQQgLgplFssrUA1cEsFs
5Ccey52cPTiUL4Lp5Wx4CGxNo1ahRANCAATiW7yHWn436uMobqog4MbrVDv8p8wY
VG4xnRf2PJD4OKPQFLHYPWlhogCT1CYT9HqE1xybrXe9qrkZ4LDCii1L
-----END PRIVATE KEY-----
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Offset	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+a	+b	+c	+d	+e	+f	Equivalent ASCII characters
00000000	2d	2d	2d	2d	2d	42	45	47	49	4e	20	50	52	49	56	41	- - - - - B E G I N P R I V A
00000010	54	45	20	4b	45	59	2d	2d	2d	2d	2d	0d	0a	4d	49	47	T E K E Y - - - - - M I G
00000020	48	41	67	45	41	4d	42	4d	47	42	79	71	47	53	4d	34	H A g E A M B M G B y q G S M 4
00000030	39	41	67	45	47	43	43	71	47	53	4d	34	39	41	77	45	9 A g E G C C q G S M 4 9 A w E
00000040	48	42	47	30	77	61	77	49	42	41	51	51	67	4c	67	70	H B G 0 w a w I B A Q Q g L g p
00000050	6c	46	73	73	72	55	41	31	63	45	73	46	73	0d	0a	35	l F s s r U A 1 c E s F s 5
00000060	43	63	65	79	35	32	63	50	54	69	55	4c	34	4c	70	35	C c e y 5 2 c P T i U L 4 L p 5
00000070	57	78	34	43	47	78	4e	6f	31	61	68	52	41	4e	43	41	W x 4 C G x N o 1 a h R A N C A
00000080	41	54	69	57	37	79	48	57	6e	34	33	36	75	4d	6f	62	A T i W 7 y H W n 4 3 6 u M o b
00000090	71	6f	67	34	4d	62	72	56	44	76	38	70	38	77	59	0d	q o g 4 M b r V D v 8 p 8 w Y
000000a0	0a	56	47	34	78	6e	52	66	32	50	4a	44	34	4f	4b	50	V G 4 x n R f 2 P J D 4 O K P
000000b0	51	46	4c	48	59	50	57	6c	68	6f	67	43	54	31	43	59	Q F L H Y P W l h o g C T 1 C Y
000000c0	54	39	48	71	45	31	78	79	62	72	58	65	39	71	72	6b	T 9 H q E 1 x y b r X e 9 q r k
000000d0	5a	34	4c	44	43	69	69	31	4c	0d	0a	2d	2d	2d	2d	2d	Z 4 L D C i i 1 L - - - - -
000000e0	45	4e	44	20	50	52	49	56	41	54	45	20	4b	45	59	2d	E N D P R I V A T E K E Y -
000000f0	2d	2d	2d	2d	0d	0a											- - - -

This file is necessary for **Regression**.

cert_leaf_chain.b64

Offset	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+a	+b	+c	+d	+e	+f	Equivalent ASCII characters
00000000	41	41	41	42	41	74	51	41	41	41	41	42	41	41	45	42	A A A B A t Q A A A A B A A E B
00000010	41	51	45	41	41	41	41	41	41	41	41	41	41	41	41	41	A Q E A A A A A A A A A A A A
00000020	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	A A A A A A A A A A A A A A A A
00000030	41	41	41	41	41	41	41	41	41	41	42	33	55	41	41	41	A A A A A A A A A A B 3 U A A A
00000040	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	A A A A A A A A A A A A A A A A
00000050	58	72	6d	44	53	54	67	63	66	63	57	64	70	69	47	74	X r m D S T g c f c W d p i G t
00000060	73	70	71	76	4f	4c	47	57	6f	31	48	57	4b	78	47	66	s p q v O L G W o 1 H W K x G f
00000070	47	76	45	36	4a	56	6b	43	35	4e	5a	54	2f	70	35	30	G v E 6 J V k C 5 N Z T / p 5 0
00000080	77	59	53	6a	5a	6d	42	47	78	59	55	4e	62	65	36	2b	w Y S j Z m B G x Y U N b e 6 +
00000090	7a	72	64	53	6e	57	4e	4f	4a	37	7a	74	57	52	4b	67	z r d S n W N O J 7 z t W R K g
000000a0	2f	54	41	30	74	51	41	41	41	41	41	41	41	41	41	41	/ T A 0 t Q A A A A A A A A A A
000000b0	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	A A A A A A A A A A A A A A A A
000000c0	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	A A A A A A A A A A A A A A A A
...																	
...																	
00000300	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	A A A A A A A A A A A A A A A A
00000310	41	41	41	41	41	41	41	41	41	41	43	45	7a	73	38	78	A A A A A A A A A A C E z s 8 x
00000320	77	73	58	53	68	71	4e	52	77	64	6b	75	45	66	43	30	w s X S h q N R w d k u E f C 0
00000330	59	51	2b	77	75	65	44	39	52	52	47	70	36	4d	35	33	Y Q + w u e D 9 R R G p 6 M 5 3
00000340	44	57	4b	65	6b	31	61	76	33	6f	38	2f	44	38	63	31	D W K e k 1 a v 3 o 8 / D 8 c 1
00000350	54	36	51	69	5a	49	63	30	44	43	72	7a	70	57	2b	4a	T 6 Q i Z I c 0 D C r z p W + J
00000360	4e	45	4c	43	73	32	66	41	36	51	49	2f	78	57	65	4a	N E L C s 2 f A 6 Q I / x W e J

This file is necessary for **Regression**.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

FRB file generation for Provisioning and Regression:

Now you need to import five files inside the Workbench to create a correct FRB file for Debug Authentication.

If you want you can import only the first two files if you need to perform Provisioning with Password and only the last three if you need to perform Provisioning with Certificate.

In this example we import all of these files.

Import **DA_ConfigWithPassword.obk** as a binary (.bin extension) and edit the address position to 0xF0000000
Import **DA_Config.obk** as a binary (.bin extension) and edit the address position to 0xF0002000

[P] - PROVISIONING			
VIRTUAL SECTION NAME	VIRTUAL SECTION START ADDRESS	VIRTUAL SECTION END ADDRESS	FILE NAME
PASSWORD	0xF0000000	0xF0001FFF	DA_ConfigWithPassword.obk
CERTIFICATE	0xF0002000	0xF0003FFF	DA_Config.obk
iROT CONFIG	0xF0004000	0xF0005FFF	OEMiRoT_Config.obk *
iROT DATA	0xF0006000	0xF0007FFF	OEMiRoT_Data.obk *
uROT CONFIG	0xF0008000	0xF0009FFF	OEMuRoT_Config.obk *
uROT DATA	0xF000A000	0xF000BFFF	OEMuRoT_Data.obk *

Import **password.bin** as a binary (.bin extension) and edit the address position to 0xF0010000
Import **key_3_leaf.pem** as a binary (.bin extension) and edit the address position to 0xF0012000
Import **cert_leaf_chain.b64** as a binary (.bin extension) and edit the address position to 0xF0014000

[R] - REGRESSION			
VIRTUAL SECTION NAME	VIRTUAL SECTION START ADDRESS	VIRTUAL SECTION END ADDRESS	FILE NAME
PASSWORD	0xF0010000	0xF0011FFF	password.bin
PRIVATE KEY	0xF0012000	0xF0013FFF	key_3_leaf.pem
CERTIFICATE CHAIN	0xF0014000	0xF0015FFF	cert_leaf_chain.b64

***** As you can see on Provisioning Table, we have two lines highlighted in yellow and two lines highlighted in green.
In these four virtual regions you can import OBK file for OEMiROT flow or OEMiROT plus uROT flow.

For OEMiROT flow without uROT:

Import **OEMiRoT_Config.obk** as a binary (.bin extension) and edit the address position to 0xF0004000
Import **OEMiRoT_Data.obk** as a binary (.bin extension) and edit the address position to 0xF0006000

For OEMiROT flow with uROT:

Import **OEMiRoT_Config.obk** as a binary (.bin extension) and edit the address position to 0xF0004000
Import **OEMiRoT_Data.obk** as a binary (.bin extension) and edit the address position to 0xF0006000
Import **OEMuRoT_Config.obk** as a binary (.bin extension) and edit the address position to 0xF0008000
Import **OEMuRoT_Data.obk** as a binary (.bin extension) and edit the address position to 0xF000A000

For all of these two cases you must put the **OEMiROT_Boot.bin** into secure Flash memory at address 0x0C000000

If you want you can also add the **Flash Firmware** inside **non-secure Flash** memory:

rot_tz_s_app_enc_sign.hex -> Flash address 0x080C2000
rot_tz_ns_app_enc_sign.hex -> Flash address 0x080C8000
s_data_enc_sign.hex -> Flash address 0x08168000
ns_data_enc_sign.hex -> Flash address 0x0816A000

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale
Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255
D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021
→smh-tech.com
info@smh-tech.com

Here your FRB file from Advanced FRB Manager:

File Name	Memory Area	Target Start Address	Target End Address
rot_tz_s_app_enc_sign.hex	[F] - Non Secure Internal Flash	0x080BE000	0x080C2157
rot_tz_s_app_enc_sign.hex	[F] - Non Secure Internal Flash	0x080C3FD0	0x080C3FFF
rot_tz_ns_app_enc_sign.hex	[F] - Non Secure Internal Flash	0x080C4000	0x080CBA07
s_data_enc_sign.hex	[F] - Non Secure Internal Flash	0x08160000	0x08160157
s_data_enc_sign.hex	[F] - Non Secure Internal Flash	0x08161FD0	0x08161FFF
ns_data_enc_sign.hex	[F] - Non Secure Internal Flash	0x08162000	0x08162157
ns_data_enc_sign.hex	[F] - Non Secure Internal Flash	0x08163FD0	0x08163FFF
OEMiROT_Boot.bin	[G] - Secure Internal Flash	0x0C000000	0x0C012383
DA_ConfigWithPassword.obk	[P] - Provisioning (virtual memory)	0xF0000000	0xF000006B
DA_Config.obk	[P] - Provisioning (virtual memory)	0xF0002000	0xF000206B
OEMiRoT_Config.obk	[P] - Provisioning (virtual memory)	0xF0004000	0xF000412B
OEMiRoT_Data.obk	[P] - Provisioning (virtual memory)	0xF0006000	0xF00060CB
OEMuRoT_Config.obk	[P] - Provisioning (virtual memory)	0xF0008000	0xF000815B
OEMuRoT_Data.obk	[P] - Provisioning (virtual memory)	0xF000A000	0xF000A0CB
password.bin	[R] - Regression (virtual memory)	0xF0010000	0xF0010017
key_3_leaf.pem	[R] - Regression (virtual memory)	0xF0012000	0xF00120F5
cert_leaf_chain.b64	[R] - Regression (virtual memory)	0xF0014000	0xF001436F

Possible Flow using Provisiong and Regression:

First of all, we need to understand the possible flow for Provisioning and Regression for STM32H5x non-crypto devices. Using the `#TPCMD PROVISIONING` command, you can choose one of these possible flows:

Some possible flow for Provisiong:

Example 1:

```
Trust Zone Disabled    -> TRUST_ZONE_DISABLED
Skip Program Flash     -> SKIP_PROGRAM_FLASH
Skip OEMiROT           -> STANDARD
End State              -> CLOSED

#TPCMD PROVISIONING TRUST_ZONE_DISABLED
                    SKIP_PROGRAM_FLASH
                      STANDARD
                        CLOSED
```

Example 2:

```
Trust Zone Disabled    -> TRUST_ZONE_DISABLED
Program Flash          -> PROGRAM_FLASH
Skip OEMiROT           -> STANDARD
End State              -> CLOSED

#TPCMD PROVISIONING TRUST_ZONE_DISABLED
                    PROGRAM_FLASH
                      STANDARD
                        CLOSED
```

Example 3:

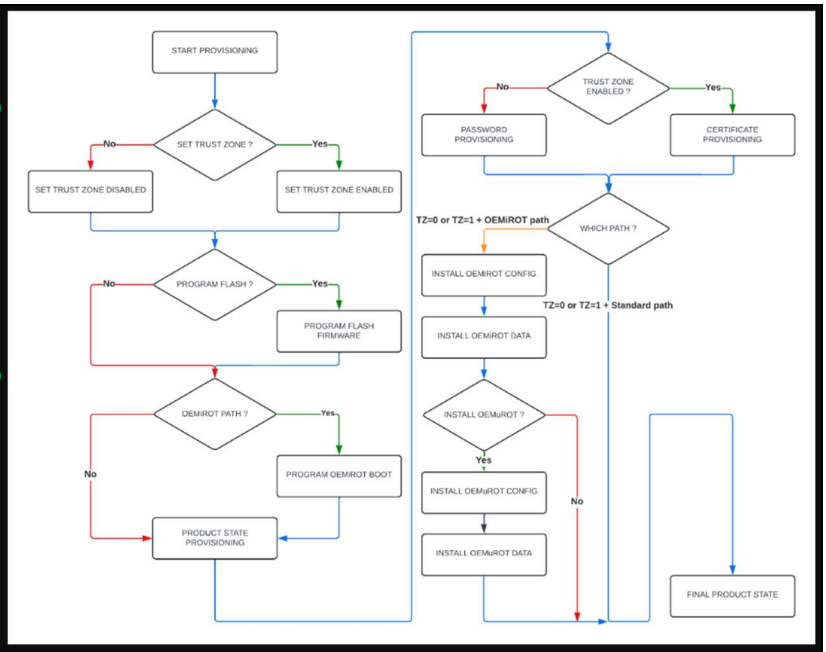
```
Trust Zone Enabled     -> TRUST_ZONE_ENABLED
Program Flash          -> PROGRAM_FLASH
Skip OEMiROT           -> STANDARD
End State              -> PROVISIONING

#TPCMD PROVISIONING TRUST_ZONE_ENABLED PROGRAM_FLASH STANDARD PROVISIONING
```

Example 4:

```
Trust Zone Enabled     -> TRUST_ZONE_ENABLED
Program Flash          -> PROGRAM_FLASH
OEMiROT + uROT         -> OEMiROT_uROT
End State              -> CLOSED

#TPCMD PROVISIONING TRUST_ZONE_ENABLED PROGRAM_FLASH OEMiROT_uROT CLOSED
```



HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Using the `#TPCMD REGRESSION` command, you can choose one of these possible flows:

```
There is a lot of possible flow for Regression:

#TPCMD REGRESSION FULL_REGRESSION

#TPCMD REGRESSION TO_TRUST_ZONE

#TPCMD REGRESSION LEVEL_3_INTRUSIVE_DEBUG
#TPCMD REGRESSION LEVEL_2_INTRUSIVE_DEBUG
#TPCMD REGRESSION LEVEL_1_INTRUSIVE_DEBUG

#TPCMD REGRESSION LEVEL_3_INTRUSIVE_NON_SECURE_DEBUG
#TPCMD REGRESSION LEVEL_2_INTRUSIVE_NON_SECURE_DEBUG
#TPCMD REGRESSION LEVEL_1_INTRUSIVE_NON_SECURE_DEBUG
```

Example using Provisiong and Regression:

Now here we can show to you an example of Provisioning and Regression.
In this case we decide to select example n.4 for Provisioning and the FULL_REGRESSION for Regression.

Project commands:

```
#TPSETSRC STM32H5xx_Debug_Authentication.frb
#TPSTART
#TPCMD CONNECT
#TPCMD PROVISIONING TRUST_ZONE_ENABLED PROGRAM_FLASH OEMiROT_uROT CLOSED
#TPCMD DISCOVERY
#TPCMD REGRESSION FULL_REGRESSION
#TPCMD DISCONNECT
#TPEND
```

Project execution:

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
Good samples: 16 [Range 0-15].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x54770002, Type: AMBA APB2 or APB3 bus.
AP[1] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[1] ROM table base address 0xE00FE000.
CPUID: 0x410FD214.
Implementer Code: 0x41 - [ARM].
Found Cortex M33 revision r0p4.
Program counter value is 0x0BF9AA02.
Cortex M33 Core halted [0.001 s].
> Device configuration 1: [0x2D30EDF8]
 * The device's product state is Open [0xED].
 * BOR level 1, the threshold level is low (around 2.1 V).
 * Brownout high disabled.
 * IWDG watchdog is controlled by software.
 * WWDG watchdog is controlled by software.
 * No reset generated when entering Stop mode on core domain.
 * No reset generated when entering Standby mode on core domain.
 * High-speed IO at low VDD voltage feature disabled (VDD can exceed 2.7 V).
 * High-speed IO at low VDDIO2 voltage feature disabled (VDDIO2 can exceed 2.7 V).
 * Independent watchdog keep running in system Stop mode.
 * Independent watchdog keep running in Standby mode.
 * Bank1 and Bank2 not swapped.
> Device configuration 2: [0xC300017C]
 * SRAM1 and SRAM3 not erased when a system reset occurs.
 * SRAM2 not erased when a system reset occurs.
 * BKPRAM ECC check disabled.
 * SRAM3 ECC check disabled.
 * SRAM2 ECC check disabled.
 * TrustZone is disabled.
PLL enabled using internal HSI oscillator.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
Requested Clock is 37.50 MHz.
Generated Clock is 37.50 MHz.
Good samples: 4 [Range 4-7].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 37.50 MHz.
Time for Connect: 0.441 s.
>|
---#TPCMD PROVISIONING TRUST_ZONE_ENABLED PROGRAM_FLASH OEMiROT_uROT CLOSED
Provisioning request:
 * Trust Zone requested is enabled.
 * Program Flash is executed before moving to final state.
 * Immutable Root of Trust requested is STiROT.
 * Finish Product State requested is Locked.
> Provisioning command set correctly.
Check device configuration:
 * Device: STM32H563ZI.
 * Flash Size: 2048 KB.
 * Product State: 0xED - Open.
 * Trust Zone: 0xC3 - Disabled.
 * SFSP Silicium Version: v2.5.0.
> Device Configuration is correct.
Set Trust Zone to enabled:
> Trust Zone is enabled correctly.
Program the Firmware inside Non-secure Flash memory:
 * Time for Masserase Non-secure Flash: 0.413 s.
 * Time for Blankcheck Non-secure Flash: 0.035 s.
 * Time for Program Non-secure Flash: 0.103 s.                    Program Flash memory
 * Time for Verify Readout Non-secure Flash: 0.030 s.
 * Time for Verify Checksum 32bit Non-secure Flash: 0.005 s.
> Firmware programmed correctly into Non-secure Flash memory.
Program OEMiROT Boot into Secure Flash memory:
 * Time for Program Secure Flash: 0.159 s.
 * Time for Verify Readout Secure Flash: 0.031 s.                 Program OEMiROT boot into Secure Flash memory
 * Time for Verify Checksum 32bit Secure Flash: 0.006 s.
> OEMiROT Boot correctly programmed into Secure Flash memory.
Set Product State to 0x17 - Provisioning:
> Product State set to Provisioning correctly.
Provision the Certificate OBK file:
 * Process the OBK file:
   * Destination address: 0x0FFD0100.
   * Size: 0x00000060.                                           Send Certificate
   * Do Encryption: 0.
 * Send the OBK file:
> Certificate OBK file provisioned successfully.
Provision the OEMiROT Config OBK file:
 * Process the OBK file:
   * Destination address: 0x0FFD0160.
   * Size: 0x00000120.
   * Do Encryption: 0.
 * Send the OBK file:
> OEMiROT Config OBK file provisioned successfully.              Program OEMiROT
Provision the OEMiROT Data OBK file:
 * Process the OBK file:
   * Destination address: 0x0FFD0280.
   * Size: 0x000000C0.
   * Do Encryption: 0.
 * Send the OBK file:
> OEMiROT Data OBK file provisioned successfully.
Provision the OEMuROT Config OBK file:
 * Process the OBK file:
   * Destination address: 0x0FFD0900.
   * Size: 0x00000150.
   * Do Encryption: 0.
 * Send the OBK file:
> OEMuROT Config OBK file provisioned successfully.              Program OEMuROT
Provision the OEMuROT Data OBK file:
 * Process the OBK file:
   * Destination address: 0x0FFD0A50.
   * Size: 0x000000C0.
   * Do Encryption: 0.
 * Send the OBK file:
> OEMuROT Data OBK file provisioned successfully.
Final Product State requested is 0x72 - Closed:
> Final Product State set correctly.
Time for Provisioning: 1.503 s.
>|
---#TPCMD DISCOVERY
Receiving Discovery response from STM32H5 device:
 * PSA auth version:                v1.0
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
 * Vendor ID:                        STMicroelectronics
 * SoC Class:                        0x484 - STM32H5xxxx                          ->  Product Id: STM32H5xxx
 * SoC ID:                           0x00000000 0x00000000 0x00000000 0x00000000  ->  Unique Identifier
 * HW permission mask:               0x00005077 0x00000000 0x00000000 0x00000000
 * PSA lifecycle:                    ST_LIFECYCLE_CLOSED                          ->  End Product State
 * SDA version:                      v2.4.0
 * Token Formats:                    0x0200
 * Certificate Formats:              0x0201
 * Cryptosystems:                    Ecdsa-P256 SHA256                            ->  Certificate Mode
 * ST provisioning integrity status: 0xEAEAEAEA
 * ST HDPL1 status:                  0xFFFFFFFF
 * ST HDPL2 status:                  0xFFFFFFFF
 * ST HDPL3 status:                  0xFFFFFFFF
> Received Discovery response correctly.
Available Permission for selected device:
 * [1] ==> Full Regression
 * [2] ==> To Trust Zone Regression
 * [3] ==> Level 3 Intrusive Debug
 * [4] ==> Level 2 Intrusive Debug
 * [5] ==> Level 1 Intrusive Debug
 * [6] ==> Level 3 Intrusive Non Secure Debug
 * [7] ==> Level 2 Intrusive Non Secure Debug
 * [8] ==> Level 1 Intrusive Non Secure Debug
> Completed Discovery command.
Time for Discovery: 0.330 s.
>|
---#TPCMD REGRESSION FULL_REGRESSION
ST provisioning integrity status: 0xEAEAEAEA.
Import Private Key from FRB file:
 * Importing EC P-256 key.
> Private Key imported correctly.
Import Trust of Chain from FRB file:
 * Found 3 certificates.
> Trust of Chain imported correctly.
Sending Challenge request to the device:
> Received Challenge correctly.
Sign the Token based on Challenge received:
> Signing Token completed successfully.
Send the data to the device to perform Authentication:
 * Sending Certificate n. 1.
 * Sending Certificate n. 2.
 * Sending Certificate n. 3.
 * Sending Authentication Token.
> Authentication completed successfully.
Wait until device reset and reboot:
> Reset detected and device reboots now.
Receiving Discovery response from STM32H5 device:
 * PSA auth version:                v1.0
 * Vendor ID:                       STMicroelectronics
 * SoC Class:                       0x484 - STM32H5xxxx
 * SoC ID:                          0x00000000 0x00000000 0x00000000 0x00000000
 * HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
 * PSA lifecycle:                   ST_LIFECYCLE_OPEN
 * SDA version:                     v2.4.0
 * Token Formats:                   0x0200
 * Certificate Formats:             0x0201
 * Cryptosystems:                   ST Password
 * ST provisioning integrity status: 0xF5F5F5F5
 * ST HDPL1 status:                 0xFFFFFFFF
 * ST HDPL2 status:                 0xFFFFFFFF
 * ST HDPL3 status:                 0xFFFFFFFF
> Completed Discovery command.
Time for Regression: 2.399 s.
>|
```

If you perform only the Provisioning step without Regression you leave the device protected and if you try to reconnect to the device, you can see the following log:

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
Good samples: 16 [Range 0-15].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
The device probably is locked with Password or Certificate.
Current Product State is ST_LIFECYCLE_CLOSED.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
Time for Connect: 0.433 s.
>|
---#TPCMD DISCOVERY
Receiving Discovery response from STM32H5 device:
 * PSA auth version:                v1.0
 * Vendor ID:                       STMicroelectronics
 * SoC Class:                       0x484 - STM32H5xxxx
 * SoC ID:                          0x00000000 0x00000000 0x00000000 0x00000000
 * HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
 * PSA lifecycle:                   ST_LIFECYCLE_CLOSED
 * SDA version:                     v2.4.0
 * Token Formats:                   0x0200
 * Certificate Formats:             0x0201
 * Cryptosystems:                   Ecdsa-P256 SHA256
 * ST provisioning integrity status: 0xEAEAEAEA
 * ST HDPL1 status:                 0xFFFFFFFF
 * ST HDPL2 status:                 0xFFFFFFFF
 * ST HDPL3 status:                 0xFFFFFFFF
> Received Discovery response correctly.
Available Permission for selected device:
 * [1] ==> Full Regression
 * [2] ==> To Trust Zone Regression
 * [3] ==> Level 3 Intrusive Debug
 * [4] ==> Level 2 Intrusive Debug
 * [5] ==> Level 1 Intrusive Debug
 * [6] ==> Level 3 Intrusive Non Secure Debug
 * [7] ==> Level 2 Intrusive Non Secure Debug
 * [8] ==> Level 1 Intrusive Non Secure Debug
> Completed Discovery command.
Time for Discovery: 0.331 s.
>|
---#TPCMD PROVISIONING TRUST_ZONE_ENABLED PROGRAM_FLASH OEMiROT_uROT CLOSED
Product State is 0x72.
Provisioning cannot be executed in current state. Skipped operation.
Time for Provisioning: 0.001 s.
>|
```

Instead, if you perform Regression, you are able to return to OPEN Product State:

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
Good samples: 16 [Range 0-15].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
The device probably is locked with Password or Certificate.
Current Product State is ST_LIFECYCLE_CLOSED.
Time for Connect: 0.432 s.
>|
---#TPCMD REGRESSION FULL_REGRESSION
ST provisioning integrity status: 0xEAEAEAEA.
Import Private Key from FRB file:
 * Importing EC P-256 key.
> Private Key imported correctly.
Import Trust of Chain from FRB file:
 * Found 3 certificates.
> Trust of Chain imported correctly.
Sending Challenge request to the device:
> Received Challenge correctly.
Sign the Token based on Challenge received:
> Signing Token completed successfully.
Send the data to the device to perform Authentication:
 * Sending Certificate n. 1.
 * Sending Certificate n. 2.
 * Sending Certificate n. 3.
 * Sending Authentication Token.
> Authentication completed successfully.
Wait until device reset and reboot:
> Reset detected and device reboots now.
Receiving Discovery response from STM32H5 device:
 * PSA auth version:                v1.0
 * Vendor ID:                       STMicroelectronics
 * SoC Class:                       0x484 - STM32H5xxxx
 * SoC ID:                          0x00000000 0x00000000 0x00000000 0x00000000
 * HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
 * PSA lifecycle:                   ST_LIFECYCLE_OPEN
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
 * SDA version:                     v2.4.0
 * Token Formats:                   0x0200
 * Certificate Formats:             0x0201
 * Cryptosystems:                   ST Password
 * ST provisioning integrity status: 0xF5F5F5F5
 * ST HDPL1 status:                 0xFFFFFFFF
 * ST HDPL2 status:                 0xFFFFFFFF
 * ST HDPL3 status:                 0xFFFFFFFF
> Completed Discovery command.
Time for Regression: 2.399 s.
>|
```

As you can see from Regression command at the end the device Product State is OPEN so the device is not protected anymore.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Debug Authentication STM32H5xx crypto devices

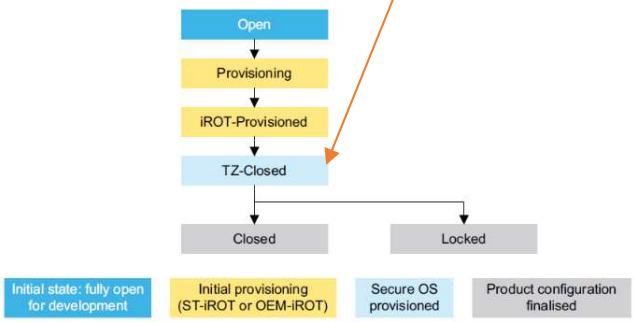
Device configuration and setup:

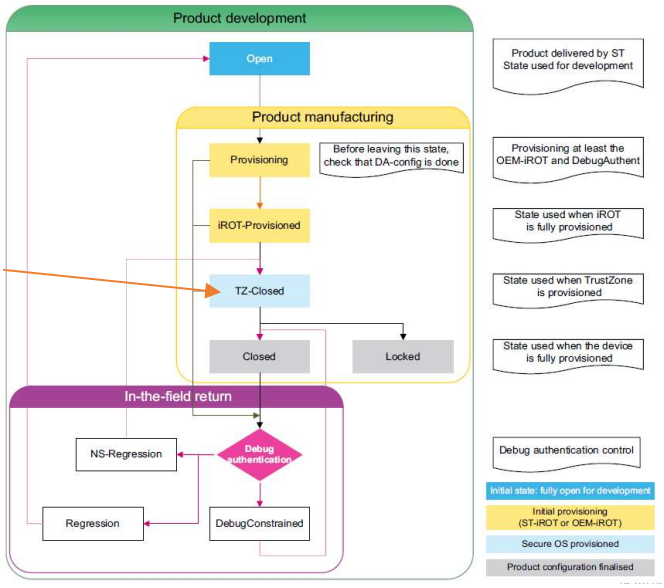
STM32H57x or STM32H53x

Trust Zone is available, can be Enabled or Disabled

Trust Zone Disabled -> **Password** for Debug Authentication
Trust Zone Enabled -> **Certificate** for Debug Authentication

Trust Zone Closed state is available only if Trust Zone is enabled





Flow for Debug Authentication:

OPEN -> PROVISIONING -> iROT-PROVISIONED -> END STATE -> REGRESSION

Possible Product State Transition:

Is only allowed to transit from Open to other possible product states, but not the other way around.

From	To[1]							
	Open	Provisioning	iROT-Provisioned	TZ-Closed[2]	Closed	Locked	Regression	NS-Regression[2]
Open	-	OK	OK	X	X	X	X	X
Provisioning	X	-	OK	OK	OK	OK	OK (HDPL1)	X
iROT-Provisioned	X	X	-	OK	OK	OK	OK (HDPL1)	X
TZ-Closed	X	X	X	-	OK	OK	OK (HDPL1)	X
Closed	X	X	X	X	-	X	OK (HDPL1)	OK (HDPL1)
Locked	X	X	X	X	X	-	X	X
Regression	OK (HDPL0)	X	X	X	X	X	-	X
NS-Regression	X	X	X	OK (HDPL0)	X	X	X	-

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Prepare data for Debug Authentication using Workbench and FRB files:

To perform Debug Authentication with Password or Certificate (depending on Trust Zone configuration) you need to use the following files:

For Trust Zone Disabled -> Password Mode:

DA_ConfigWithPassword.obk

Offset	+0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +a +b +c +d +e +f	Equivalent ASCII characters
00000000	00 01 fd 0f 60 00 00 00 01 00 00 00 5e 9b d6 5f	` ^ _
00000010	e4 8e a1 7e 40 ae 12 6f 76 e7 c1 01 3a 73 bf 35	~ @ o v : s 5
00000020	c9 5d 48 97 5c 5d 15 75 a7 19 84 42 18 4a a4 6d	] H \ ] u B J m
00000030	81 34 11 72 7d a0 dc 9e 64 18 6b b9 90 72 89 b5	4 r } d k r
00000040	aa b4 b3 20 d2 6f ff 5e a4 5d 8e 3d 77 50 00 00	o ^] = w P
00000050	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
00000060	00 00 00 00 00 00 00 00 00 00 00 00	

This file is necessary for **Provisioning**.

password.bin

Offset	+0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +a +b +c +d +e +f	Equivalent ASCII characters
00000000	00 00 00 80 10 00 00 00 30 31 32 33 34 35 36 37	0 1 2 3 4 5 6 7
00000010	38 39 30 31 32 33 34 35	8 9 0 1 2 3 4 5

This file is necessary for **Regression**.

For Trust Zone Enabled -> Certificate Mode:

DA_Config.obk

Offset	+0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +a +b +c +d +e +f	Equivalent ASCII characters
00000000	00 01 fd 0f 60 00 00 00 01 00 00 00 98 9c 08 e9	`
00000010	69 34 b8 b0 c7 54 e7 b3 87 04 dc ca 96 11 55 18	i 4 T U
00000020	38 f2 f4 29 d3 0f c0 56 98 6a 75 7f cc 05 b0 13	8) V j u
00000030	17 77 5c 86 5a 3f 98 7d 17 78 c1 2d 7e 51 78 51	w \ Z ? } x - ~ Q x Q
00000040	a0 f4 2d d9 2d fa 6c 1d 85 9b 25 fe 77 50 00 00	- - l % w P
00000050	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
00000060	00 00 00 00 00 00 00 00 00 00 00 00	

This file is necessary for **Provisioning**.

key_3_leaf.pem

```
-----BEGIN PRIVATE KEY-----
MIGHAgEAMBMGByqGSM49AgEGCCqGSM49AwEHBG0wawIBAQQgLgplFssrUA1cEsFs
5Ccey52cPTiUL4Lp5Wx4CGxNo1ahRANCAATiW7yHWn436uMobqog4MbrVDv8p8wY
VG4xnRf2PJD4OKPQFLHYPWlhogCT1CYT9HqE1xybrXe9qrkZ4LDCii1L
-----END PRIVATE KEY-----
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Offset	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+a	+b	+c	+d	+e	+f	Equivalent ASCII characters
00000000	2d	2d	2d	2d	2d	42	45	47	49	4e	20	50	52	49	56	41	- - - - - B E G I N P R I V A
00000010	54	45	20	4b	45	59	2d	2d	2d	2d	2d	0d	0a	4d	49	47	T E K E Y - - - - - M I G
00000020	48	41	67	45	41	4d	42	4d	47	42	79	71	47	53	4d	34	H A g E A M B M G B y q G S M 4
00000030	39	41	67	45	47	43	43	71	47	53	4d	34	39	41	77	45	9 A g E G C C q G S M 4 9 A w E
00000040	48	42	47	30	77	61	77	49	42	41	51	51	67	4c	67	70	H B G 0 w a w I B A Q Q g L g p
00000050	6c	46	73	73	72	55	41	31	63	45	73	46	73	0d	0a	35	l F s s r U A 1 c E s F s 5
00000060	43	63	65	79	35	32	63	50	54	69	55	4c	34	4c	70	35	C c e y 5 2 c P T i U L 4 L p 5
00000070	57	78	34	43	47	78	4e	6f	31	61	68	52	41	4e	43	41	W x 4 C G x N o 1 a h R A N C A
00000080	41	54	69	57	37	79	48	57	6e	34	33	36	75	4d	6f	62	A T i W 7 y H W n 4 3 6 u M o b
00000090	71	6f	67	34	4d	62	72	56	44	76	38	70	38	77	59	0d	q o g 4 M b r V D v 8 p 8 w Y
000000a0	0a	56	47	34	78	6e	52	66	32	50	4a	44	34	4f	4b	50	V G 4 x n R f 2 P J D 4 O K P
000000b0	51	46	4c	48	59	50	57	6c	68	6f	67	43	54	31	43	59	Q F L H Y P W l h o g C T 1 C Y
000000c0	54	39	48	71	45	31	78	79	62	72	58	65	39	71	72	6b	T 9 H q E 1 x y b r X e 9 q r k
000000d0	5a	34	4c	44	43	69	69	31	4c	0d	0a	2d	2d	2d	2d	2d	Z 4 L D C i i 1 L - - - - -
000000e0	45	4e	44	20	50	52	49	56	41	54	45	20	4b	45	59	2d	E N D P R I V A T E K E Y -
000000f0	2d	2d	2d	2d	0d	0a											- - - -

This file is necessary for **Regression**.

cert_leaf_chain.b64

Offset	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+a	+b	+c	+d	+e	+f	Equivalent ASCII characters
00000000	41	41	41	42	41	74	51	41	41	41	41	42	41	41	45	42	A A A B A t Q A A A A B A A E B
00000010	41	51	45	41	41	41	41	41	41	41	41	41	41	41	41	41	A Q E A A A A A A A A A A A A
00000020	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	A A A A A A A A A A A A A A A A
00000030	41	41	41	41	41	41	41	41	41	41	42	33	55	41	41	41	A A A A A A A A A A B 3 U A A A
00000040	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	A A A A A A A A A A A A A A A A
00000050	58	72	6d	44	53	54	67	63	66	63	57	64	70	69	47	74	X r m D S T g c f c W d p i G t
00000060	73	70	71	76	4f	4c	47	57	6f	31	48	57	4b	78	47	66	s p q v O L G W o 1 H W K x G f
00000070	47	76	45	36	4a	56	6b	43	35	4e	5a	54	2f	70	35	30	G v E 6 J V k C 5 N Z T / p 5 0
00000080	77	59	53	6a	5a	6d	42	47	78	59	55	4e	62	65	36	2b	w Y S j Z m B G x Y U N b e 6 +
00000090	7a	72	64	53	6e	57	4e	4f	4a	37	7a	74	57	52	4b	67	z r d S n W N O J 7 z t W R K g
000000a0	2f	54	41	30	74	51	41	41	41	41	41	41	41	41	41	41	/ T A 0 t Q A A A A A A A A A A
000000b0	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	A A A A A A A A A A A A A A A A
000000c0	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	A A A A A A A A A A A A A A A A
...																	
...																	
00000300	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	A A A A A A A A A A A A A A A A
00000310	41	41	41	41	41	41	41	41	41	41	43	45	7a	73	38	78	A A A A A A A A A A C E z s 8 x
00000320	77	73	58	53	68	71	4e	52	77	64	6b	75	45	66	43	30	w s X S h q N R w d k u E f C 0
00000330	59	51	2b	77	75	65	44	39	52	52	47	70	36	4d	35	33	Y Q + w u e D 9 R R G p 6 M 5 3
00000340	44	57	4b	65	6b	31	61	76	33	6f	38	2f	44	38	63	31	D W K e k 1 a v 3 o 8 / D 8 c 1
00000350	54	36	51	69	5a	49	63	30	44	43	72	7a	70	57	2b	4a	T 6 Q i Z I c 0 D C r z p W + J
00000360	4e	45	4c	43	73	32	66	41	36	51	49	2f	78	57	65	4a	N E L C s 2 f A 6 Q I / x W e J

This file is necessary for **Regression**.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

FRB file generation for Provisioning and Regression:

Now you need to import five files inside the Workbench to create a correct FRB file for Debug Authentication.

If you want you can import only the first two files if you need to perform Provisioning with Password and only the last three if you need to perform Provisioning with Certificate.

In this example we import all of these files.

Import DA_ConfigWithPassword.obk as a binary (.bin extension) and edit the address position to 0xF0000000
Import DA_Config.obk as a binary (.bin extension) and edit the address position to 0xF0002000

[P] - PROVISIONING			
VIRTUAL SECTION NAME	VIRTUAL SECTION START ADDRESS	VIRTUAL SECTION END ADDRESS	FILE NAME
PASSWORD	0xF0000000	0xF0001FFF	DA_ConfigWithPassword.obk
CERTIFICATE	0xF0002000	0xF0003FFF	DA_Config.obk
iROT CONFIG	0xF0004000	0xF0005FFF	iRoT_Config.obk *
iROT DATA	0xF0006000	0xF0007FFF	iRoT_Data.obk *
uROT CONFIG	0xF0008000	0xF0009FFF	uRoT_Config.obk *
uROT DATA	0xF000A000	0xF000BFFF	uRoT_Data.obk *

Import password.bin as a binary (.bin extension) and edit the address position to 0xF0010000
Import key_3_leaf.pem as a binary (.bin extension) and edit the address position to 0xF0012000
Import cert_leaf_chain.b64 as a binary (.bin extension) and edit the address position to 0xF0014000

[R] - REGRESSION			
VIRTUAL SECTION NAME	VIRTUAL SECTION START ADDRESS	VIRTUAL SECTION END ADDRESS	FILE NAME
PASSWORD	0xF0010000	0xF0011FFF	password.bin
PRIVATE KEY	0xF0012000	0xF0013FFF	key_3_leaf.pem
CERTIFICATE CHAIN	0xF0014000	0xF0015FFF	cert_leaf_chain.b64

* As you can see on Provisioning Table, we have two lines highlighted in yellow and two lines highlighted in green.
In these four virtual regions you can import OBK file for OEMiROT flow, OEMiROT plus uROT flow, STiROT flow or STiROT plus uROT flow.

For OEMiROT flow without uROT:

Import OEMiRoT_Config.obk as a binary (.bin extension) and edit the address position to 0xF0004000
Import OEMiRoT_Data.obk as a binary (.bin extension) and edit the address position to 0xF0006000

For OEMiROT flow with uROT:

Import OEMiRoT_Config.obk as a binary (.bin extension) and edit the address position to 0xF0004000
Import OEMiRoT_Data.obk as a binary (.bin extension) and edit the address position to 0xF0006000
Import OEMuRoT_Config.obk as a binary (.bin extension) and edit the address position to 0xF0008000
Import OEMuRoT_Data.obk as a binary (.bin extension) and edit the address position to 0xF000A000

For all of these two cases you must put the OEMiROT_Boot.bin into secure Flash memory at address 0x0C000000

For STiROT flow without uROT:

Import STiRoT_Config.obk as a binary (.bin extension) and edit the address position to 0xF0004000
Import STiRoT_Data.obk as a binary (.bin extension) and edit the address position to 0xF0006000

UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

For STiROT flow with uROT:

Import **STiRoT_Config.obk** as a binary (.bin extension) and edit the address position to 0xF0004000
Import **STiRoT_Data.obk** as a binary (.bin extension) and edit the address position to 0xF0006000
Import **STuRoT_Config.obk** as a binary (.bin extension) and edit the address position to 0xF0008000
Import **STuRoT_Data.obk** as a binary (.bin extension) and edit the address position to 0xF000A000

If you want you can also add the **Flash Firmware** inside **non-secure Flash** memory.

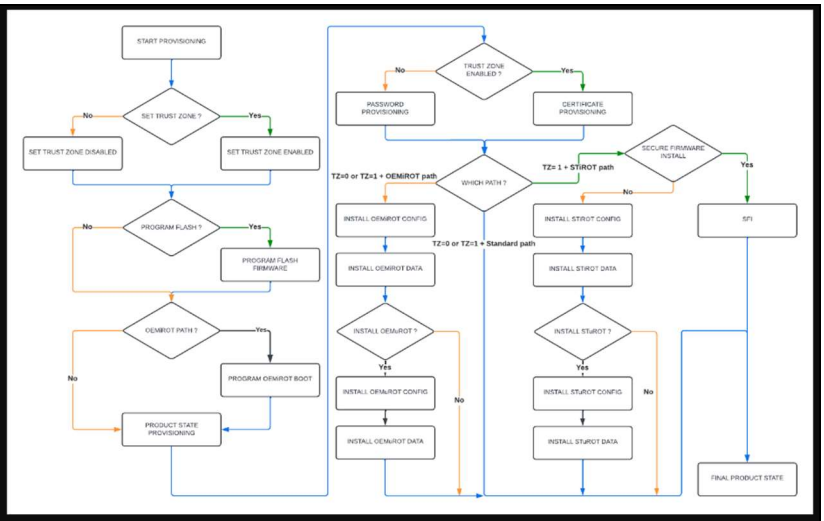
Here your FRB file from Advanced FRB Manager:

File Name	Memory Area	Target Start Address	Target End Address
s_data_enc_sign.hex	[F] - Non Secure Internal Flash	0x08160000	0x08160157
s_data_enc_sign.hex	[F] - Non Secure Internal Flash	0x08161FD0	0x08161FFF
ns_data_enc_sign.hex	[F] - Non Secure Internal Flash	0x08162000	0x08162157
ns_data_enc_sign.hex	[F] - Non Secure Internal Flash	0x08163FD0	0x08163FFF
OEMuRoT_Config.hex	[F] - Non Secure Internal Flash	0x081EE000	0x081EE14F
DA_ConfigWithPassword.obk	[P] - Provisioning (virtual memory)	0xF0000000	0xF000006B
DA_Config.obk	[P] - Provisioning (virtual memory)	0xF0002000	0xF000206B
STiRoT_Config.obk	[P] - Provisioning (virtual memory)	0xF0004000	0xF000410B
STiRoT_Data.obk	[P] - Provisioning (virtual memory)	0xF0006000	0xF000606B
OEMuRoT_Config.obk	[P] - Provisioning (virtual memory)	0xF0008000	0xF000815B
OEMuRoT_Data.obk	[P] - Provisioning (virtual memory)	0xF000A000	0xF000A0CB
password.bin	[R] - Regression (virtual memory)	0xF0010000	0xF0010017
key_3_leaf.pem	[R] - Regression (virtual memory)	0xF0012000	0xF00120F5
cert_leaf_chain.b64	[R] - Regression (virtual memory)	0xF0014000	0xF001436F

Possible Flow using Provisiong and Regression:

First of all, we need to understand the possible flow for Provisioning and Regression for STM32H5x crypto devices.
Using the `#TPCMD PROVISIONING` command, you can choose one of these possible flows:

Some possible flow for Provisiong:




```
Example 1:

Trust Zone Disabled    -> TRUST_ZONE_DISABLED
Skip Program Flash     -> SKIP_PROGRAM_FLASH
Skip OEMiROT           -> STANDARD
End State              -> CLOSED

#TPCMD PROVISIONING TRUST_ZONE_DISABLED
                    SKIP_PROGRAM_FLASH
                      STANDARD
                        CLOSED


Example 2:

Trust Zone Disabled    -> TRUST_ZONE_DISABLED
Program Flash          -> PROGRAM_FLASH
Skip OEMiROT           -> STANDARD
End State              -> CLOSED

#TPCMD PROVISIONING TRUST_ZONE_DISABLED
                    PROGRAM_FLASH
                      STANDARD
                        CLOSED


Example 3:

Trust Zone Enabled     -> TRUST_ZONE_ENABLED
Program Flash          -> PROGRAM_FLASH
Skip OEMiROT           -> STANDARD
End State              -> PROVISIONING

#TPCMD PROVISIONING TRUST_ZONE_ENABLED PROGRAM_FLASH STANDARD PROVISIONING

Example 4:

Trust Zone Enabled     -> TRUST_ZONE_ENABLED
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

```
Program Flash          -> PROGRAM_FLASH
OEMiROT + uROT         -> OEMiROT_uROT
End State              -> CLOSED

#TPCMD PROVISIONING TRUST_ZONE_ENABLED PROGRAM_FLASH OEMiROT_uROT CLOSED

Example 5:

Trust Zone Enabled     -> TRUST_ZONE_ENABLED
Program Flash          -> PROGRAM_FLASH
STiROT                 -> STiROT
End State              -> CLOSED

#TPCMD PROVISIONING TRUST_ZONE_ENABLED PROGRAM_FLASH STiROT CLOSED

Example 6:

Trust Zone Enabled     -> TRUST_ZONE_ENABLED
Program Flash          -> PROGRAM_FLASH
STiROT + uROT          -> STiROT_uROT
End State              -> CLOSED

#TPCMD PROVISIONING TRUST_ZONE_ENABLED PROGRAM_FLASH STiROT_uROT CLOSED
```

Using the `#TPCMD REGRESSION` command, you can choose one of these possible flows:

```
There is a lot of possible flow for Regression:

#TPCMD REGRESSION FULL_REGRESSION

#TPCMD REGRESSION TO_TRUST_ZONE

#TPCMD REGRESSION LEVEL_3_INTRUSIVE_DEBUG
#TPCMD REGRESSION LEVEL_2_INTRUSIVE_DEBUG
#TPCMD REGRESSION LEVEL_1_INTRUSIVE_DEBUG

#TPCMD REGRESSION LEVEL_3_INTRUSIVE_NON_SECURE_DEBUG
#TPCMD REGRESSION LEVEL_2_INTRUSIVE_NON_SECURE_DEBUG
#TPCMD REGRESSION LEVEL_1_INTRUSIVE_NON_SECURE_DEBUG
```

Example using Provisiong and Regression:

Now here we can show to you an example of Provisioning and Regression.
In this case we decide to select example n.6 for Provisioning and the FULL_REGRESSION for Regression.

Project commands:

```
#TPSETSRC STM32H5xx_Debug_Authentication.frb
#TPSTART
#TPCMD CONNECT
#TPCMD PROVISIONING TRUST_ZONE_ENABLED PROGRAM_FLASH STiROT_uROT CLOSED
#TPCMD DISCOVERY
#TPCMD REGRESSION FULL_REGRESSION
#TPCMD DISCONNECT
#TPEND
```

Project execution:

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 1.00 MHz.
Trying Hot Plug connect procedure.
Good samples: 16 [Range 0-15].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 1.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x54770002, Type: AMBA APB2 or APB3 bus.
AP[1] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[1] ROM table base address 0xE00FE000.
CPUID: 0x410FD214.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
Implementer Code: 0x41 - [ARM].
Found Cortex M33 revision r0p4.
Program counter value is 0xEFFFFFFE.
Cortex M33 Core halted [0.002 s].
> Device configuration 1: [0x30F0EDF8]
 * The device's product state is Open [0xED].
 * BOR level 1, the threshold level is low (around 2.1 V).
 * Brownout high disabled.
 * IWDG watchdog is controlled by software.
 * WWDG watchdog is controlled by software.
 * No reset generated when entering Stop mode on core domain.
 * No reset generated when entering Standby mode on core domain.
 * High-speed IO at low VDD voltage feature disabled (VDD can exceed 2.7 V).
 * High-speed IO at low VDDIO2 voltage feature disabled (VDDIO2 can exceed 2.7 V).
 * Independent watchdog keep running in system Stop mode.
 * Independent watchdog keep running in Standby mode.
 * Bank1 and Bank2 not swapped.
> Device configuration 2: [0xC3000034]
 * SRAM1 and SRAM3 not erased when a system reset occurs.
 * SRAM2 erased when a system reset occurs.
 * BKPRAM ECC check disabled.
 * SRAM3 ECC check disabled.
 * SRAM2 ECC check enabled.
 * TrustZone is disabled.
 * UBE OEM-iRoT (user flash) selected. In Open Product State this value selects bootloader.
PLL enabled using internal HSI oscillator.
Requested Clock is 37.50 MHz.
Generated Clock is 37.50 MHz.
Good samples: 4 [Range 3-6].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 37.50 MHz.
Time for Connect: 0.472 s.
>|
---#TPCMD PROVISIONING TRUST_ZONE_ENABLED PROGRAM_FLASH STiROT_uROT CLOSED
Provisioning request:
 * Trust Zone requested is enabled.
 * Program Flash is executed before moving to final state.
 * Immutable Root of Trust requested is STiROT.
 * Finish Product State requested is Locked.
> Provisioning command set correctly.
Check device configuration:
 * Device: STM32H573ZI.
 * Flash Size: 2048 KB.
 * Product State: 0xED - Open.
 * Trust Zone: 0xC3 - Disabled.
 * SFSP Silicium Version: v2.5.0.
> Device Configuration is correct.
Set Trust Zone to enabled:
> Trust Zone is enabled correctly.
Set Boot UBE to ST-iRot:
> Boot UBE is already ST-iRot.
Program the Firmware inside Non-secure Flash memory:
 * Time for Masserase Non-secure Flash: 0.400 s.
 * Time for Blankcheck Non-secure Flash: 0.035 s.
 * Time for Program Non-secure Flash: 0.012 s. .                    Program Flash memory
 * Time for Verify Readout Non-secure Flash: 0.011 s.
 * Time for Verify Checksum 32bit Non-secure Flash: 0.001 s.
> Firmware programmed correctly into Non-secure Flash memory.
Set Product State to 0x17 - Provisioning:
> Product State set to Provisioning correctly.
Provision the Certificate OBK file:
 * Process the OBK file:
   * Destination address: 0x0FFD0100.
   * Size: 0x00000060.                                             Send Certificate
   * Do Encryption: 1.
 * Send the OBK file:
> Certificate OBK file provisioned successfully.
Provision the STiROT Config OBK file:
 * Process the OBK file:
   * Destination address: 0x0FFD0200.
   * Size: 0x00000100.
   * Do Encryption: 1.
 * Send the OBK file:
> STiROT Config OBK file provisioned successfully.                 Program STiROT
Provision the STiROT Data OBK file:
 * Process the OBK file:
   * Destination address: 0x0FFD0400.
   * Size: 0x00000060.
   * Do Encryption: 1.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
 * Send the OBK file:
> STiROT Data OBK file provisioned successfully.
Provision the STuROT Config OBK file:
 * Process the OBK file:
   * Destination address: 0x0FFD0900.
   * Size: 0x00000150.
   * Do Encryption: 1.
 * Send the OBK file:
> STuROT Config OBK file provisioned successfully.                Program STuROT
Provision the STuROT Data OBK file:
 * Process the OBK file:
   * Destination address: 0x0FFD0A50.
   * Size: 0x000000C0.
   * Do Encryption: 0.
 * Send the OBK file:
> STuROT Data OBK file provisioned successfully.
Final Product State requested is 0x72 - Closed:
> Final Product State set correctly.
Time for Provisioning: 1.219 s.
>|
---#TPCMD DISCOVERY
Receiving Discovery response from STM32H5 device:
 * PSA auth version:                v1.0
 * Vendor ID:                       STMicroelectronics
 * SoC Class:                       0x484 - STM32H5xxxx
 * SoC ID:                          0x00630052 0x33325116 0x38363236 0x00000000
 * HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
 * PSA lifecycle:                   ST_LIFECYCLE_CLOSED
 * SDA version:                     v2.4.0
 * Token Formats:                   0x0200
 * Certificate Formats:             0x0201
 * Cryptosystems:                   Ecdsa-P256 SHA256
 * ST provisioning integrity status: 0xEAEAEAEA
 * ST HDPL1 status:                 0x00004707
 * ST HDPL2 status:                 0xFFFFFFFF
 * ST HDPL3 status:                 0xFFFFFFFF
> Received Discovery response correctly.
Available Permission for selected device:
 * [1] ==> Full Regression
 * [2] ==> To Trust Zone Regression
 * [3] ==> Level 3 Intrusive Debug
 * [4] ==> Level 2 Intrusive Debug
 * [5] ==> Level 1 Intrusive Debug
 * [6] ==> Level 3 Intrusive Non Secure Debug
 * [7] ==> Level 2 Intrusive Non Secure Debug
 * [8] ==> Level 1 Intrusive Non Secure Debug
> Completed Discovery command.
Time for Discovery: 0.352 s.
>|
---#TPCMD REGRESSION FULL_REGRESSION
ST provisioning integrity status: 0xEAEAEAEA.
Import Private Key from FRB file:
 * Importing EC P-256 key.
> Private Key imported correctly.
Import Trust of Chain from FRB file:
 * Found 3 certificates.
> Trust of Chain imported correctly.
Sending Challenge request to the device:
> Received Challenge correctly.
Sign the Token based on Challenge received:
> Signing Token completed successfully.
Send the data to the device to perform Authentication:
 * Sending Certificate n. 1.
 * Sending Certificate n. 2.
 * Sending Certificate n. 3.
 * Sending Authentication Token.
> Authentication completed successfully.
Wait until device reset and reboot:
> Reset detected and device reboots now.
Receiving Discovery response from STM32H5 device:
 * PSA auth version:                v1.0
 * Vendor ID:                       STMicroelectronics
 * SoC Class:                       0x484 - STM32H5xxxx
 * SoC ID:                          0x00630052 0x33325116 0x38363236 0x00000000
 * HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
 * PSA lifecycle:                   ST_LIFECYCLE_OPEN
 * SDA version:                     v2.4.0
 * Token Formats:                   0x0200
 * Certificate Formats:             0x0201
 * Cryptosystems:                   ST Password
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
 * ST provisioning integrity status: 0xF5F5F5F5
 * ST HDPL1 status:                  0xFFFFFFFF
 * ST HDPL2 status:                  0xFFFFFFFF
 * ST HDPL3 status:                  0xFFFFFFFF
> Completed Discovery command.
Time for Regression: 2.434 s.
>|
```

If you perform only the Provisioning step without Regression you leave the device protected and if you try to reconnect to the device, you can see the following log:

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 1.00 MHz.
Trying Hot Plug connect procedure.
Good samples: 16 [Range 0-15].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 1.00 MHz.
JTAG-SWD Debug Port enabled.
The device probably is locked with Password or Certificate.
Current Product State is ST_LIFECYCLE_CLOSED.
Time for Connect: 0.457 s.
>|
---#TPCMD DISCOVERY
Receiving Discovery response from STM32H5 device:
 * PSA auth version:                v1.0
 * Vendor ID:                       STMicroelectronics
 * SoC Class:                       0x484 - STM32H5xxxx
 * SoC ID:                          0x00630052 0x33325116 0x38363236 0x00000000
 * HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
 * PSA lifecycle:                   ST_LIFECYCLE_CLOSED
 * SDA version:                     v2.4.0
 * Token Formats:                   0x0200
 * Certificate Formats:             0x0201
 * Cryptosystems:                   Ecdsa-P256 SHA256
 * ST provisioning integrity status: 0xEAEAEAEA
 * ST HDPL1 status:                 0x00004707
 * ST HDPL2 status:                 0xFFFFFFFF
 * ST HDPL3 status:                 0xFFFFFFFF
> Received Discovery response correctly.
Available Permission for selected device:
 * [1] ==> Full Regression
 * [2] ==> To Trust Zone Regression
 * [3] ==> Level 3 Intrusive Debug
 * [4] ==> Level 2 Intrusive Debug
 * [5] ==> Level 1 Intrusive Debug
 * [6] ==> Level 3 Intrusive Non Secure Debug
 * [7] ==> Level 2 Intrusive Non Secure Debug
 * [8] ==> Level 1 Intrusive Non Secure Debug
> Completed Discovery command.
Time for Discovery: 0.352 s.
>|
---#TPCMD PROVISIONING TRUST_ZONE_ENABLED PROGRAM_FLASH STiROT_uROT CLOSED
Product State is 0x72.
Provisioning cannot be executed in current state. Skipped operation.
Time for Provisioning: 0.001 s.
>|
```

Instead, if you perform Regression, you are able to return to OPEN Product State:

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 1.00 MHz.
Trying Hot Plug connect procedure.
Good samples: 16 [Range 0-15].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 1.00 MHz.
JTAG-SWD Debug Port enabled.
The device probably is locked with Password or Certificate.
Current Product State is ST_LIFECYCLE_CLOSED.
Time for Connect: 0.480 s.
>|
---#TPCMD REGRESSION FULL_REGRESSION
ST provisioning integrity status: 0xEAEAEAEA.
Import Private Key from FRB file:
 * Importing EC P-256 key.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
> Private Key imported correctly.
Import Trust of Chain from FRB file:
 * Found 3 certificates.
> Trust of Chain imported correctly.
Sending Challenge request to the device:
> Received Challenge correctly.
Sign the Token based on Challenge received:
> Signing Token completed successfully.
Send the data to the device to perform Authentication:
 * Sending Certificate n. 1.
 * Sending Certificate n. 2.
 * Sending Certificate n. 3.
 * Sending Authentication Token.
> Authentication completed successfully.
Wait until device reset and reboot:
> Reset detected and device reboots now.
Receiving Discovery response from STM32H5 device:
 * PSA auth version:                v1.0
 * Vendor ID:                       STMicroelectronics
 * SoC Class:                       0x484 - STM32H5xxxx
 * SoC ID:                          0x00630052 0x33325116 0x38363236 0x00000000
 * HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
 * PSA lifecycle:                   ST_LIFECYCLE_OPEN
 * SDA version:                     v2.4.0
 * Token Formats:                   0x0200
 * Certificate Formats:             0x0201
 * Cryptosystems:                   ST Password
 * ST provisioning integrity status: 0xF5F5F5F5
 * ST HDPL1 status:                 0xFFFFFFFF
 * ST HDPL2 status:                 0xFFFFFFFF
 * ST HDPL3 status:                 0xFFFFFFFF
> Completed Discovery command.
Time for Regression: 2.413 s.
>|
```

As you can see from Regression command at the end the device Product State is OPEN so the device is not protected anymore.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

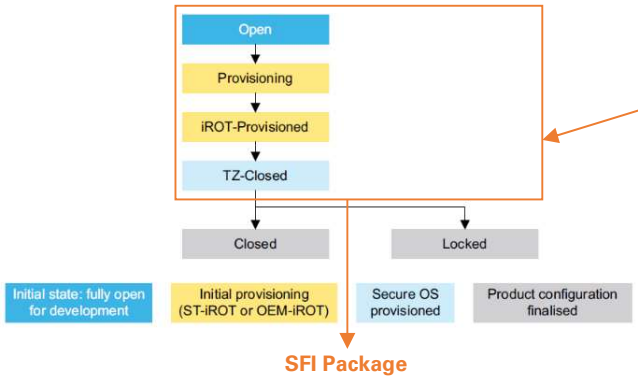
info@smh-tech.com

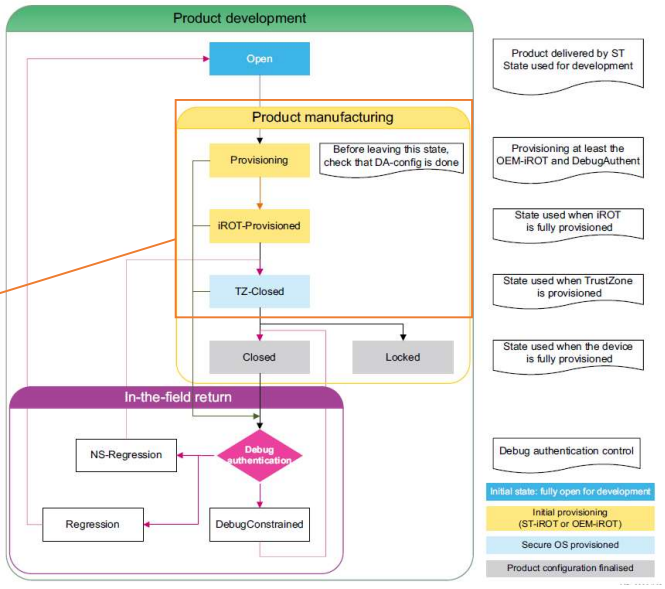
Debug Authentication STM32H5xx crypto devices Secure Firmware Install

Device configuration and setup:

STM32H57x or STM32H53x

Trust Zone Enabled
Trust Zone Enabled -> **Certificate** for Debug Authentication





Flow for Debug Authentication:

OPEN -> SECURE FIRMWARE INSTALL (SFI) -> REGRESSION

Possible Product State Transition:

Is only allowed to transit from Open to other possible product states, but not the other way around.

From	To(1)							
	Open	Provisioning	iROT-Provisioned	TZ-Closed(2)	Closed	Locked	Regression	NS-Regression(2)
Open	-	OK	OK	X	X	X	X	X
Provisioning	X	-	OK	OK	OK	OK	OK (HDPL1)	X
iROT-Provisioned	X	X	-	OK	OK	OK	OK (HDPL1)	X
TZ-Closed	X	X	X	-	OK	OK	OK (HDPL1)	X
Closed	X	X	X	X	-	X	OK (HDPL1)	OK (HDPL1)
Locked	X	X	X	X	X	-	X	X
Regression	OK (HDPL0)	X	X	X	X	X	-	X
NS-Regression	X	X	X	OK (HDPL0)	X	X	X	-

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Prepare data for Secure Firmware Install using Workbench and FRB files:

To perform Secure Firmware Install (SFI) you need to use the following files:

key_3_leaf.pem

```
-----BEGIN PRIVATE KEY-----
MIGHAgEAMBMGByqGSM49AgEGCCqGSM49AwEHBG0wawIBAQQgLgplFssrUA1cEsFs
5Ccey52cPTiUL4Lp5Wx4CGxNo1ahRANCAATiW7yHWn436uMobqog4MbrVDv8p8wY
VG4xnRf2PJD4OKPQFLHYPWlhogCT1CYT9HqE1xybrXe9qrkZ4LDCii1L
-----END PRIVATE KEY-----
```

Offset	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+a	+b	+c	+d	+e	+f	Equivalent ASCII characters
00000000	2d	2d	2d	2d	2d	42	45	47	49	4e	20	50	52	49	56	41	- - - - - B E G I N P R I V A
00000010	54	45	20	4b	45	59	2d	2d	2d	2d	2d	0d	0a	4d	49	47	T E K E Y - - - - - M I G
00000020	48	41	67	45	41	4d	42	4d	47	42	79	71	47	53	4d	34	H A g E A M B M G B y q G S M 4
00000030	39	41	67	45	47	43	43	71	47	53	4d	34	39	41	77	45	9 A g E G C C q G S M 4 9 A w E
00000040	48	42	47	30	77	61	77	49	42	41	51	51	67	4c	67	70	H B G 0 w a w I B A Q Q g L g p
00000050	6c	46	73	73	72	55	41	31	63	45	73	46	73	0d	0a	35	l F s s r U A 1 c E s F s 5
00000060	43	63	65	79	35	32	63	50	54	69	55	4c	34	4c	70	35	C c e y 5 2 c P T i U L 4 L p 5
00000070	57	78	34	43	47	78	4e	6f	31	61	68	52	41	4e	43	41	W x 4 C G x N o 1 a h R A N C A
00000080	41	54	69	57	37	79	48	57	6e	34	33	36	75	4d	6f	62	A T i W 7 y H W n 4 3 6 u M o b
00000090	71	6f	67	34	4d	62	72	56	44	76	38	70	38	77	59	0d	q o g 4 M b r V D v 8 p 8 w Y
000000a0	0a	56	47	34	78	6e	52	66	32	50	4a	44	34	4f	4b	50	V G 4 x n R f 2 P J D 4 O K P
000000b0	51	46	4c	48	59	50	57	6c	68	6f	67	43	54	31	43	59	Q F L H Y P W l h o g C T 1 C Y
000000c0	54	39	48	71	45	31	78	79	62	72	58	65	39	71	72	6b	T 9 H q E 1 x y b r X e 9 q r k
000000d0	5a	34	4c	44	43	69	69	31	4c	0d	0a	2d	2d	2d	2d	2d	Z 4 L D C i i 1 L - - - - -
000000e0	45	4e	44	20	50	52	49	56	41	54	45	20	4b	45	59	2d	E N D P R I V A T E K E Y -
000000f0	2d	2d	2d	2d	0d	0a											- - - -

This file is necessary for **Regression**.

cert_leaf_chain.b64

Offset	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+a	+b	+c	+d	+e	+f	Equivalent ASCII characters
00000000	41	41	41	42	41	74	51	41	41	41	41	42	41	41	45	42	A A A B A t Q A A A A B A A E B
00000010	41	51	45	41	41	41	41	41	41	41	41	41	41	41	41	41	A Q E A A A A A A A A A A A A A
00000020	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	A A A A A A A A A A A A A A A A
00000030	41	41	41	41	41	41	41	41	41	41	42	33	55	41	41	41	A A A A A A A A A A B 3 U A A A
00000040	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	A A A A A A A A A A A A A A A A
00000050	58	72	6d	44	53	54	67	63	66	63	57	64	70	69	47	74	X r m D S T g c f c W d p i G t
00000060	73	70	71	76	4f	4c	47	57	6f	31	48	57	4b	78	47	66	s p q v O L G W o 1 H W K x G f
00000070	47	76	45	36	4a	56	6b	43	35	4e	5a	54	2f	70	35	30	G v E 6 J V k C 5 N Z T / p 5 0
00000080	77	59	53	6a	5a	6d	42	47	78	59	55	4e	62	65	36	2b	w Y S j Z m B G x Y U N b e 6 +
00000090	7a	72	64	53	6e	57	4e	4f	4a	37	7a	74	57	52	4b	67	z r d S n W N O J 7 z t W R K g
000000a0	2f	54	41	30	74	51	41	41	41	41	41	41	41	41	41	41	/ T A 0 t Q A A A A A A A A A A
000000b0	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	A A A A A A A A A A A A A A A A
000000c0	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41	A A A A A A A A A A A A A A A A

...

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

...

00000300	41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41	A A A A A A A A A A A A A A A A
00000310	41 41 41 41 41 41 41 41 41 41 43 45 7a 73 38 78	A A A A A A A A A A C E z s 8 x
00000320	77 73 58 53 68 71 4e 52 77 64 6b 75 45 66 43 30	w s X S h q N R w d k u E f C 0
00000330	59 51 2b 77 75 65 44 39 52 52 47 70 36 4d 35 33	Y Q + w u e D 9 R R G p 6 M 5 3
00000340	44 57 4b 65 6b 31 61 76 33 6f 38 2f 44 38 63 31	D W K e k 1 a v 3 o 8 / D 8 c 1
00000350	54 36 51 69 5a 49 63 30 44 43 72 7a 70 57 2b 4a	T 6 Q i Z I c 0 D C r z p W + J
00000360	4e 45 4c 43 73 32 66 41 36 51 49 2f 78 57 65 4a	N E L C s 2 f A 6 Q I / x W e J

This file is necessary for **Regression**.

FRB file generation for Secure Firmware Install and Regression:

Now you need to import five files inside the Workbench to create a correct FRB file for Secure Firmware Install and Regression.

Import **enc_signed_RSSe_SFI_STM32H5-2M_v2.0.1.0-rc2** as a binary (.bin extension) and edit the address position to 0xF0020000

Offset	+0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +a +b +c +d +e +f	Equivalent ASCII characters
00000000	53 54 52 53 53 45 4d 47 02 00 00 00 00 00 53 54	S T R S S E M G S T
00000010	4d 33 32 48 35 53 46 49 01 00 02 00 07 00 00 00	M 3 2 H 5 S F I
00000020	00 00 00 00 00 00 00 00 00 00 00 00 a0 dc 00 00	
00000030	4b b7 ca 14 76 04 27 82 61 4b 3f 33 9e c4 e8 d1	K v ' a K ? 3
00000040	e9 db ec 42 e9 e3 4f d3 1d 91 7e d2 81 34 13 4b	B O ~ 4 K
00000050	15 fa 29 aa a9 41 33 2b ad a0 d3 fa 23 c8 7b b8	) A 3 + # {
00000060	d7 5d d5 71 ef 07 19 31 9e 91 e9 bc 30 cd f4 fb	] q 1 0

Import **SFI_Global_License.bin** as a binary (.bin extension) and edit the address position to 0xF0060000

Offset	+0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +a +b +c +d +e +f	Equivalent ASCII characters
00000000	53 46 49 47 00 00 00 00 00 00 00 00 00 00 00 00	S F I G
00000010	00 00 00 00 00 00 00 00 3f b3 90 42 df 9a 70 ae	? B p
00000020	3d 5b a4 18 67 05 c1 6a 96 dc b5 53 d5 f4 86 47	= [g j S G
00000030	97 df e2 b5 d6 63 b2 d9 14 d5 e7 84 b6 ed 5e 82	c ^
00000040	36 f5 3c 6e ac 6b e2 0f 2d 21 f6 51 e0 85 83 54	6 < n k - ! Q T
00000050	4f 24 28 c0 ad c4 64 93 c5 d0 41 cc cc 58 7c a4	O $ (d A X |
00000060	47 bc 03 64 b2 bb 35 46 a8 7e 85 57 67 ed 0f 57	G d 5 F ~ W g W

Import **SecureManagerPackage.sfi** as a binary (.bin extension) and edit the address position to 0xF00A0000

Offset	+0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +a +b +c +d +e +f	Equivalent ASCII characters
00000000	53 46 49 4d 41 47 49 43 02 00 0d 00 a6 47 fd a0	S F I M A G I C G
00000010	2e f6 99 d8 0c 91 12 51 d8 20 75 ec 48 00 01 00	. Q u H
00000020	10 00 00 00 00 00 00 00 0d 00 68 04 68 b8 ab dd	h h
00000030	78 ca 82 c0 75 d7 62 32 49 82 4f e4 44 e2 64 a4	x u b 2 I O D d
00000040	c6 02 9e 1e 1f ec 7a 00 fa 3b 92 80 2b be fe 4a	z ; + J
00000050	2d 46 09 28 eb 50 5f c9 f1 c0 4f 00 02 00 70 00	- F (P _ O p
00000060	00 00 00 01 fd 0f 0d 00 61 32 c2 6d d4 98 16 f4	a 2 m

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

[M] – SECURE MANAGER SFI			
VIRTUAL SECTION NAME	VIRTUAL SECTION START ADDRESS	VIRTUAL SECTION END ADDRESS	FILE NAME
RSSe LIBRARY	0xF0020000	0xF005FFFF	enc_signed_RSSe_SFI_STM32H5-2M_v2.0.1.0-rc2
GLOBAL LICENSE	0xF0060000	0xF009FFFF	SFI_Global_License.bin
SFI FILE	0xF00A0000	0xF049FFFF	SecureManagerPackage.sfi

Import **key_3_leaf.pem** as a binary (.bin extension) and edit the address position to 0xF0012000
Import **cert_leaf_chain.b64** as a binary (.bin extension) and edit the address position to 0xF0014000

[R] - REGRESSION			
VIRTUAL SECTION NAME	VIRTUAL SECTION START ADDRESS	VIRTUAL SECTION END ADDRESS	FILE NAME
PASSWORD	0xF0010000	0xF0011FFF	-
PRIVATE KEY	0xF0012000	0xF0013FFF	key_3_leaf.pem
CERTIFICATE CHAIN	0xF0014000	0xF0015FFF	cert_leaf_chain.b64

Here your FRB file from Advanced FRB Manager:

File Name	Memory Area	Target Start Address	Target End Address
key_3_leaf.pem	[R] - Regression (virtual memory)	0xF0012000	0xF00120F5
cert_leaf_chain.b64	[R] - Regression (virtual memory)	0xF0014000	0xF001436F
enc_signed_RSSe_SFI_STM32H5_v2.0.0.0.bin	[M] - Secure Manager SFI (virtual memory)	0xF0020000	0xF002D71F
SFI_Global_License.bin	[M] - Secure Manager SFI (virtual memory)	0xF0060000	0xF0060087
SecureManagerPackage.sfi	[M] - Secure Manager SFI (virtual memory)	0xF00A0000	0xF0105775

Possible Flow using Secure Firmware Install and Regression:

Using the `#TPCMD SECURE_FIRMWARE_INSTALL` command you install the SFI file inside the device.

Using the `#TPCMD REGRESSION` command, you can choose one of these possible flows:

```
There is a lot of possible flow for Regression:

#TPCMD REGRESSION FULL_REGRESSION

#TPCMD REGRESSION TO_TRUST_ZONE

#TPCMD REGRESSION LEVEL_3_INTRUSIVE_DEBUG
#TPCMD REGRESSION LEVEL_2_INTRUSIVE_DEBUG
#TPCMD REGRESSION LEVEL_1_INTRUSIVE_DEBUG

#TPCMD REGRESSION LEVEL_3_INTRUSIVE_NON_SECURE_DEBUG
#TPCMD REGRESSION LEVEL_2_INTRUSIVE_NON_SECURE_DEBUG
#TPCMD REGRESSION LEVEL_1_INTRUSIVE_NON_SECURE_DEBUG
```

Example using Secure Firmware Install and Regression:

Now here we can show to you an example of Provisioning and Regression.
In this case we decide to perform FULL_REGRESSION for Regression.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Project commands:

```
#TPSETSRC STM32H5xx_Secure_Firmware_Install.frb
#TPSTART
#TPCMD CONNECT
#TPCMD SECURE_FIRMWARE_INSTALL
#TPCMD DISCOVERY
#TPCMD REGRESSION FULL_REGRESSION
#TPCMD DISCONNECT
#TPEND
```

Project execution:

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 1.00 MHz.
Trying Hot Plug connect procedure.
Good samples: 16 [Range 0-15].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 1.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x54770002, Type: AMBA APB2 or APB3 bus.
AP[1] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[1] ROM table base address 0xE00FE000.
CPUID: 0x410FD214.
Implementer Code: 0x41 - [ARM].
Found Cortex M33 revision r0p4.
Program counter value is 0x0BF98D94.
Cortex M33 Core halted [0.002 s].
> Device configuration 1: [0x30F0EDF8]
 * The device's product state is Open [0xED].
 * BOR level 1, the threshold level is low (around 2.1 V).
 * Brownout high disabled.
 * IWDG watchdog is controlled by software.
 * WWDG watchdog is controlled by software.
 * No reset generated when entering Stop mode on core domain.
 * No reset generated when entering Standby mode on core domain.
 * High-speed IO at low VDD voltage feature disabled (VDD can exceed 2.7 V).
 * High-speed IO at low VDDIO2 voltage feature disabled (VDDIO2 can exceed 2.7 V).
 * Independent watchdog keep running in system Stop mode.
 * Independent watchdog keep running in Standby mode.
 * Bank1 and Bank2 not swapped.
> Device configuration 2: [0xC3000034]
 * SRAM1 and SRAM3 not erased when a system reset occurs.
 * SRAM2 erased when a system reset occurs.
 * BKPRAM ECC check disabled.
 * SRAM3 ECC check disabled.
 * SRAM2 ECC check enabled.
 * TrustZone is disabled.
 * UBE OEM-iRoT (user flash) selected. In Open Product State this value selects bootloader.
PLL enabled using internal HSI oscillator.
Requested Clock is 37.50 MHz.
Generated Clock is 37.50 MHz.
Good samples: 4 [Range 3-6].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 37.50 MHz.
Time for Connect: 0.468 s.
>|
---#TPCMD SECURE_FIRMWARE_INSTALL
Parse SFI file from FRB to check configuration:
> SFI file contains supported areas.
Check current device Product State:
> Device is in a valid life state.
Set correct Option Bytes configuration:
> Option Bytes programmed correctly.
Read device Descriptor:
 * Descriptor version: 01.00.00.
 * Start address of the available SRAM for host: 0x20054000.
 * End address of the available SRAM for host: 0x2008FFFF.
> Device Descriptor read successfully.
Load the RSSe library into STM32 device:
 * Import RSSe library from FRB file.
 * Send RSSe library into the device.
 * Verify RSSe library sent to the device.
> RSSe library programmed correctly.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
Start RSSe library service:
 * RSSe library service installed.
 * Reset the device to have RSSe ready.
 * Wait until device reboots.
 * Installed RSSe version 2.0.1.
> RSSe library service is ready.
Processing license:
 * Import License from FRB file.
 * Send License into the device.
 * Verify License sent to the device.
> License programmed correctly.
Execute Process License payload.
> Succeed to execute Process License payload.
Import SFI file from FRB:
 * Import SFI from FRB file.
 * Imported file has a valid SFI image header.
 * Protocol version is 2.
 * Total areas number is 13.
> SFI file imported correctly.
Processing SFI Image Header.
> Completed processing SFI Image Header after 0.014 s.
Processing SFI Area n. 1 marked as H.
> Completed processing SFI Area H after 0.951 s.
Processing SFI Area n. 2 marked as O.
> Completed processing SFI Area O after 0.018 s.
Processing SFI Area n. 3 marked as O.
> Completed processing SFI Area O after 0.023 s.
Processing SFI Area n. 4 marked as O.
> Completed processing SFI Area O after 0.017 s.
Processing SFI Area n. 5 marked as O.
> Completed processing SFI Area O after 0.022 s.
Processing SFI Area n. 6 marked as S.
> Completed processing SFI Area S after 4.189 s.
Processing SFI Area n. 7 marked as S.
> Completed processing SFI Area S after 4.344 s.
Processing SFI Area n. 8 marked as S.
> Completed processing SFI Area S after 4.260 s.
Processing SFI Area n. 9 marked as S.
> Completed processing SFI Area S after 1.040 s.
Processing SFI Area n. 10 marked as F.
> Completed processing SFI Area F after 0.516 s.
Processing SFI Area n. 11 marked as F.
> Completed processing SFI Area F after 0.031 s.
Processing SFI Area n. 12 marked as F.
> Completed processing SFI Area F after 0.015 s.
Processing SFI Area n. 13 marked as C.
> Completed processing SFI Area C after 0.014 s.
Device reboots now.
> SFI process completed successfully.
Time for Secure Firmware Install: 18.131 s.
>|
---#TPCMD DISCOVERY
Receiving Discovery response from STM32H5 device:
 * PSA auth version:                v1.0
 * Vendor ID:                       STMicroelectronics
 * SoC Class:                       0x484 - STM32H5xxxx
 * SoC ID:                          0x00630052 0x33325116 0x38363236 0x00000000
 * HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
 * PSA lifecycle:                   ST_LIFECYCLE_TZ_CLOSED
 * SDA version:                     v2.4.0
 * Token Formats:                   0x0200
 * Certificate Formats:             0x0201
 * Cryptosystems:                   Ecdsa-P256 SHA256
 * ST provisioning integrity status: 0xEAEAEAEA
 * ST HDPL1 status:                 0x00002717
 * ST HDPL2 status:                 0x00000000
 * ST HDPL3 status:                 0xFFFFFFFF
> Received Discovery response correctly.
Available Permission for selected device:
 * [1] ==> Full Regression
 * [2] ==> To Trust Zone Regression
 * [3] ==> Level 3 Intrusive Debug
 * [4] ==> Level 2 Intrusive Debug
 * [5] ==> Level 1 Intrusive Debug
 * [6] ==> Level 3 Intrusive Non Secure Debug
 * [7] ==> Level 2 Intrusive Non Secure Debug
 * [8] ==> Level 1 Intrusive Non Secure Debug
> Completed Discovery command.
Time for Discovery: 0.353 s.
>|
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
---#TPCMD REGRESSION FULL_REGRESSION
ST provisioning integrity status: 0xEAEAEAEA.
Import Private Key from FRB file:
 * Importing EC P-256 key.
> Private Key imported correctly.
Import Trust of Chain from FRB file:
 * Found 3 certificates.
> Trust of Chain imported correctly.
Sending Challenge request to the device:
> Received Challenge correctly.
Sign the Token based on Challenge received:
> Signing Token completed successfully.
Send the data to the device to perform Authentication:
 * Sending Certificate n. 1.
 * Sending Certificate n. 2.
 * Sending Certificate n. 3.
 * Sending Authentication Token.
> Authentication completed successfully.
Wait until device reset and reboot:
> Reset detected and device reboots now.
Receiving Discovery response from STM32H5 device:
 * PSA auth version:                v1.0
 * Vendor ID:                       STMicroelectronics
 * SoC Class:                       0x484 - STM32H5xxxx
 * SoC ID:                          0x00630052 0x33325116 0x38363236 0x00000000
 * HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
 * PSA lifecycle:                   ST_LIFECYCLE_OPEN
 * SDA version:                     v2.4.0
 * Token Formats:                   0x0200
 * Certificate Formats:             0x0201
 * Cryptosystems:                   ST Password
 * ST provisioning integrity status: 0xF5F5F5F5
 * ST HDPL1 status:                 0xFFFFFFFF
 * ST HDPL2 status:                 0xFFFFFFFF
 * ST HDPL3 status:                 0xFFFFFFFF
> Completed Discovery command.
Time for Regression: 2.463 s.
>|
```

If you perform only the Provisioning step without Regression you leave the device protected and if you try to reconnect to the device, you can see the following log:

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 1.00 MHz.
Trying Hot Plug connect procedure.
Good samples: 16 [Range 0-15].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 1.00 MHz.
JTAG-SWD Debug Port enabled.
The device probably is locked with Password or Certificate.
Current Product State is ST_LIFECYCLE_TZ_CLOSED.
Time for Connect: 0.456 s.
>|
---#TPCMD DISCOVERY
Receiving Discovery response from STM32H5 device:
 * PSA auth version:                v1.0
 * Vendor ID:                       STMicroelectronics
 * SoC Class:                       0x484 - STM32H5xxxx
 * SoC ID:                          0x00630052 0x33325116 0x38363236 0x00000000
 * HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
 * PSA lifecycle:                   ST_LIFECYCLE_TZ_CLOSED
 * SDA version:                     v2.4.0
 * Token Formats:                   0x0200
 * Certificate Formats:             0x0201
 * Cryptosystems:                   Ecdsa-P256 SHA256
 * ST provisioning integrity status: 0xEAEAEAEA
 * ST HDPL1 status:                 0x00002717
 * ST HDPL2 status:                 0x500001BF
 * ST HDPL3 status:                 0xFFFFFFFF
> Received Discovery response correctly.
Available Permission for selected device:
 * [1] ==> Full Regression
 * [2] ==> To Trust Zone Regression
 * [3] ==> Level 3 Intrusive Debug
 * [4] ==> Level 2 Intrusive Debug
 * [5] ==> Level 1 Intrusive Debug
 * [6] ==> Level 3 Intrusive Non Secure Debug
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
 * [7] ==> Level 2 Intrusive Non Secure Debug
 * [8] ==> Level 1 Intrusive Non Secure Debug
> Completed Discovery command.
Time for Discovery: 0.351 s.
>|
```

Instead, if you perform Regression, you are able to return to OPEN Product State:

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 1.00 MHz.
Trying Hot Plug connect procedure.
Good samples: 16 [Range 0-15].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 1.00 MHz.
JTAG-SWD Debug Port enabled.
The device probably is locked with Password or Certificate.
Current Product State is ST_LIFECYCLE_CLOSED.
Time for Connect: 0.480 s.
>|
---#TPCMD REGRESSION FULL_REGRESSION
ST provisioning integrity status: 0xEAEAEAEA.
Import Private Key from FRB file:
 * Importing EC P-256 key.
> Private Key imported correctly.
Import Trust of Chain from FRB file:
 * Found 3 certificates.
> Trust of Chain imported correctly.
Sending Challenge request to the device:
> Received Challenge correctly.
Sign the Token based on Challenge received:
> Signing Token completed successfully.
Send the data to the device to perform Authentication:
 * Sending Certificate n. 1.
 * Sending Certificate n. 2.
 * Sending Certificate n. 3.
 * Sending Authentication Token.
> Authentication completed successfully.
Wait until device reset and reboot:
> Reset detected and device reboots now.
Receiving Discovery response from STM32H5 device:
 * PSA auth version:                v1.0
 * Vendor ID:                       STMicroelectronics
 * SoC Class:                       0x484 - STM32H5xxxx
 * SoC ID:                          0x00630052 0x33325116 0x38363236 0x00000000
 * HW permission mask:              0x00005077 0x00000000 0x00000000 0x00000000
 * PSA lifecycle:                   ST_LIFECYCLE_OPEN
 * SDA version:                     v2.4.0
 * Token Formats:                   0x0200
 * Certificate Formats:             0x0201
 * Cryptosystems:                   ST Password
 * ST provisioning integrity status: 0xF5F5F5F5
 * ST HDPL1 status:                 0xFFFFFFFF
 * ST HDPL2 status:                 0xFFFFFFFF
 * ST HDPL3 status:                 0xFFFFFFFF
> Completed Discovery command.
Time for Regression: 2.490 s.
>|
```

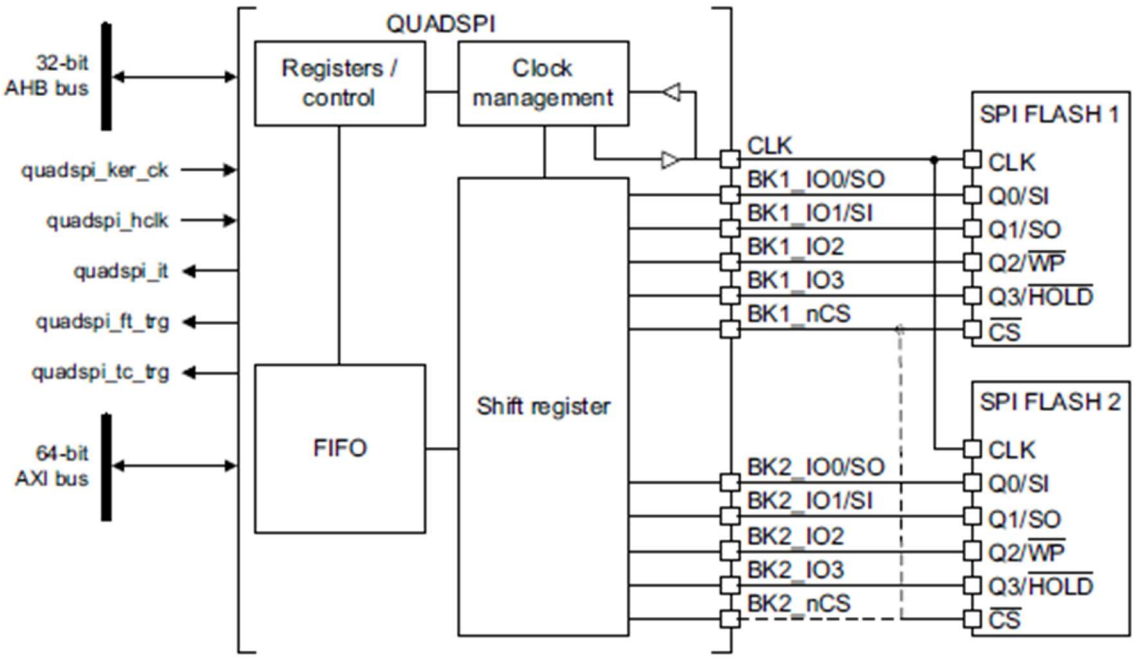
As you can see from Regression command at the end the device Product State is OPEN so the device is not protected anymore.

STM32 Program External Flash Memory

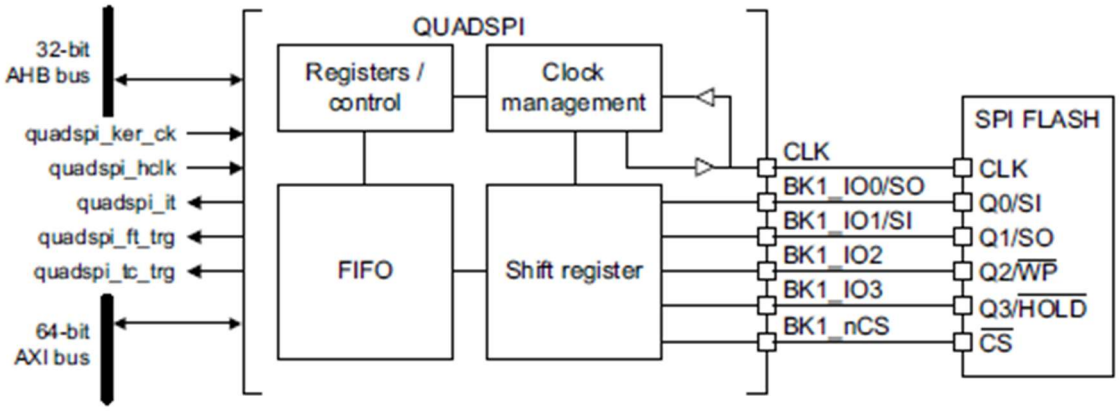
Program External Flash memory introduction H7 Family

This chapter describes how to program and External Memory through STM32H7 series.

Here the scheme when Dual Bank mode configuration is enabled.



Here the scheme when Dual Bank mode configuration is not enabled.



HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Specific Additional Parameters for STM32H7 External Memory

This chapter explains how to configure the STM32 driver Additional Parameters to program an external memory connected to an STM32H7 device.

The QUADSPI is a specialized communication interface targeting single, dual or quad SPI Flash memories.

Both throughput and capacity can be increased two-fold using dual-flash mode, where two Quad-SPI Flash memories are accessed simultaneously.

The **TCSETPARs** for configuring the GPIOs for bank 1 and bank 2 of the QSPI are listed below.

You will have to configure only one bank if you are using QSPI single bank configuration, while if you are using the QSPI Dual Bank configuration then you will have to configure all the parameters present here.

-- Bank 1 --
#TCSETPAR PIN_QSPI_BANK1_NCS -> STM32 QSPI Bank 1 GPIO Pin for NCS signal.
#TCSETPAR PIN_QSPI_BANK1_IO0 -> STM32 QSPI Bank 1 GPIO Pin for IO0 signal.
#TCSETPAR PIN_QSPI_BANK1_IO1 -> STM32 QSPI Bank 1 GPIO Pin for IO1 signal.
#TCSETPAR PIN_QSPI_BANK1_IO2 -> STM32 QSPI Bank 1 GPIO Pin for IO2 signal.
#TCSETPAR PIN_QSPI_BANK1_IO3 -> STM32 QSPI Bank 1 GPIO Pin for IO3 signal.

-- Bank 2 --
#TCSETPAR PIN_QSPI_BANK2_NCS -> STM32 QSPI Bank 2 GPIO Pin for NCS signal.
#TCSETPAR PIN_QSPI_BANK2_IO0 -> STM32 QSPI Bank 2 GPIO Pin for IO0 signal.
#TCSETPAR PIN_QSPI_BANK2_IO1 -> STM32 QSPI Bank 2 GPIO Pin for IO1 signal.
#TCSETPAR PIN_QSPI_BANK2_IO2 -> STM32 QSPI Bank 2 GPIO Pin for IO2 signal.
#TCSETPAR PIN_QSPI_BANK2_IO3 -> STM32 QSPI Bank 2 GPIO Pin for IO3 signal.

Obviously, the clock parameter must always be set, it is unique whether you use the QSPI configured in Single or Dual Bank:

#TCSETPAR PIN_QSPI_CLOCK -> STM32 QSPI GPIO Pin for CLOCK signal.

For all the **#TCSETPAR**s listed above, the following 3 parameters must be configured:

#TCSETPAR <PIN_QSPI_...> [*1] [*2] [*3]

[*1] -> NO_PULLUP_NO_PULLDOWN / PULLUP / PULLDOWN
Configure the status of a given data line. For the NCS and CLOCK lines, the PULLUP is usually set, while for the other lines, NO_PULLUP_NO_PULLDOWN is set.
If you have defined your setup via STM32CubeIDE you can find these parameters in the "Connectivity->QSPI" section.

[*2] -> VERY_HIGH / HIGH / MEDIUM / LOW
Configure the maximum output of a given data line. Normally the VERY_HIGH value is set.
If you have defined your setup via STM32CubeIDE you can find these parameters in the "Connectivity->QSPI" section.

[*3] -> PHYSICAL PIN
Configure the physical pin to associate to the QSPI peripheral signal.
If you have defined your setup via STM32CubeIDE you can find these parameters in the "Connectivity->QSPI" section.

#TCSETPAR QSPI_BANK [1/2]

Select QSPI bank 1 or 2.
You can select bank 1 or bank 2 of the QSPI if you are using a Single bank memory.
If you are using the Dual Bank configuration this parameter is superfluous.

#TCSETPAR QSPI_PROTOCOL [SPI/QUAD-SPI]

Select if you want to communicate with the memory via the SPI or QUAD-SPI protocol.

UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Obviously, if the QUAD-SPI is used, it is necessary to configure the four data lines in addition to the Clock and the Chip Select.

#TCSETPAR QSPI_DDR_ENABLE [YES/NO]

Select if you want to communicate with DRR (Double Data Rate - Double Transfer Rate) enabled.

Now we need to configure the PLL of the STM32 device we are using.
This is essential because the PLL also sets the clock frequency that will be used by the STM32 QSPI peripheral.
For the configuration of the PLL it is necessary to set the following TCSETPAR:

#TCSETPAR QSPI_OSCILLATOR_TYPE [CSI/HSI] -> Select internal CSI (4MHz) or HSI (64MHz) oscillator.

#TCSETPAR QSPI_PLLM [Value] -> PLL Division factor M.
#TCSETPAR QSPI_PLLN [Value] -> PLL Multiplication factor N.
#TCSETPAR QSPI_PLLP [Value] -> PLL Division factor P.
#TCSETPAR QSPI_PLLQ [Value] -> PLL Division factor Q.
#TCSETPAR QSPI_PLLR [Value] -> PLL Division factor R.

#TCSETPAR QSPI_SYSCLK_DIV [Value] -> Divider to obtain System Clock.
#TCSETPAR QSPI_AHBCLK_DIV [Value] -> Divider to obtain AXI, HCLK3, AHB1 Peripheral, AHB2 and AHB4 Peripheral clocks.
#TCSETPAR QSPI_APB1CLK_DIV [Value] -> Divider to obtain APB1 Peripheral and into APB1 timer clocks.
#TCSETPAR QSPI_APB2CLK_DIV [Value] -> Divider to obtain APB2 Peripheral and into APB2 timer clocks.
#TCSETPAR QSPI_APB3CLK_DIV [Value] -> Divider to obtain APB3 Peripheral clock.
#TCSETPAR QSPI_APB4CLK_DIV [Value] -> Divider to obtain APB4 Peripheral clock.

#TCSETPAR QSPI_SPI_PRESCALER [Value] -> Divider for QSPI Peripheral clock.

The first five parameters are the multiplicator and divisor factor.

If you have defined your setup via STM32CubeIDE you can find these parameters in the "Clock Configuration" section using PLLCLK into System Clock Mux.

The formula you have to take into consideration is this **((OscillatorMHz /pllM) * pllN) / pllP** and the maximum frequency value following this step is **480 MHz**.

After this first step you can obtain the SYSCLK clock and you can divide it by QSPI_SYSCLK_DIV factor.

#TCSETPAR QSPI_SYSCLK_DIV [Value]

This can be set from a minimum value of 1 to a maximum value of 512 and with this division you to obtain the System Clock, which also cannot exceed 480 MHz.

This System Clock is used for the CPU Clock (MHz) and for CPU Systick Clock (MHz).
Now we will use the next five parameters which are dividers that allow us to obtain the clocks for the various peripherals.

#TCSETPAR QSPI_AHBCLK_DIV [Value]

This parameter is set by HPRE Prescaler into STM32CubeIDE and the maximum frequency available after this division is 240 MHz. The obtained clock is used for AXI Peripheral, HCLK3, AHB1 Peripheral, AHB2 Peripheral and AHB4 Peripheral (MHz).

#TCSETPAR QSPI_APB3CLK_DIV [Value]

This parameter is set by D1PRE into STM32CubeIDE and the maximum frequency available after this division is 120 MHz. The obtained clock is used for APB3 Peripheral (MHz).

#TCSETPAR QSPI_APB1CLK_DIV [Value]

This parameter is set by D2PRE1 into STM32CubeIDE and the maximum frequency available after this division is 120 MHz. The obtained clock is used for APB1 Peripheral and into APB1 timer clocks (MHz).

#TCSETPAR QSPI_APB2CLK_DIV [Value]

UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

This parameter is set by D2PRE2 into STM32CubeIDE and the maximum frequency available after this division is 120 MHz. The obtained clock is used for APB2 Peripheral and into APB2 timer clocks (MHz).

```
#TCSETPAR QSPI_APB4CLK_DIV [Value]
```

This parameter is set by D3PRE into STM32CubeIDE and the maximum frequency available after this division is 120 MHz. The obtained clock is used for APB4 Peripheral clocks (MHz).

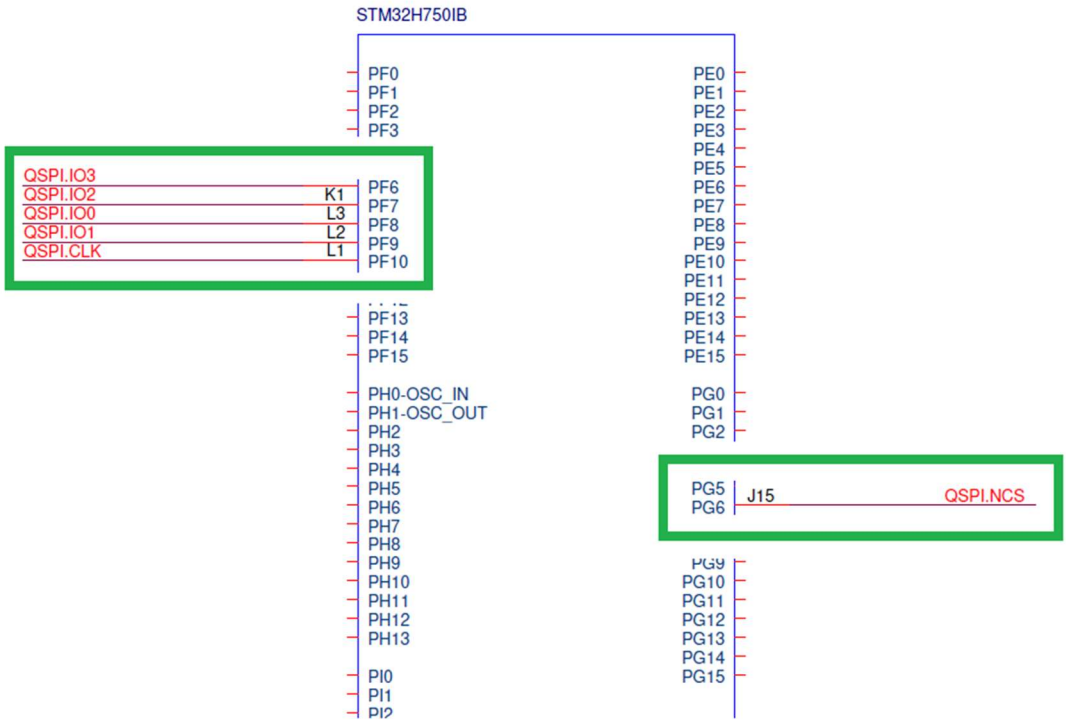
The HCLK3 clock is supplied to the QSPI peripheral of the STM32 device.
Through the `#TCSETPAR QSPI_SPI_PRESCALER [Value]` parameter it is possible to divide this clock value to obtain the actual QSPI clock.

Please be careful, it is not possible to divide by a value less than 2, so the minimum value of the QSPI_SPI_PRESCALER parameter is 1.

STM32H7 Single Bank External Memory

Now I show an example on how to correctly configure the QSPI parameters of the STM32 device.
Let's start by considering this schematic:

We have one STM32H750IB connected to an External MX25R3235F memory



STM32H750IB_1x_MX25R3235F Additional Parameters

So, we need to configure the `#TCSETPAR` in this way:

```
#TCSETPAR PIN_QSPI_BANK1_NCS PULLUP VERY_HIGH PG6
#TCSETPAR PIN_QSPI_BANK1_IO0 NO_PULLUP_NO_PULLDOWN VERY_HIGH PF8
#TCSETPAR PIN_QSPI_BANK1_IO1 NO_PULLUP_NO_PULLDOWN VERY_HIGH PF9
#TCSETPAR PIN_QSPI_BANK1_IO2 NO_PULLUP_NO_PULLDOWN VERY_HIGH PF7
#TCSETPAR PIN_QSPI_BANK1_IO3 NO_PULLUP_NO_PULLDOWN VERY_HIGH PF6
#TCSETPAR PIN_QSPI_CLOCK PULLUP VERY_HIGH PF10
```


UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

And we need to configure properly the others #TCSETPAR parameters:

```
#TCSETPAR QSPI_OSCILLATOR_TYPE CSI
```

We start selecting CSI oscillator so internal basic frequency is 4 MHz.
Now we want to use the QSPI peripheral at 110 MHz so we need to set the parameters below following the formula ((OscillatorMHz /pllM) * pllN) / pllP.
So, we need to set:

```
#TCSETPAR QSPI_PLLM 1
#TCSETPAR QSPI_PLLN 220
#TCSETPAR QSPI_PLLP 2
#TCSETPAR QSPI_PLLQ 4
#TCSETPAR QSPI_PLLR 2
```

We obtain:

QSPI peripheral frequency = **((4 MHz / 1) * 220) / 2 = 440 MHz**

We need to have the clock after DC1PRE at maximum 220MHz and after DC2PRE at maximum 110MHz.
So, we need to set:

```
#TCSETPAR QSPI_SYSCLK_DIV 2
#TCSETPAR QSPI_SYSCLK_DIV 2
```

We obtain:

DC1PRE = **440 MHz** / 2 = **220 MHz**
DC2PRE = **220 MHz** / 2 = **110 MHz**

We need to set:

```
#TCSETPAR QSPI_APB1CLK_DIV 2
#TCSETPAR QSPI_APB2CLK_DIV 2
#TCSETPAR QSPI_APB3CLK_DIV 2
#TCSETPAR QSPI_APB4CLK_DIV 2
```

Now for the Quad-SPI protocol we need to select only Bank n. 1, Double Data Rate enabled and QSPI Prescaler set to 1, so QSPI frequency is 110 MHz / 1 = 110 MHz.

```
#TCSETPAR QSPI_BANK 1
#TCSETPAR QSPI_DDR_ENABLE YES
#TCSETPAR QSPI_SPI_PRESCALER 1
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

STM32H750IB_1x_MX25R3235F Execution with Blank External Memory 4MB

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[1] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[2] IDR: 0x54770002, Type: AMBA APB2 or APB3 bus.
AP[0] ROM table base address 0xE00FE000.
CPUID: 0x411FC271.
Implementer Code: 0x41 - [ARM].
Found Cortex M7 revision r1p1.
Cortex M7 Core halted [0.001 s].
Device configuration: [0x1BC6AAF0]
 * The device's RDP level is 0 [0xAA].
 * Brownout reset threshold 0 (VBOR0).
 * IWDG1 watchdog is controller by software.
 * No reset generated when entering DStop mode on D1 domain.
 * No reset generated when entering DStandby mode on D1 domain.
 * Independent watchdog keep running in System Stop mode.
 * Independent watchdog keep running in Standby mode.
 * Security feature disabled.
 * Product working in the full voltage range, I/O speed optimization at low-voltage disabled.
 * Flash bank 1 and bank 2 are not swapped.
PLL enabled using internal HSI oscillator.
Requested Clock is 37.50 MHz.
Generated Clock is 37.50 MHz.
Sampling Point Selected: 7.
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 37.50 MHz.
Time for Connect: 0.108 s.
>|
---#IFERR TPCMD BLANKCHECK X
PLL settings configuration:
 * Selected Oscillator: CSI.
 * SYSCLK: ((4MHz/pllM)*pllN)/pllP -> 440MHz.
 * Clock after DC1PRE 220MHz.
 * Clock after DC2PRE 110MHz.
 * Clock to QUADSPI module 110.00MHz.
> Base system clock initialized.
Initialize QSPI GPIO pins.
> QSPI GPIO pins are initialized.
Enabling STM32 QSPI peripheral.
 * Clock prescaler is 1, QSPI peripheral clock is: 55.00MHz.
 * Flash Size is 4 MiB.
 * Dual Flash Mode: Disabled.
> STM32 QSPI peripheral enabled.
Read and check external memory.
 * QSPI Bank 1: Memory ID: Expected 0x1628C2 - Read 0x1628C2.
 * QSPI Bank 1: Status register 0x40.
 * Clean memory status register.
 * QSPI Bank 1: Status register 0x00.
 * QSPI Bank 1: SFDP table supported.
 * DDR cannot be enabled for selected memory [Read 0xF1 at 0x30].
 * QSPI Bank 1: Flash size check passed: 4MiB.
> Completed read and check external memory.
Configure external memory.
 * Switching from SPI to QUAD-SPI protocol.
 * QSPI Bank 1: Memory ID: Expected 0x1628C2 - Read 0x1628C2.
 * DDR [Double Data Rate] mode disabled.
 * Set eight dummy cycles.
 * Use 3-Byte address mode operation.
> External memory is configured.
Execute selected command...
External Memory enter Memory Mapping Mode.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
 * DDR [Double Data Rate] disabled.
> Enter Memory Mapping Mode completed.
Time for Blankcheck X: 0.151 s.
>|
---#TPCMD PROGRAM X
FRB CRC32 check passed.
FRB Headers collected.
External Memory enter Direct Mode.
> Enter Direct Mode completed.
Time for Program X: 16.045 s.
>|
---#TPCMD VERIFY X R
External Memory enter Memory Mapping Mode.
 * DDR [Double Data Rate] disabled.
> Enter Memory Mapping Mode completed.
Time for Verify Readout X: 1.639 s.
>|
---#TPCMD VERIFY X S
External Memory is already in Memory Mapping Mode.
Time for Verify Checksum 32bit X: 0.082 s.
>|
---#TPCMD DISCONNECT
```

STM32H750IB_1x_MX25R3235F Programming Times with Blank External Memory 4MB

Operation	Timings FlashRunner 2.0
Time for Connect	0.106 s
Time for Blankcheck External Flash	0.151 s
Time for Program External Flash	16.045 s
Time for Verify Readout External Flash	1.639 s
Time for Verify Checksum External Flash	0.082 s
Cycle Time	**00:18.089 s**

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

STM32H750IB_1x_MX25R3235F Execution with Not Blank External Memory 4MB

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[1] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[2] IDR: 0x54770002, Type: AMBA APB2 or APB3 bus.
AP[0] ROM table base address 0xE00FE000.
CPUID: 0x411FC271.
Implementer Code: 0x41 - [ARM].
Found Cortex M7 revision r1p1.
Cortex M7 Core halted [0.001 s].
Device configuration: [0x1BC6AAF0]
 * The device's RDP level is 0 [0xAA].
 * Brownout reset threshold 0 (VBOR0).
 * IWDG1 watchdog is controller by software.
 * No reset generated when entering DStop mode on D1 domain.
 * No reset generated when entering DStandby mode on D1 domain.
 * Independent watchdog keep running in System Stop mode.
 * Independent watchdog keep running in Standby mode.
 * Security feature disabled.
 * Product working in the full voltage range, I/O speed optimization at low-voltage disabled.
 * Flash bank 1 and bank 2 are not swapped.
PLL enabled using internal HSI oscillator.
Requested Clock is 37.50 MHz.
Generated Clock is 37.50 MHz.
Sampling Point Selected: 7.
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 37.50 MHz.
Time for Connect: 0.108 s.
>|
---#IFERR TPCMD BLANKCHECK X
PLL settings configuration:
 * Selected Oscillator: CSI.
 * SYSCLK: ((4MHz/pllM)*pllN)/pllP -> 440MHz.
 * Clock after DC1PRE 220MHz.
 * Clock after DC2PRE 110MHz.
 * Clock to QUADSPI module 110.00MHz.
> Base system clock initialized.
Initialize QSPI GPIO pins.
> QSPI GPIO pins are initialized.
Enabling STM32 QSPI peripheral.
 * Clock prescaler is 1, QSPI peripheral clock is: 55.00MHz.
 * Flash Size is 4 MiB.
 * Dual Flash Mode: Disabled.
> STM32 QSPI peripheral enabled.
Read and check external memory.
 * QSPI Bank 1: Memory ID: Expected 0x1628C2 - Read 0x1628C2.
 * QSPI Bank 1: Status register 0x40.
 * Clean memory status register.
 * QSPI Bank 1: Status register 0x00.
 * QSPI Bank 1: SFDP table supported.
 * DDR cannot be enabled for selected memory [Read 0xF1 at 0x30].
 * QSPI Bank 1: Flash size check passed: 4MiB.
> Completed read and check external memory.
Configure external memory.
 * Switching from SPI to QUAD-SPI protocol.
 * QSPI Bank 1: Memory ID: Expected 0x1628C2 - Read 0x1628C2.
 * DDR [Double Data Rate] mode disabled.
 * Set eight dummy cycles.
 * Use 3-Byte address mode operation.
> External memory is configured.
Execute selected command...
External Memory enter Memory Mapping Mode.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
 * DDR [Double Data Rate] disabled.
> Enter Memory Mapping Mode completed.
External memory is not blank.
0800A2CD!|
---#THEN TPCMD MASSERASE X
External Memory enter Direct Mode.
> Enter Direct Mode completed.
Time for Masserase X: 19.029 s.
>|
---#THEN TPCMD BLANKCHECK X
External Memory enter Memory Mapping Mode.
 * DDR [Double Data Rate] disabled.
> Enter Memory Mapping Mode completed.
Time for Blankcheck X: 0.081 s.
>|
---#TPCMD PROGRAM X
External Memory enter Direct Mode.
> Enter Direct Mode completed.
Time for Program X: 16.045 s.
>|
---#TPCMD VERIFY X R
External Memory enter Memory Mapping Mode.
 * DDR [Double Data Rate] disabled.
> Enter Memory Mapping Mode completed.
Time for Verify Readout X: 1.639 s.
>|
---#TPCMD VERIFY X S
External Memory is already in Memory Mapping Mode.
Time for Verify Checksum 32bit X: 0.082 s.
>|
---#TPCMD DISCONNECT
```

STM32H750IB_1x_MX25R3235F Programming Times with Not Blank External Memory 4MB

Operation	Timings FlashRunner 2.0
Time for Connect	0.106 s
Time for Masserase External Flash	20.044 s
Time for Blankcheck External Flash	0.081 s
Time for Program External Flash	2.045 s
Time for Verify Readout External Flash	0.242 s
Time for Verify Checksum External Flash	0.059 s
Cycle Time	**00:23.763 s**

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

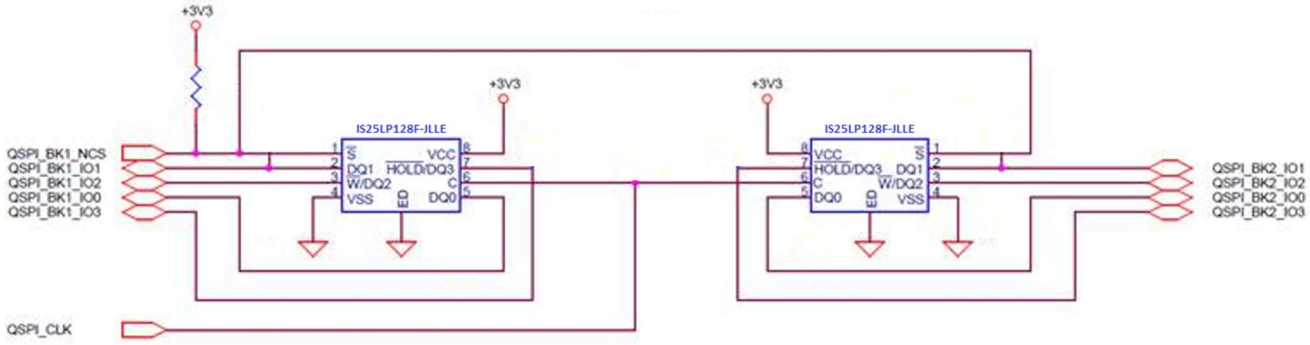
→smh-tech.com

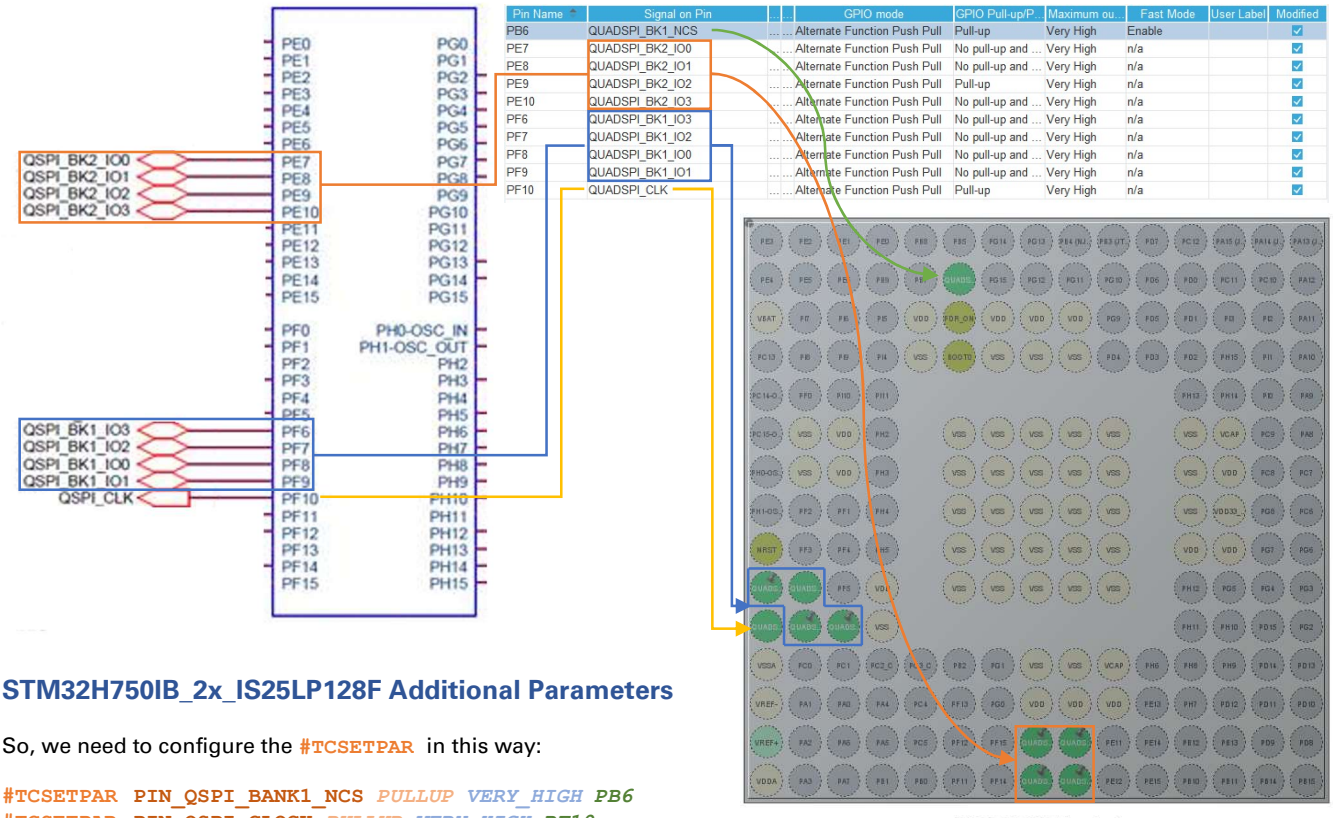
info@smh-tech.com

STM32H7 Dual Bank External Memory

Now I show an example on how to correctly configure the QSPI parameters of the STM32 device.
Let's start by considering this schematic:

We have one STM32H750IB connected to two External IS25LP128F memories:





UFBGA176 +25 (Top view)

STM32H750IB_2x_IS25LP128F Additional Parameters

So, we need to configure the `#TCSETPAR` in this way:

```
#TCSETPAR PIN_QSPI_BANK1_NCS PULLUP VERY_HIGH PB6
#TCSETPAR PIN_QSPI_CLOCK PULLUP VERY_HIGH PF10

#TCSETPAR PIN_QSPI_BANK1_IO0 NO_PULLUP_NO_PULLDOWN VERY_HIGH PF8
#TCSETPAR PIN_QSPI_BANK1_IO1 NO_PULLUP_NO_PULLDOWN VERY_HIGH PF9
#TCSETPAR PIN_QSPI_BANK1_IO2 NO_PULLUP_NO_PULLDOWN VERY_HIGH PF7
#TCSETPAR PIN_QSPI_BANK1_IO3 NO_PULLUP_NO_PULLDOWN VERY_HIGH PF6

#TCSETPAR PIN_QSPI_BANK2_IO0 NO_PULLUP_NO_PULLDOWN VERY_HIGH PE7
#TCSETPAR PIN_QSPI_BANK2_IO1 NO_PULLUP_NO_PULLDOWN VERY_HIGH PE8
#TCSETPAR PIN_QSPI_BANK2_IO2 NO_PULLUP_NO_PULLDOWN VERY_HIGH PE9
#TCSETPAR PIN_QSPI_BANK2_IO3 NO_PULLUP_NO_PULLDOWN VERY_HIGH PE10
```


UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

And we need to configure properly the others #TCSETPAR parameters:

```
#TCSETPAR QSPI_OSCILLATOR_TYPE CSI
```

We start selecting CSI oscillator so internal basic frequency is 4 MHz.
Now we want to use the QSPI peripheral at 110 MHz so we need to set the parameters below following the formula ((OscillatorMHz /pllM) * pllN) / pllP.
So, we need to set:

```
#TCSETPAR QSPI_PLLM 1
#TCSETPAR QSPI_PLLN 220
#TCSETPAR QSPI_PLLP 2
#TCSETPAR QSPI_PLLQ 4
#TCSETPAR QSPI_PLLR 2
```

We obtain:

QSPI peripheral frequency = **((4 MHz / 1) * 220) / 2 = 440 MHz**

We need to have the clock after DC1PRE at maximum 220MHz and after DC2PRE at maximum 110MHz.
So, we need to set:

```
#TCSETPAR QSPI_SYSCLK_DIV 2
#TCSETPAR QSPI_SYSCLK_DIV 2
```

We obtain:

DC1PRE = **440 MHz** / 2 = **220 MHz**
DC2PRE = **220 MHz** / 2 = **110 MHz**

We need to set:

```
#TCSETPAR QSPI_APB1CLK_DIV 1
#TCSETPAR QSPI_APB2CLK_DIV 1
#TCSETPAR QSPI_APB3CLK_DIV 1
#TCSETPAR QSPI_APB4CLK_DIV 1
```

Now for the Quad-SPI protocol we need to select Double Data Rate enabled and QSPI Prescaler set to 1, so QSPI frequency is 110 MHz / 1 = 110 MHz.

```
#TCSETPAR QSPI_BANK 1 -> In this case this parameter is not used
#TCSETPAR QSPI_DDR_ENABLE YES
#TCSETPAR QSPI_SPI_PRESCALER 1
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

STM32H750IB_2x_IS25LP128F Execution with Blank External Memories 32MB

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[1] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[2] IDR: 0x54770002, Type: AMBA APB2 or APB3 bus.
AP[0] ROM table base address 0xE00FE000.
CPUID: 0x411FC271.
Implementer Code: 0x41 - [ARM].
Found Cortex M7 revision r1p1.
Cortex M7 Core halted [0.005 s].
Device configuration: [0x1416AAFC]
 * The device's RDP level is 0 [0xAA].
 * Brownout reset threshold 3 (VBOR3).
 * IWDG1 watchdog is controller by software.
 * No reset generated when entering DStop mode on D1 domain.
 * No reset generated when entering DStandby mode on D1 domain.
 * Independent watchdog keep running in System Stop mode.
 * Independent watchdog keep running in Standby mode.
 * Security feature disabled.
 * Product working in the full voltage range, I/O speed optimization at low-voltage disabled.
 * Flash bank 1 and bank 2 are not swapped.
PLL not enabled.
Requested Clock is 37.50 MHz.
Generated Clock is 37.50 MHz.
Good samples: 4 [Range 4-7].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 37.50 MHz.
Time for Connect: 0.110 s.
>|
---#IFERR TPCMD BLANKCHECK X
PLL settings configuration:
 * Selected Oscillator: CSI.
 * SYSCLK: ((4MHz/pllM)*pllN)/pllP -> 440MHz.
 * Clock after DC1PRE 440MHz.
 * Clock after DC2PRE 220MHz.
 * Clock to QUADSPI module 220.00MHz.
> Base system clock initialized.
Initialize QSPI GPIO pins.
> QSPI GPIO pins are initialized.
Enabling STM32 QSPI peripheral.
 * Clock prescaler is 1, QSPI peripheral clock is: 110.00MHz.
 * Flash Size is 32 MiB.
 * Dual Flash Mode: Enabled.
> STM32 QSPI peripheral enabled.
Read and check external memories.
 * QSPI Bank 1: Memory ID: Expected 0x18609D - Read 0xFFFFFF.
 * QSPI Bank 2: Memory ID: Expected 0x18609D - Read 0xFFFFFF.
 * Trying to read the Memory ID through QUAD-SPI protocol.
 * QSPI Bank 1: Memory ID: Expected 0x18609D - Read 0x18609D.
 * QSPI Bank 2: Memory ID: Expected 0x18609D - Read 0x18609D.
 * QSPI Bank 1: Status register 0x40.
 * QSPI Bank 2: Status register 0x40.
 * Clean memory status register.
 * QSPI Bank 1: Status register 0x00.
 * QSPI Bank 2: Status register 0x00.
 * QSPI Bank 1 and 2: SFDP table supported.
 * QSPI Bank 1 and 2: Flash size check passed: 2x 16MiB.
> Completed read and check external memories.
Configure external memories.
 * Switching from SPI to QUAD-SPI protocol.
 * QSPI Bank 1: Memory ID: Expected 0x18609D - Read 0x18609D.
 * QSPI Bank 2: Memory ID: Expected 0x18609D - Read 0x18609D.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
 * DDR [Double Data Rate] mode enabled.
 * Set eight dummy cycles.
 * Use 4-Byte address mode operation.
> External memories are configured.
Execute selected command...
External Memory enter Memory Mapping Mode.
 * DDR [Double Data Rate] enabled.
> Enter Memory Mapping Mode completed.
Time for Blankcheck X: 0.262 s.
>|
---#TPCMD PROGRAM X
FRB CRC32 check passed.
FRB Headers collected.
External Memory enter Direct Mode.
> Enter Direct Mode completed.
Time for Program X: 14.269 s.
>|
---#TPCMD VERIFY X R
External Memory enter Memory Mapping Mode.
 * DDR [Double Data Rate] enabled.
> Enter Memory Mapping Mode completed.
Time for Verify Readout X: 13.096 s.
>|
---#TPCMD VERIFY X S
External Memory is already in Memory Mapping Mode.
Time for Verify Checksum 32bit X: 0.183 s.
>|
---#TPCMD DISCONNECT
>|
```

STM32H750IB_2x_IS25LP128F Programming Times with Blank External Memories 32MB

Operation	Timings FlashRunner 2.0
Time for Connect	0.110 s
Time for Blankcheck External Flash	0.262 s
Time for Program External Flash	14,269 s
Time for Verify Readout External Flash	13,069 s
Time for Verify Checksum External Flash	0.183 s
Cycle Time	**00:27.980 s**

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

STM32H750IB_2x_IS25LP128F Execution with Not Blank External Memories 32MB

```
---#TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Trying Hot Plug connect procedure.
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[1] IDR: 0x84770001, Type: AMBA AHB3 bus.
AP[2] IDR: 0x54770002, Type: AMBA APB2 or APB3 bus.
AP[0] ROM table base address 0xE00FE000.
CPUID: 0x411FC271.
Implementer Code: 0x41 - [ARM].
Found Cortex M7 revision r1p1.
Cortex M7 Core halted [0.005 s].
Device configuration: [0x1416AAFC]
 * The device's RDP level is 0 [0xAA].
 * Brownout reset threshold 3 (VBOR3).
 * IWDG1 watchdog is controlled by software.
 * No reset generated when entering DStop mode on D1 domain.
 * No reset generated when entering DStandby mode on D1 domain.
 * Independent watchdog keep running in System Stop mode.
 * Independent watchdog keep running in Standby mode.
 * Security feature disabled.
 * Product working in the full voltage range, I/O speed optimization at low-voltage disabled.
 * Flash bank 1 and bank 2 are not swapped.
PLL not enabled.
Requested Clock is 37.50 MHz.
Generated Clock is 37.50 MHz.
Good samples: 4 [Range 4-7].
IDCODE: 0x6BA02477.
Designer: 0x23B, Part Number: 0xBA02, Version: 0x6.
ID-Code read correctly at 37.50 MHz.
Time for Connect: 0.110 s.
>|
---#IFERR TPCMD BLANKCHECK X
PLL settings configuration:
 * Selected Oscillator: CSI.
 * SYSCLK: ((4MHz/pllM)*pllN)/pllP -> 440MHz.
 * Clock after DC1PRE 440MHz.
 * Clock after DC2PRE 220MHz.
 * Clock to QUADSPI module 220.00MHz.
> Base system clock initialized.
Initialize QSPI GPIO pins.
> QSPI GPIO pins are initialized.
Enabling STM32 QSPI peripheral.
 * Clock prescaler is 1, QSPI peripheral clock is: 110.00MHz.
 * Flash Size is 32 MiB.
 * Dual Flash Mode: Enabled.
> STM32 QSPI peripheral enabled.
Read and check external memories.
 * QSPI Bank 1: Memory ID: Expected 0x18609D - Read 0xFFFFFF.
 * QSPI Bank 2: Memory ID: Expected 0x18609D - Read 0xFFFFFF.
 * Trying to read the Memory ID through QUAD-SPI protocol.
 * QSPI Bank 1: Memory ID: Expected 0x18609D - Read 0x18609D.
 * QSPI Bank 2: Memory ID: Expected 0x18609D - Read 0x18609D.
 * QSPI Bank 1: Status register 0x40.
 * QSPI Bank 2: Status register 0x40.
 * Clean memory status register.
 * QSPI Bank 1: Status register 0x00.
 * QSPI Bank 2: Status register 0x00.
 * QSPI Bank 1 and 2: SFDP table supported.
 * QSPI Bank 1 and 2: Flash size check passed: 2x 16MiB.
> Completed read and check external memories.
Configure external memories.
 * Switching from SPI to QUAD-SPI protocol.
 * QSPI Bank 1: Memory ID: Expected 0x18609D - Read 0x18609D.
 * QSPI Bank 2: Memory ID: Expected 0x18609D - Read 0x18609D.
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

```
 * DDR [Double Data Rate] mode enabled.
 * Set eight dummy cycles.
 * Use 4-Byte address mode operation.
> External memories are configured.
Execute selected command...
External Memory enter Memory Mapping Mode.
 * DDR [Double Data Rate] enabled.
> Enter Memory Mapping Mode completed.
External memory is not blank.
0800A2CD!|
---#THEN TPCMD MASSERASE X
External Memory enter Direct Mode.
> Enter Direct Mode completed.
Time for Masserase X: 33.369 s.
>|
---#THEN TPCMD BLANKCHECK X
External Memory enter Memory Mapping Mode.
* DDR [Double Data Rate] enabled.
> Enter Memory Mapping Mode completed.
Time for Blankcheck X: 0.200 s.
>|
---#TPCMD PROGRAM X
External Memory enter Direct Mode.
> Enter Direct Mode completed.
Time for Program X: 13.707 s.
>|
---#TPCMD VERIFY X R
External Memory enter Memory Mapping Mode.
 * DDR [Double Data Rate] enabled.
> Enter Memory Mapping Mode completed.
Time for Verify Readout X: 13.097 s.
>|
---#TPCMD VERIFY X S
External Memory is already in Memory Mapping Mode.
Time for Verify Checksum 32bit X: 0.184 s.
>|
---#TPCMD DISCONNECT
>|
```

STM32H750IB_2x_IS25LP128F Programming Times with Not Blank External Memories 32MB

Operation	Timings FlashRunner 2.0
Time for Connect	0.110 s
Time for Masserase External Flash	33.369 s
Time for Blankcheck External Flash	0.200 s
Time for Program External Flash	13.707 s
Time for Verify Readout External Flash	13.097 s
Time for Verify Checksum External Flash	0.184 s
Cycle Time	**01:00.793 s**

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

STM32F427ZI_1x_MX29GL640E

The memory must be connected to the chip enable 1. This memory is setup as FMC for the microcontroller.

MX29GL640E is a parallel Nor memory.

STM32F427ZI_1x_MX29GL640E Execution with Not Blank External Memory 8MB

```
---#TPCMD CONNECT
01|2|250703-09:02:01.850|Protocol selected SWD.
01|2|250703-09:02:01.850|Entry Clock is 4.00 MHz.
01|2|250703-09:02:01.850|Trying Hot Plug connect procedure.
01|2|250703-09:02:01.851|Good samples: 16 [Range 0-15].
01|2|250703-09:02:01.851|IDCODE: 0x2BA01477.
01|1|250703-09:02:01.851|Designer: 0x23B, Part Number: 0xBA01, Version: 0x2.
01|2|250703-09:02:01.851|ID-Code read correctly at 4.00 MHz.
01|1|250703-09:02:01.852|JTAG-SWD Debug Port enabled.
01|1|250703-09:02:01.852|Scanning AP map to find all APs.
01|1|250703-09:02:01.852|AP[0] IDR: 0x24770011, Type: AMBA AHB3 bus.
01|1|250703-09:02:01.852|AP[0] ROM table base address 0xE00FF000.
01|1|250703-09:02:01.852|CPUID: 0x410FC241.
01|1|250703-09:02:01.852|Implementer Code: 0x41 - [ARM].
01|1|250703-09:02:01.852|Found Cortex M4 revision r0p1.
01|1|250703-09:02:01.853|Program counter value is 0x20726568.
01|1|250703-09:02:01.853|Cortex M4 Core halted [0.002 s].
01|1|250703-09:02:01.854|> Device configuration: [0x0FFFAAED]
01|1|250703-09:02:01.854| * The device's Readout Protection level is 0 [0xAA].
01|1|250703-09:02:01.854| * BOR off (VBOR0), POR/PDR reset threshold level is applied.
01|1|250703-09:02:01.854| * Software independent watchdog selected.
01|1|250703-09:02:01.854| * No reset generated when entering the Stop mode.
01|1|250703-09:02:01.854| * No reset generated when entering the Standby mode.
01|2|250703-09:02:01.855|PLL enabled using internal HSI oscillator.
01|2|250703-09:02:01.855|Requested Clock is 25.00 MHz.
01|2|250703-09:02:01.856|Generated Clock is 25.00 MHz.
01|2|250703-09:02:01.856|Good samples: 6 [Range 3-8].
01|2|250703-09:02:01.856|IDCODE: 0x2BA01477.
01|1|250703-09:02:01.856|Designer: 0x23B, Part Number: 0xBA01, Version: 0x2.
01|2|250703-09:02:01.856|ID-Code read correctly at 25.00 MHz.
01|2|250703-09:02:01.856|Time for Connect: 0.558 s.
01|2|250703-09:02:01.856|>|
01|2|250703-09:02:01.856|---#TPCMD MASSERASE X
01|1|250703-09:03:03.794|Time for Masserase X: 61.939 s.
01|2|250703-09:03:03.795|>|
01|2|250703-09:03:03.795|---#TPCMD BLANKCHECK X
01|1|250703-09:03:05.804|Time for Blankcheck X: 2.011 s.
01|2|250703-09:03:05.805|---#TPCMD PROGRAM X
01|1|250703-09:03:51.413|Time for Program X: 45.609 s.
01|2|250703-09:03:51.413|>|
01|2|250703-09:03:51.413|---#TPCMD VERIFY X R
01|1|250703-09:03:56.268|Time for Verify Readout X: 4.855 s.
01|2|250703-09:03:56.268|>|
01|2|250703-09:03:56.268|---#TPCMD VERIFY X S
01|1|250703-09:03:57.665|Time for Verify Checksum 32bit X: 1.398 s.
01|2|250703-09:03:57.666|>|
01|2|250703-09:03:57.666|---#TPCMD DISCONNECT
01|2|250703-09:03:57.668|>|
01|2|250703-09:03:57.669|---#TPEND
01|2|250703-09:03:57.671|>|
01|2|250703-09:03:57.672|>|#1|RUN STM32F427ZI_1x_MX29GL640E.prj
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

STM32F427ZI_1x_MX29GL640E Programming Times with Not Blank External Memory 8MB

Operation	Timings FlashRunner 2.0
Time for Connect	0.558 s
Time for Masserase External Mem	61.939 s
Time for Blankcheck External Mem	2.011 s
Time for Program External Mem	45.609 s
Time for Verify Readout External Mem	4.855 s
Time for Verify Checksum External Mem	1.398 s
Cycle Time	**116.370 s**

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

STM32 Driver Changelog

Info about driver version 1.00 - 25/10/2019 [legacy]
Supported Flash memory Commands for STM32 L0 and L4 series.
Supported Flash memory Commands for STM32 G0 series.

Info about driver version 1.01 - 05/12/2019 [legacy]
Supported Flash memory Commands for STM32 L1 series.

Info about driver version 1.02 - 11/12/2019 [legacy]
Supported Flash memory Commands for STM32 F0 series.

Info about driver version 1.03 - 20/12/2019 [legacy]
Supported Flash memory Commands for STM32 F1 and F2 series.

Info about driver version 1.04 - 29/01/2020 [legacy]
Supported Option Bytes Commands for STM32 F4 series.

Info about driver version 2.00 - 06/02/2020 [legacy]
Added new SWD FPGA Static 9 and upgrade code performances.

Info about driver version 2.01 - 12/02/2020 [legacy]
Implemented Restore Option Bytes for supported STM32 series.
Implemented Overview Option Bytes for supported STM32 series.

Info about driver version 2.02 - 17/02/2020 [legacy]
Implemented Get and Set Protection for supported STM32 series.

Info about driver version 2.03 - 19/02/2020 [legacy]
Fixed Option Bytes program for STM32 L0 and L1 series.

Info about driver version 2.04 - 22/02/2020 [legacy]
Supported Flash and Option Bytes Commands for STM32 L4 series.
Improved code stability and performances.
Supported all STM32 G0 series.

Info about driver version 2.05 - 10/03/2020 [legacy]
Implemented EEprom Program for STM32 L0 series.
Upgraded code, Added Restore Option Bytes for STM32 L0 series.
Implemented Get Device Informations command for STM32 L4 series.
Added Set and Get Protection commands for STM32 G0 series.
Implemented Option Byte Program and verify for STM32 G0 series.

Info about driver version 2.06 - 15/03/2020 [legacy]
Supported standard Flash and Option Byte Commands for STM32 F1 XL-density series.

Info about driver version 2.07 - 17/03/2020 [legacy]
Supported standard Flash and Option Byte Commands for STM32 H7 series.

Info about driver version 2.08 - 21/03/2020 [legacy]
Supported standard Flash and Option Byte Commands for STM32 F4 series.

Info about driver version 2.09 - 25/03/2020 [legacy]
Upgraded code for F2, H7 and F4 STM32 series.

Info about driver version 2.10 - 28/03/2020 [legacy]
Upgraded code for F0, F1, F2 and supported STM32 F3 series.

Info about driver version 2.11 - 30/03/2020 [legacy]
Improved Masserase command and added Overview Option Bytes for STM32 F4 series.
Enabled JTAG for STM32 F4 series.
Implemented Restore Option Bytes STM32 F4 series.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Info about driver version 2.12 - 08/04/2020 [legacy]
Enabled JTAG for STM32 F3 series.
Upgraded performances for all STM32 families.
Supported FLASH operation for STM32 F7 series.

Info about driver version 2.13 - 14/04/2020 [legacy]
Improved code performance and increased stability of various operations.

Info about driver version 2.14 - 19/04/2020 [legacy]
Supported all memory Commands for STM32 L4 plus series.

Info about driver version 2.15 - 21/04/2020 [legacy]
Supported all memory Commands for STM32 G4 series.
Implemented Option Byte Program and verify for STM32 L4 plus series.

Info about driver version 2.16 - 11/05/2020 [legacy]
Supported all memory Commands for STM32 L5 series.
Implemented Option Byte Program and verify for STM32 G4 series.
Implemented Option Byte Program and verify for STM32 L5 series.

Info about driver version 2.17 - 13/05/2020 [legacy]
Supported OTP memory area for STM32 F2 series.
Added Get Device Informations command for STM32 F2 series.
Added Get Device Informations command for STM32 F3 series.
Page Erase for STM32 F1 series works well.

Info about driver version 2.18 - 16/05/2020 [legacy]
Added Page Erase for STM32 F7 series Single and Dual Bank.
Added Get Device Informations for STM32 F7 series.
Enabled JTAG for STM32 F7 series.
Upgraded Halt Cortex Core M33 procedure for STM32 L5 series.
Added Set Protection and Get Protection for STM32 L5 series.
All Option Bytes commands are supported for STM32 L5 series.

Info about driver version 2.19 - 06/07/2020 [legacy]
Fixed JTAG FPGA, now JTAG works fine for all STM32 series.

Info about driver version 2.20 - 24/07/2020 [legacy]
Improved SWD Connect Sequence.
Added Halt Cortex Core Mx procedure with Hardware reset option.

Info about driver version 2.21 - 20/08/2020 [legacy]
Added new SWD FPGA to improve code performances.

Info about driver version 3.00 - 07/09/2020 [legacy]
Upgraded SWD Connect Sequence and auto-alignment.
Refactored some internal procedure to better manage device protection.

Info about driver version 3.01 - 11/09/2020 [legacy]
Fixed Option Bytes verify Readout for STM32 F7 series.

Info about driver version 3.02 - 30/09/2020 [legacy]
Added RUN command for all STM32 devices.
Info about driver version 4.00 - 20/11/2020
Added new SWD and JTAG FPGA Static 9.
Added new SWD and JTAG FPGA to support FlashRunner HS.
Improved code performance for all operations.

Info about driver version 4.01 - 24/11/2020
Internal code update.

Info about driver version 4.02 - 25/11/2020
Added the new SWD FPGA optimized.
Upgraded Masserase command for STM32 F0 series to manage Flash Sector Write Protected.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Added UNPROTECT command for STM32 F0 series.
Supported Flash and Option Bytes Commands for STM32 H7 series.

Info about driver version 4.03 - 19/01/2021
Improved code performance for all operations.
Fixed Option Byte Management for STM32 H7 series.

Info about driver version 4.04 - 12/02/2021
Fixed print order for Get Device Informations command.

Info about driver version 4.05 - 01/04/2021
Added definitive SWD FPGA for FlashRunner HS, GP2 and GP4 active modules.
Fixed a bug in the level 1 RDP removal procedure for the STM32 L1 series added in version 4.01.

Info about driver version 4.06 - 10/04/2021
Improved code performance for all STM32 device through SWD protocol.
Updated some prints that can be observed in Realtime Log and improved internal code management.

Info about driver version 4.07 - 13/05/2021
Supported STM32G0 Dual Bank device.
Supported UNPROTECT command for STM32G, STM32L, STM32F and STM32H series.
Added support of Option Bytes at address 0x40022068 and 0x4002206C for STM32L5 family.
Added support of Option Bytes at address 0x1FFF0020 and 0x1FFF0028 for STM32F76x and STM32F77x Dual Bank.
Added support of Option Bytes for all STM32H7 series.
Added UNPROTECT command to Workbench (GUI) for all STM32.
Added ERASE [start address] [size] command to Workbench (GUI) for all STM32.
Added WRITE_OPTION_BYTE [address] [value] command to Workbench (GUI) for all STM32.
Added COMPARE_OPTION_BYTE [address] [value] command to Workbench (GUI) for all STM32.
Added CONNECT_UNDER_RESET option to make a hardware reset before connect procedure.
Fixed ERASE [address] [size] command for STM32L4+ and STM32H7 family.

Info about driver version 4.08 - 19/07/2021
Internal update for FPGA name and version tracking.

Info about driver version 4.09 - 29/07/2021
Upgraded management of Option Bytes complementary part for STM32L0x and STM32L1x series.

Info about driver version 4.10 - 12/08/2021
Upgraded Connect Under Reset Procedure.
Internal upgrade of the algorithm, no change to the operations it performs.

Info about driver version 4.11 - 27/08/2021
Connection procedure updated, now STM32 driver automatically find the best entry sequence.
Removed Connect Under Reset Option from Workbench.

Info about driver version 4.12 - 01/09/2021
Fast Programming and Standard Programming Modes available and selectable via **#TCSETPAR** PROGRAM MODE [F/S] [Fast/Standard] for STM32G0 and STM32G4 series.

Info about driver version 4.13 - 28/09/2021
Fixed Flash Size management for STM32L151xx with DEV_ID = 0x429.

Info about driver version 4.14 - 27/10/2021
Internal update, upgraded management for reset and halt Cortex core procedure.

Info about driver version 4.15 - 13/11/2021
Supported new STM32 32-bit Arm Cortex MCUs STM32U5 series.

Info about driver version 4.16 - 25/01/2022
Added READ_MEM8, READ_MEM16, READ_MEM32 commands.
New Option Bytes management for STM32G0 Dual Bank devices.
New QSPI peripheral management for STM32H7 series.
Prevent RAM parity error if parity checking is enabled.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Info about driver version 4.17 - 07/02/2022
Upgraded and optimized QSPI peripheral management for STM32H7 series.

Info about driver version 4.18 - 04/03/2022
Fixed STM32G4 Masserase when dual bank mode is set to 0 into OPTCR.
Added delay time for STM32G4 series of 50ms after Option Byte Launch.

Info about driver version 4.19 - 14/03/2022
Internal upgrade of the algorithm, no change to the operations it performs.

Info about driver version 4.20 - 16/03/2022
Fixed STM32H7 QSPI clock frequency print into Real Time Log.

Info about driver version 4.21 - 25/03/2022
The STM32 driver is able to manage QUAD-SPI communication with an external memory that is already configured in QPI mode at startup.

Info about driver version 4.22 - 15/04/2022
Added new SWD-UART FPGA with improved UART mode.
Fixed STM32G4 PLL calibration value.
Upgraded connect procedure for STM32 devices with new print message in case of error.
Fully supports STM32WB series (STM32WBx0, STM32WBx5 and STM32WBxM) of STM32 Wireless MCUs.
Added PLL support for STM32WB series.
Added FUS (Firmware Service Routine) commands for the STM32WB series (required SMH 0S version higher than v. 3.19).
Added PLL support for STM32U5 series.

Info about driver version 4.23 - 20/04/2022
Supported STM32 devices with JTAG protocol.

Info about driver version 4.24 - 27/04/2022
Supported STM32WL series (STM32WL5x and STM32WLEx) of STM32 Wireless MCUs.

Info about driver version 4.25 - 13/05/2022
Upgraded Restore Option Bytes and Set Protection for STM32H7 series when the device has RDP level set to 1.

Info about driver version 4.26 - 16/06/2022
Print current FPGA version loaded into TPSTART command.
Upgraded internal code to align all drivers.
Upgraded FUS procedure for all STM32WBx series.

Info about driver version 4.27 - 27/06/2022
Upgraded FUS procedure for STM32WBxx series.

Info about driver version 5.00 - 27/07/2022
Added FPGA for new FlashRunner 2.0 models.

Info about driver version 5.01 - 29/09/2022
Fixed Read/Write16 operation for JTAG protocol only.

Info about driver version 5.02 - 25/10/2022
Upgraded procedure when the user set the maximum RDP level.
For example, using "#TPCMD SET_PROTECTION 0xCC Y", you can see that into the log:

```
---#TPCMD SET_PROTECTION 0xCC
═══════════════════════════════════════════════════════════ *** WARNING *** ═══════════════════════════════════════════════════════════
Are you sure to set Readout Protection Level to 0xCC?.
Level 2: Enabled debug/chip read protection.
 - All protections provided by Level 1 are active.
 - Booting from system memory is not allowed anymore.
 - JTAG, SWD, SWV (single-wire viewer) are disabled.
 - User option bytes can no longer be changed.
 - When booting from Flash memory, accesses to Flash memory and backup SRAM from user code are allowed.
Memory read protection Level 2 is an irreversible operation.
When Level 2 is activated, the level of protection cannot be decreased to Level 0 or Level 1.
The JTAG port is permanently disabled when Level 2 is active (acting as a JTAG fuse).
As a consequence, boundary scan cannot be performed.
If you are sure to set RDP to 0xCC, please use this command syntax: \" #TPCMD SET_PROTECTION 0xCC Y \".
═══════════════════════════════════════════════════════════════════════════════════════════════════════════════════════════════
```


UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

Info about driver version 5.03 - 28/10/2022
Added management for Hardware Watchdog enabled through Option Bytes into STM32F1 series.

Info about driver version 5.04 - 31/10/2022
Improved performances for STM32F7 series.

Info about driver version 5.05 - 04/11/2022
Connect command has been enhanced with a description of the device configuration for all STM32 devices.

Info about driver version 5.06 - 19/11/2022
Fixed Option Bytes program for STM32F0 series when WRPxx bits are not available.

Info about driver version 5.07 - 28/11/2022
Upgraded PLL management for STM32H72xxx and STM32H73xxx devices.

Info about driver version 5.08 - 29/11/2022
Upgraded procedure when the user set the maximum RDP level (level 2) for STM32WB and STM32WL series.

Info about driver version 5.09 - 16/12/2022
Fixed category check for STM32F0 series.

Info about driver version 5.10 - 21/12/2022
Supported STSPIN devices: *STSPIN32F0, STSPIN32F0251, STSPIN32F0252, STSPIN32F0601, STSPIN32F0602, STSPIN32F0A, STSPIN32F0B and STSPIN32G4*.

Info about driver version 5.11 - 17/04/2023
Supported STM32C0 series.
Supported new STM32U5 devices.
Fixed Page Erase command for STM32F100x devices.

Info about driver version 5.12 - 05/05/2023
Fixed small issue into SET_PROTOTECTION command for STM32H7 series introduced into driver version 5.11.

Info about driver version 5.13 - 22/05/2023
Supported STM32WBA series.
Supported STM32H5 series.
Added new commands for all STM32 devices:
#TPCMD GET_UNIQUE_ID
#TPCMD GET_FLASH_SIZE
#TPCMD GET_PACKAGE_ID
#TPCMD GET_DEVICE_ID
#TPCMD GET_REVISION_ID

Info about driver version 5.14 - 23/05/2023
Supported JTAG for STM32H7 series.

Info about driver version 5.15 and 5.16 - 24/07/2023
Added management for RDP value 0.5 when TrustZone is Enabled.
Added basic support for STM32H5.
Added RDP regression with password for STM32U5 series.

Info about driver version 5.17 and 5.18 - 10/08/2023
Internal driver upgrade.

Info about driver version 5.19 - 11/08/2023
Added new command #TPCMD FUS_CHECK_VERSION.

Info about driver version 5.20 - 11/09/2023
Fixed management for STM32F4 with 32KB RAM size.

Info about driver version 5.21 - 16/09/2023
Supported STM32WB1xx FUS operations.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Info about driver version 5.22 and 5.23 - 02/11/2023
Upgraded procedure when the user set the maximum RDP level (level 2) for STM32L4, STM32L4plus and STM32L5 series.

Info about driver version 5.24 - 27/11/2023
Supported new STM32WL devices.

Info about driver version 5.25 - 30/11/2023
Fixed BOR level check for STM32WB35x devices.
Improved reset management into Connect procedure.

Info about driver version 5.26 - 07/12/2023
Fixed Option Bytes programming procedure when JTAG/SWD frequency is very low and we need to wait a little bit more.

Info about driver version 5.27 - 12/12/2023
Fast Programming and Standard Programming Modes available and selectable via **#TCSETPAR** PROGRAM MODE [F/S] [Fast/Standard] for STM32L4 and STM32L4plus series.

Info about driver version 5.28 - 16/01/2024
Supported STM32U5F7/5G7 and STM32U5F9/5G9 subfamily of STM32U5 series.

Info about driver version 5.29 – 23/02/2024
Updated procedure for STM32U5 RDP regression with password from RDP level 2 and/or RDP level 1 to RDP level 0.
Updated Trust Zone Management for all STM32 device with Trust Zone.
Added procedure for STM32WBA RDP regression with password.
Added #TPCMD FUS_GET_VERSION command.
Added #TPCMD FUS_GET_WIRELESS_STACK_VERSION command.
Added #TPCMD FUS_CHECK_WIRELESS_STACK_VERSION command.
Updated STM32WB Firmware Upgrade Service procedure.
Supported new STM32 series.
Fixed management of Write Protected Sector for STM32H750xx with 128KB of Flash Memory.
Fixed management for STM32F3 halt Cortex core with breakpoint software.

Info about driver version 5.30 – 06/03/2024
Fixed management of Write Protected Sector for STM32H7xx with more than 1MB of Flash Memory.
Handled case where the firmware sets the SWDIO pin as Analog Input.
Updated some prints into Connect procedure with JTAG protocol.

Info about driver version 5.31 – 09/05/2024
Updated FPGA for JTAG protocol.
Updated Read and Dump execution for SWD and JTAG protocol.

Info about driver version 5.32 – 12/07/2024
Added #TPCMD GET_MEMORY_HASH command.
Added **#TCSETPAR** HASH_TYPE SHA512 parameter.
Updated PLL for STM32F4 devices.
Updated SWD and JTAG connect procedure to support new STM32 devices.

Info about driver version 5.33 – 16/08/2024
Supported STM32H5 devices with Debug Authentication complete procedure and SFI.
Added commands for debug authentication and SFI for STM32H5 devices:
#TPCMD DISCOVERY, #TPCMD PROVISIONING, #TPCMD REGRESSION, #TPCMD SECURE_FIRMWARE_INSTALL.
Added commands for product state and trust zone management for STM32H5 devices:
#TPCMD GET_PRODUCT_STATE, #TPCMD SET_PRODUCT_STATE, #TPCMD CHECK_PRODUCT_STATE.
#TPCMD GET_TRUST_ZONE, #TPCMD SET_TRUST_ZONE, #TPCMD CHECK_TRUST_ZONE.
Updated small details into connect procedure for all STM32.
Updated WRP write protection management for STM32U5 series into UNPROTECT command.

Info about driver version 5.34 – 25/10/2024
Supported new STM32C0 devices.
Fixed sector erase for STM32G0 dual bank devices.

Info about driver version 5.35 – 15/11/2024
Added support for new STM32H7Rx and STM32H7Sx

UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

Updated command #TPCMD WRITE_OPTION_BYTE with new optional parameter Reload Option Bytes TRUE/FALSE.
Upgraded procedure when the user set the maximum RDP level for all STM32 devices.
Fixed print '>' character on terminal for some specific commands. The '>' character is automatically replaced by '-' character.

Info about driver version 5.36 – 20/12/2024
Supported new STM32Ux series.

Info about driver version 5.37 – 24/01/2025
Supported MT25QL512 memory through STM32H7A3II device.

Info about driver version 5.38 – 24/02/2025
Managed correctly STM32L100xx device sub-family.
Added **#TCSETPAR** CONNECT_MODE <mode> parameter to manage unique connect entry.

Info about driver version 5.39 – 28/02/2025
Improved disconnect command.

Info about driver version 5.40 – 17/03/2025
Enhanced the Connect command to support devices with non-standard configurations.
Improved Option Bytes management for STM32H5 devices when Trust Zone is enabled.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com