

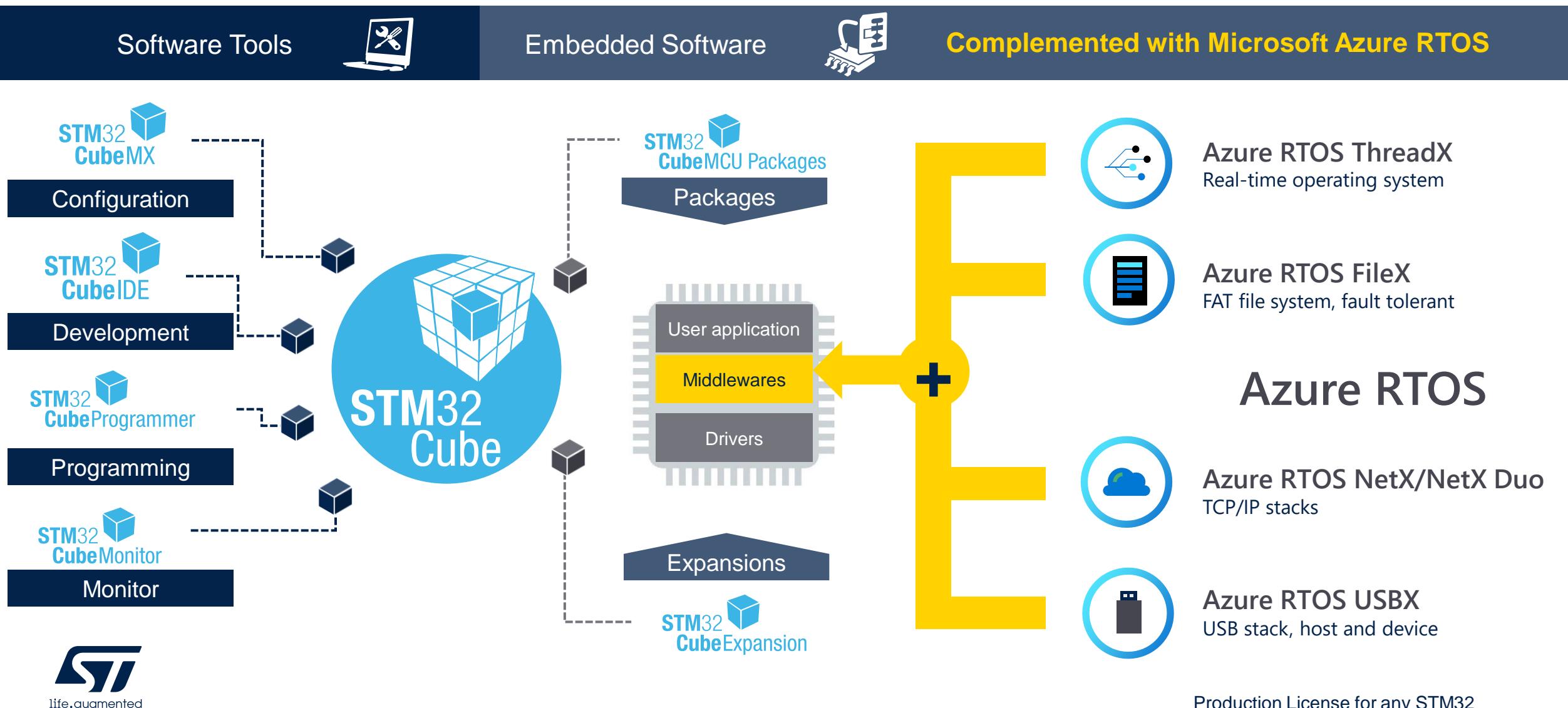


STM32Cube ecosystem overview

Making STM32 development easier



Inside the STM32Cube ecosystem

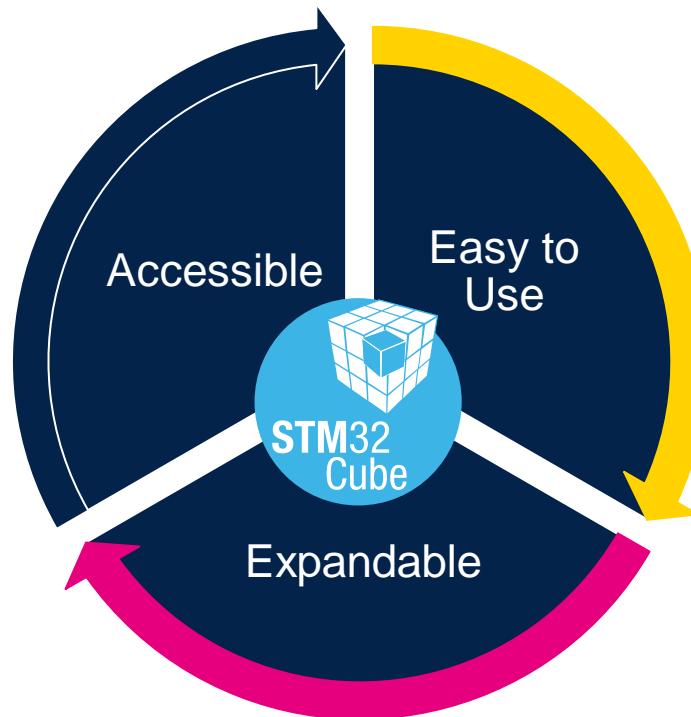


STM32Cube Ecosystem: User benefits

Easily getting the most out of STM32 MCUs for a drastically reduced customer development cycle and time-to-market

Fits many developer profiles from beginners to experts

- Exhaustive software development framework
- Free ST IDE (GCC) and professional IDE partners
- Free of charge and business-friendly license terms



Easy and fast learning curve for a competitive advantage

- Reduced time-to-market
- Allows focusing on applicative differentiation

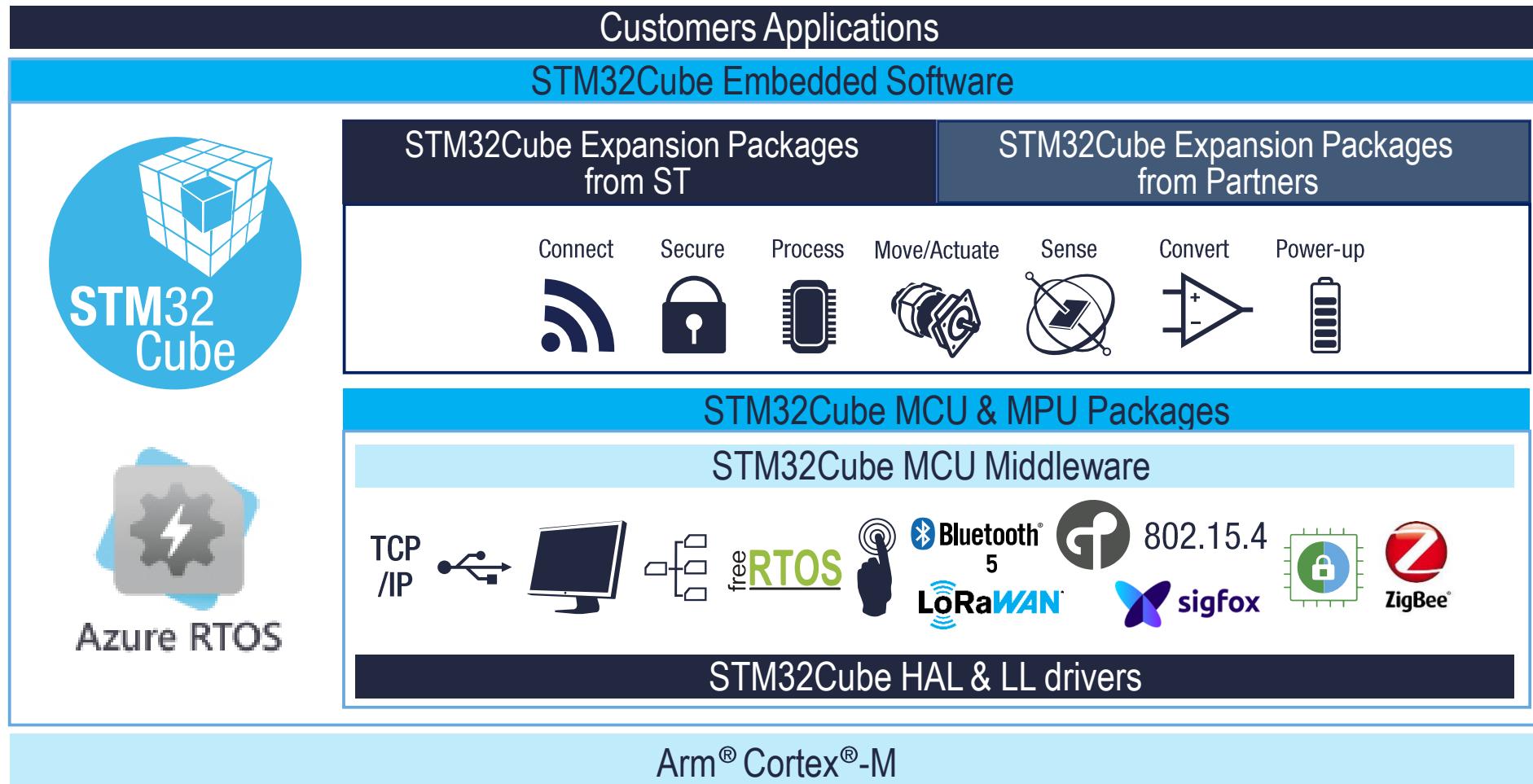
Fits many customer use cases

- Complemented by many solutions from official ST partners
- Production-ready

STM32Cube embedded software



A flexible, scalable and consistent MCU offer

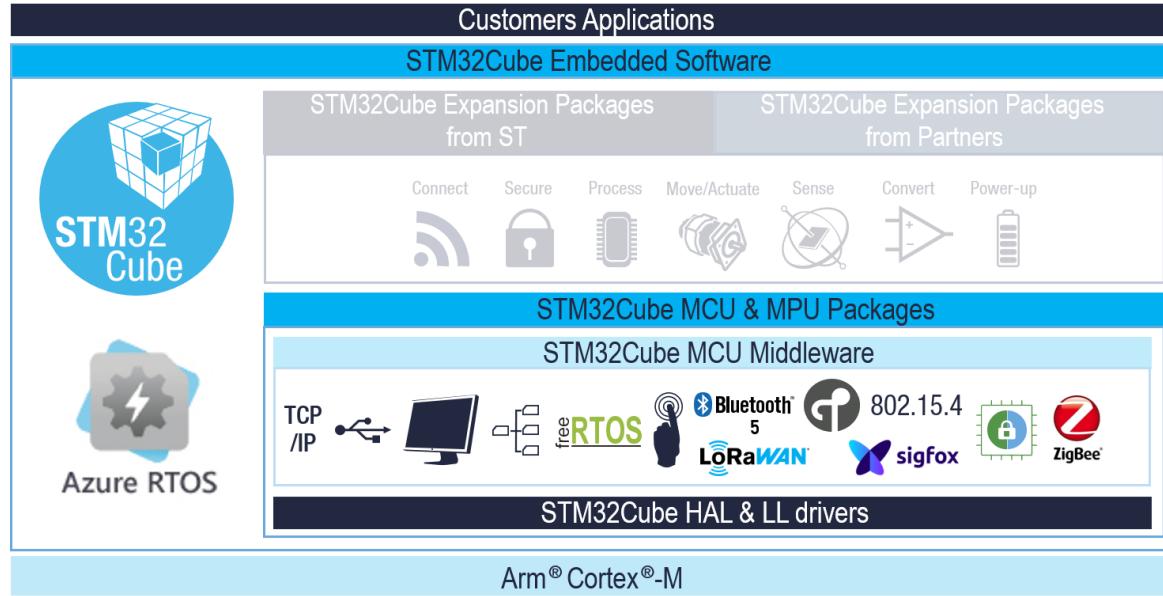


STM32 CubeMCU Packages

STM32Cube MCU packages

Dedicated to each STM32 Series

Mainstream MCU	High Performance MCU	MPU	Ultra-Low Power MCU	Wireless MCU
STM32  CubeG4	STM32  CubeH7	STM32  CubeMP1	STM32  CubeL0	STM32  CubeWB
STM32  CubeF3	STM32  CubeF7		STM32  CubeL1	STM32  CubeWL
STM32  CubeF1	STM32  CubeF4		STM32  CubeL4	
STM32  CubeG0	STM32  CubeF2		STM32  CubeL5	
STM32  CubeF0				



One-stop-shop SW packages

Peripheral drivers

HAL API

Hardware Abstraction Layer, highly portable and easy to use

LL APIs

Low-Layer APIs, light weight and highly optimized for runtime efficiency

STM32Cube Middleware

Generic MW

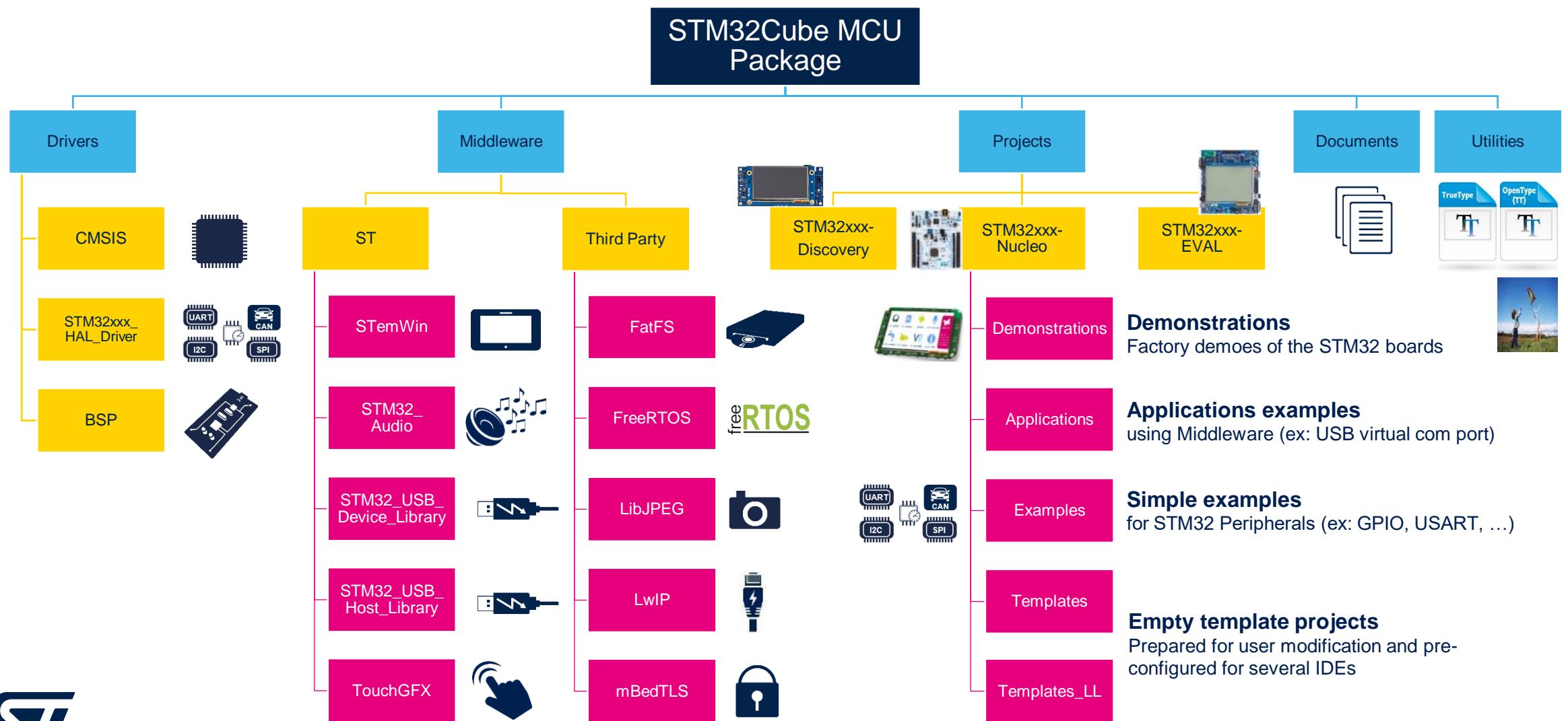
- FreeRTOS
- FatFS file system
- LwIP TCP/IP stack
- mbedTLS and mbedCrypto
- Open Bootloader

Dedicated MW

- ST Bluetooth 5 stack
- OpenThread stack
- ST 802.15.4 MAC
- Zigbee 3 stack
- STM32 WPAN
- LoRaWAN stack
- Sigfox stack
- Sub-GHz phy
- ST Key Management Services (KMS)

- TF-M
- ST USB Host & Device stacks
- STM32 Touch Sensing library
- STemWin graphics stack

Detailed content and organization



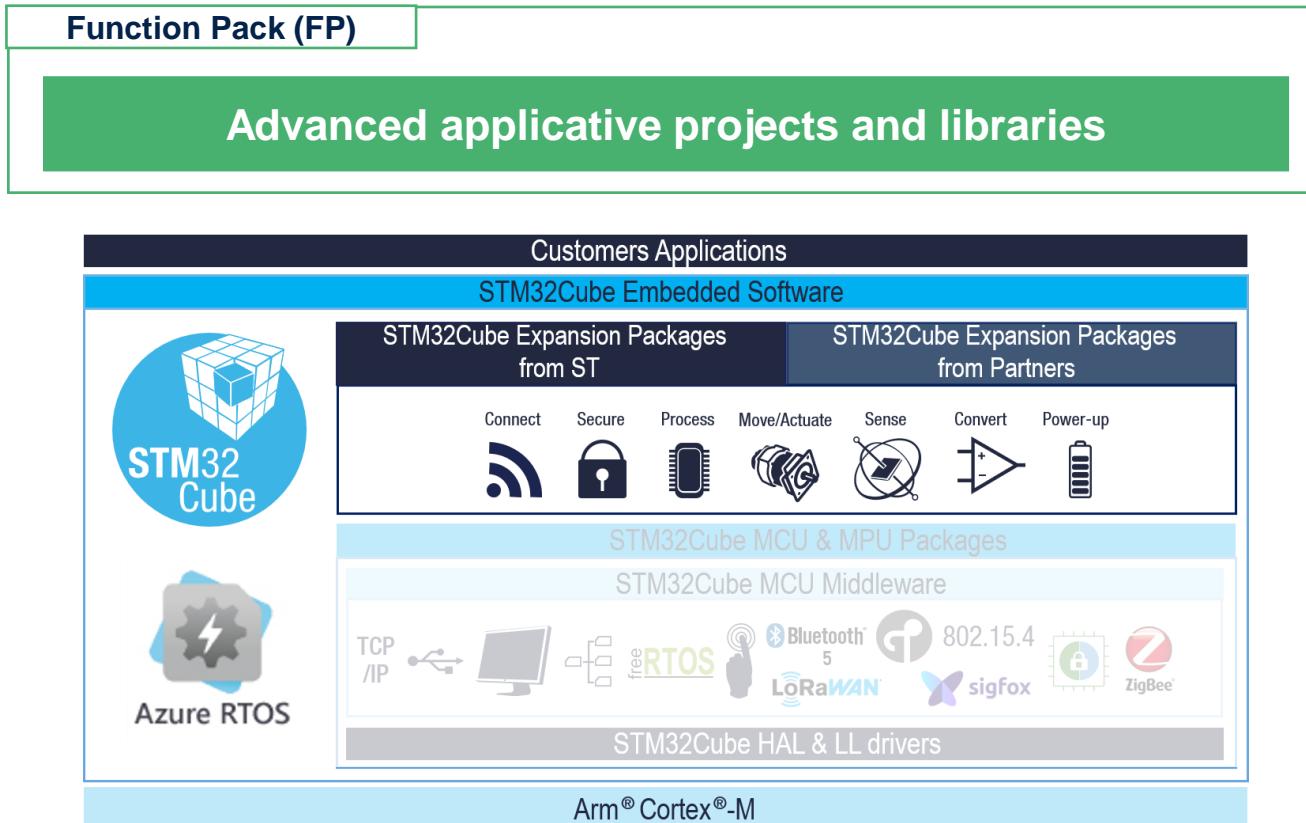
Middleware tailored for each series

Family	LL API	HAL API	FreeRTOS	FatFS	STemWin	USB Host	USB Device	LwIP mBedTLS	Touch Sense	USB PD	OpenAMP	BLE Stack	OpenThread Stack	Zigbee 3	LoRaWAN	Sigfox	TF-M
STM32CubeF0	✓	✓	✓	✓	✓		✓		✓								
STM32CubeF1	✓	✓	✓	✓	✓	✓	✓	✓	✓								
STM32CubeF2	✓	✓	✓	✓	✓	✓	✓	✓	✓								
STM32CubeF3	✓	✓	✓	✓	✓		✓		✓								
STM32CubeF4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓							
STM32CubeF7	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓							
STM32CubeH7	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			✓				
STM32CubeG0	✓	✓	✓	✓							✓						
STM32CubeG4	✓	✓	✓	✓			✓				✓						
STM32CubeL0	✓	✓	✓	✓				✓				✓					
STM32CubeL1	✓	✓	✓	✓	✓	✓	✓	✓	✓			✓					
STM32CubeL4	✓	✓	✓	✓	✓	✓	✓	✓	✓			✓					
STM32CubeL5	✓	✓	✓	✓	✓			✓			✓						✓
STM32CubeWB	✓	✓	✓	✓	✓			✓					✓	✓	✓		
STM32CubeWL	✓	✓	✓	✓	✓											✓	✓
STM32CubeMP1	✓	✓	✓										✓				

STM32Cube expansion packages

STM32
CubeExpansion

X-CUBE / I-CUBE / Function Pack (FP)



Complementing and expanding the STM32Cube MCU Packages with middleware alternatives or straightforward implementations of real applicative use cases

X-CUBE packages

From ST

Ex: X-CUBE-AZRTOS-H7, X-CUBE-AI, X-CUBE-TOUCHGFX, X-CUBE-SBSFU, X-CUBE-CRYPTO, ...

I-CUBE packages

From 3rd parties

Ex: I-CUBE-EMBOS, I-CUBE-UNISON, I-CUBE-CANOPEN...

Expansions with middleware



Azure RTOS

Azure RTOS

X-CUBE

X-CUBE-AZRTOS-H7

Enhanced for STM32 Toolset



Azure RTOS ThreadX
Real-time operating system



Azure RTOS NetX/NetX Duo
TCP/IP stacks

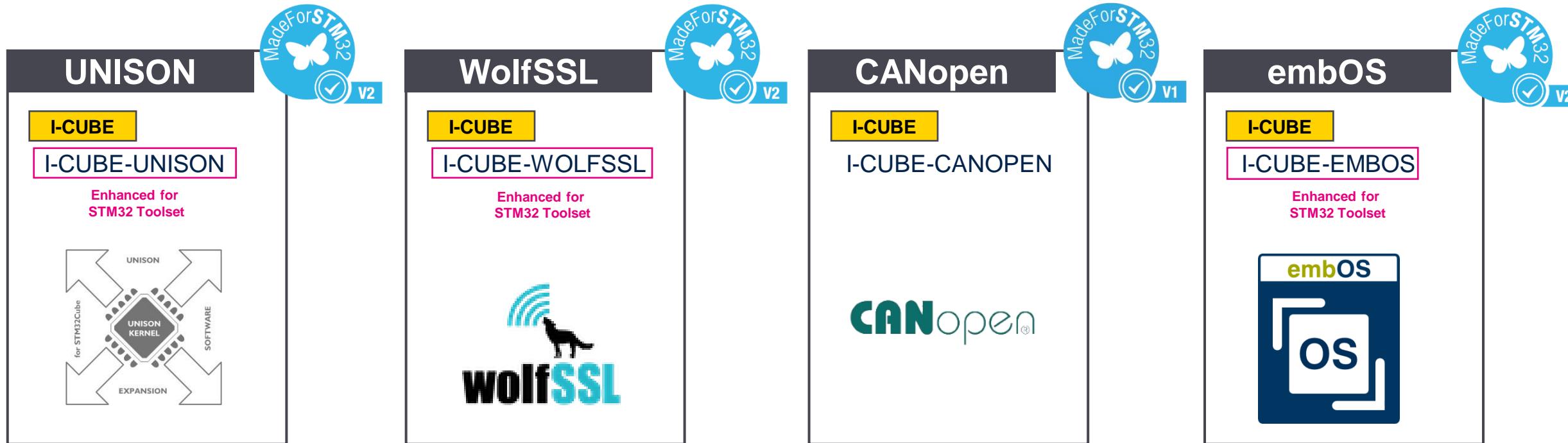


Azure RTOS FileX
FAT file system, fault tolerant



Azure RTOS USBX
USB stack, host and device

Expansions with middleware



Expansions for various applications

Audio



X-CUBE
X-CUBE-AUDIO
X-CUBE-VS4A
X-CUBE-USB-AUDIO

Bootloader/Secure Boot



X-CUBE
X-CUBE-IAP-USART
X-CUBE-IAP-SD
X-CUBE-SBSFU

Safety



X-CUBE
X-CUBE-CLASSB
X-CUBE-STL¹

Crypto



X-CUBE
X-CUBE-CRYPTOLIB

USB



X-CUBE
X-CUBE-USB-PD

SigFox



X-CUBE
X-CUBE-SFOX

LoRa



I-CUBE
I-CUBE-LRWAN

Sub-1G



X-CUBE
X-CUBE-SUBG1

BLE



X-CUBE
X-CUBE-BLE1 Enhanced for STM32 Toolset
X-CUBE-BLE2 Enhanced for STM32 Toolset

NFC



X-CUBE
X-CUBE-NFC4² Enhanced for STM32 Toolset

GRAPHICS



X-CUBE
X-CUBE-TOUCHGFX Enhanced for STM32 Toolset

(1) Contact STMicroelectronics local representative

(2) Non-exhaustive list, please refer to st.com

Expansions with Function Packs

Cloud

X-CUBE

X-CUBE-CLD-GEN1
X-CUBE-AWS
X-CUBE-AZURE
X-CUBE-WATSON
X-CUBE-GCP

FP

FP-CLD-AWS1
FP-CLD-AZURE1
FP-CLD-WASTON1



Motion

X-CUBE

X-CUBE-6180XA1
X-CUBE-IKA02A1
X-CUBE-MEMS-XT1
X-CUBE-MEMS1
X-CUBE-MEMS1-V4

FP

FP-SNS-6LPNODE1
FP-SNS-ALLMEMS1
FP-SNS-FLIGHT1
FP-SNS-MOTENV1



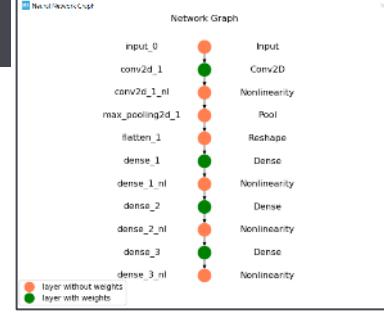
AI

X-CUBE

X-CUBE-AI

Enhanced for STM32 toolset

STM32 Cube.AI



FP

FP-AI-SENSING1
FP-AI-VISION1



Expansions for motor control

Motor - PMSM

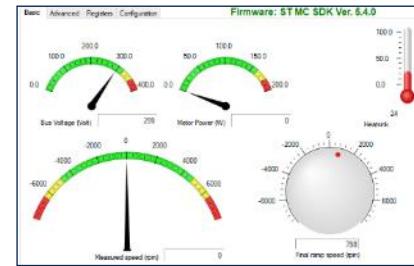
X-CUBE

X-CUBE-MCSDK

STM32 Motor Control SDK (MCSDK)

X-CUBE-MCSDK-FUL

STM32 Motor Control SDK (MCSDK) –
FULL source code –
Registration/approval needed for
download



X-CUBE-SPN7

Three-phase brushless DC motor driver

Motor - Stepper



X-CUBE

X-CUBE-SPN1

Stepper bipolar motor driver software expansion for STM32Cube

X-CUBE-SPN2

Two axes stepper motor driver software expansion for STM32Cube

X-CUBE-SPN3

High-power stepper motor driver software expansion for STM32Cube

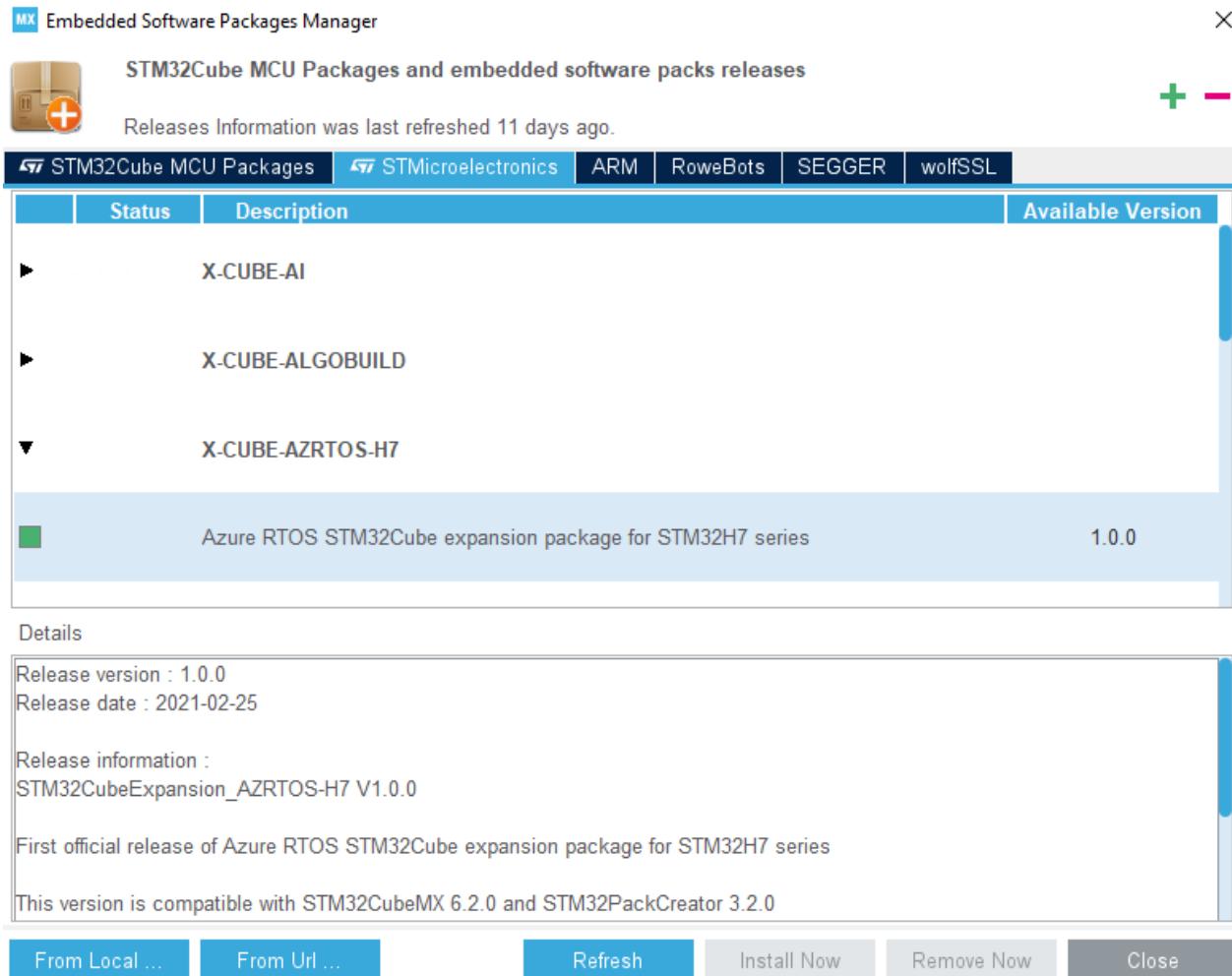
X-CUBE-SPN4

Dual-brush DC motor driver software expansion for STM32Cube

X-CUBE-SPN5

Stepper bipolar motor driver software expansion for STM32Cube

Expansions enhanced for STM32 Toolset



I-CUBE-EMBOS
I-CUBE-UNISON
I-CUBE-WOLFSSL
X-CUBE-AI
X-CUBE-ALGOBUILD
X-CUBE-BLE1
X-CUBE-BLE2
X-CUBE-EEPRMA1
X-CUBE-GNSS1
X-CUBE-MEMS1
X-CUBE-NFC4
X-CUBE-SUBG2
X-CUBE-SFXS2LP1
X-CUBE-TOUCHGFX



STM32Cube expansions on ST website

Google search results for "stm32cube expansion":

- [STM32 Embedded Software](https://www.st.com)
STM32Cube Expansion Packages - STMicroelectronics
The STM32Cube expansion software contains embedded software components that enable the usage of a multitude of ST devices in domains such as ...
- [STM32Cube Expansion Software - STMicroelectronics](https://www.st.com/ecosystems/stm32cube-expansion...)
The STM32Cube expansion software contains embedded software components that complement the functionalities of the STM32Cube and/or enable the usage ...
- [Development guidelines for STM32Cube Expansion Packages](https://www.st.com/resource/user_manual/d...)
1 sept. 2020 — STM32Cube low-layer APIs, a consistent set of middleware components, and all embedded software utilities. • **STM32Cube Expansion** ...
- [X-CUBE-AI - AI expansion pack for STM32CubeMX ...](https://www.st.com/.../STM32Cube Expansion Packages)
X-CUBE-AI is an STM32Cube Expansion Package part of the STM32Cube.AI ecosystem and extending STM32CubeMX capabilities with automatic conversion ...
- [I-CUBE-LRWAN - LoRaWAN software expansion for ...](https://www.st.com/.../STM32Cube Expansion Packages)
STM32CubeMX is part of the STM32Cube initiative designed to simplify and accelerate the development of applications for STM32 microcontrollers.

STM logo: life.augmented

STM32Cube Expansion Packages on ST website:

STM32Cube Expansion Packages

Overview Product selector Documentation

[View all STM32Cube Expansion Packages products](#)

STM32Cube Expansion Packages complement and build on the [STM32Cube MCU Packages](#) by delivering additional embedded software components that enable specific applicative use cases, as well as the implementation of STM32 MCUs and companion chips as well as multiple ST devices in domains such as sensing, power management, connectivity and audio.

The number of STM32Cube Expansion Packages available is continuously growing thanks to the extensive and scalable ecosystem around STM32, created by ST and Authorized Partners. The X-CUBE part numbers are offered and maintained by ST, and I-CUBE part numbers are developed by ST and its partners.

How to take advantage of STM32Cube

Today, developers can find ready-to-use components for various applications (e.g. IoT, AI, GCP, etc.), LoRa, Sigfox and cellular connectivity, motor control, sensor fusion, motor control algorithms, safety self-test functions, and more.

Now that the Azure RTOS middleware suite is available, it is easier than ever to develop real-time systems for the cloud.

STM32Cube Expansion Packages

Overview Product selector Documentation

149 total entries

Compare Part Number General Description Supplier Function Middleware MadeForSTM32 Enhanced for STM32 Toolset

FILTERS Reset Filters

Part Number Search ...

Supplier Function Middleware MadeForSTM32 Enhanced for STM32 Toolset

Feedback

More details →

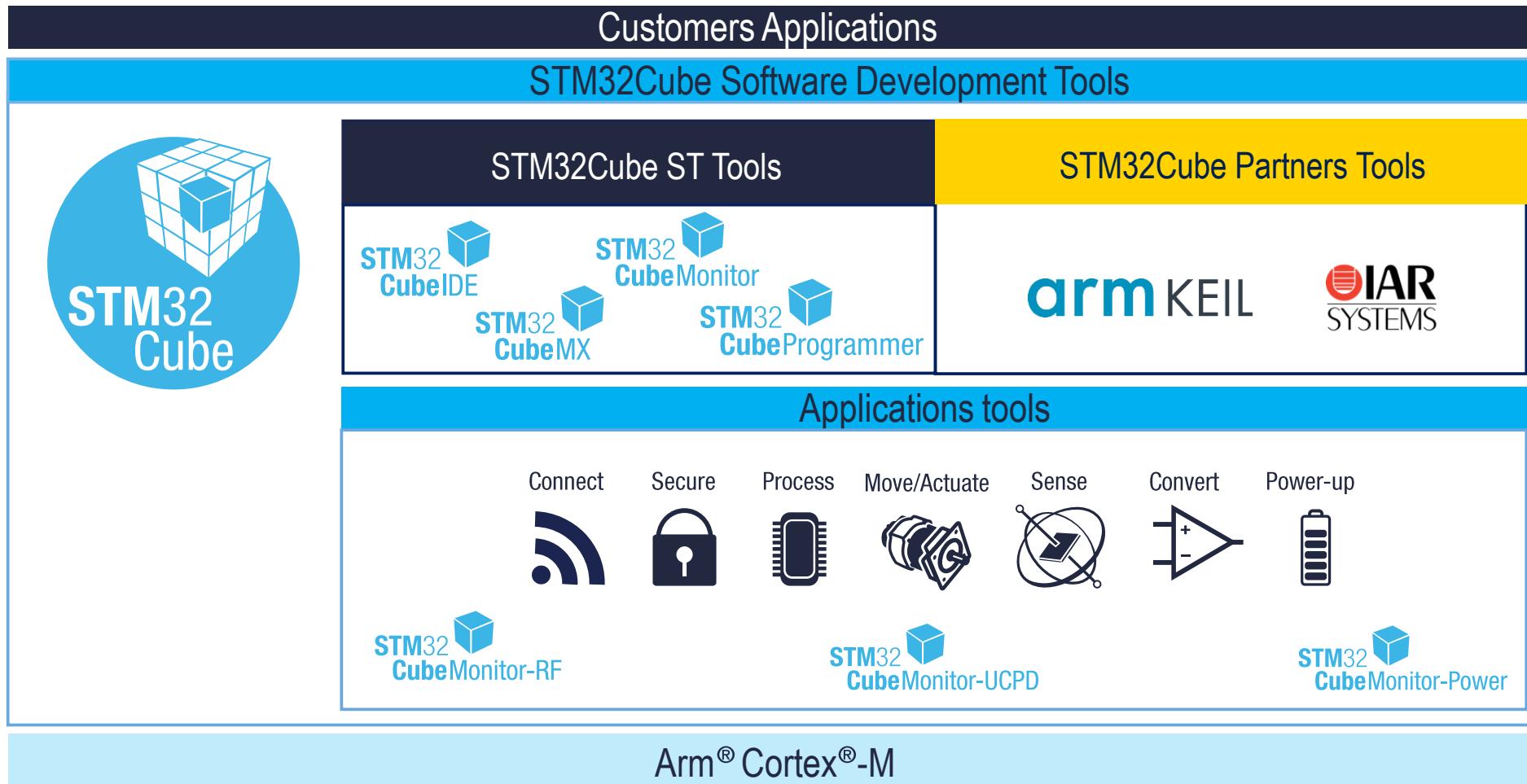
Part Number	General Description	Supplier	Function	Middleware	MadeForSTM32	Enhanced for STM32 Toolset
> FP-AI-NANOEDG1	Artificial Intelligence (AI) condition monitoring function pack for STM32cube	ST	-	-	-	-
> X-CUBE-AZRTOS-H7	Azure RTOS software expansion for STM32cube for STM32H7 series	ST	Connect	Azure RTOS, File system, KernelRTOS, MQTT, TCP/IP, TLS, USB	-	-
> X-CUBE-OPUS	Opus evaluation and profiling software expansion for STM32Cube	ST	-	-	-	-
> FP-AI-VISION1	STM32Cube function pack for high performance STM32 with artificial intelligence (AI) application for Computer Vision	ST	-	-	-	-
> FP-AUD-BVLINK1	STM32 ODE function pack for half-duplex voice streaming over Bluetooth low energy	ST	Connect, Sense	Audio, B	-	-

STM32Cube software

Development tools



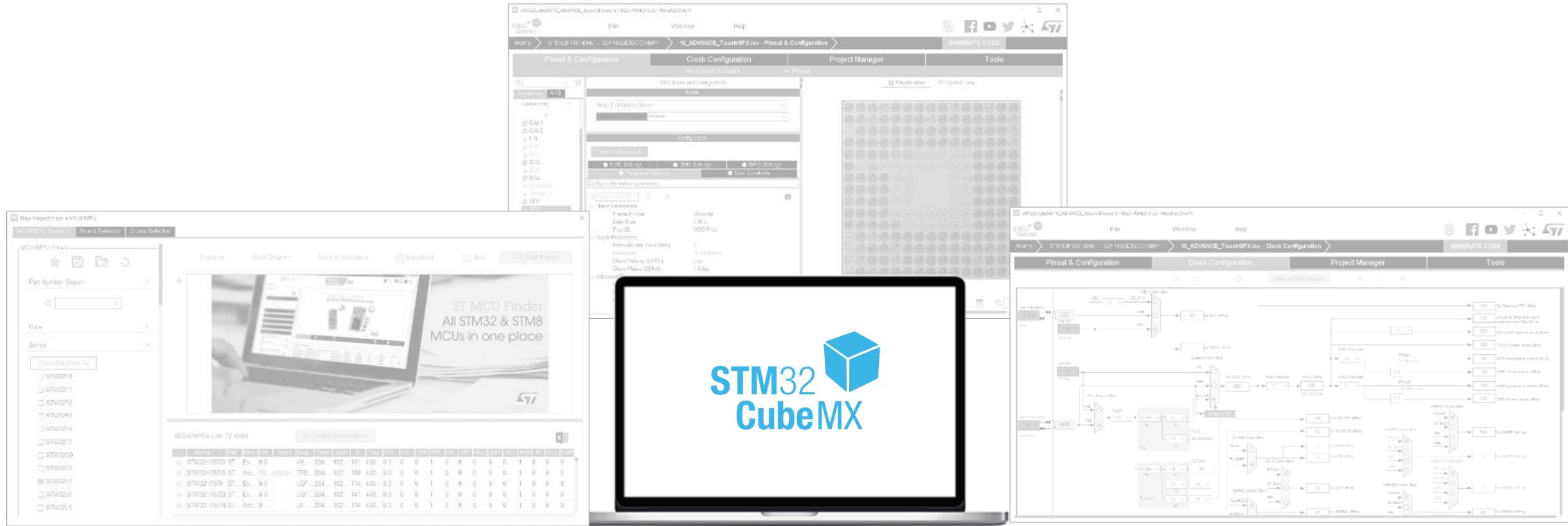
A complete Software Development Tools offer



STM32CubeMX

STM32
CubeMX

What's STM32CubeMX?

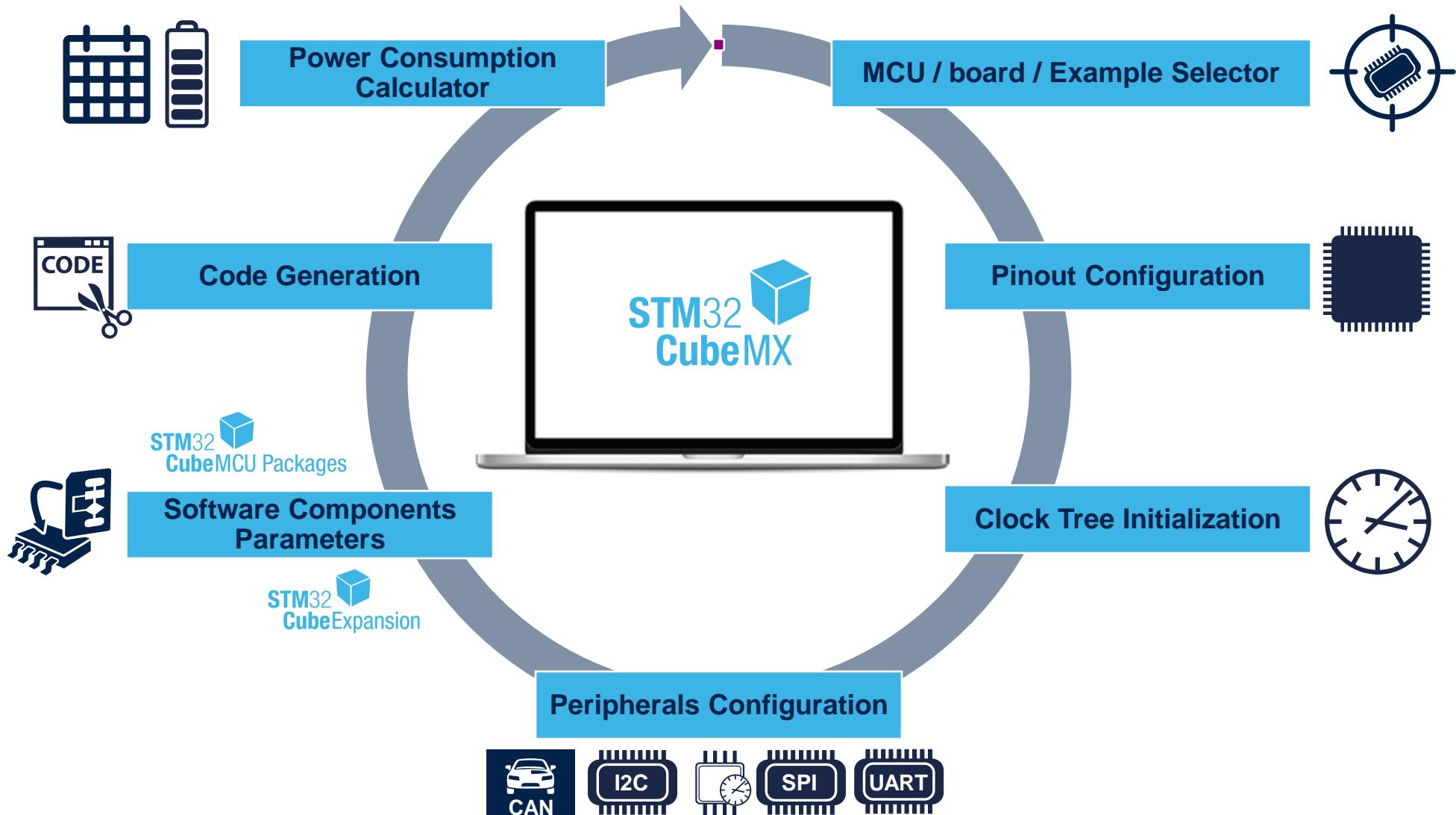


Graphical Configuration
(Pinout/Peripherals/Middleware/Clock-Tree)

IDE Project Generation
(IAR™, Keil™ and GCC compilers)

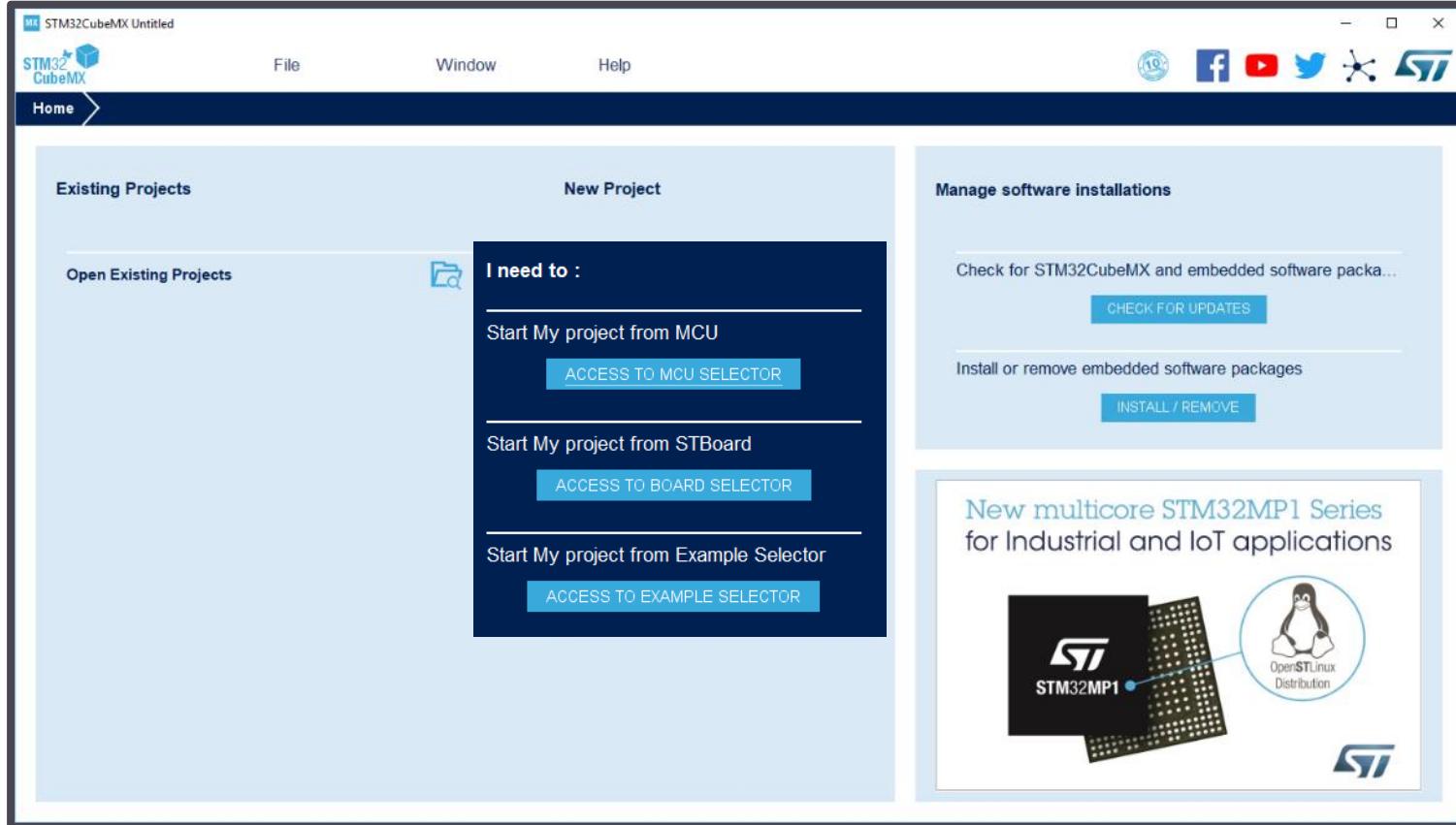
Multi-platform
(Windows, Linux, macOS)

STM32CubeMX key steps





MCU / MPU selection



MCU / MPU
SELECTOR

BOARD
SELECTOR

EXAMPLE
SELECTOR



MCU/MPU selector

MCU / MPU SELECTOR

BOARD SELECTOR

EXAMPLE SELECTOR

DEDICATED
FILTERS

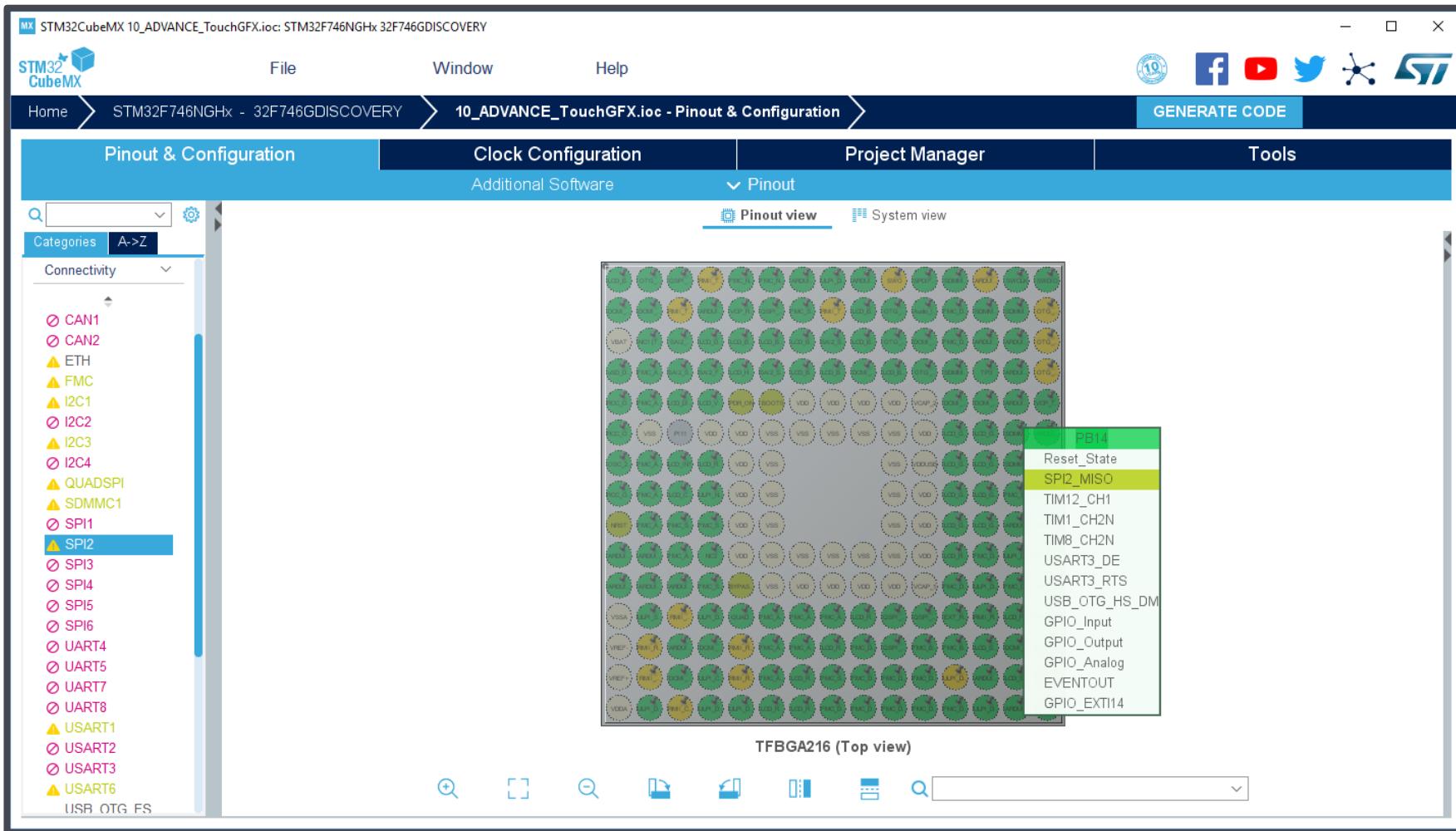
The screenshot shows the 'Example Selector' tab of the MCU/MPU selector interface. On the left, there are two sets of filters: 'Example Filters' and 'MCU / MPU'. The 'Example Filters' section includes fields for Name, Keyword, Vendor, Board, Name, Type, and checkboxes for Discovery Kit, Evaluation Board, Nucleo-144, Nucleo-32, and Nucleo-64. The 'MCU / MPU' section includes fields for Name, Series, and checkboxes for STM32F4, STM32F7, STM32G0 (which is checked), and STM32G4. The main area displays a project card for 'STM32G0' with 'SPI_FullDuplex_ComDMA_Slave' example. It shows details like Required Software Package (STM32Cube_FW_G0_V1.4.0), Vendor (STMicroelectronics), Board (STM32G081B-EVAL), and Mounted device (STM32G081RBTx). Below the card is a table titled 'Examples List: 632 items' with columns for Name, Board, Board Type, STM32Cube Version, Series, Project Type, and Driver. The first few rows of the table are listed below:

Name	Board	Board Type	STM32Cube Version	Series	Project Type	Driver
SPI_FullDuplex_ComDMA_Slave	STM32G081B-EVAL	Evaluation Board	V1.4.0	STM32G0	Example	HAL
SPI_FullDuplex_ComDMA_Slave	NUCLEO-G031KB	Nucleo-32	V1.4.0	STM32G0	Example	HAL
SPI_FullDuplex_ComIT_Master	NUCLEO-G070RB	Nucleo-64	V1.4.0	STM32G0	Example	HAL
SPI_FullDuplex_ComIT_Master	NUCLEO-G071RB	Nucleo-64	V1.4.0	STM32G0	Example	HAL
SPI_FullDuplex_ComIT_Master	STM32G081B-EVAL	Evaluation Board	V1.4.0	STM32G0	Example	HAL
SPI_FullDuplex_ComIT_Master	NUCLEO-G031KB	Nucleo-32	V1.4.0	STM32G0	Example	HAL
SPI_FullDuplex_ComIT_Slave	NUCLEO-G070RB	Nucleo-64	V1.4.0	STM32G0	Example	HAL
SPI_FullDuplex_ComIT_Slave	NUCLEO-G071RB	Nucleo-64	V1.4.0	STM32G0	Example	HAL
SPI_FullDuplex_ComIT_Slave	STM32G081B-EVAL	Evaluation Board	V1.4.0	STM32G0	Example	HAL
SPI_FullDuplex_ComIT_Slave	NUCLEO-G031KB	Nucleo-32	V1.4.0	STM32G0	Example	HAL
SPI_FullDuplex_ComPolling_Master	NUCLEO-G0B1RE	Nucleo-64	V1.4.0	STM32G0	Example	MIX
SPI_FullDuplex_ComPolling_Master	STM32G0C1E-EV	Evaluation Board	V1.4.0	STM32G0	Example	HAL

DESCRIPTION
&
INFORMATION

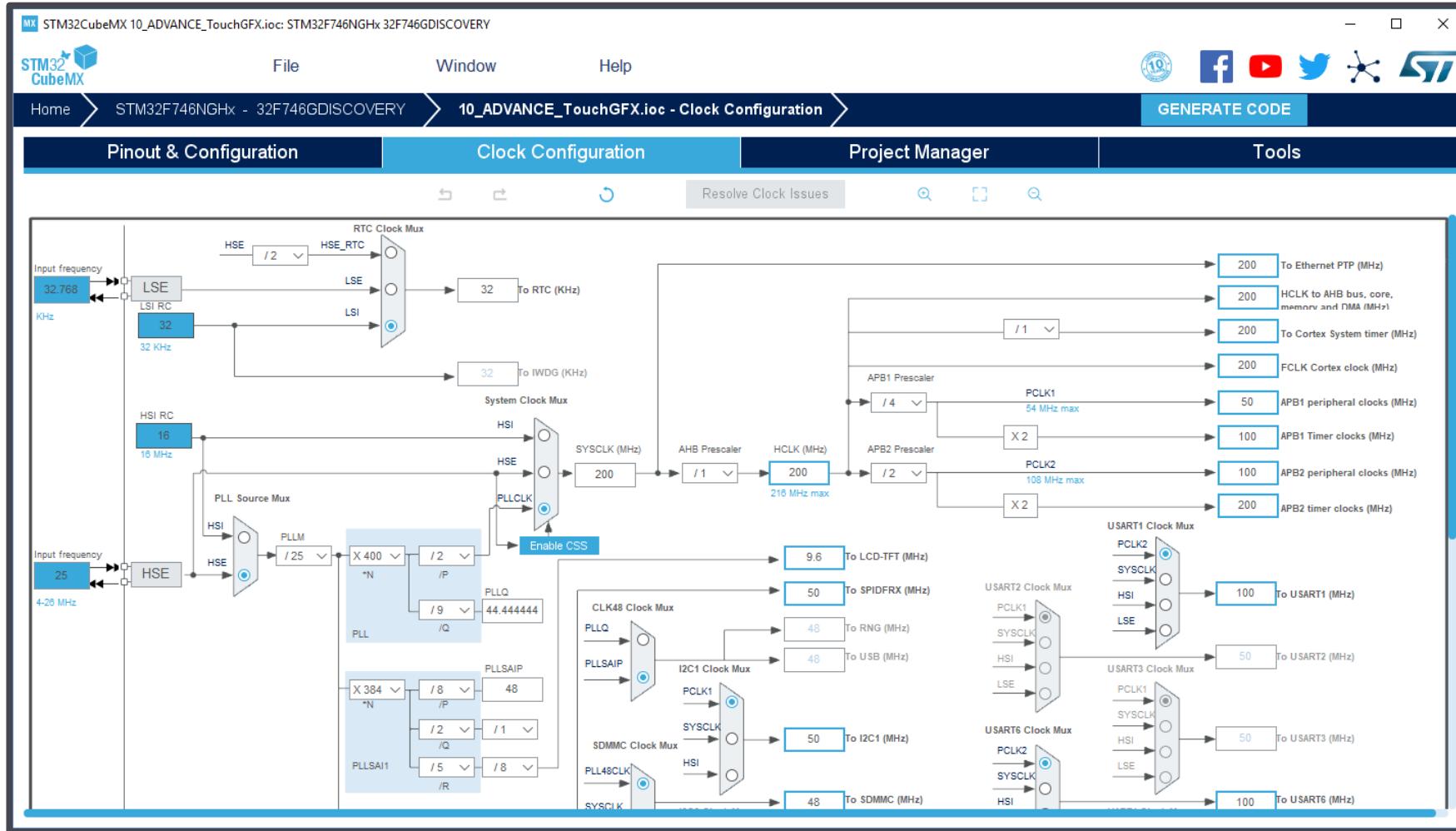
PRODUCT
LIST

Pinout configuration



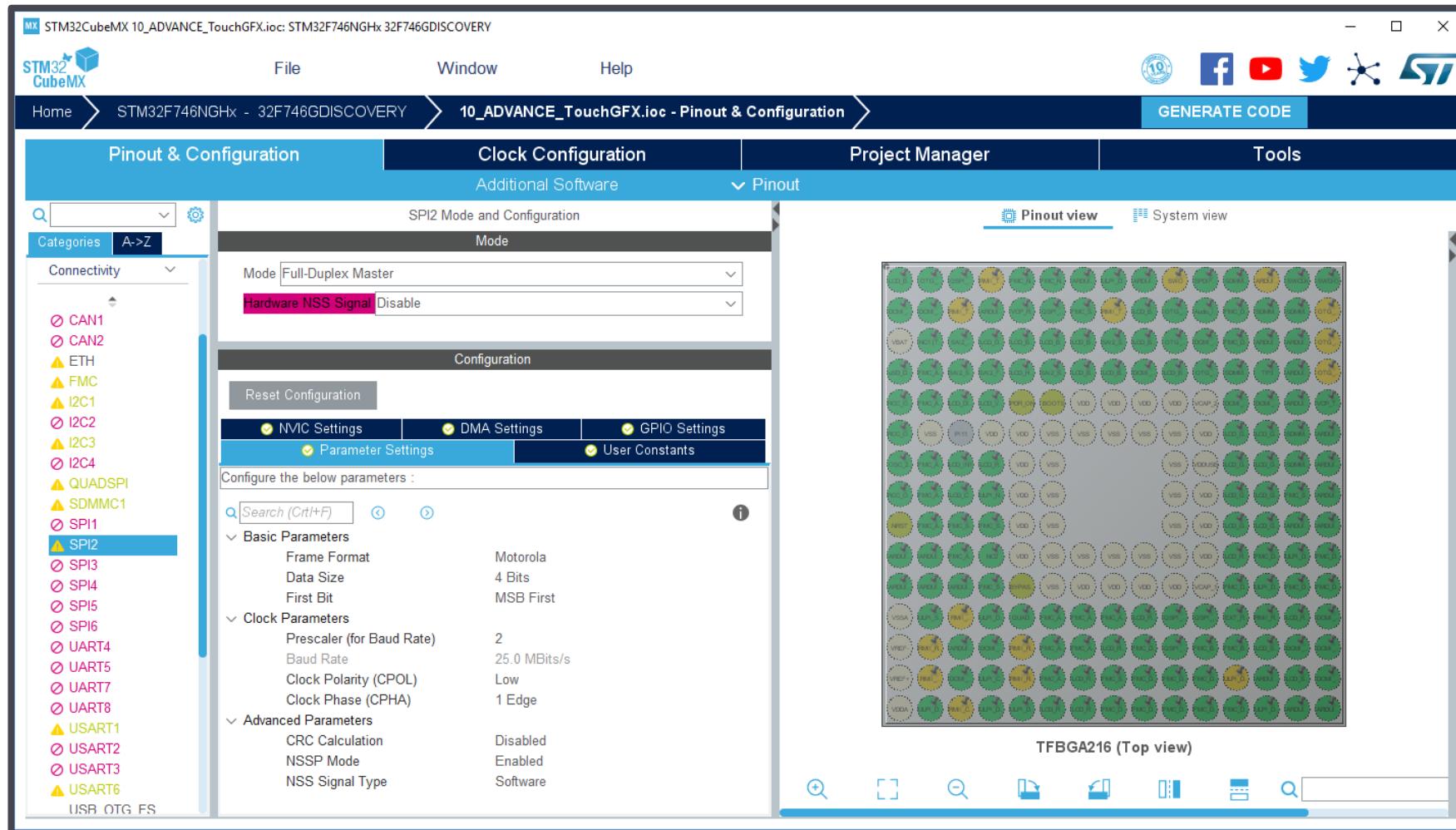


Clock-tree configuration





Peripheral parameters





Middleware and software components parameters

The screenshot shows the STM32CubeMX software interface. The title bar indicates the project is "STM32CubeMX 10_ADVANCE_TouchGFX.ioc: STM32F746NGHx 32F746GDISCOVERY". The main menu includes File, Window, Help, and various social media and ST logos.

The left sidebar lists categories: System Core, Analog, Timers, Connectivity, Multimedia, Security, Computing, Middleware (with sub-options: FATFS, FREERTOS, GRAPHICS, LIBJPEG, LWIP, MBEDTLS, PDM2PCM, USB_DEVICE, USB_HOST), and others.

The central area has tabs: Pinout & Configuration (selected), Clock Configuration, Project Manager, and Tools. The Pinout & Configuration tab shows "FREERTOS Mode and Configuration" with sections for Mode (Interface: CMSIS_V1) and Configuration (Reset Configuration, Mutexes, Tasks and Queues, Config parameters, FreeRTOS Heap Usage, Timers and Semaphores, Include parameters, User Constants). It also lists Kernel settings:

Setting	Value
USE_PREEEMPTION	Enabled
CPU_CLOCK_HZ	SystemCoreClock
TICK_RATE_HZ	1000
MAX_PRIORITIES	7
MINIMAL_STACK_SIZE	128 Words
MAX_TASK_NAME_LEN	16
USE_16_BIT_TICKS	Disabled
IDLE_SHOULD_YIELD	Enabled
USE_MUTEXES	Enabled
USE_RECURSIVE_MUTEXES	Enabled
USE_COUNTING_SEMAPHOR...	Enabled
QUEUE_REGISTRY_SIZE	8
USE_APPLICATION_TASK_TAG	Enabled

The right side displays the "Pinout view" for a TFBGA216 package, showing a grid of pins labeled with VDD, GND, and various digital and analog pin names. A "System view" tab is also present.



Add Expansion Middleware and build your own

The screenshot shows the STM32CubeMX Embedded Software Packages Manager interface. At the top, it displays "STM32Cube MCU Packages and embedded software packs releases". Below this, there's a search bar and tabs for "STM32Cube MCU Packages", "STMicroelectronics", and "ARM". A message indicates "Releases Information was last refreshed 1 days ago." The main area shows a list of packages under "X-CUBE-AI", specifically the "Artificial Intelligence" component, which is 23.42 MB in size. A modal window titled "Software Packs Component Selector" is open, showing a grid of components from various STM32Cube packages. One row for "Artificial Intelligence X-CUBE-AI" has its "Core" component selected and marked for installation. The bottom of the main window shows "Component dependencies" and "All conditions are solved".

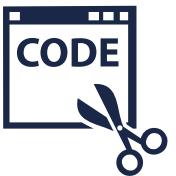
Download and install
existing Expansion Package...

...and select components to add to your project

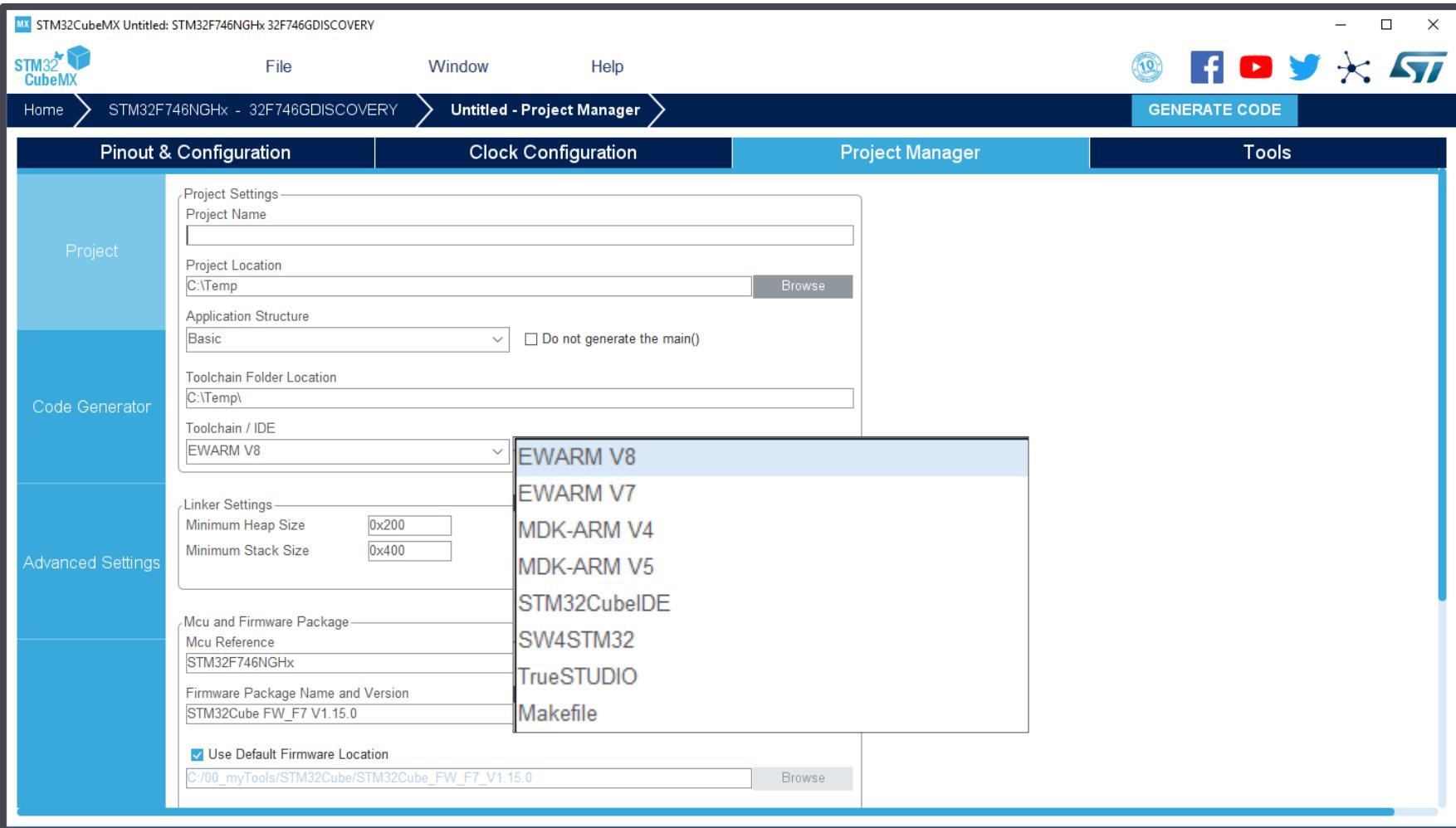
or

The screenshot shows the STM32PackCreator tool interface. It features a sidebar with options like "Add Parameter", "Add Platform Setting", "Add Custom Templates", and "Manage Mode Conditions". The main area is titled "Preview STM32CubeMX / Edit Tabs" and shows two tabs: "Mode" and "Configuration". Under "Mode", "Wireless BlueNRG-2" and "Device BLE2_Applications" are selected. Under "Configuration", there are sections for "Parameter Settings" and "Platform Settings", with various parameters like "HCI_READ_PACKET_SIZE" and "BLE2_DEBUG" being configured. A message at the bottom right says "No debug message".

Build your own Expansion Package
with **STM32PackCreator**



Code generation



STM32
CubelDE

IAR
SYSTEMS

arm KEIL



Power consumption calculator

STM32CubeMX 10_ADVANCE_TouchGFX.ioc*: STM32F746NGHx 32F746GDISCOVERY

STM32CubeMX File Window Help

Home > STM32F746NGHx - 32F746GDISCOVERY > 10_ADVANCE_TouchGFX.ioc - Tools > GENERATE CODE

Pinout & Configuration | Clock Configuration | Project Manager | Tools

Power

New Step Step Sequence

Sequence Table

Step	Mode	Vdd	Range/Scale	Memory	CPU/Bus Freq	Clock Config	Peripherals	Step Current	Duration
1	RUN	3.3	Scale1-High	SRAM/FLAS...	216 MHz	HSE PLL	ADC3 CRC D...	163.5 mA	1 ms
2	STOP_NM (N...)	3.3	No Scale	n/a	0 Hz	Regulator_LP ...	IWDG	270.25 μ A	1 ms

Edit Step

Reset Step Settings | Enable All IPs | Disable All IPs | Enable IPs from Pinout

Power/Memory

Power Mode: RUN

Power Scale: Scale1-High

Memory Fetch Type: SRAM/LASH-ART/L1Cache/REGON

V_{DD}: 3.3

Voltage Source: Battery

Clocks

CPU Frequency: 216 MHz

Interpolation Ranges

User Choice (Hz):

Clock Configuration: HSE PLL

Clock Source Frequency: 4 MHz

Optional Settings

Step Duration: 1 ms

Additional Consumption: 0 mA

Results

Step Consumption: 163.5 mA

Without Peripherals: 108 mA

Peripherals Part: 55.5 mA (A: 1.73 mA - D: 53.77 mA)

T_a Max (°C): 89.35

Warnings

Peripherals Selection

Enabled Peripherals:

- ADC1
- ADC2
- ADC3
- SRAM
- BusMatrix
- CAN
- CAN2
- CEC
- CRC
- CRYP
- DAC
- OUT1
- OUT1+OUT2
- OUT2
- DCMI
- DMA1
- 1_Streams
- 2_Streams
- 3_Streams
- 4_Streams
- 5_Streams
- 6_Streams
- 7_Streams
- 8_Streams
- DMA2
- 1_Streams
- 2_Streams

Display Selection

Select your Preferred Display Plot: All Steps

Consumption Profile by Step

Consumption (mA) vs Time (ms)

Plot: Idd by Step (Red Line) and Average Current (Blue Line)

Step 1: RUN (0.0 ms to 1.0 ms, Consumption ~163.5 mA)

Step 2: STOP_NM (1.0 ms to 2.1 ms, Consumption ~270.25 μ A)

Sequence Time / Ta Max: 2 ms / 89.35 °C

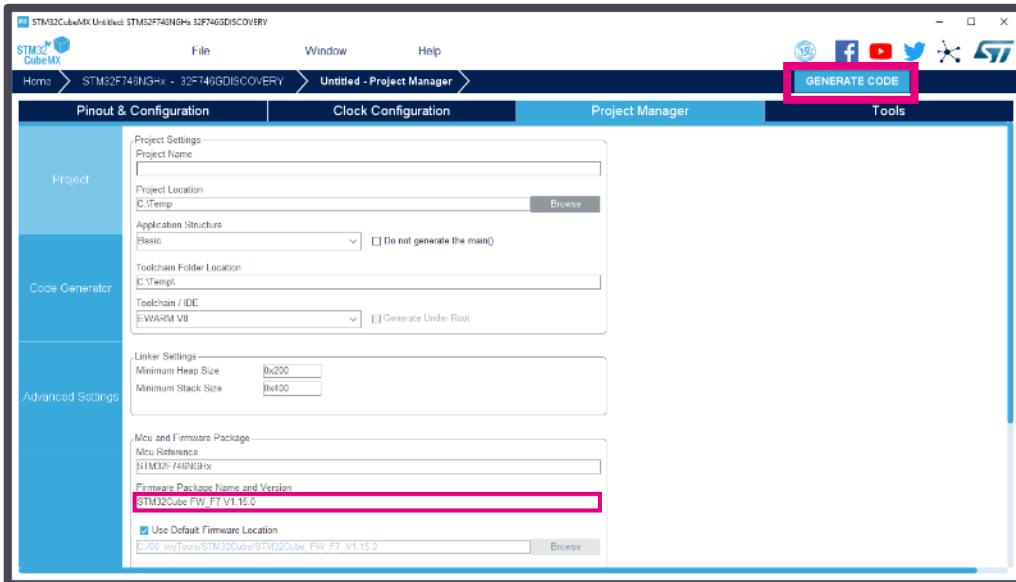
Average Consumption: 81.89 mA

Battery Life Estimation: 4 days, 5 hours

Average DMIPS: 462.24 DMIPS



STM32CubeMX repository



After pressing “GENERATE CODE”:

1. STM32CubeMX grabs necessary peripheral drivers based on your pinout/peripheral configuration from STM32Cube MCU Package in STM32CubeMX Repository
2. STM32CubeMX grabs necessary middleware based on your middleware configuration from STM32Cube MCU Package in STM32CubeMX repository
3. Generate IDE project

Description	Installed Version	Available Version
STM32Cube MCU Package for STM32F7 Series	1.15.0	1.15.0
STM32Cube MCU Package for STM32F7 Series	1.14.0	1.14.0
STM32Cube MCU Package for STM32F7 Series (Size : 584 MB)	1.12.0	
STM32Cube MCU Package for STM32F7 Series (Size : 569 MB)	1.11.0	

STM32CubeMX Repository



IDE Project

```
void SystemClock_Config(void)
{
    /* Header source clock configuration */
    RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};
    RCC_OscInitTypeDef RCC_OscInitStruct = {0};

    /* Configure LSE Drive Capability */
    HAL_PWR_EnableBxIoscPower();
    /* Configure the main system oscillator source selection */
    __HAL_RCC_PORRCCConfig();
    /* Initialize the HSE source clock */
    RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSE;
    RCC_OscInitStruct.HSEState = RCC_HSE_ON;
    RCC_OscInitStruct.PLL.PLLState = RCC_PLL_ON;
    RCC_OscInitStruct.PLL.PLLSource = RCC_PLLSOURCE_HSE;
    RCC_OscInitStruct.PLL.PLLM = 4;
    RCC_OscInitStruct.PLL.PLLN = 84;
    RCC_OscInitStruct.PLL.PLLP = 2;
    RCC_OscInitStruct.PLL.PLLQ = 2;
    RCC_OscInitStruct.PLL.PLLR = 2;
    if (HAL_RCC_OscConfig(&RCC_OscInitStruct) != HAL_OK)
    {
        Error_Handler();
    }
}
```



MCU/MPU selection for AI application

MX New Project from a MCU/MPU

MCU/MPU Selector | Board Selector | Cross Selector

Core >
Series >
Line >
Package >
Other >
Advanced Graphic >
Artificial Intelligence >

Enable

Model: Keras (highlighted)
Type: Saved model
Model: HAR-CNN-Kera....h5
Compression: 4
Analyze
Peripheral >

Features | Block Diagram | Docs & Resources | Datasheet | Buy | Start Project

New multicore STM32MP1 Series for Industrial and IoT applications

STM32MP1 (with a Linux logo) and OpenSTLinux Distribution

Graphic Summary | AI Summary

K Keras
Minimum Ram: 44.50 KBytes
Minimum Flash: 775.52 KBytes
C:\100_myProjects\AI_ML_DL\Trials\HAR-CNN-Keras_model.h5

Keras
Lasagne
Caffe
Convnetjs
List: 338 items
Display similar items

	Ref...	Mark...	Unit...	Board	Pack...	Flash	RAM	IO	Freq.	GFX...	CO...	DDR	DEB...	FM...	HDP	HMAC	MD5	PKA	PWR	RF	SHA	TAMP
★	STM32F405...	ST...	Act...4....	WL...	102...	192...	72	168...	0.0	0	0	0	0	0	0	0	0	0	0	0	0	0
★	STM32F405...	ST...	Act...4....	LQF...	102...	192...	51	168...	0.0	0	0	0	0	0	0	0	0	0	0	0	0	0
★	STM32F405...	ST...	Act...5....	LQF...	102...	192...	82	168...	0.0	0	0	0	0	0	0	0	0	0	0	0	0	0
★	STM32F405...	ST...	Act...5....	LQF...	102...	192...	114	168...	0.0	0	0	0	0	0	0	0	0	0	0	0	0	0
★	STM32F407IG	ST...	Act...6....	STM324...	UF...	102...	192...	140	168...	0.0	0	0	0	0	0	0	0	0	0	0	0	0

STM32CubeIDE

STM32
CubeIDE

Background of STM32CubeIDE

History

a atollc
TrueSTUDIO®

ST a atollc
TrueSTUDIO® for STM32

STM32
CubeMX

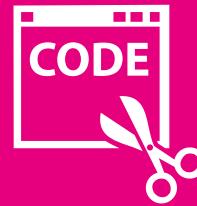


One tool for all your STM32 development

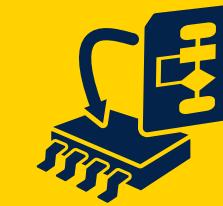
Chipset / Board Configuration



Code Development

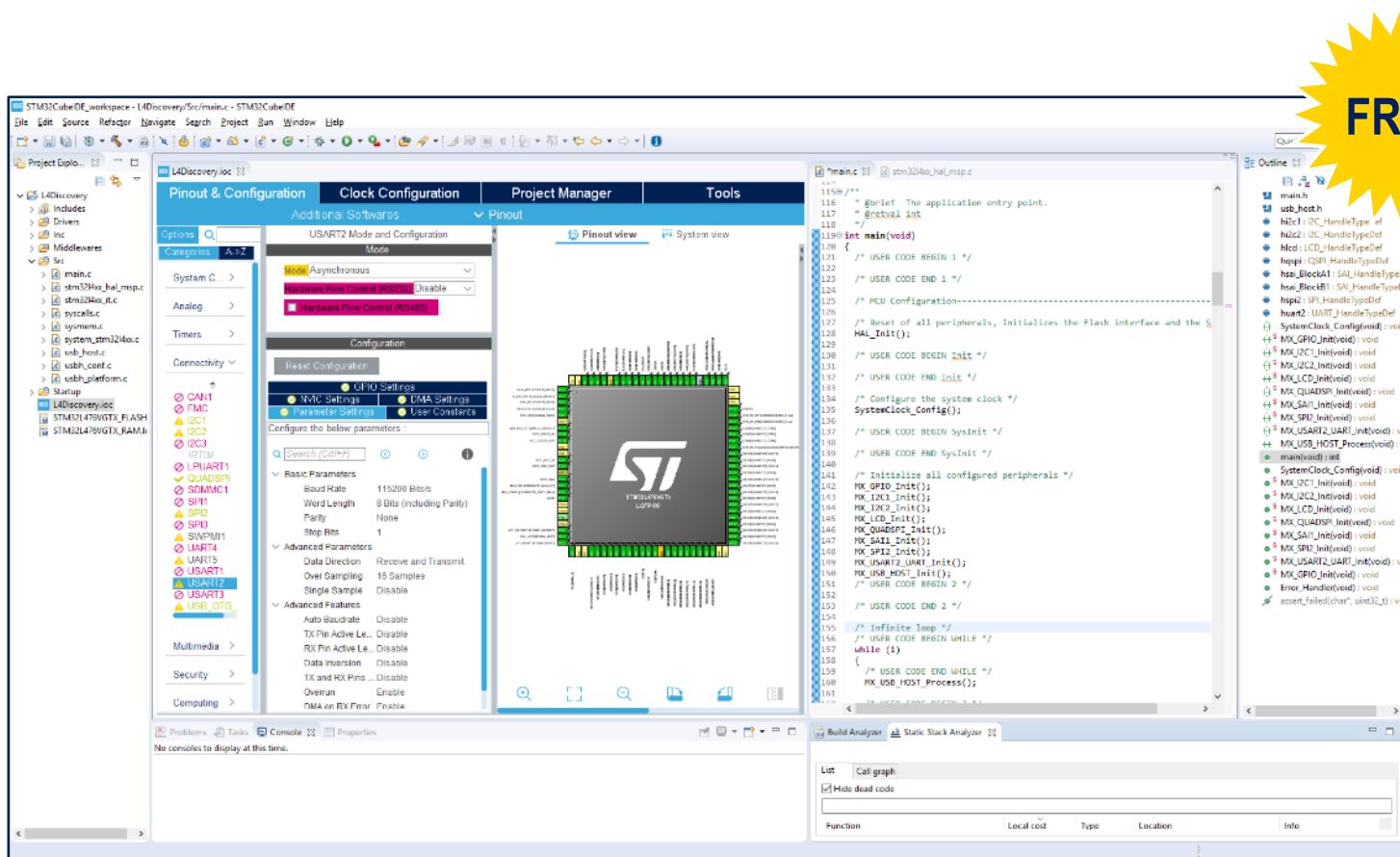


Validation Debug



Free multi-platform development tool

eclipse



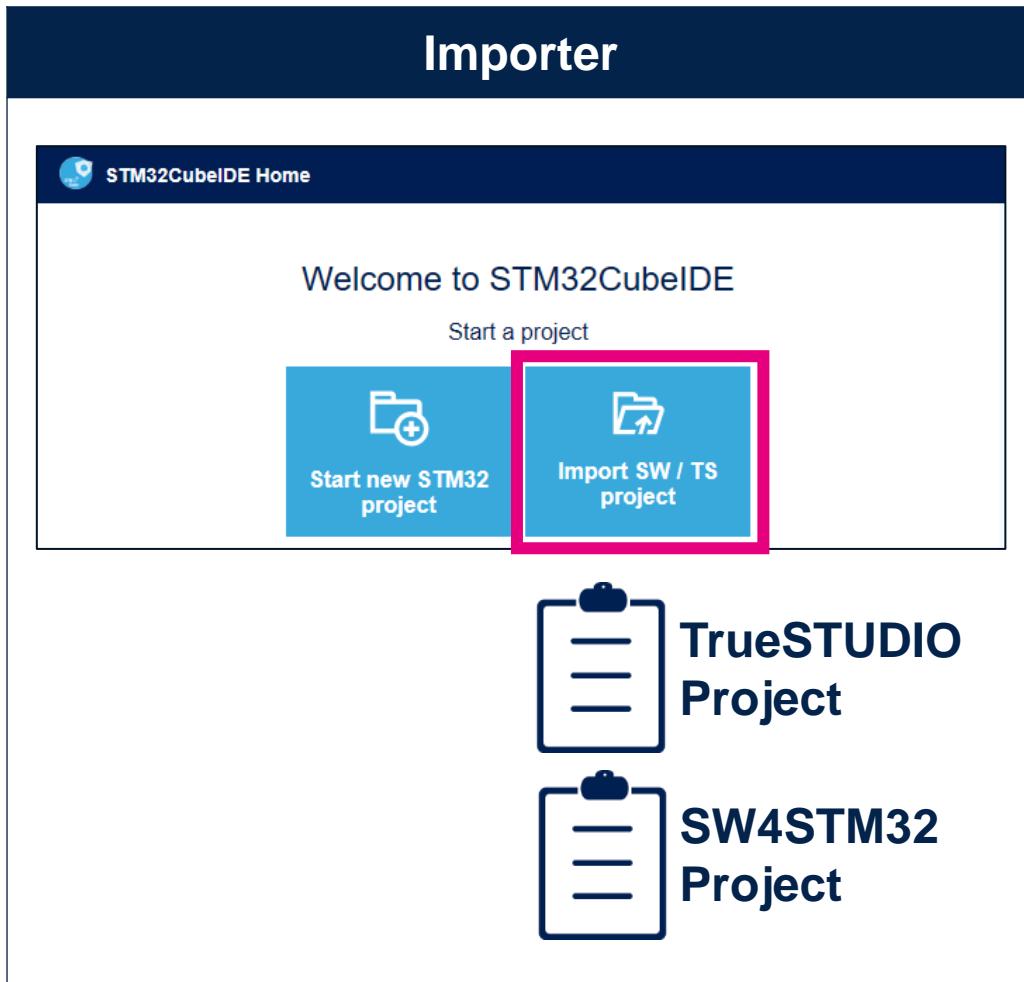
Eclipse/GCC Based

Free for Commercial Development

macOS



Project management





Code editor – navigation

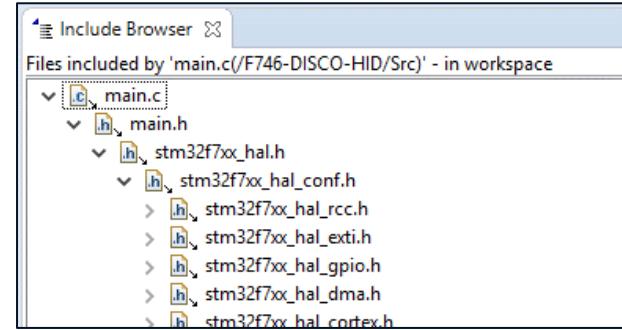
Symbol Hyperlink

```
BSP_LED_Init(LED1);
BSP_LED_Init(LED2);
BSP_LED_Init(LED3);
BSP_LED_Init(LED4);

void BSP_LED_Init(Led_TypeDef Led)
{
    GPIO_InitTypeDef gpio_init_structure;

    if (Led <= LED4)
    {
        /* Configure the GPIO_LED pin */
        gpio_init_structure.Pin = GPIO_PIN[Led];
        gpio_init_structure.Mode = GPIO_MODE_OUTPUT_PP;
        gpio_init_structure.Pull = GPIO_PULLUP;
        gpio_init_structure.Speed = GPIO_SPEED_HIGH;
    }
}
```

Include Browser



Call Hierarchy

```
HAL_StatusTypeDef USB_WritePacket(USB_OTG_GlobalTypeDef *pUSBx, uint8_t *src, uint8_t
uint32_t USbx_BASE = (uint32_t)USbx;
uint32_t *pSrc = (uint32_t *)src;
uint32_t count32b, 1;
```

Call Hierarchy X
Callers of USB_WritePacket(USB_OTG_GlobalTypeDef *, uint8_t, uint16_t, uint8_t) - /F746-DISCO-TEST/Drivers/S

- USB_WritePacket(USB_OTG_GlobalTypeDef *, uint8_t, uint16_t, uint8_t) : HAL_StatusTypeDef
- HAL_PCD_IROHandler(PCD_HandleTypeDef *pCD, HandleTypeDef *, void)
- OTG_FS_IRQHandler() : void
- HAL_PCD_EP_Receive(PCD_HandleTypeDef *pCD, HandleTypeDef *, uint8_t, uint8_t, uint32_t) : HAL_StatusTypeDef
- HAL_PCD_EP_Transmit(PCD_HandleTypeDef *pCD, HandleTypeDef *, uint8_t, uint8_t, uint32_t) : HAL_StatusTypeDef
- USBD_LL_Transmit(USBD_HandleTypeDef *pUSBx, uint8_t, uint8_t, uint16_t) : USBD_StatusTypeDef
- USB_HC_StartXfer(USB_OTG_GlobalTypeDef *, USB_OTG_HCTypeDef *, uint8_t) : HAL_StatusTypeDef

Brace Navigation

```
294     /* Check the parameters */
295     assert_param(IS_OPTIONBYTE(pOBInit->OptionType));
296
297     /* Write protection configuration */
298     if((pOBInit->OptionType & OPTIONBYTE_WRP) == OPTIONBYTE_WRP)
299     {
300         assert_param(IS_WRPSTATE(pOBInit->WRPState));
301         if(pOBInit->WRPState == OB_WRPSTATE_ENABLE)
302         {
303             /*Enable of Write protection on the selected Sector*/
304             status = FLASH_OB_EnableWRP(pOBInit->WRPSector);
305         }
306         else
307         {
308             /*Disable of Write protection on the selected Sector*/
309             status = FLASH_OB_DisableWRP(pOBInit->WRPSector);
310         }
311     }
312 }
```

Macro Expansion Browser

Explore Macro Expansion - 2 step(s)
#define USB_OTG_FS ((USB_OTG_GlobalTypeDef *) USB_OTG_FS_PERIPH_BASE)

Original	Fully Expanded
USB_OTG_FS	1 ((USB_OTG_GlobalTypeDef *) 0x50000000UL)

Explore Macro Expansion - 2 step(s)
#define USB_OTG_FS ((USB_OTG_GlobalTypeDef *) USB_OTG_FS_PERIPH_BASE)

Original	Expansion #1 of 2
1 USB_OTG_FS	1 ((USB_OTG_GlobalTypeDef *) USB_OTG_FS_PERIPH_BASE)

Explore Macro Expansion - 2 step(s)
#define USB_OTG_FS_PERIPH_BASE 0x50000000UL

Expansion #1 of 2	Fully Expanded
1 USB_OTG_FS	1 ((USB_OTG_GlobalTypeDef *) 0x50000000UL)

Type Hierarchy

```
46 _UART_HandleTypeDef huart2;
47
48 /* USER CODE BEGIN PV */
49
50 % Type Hierarchy X
51
52 _UART_HandleTypeDef
```

_UART_HandleTypeDef

- AdvancedInit: _UART_AdvFeatureInitTypeDef - _UART_HandleTypeDef
- ErrorCode: volatile uint32_t - _UART_HandleTypeDef
- gState: volatile HAL_UART_StateTypeDef - _UART_HandleTypeDef
- hdmax: DMA_HandleTypeDef * - _UART_HandleTypeDef
- hdmin: DMA_HandleTypeDef * - _UART_HandleTypeDef
- init: _UART_InitTypeDef - _UART_HandleTypeDef
- Instance: USART_HandleTypeDef - _UART_HandleTypeDef
- Lock: HAL_LockTypeDef - _UART_HandleTypeDef
- Mask: uint16_t - _UART_HandleTypeDef
- pRxBuffPtr: uint8_t * - _UART_HandleTypeDef
- pTxBuffPtr: uint8_t * - _UART_HandleTypeDef

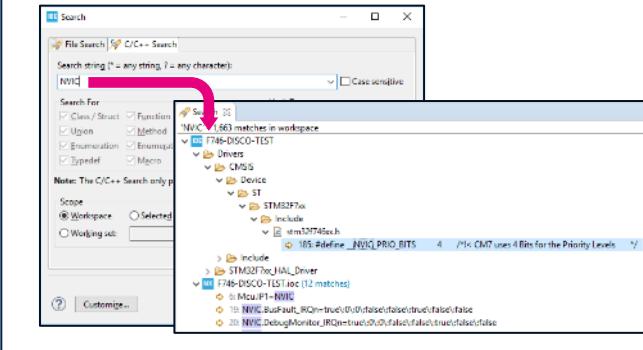


Code editor – navigation (2/2)

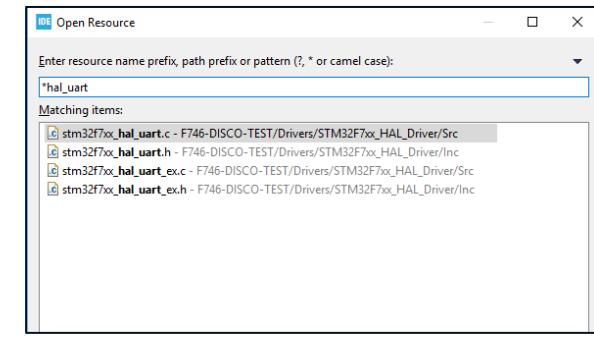
Outline View



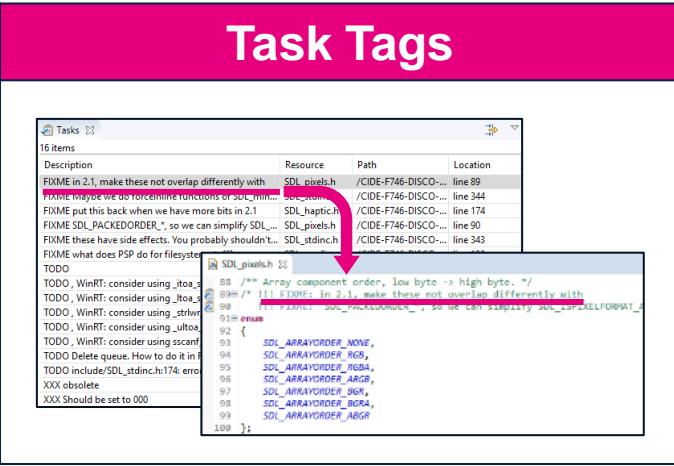
Powerful Search



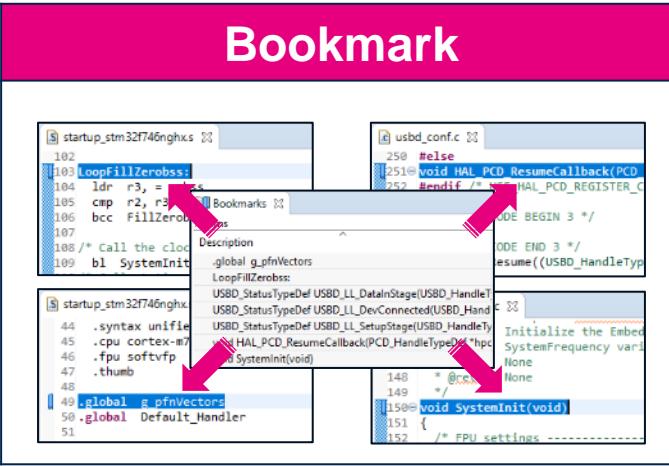
Open Resource



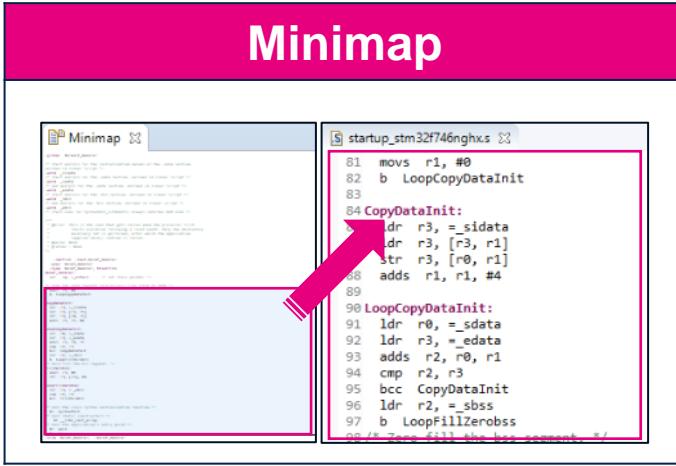
Task Tags



Bookmark



Minimap





Code editor – writing

Highlight Inactive Code

```

157
158 #if defined ( __ICCARM__ ) /* IAR Compiler */
159 #pragma data_alignment=4
160 #endif /* defined ( __ICCARM__ ) */
161 /* USB standard device descriptor. */
162 __ALIGN_BEGIN uint8_t USBD_Fs_DeviceDesc[USB_LEN_DEV_DESC] __ALIGN_END =
163 {
164     .bLength = 0x12,
165     .bDescriptorType = 0x0B,
166     #if (USBD_LPM_ENABLED == 1)
167         .bcdUSB = 0x01, /* bcdUSB */ // changed to USB version 2.01
168         .bDeviceClass = 0x02, /* bDescriptorType */
169         .bDeviceSubClass = 0x00, /* bDeviceProtocol */
170         .bMaxPacketSize0 = 0x40, /* bMaxPacketSize */
171         .bDeviceID = 0x00, /* idVendor */
172         .bDeviceVID = 0x00, /* idProduct */
173         .bcdDevice = 0x00, /* bcdDevice */
174         .wTotalLength = 0x00, /* wTotalLength */
175         .wDeviceChange = 0x00, /* wDeviceChange */
176         .wProduct = 0x00, /* wProduct */
177         .wSerialNumber = 0x00, /* wSerialNumber */
178         .bNumConfigurations = 1, /* bNumConfigurations */
179         .bConfigurationValue = 0x01, /* bConfigurationValue */
180         .bMaxPower = 0x03, /* bMaxPower */
181     #else
182         .bcdUSB = 0x00, /* bcdUSB */
183         .bDeviceClass = 0x00, /* bDescriptorType */
184         .bDeviceSubClass = 0x00, /* bDeviceProtocol */
185         .bMaxPacketSize0 = 0x08, /* bMaxPacketSize */
186         .bDeviceID = 0x00, /* idVendor */
187         .bDeviceVID = 0x00, /* idProduct */
188         .bcdDevice = 0x00, /* bcdDevice */
189         .wTotalLength = 0x00, /* wTotalLength */
190         .wDeviceChange = 0x00, /* wDeviceChange */
191         .wProduct = 0x00, /* wProduct */
192         .wSerialNumber = 0x00, /* wSerialNumber */
193         .bNumConfigurations = 1, /* bNumConfigurations */
194         .bConfigurationValue = 0x01, /* bConfigurationValue */
195         .bMaxPower = 0x03, /* bMaxPower */
196     #endif
197 }

```

Auto-Complete

```

97 /* USER CODE BEGIN 2 */
98 HAL_GPIO
99 /* USER CO
100     ● HAL_GPIO_DelInit(GPIO_TypeDef* GPIOx, uint32_t GPIO_Pin) : void
101     ● HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin) : void
102     ● HAL_GPIO_EXTI_IRQHandler(uint16_t GPIO_Pin) : void
103     ● HAL_GPIO_Init(GPIO_TypeDef* GPIOx, GPIO_InitTypeDef * GPIO_InitStruct) : void
104     ● HAL_GPIO_LockPin(GPIO_TypeDef* GPIOx, uint16_t GPIO_Pin) : HAL_StatusTypeDef
105     ● HAL_GPIO_ReadPin(GPIO_TypeDef* GPIOx, uint16_t GPIO_Pin) : GPIO_PinState
106     ● HAL_GPIO_TogglePin(GPIO_TypeDef* GPIOx, uint16_t GPIO_Pin) : void
107     ● HAL_GPIO_WritePin(GPIO_TypeDef* GPIOx, uint16_t GPIO_Pin, GPIO_PinState)
108     ● HAL_GPIO_MODULE_ENABLED
109 */
110
111 /**
112  * @brief S
113  * @retval
114

```

Press 'Ctrl+Space' to show Template Proposals

Syntax Highlight

```

116 void SystemClock_Config(void)
117 {
118     RCC_OscInitTypeDef RCC_OscInitStruct = {0};
119     RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};
120     RCC_PeriphCLKInitTypeDef PeriphClkInitStruct = {0};
121
122     /* Configure the main internal regulator output voltage
123     */
124     __HAL_RCC_PWR_CLK_ENABLE();
125     __HAL_RCC_VOLTAGESCALING_CONFIG(PWR_REGULATOR_VOLTAGE_SCALE3);
126
127     /**
128      * @brief Oscillator Configuration
129      */
130     RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSI|RCC_OSCILLATOR_TYPE_NONE;
131     RCC_OscInitStruct.HSEState = RCC_HSE_ON;
132     RCC_OscInitStruct.HSICalibrationValue = RCC_HSICALIBRATION_DEFAULT;
133     RCC_OscInitStruct.PLL.PLLState = RCC_PLL_ON;
134     RCC_OscInitStruct.PLL.PLLSource = RCC_PLLSOURCE_HSE;
135     RCC_OscInitStruct.PLL.PLLM = 15;
136     RCC_OscInitStruct.PLL.PLLN = 144;

```

File Diff/Compare

The screenshot shows a 'Compare' window with two tabs: 'C Compare' and 'C Compare Viewer'. The 'C Compare' tab lists differences between 'F746-DISCO-TEST/Src/main.c' and 'F746-DISCO-TEST-2/Src/main.c'. The 'C Compare Viewer' tab shows the actual code differences side-by-side.

Block Select

The screenshot shows a code editor with a red box highlighting a block of code. The code block is enclosed in a dashed rectangle and contains several lines of C code, including GPIO operations and a loop.

Code Style

The screenshot shows the 'Formatter' settings in the 'Preferences' dialog. The 'Active profile' dropdown is set to 'K&R [built-in]'. A preview window shows a sample C code snippet with specific styling applied.



Build tools

Build Analyzer

Build Analyzer Static Stack Analyzer Search

F769-DISCO-ITCM.elf - /F769-DISCO-ITCM/Debug - May 10, 2019 3:28:29 PM

Memory Regions Memory Details

Region	Start address	End address	Size	Free	Used	Usage (%)
FLASH	0x08000000	0x08200000	2048 KB	2043.22 KB	4.78 KB	0.23%
RAM	0x20000000	0x20080000	512 KB	510.45 KB	1.55 KB	0.30%
ITCMRAM	0x00000000	0x00004000	16 KB	15.48 KB	528 B	3.22%

Build Analyzer Static Stack Analyzer Search

F769-DISCO-ITCM.elf - /F769-DISCO-ITCM/Debug - May 10, 2019 3:28:29 PM

Memory Regions Memory Details

Search

Name	Run address (VMA)	Load address (LMA)	Size
ITCMRAM	0x00000000		16 KB
> .itcmram	0x00000000	0x0800110c	528 B
ITCMRAM	0x00000000		2048 KB
> .itcmram	0x00000000	0x0800110c	528 B
FLASH	0x08000000		2048 KB
> .itcmram	0x08000000	0x0800110c	528 B
> .isr_vector	0x08000000	0x08000000	60 B
> .text	0x0800003c	0x0800003c	4.16 KB
> .rodata	0x080010e0	0x080010e0	16 B
> .ARM	0x080010f0	0x080010f0	8 B
> .preinit_array	0x080010f8	0x080010f8	0 B
> .init_array	0x080010f8	0x080010f8	4 B
> .fini_array	0x080010fc	0x080010fc	4 B
> .data	0x20000000	0x08001100	12 B
RAM	0x20000000		512 KB
> .data	0x20000000	0x08001100	12 B
> .bss	0x2000000c		32 B
> _user_heap_stack	0x2000002c		1.5 KB

Static Stack Analyzer

Build Analyzer Static Stack Analyzer Search

F769-DISCO-ITCM.elf - /F769-DISCO-ITCM/Debug - May 10, 2019 3:28:29 PM

List Call graph

Hide dead code

Function	L...	Type	Location	Info
SystemClock_Config	88	STATIC	main.c:108	
NVIC_EncodePriority	40	STATIC	core_cm7.h:2071	
HAL_RCC_GetSysClockFreq	40	STATIC	stm32fxx_hal_rcc.c:982	
HAL_NVIC_SetPriority	32	STATIC	stm32fxx_hal_cortex.c:165	
HAL_NVIC_OscConfig	32	STATIC	stm32fxx_hal_rcc.c:344	
_NVIC_SetPriorityGrouping	24	STATIC	core_cm7.h:1865	
HAL_RCC_ClockConfig	24	STATIC	stm32fxx_hal_rcc.c:703	
HAL_InitTick	16	STATIC	stm32fxx_hal.c:231	

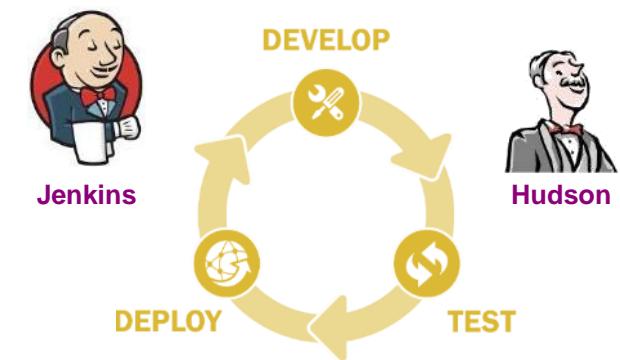
Build Analyzer Static Stack Analyzer Search

F769-DISCO-ITCM.elf - /F769-DISCO-ITCM/Debug - May 10, 2019 3:28:29 PM

List Call graph

Function	Depth	Max cost	Local cost	Type	Location
> LoopFillZeroBoss	7	208	0		stm32fxx_it.c:182
> SysTick_Handler	0	8	8	STATIC	stm32fxx_it.c:70
> NMI_Handler	0	4	4	STATIC	stm32fxx_it.c:128
> UsageFault_Handler	0	4	4	STATIC	stm32fxx_it.c:160
> PendSV_Handler	0	4	4	STATIC	stm32fxx_it.c:169
> HardFault_Handler	0	4	4	STATIC	stm32fxx_it.c:83
> HAL_IncTick	0	4	4	STATIC	stm32fxx_hal.c:290
> SVC_Handler	0	4	4	STATIC	stm32fxx_it.c:143
> DebugMon_Handler	0	4	4	STATIC	stm32fxx_it.c:156
> MemManage_Handler	0	4	4	STATIC	stm32fxx_it.c:98
> BusFault_Handler	0	4	4	STATIC	stm32fxx_it.c:113
> Reset_Handler	0	0	0		
> init	0	0	0		

Headless Build



Continuous Integration

- Build project without opening IDE
- No GUI shown but build system becomes active
- Supported for makefile and managed projects



Debug

Debugger

The screenshot shows the 'Debugger' tab of the STM32CubeIDE interface. It includes fields for 'Debug probe' (SEGGER J-LINK selected), 'GDB Connection' (ST-LINK (ST-LINK GDB server) selected), 'Autostart' (radio button selected), and 'Connect to remote GDB server' (unchecked). A 'GDB Server Command Line Options' section is also present.

Live expressions

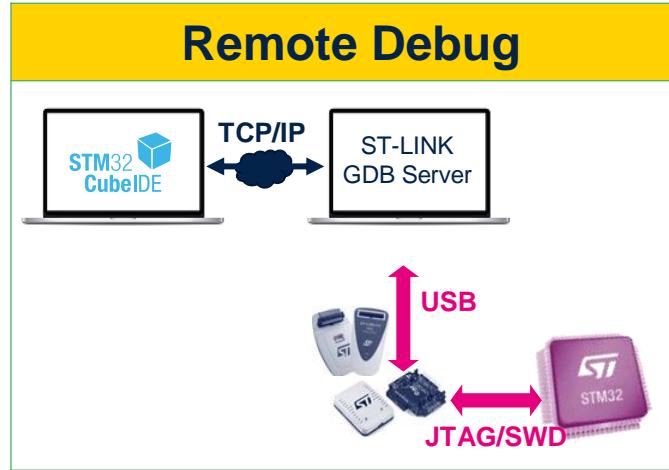
The screenshot shows the 'Live Expressions' window. It displays a table with one row: Expression (x)= uwTick, Type volatile uint32_t, Value 1603. There is also a button to 'Add new expression'.

SWV

The screenshot shows two windows related to SWV. The left window is 'SWV Statistical Profiling' showing a table of functions and their usage statistics. The right window is 'SWV Data Trace Timeline Graph' showing a graph of three data series (accel[xAxis], accel[yAxis], accel[zAxis]) over time.

Integrated UART Terminal

The screenshot shows the 'Integrated UART Terminal' window. It displays a 'Select Remote Connection' dialog with 'Connection Type' set to 'Serial Port' and 'Connection name' set to 'F746-DISCO-VirtualCom (CONNECTED)'. The terminal console shows the output of the command 'LD1:'. An 'OK' button is visible at the bottom.



RTOS aware Debug

The screenshot shows the 'RTOS aware Debug' interface. It features a navigation tree on the left with sections for 'FreeRTOS' (Device Configuration Tool, FreeRTOS Queues, FreeRTOS Semaphores, FreeRTOS Task List, FreeRTOS Timers) and 'ThreadX' (ThreadX Event Flags, ThreadX Memory Block Pools, ThreadX Memory Byte Pools). The 'FreeRTOS' section is highlighted with a pink rectangle.

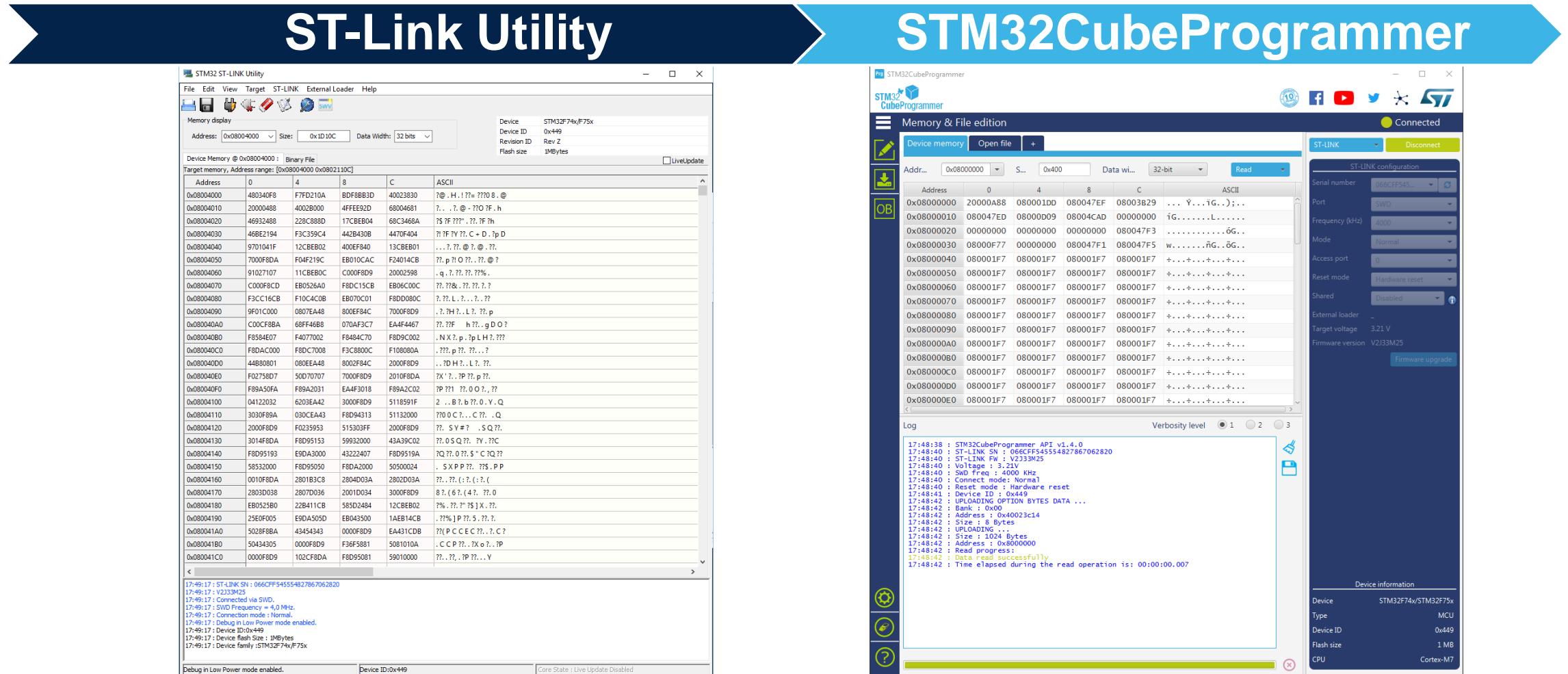
FreeRTOS

ThreadX Microsoft Azure

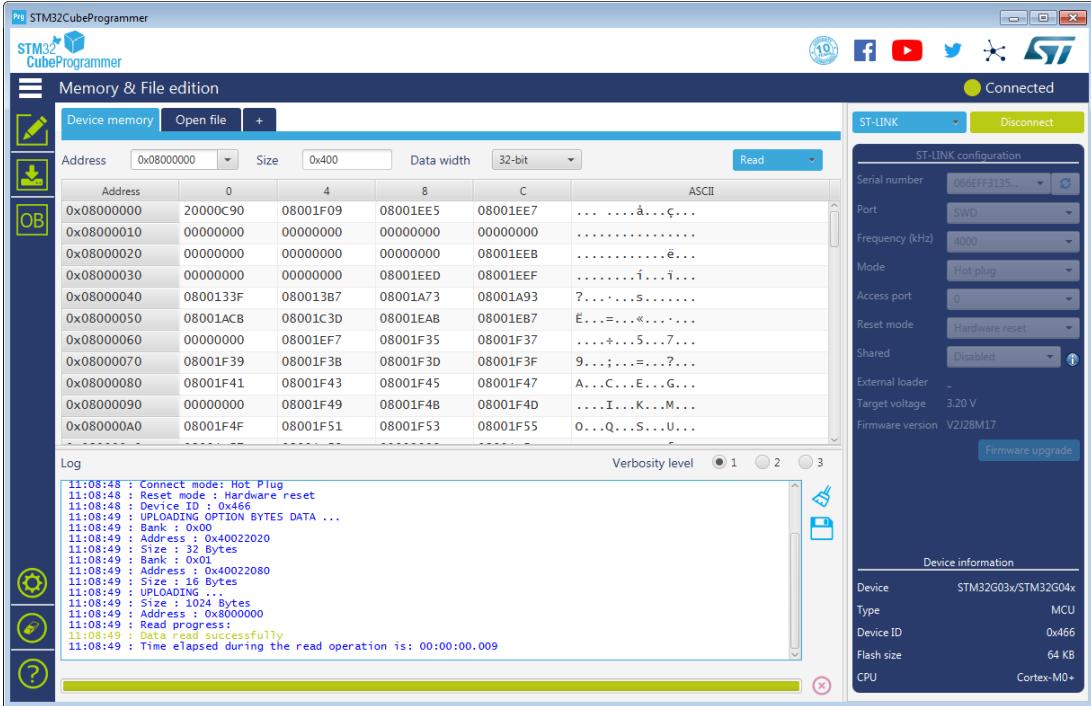
STM32CubeProgrammer

STM32
CubeProgrammer

From ST-LINK Utility to STM32CubeProgrammer



All-in-one programming software tool



Intuitive GUI

Multi-platform
(Windows, Linux, macOS)

STLink Direct Support
(JTAG, SWD)

Automatic Mode

Option Bytes
Program & Upload

Bootloader Interface Support
(USB, UART, SPI, I2C, CAN)

Internal/External
Flash Services

API DLL
for Custom Integration

Command Line Interface
for Scripting

Trusted Package Creator
(secure programming)

From ST-Link V2 to STLink-V3

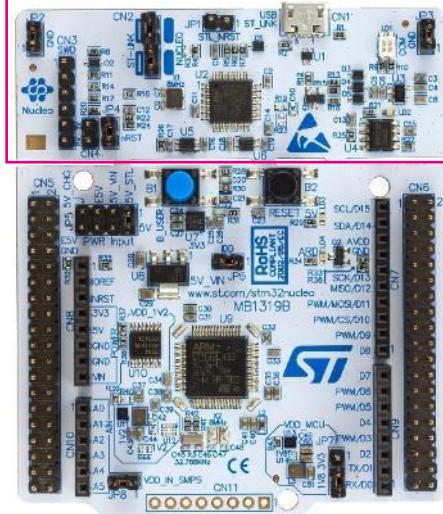
ST-Link V2



ST-LINK/V2

ST-LINK/V2-ISOL

ST-LINK/V2-1



STLink-V3

STLINK-V3MINI

STLINK-V3MODS

STLINK-V3SET



STLink-V3 debugger / programmer

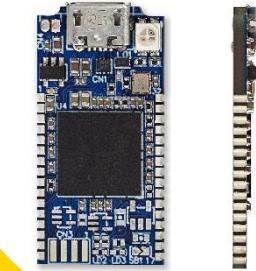
Easier/Faster/Affordable

Stand-alone and scalable (V3SET)

Multi-path Bridge (I²C/SPI/CAN/UART/USB)

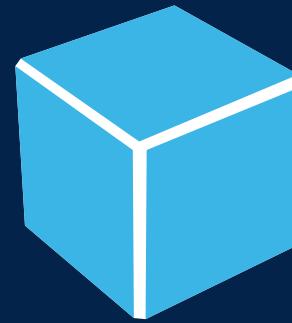
Drag&Drop Flash Programming

Compact (V3MINI) or On-board (V3MODS)

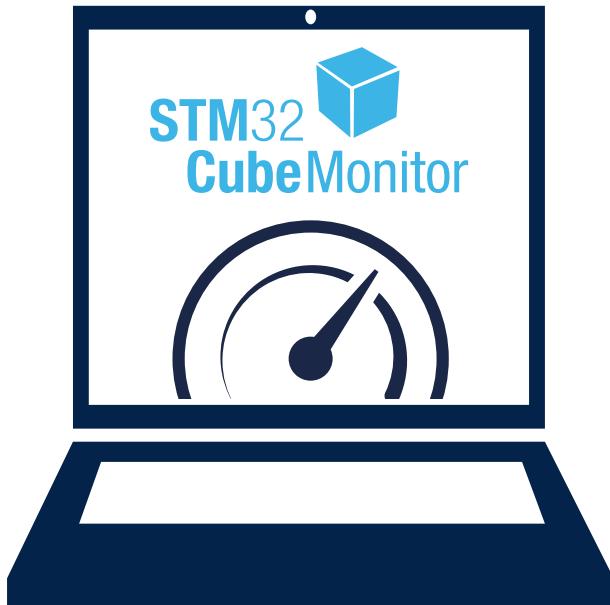


STM32CubeMonitor

STM32
CubeMonitor



STM32CubeMonitor overview



Monitoring application variables during runtime

- Non-intrusive tool to follow application behavior without interruption.
- Real-time analysis to finetune application configuration.

Drag & drop creation of dashboard UI

- Large choice of graphical components (gauges, bar graphs, plots...)
- Customize settings. No need for programming.
- Direct support of the Node-RED® open community.

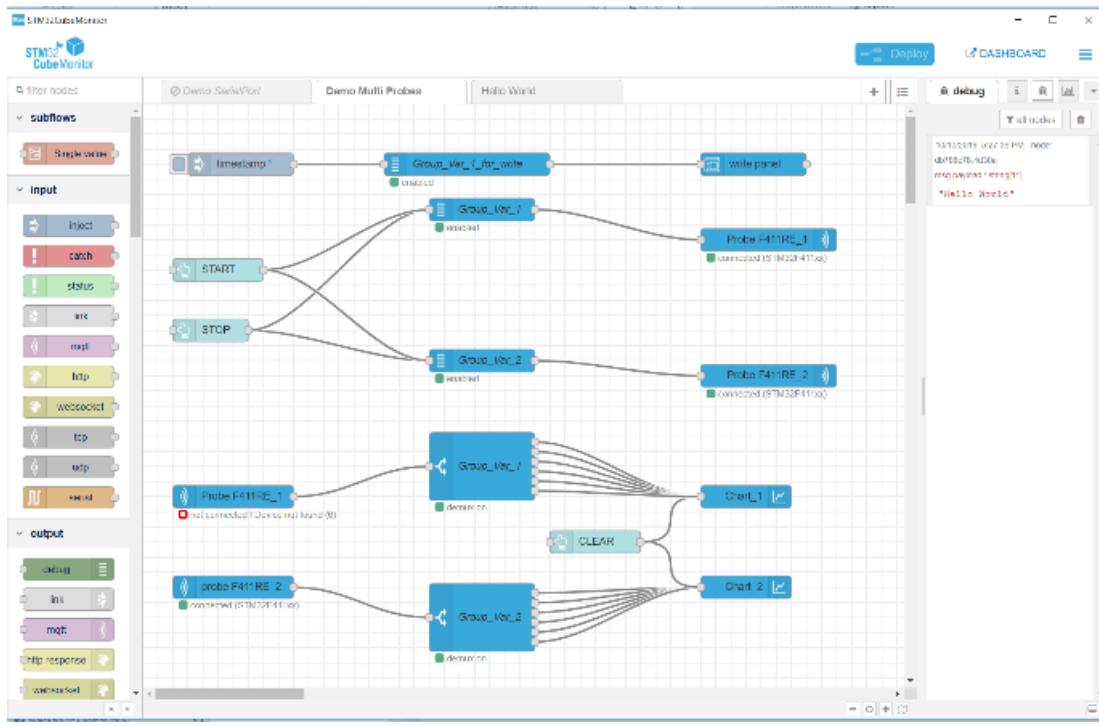
Graphical visualization on any display

- Multi-OS tool: direct support of PC, tablets and smartphones.
- Remote monitoring.

Graphical custom data visualization

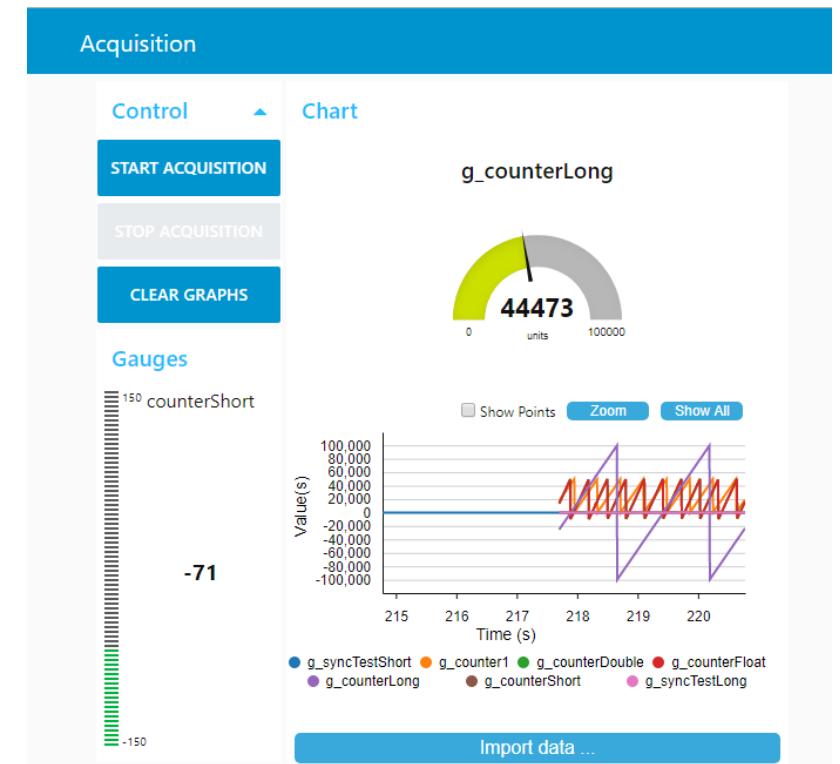
Design mode to create

Build and edit the logical data flow and graphical rendering of the custom monitoring UI.



Dashboard mode to visualize

Use the monitoring UI built previously and visualize locally or remotely.



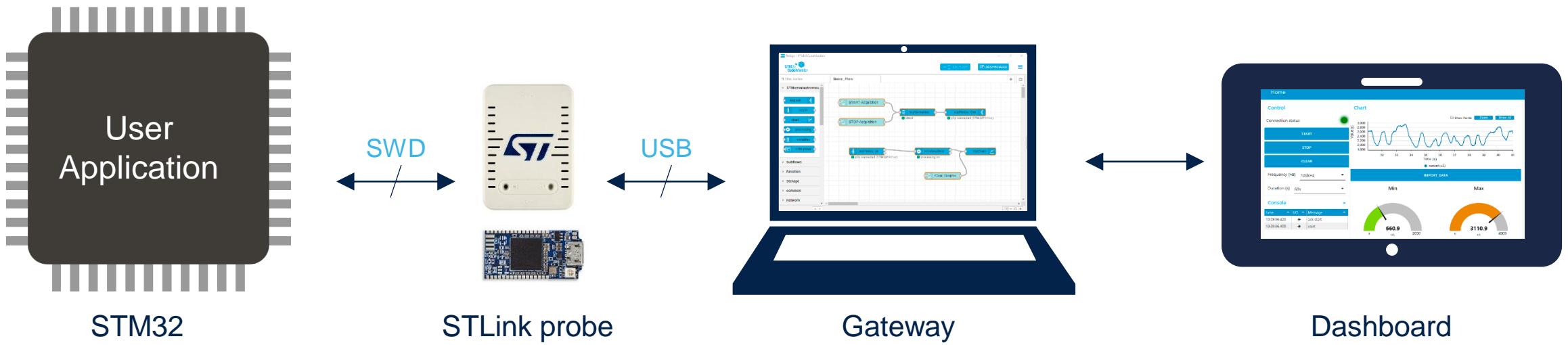
Remote monitoring

Native support of multi-format displays

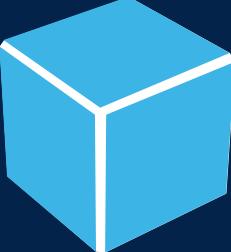
Dynamic layout of dashboard UI on PCs, tablets, smartphones.

Remote data acquisition with web server technology

Monitor across a network with a web browser

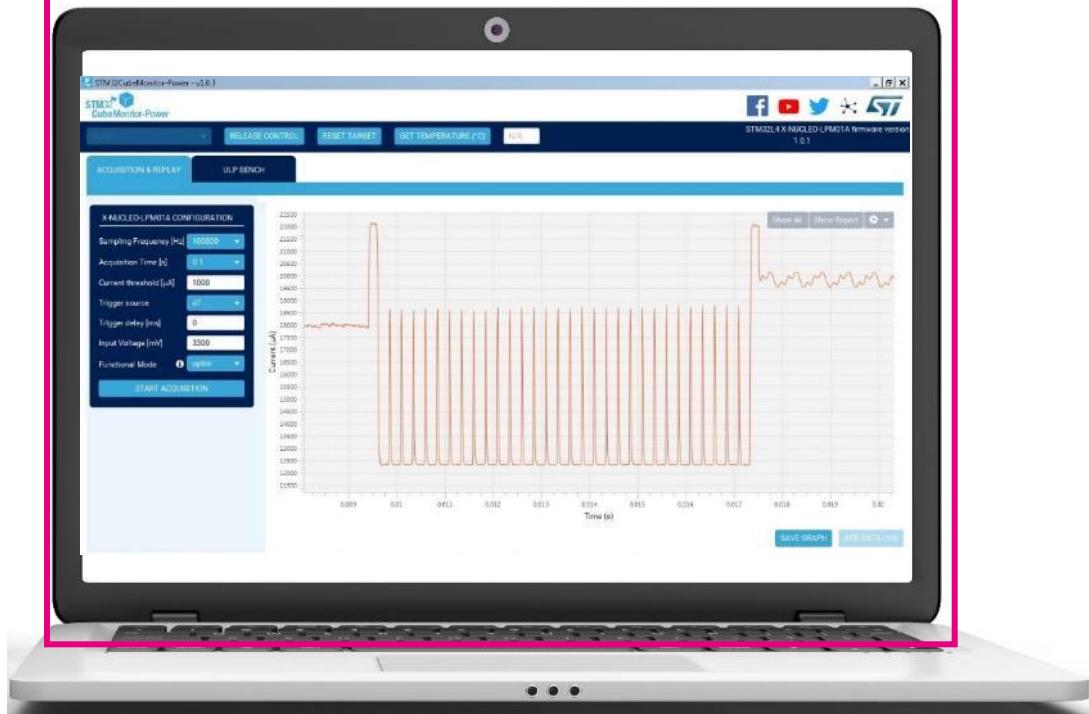


STM32CubeMonitor-Power

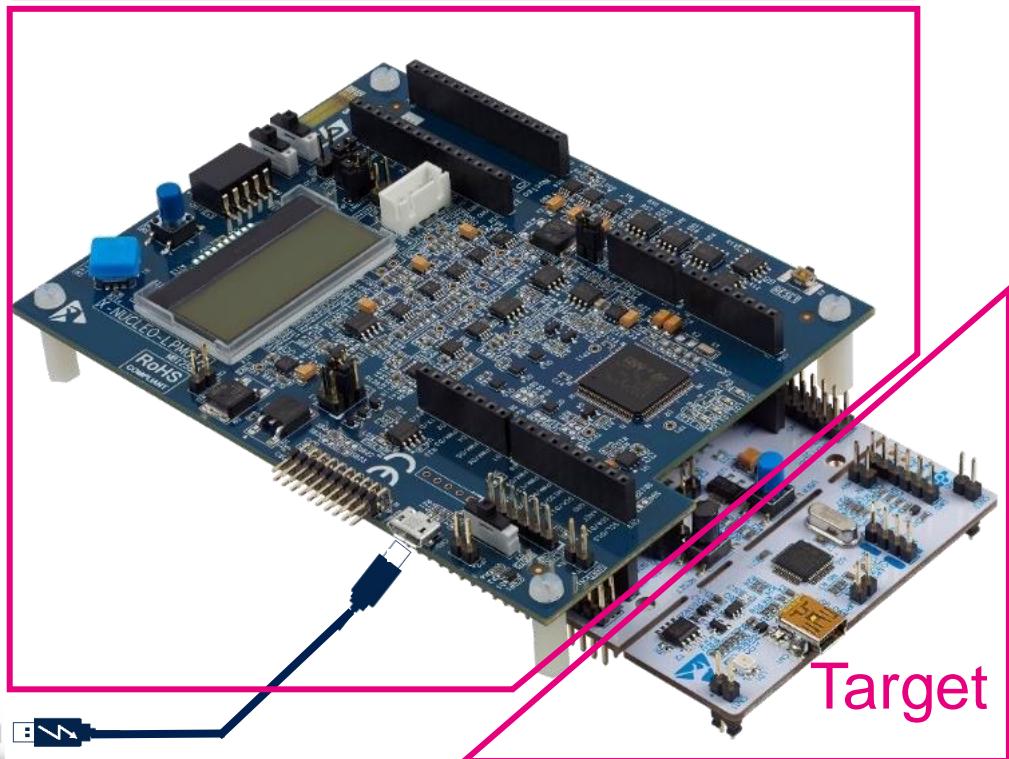
STM32  **CubeMonitor-Power**

STM32CubeMonitor-Power

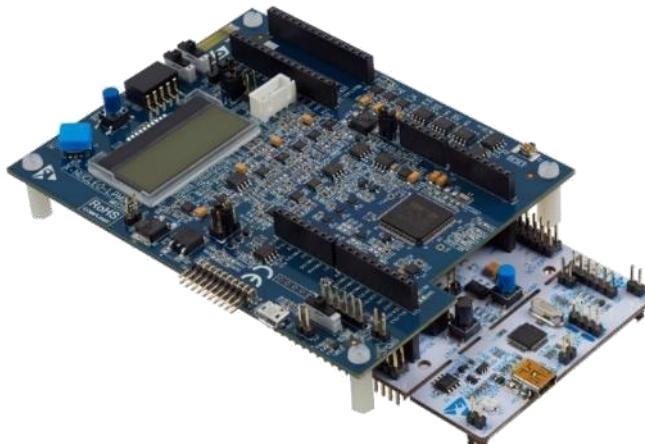
STM32CubeMonPwr



X-NUCLEO-LPM01A



STM32 power profiling



Ultra-Low-Power Consumption Measurements

- Supply target board from **1.8V to 3.3V**
- Dynamic current from **100 nA to 50 mA (100 dB)**
- Static current from **1 nA to 200 mA**
- Accuracy approximately **2%**

Intuitive User Experience

- Two operating modes (**stand-alone or PC-controlled**)
- Graphical PC application (reference: **STM32CubeMonPwr**)

Official EEMBC Energy Monitor v2.0

Direct computation of ULPMark scores

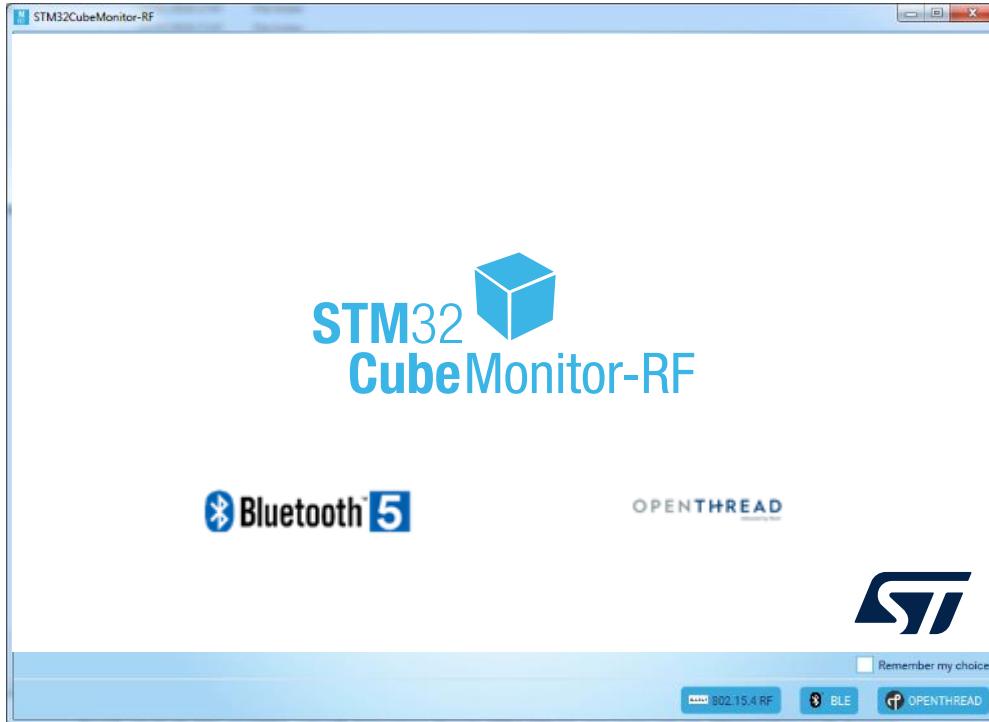


STM32CubeMonitor-RF

STM32
CubeMonitor-RF

STM32CubeMonitor-RF

A software tool allowing to test the radio performances of STM32WB MCUs for BLE and 802.15.4 technologies



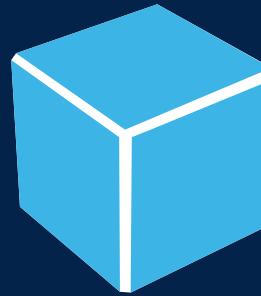
Test protocol sequences

Configure static / dynamic beacons

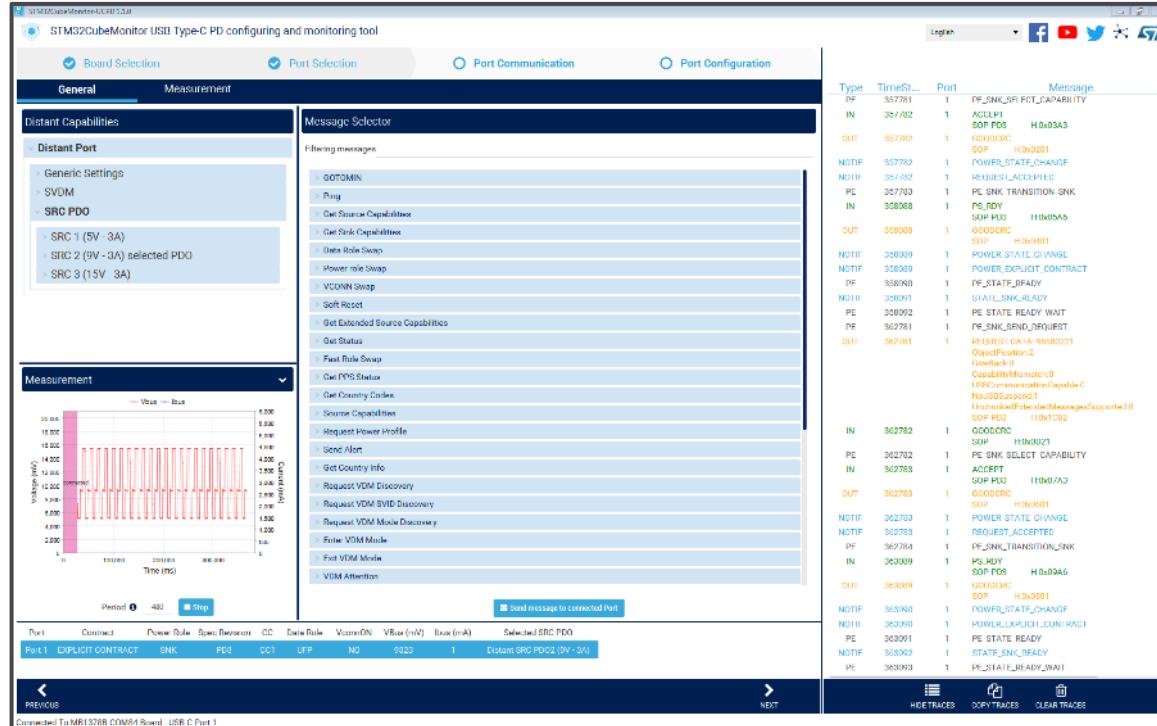
Manage Over the Air (OTA) file transfer

STM32CubeMonitor-UCPD

STM32
CubeMonitor-UCPD



Monitoring and configuring tool for **USB Type-C™** and **USB Power Delivery** applications using STM32 microcontroller



Support of **USB Type-C™ 1.2** and **USB PD 2.0/3.0**

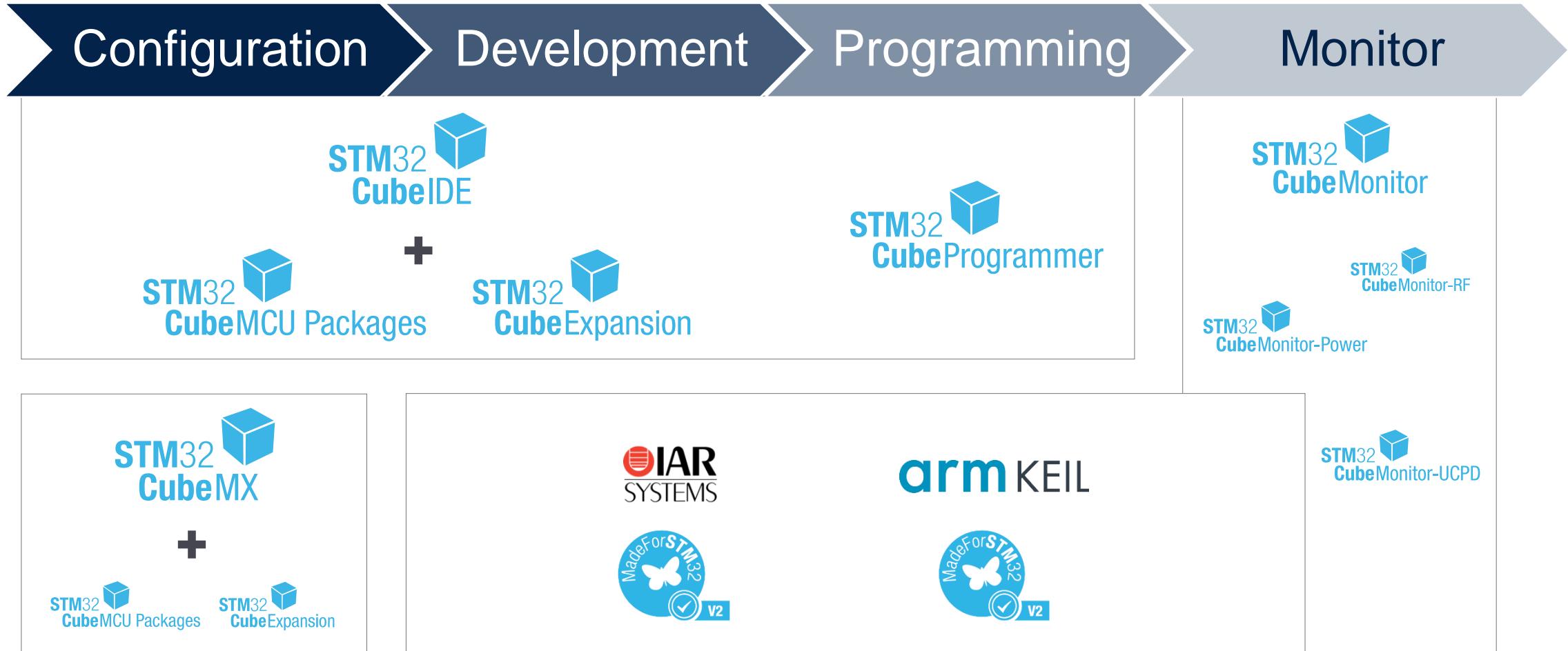
Port configuration pane for **PD setting, VDM, SOP, Source and Sink Capabilities**

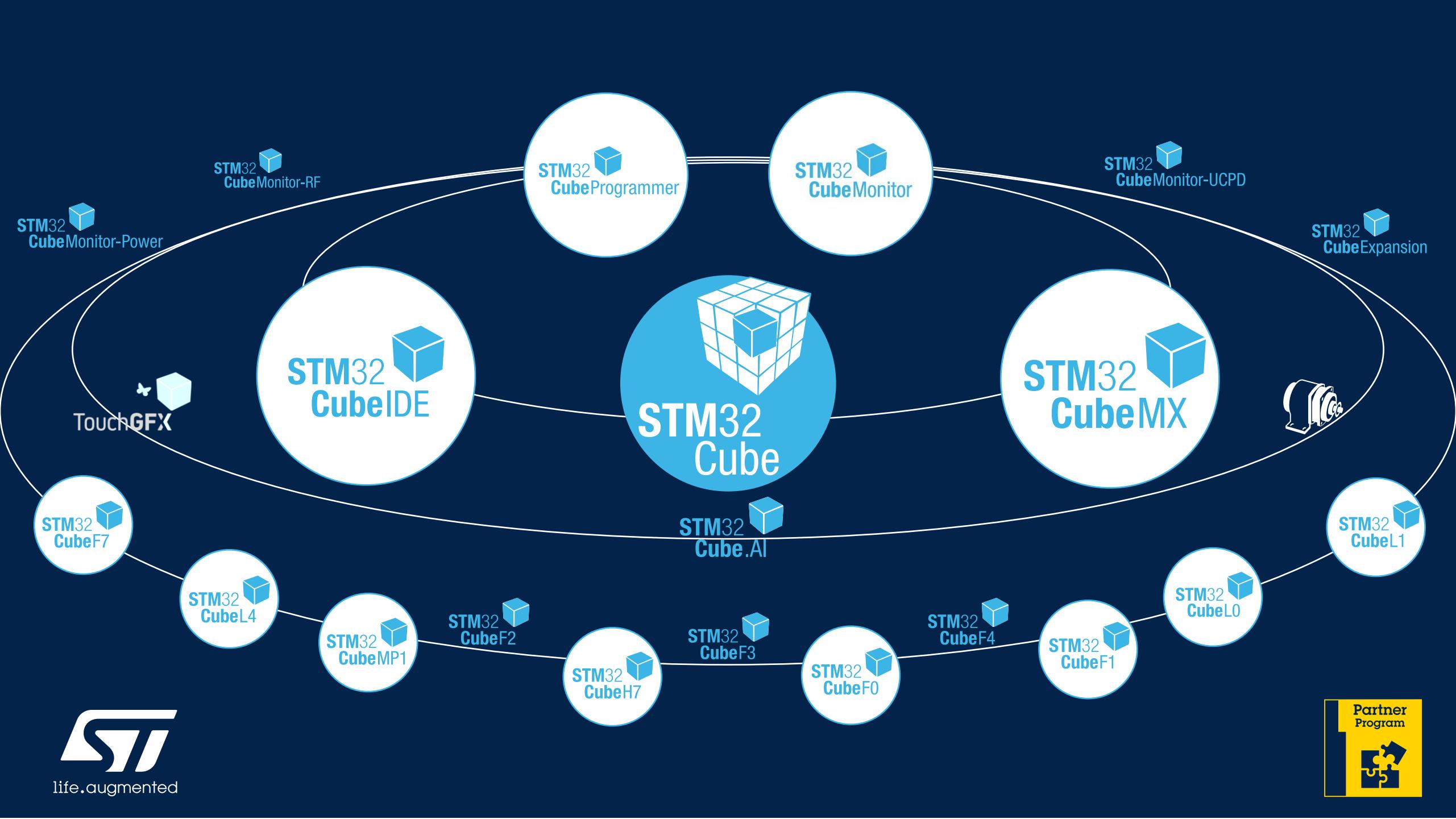
Port communication pane for **VBUS and IBUS monitoring, distant port capabilities, message selector, and real-time traces**

Key takeaways



Iterative development process





Thank you

© STMicroelectronics - All rights reserved.

ST logo is a trademark or a registered trademark of STMicroelectronics International NV or its affiliates in the EU and/or other countries.

For additional information about ST trademarks, please refer to www.st.com/trademarks.

All other product or service names are the property of their respective owners.

