



life.augmented

Showcasing prototype with BLE

Krishan SOLANKI

Agenda

1 Day1 Demo Recap

2 BLE device Types

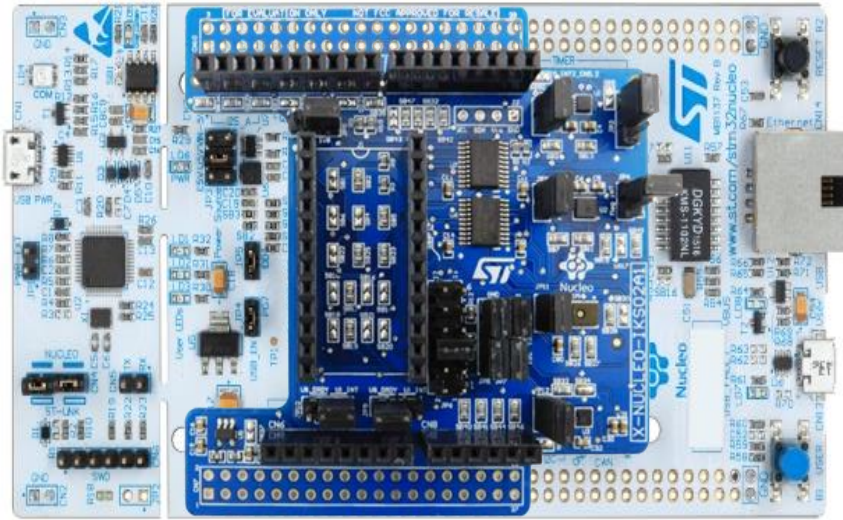
3 BLE Modules

4 Hardware Setup

5 Live Demo 1: Connection
Sensor board Over BLE

Recap – Day 1 Live Demo

STM32L4 + Sensor IKS02A1



Onboard

IIS2MDC

3D magnetometer

ISM330DHCX

3D accelerometer
plus 3D gyroscope

IIS2DLPC

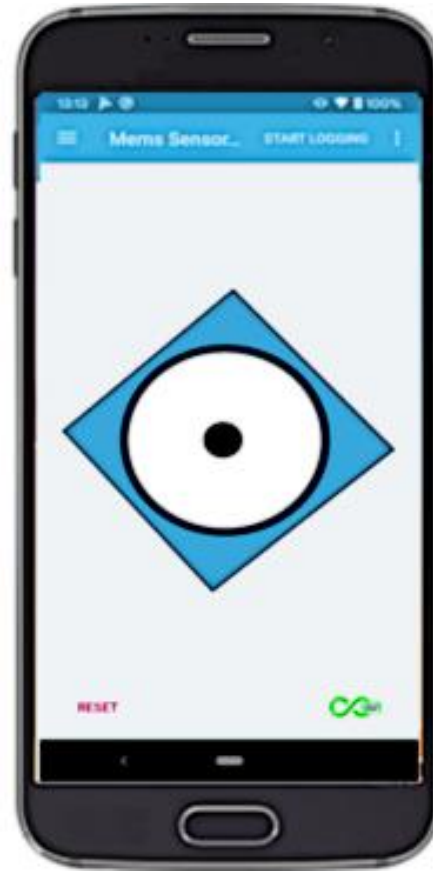
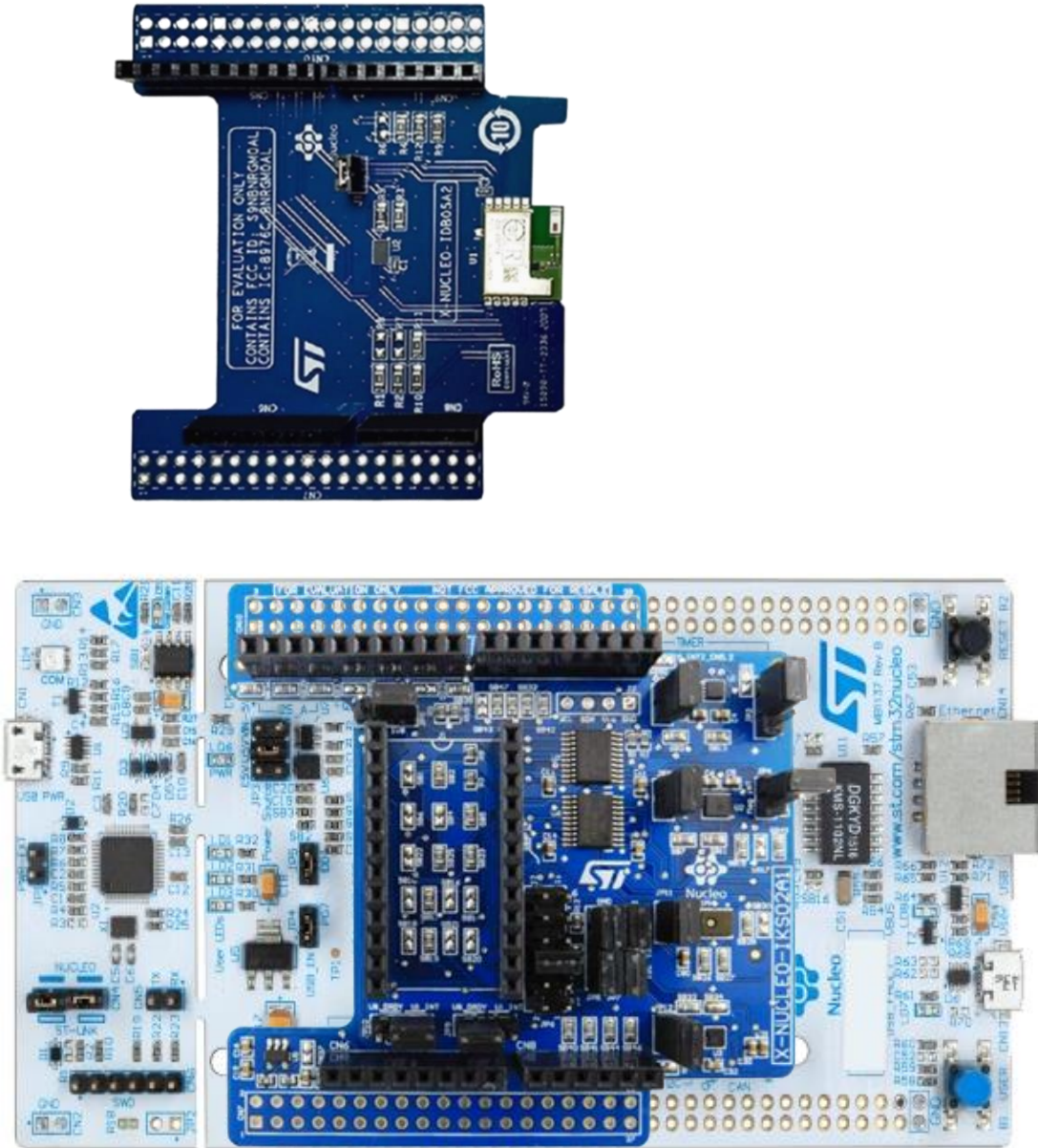
3D accelerometer
– Low Power

IMP34DT05

Digital
omnidirectional
microphone

Expressions 1016 0101 Registers Live Expressions X SFRs		
Expression	Type	Value
acceleration	IKS02A1_MOTION_SENSOR_Axes_t	{...}
(x)= x	int32_t	66
(x)= y	int32_t	-30
(x)= z	int32_t	1031
angular_velocity	IKS02A1_MOTION_SENSOR_Axes_t	{...}
(x)= x	int32_t	140
(x)= y	int32_t	-280
(x)= z	int32_t	-490
magnetic_field	IKS02A1_MOTION_SENSOR_Axes_t	{...}
(x)= x	int32_t	-582
(x)= y	int32_t	-343
(x)= z	int32_t	-369
+ Add new expression		

What Next- Add Connectivity



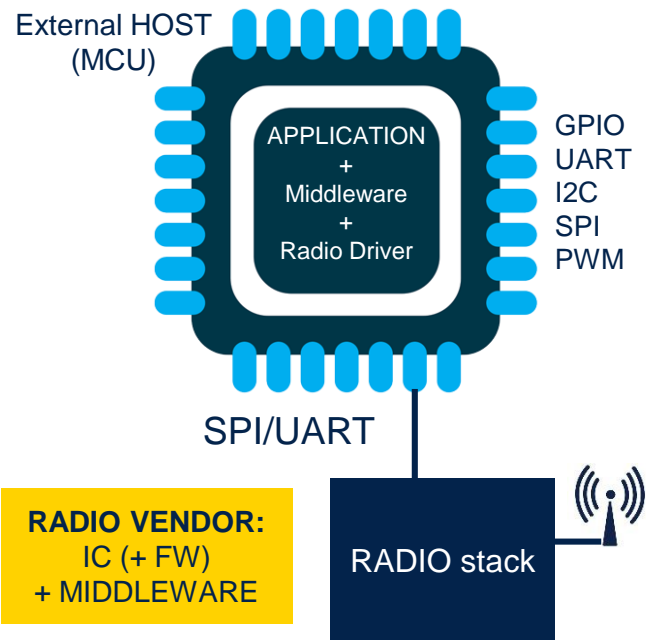
Send Sensor data to mobile App
over Bluetooth Communication

Bluetooth LE devices

BLE design flexibility

NETWORK PROCESSOR

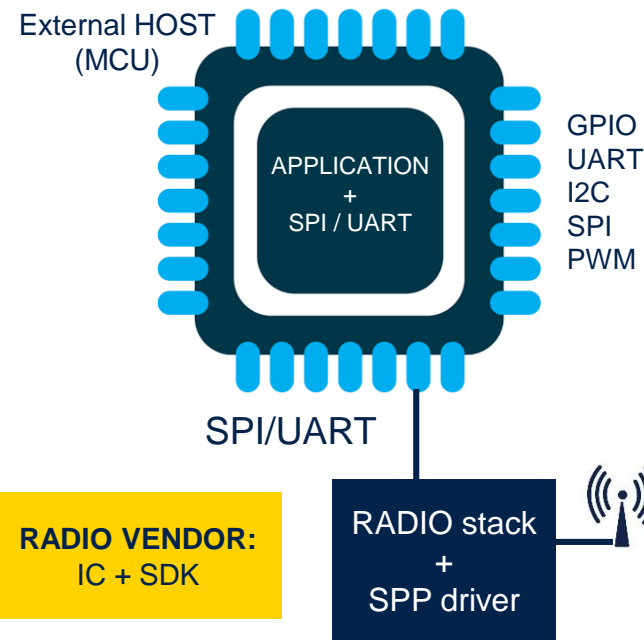
Application is running over a dedicated MCU along with **BLE middleware**



Specific integration of radio middleware/driver required

DATA-PUMP

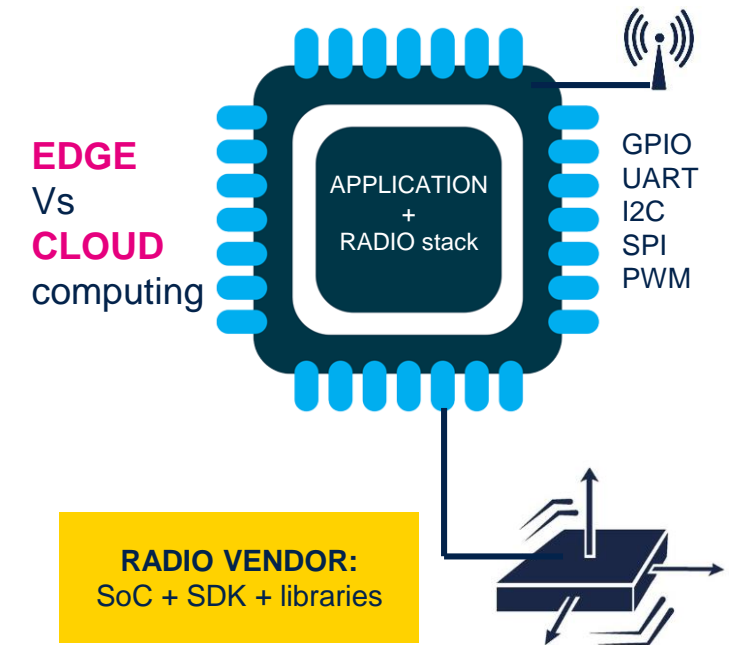
Radio link added through a simple and standard **serial interface**



Radio as a simple plug-in on a standard serial interface

APPLICATION PROCESSOR

Data acquisition, processing and radio connectivity in a single-chip



Full code ownership in all-in-one image (data, processing, radio)



BLE modules

What are the benefits?

Bluetooth® Low Energy modules allow fast time to market and huge cost savings

Modules are designed to speed up time to market

- No RF expertise required: HW/SW Connectivity is a Black-Box in your Design!
- Fast Prototyping and HW Design
- Cost efficiency for volumes up to 150kpcs per year



Modules are pre-certified

- Multi-Regional Certifications and SIG End-Product Certification allowing ~15K\$ cost saving
- Including HW Design and Certification, total cost saving up to 50~100K\$!



Modules come with scalable offer

- Cost vs. features and antenna type
- Easy migration from legacy generation SPBTLE to new generation BlueNRG-M
 - Hardware pin-to-pin compatibility and same SW resources (BlueNRG-Navigator and SDK available as well)
 - Improved performance (immunity, output power)
 - 10 years longevity (do not take care of inner components availability)





BlueNRG-M2SA/P unveiled

Full-featured solution for Bluetooth® LE connectivity

High efficiency Chip Antenna (*)

13.5 mm

Balun
BALF-NRG-02D3

32KHz XTAL to enable
ultra-low-power Bluetooth® LE sleep mode (*)

SMPS Inductor to enable
low-power Bluetooth® LE active mode (*)

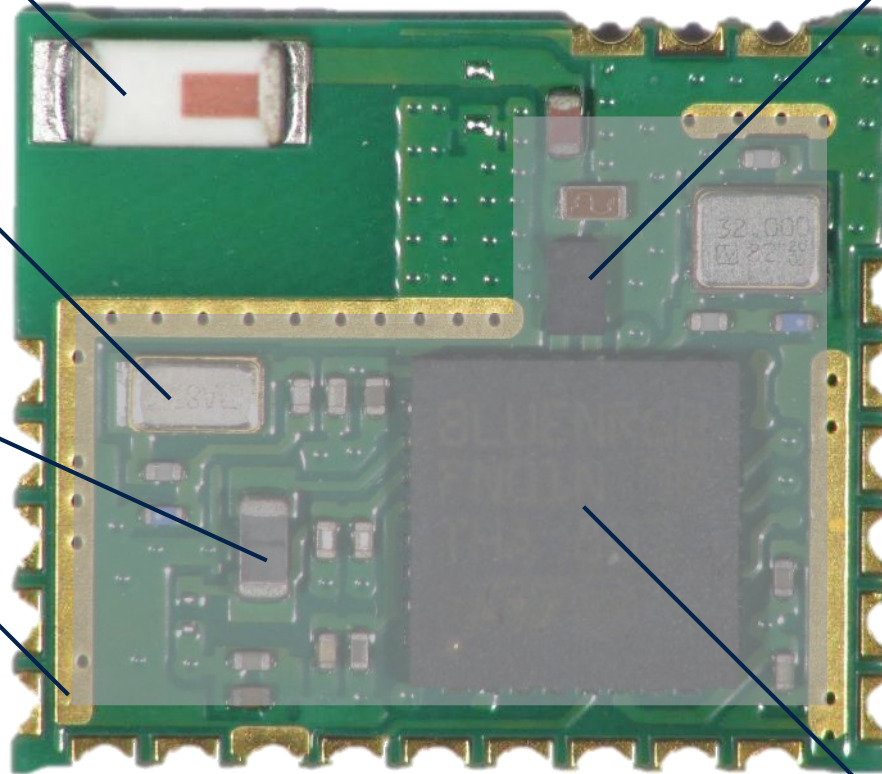
Metallic shield to protect from EMI/RFI

11.5 mm

Suitable for:

- **Motion Algorithms**
- **Voice over Bluetooth® LE**
- **Mesh over Bluetooth® LE**

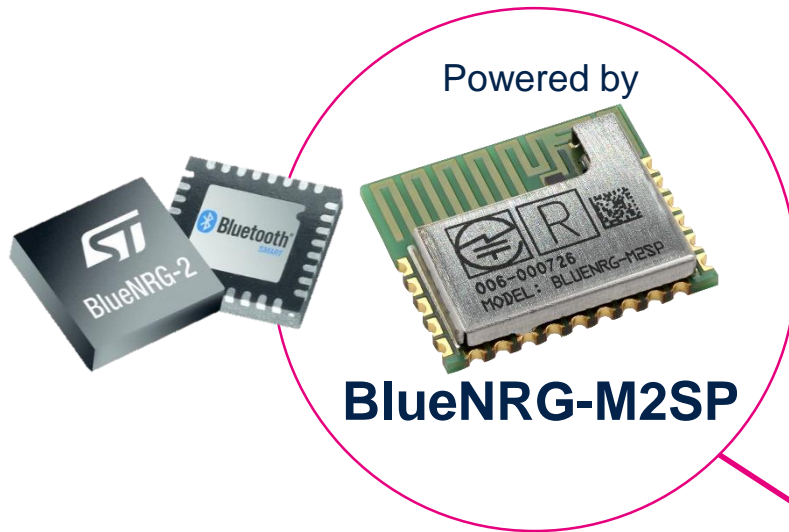
Bluetooth® Low Energy SoC
BlueNRG-232
Powered by Cortex-M0





X-NUCLEO-BNMG2A1

BlueNRG-M2SP and BlueNRG-2N evaluation board

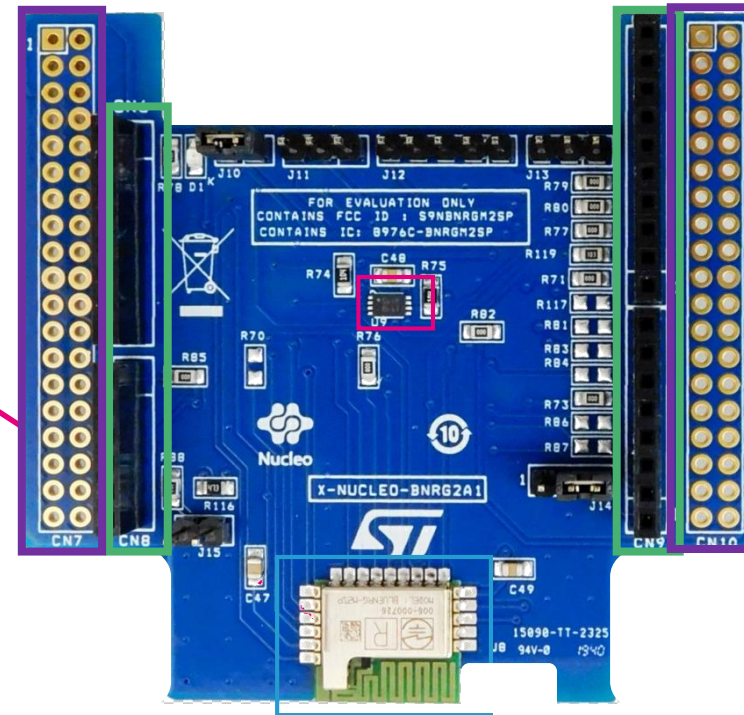


EXPANSION FW PACKAGES

- X-CUBE-BLE2 (starting v2.0.0)
- X-CUBE-BLEMESH1
- FP-SNS-BLEMESH1



BlueNRG-M2 Wireless SoC **emulating**
BlueNRG-2N Ntw Proc



BlueNRG-232, BALF-NRG-02D2,
32MHz Oscillator

M95640

Arduino UNO R3 connector

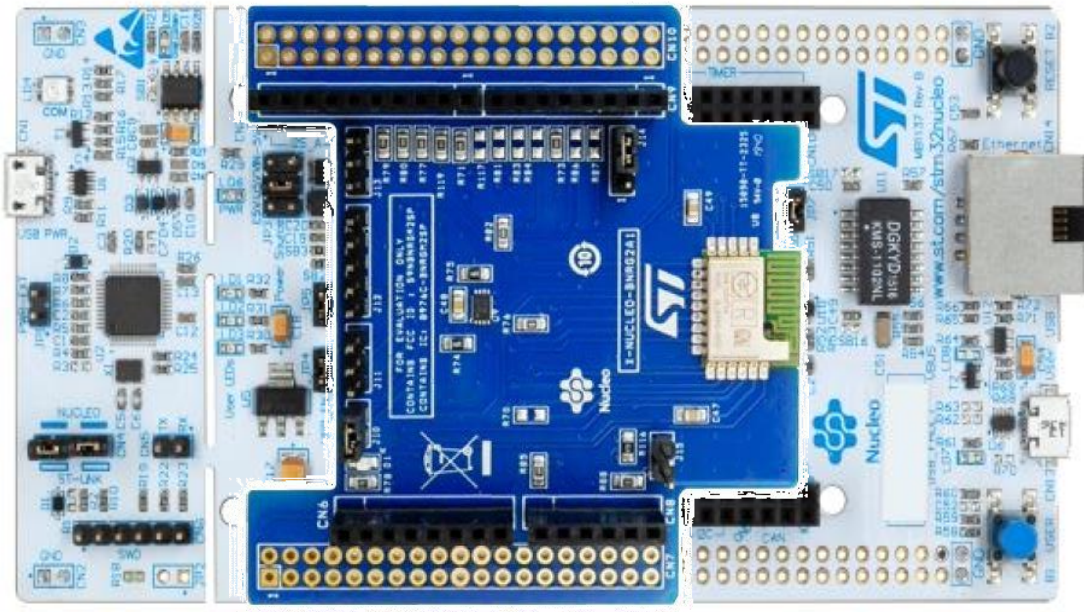
ST Morpho connector (opt)

BlueNRG
Module Family
Suitable for



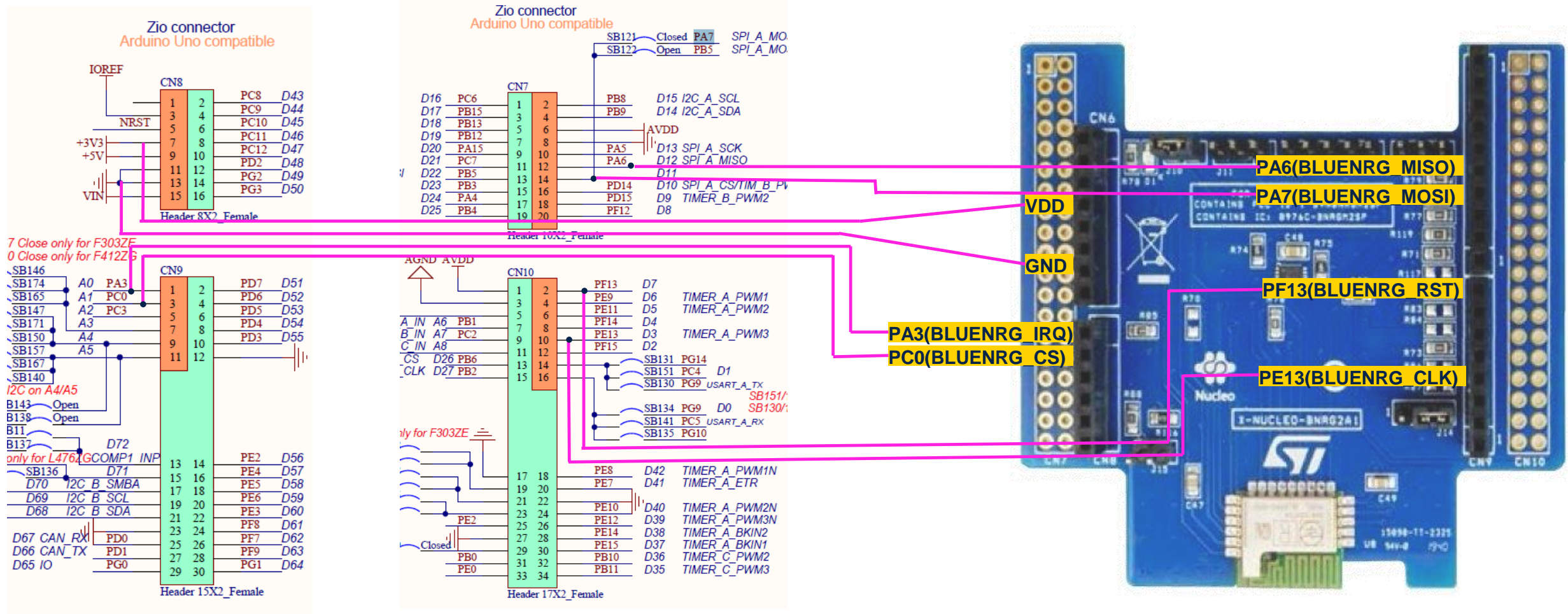
Required Hardware and Software

- The X-NUCLEO-BNRG2A1 expansion board must be plugged on top of the STM32 Nucleo board



Nucleo-L496ZG and X-NUCLEO-BNRG2A1

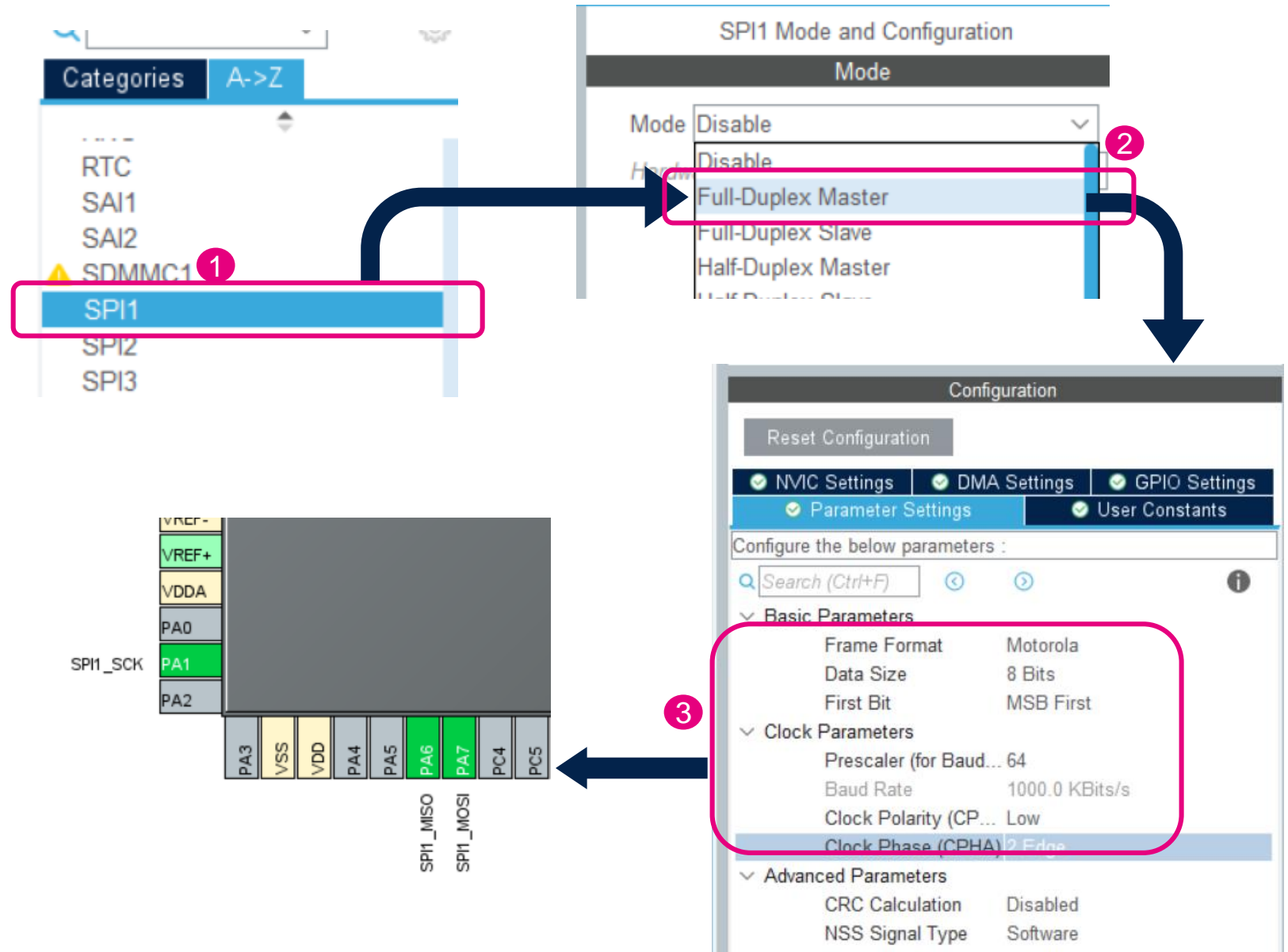
Pin connection



Live Demo – Adding BLE connectivity to Sensor Live Demo

Configure SPI communication

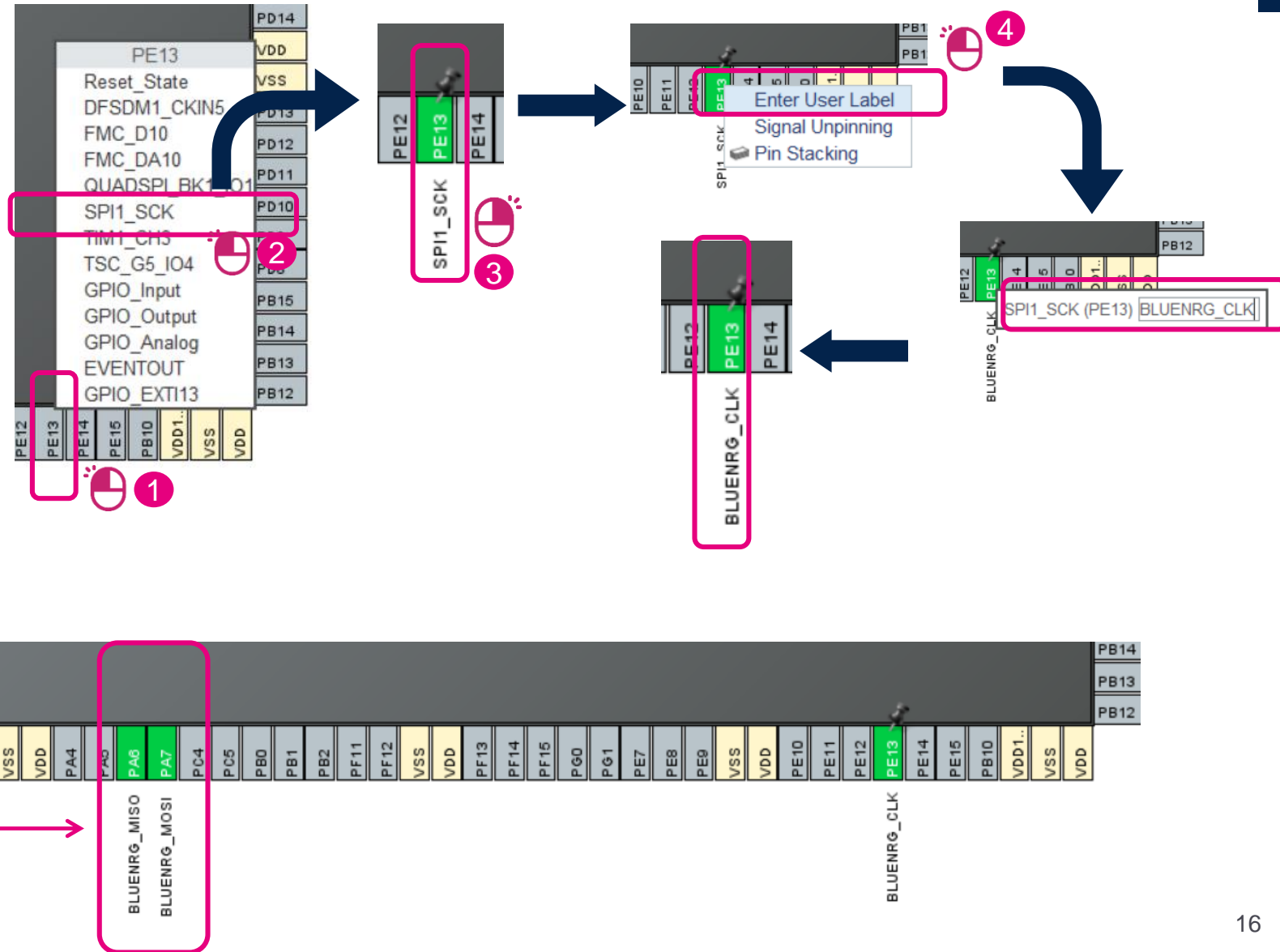
- Select SPI in **Full Duplex Master** mode
- Configure SPI as below:
 - Frame Format : **Motorola**
 - Data Size : **8 Bits**
 - First Bite : **MSP First**
 - Prescaler : **64**
 - Clock Polarity : **Low**
 - Clock Phase : **2 Edge**



Configure SPI Pins

Assign SPI PINs as Below:

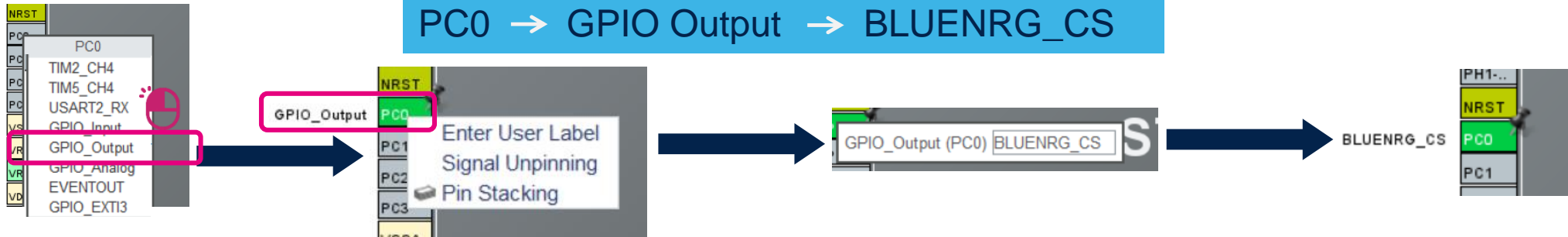
- PE13 → SPI1_CLK → BLUENRG_CLK
- PA6 → SPI1_MISO → BLUENRG_MISO
- PA7 → SPI1_MOSI → BLUENRG_MOSI



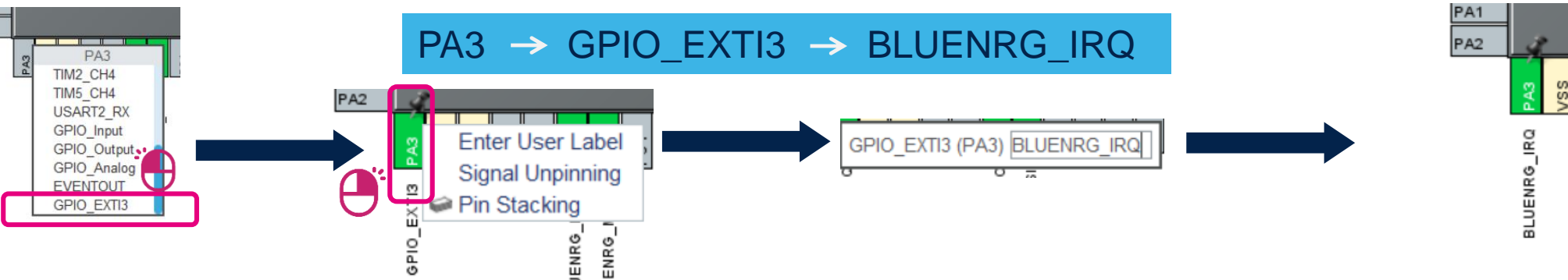
Similarly →

Configure Other Pins

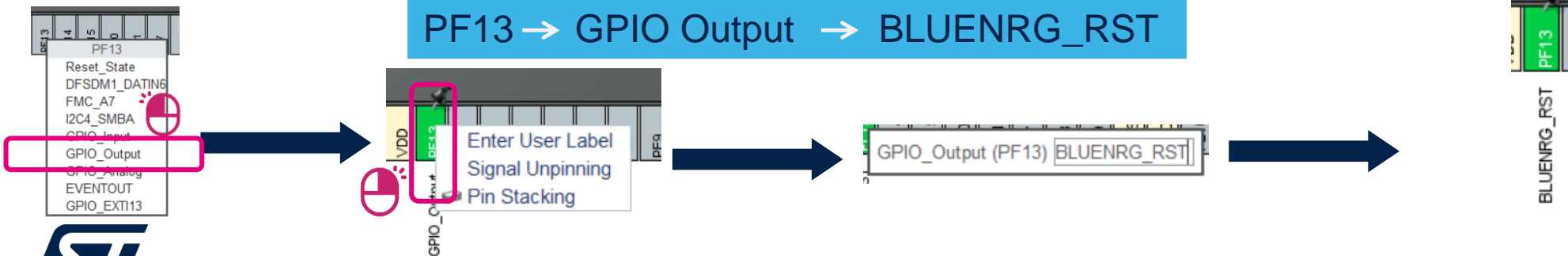
PC0 → GPIO Output → BLUENRG_CS



PA3 → GPIO_EXTI3 → BLUENRG_IRQ

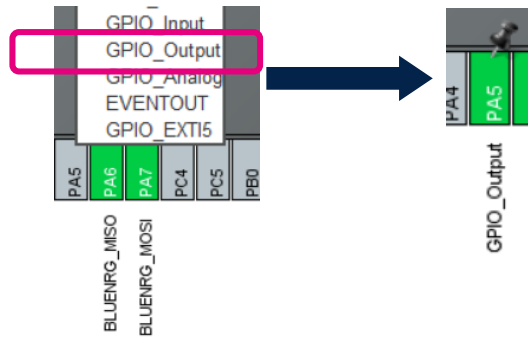


PF13 → GPIO Output → BLUENRG_RST



Configure LED, Switch and UART

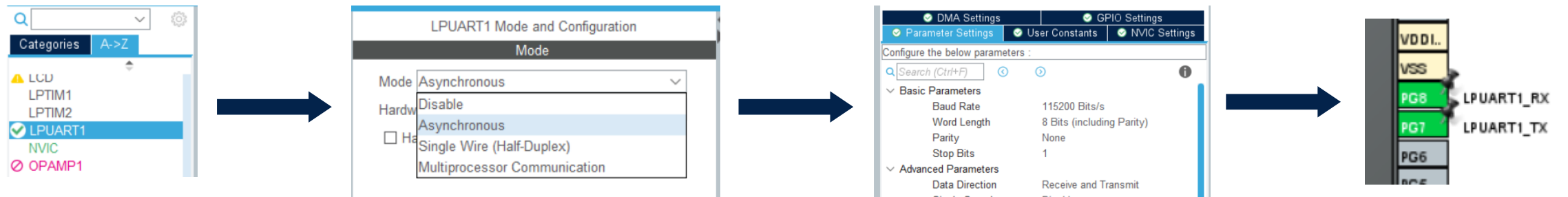
Configure PA5 → GPIO Output



Configure PC13 → GPIO_EXTI13



Configure PG8 → RX and PG7 → TX



Adding Software Pack

- In Software Packs click « **Select Components** »
- Select « **X-CUBE-BLE2** »

The screenshot shows the STM32CubeIDE interface. The 'Software Packs Component Selector' dialog is open, displaying a list of software packs. The 'X-CUBE-BLE2' pack is selected, and its components are expanded. The 'GPIO_Toggle.ioc - Pinout & Configuration' window is also open, showing the 'Pinout' tab. A red arrow points from the 'Select Components' button in the 'Software Packs Component Selector' dialog to the 'Pinout' tab in the 'GPIO_Toggle.ioc - Pinout & Configuration' window.

Software Packs Component Selector

Pack / Bundle / Component	Status	Version	Selection
> STMicroelectronics.X-CUBE-AZRTOS-L4		1.0.0	
> STMicroelectronics.X-CUBE-AZRTOS-L5		1.0.0	
> STMicroelectronics.X-CUBE-AZRTOS-WL		1.0.0	
> STMicroelectronics.X-CUBE-BLE1		6.2.3	
▼ STMicroelectronics.X-CUBE-BLE2	✓	3.3.0	
Wireless BlueNRG-2	✓	3.3.0	
BlueNRG-2	✓		
Controller	✓	3.3.0	✓
HCI_TL	✓	3.3.0	Basic
HCI_TL_INTERFACE	✓	3.3.0	UserBoard
Utils	✓	3.3.0	✓
Device BLE2_Applications	✓	3.3.0	
Application	✓	3.3.0	SensorDemoBLESensor
> STMicroelectronics.X-CUBE-DISPLAY1		2.2.0	
> STMicroelectronics.X-CUBE-EEPRMA1		3.1.1	
> STMicroelectronics.X-CUBE-GNSS1		6.0.0	
> STMicroelectronics.X-CUBE-MEMS1	✓	9.2.0	
> STMicroelectronics.X-CUBE-NFC4		2.0.4	
> STMicroelectronics.X-CUBE-NFC7		1.0.0	

GPIO_Toggle.ioc - Pinout & Configuration

1. **Select Components** (Alt-O)

Pinout view | System view

STM32L496ZGTx LQFP144

Activate Board Extension MEMs

- 1 Select Software Packs
- 2 Select STMicroelectronics X-CUBE-BLE2
- 3 Activate Board Extension BLE
- 4 Configure Platform Setting :
Select IRQ, SPI, CS, RST

Pinout & Configuration | Clock Configuration | Project Manager

Software Packs | Pinout

STMicroelectronics.X-CUBE-BLE2.3.3.0 Mode and Configuration

Mode

- ☒ Wireless BlueNRG-2
- ☒ Device BLE2 Applications

Configuration

Reset Configuration

Parameter Settings | User Constants | Platform Settings

Platform proposal

HCL TI_INTERFACE

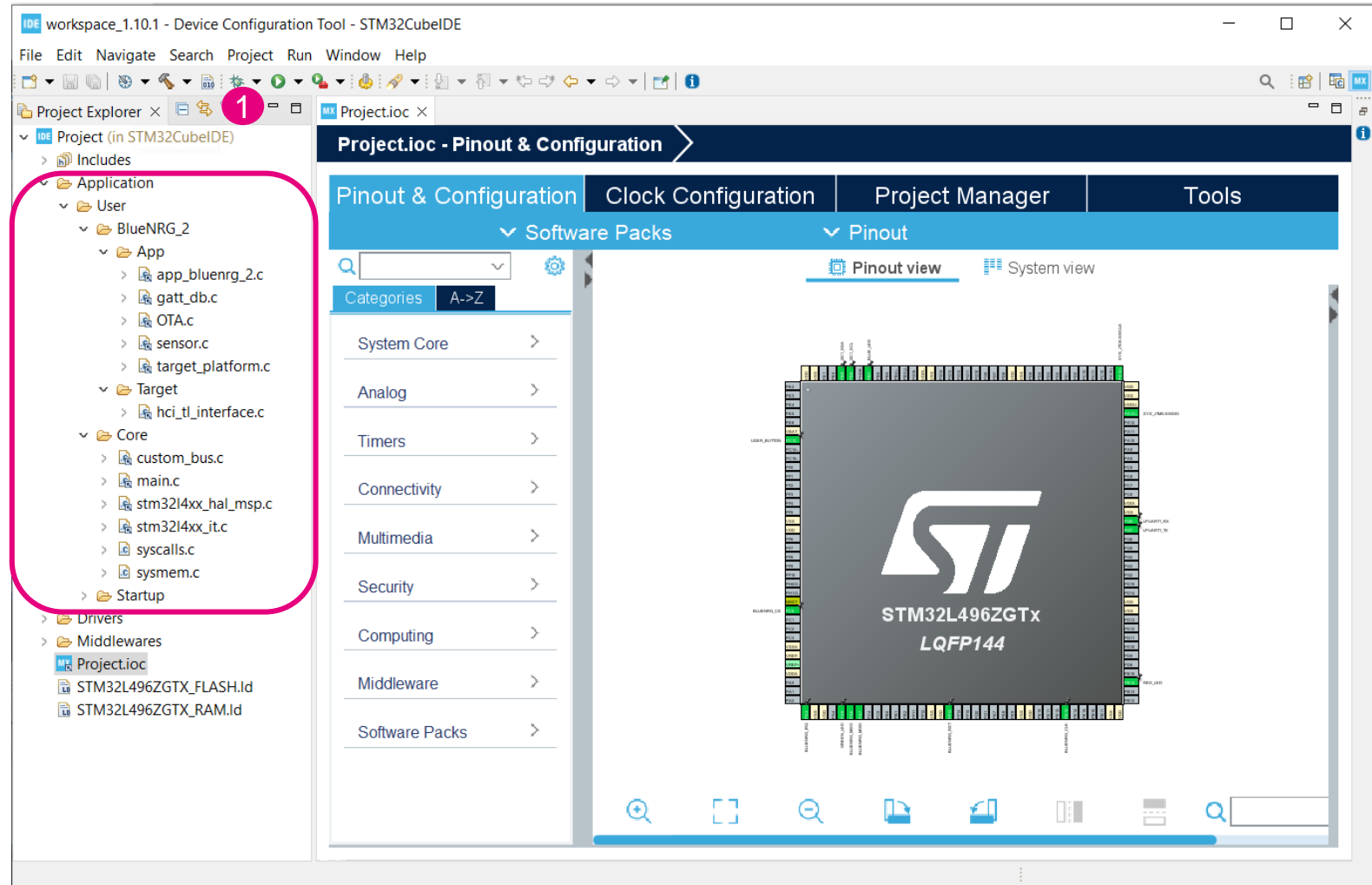
Name	IPs or Components	Found Solutions	BSP API
Exti Line	GPIO:EXTI	PA3 [BLUENRG_IRQ]	HAL_EXTI_DRIVER
BUS IO driver	SPI:Full-Duplex Master	SPI1	BSP_BUS_DRIVER
CS Line	GPIO:Output	PC0 [BLUENRG_CS]	Unknown
Reset Line	GPIO:Output	PF13 [BLUENRG_RST]	Unknown

BSP

Name	IPs or Components	Found Solutions	BSP API
BSP BUTTON	GPIO:EXTI	PC13	BSP_COMMON_DRIVER
BSP USART	LPUART:Asynchronous	LPUART1	BSP_COMMON_DRIVER
BSP LED	GPIO:Output	PA5	BSP_COMMON_DRIVER

Generate Code using STM32CubeIDE

- 1 Generate code by clicking on «Generate code icon »
- 2 All related sensor drivers will be added to your workspace



BLE Code added (main.c)

- 1 Add header files
- 2 Add sensor variable

```
Project.ioc  *main.c x
89  /* USER CODE END SysInit */
90
91  /* Initialize all configured peripherals */
92  MX_GPIO_Init();
93  MX_CRC_Init();
94  1 MX_BlueNRG_2_Init();
95  /* USER CODE BEGIN 2 */
96
97  IKS02A1_MOTION_SENSOR_Init(IKS02A1_IIS2DLPC_0 , MOTION_ACCELERO );
98
99  IKS02A1_MOTION_SENSOR_Init(IKS02A1_ISM330DHCX_0, MOTION_GYRO);
100
101  IKS02A1_MOTION_SENSOR_Init(IKS02A1_IIS2MDC_0, MOTION_MAGNETO);
102  /* USER CODE END 2 */
103
104  /* Infinite loop */
105  /* USER CODE BEGIN WHILE */
106  while (1)
107  {
108      IKS02A1_MOTION_SENSOR_GetAxes(IKS02A1_IIS2DLPC_0, MOTION_ACCELERO, &acceleration);
109      IKS02A1_MOTION_SENSOR_GetAxes(IKS02A1_ISM330DHCX_0, MOTION_GYRO, &angular_velocity);
110      IKS02A1_MOTION_SENSOR_GetAxes(IKS02A1_IIS2MDC_0, MOTION_MAGNETO, &magnetic_field);
111      /* USER CODE END WHILE */
112
113      2 MX_BlueNRG_2_Process();
114      /* USER CODE BEGIN 3 */
115  }
116  /* USER CODE END 3 */
117 }
118
119 /**
120  * @brief System Clock Configuration
121  */
```

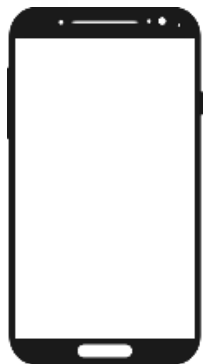
Adding code (main.c)

- ① Add sensor Initialization code
- ② Add sensor read code

```
MX GPIO_Toggle.ioc  *main.c X
82  /* Configure the system clock */
83  SystemClock_Config();
84
85  /* USER CODE BEGIN SysInit */
86
87  /* USER CODE END SysInit */
88
89  /* Initialize all configured peripherals */
90  ① MX_GPIO_Init();
91  /* USER CODE BEGIN 2 */
92
93  IKS02A1_MOTION_SENSOR_Init(IKS02A1_IIS2DLPC_0, MOTION_ACCELERO );
94  IKS02A1_MOTION_SENSOR_Init(IKS02A1_ISM330DHCX_0, MOTION_GYRO);
95  IKS02A1_MOTION_SENSOR_Init(IKS02A1_IIS2MDC_0, MOTION_MAGNETO);
96
97  /* USER CODE END 2 */
98
99  /* Infinite loop */
100 /* USER CODE BEGIN WHILE */
101 while (1)
102 {
103     /* USER CODE END WHILE */
104     ② /* USER CODE BEGIN 3 */
105     IKS02A1_MOTION_SENSOR_GetAxes(IKS02A1_IIS2DLPC_0, MOTION_ACCELERO, &acceleration);
106     IKS02A1_MOTION_SENSOR_GetAxes(IKS02A1_ISM330DHCX_0, MOTION_GYRO, &angular_velocity);
107     IKS02A1_MOTION_SENSOR_GetAxes(IKS02A1_IIS2MDC_0, MOTION_MAGNETO, &magnetic_field);
108 }
109
110 /* USER CODE END 3 */
```



Companion ST BLE Sensor app



ON YOUR PHONE

- Look for “**ST BLE Sensor**” on the App Store or Google Play





Source code

BlueST SDK for Android and IOS

Source code

Available online and maintained on GitHub repos

Android

https://github.com/stmicroelectronics-centrallabs/bluestsdk_android

iOS

https://github.com/stmicroelectronics-centrallabs/bluestsdk_ios

GitHub This repository Search Explore Features Enterprise Pricing

STMicroelectronics-Centrallabs / **BlueSTSDK_Android** Watch 3

Bluetooth low energy Sensors Technology Software Development Kit (Android version)

2 commits 2 branches 0 releases 1 contributor

Branch: master BlueSTSDK_Android / +

GiovanniVisentini Update README Latest commit 573ce4 10 days ago

BlueSTExample	Initial commit	14 days ago
BlueSTSDK	Initial commit	14 days ago
.gitignore	Initial commit	14 days ago
LICENSE	Initial commit	14 days ago
README.md	Update README	10 days ago
build.gradle	Initial commit	14 days ago
settings.gradle	Initial commit	14 days ago

STMicroelectronics-Centrallabs / **BlueSTSDK_iOS**

Bluetooth low energy Sensors Technology Software Development Kit (iOS version)

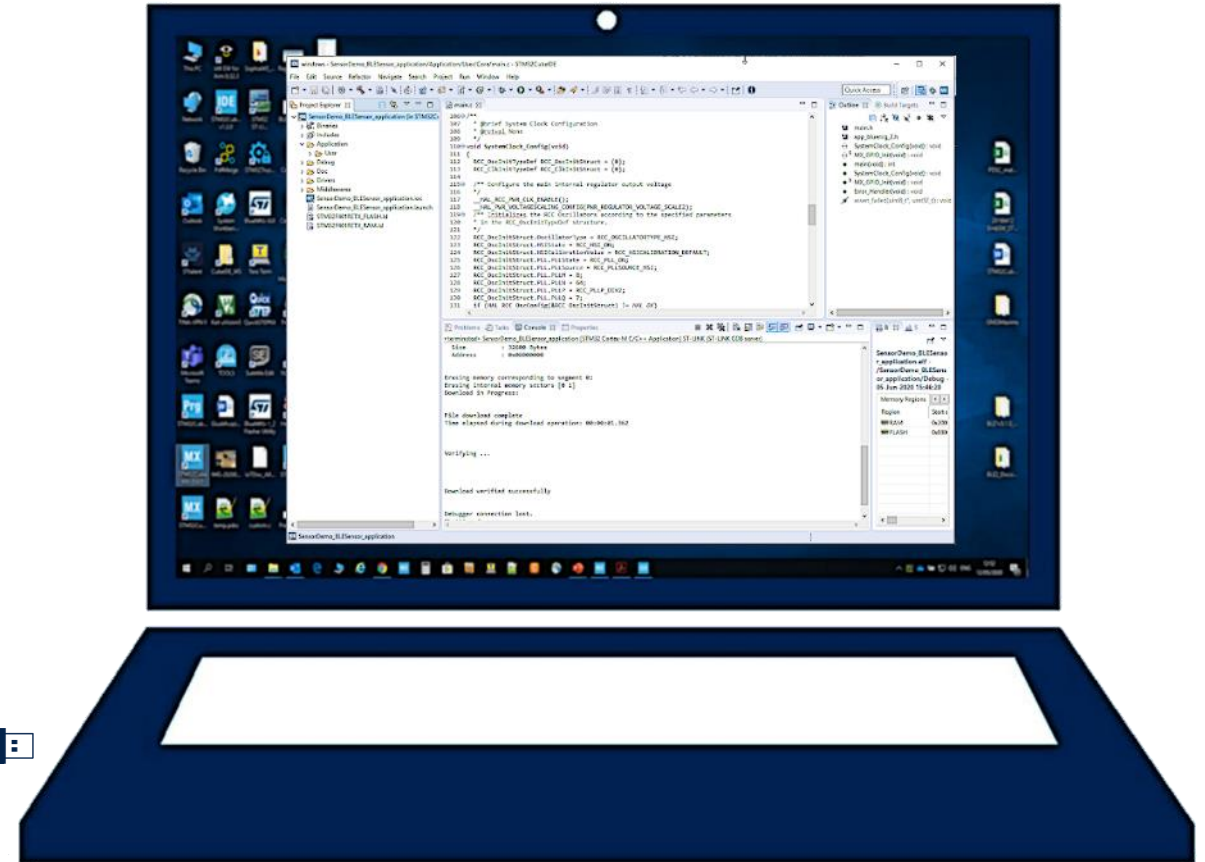
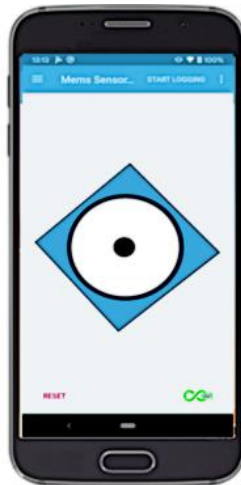
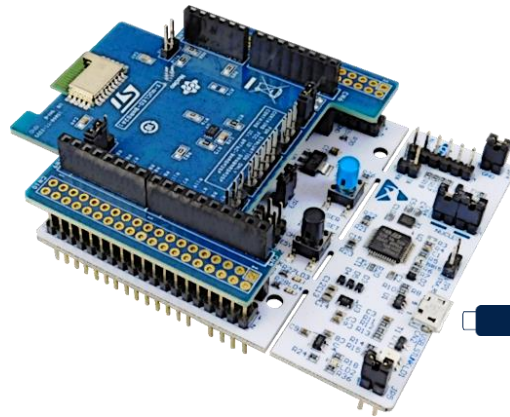
2 commits 2 branches 0 releases 1 contributor

Branch: master BlueSTSDK_iOS / +

GiovanniVisentini fix parse error with multiple feature in the same characteristics Latest commit c0e8d1 2 days ago

BlueSTSDK	fix parse error with multiple feature in the same characteristics	2 days ago
BlueSTSDKExample	fix parse error with multiple feature in the same characteristics	2 days ago
LICENSE	Initial commit	9 days ago
Readme.md	Initial commit	9 days ago

Launch the ST BLE Sensor app on your Android/iOS device and start playing with the application



Our technology starts with You



Find out more at www.st.com

© 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.



life.augmented