



## **DecaWave: UWB-IOT tools Infrastructure BSP UserGuide**

*The authorized version of this document is an electronic master stored in the Document Repository (PP Intranet Portal / PAL). It is advised that the version of the document in the repository be matched with the hard copy before using it. The information contained in this document is proprietary to PathPartner Technology Consulting Pvt Ltd.*

**Copyright** © PathPartner Technology Consulting Pvt Ltd. This material, including documentation and any related computer programs, is protected by copyright controlled by PathPartner Technology. All rights are reserved. Copying, including reproducing, storing, adapting or translating, any or all of this material requires the prior written consent of the authorized personnel of PathPartner Technology.

*This material contains confidential information, which shall not be disclosed to others without the prior written consent of the authorized personnel of PathPartner Technology.*

**Table of Contents**

<b>1.</b>	<b>About this Manual.....</b>	<b>3</b>
<b>2.</b>	<b>Getting Started .....</b>	<b>3</b>
2.1.	System Requirements.....	3
2.1.1.	Installing Jlink.....	3
2.2.	Running DecaRange RTLS ARM Application on Nordic NRF52840 and STM32F401RE Nucleo platforms.....	3
2.2.1.	Common Steps for all examples .....	3
2.2.2.	Running DecaRange RTLS ARM Application.....	4
2.3.	Connecting Hardware Platform to EVM .....	5
2.3.1.	NRF52840 Nordic .....	5
2.3.2.	STM32F401RE Nucleo.....	5
2.4.	BSP Overview .....	6
2.4.1.	core .....	6
2.4.2.	boards.....	6
2.4.3.	cpu .....	6
2.4.4.	cpu/periph .....	7
2.4.5.	drivers.....	7
2.4.6.	sys .....	7
2.4.7.	sys/net .....	7
2.4.8.	pkg .....	7
2.4.9.	examples.....	7
2.4.10.	tests.....	7
2.4.11.	dist and doc.....	7
2.5.	BSP Code / Directory Organization .....	7
2.5.1.	Top Level .....	7
2.5.2.	Decawave Board Module for Nordic NRF52840.....	8
2.5.3.	Decawave Board Module for STM32F401RE Nucleo .....	8
2.5.4.	Docs.....	8
2.5.5.	Examples.....	9

## 1. About this Manual

This document describes how to build and work with the BSP for Decawave on RIOT for Nordic NRF52840 and STM32F401RE Nucleo platforms. The BSP package for Decawave on RIOT serves to provide a software platform for development, deployment and execution of DecaRange RTLS ARM application.

In this context, the document contains instructions for:

- Setting up the Environment
- Build the DecaRange RTLS ARM application for defined platform
- Flash the DecaRange RTLS ARM application for defined platform

## 2. Getting Started

### 2.1. System Requirements

Refer to 'Dependencies' section of release notes

#### 2.1.1. Installing Jlink

- To flash the binary JLink is use, please install the tool manually from <https://www.segger.com/downloads/jlink>.
- Expand J-Link Software and Documentation Pack and select the suitable package and click download.
- Double click on the downloaded package to install.

### 2.2. Running DecaRange RTLS ARM Application on Nordic NRF52840 and STM32F401RE Nucleo platforms

This section describes how to run the application involving BSP for Decawave on RIOT

#### 2.2.1. Common Steps for all examples

- Installing Dependent Libraries only for one time
- Inside the cloned repo run the script - `"/setup.sh INITIAL"`
- Setup without installing Dependent Libraries (For Next Time on Wards)
- Inside the cloned repo run the script - `"/setup.sh"`
- To connect EVB1000 board to Nordic NRF52840 or STM32F401RE Nucleo External Microcontroller
- Turn off S2 DIP switches
- Turn off S3-1 DIP switch
- Remove the jumper J10
- Make sure the pins of Nordic NRF52840 or STM32F401RE Nucleo Platforms and EVB1000 Device are wired as defined below

## BSP UserGuide

## EVB 1000 to Nordic NRF52840

Pin	EVB1000	Nordic NRF52840
IRQ	J6: Pin4	P0.30
MISO	J6: Pin5	P0.28
MOSI	J6: Pin8	P0.04
SCLK	J6: Pin7	P0.03
CS	J6: Pin9	P0.29
GND	J6: Pin10	GND
VDD	J10: Pin2	VDD

## EVB 1000 to STM32F401RE Nucleo

Pin	EVB1000	STM32F401RE Nucleo
IRQ	J6: Pin4	PA06
MISO	J6: Pin5	PB04
MOSI	J6: Pin8	PB05
SCLK	J6: Pin7	PB03
CS	J6: Pin9	PA04
GND	J6: Pin10	GND
VDD	J10: Pin2	3V3

## 2.2.2. Running DecaRange RTLS ARM Application

- Build
- Inside Ubuntu, open a terminal and input –  
"cd riot-dw1000/RIOT"
- To compile – "make -C examples/decarange\_rtls\_arm/ BOARD=  
**HW\_Platform**"
  - For Nordic Platform - HW\_Platform is "nrf52dk"
    - make -C examples/decarange\_rtls\_arm/ BOARD = nrf52dk"
  - For STM32 Nucleo F401RE - **HW\_Platform** is "nucleo-f401"
- To flash – "sudo make -C examples/dw1000\_rtls\_arm/ BOARD=  
**HW\_Platform** flash"
  - For Nordic Platform - HW\_Platform is "nrf52dk"
  - For STM32 Nucleo F401RE - **HW\_Platform** is "nucleo-f401"
- open a new terminal and input –  
"cd riot-dw1000/RIOT/dist/tools/pyterm"  
"sudo ./pyterm -p /dev/ttyACM0"
- Select the EVB1000 device configuration such as  
Mode(Mode1/Mode2/Mode3/Mode4), UNIT(Tag/Anchor) and UNIT ID (0 to 2)  
For Example:

## BSP UserGuide

- If the user selects Mode as 'Mode1', UNIT as 'Tag' and UNIT ID as '0' then EVB1000 with Nordic NRF52840 or STM32F401RE Nucleo is configured for Tag0 with Mode1 Configuration
- In Case of Nordic NRF52840, Configure A0, A1 and A2 for Mode 1 configuration as defined above
- In Case of STM32F401RE Nucleo, Select the EVB1000 with STM32F10x (internal controller) as A0, A1 and A2 for Mode 1 configuration
- Now calculated range distance is printed on the terminal

## 2.3. Connecting Hardware Platform to EVM

### 2.3.1. NRF52840 Nordic

#### Interfaces

SPI interface with EVB1000 is highlighted below

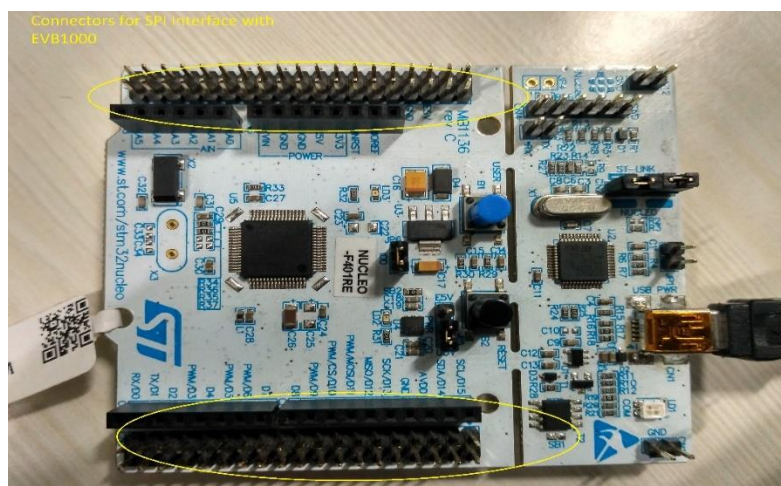


### Connectors for SPI Interface on NRF52840 Nordic Platform to EVB1000

### 2.3.2. STM32F401RE Nucleo

#### Interfaces

SPI interface with EVB1000 is highlighted below

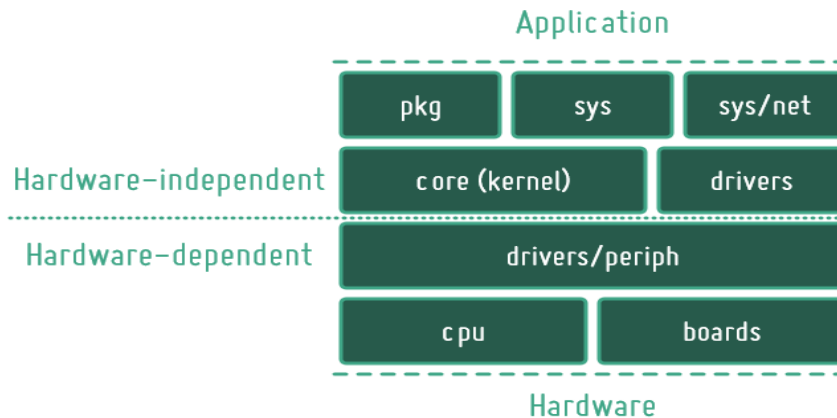


### Connectors for SPI Interface on STM32F401RE Nucleo Platform to EVB1000

## 2.4. BSP Overview

RIOT Board support package supports generic BSP module for different H/W Platforms which helps to integrate an application easily for different H/W Platforms. Based on RIOT, we have come up with BSP module-specific design for Decawave peripheral.

Software layer structure of the DecaRange RTLS ARM application based on RIOT is shown below



**Software layer for DecaRange RTLS ARM application**

RIOT Board support package is structured into five groups.

- The kernel (core)
- Platform specific code (cpu; boards)
- Device drivers (drivers)
- Libraries and network code (sys; pkg)
- Applications for demonstrating features and for testing (examples; tests)

The following list gives a more detailed description of BSP top-level directories

### 2.4.1. core

It contains the actual kernel. The kernel consists of the scheduler, inter-process-communication (messaging), threading, thread synchronization, and supporting data-structures and type definitions

### 2.4.2. boards

The platform dependent code is split into two logic elements: CPUs and boards, a board has exactly one CPU, while a CPU can be part of multiple boards. This configuration mainly includes the peripheral configuration and pin-mapping, the configuration of on-board devices, and the CPU's clock configuration

### 2.4.3. cpu

It contains all CPU specific configurations, such as implementations of power management, interrupt handling and vectors, startup code, clock initialization code and thread handling code

**2.4.4. cpu/periph**

It contains CPU's peripheral drivers like SPI, UART, GPIO, etc.,

**2.4.5. drivers**

It contains the drivers for external devices such as network interfaces, sensors and actuators. All of device drivers are based on the peripheral driver API (e.g. SPI, GPIO, etc.) and other modules like the xtimer. This way the drivers are completely platform agnostic and they don't have any dependencies into the CPU and board code

**2.4.6. sys**

It contains modules that are not part of the hardware abstraction nor device. The libraries include data structures (e.g. bloom, color), crypto libraries (e.g. hashes, AES), high-level APIs (e.g. Posix implementations), memory management (e.g. malloc), the RIOT shell etc.,

**2.4.7. sys/net**

The sys/net contains networking code

**2.4.8. pkg**

It comes with support for a number of external libraries (e.g. ccn-lite, microcoap)

**2.4.9. examples**

It contains example applications that demonstrate certain features

**2.4.10. tests**

It contains test application of many features/modules in RIOT

**2.4.11. dist and doc**

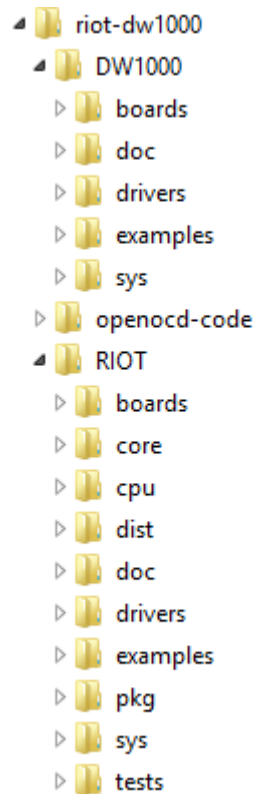
It contains all the tooling around RIOT

"doc" contains the doxygen configuration and also contains the compiled doxygen output after running "make doc"

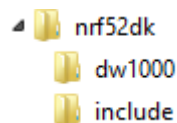
"dist" contains tools for help with RIOT. These include the serial terminal application pyterm, generic scripts for flashing, debugging, resetting (e.g. support for OpenOCD, Jlink), as well as code for enabling easy integration to open testbeds such as the IoT-LAB

**2.5. BSP Code / Directory Organization****2.5.1. Top Level**

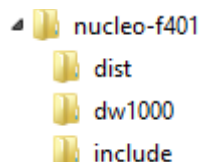
BSP source code directory will have following folders. Let's consider 01\_00\_03 version release as an example

**BSP UserGuide****2.5.2. Decawave Board Module for Nordic NRF52840**

The boards folder contains platform specific folder such as nrf52dk etc., The EVB1000 related changes for Nordic NRF52840 has been defined in "boards/nrf52dk/dw1000" folder. This configures peripheral, interrupt configuration and pin-mapping specific to EVB1000. Any special programming needed by the EVB1000 board is also added here.

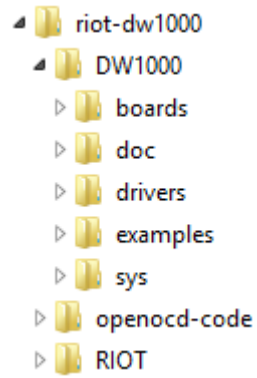
**2.5.3. Decawave Board Module for STM32F401RE Nucleo**

The boards folder contains platform specific folder such as nucleo-f401 etc., The EVB1000 related changes for STM32F401-Nucleo has been defined in "boards/ nucleo-f401/dw1000" folder. This configures peripheral, interrupt configuration and pin-mapping specific to EVB1000. Any special programming needed by the EVB1000 board is also added here.

**2.5.4. Docs**

The doc folder contains release notes, user guide and test Reports for Nordic NRF52840 and STM32F401RE Nucleo platforms. The relnotes\_archive folder contains the user guides and release notes for the previous releases.



**BSP UserGuide****2.5.5. Examples**

The examples folder is the root folder for all the DecaRange RTLS ARM application for various supported platforms.

