Contiki 6LoWPAN Quick Guide

Contiki on STM32 Nucleo plugged with Sub-1 GHz RF expansion board (X-NUCLEO-IDS01A4, X-NUCLEO-IDS01A5)

life.augmented

Version 1.0 (Sept. 22, 2015)

- Contiki (*) is an open source operating system (OS) for the Internet of Things (IoT)
- ST has developed a Contiki 3.x port for the STM32 Nucleo board (NUCLEO) plugged with the supported expansion boards (X-NUCLEO)
- The guide explains how to quickly get started with this platform

(*) Information on Contiki OS are available at www.contiki-os.org

ST port overview

- The ST port allows running the Contiki OS, 6LoWPAN protocol stack and related applications on an STM32 Nucleo board plugged with one sub-1 GHz RF expansion board and, optionally, one motion MEMS and environmental sensors expansion board

- Software available for download on GitHub repository: https://github.com/STclab/contiki/tree/stm32nucleo-spirit1

- Boards supported:
  - NUCLEO-L152RE based on the STM32L152RET6 ultra-low power microcontroller
  - X-NUCLEO-IDS01A4 based on sub-1 GHz SPSGRF-868 SPIRIT1 module (operating at 868 MHz)
  - X-NUCLEO-IDS01A5 based on sub-1 GHz SPSGRF-915 SPIRIT1 module (operating at 915 MHz)
  - X-NUCLEO-IKS01A1 based on motion MEMS and environmental sensors (optional)

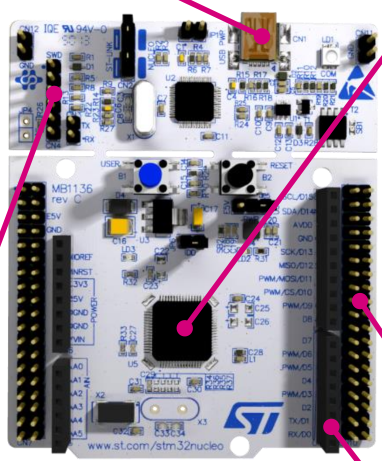
- License: BSD-3 (same as the Contiki distribution license)

STM32 Nucleo
Development Boards (NUCLEO)

- A comprehensive range of affordable development boards for the entire STM32 microcontroller series, with unlimited unified expansion capabilities and integrated debugger/programmer functionality.

Power supply through USB or external source

Integrated debugging and programming ST-LINK probe



STM32 microcontroller

Complete product range
from ultra-low power to high-performance

ST morpho extension header

Arduino™ UNO R3 extension headers

STM32 Nucleo

Sub-1 GHz RF expansion boards
Overview

Description

- The X-NUCLEO-IDS01A4, X-NUCLEO-IDS01A5 are evaluation boards based on the SPIRIT1 RF modules SPSGRF-868 and SPSGRF-915
- The SPIRIT1 module communicates with the STM32 Nucleo board host microcontroller through an SPI link available on the Arduino UNO R3 connector.

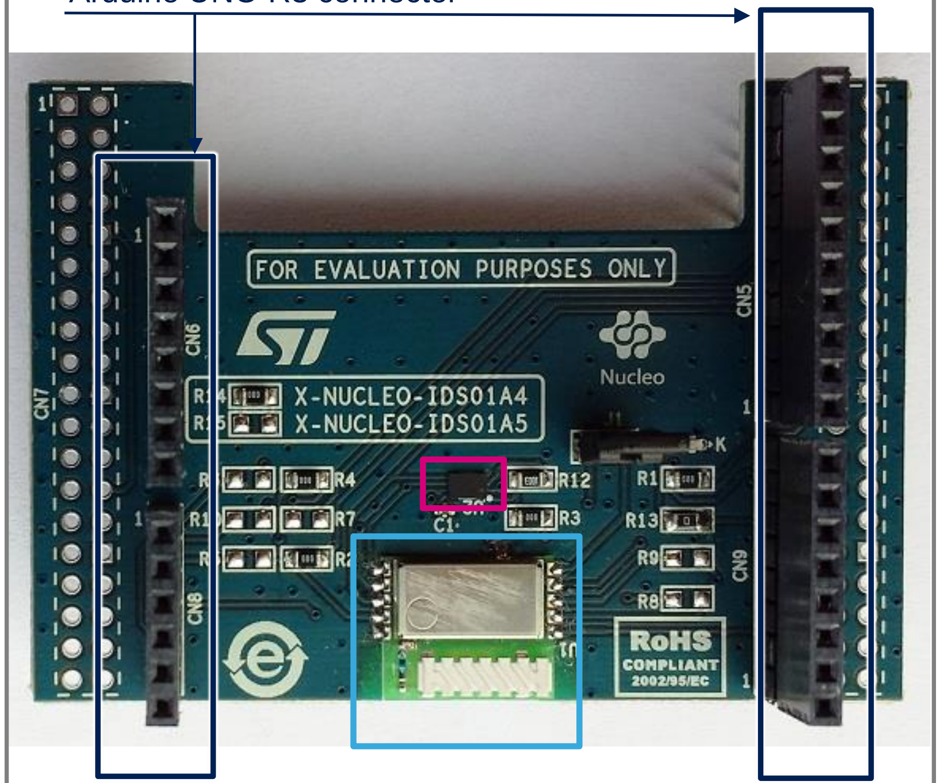
Key products on board

SPSGRF
SPIRIT1 (Low data-rate, low-power sub-1GHz transceiver) module

M95640-RMC6TG
64-Kbit serial SPI bus EEPROM

Latest info available at
X-NUCLEO-IDS01A4
X-NUCLEO-IDS01A5

Arduino UNO R3 connector



SPSGRF-868 or SPSGRF-915 (*)

EEPROM

Order code: **X-NUCLEO-IDS01A4, X-NUCLEO-IDS01A5**

(*) Identification of the operating frequency of the X-NUCLEO-IDS01Ax (x=4 or 5) is performed through two resistors (R14 and R15).

Motion MEMS and environmental sensor expansion board

Description

- The X-NUCLEO-IKS01A1 is a motion MEMS and environmental sensor evaluation board.
- It is compatible with the Arduino UNO R3 connector layout, and is designed around ST's sensors.

Key products on board

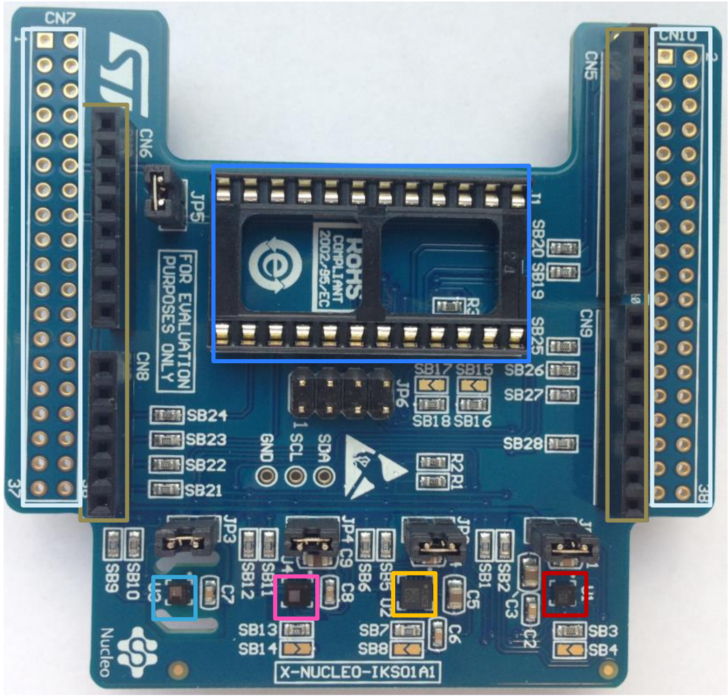
LSM6DS0: MEMS 3D accelerometer (±2/±4/±8 g) + 3D gyroscope (±245/±500/±2000 dps)

LIS3MDL: MEMS 3D magnetometer (±4/ ±8/ ±12/ 16 gauss)

LPS25HB: MEMS pressure sensor, 260-1260 hPa absolute digital output barometer

HTS221: **C**apacitive digital relative humidity and temperature

DIL 24-pin: Socket available for additional MEMS adapters and other sensors (UV index)



HTS221 LSM6DS0 ST morpho connector**

LPS25HB LIS3MDL Arduino UNO R3 connector

DIL 24-pin

Latest info available at X-NUCLEO-IKS01A1

Order code: X-NUCLEO-IKS01A1

** Connector for the STM32 Nucleo Board

Hardware prerequisites

- 1 x NUCLEO-L152RE (STM32 Nucleo board)
- 1 x X-NUCLEO-IDS01A4 (Sub-1 GHz RF expansion board based on the SPSGRF-868 module) or 1 x X-NUCLEO-IDS01A5 (Sub-1 GHz RF expansion board based on the SPSGRF-915 module)
- (OPTIONAL) 1 x X-NUCLEO-IKS01A1 (Motion MEMS and environmental sensor expansion board)
- Laptop/PC with Windows 8/7 or Linux Ubuntu 15.4
- 1 x USB type A to Mini-B USB cable

NUCLEO-L152RE

X-NUCLEO-IDS01A4 or X-NUCLEO-IDS01A5

Mini USB

X-NUCLEO-IKS01A1 (OPTIONAL)

Software prerequisites (1/2)

- Download the ST port from GitHub repository (https://github.com/STclab/contiki/tree/stm32nucleo-spirit1)

- The ST port is installed automatically when the Contiki and sub-module repositories are cloned. It can be done using the following commands:
  - git clone https://github.com/STclab/contiki.git
    cd contiki/
    git checkout stm32nucleo-spirit1
    git submodule init
    git submodule update

- Contiki Platform name for ST port: stm32nucleo-spirit1

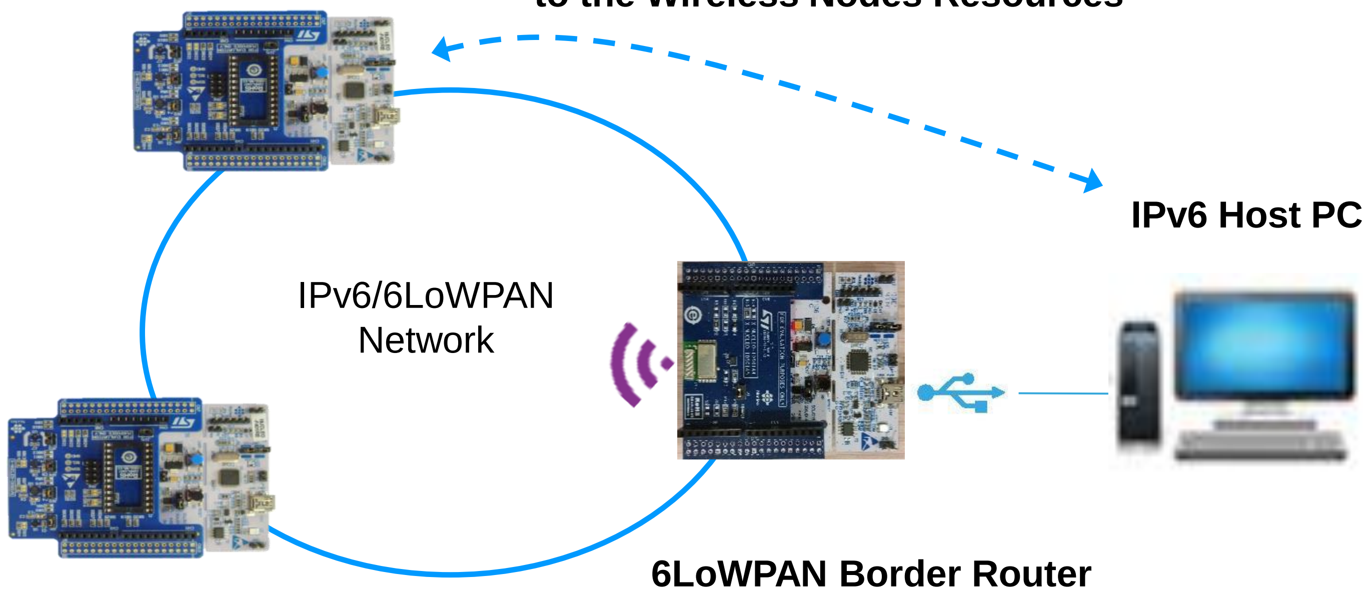
Software prerequisites (2/2)

- PC software
  - Windows PC:
    - Linux environment on Windows using Cygwin (Link)
    - GCC is provided in the System Workbench for STM32 (SW4STM32) (Link)
    - Git package for Cygwin or Git for Windows (Link)
    - WinPcaP (for demo purpose) (Link)
  - Linux PC:
    - GNU Tools for ARM Embedded Processors (Link)
- Firefox web browser (Link)
- Firefox Copper plug-in (only for CoAP demo purpose) (Link)

Demo Overview

CoAP REST Access
to the Wireless Nodes Resources

IPv6 Host PC



6LoWPAN Border Router
NUCLEO-L152RE
X-NUCLEO-IDS01A4/5 (sub-1 GHz)

Wireless Sensors Nodes
NUCLEO-L152RE
X-NUCLEO-IDS01A4/5 (sub-1 GHz)
X-NUCLEO-IKS01A1 (sensors) [optional]

Contiki on STM32 Nucleo in a few steps (1/2)

1 Clone the online repository

```
git clone https://github.com/STclab/contiki.git
cd contiki/
git checkout stm32nucleo-spirit1
git submodule init
git submodule update
```

2 Compile the FW for a wireless node: REST example (using the standard Contiki provided “er-rest-example”)

```
cd examples/er-rest-example
make TARGET=stm32nucleo-spirit1 USE_SUBGHZ_BOARD=IDS01A5
arm-none-eabi-objcopy -O binary er-example-server.stm32nucleo-spirit1 er-example-server.bin
```

3 Connect the wireless sensor board to a PC USB slot and program the device

Copy the “er-example-server.bin” file (e.g. drag & drop) to the USB mass storage corresponding to the STM32 Nucleo board

Contiki on STM32 Nucleo in a few steps (2/2)

4

Compile the FW for the Border Router node

```
cd examples/ipv6/rpl-border-router
make TARGET=stm32nucleo-spirit1 USE_SUBGHZ_BOARD=IDS01A5
arm-none-eabi-objcopy -O binary border-router.stm32nucleo-spirit1 br.bin
```

5

Connect the board to USB and program the device

copy the “br.bin” file
(e.g. drag & drop) to the USB mass storage corresponding to the STM32 Nucleo board

Contiki on STM32 Nucleo in a few steps

6 Setup the IPv6 Host PC for IP traffic bridging between host and 6LowPAN border Router

Windows PC setup (Win 7/8)
using “**wpcapslip6**” utility

OR

Linux PC setup (Ubuntu)
using “**tunslip6**” utility

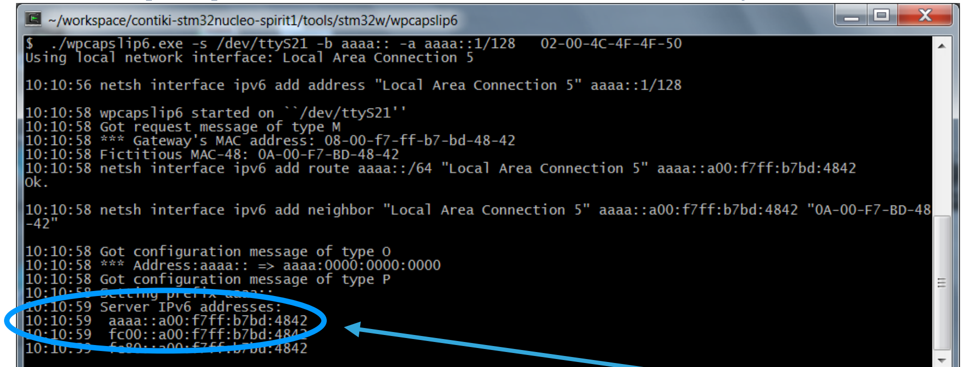
1. wpcapslip6 needs a working network adapter:
The Microsoft loopback adapter can be installed via “Add legacy hardware” in the Windows Device Manager (reboot is needed after installation of the loopback adapter)
2. Copy “cygwin1.dll” from “contiki/tools/cygwin” to wcapslip6 folder
3. Install WinPcaP
4. run Cygwin as administrator

```
cd ./tools
cc tunslip6.c –o tunslip6
sudo ./tunslip6 –s /dev/ttyACM0 aaaa::1/64
```

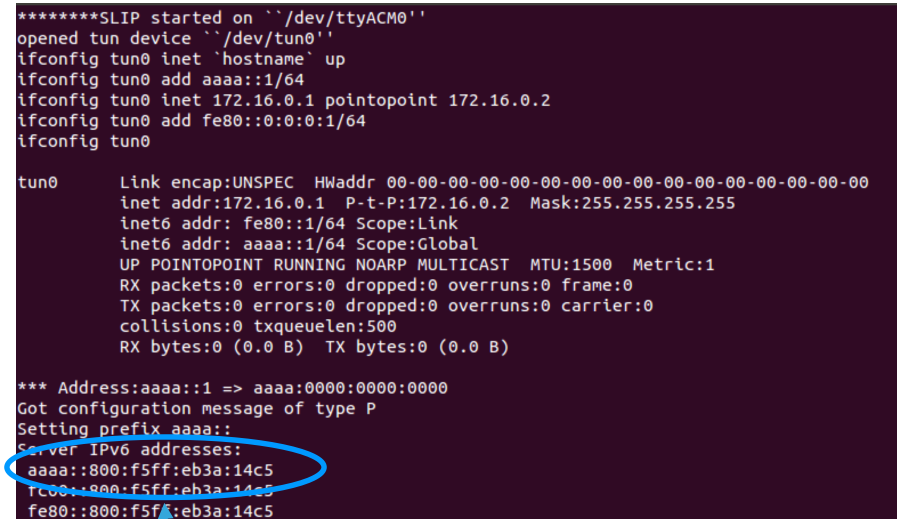
wpcapslip6 utility can then be used with the rpl-border-router example

```
cd ./tools/stm32w/wpcapslip6
./wpcapslip6 –s /dev/ttyS21 –b aaaa:: -a aaaa::1/128 [addr]
```

Where *[addr]* is the MAC address of the local net adapter



wpcapslip6 terminal window output



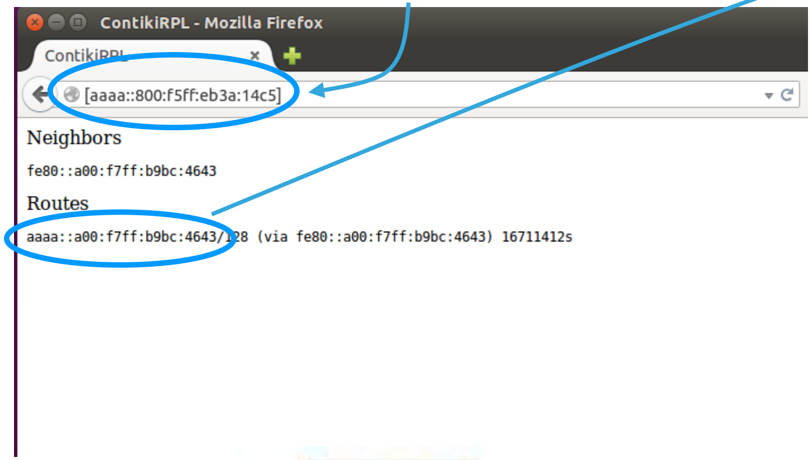
Tunslip6 terminal window output

Contiki server address (used in the next step)

Contiki on STM32 Nucleo in a few steps

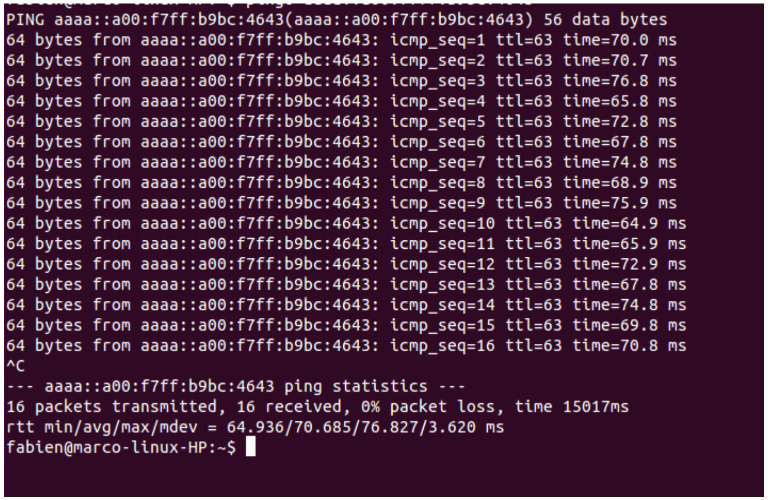
7 Open a Web browser (Firefox) to access the Contiki server providing the RPL neighbors and routes information.

Contiki server address (see previous step) between brackets, e.g. [aaaa::800:f5ff:eb3a:14c5]



ping6 aaaa::a00:f7ff:b9bc:4643

8 Ping the wireless Node to test the 6LoWPAN connectivity



9 Install the "Copper" CoAP plugin for Firefox
https://addons.mozilla.org/en-US/firefox/addon/copper-270430

Then access the CoAP Server on the wireless node by typing the URL with the node IP address

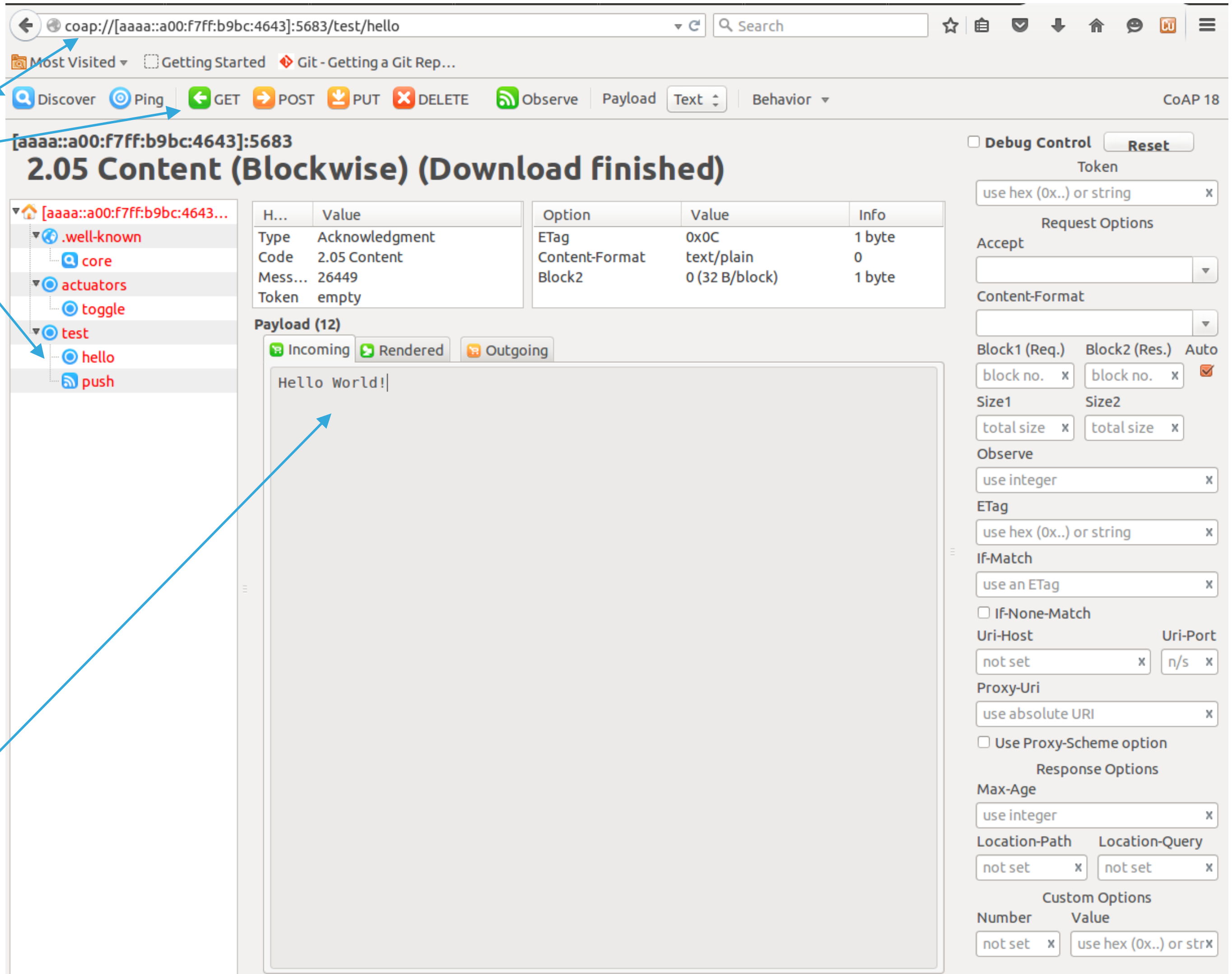
coap://[aaaa::a00:f7ff:b9bc:4643]:5683/

Contiki on STM32 Nucleo in a few steps

Example: “Hello World!” Resource Access using CoAP

(1) CoAP Resource Discovery

(2) CoAP GET Access to the “test/hello” resource



CoAP Client

6LoWPAN

GET [addr]/test /hello

ACK 2.05 Content “Hello World!”

CoAP Server

Firefox Browser window on Linux PC with “Copper” plugin (CoAP client)

Sensors Access using CoAP Demo

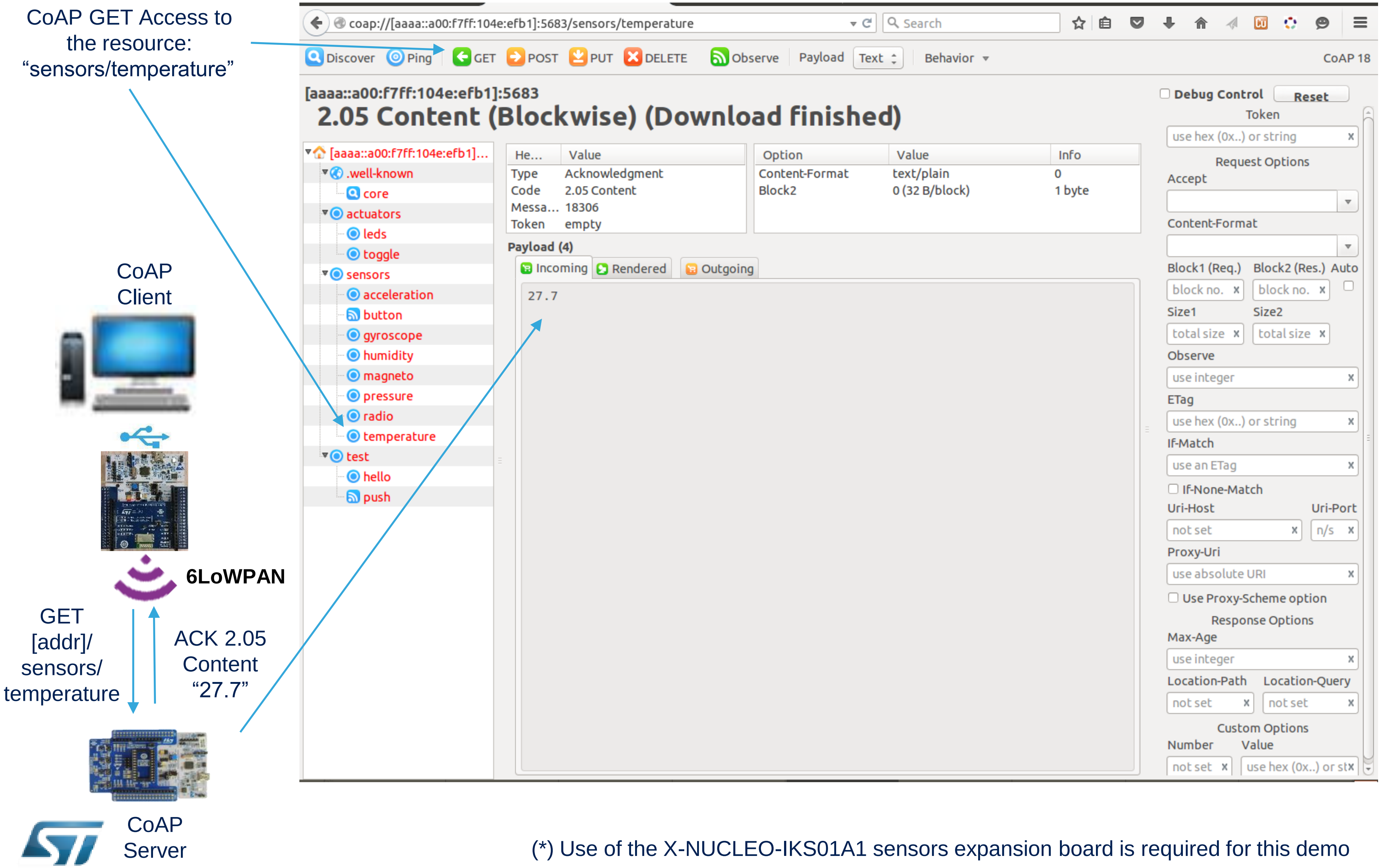
- This demo requires an X-NUCLEO-IKS01A1 expansion board for STM32 Nucleo to be mounted on a wireless node
  - The X-NUCLEO-IKS01A1 should be plugged on top of X-NUCLEO-IDS01A4/5 and NUCLEO-L152RE

- To get the demo running, a modified version of the standard Contiki “er-rest-example” application needs to be used
  - The modification is needed to update the names of the sensors used in the “er-rest-example” application and match the names of the X-NUCLEO-IKS01A1 sensors
  - The modified application is available for download from the following GitHub repository: https://github.com/STclab/stm32nucleo-spirit1-examples
  - The step-by-step setup is identical to the one described in the previous “Hello World” demo, except for “step 2” in which the modified “sensor-er-rest-example” is used

- The next slide shows the result of a CoAP GET access to the “temperature” resource hosted by the CoAP server on the wireless node

Sensors Resource Access using CoAP

Example of temperature sensor reading



(*) Use of the X-NUCLEO-IKS01A1 sensors expansion board is required for this demo

www.st.com