TALLER DE COMUNICACIÓN Y OPTIMIZACIÓN ENERGÉTICA CON DISPOSITIVOS IOT CELULAR (NB-IOT, LTE-M)

TELECO RENTA

PLAN DE
PROMOCIÓN DE LOS ESTUDIOS
DE TELECOMUNICACIÓN

Plan de Recuperación, Transformación y Resiliencia

Financiado por la Unión Europea NextGenerationEU

UNICO I+D

Workshop on Cellular IoT (NB-IoT, LTE-M) communication and energy optimization

(project TSI-063000-2021-68)

Student guide

Dr. Marisa Catalan

Marisa.catalan@i2cat.net

i2cat

Never stop designing the digital future

i2CAT.net

TELECO RENTA

PLAN DE PROMOCIÓN DE LOS ESTUDIOS DE TELECOMUNICACIÓN

Institució CERCA Centres de Recerca de Catalunya

tecnio catalonia | ACCIÓ Generalitat de Catalunya

i2cat | UNICO I+D 6G | Financiado por la Unión Europea NextGenerationEU | GOBIERNO DE ESPAÑA | Plan de Recuperación, Transformación y Resiliencia

Content

Objectives of the workshop

Brief introduction to NB-IoT and LTE-M

Hardware/Software needed for the project

Practice

- Communications
- Energy consumption and optimization

Objectives of the workshop

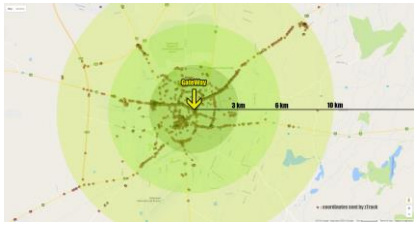
- Understand the needs of IoT devices
- Learn about NB-IoT and LTE-M technologies and their improvements in terms of energy consumption (eDRX, PSM)
- See examples of NB-IoT and LTE-M modules and fast prototyping boards
- Emulate an IoT microcontroller: Usage of AT commands to send TCP and TCP data
- Measure the consumption of a NB-IoT/LTE-M module, understand the impact the network and the device configuration in the battery lifetime and learn how to improve it.

Introduction to NB-IoT and LTE-M

General needs of the Internet of Things

- **Massive.** Connecting everything implies billions of devices wanting to communicate their data, each of the with very different needs (e.g., vehicle trackers, e-health devices, gas detectors, home appliances, …)
- **Low cost.** Deploying billons of devices can be expensive. Thus, costs must be minimized. This affects the costs of:
  - Infrastructure (deployment and maintenance)
  - Device (hardware components, battery and maintenance)
  - Connectivity costs (subscriptions, SIMs, licenses….)
- **Small dimensions and weight.** IoT devices should be smaller and lighter than the objects they monitor (e.g., keys/pet tracker)
- **Few data.** In most cases, the amount of data (for sensing or actuation) that needs to be transmitted is small. Use cases usually do not demand high bandwidth.
- **Long Range and Coverage.** To minimize infrastructure needs and to reach even underground scenarios (e.g. buried pipes).
- **Low consumption.** To enlarge the life of batteries. Reduce consumption needs to reach battery-free/energy harvesting devices (sustainability goal).

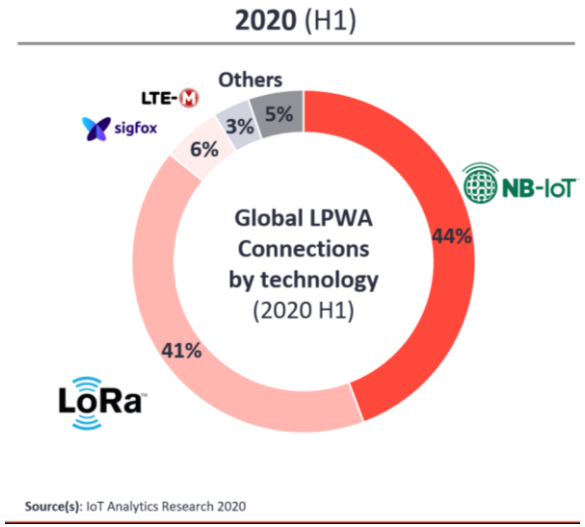
Energy consumption and battery (cost, dimensions and duration) is critical if the device needs to run for years!



Low Power WAN Networks

- They appear as a response to the connectivity needs of IoT devices
  - Long range (kilometres) /Deep Penetration
  - Low cost
  - Low consumption (> 10 years lifetime)
  - Massive connections
  - Location (wide range)
- Limitations
  - Transmission frequency
  - Capacity (bandwidth, amount of data)
  - Bidirectionality
- Technologies
  - Unlicensed band
    - Sigfox: Operator (Infrastructure + platform)
    - LoRa: Anyone can deploy the infrastructure
  - Licensed band
    - NB-IoT and LTE-M

The workshop will focus on them!



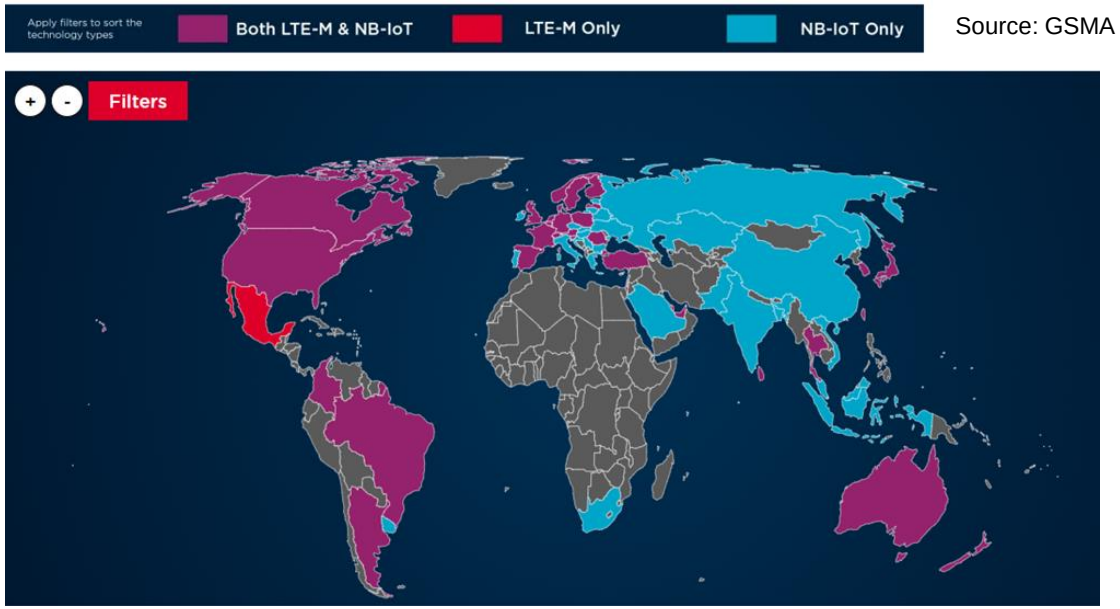
	Sigfox	LoRa	EC-GSM	Cat-1	Cat-0	eMTC	NB-IoT
Standardization	Private	Open	3GPP	3GPP	3GPP	3GPP	3GPP
Spectrum	Unlicensed	Unlicensed	Licensed	Licensed	Licensed	Licensed	Licensed
Channel BW	100Hz	7.8~500kHz	200kHz	1.4~20MHz	1.4~20MHz	1.4MHz	180KHz
System BW	100KHz	125kHz	1.4MHz	1.4~20MHz	1.4~20MHz	1.4MHz	180KHz
Peak Data Rate	UL:100bps DL:600bps	180bps~37.5kbps	DL: 74kbps UL:74kbps	DL:10Mbps UL:5Mbps	DL:2Mbps UL:1Mbps	DL:800kbps UL:1Mbps	DL:234.7kbps UL:204.8kbps
Max. number of Message per day	140(Device) 50000(BTS)	50000(BTS)	unlimited	unlimited	unlimited	unlimited	unlimited
Device Peak Tx Power	14dBm	14dBm	26dBm	23dBm	23dBm	23dBm	23dBm
MCL(Maximum Coupling Loss)	UL:156dB DL: 147dB	UL: 156dB DL: 168(SF12, BW7.8) 132(SF6, BW125)	164dB	144dB	144dB	156dB	164dB
Device Power Consumption	Low	Low-Medium	Low	Medium	Medium	Low-Medium	Low

Source:5G: a revolution or an evolution for IoT by Merouane DEBBAH, Huawei. Slideshare (Slide 24)

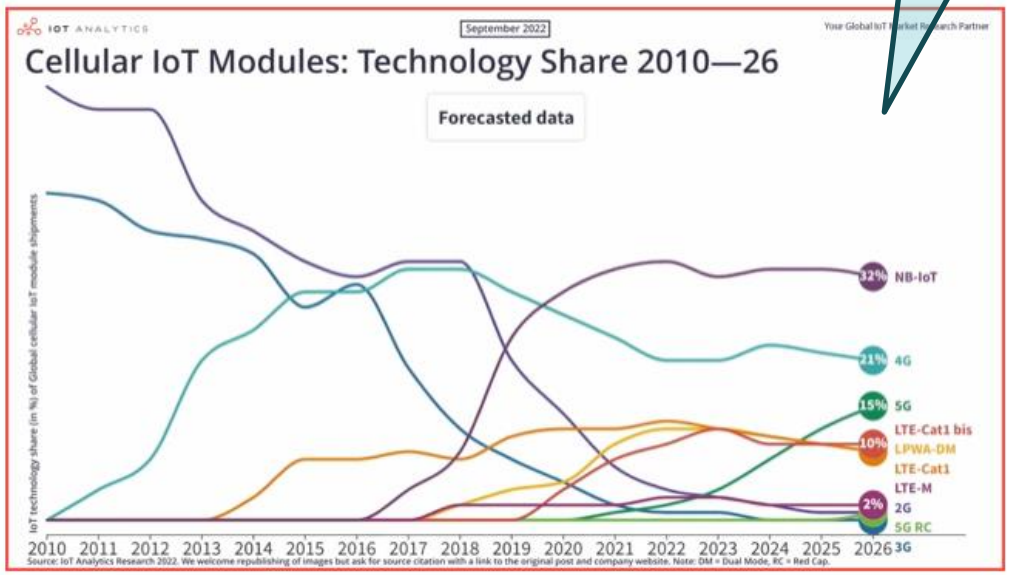
About NB-IoT and LTE-M

- Licensed technology (commercially provided by cellular operators)
  - In Spain: Vodafone, Movistar, Orange offer NB-IoT and LTE-M
- They are gaining the market of traditional cellular technologies (2G, 3G, 4G) and are part of the 5G roadmap

2G and 3G networks are already shutting down in some parts of the world



Source: GSMA



Source: IoT Analytics 2022

NB-IoT vs LTE-M

What is the difference of both technologies? Which technology to choose?

NB-IoT

- Simpler and cheaper chipsets
- No mobility
- No VoIP
- Throughput (< 250kbps)
- Good penetration
- Bidirectional
- Supports UDP and TCP
- Lower consumption

To optimize device cost and consumption

LTE-M

- Supports mobility
- VoIP
- Throughput (< 1 Mbps)
- Good penetration
- Supports UDP and TCP
- Bidirectional

When support for mobility, voice or higher throughput are necessary

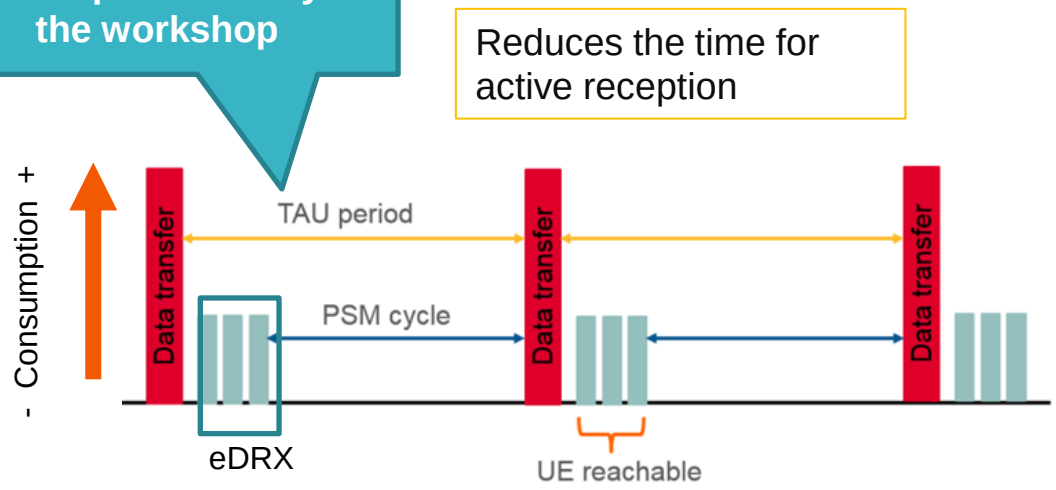
There exist dual options (NB-IoT and LTE-M) or options that are compatible with LTE/2G. This gives flexibility to IoT designs but at a higher cost (Price and consumption)

NB-IoT and LTE-M. Enhanced features

We will see these mechanisms and their impact on consumption/latency in the workshop

Mechanisms and functionalities to improve energy consumption:

- **eDRX**: Reception times are spaced (there exists a window where data can be received). Trade-off between consumption improvements (few mA) and latency for bidirectional/asynchronous communications
- **PSM**: "Idle" mode that maintains the connectivity context. Lowest consumption (µA), but it's not possible to receive data.
- **Release Assistant Indication (RAI)**: Allows to tell that no more data is expected. The device can directly go to a low-consumption mode. Otherwise, it remains active for some time (seconds) after each transmission. Supported from Release 14.



Source: NB-IoT deployment guide

Mechanisms to improve coverage

- Based on repetitions
- Trade-off between coverage/range and consumption.

Source: NB-IoT deployment guide

Mode	Level	Description
Mode A	Level 1	No Repetition for PRACH
	Level 2	Small Number of Repetition for PRACH
Mode B	Level 3	Medium Number of Repetition for PRACH
	Level 4	Large Number of Repetition for PRACH

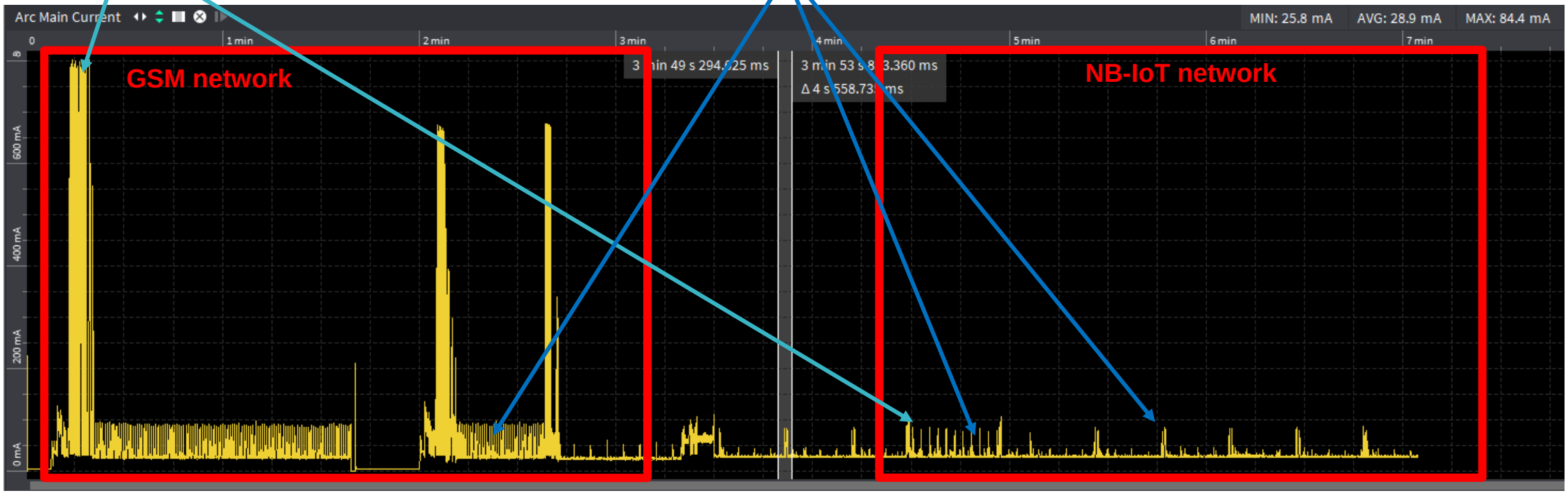
NB-IoT and LTE-M. Consumption enhancements

- **Is consumption really improved in comparison to GSM? Totally. Let's see a real example!**

Less consumption improves battery lifetime

Peak consumption reduced by ten (ca. 800mA GMS; ca. 80mA NB-IoT

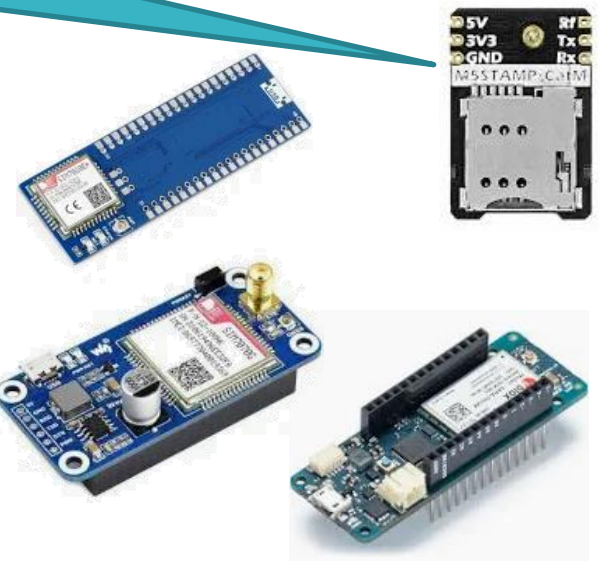
Improved consumption during the reception/idle intervals → eDRX mechanism allows to tune the reception spacing

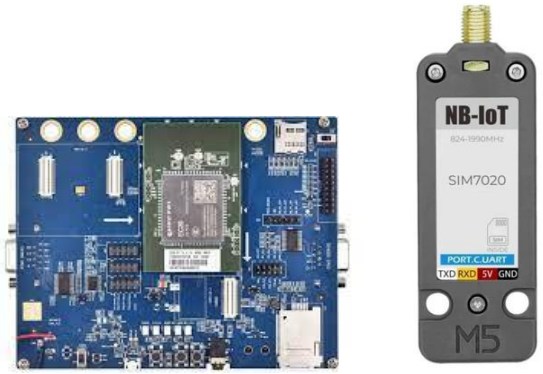


Cellular Transceivers

In the workshop we will use the SIM7080G module mounted in the M5Stamp CAT-M board

- Communication modules used to transmit data using a cellular interface
- There exist different providers that offer NB-IoT, LTE-M modules. Some examples:
  - Quectel (BC95, BG96, BG77…) ; Nordic (nrf9160); Ublox (SARA-R4, SARA-R5…); Simcom (SIM7000E, SIM7020G, SIM7070G, **SIM7080G**…)
- A processor (e.g. a computer, a Raspberry Pi or even a microcontroller) are needed to interact with them
  - The interaction is based on AT commands. Some are standard (common) and others are module-dependent
- Multiple development boards are also available for easy prototyping.
  - They include electronics to make the integration with a processor/sensors easier and to test/validate connectivity (e.g. regulators, leds, level converters, usb port…)
  - They do not focus on optimizing consumption. Consumption tends to be higher than the one targeted by the modules due to the additional components.
  - Some examples:
    - Quectel BG96, Pycom, Thingy91,MKR NB 1500; Waveshare SIM7080G/SIM7070G, M5Stack SIM7020G, **M5Stamp CAT-M**…





A real-world example

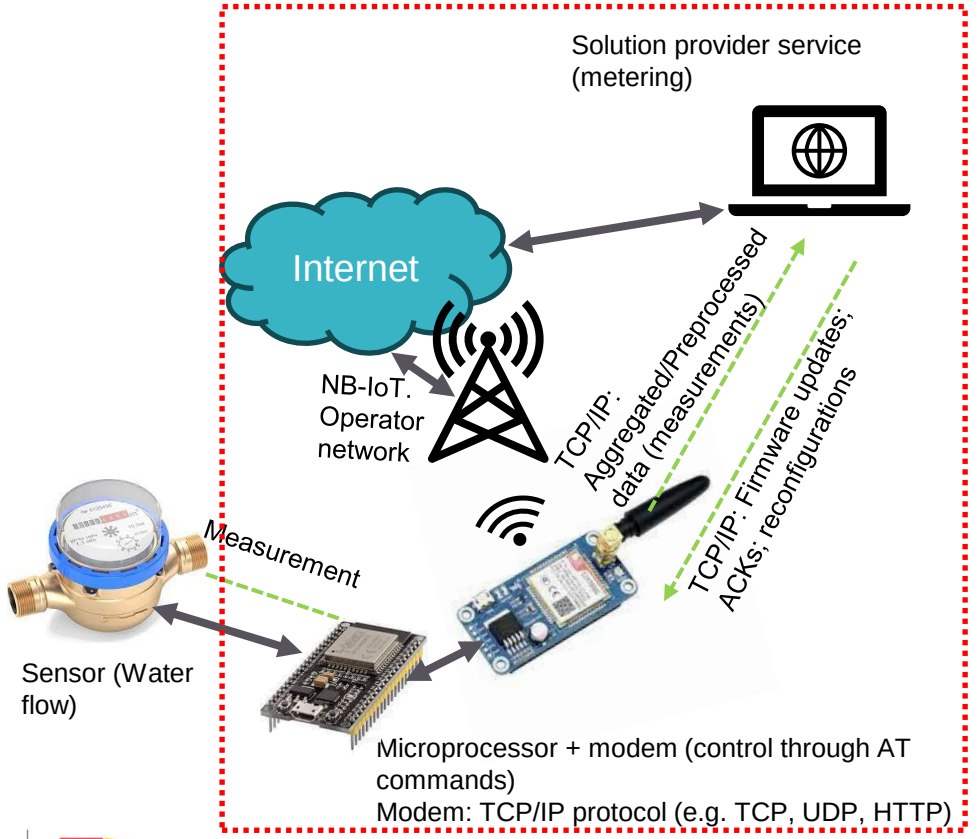
How are cellular IoT technologies enabling IoT uses cases?
Let's see an example!

- Use cases:
  - Water, gas, electricity meters
  - Water irrigation control
  - Control leakages
- Requirements:
  - Penetration: As meters might be in areas with bad coverage (e.g. basements; water chests…)
  - Low consumption: Devices usually use batteries
  - Low cost: Massive deployments
  - Bidirectionality: To support firmware updates; reconfiguration; actuation…

In the workshop we will work on the communication (We will send and receive data emulating the communication between a real sensor device and a remote service. We will use the PC instead of a microcontroller)

Smart Metering

- Solution design



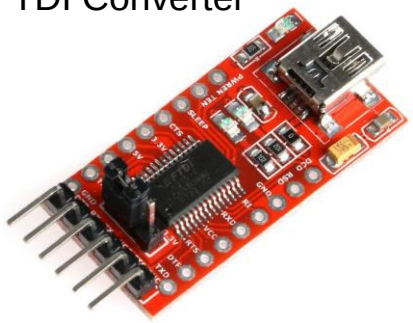
Hardware components and software

Hardware needed for the workshop

Cables

OPTION 1: Communications practice (Without experimental consumption analysis)

FTDI Converter



USB- Mini-USB

SIM7080G NB-IoT/LTE-M module (M5STAMP CAT-M) + ANTENNA + SIM

OPTION 2: Communications and consumption analysis practice

USB- Micro-USB

Consumption analyzer (OTII)

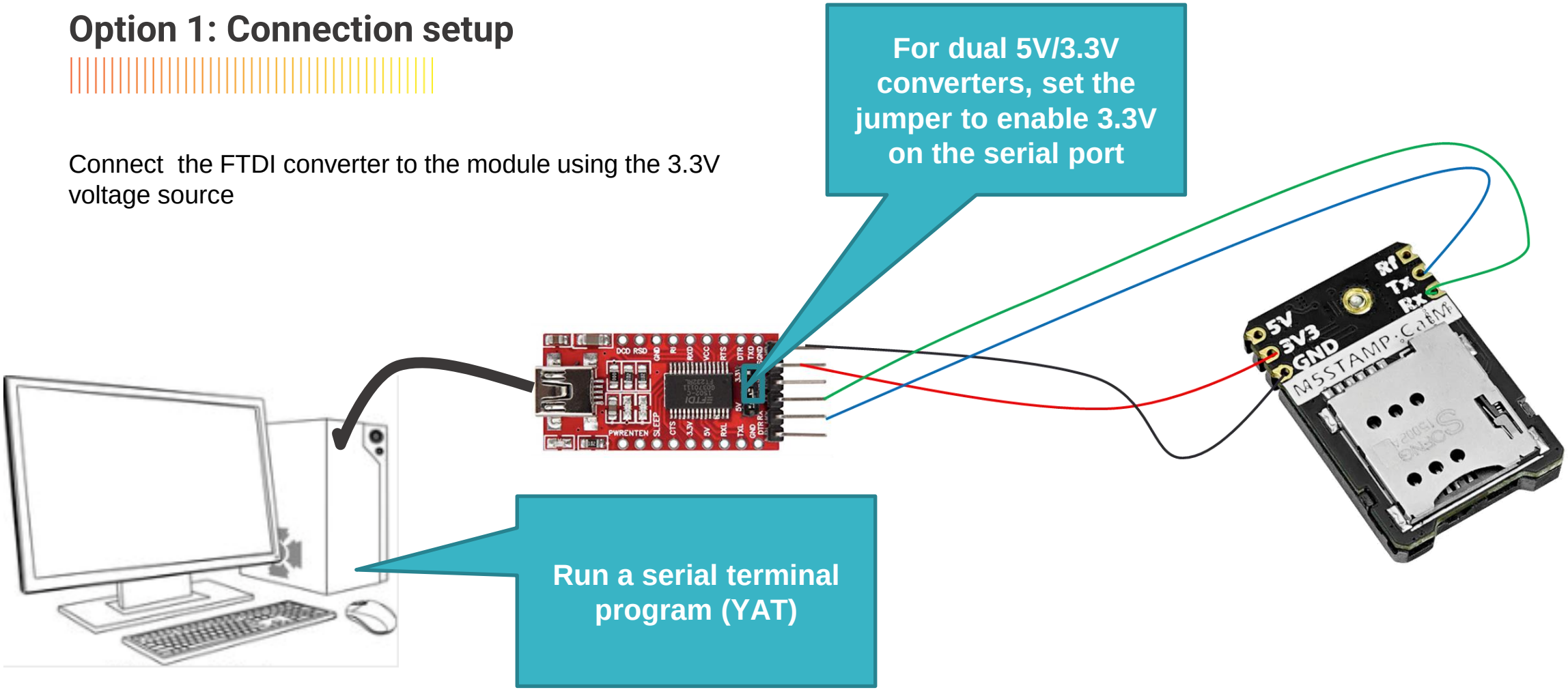
M5STAMP CAT-M module + SIM

- The development board has been chosen for the board for the following reasons:
  - 3.3 Voltage input option : Less consumption than 5V options
  - Small form factor: Similar to a minimal prototype deployment
  - Easy integration with a processor : TX/RX pins
  - Competitive price
  - Flexibility. SIM7080G module or its variants (SIM7070G (GSM compatible); SIM7020G (only NB-IoT)) are integrated in multiple easy prototyping boards. The knowledge learnt in the workshop can be easily applied to them.
- Limitations:
  - Mostly meant for standby mode (average 10mA consumption)
  - Can enter sleep mode (lowest consumption) but has no access to DTR pin to leave this mode (unpractical)
- 1nce SIM provider
  - Multi-operator
  - Meant for IoT traffic
  - Flat rate: 10€ for 10 years. 500MB of data.

Verify the correct SIM collocation.

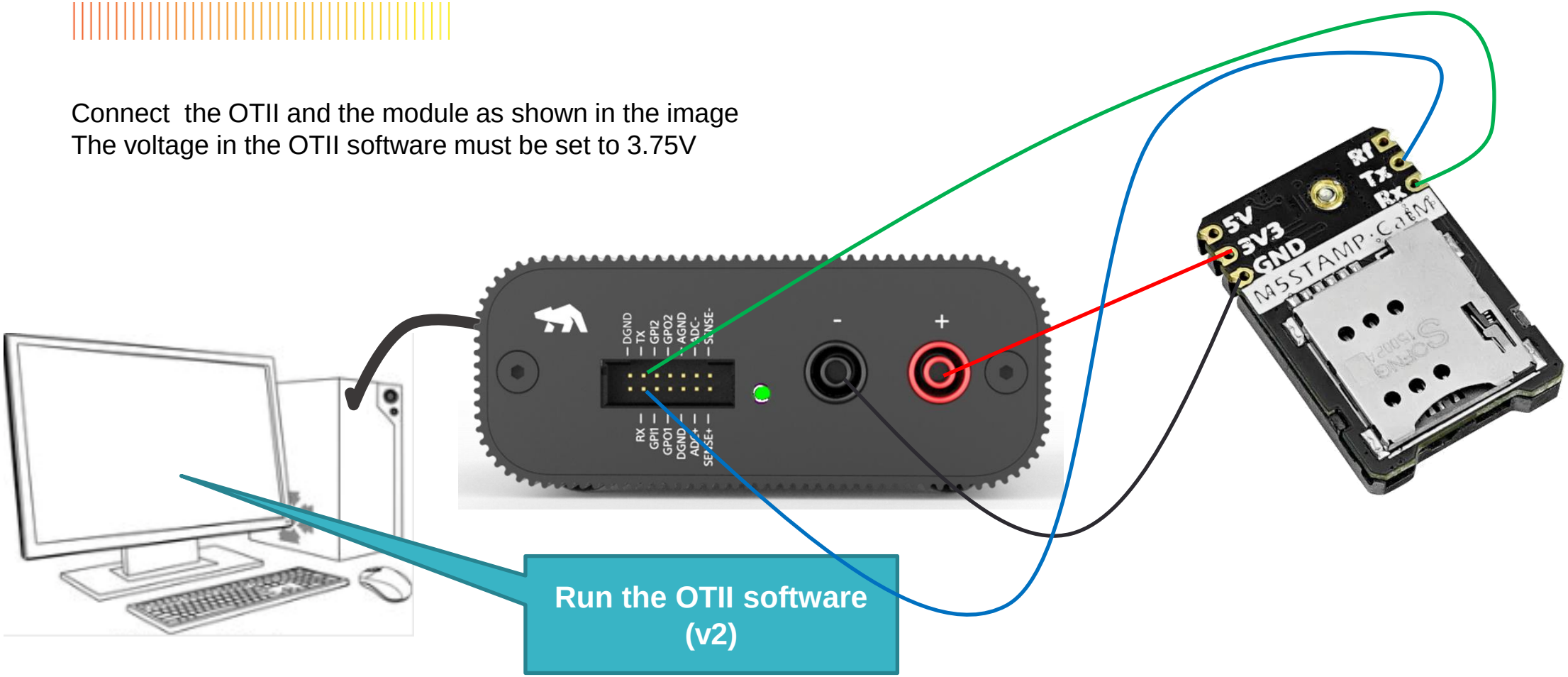
Option 1: Connection setup

Connect the FTDI converter to the module using the 3.3V voltage source



Option 2: Connection setup

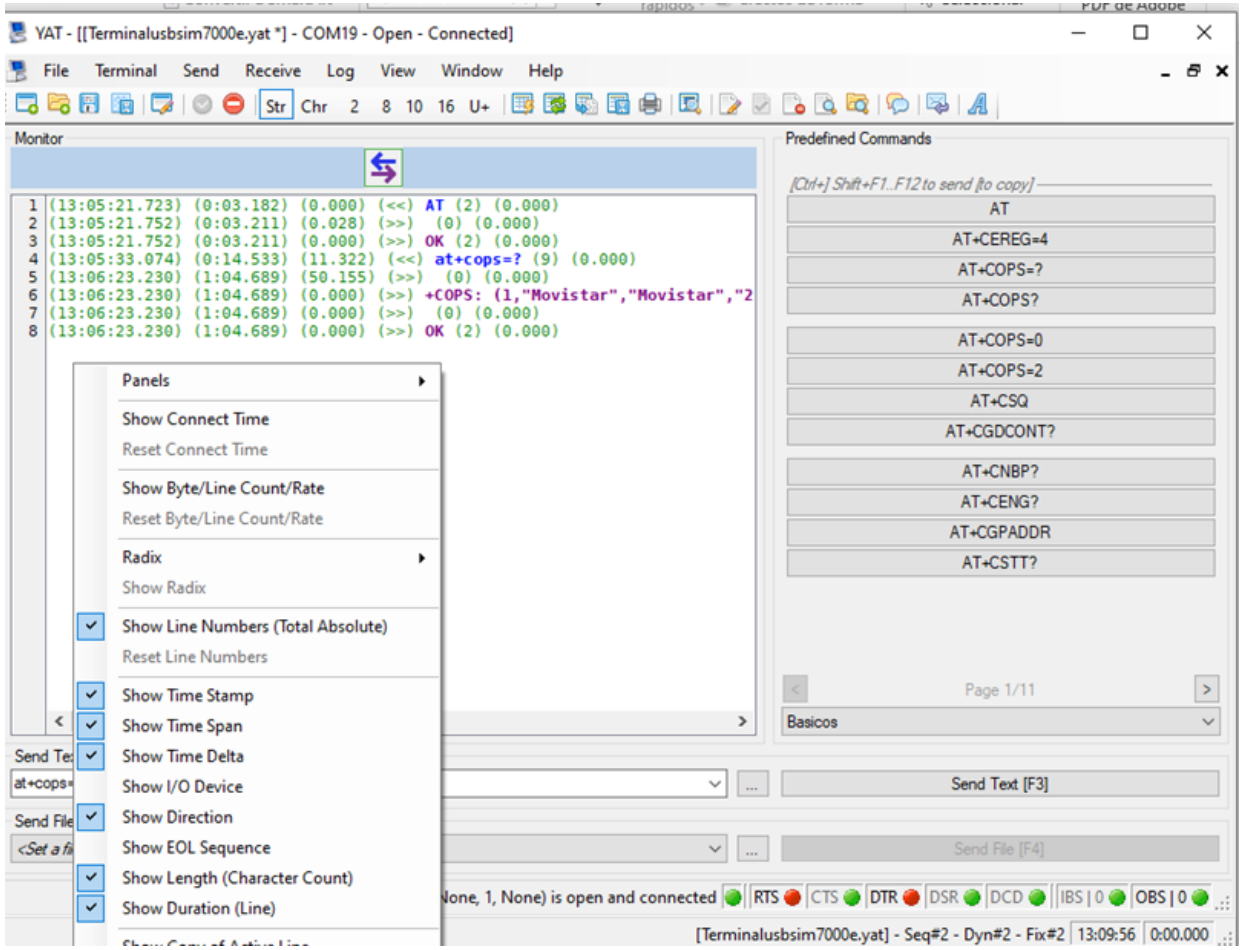
Connect the OTII and the module as shown in the image
The voltage in the OTII software must be set to 3.75V



Option 1: Yet Another Terminal software

- Freeware
- Allows to input AT commands
- Allows to save logs with a timestamp.
- Allows to define and save preconfigured values
- Windows support

Note that the FTDI converter might require to install an FTDI driver to map the serial port (https://ftdichip.com/drivers/vcp-drivers/)



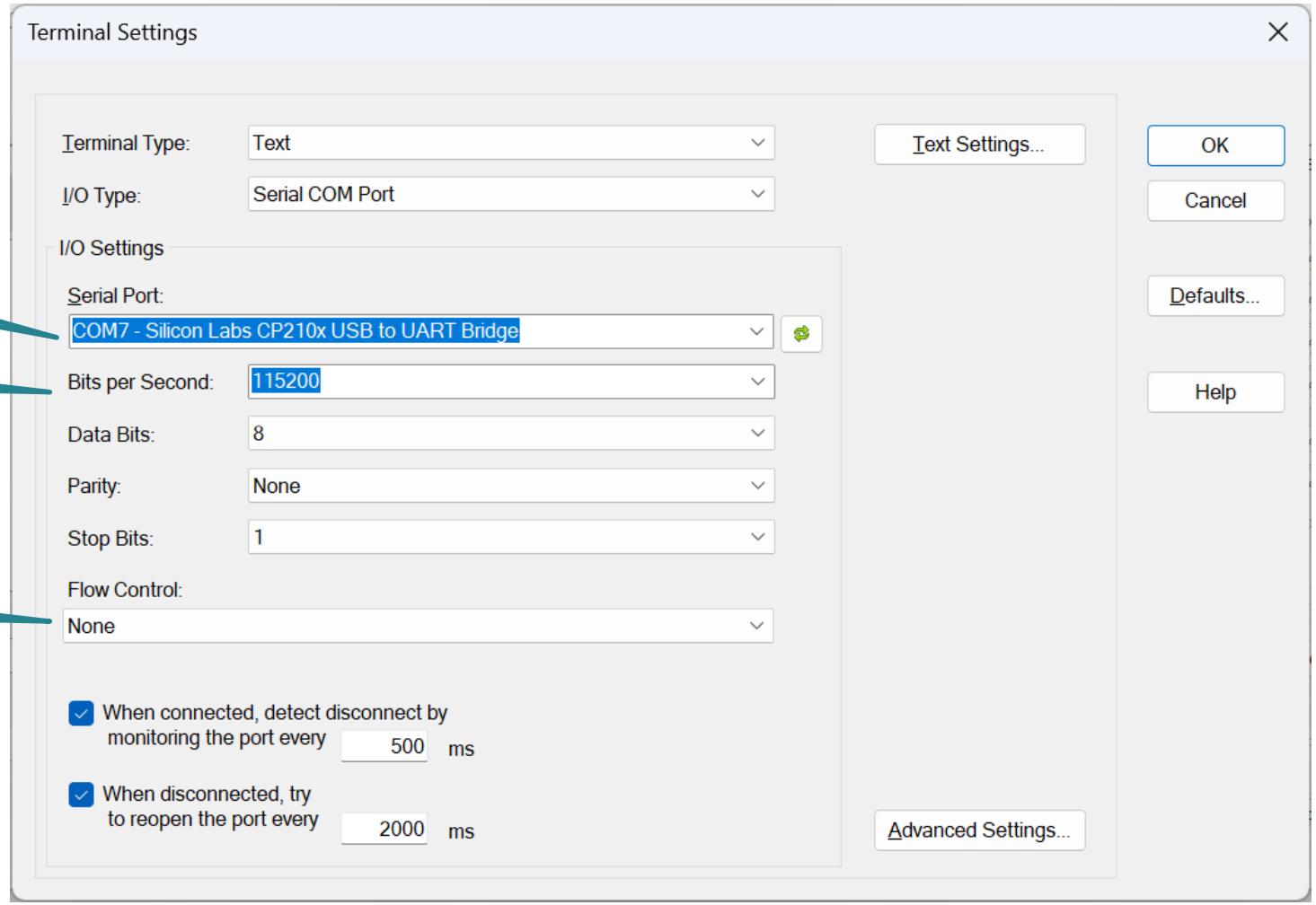
Option 1: Yet Another Terminal software

- Configuration of the serial port

Select the serial port associated to the NB-IoT/LTE-M module

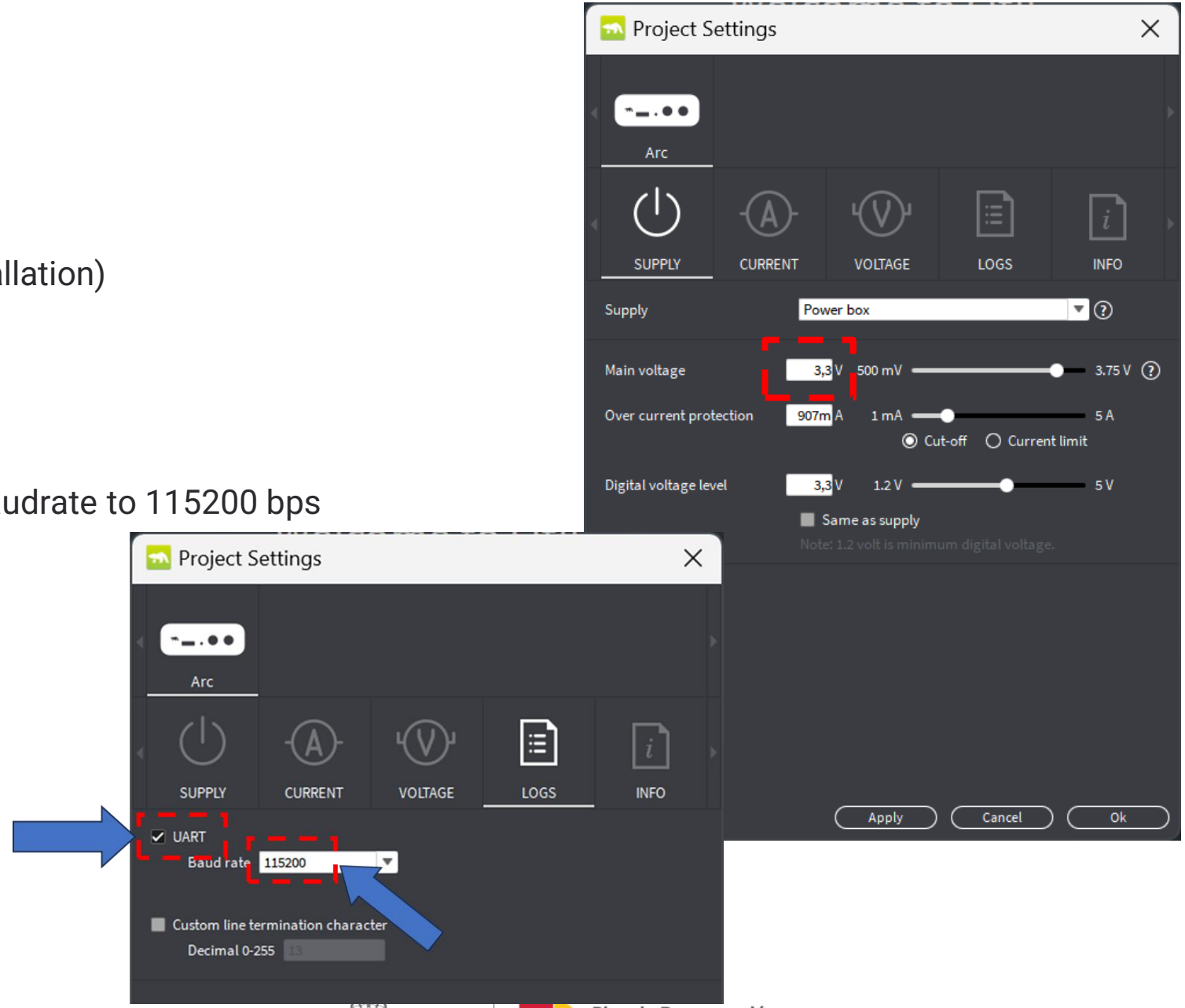
115200 bps

No flow control



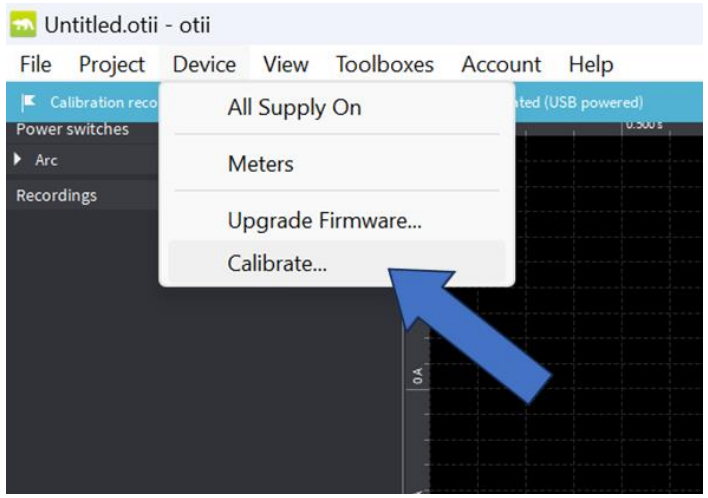
Option 2: OTII software

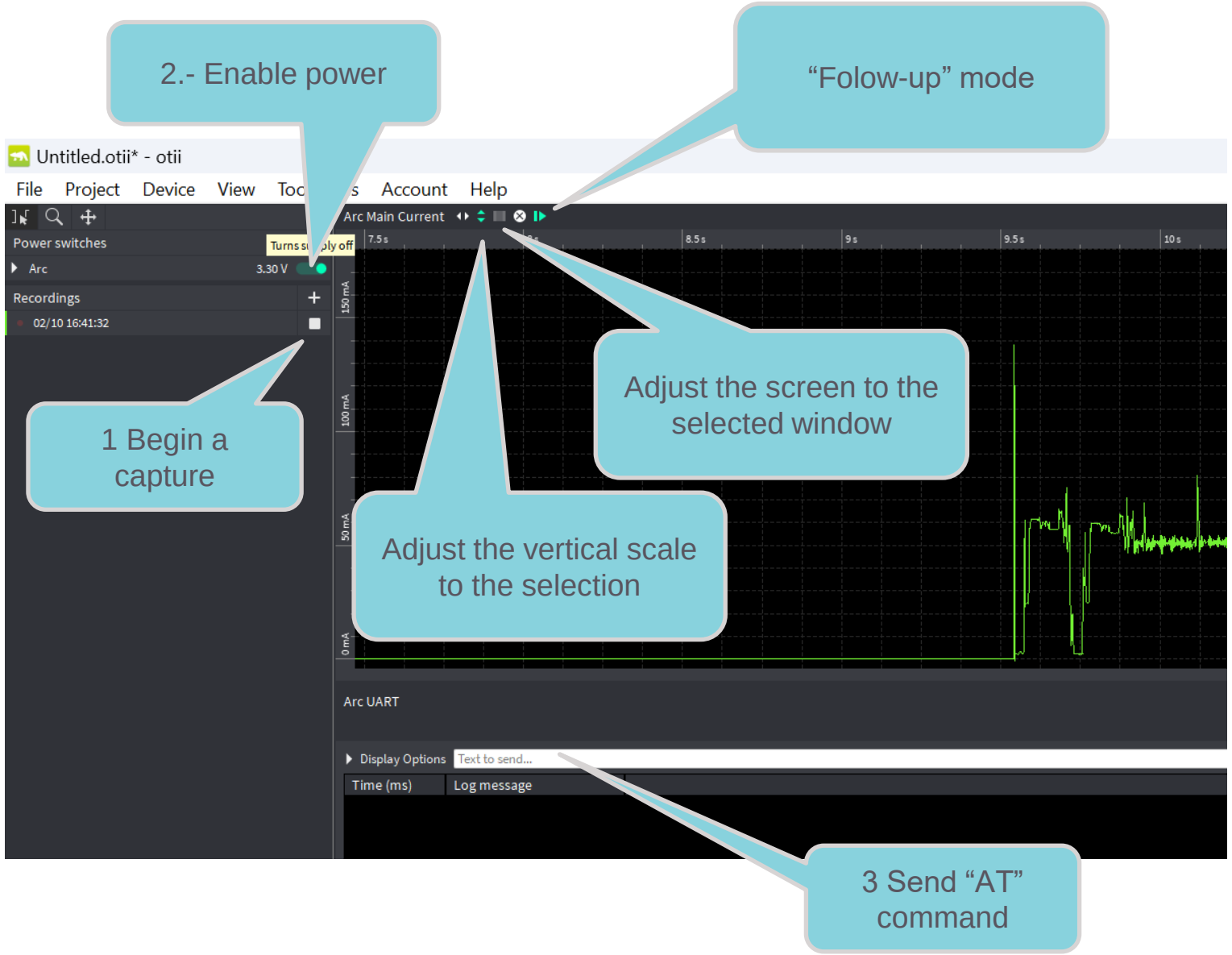
- Install Version 2.8.4 (might require installation)
- Execute the Otii program
- Select New Project
- Configure Voltage to 3.3V
- Go to Logs. Enable UART and set the baudrate to 115200 bps



Option 2: OTII software

- Software might ask for a calibration. Go to “Device” Menu
- Start a recording and enable the supply





Practice

Before we begin (read carefully)

- Be **especially careful with the connection setup** and avoid undesirable contacts between cables.
  - Take care to not mix the Ground and 3.3V (VCC) cables. This might harm the module.
  - RX and TX cables are crossed between the module and the OTII/FTDI converter.
- When there is a **reference to send an AT command**; proceed as follows:
  - Option 1: Send the command using the YAT (or equivalent) terminal
  - Option 2: Send the command from the UART window in the Otii software
- When there is a reference to observe/measure the power consumption?
  - Option 1: Check the images provided as an example. Open the example files provided and check the section referred in each slide. This will require to install the OTII software (see slide 19)
  - Option 2: Observe the values in the OTII visualization screen and check that they are similar to the ones provided as an example (in case of doubt ask the instructor for advise/explanation)
- If you get stuck or have some doubts, **ask the instructor for advice.**

AT commands

- The communication between the processor and the modem is based on AT commands.
- This is a common process to communicate some devices (e.g. modems or transceptors)
- AT(ATtention)
  - Commands send to the modem always begin with AT
  - The modem should reply with OK (this indicates that the command is interpreted)
  - Some AT commands for LTE-M and NB-IoT networks are standard
  - Others depend on the provider of the modem/transceptor
  - For the Practice, download and check the SIM7080G AT commands manual at: https://www.waveshare.com/w/upload/3/39/SIM7080_Series_AT_Command_Manual_V1.02.pdf
  - We will refer to the manual during the practice
  - Specific commands

    AT+Command (e.g; AT+COPS; AT+CEREG…)

    AT+Command ? Ask for the syntaxes of the command

    AT+Command Ask for the execution of the command

    AT+Command =? Ask for detailed information

Step 1: Verify the communication with the module

1. **Verify the communication with the module**

- Send the command: **AT**
- The module should return: **OK**
- You might need to repeat this process several times before the module responds
- If there is no response
  - Check that the module is powered
  - Check the connection setup (TX/RX cables)

2. Verify the communication with the SIM (check AT commands guide, pag. 41)

- Send the command: **AT+CIMI**
  This command returns the SIM identifier (requires interaction with the SIM card)
- The module should return a number. Example: 901405103839430
- If the response is: **ERROR**
  - Check that the SIM is correctly collocated

In a real NB-IoT device implementation, the microcontroller might verify the communication with the modem to identify problems or assure that the communication can take place.

The aim of Step 1 is to:
1. Assure that the hardware setup has been performed correctly
2. Assure that there are no issues with the SIM card or the SIM connector.

These commands are standard.

Step 2: Some configuration/debug commands

1. **Enable local ECHO (check AT commands guide, pag. 20)**

- Send the command: **ATE1**
- The module should return: **OK**

2. Enable a textual explanation of errors (check AT commands guide, pag. 44)

- Send the command: **AT+CMEE**

This command provides verbose information about some errors in case they occur; by default, the terminal returns an error code, which is difficult to interpret.

- The module should return: **OK**

The aim of Step 2 is to set some configurations that are useful to understand the practice or to debug the communication.

In a real implementation some of these commands might be used occasionally to debug the program or to get information that can be useful to adapt the behavior of the program.

These commands are standard.

Step 2: Some configuration/debug commands

3. Enable CEREG messages (check AT commands guide, pag. 139 and ANNEX slides)

- Description: Notify when the module **changes its registration status**. The configuration "4" gets the maximum information from the device (registration details + **PSM configuration**). CEREG provides **also information useful to locate a module** (it return the id of the station to which it is connected). Once configured the module sends automatic notifications when status changes occur.
- Send the command: **AT+CEREG=4**
- The module should return: **OK**

4. Enable RCC STATE messages (check AT commands guide, pag. 108)

- Description: Notify when the **module enters/leaves the active state**. The active state implies a higher consumption. The time in active state depends on the network configuration. Once configured the module sends automatic notifications when status changes occur.
- Send the command: **AT+CRRCSTATE=1**
- The module should return: **OK**

The aim of Step 2 is to set some configurations that are useful to understand the practice or to debug the communication.

In a real implementation, the device can be programmed to use the automatic notifications from 3. and 4. to get information from the network (e.g. time in active mode; disconnections) and adapt/react accordingly.

These commands are standard.

Step 3: Registration setup

1. Disconnect from any network (check AT commands guide, pag. 44 and ANNEX slides)

- Description: To begin with a clean registration. We will deregister the module in case it has already registered. “2” is the “deregister” order
- Send the command: **AT+COPS=2**
- The module should return: **OK**
- Observe the CEREG message. A “0” indicates that the status of the module is “not registered”. Example: - +CEREG: 0,,,,,,"00000000","01100000"

2. Enable the APN for the 1NCE network (check AT commands guide, pag. 133)

- Description: It is necessary to set the correct access point for the connection. In this case, “"iot.1nce.net"
- Send the command: **AT+CGDCONT=1,"IP","iot.1nce.net“**
- The module should return: **OK**

The aim of Step 3 is to perform the setup of the module before the registration.

In a real implementation, the device must be initialized with the necessary parameters to be able to register and connect to the network.

These commands are standard.

The aim of the slide is to check the available networks and understand the response (see ANNEX slides for details about the command)

Step 3: Registration setup

Consumption example available at: example_workshopCIoT/Step3 Check networks; 19.738s

Duration: 3:06min

3. Check the available networks

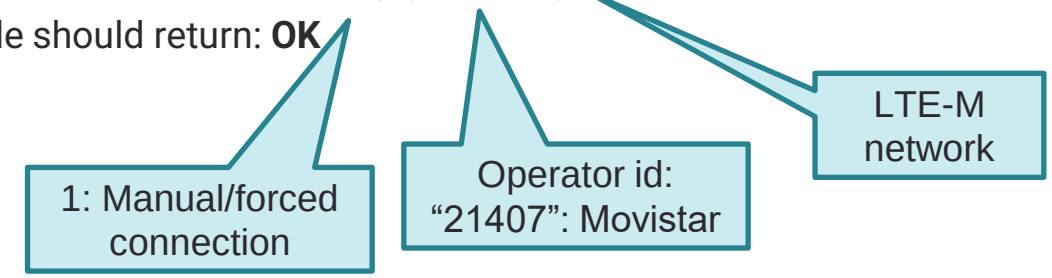
- Description: Check the **available networks**. Remark. The command might take several minutes (e.g 3 minutes). Wait until you get a response.
- Send the command: **AT+COPS=? (check AT commands guide, pag. 44 and ANNEX slides)**
- The module should return the networks. Example: +COPS: (0,"vodafone ES","voda ES","21401",9),(0,"vodafone ES","voda ES","21401",7),(0,"Orange SP","ESPRT","21403",7),(0,"Movistar","Movistar","21407",7),(0,"Movistar","Movistar","21407",9),(0,"Orange SP","ESPRT","21403",9),,(0,1,2,3,4),(0,1,2)
- Activity 1: How much time does it took to get the response? You can visualize it at the logs of the serial connection (OTII or YAT).
- Activity 2: Observe the consumption of the module during the full process.

Step 3: Registration setup

Consumption example available at:
example_workshopCIoT/Step3 Register LTE-M network; 12.840s

4. Connect to a network

- Description: Manually connect to one of the available networks. To follow the example, connect to the LTE-M Movistar network
- Connect to the LTE-M Movistar network (**check AT commands guide, pag. 44 and ANNEX slides**)
  - Send the command: **AT+COPS=1,2,"21407",7**
  - The module should return: **OK**



- Note: If the Movistar LTE-M network is not available or for further testing, follow the steps in Slide 42 to connect to another network

The aim of this and the next slide is to understand the command to connect to the network and perform a registration.

In a real implementation, the device must be programmed to register to the network before connecting. The chosen network could be predefined. Otherwise, the module might check the available networks or might try to connect automatically to any network.

Step 3: Registration setup

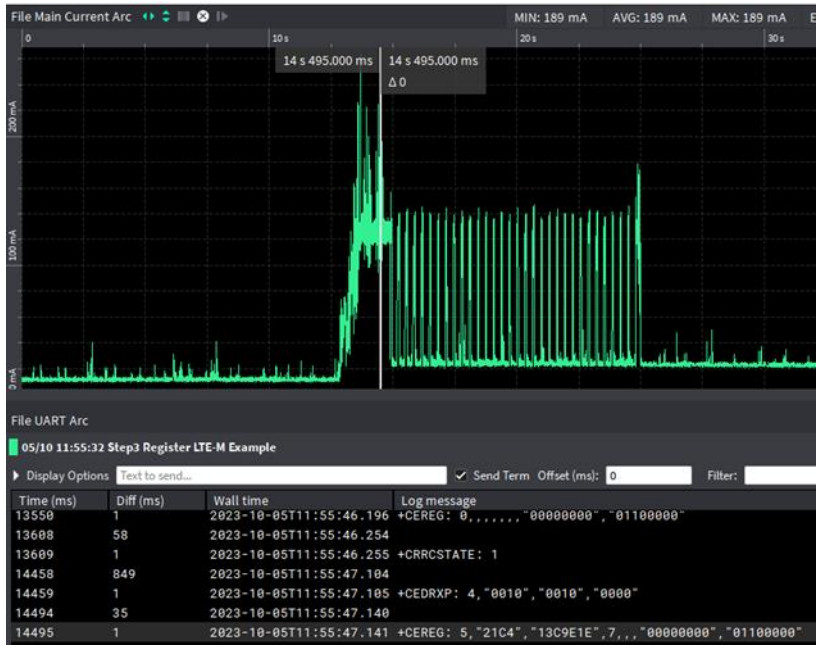
"21C4" and "13C9E1E" refer to the station/cellID to which the device is connected. This info could be used by geolocation apis (e.g. Google) to track the device.

5. Connect to a LTE-M network

- Activity 1: Observe the CEREG message. A "5" indicates that the status of the module is "registered". Example: +CEREG: 5,"21C4","13C9E14",7,,,"00000000","01100000"

Related to PSM configuration (lowest consumption). "00000000" means "not supported"

CEREG indicates that the device is registered. Register duration: 1.655s



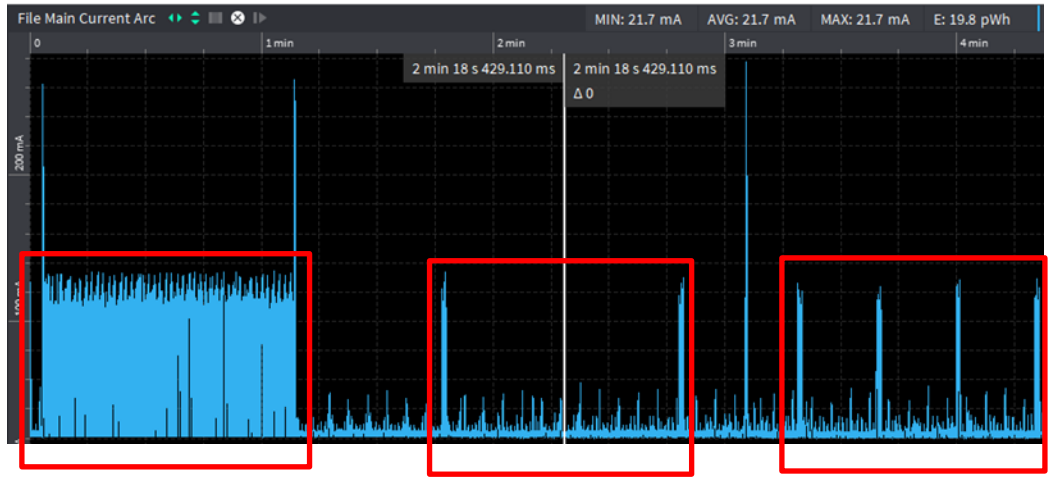
- Activity 2: Observe the consumption of the process (the peak and average values)
- Activity 3: Observe the RCC messages and how they are related to the consumption of the device

Though the device is already registered, each network defines a minimum active time (RRCSTATE=1; higher consumption). This time determines when the device can go to idle/standby state (RRCSTATE=0) to reduce its energy consumption

Consumption example available at:
example_workshopCIoT/Step3 Register LTE-M network; 12.840s

Step 4: Understand eDRX improvements

Consumption example available at:
example_workshopCIoT/Step4 EDRX impact



No EDRX | EDRX = 61.44 | EDRX = 21.48

1. Disable eDRX mode

- Description: The module will be continuously in receiver state
- Send the command: **AT+CEDRXS=0** (**check AT commands guide, pag. 125)**
- The module should return: **OK**
- Wait for at least 30 seconds

Activity 1: Observe the consumption pattern

2. Enable different eDRX modes

- Description: Set the eDRX mechanisms for two different cycles.
- Send the command: **AT+CEDRXS=2,4,"0100"** (Cycle: 61.44s)
- Wait for about 150 seconds
- Send the command: **AT+CEDRXS=2,4,"0010"** (Cycle: 21.48s)

Activity 2: Observe the consumption pattern and duration of the eDRX cycles in each case. Compare the average consumption for a full cycle.

The aim of Step 4 is to understand the eDRX mode and the impact of its configuration on consumption.

In a real implementation, the device would configure the best eDRX configuration based on the needs of the use case/application and the network capabilities.

These commands are standard.

More about the eDRX mechanism

(check AT commands guide, pag. 125)

- The eDRX mechanism can be configured from 5.12s to almost 3h (not all the values are valid for NB-IoT and LTE-M)
- The value of the cycle is expressed in binary format
- The format of the command is **AT+CEDRXS=<n>, <Act Type> <Req eDRX value>**
- Enabling **eDRX** mechanism is **relevant to optimize the consumption** of the module and reach large battery durations.
- **Note: Not all the networks might support the eDRX mechanisms or accept eDRX configuration**
  - The message +CEDRXP:2 indicates that the network does not support eDRX
  - Networks can support eDRX but they might not accept the configuration (you will see no effect after sending a different eDRX setup).

```
0-      5.12,
1-     10.24,
2-     20.48,
3-     40.96,
4-     61.44,
5-     81.92,
6-    102.40,
7-    122.88,
8-    143.36,
9-    163.84,
10-   327.68,
11-   655.36,
12- 1310.72,
13- 2621.44,
14- 5242.88,
15-10485.76.
(seconds)
```

Parameters

<n>

0 Disable the use of eDRX

1 Enable the use of eDRX

2 Enable the use of eDRX and auto report

3 Disable the use of eDRX(Reserved)

<AcT-type>

4 CAT-M

5 NB-IoT

<Requested_eDRX_value> Requested eDRX value. 4 bit format.

"0000"-"1111"

Steps 5 and 6: Connect and transmit data

The aim of the Steps 5 and 6 (this and the following slides) is to understands the commands necessary to establish a connection using UDP and TCP protocols and be able to transmit and receive data.

In a real implementation, the device would be programmed to establish a connection and send the information using the most convenient protocol (e.g. TCP, UDP, HTTP or even MQTT or CoAP) depending on the needs of the application/use case or the requirements of the remote service.

In this practice, the transmission and the reception will be synchronous (ECHO server). In a real case, the bidirectionality needs will depend on the scenario; e.g. in some case a response might not be necessary or messages might be received in an asynchronous manner. The device should be programmed accordingly to the needs of use case.

The commands to connect and transmit data are not standard (they might differ even for different SIMCOM modules).

Step 5: Connect and transmit data. UDP connection to an Echo server

Consumption example available at: example_workshopCIoT/Step5 UDP Echo Example; 2.952s

Aim: Send UDP data to an Echo server that responds immediately. Receive the response

1. Activate the context (check AT commands guide, pag. 143)

- Description: The context must be activated before a connection
- Send the command: **AT+CNACT=0,1**

Instructors might provide a different IP and port for the ECHO server that the one provided in the example

2. Open the connection to a UDP port (check AT commands guide, pag. 178)

- Description: The IP address and the port must be specified. Will be indicated by the instructor
- Send : **AT+CAOPEN=0,0,"UDP","ipserver",portserver.** Example: AT+CAOPEN=0,0,"UDP"," 52.43.121.77 ",10001

3. Send a message (check AT commands guide, pag. 180)

- Description: To send a message, first the size is indicated and the time to wait for the message input. Then, the message is sent.
- Send the command: **AT+CASEND=0,10,10000** ; this indicates that 10 bytes will be sent and that the module will wait for 10 seconds for them.
- Send rapidly the message (10 characters + enter) (without AT command)
- The module will respond: **OK** . Otherwise, the 10 seconds timer might have expired. Repeat the process faster.

Step 5: Connect and transmit data. UDP connection to an Echo server

Consumption example available at:
example_workshopCIoT/Step5 UDP Echo Example; 2.952s

4. Receive data (check AT commands guide, pag. 179)

- Description: One the response from the server is received the module will notify it with a “Data Indication”
- Observe the messages
- The module should have returned: **+CADATAIND: 0**

5. Retrieve the data (check AT commands guide, pag. 181)

- Description: Retrieve a number of bytes from the module
- Send the command: **AT+CARECV=0, 10.** This will retrieve 10 bytes from the module
- The module will return your message.

6. Close the connection and the context

- Send the command: **AT+CACLOSE=0** (Returns OK)
- Send the command: **AT+CNACT=0,0** (Returns OK)

(check AT commands guide, pag. 183)

Activity 1: Succesfully send and receive data with UDP
Activity 2: Observe the consumption of all the steps

Do all the steps show consumption peaks? Response: No. As it’s UDP it does not require to actively establish/close a connection with the server side

Financiado por la Unión Europea
NextGenerationEU

Step 6: Connect and transmit data. TCP connection to an Echo server and impact on delay

Consumption example available at:
example_workshopCIoT/Step6 TCP Echo Delay 30s; 3.371s

Aim 1: Send TCP data to an Echo server.

Aim 2: Understand the impact of a delayed response (asymmetric response). Check the consumption example.

1. **Repeat the instructions of Step 5 but for TCP** (the only command that changes is CAOPEN)

- Send the command: **AT+CNACT=0, 0**
- Send the command: **AT+CAOPEN=0,0,"TCP","52.43.121.77",9001**
- Send the command: **AT+CASEND=0,10,10000**
- Send the message immediately
- Activity 1: Wait until the module returns **+CADATAIND: 0** .How much time does it take? More than 30 seconds? Compare this to the provided consumption example for a delayed response.
- Send the command: **AT+CARECV=0, 10**
- Send the command: **AT+CACLOSE=0** (Returns OK)
- Send the command: **AT+CNACT=0,0** (Returns OK)

Instructors might provide a different IP and port of the ECHO server

For delayed responses, the module does not receive the message from the network until the next eDRX cycle.
If the cycle is very long any incomming message will be notably delayed
In some cases data might be lost (depending on the setup of the network)

Congratulations!

- You have finished the workshop. If you have time, you can still experiment trying to connect to a different network and observing the differences/similarities (See slide 42).
- To sum up
  - You have learnt how to send UDP and TCP data with NB-IoT and LTE-M technologies which are key technologies for connecting IoT devices.
  - You have experienced the relevance of eDRX mechanisms and how they are an improvement in terms of energy consumption
  - Though the configuration of the network cannot be modified/chosen, **the tuning of the end devices (e.g. eDRX) can help to optimize the communication**
  - **There is always a trade-off**. Spacing receiving windows could delay the reception of data (unless well planned or synchronized).
- But this was just a first touch, there is more to learn and experiment!
  - PSM, sleep mechanisms and RAI indications can largely improve consumption.
  - Devices can be located based on their registration information
  - You will get some hints about these topics in the references

To experiment more with NB-IoT and LTE-M

Some references

- Documentation M5STAMP CAT-M: https://docs.m5stack.com/en/stamp/stamp_catm
- SIM7080G AT Commands Manual: https://www.waveshare.com/w/upload/3/39/SIM7080_Series_AT_Command_Manual_V1.02.pdf
- SIM7080G Low-Power mode application note. https://www.waveshare.com/w/upload/a/aa/SIM7080_Series_Low_Power_Mode_Application_Note_V1.01.pdf
- 201906-GSMA-NB-IoT-Deployment-Guide-v3

https://www.gsma.com/iot/wp-content/uploads/2019/07/201906-GSMA-NB-IoT-Deployment-Guide-v3.pdf

- 201906-GSMA-LTE-M-Deployment-Guide-v3

https://www.gsma.com/iot/wp-content/uploads/2019/08/201906-GSMA-LTE-M-Deployment-Guide-v3.pdf

- Improving Energy Efficiency for Mobile IoT. March 2022. Whitepaper GSMA https://www.gsma.com/iot/wp-content/uploads/2022/02/2022.03-GSMA-Energy-Efficiency-for-Mobile-IoT-1.pdf
- YAT terminal emulator. https://sourceforge.net/projects/y-a-terminal/
- API Radiocells.org. Geolocation. https://radiocells.org/geolocation
- Google API localization. https://developers.google.com/maps/documentation/geolocation/overview

AT+COPS. How to register to any NB-IoT and LTE-M network

1. Connect to a network

- Description: Manually connect to one of the available networks. Op-nr is the operator number (“21401” for Vodafone, “21403” for Orange, “21407” for Movistar). Tech-nr is the technology number (9:NB-IoT, 7:LTE-M).
- To connect to a NB-IoT network:
  - The network type must be forced to NB-IoT only (otherwise it might connect to LTE-M)
  - Send the command: **AT+CMNB=2** (only NB-IoT)**.** The module should return: **OK**
  - Send the command: **AT+COPS=1,2,*op-nr*,*9*** *(Example: AT+COPS=1,2,”21407”,9 would connect to Movistar and NB-IoT)*
  - The module should return: **OK**
- To connect to an LTE-M network:
  - Send the command: **AT+CMNB=1** (only LTE-M) **or AT+CMNB=3** (dual mode)**.** The module should return: **OK**
  - Send the command: **AT+COPS=1,2,*op-nr*,*7*** *(Example: AT+COPS=1,2,”21407”,7 would connect to Movistar and LTE-M)*
  - The module should return: **OK**

LPWAN Laboratory: TheThinx 5GBarcelona

Initiative of Telefonica, i2CAT, MWC and Fira de Barcelona sited in Barcelona

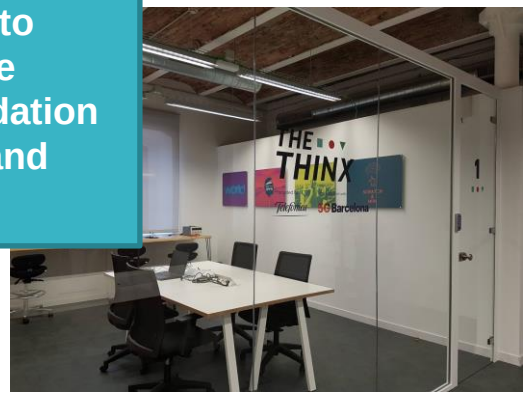
Open laboratory focused on:

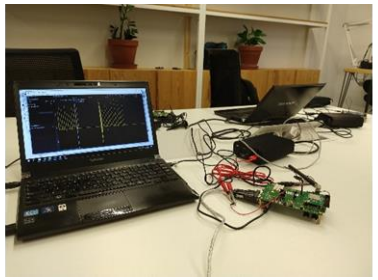
- Accelerate the development of LPWAN IoT prototypes and solutions.
- Test cellular connectivity (currently NB-IoT and LTE-M)
- Apply mechanisms to improve energy consumption
- Provide a controlled environment to test enhanced functionalities as in a real network
- Make the migration from products based on 2G, 2.5G or 3G technologies easier

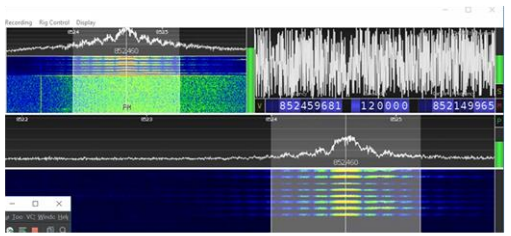
Equipment to :

- Test end-to-end connectivity
- Validate HW/FW prototypes
- Analyze consumption
- Test coverage conditions

Controlled laboratories help to complement measurements/tests in real networks to accelerate the development/validation of prototypes and solutions







Financiado por la Unión Europea NextGenerationEU

GOBIERNO DE ESPAÑA

Plan de Recuperación, Transformación y Resiliencia

Annex

AT+COPS (OPerator Selection). Description

- This command is standard for any NB-IoT/LTE-M device.
- Parameters in [ ] are optional
- Provides information about operators present. Returns quadruplets

  AT+COPS=?

  +COPS: (list of supported<**stat**>,long alphanumeric<**oper**>,short alphanumeric<**oper**>,numeric <**oper**>,<**netact**>)s[,,(list of supported <**mode**>s),(list of supported <**format**>s)]
- Returns the current network

  AT+COPS?

  +COPS:**<mode>**,[,**<format>**,**<oper>**,**<nectact>**
- Force the connection to a network

  AT+COPS=<**mode**>,[<**format**>[,<**oper**>]]
- Disconnect from the network

  AT+COPS=2
- Automatically connect to a network

  AT+COPS=0
- Manually connect to a network

  AT+COPS=**1**,[**<format>**[,**<oper>**]]

<stat>	0	Unknown
	1	Operator available
	2	Operator current
	3	Operator forbidden
<oper>	Refer to [27.007] operator in format as per **<format>**	
<mode>	0	Automatic mode; **<oper>** field is ignored
	1	Manual (**<oper>** field shall be present, and **<AcT>** optionally)
	2	manual deregister from network
	3	set only **<format>** (for read Command +COPS?) - not shown in Read Command response
	4	Manual/automatic (**<oper>** field shall be present); if manual selection fails, automatic mode (**<mode>**=0) is entered
<format>	0	Long format alphanumeric **<oper>**
	1	Short format alphanumeric **<oper>**
	2	Numeric **<oper>**; GSM Location Area Identification number

<netact>	0	User-specified GSM access technology
	1	GSM compact
	3	GSM EGPRS
	7	User-specified LTE M1 A GB access technology
	9	User-specified LTE NB S1 access technology

MCC and MNC codes in Spain

- These codes are provided in the AT+COPS <oper> field as the operator identifier (e.g. Vodafone: “21401”

Networks	MCC + MNC codes
Vodafone España, SAU	214 01
Alta Tecnologia en Comunicacions, S.L.	214 02
France Telecom España, SA	214 03
Xfera Móviles, S.A.	214 04
Telefónica Móviles España, SAU	214 05
Vodafone España, SAU	214 06
Telefónica Móviles España, SAU	214 07
Euskaltel, SA	214 08
France Telecom España, SA	214 09
ZINNIA TELECOMUNICACIONES, S.L.U.	214 10
TELECOM CASTILLA-LA MANCHA, S.A.	214 11
SAC CONVERGENT AGGREGATION SERVICES, S.L.U.	214 12
Telecable de Asturias, SAU	214 16
R Cable y Telecomunicaciones Galicia, SA	214 17
E-Plus Móviles, SL	214 19

AT+CEREG

- Provides information about the registration status, the PSM configuration and network parameters.
- Format of the unsolicited responses (activated with AT-CEREG=4)

+CEREG: <stat>[,[<tac>],[<rac>],[<ci>],[<AcT>][,,[,[<Active-Time>],[<Periodic-TAU>]]]]

<Active-Time> String type; one byte in an 8 bit format. Requested Active Time value (T3324) to be allocated to the UE. The requested Active Time value is coded as one byte (octet 3) of the GPRS Timer 2 information element coded as bit format (e.g. "00100100" equals 4 minutes).

<Periodic-RAU> String type; one byte in an 8 bit format. Requested extended periodic TAU value (T3412) to be allocated to the UE in E-UTRAN. The requested extended periodic TAU value is coded as one byte (octet 3) of the GPRS Timer 3 information element coded as bit format (e.g. "01000111" equals 70 hours).

<stat>

0 Not registered, MT is not currently searching an operator to register to.The GPRS service is disabled, the UE is allowed to attach for GPRS if requested by the user.

1 Registered, home network.

2 Not registered, but MT is currently trying to attach or searching an operator to register to. The GPRS service is enabled, but an allowable PLMN is currently not available. The UE will start a GPRS attach as soon as an allowable PLMN is available.

3 Registration denied, The GPRS service is disabled, the UE is not allowed to attach for GPRS if it is requested by the user.

4 Unknown

5 Registered, roaming

<tac> String type (string should be included in quotation marks); two byte location area code in hexadecimal format (e.g. "00C3" equals 195 in decimal)

<ci> String type (string should be included in quotation marks); two bytes cell ID in hexadecimal format

<AcT>

0 User-specified GSM access technology

7 User-specified LTE M1 A GB access technology

9 User-specified LTE NB S1 access technology