LoRa$^{\text{TM}}$ protocol
Evaluations, limitations and practical test

Eduardo Ruano

May 11, 2016

Supervised by: Bernard Tourancheau
Olivier Alphand

Contents

1 Summary of project

The aim of this study is to shed light on the new LoRa^{TM} protocol, one of the many technologies that is presented as the definitive solution to cover the increasing needs of the Low Power Wide Area Networks created to extend the Internet of Things everywhere and to everybody.
The next document describes the main characteristics of this new technology, explaining the PHY layer LoRa^{TM} as well as the MAC layer LoRaWAN^{TM} . Even so, its characteristics and many of its applications are very broad and many of them are still under development and still to come, that's why it has been impossible to explain all its functions, for example the use of LoRa^{TM} for the geolocation.
For this propose, we have divided the document on two main parts, on the first one, we explain LoRa^{TM} protocol main features and some of their limitations due to the Duty Cycle regulation on the ISM band or due to the use of Aloha as Medium Acces Control.
On the second part, we focus on the LoRa^{TM} performance on real scenarios to check some of its principal features. Check the differences between Spreading Factors on long distances, compare the performance of the Cyclic Redundancy Checks against the Spreading Factor or evaluate how the free-licensed ISM band can affect the efficiency of the transmissions.

2 Introduction to Low Power Wide Area Network

Internet of Things is a reality. As we can see on Figure 2, every year we can found more devices connected to the network and we can agree that IoT still has a long way to go and grow.

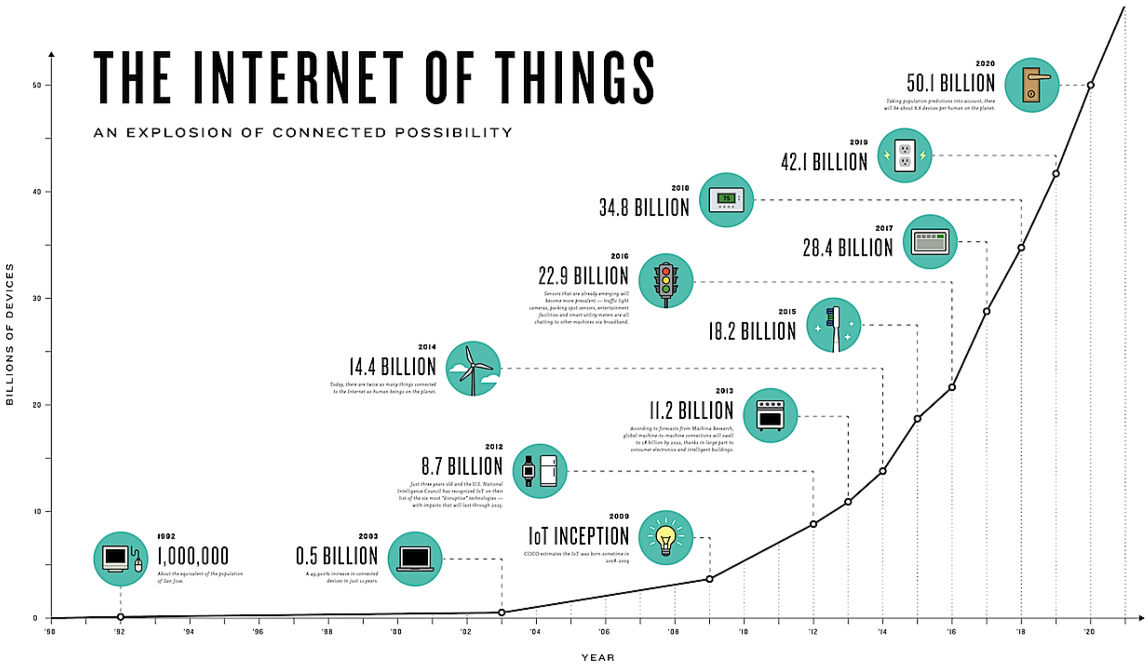

Figure 1: Growth of connected objects[1].

This rise of devices, systems and services connected and exchanging data between them, need a powerful and strong wireless technology to support them. Cellular and satellite machine-to-machine (M2M) technologies have traditionally filled the gap, but cost, power and scalability concerns make these choices less appealing for the future.
Low Power Wide Area Network (LPWAN) has arisen to solve this problem. The applications and devices connected for the IoT are diverse, but they have many common attributes: low cost, low energy consumption, extended range, scalability...
We expect that in 2024 more than the 40% of the wide area M2M connections will be LPWAN. That is why several companies are developing different protocols to implement the LPWAN technology.

[1] https://www.ncta.com/platform/industry-news/infographic-the-growth-of-the-internet-of-things/

Figure 2: Some companies that are devoloping LPWAN technology.

Some of this news protocols are standards, some are standards in progress, and some others are proprietary. Some of these protocols take advantage of industrial, scientific, and medical - ISM - frequency bands, and some others have been implemented over licensed spectrum as well. There are some of them that use collision avoidance techniques, generally using the technique ”Listen Before Talk” (LBT). Others, derived from the Aloha protocol, which does not listen, but rather transmits and waits for a brief acknowledgement. If it does not hear back, then it assumes a collision occurred - this is collision detection instead of avoidance - and resends according to a back-off algorithm.
To compare the protocols we keep in mind some of the most important characteristics of the LPWAN and some others important features like:

- **Range:** We all know that the range of any wireless technology can vary based on what is in the way of the signal. A dense urban setting, with walls, buildings, reflections, lots of other people and traffic, means much shorter range than a rural one. And a flat, unobstructed rural setting will behave better than a hilly one.

  As we see range cover is not easy to compare in real conditions so the values showed on Table 1 have to be taken cautiosly.

- **Band and spectrum:** As we have explained there are protocols that use the free ISM band and others that use licensed spectrum. We can see also differences into the spectrum use, there are some protocols that use Ultra Narrow Band, some are Narrow Band, and some others like LoRa^{TM} uses Spread Spectrum (Chirp Spread Spectrum).

- **Data Rate:** This is another highly variable parameter, it can depend on the distance to the receptor, the obstructions...

- **Over-the-air updates:** This may sound like a random characteristic, but it can be important depending on the application and the deployment of the end-devices. When the number of end-devices is hight or the devices are placed on a remote place, we will get greats advantages of this feature.

- **Handover:** IoT has an extensive application fields, we will find applications where the end-devices will be static units, in other cases they will be mobile (vehicles in urban or rural areas, for farm implementations for example). So it is natural to wonder whether a protocol that can handle the hand-off of a device as it moves between hubs, this matters only to the extent that a ”session” of some sort is being maintained.

Table 1: Comparative LPWAN protocols.

	LoRa^{TM} [1]	**SIGFOX** [2]	**NWave** [3]	**On-Ramp** [4]	**Telensa** [5]
Range [km]	3-8 urban 15-20 suburban 15-45 flat	3-10 urban 30-50 suburban	10	4	Up to 8
Band	Sub-Ghz	Sub-Ghz	Sub-Ghz	2.4 Ghz	Sub-Ghz
Modulation	Spread Spectrum	Ultra-Narrow Band	Ultra-Narrow Band	Spread Spectrum	Ultra-Narrow Band
Data Rate [kbps]	0.3-22	0,1-0,6	0,1	19000 bps/MHz	low
OTA upgrades	Yes	No	Yes	Yes	Yes
Handover	No	Yes	No	Yes	Yes

2.1 Introduction

As we have explained, Low Power Wide Area Networks (LPWAN) are projected to support a major portion of the billions of devices forecasted for the Internet of Things (IoT). To try to standardize LPWAN, it was created the LoRa^{TM} Alliance [1], a structured organization with multiple tiers of membership led by Semtech, IBM, Actility, and Microchip to name a few [6]. The Alliance members collaborate to drive the global success of the LoRa^{TM} protocol (LoRaWAN^{TM}), by sharing knowledge and experience to guarantee interoperability between operators in one open global standard.

2.1.1 LoRa^{TM} PHY layer

LoRa^{TM} is the physical layer or the wireless modulation utilized to create the long range communication link, it is Semtech^{TM}'s physical layer, and it is not open. Many legacy wireless systems use Frequency Shift Keying (FSK) modulation as physical layer because it is a very efficient modulation for achieving low power. LoRa^{TM} is based on Chirp Spread Spectrum (CSS) modulation, which maintains the same low power characteristics as FSK modulation but significantly increases the communication range. Chirp Spread Spectrum has been used in military and space communication for decades due to the long communication distances that can be achieved and due to its robustness to interferences. Nevertheless LoRa^{TM} arrives as the first low cost implementation for commercial usage.
The advantage of LoRa^{TM} is in the technology's long range capability. A single gateway or base station can cover entire cities or hundreds of square kilometres. Range highly depends on the environment or obstructions in a given location, but LoRa^{TM} and LoRaWAN^{TM} have a link budget greater than any other standardized communication technology (according to SX1272/73 Datasheet[16] the link budget is 157 dB). The link budget, typically given in decibels (dB), is the primary factor in determining the range in a given environment.
With LoRa^{TM} , the communication between end-devices and gateways is spread out on different frequency channels and data rates. The selection of the data rate is a trade-off between communication range and message duration, communications with different data rates do not interfere with each other. LoRa data rates range from 0.3 kbps to 20 kbps.
On the following table we can observe the different rates for each spreading factor [2]. On section 3.2 we will see the origin of this values, also we can find more information on LoRa^{TM} Modulation Basics [7] document.

Table 2: Spreading factors attributes.

Spreading Factor	Bandwidth [kHz]	Spreading Factor [chips/symbol]	Bit rate of the signal [bits/sec]	Chip rate [chips/sec]	Time per symbol [sec/symbol]
SF7	**125**	128	5469	125000	0,001024
SF8	**125**	256	3125	125000	0,002048
SF9	**125**	512	1758	125000	0,004096
SF10	**125**	1024	977	125000	0,008192
SF11	**125**	2048	537	125000	0,016384
SF12	**125**	4096	293	125000	0,032768

2.1.2 LoRaWAN^{TM} MAC layer

LoRaWAN^{TM} defines Media Access Control (MAC) layer protocol and system architecture for the network while the LoRa^{TM} physical layer (PHY) enables the long-range communication link. The protocol and network architecture have the most influence in determining the battery lifetime of a node, the network capacity, the quality of service, the security, and the variety of applications served by the network. LoRaWAN^{TM} is open-source and it is assembled by the LoRa^{TM} Alliance.

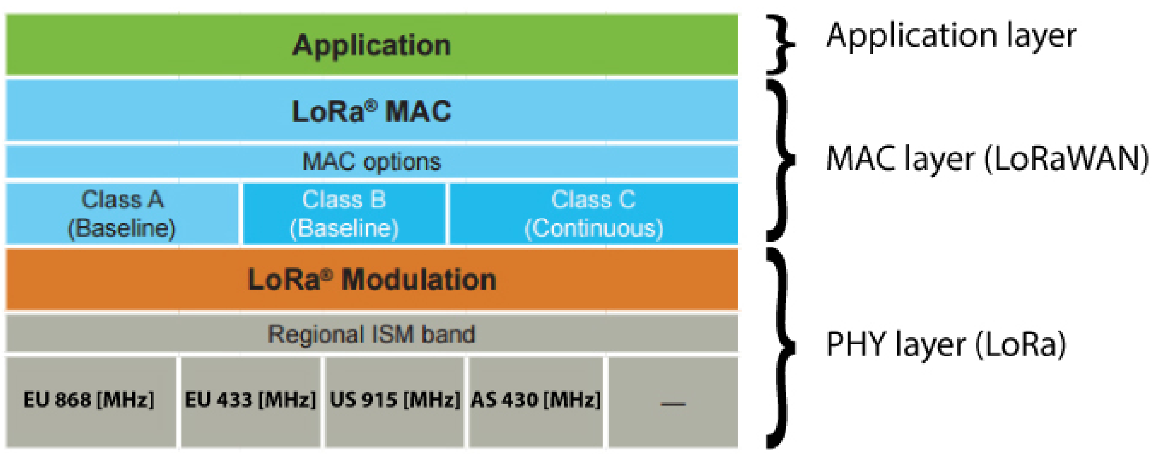

Figure 3: LoRa^{TM} protocol stack[2].

On Figure 3, we see the LoRa^{TM} protocol stack, it consists on an application layer, MAC layer and PHY layer.

Data from the application layer and MAC commands required to establish connection between end-device and gateway are carried as MAC payload, then the MAC layer constructs the MAC frame using MAC payload.

Finally the PHY layer uses MAC frame as PHY payload and constructs the PHY frame after inserting preamble, PHY header, CRC and entire frame CRC. This final frame is transmitted into the air on the required RF carrier. The RF parameters including frequencies, bands, power levels, modulation and the basic RF protocols are all encapsulated in the LoRa^{TM} RF or physical layer attributes.

[2]`https://www.lora-alliance.org/portals/0/documents/whitepapers/LoRaWAN101.pdf`

It is extremely important for any LPWAN to incorporate security. LoRaWAN^{TM} utilizes two layers of security: one for the network and one for the application.

2.2 LoRaWAN^{TM} network architecture

LoRaWAN^{TM} networks typically are laid out in a star-of-stars topology in which gateways relay messages between end-devices and a central network server at the backend.
In a LoRaWAN^{TM} network, end-devices are not associated with a specific gateway. Instead, data transmitted by a device is typically received by multiple gateways. Each gateway will forward the received packet from the end-device to the cloud-based network server via some backhaul (either cellular, Ethernet, satellite, or Wi-Fi). All communication is generally bi-directional, although uplink communication from an end-device to the network server is expected to be the predominant traffic.
The intelligence and complexity is pushed to the network server, which manages the network and will filter redundant received packets, perform security checks, schedule acknowledgements through the optimal gateway, and perform Adaptive Data Rate... If a node is mobile or moving there is no handover needed from gateway to gateway, which is a critical feature to enable asset tracking applications a major target application for IoT.

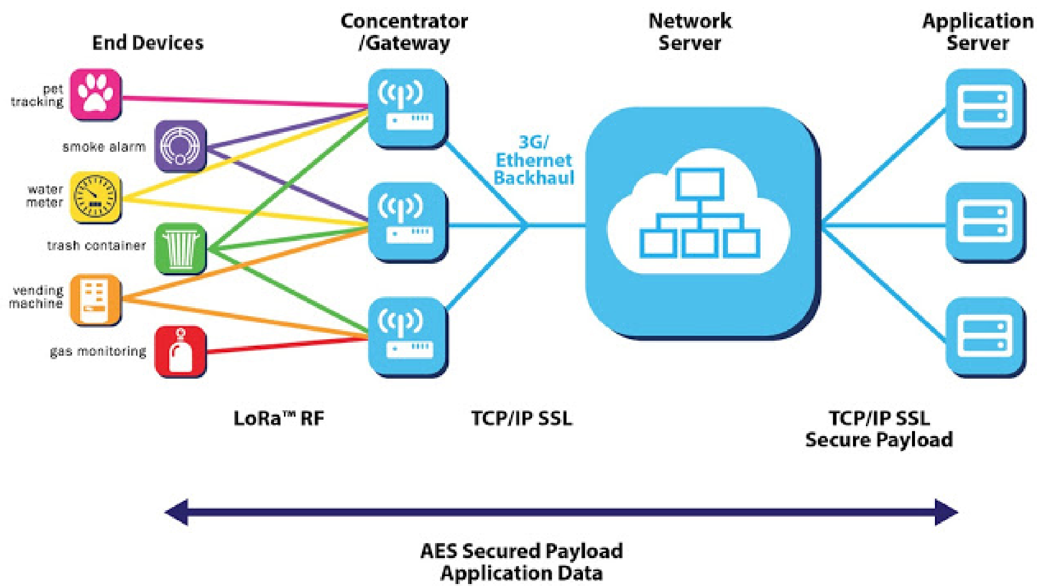

Figure 4: LoRaWAN^{TM} architecture[3].

The network security ensures the authenticity of the device in the network while the application layer security ensures that the network operator does not have access to the end users application data. AES encryption is used with the key exchange utilizing an IEEE EUI64 identifier.

- Network session key: It is specific for the end-device. It is used by both the network server and the end-device to calculate and verify the message integrity code of all data messages to ensure data integrity. It is further used to encrypt and decrypt the payload field of a MAC-only data messages.
- Application session key: It is specific for the end-device. It is used by both the network server and the end-device to encrypt and decrypt the payload field of application-specific data messages. It is also

[3] https://www.lora-alliance.org/portals/0/documents/whitepapers/LoRaWAN101.pdf

used to calculate and verify an application - level message integrity code that may be included in the payload of application-specific data messages.

- Application key: It is an AES-128 application key specific for the end-device that is assigned by the application owner to the end-device and most likely derived from an application-specific root key exclusively known to and under the control of the application provider. Whenever an end-device joins a network via over-the-air activation, the AppKey is used to derive the session keys NwkSKey and AppSKey specific for that end-device to encrypt and verify network communication and application data.

Employing these layers of encryption ensures that the LoRa™ network remains sufficiently secure.

2.2.1 LoRa™ gateway

As we have already said LoRa™ gateways relay messages between end-devices and a central network server at the backend.
Gateways responsible to receive and transmit all packets to and from the RF end-devices and the server, contain the SX1301 [13], a digital baseband chip that manages a massive digital signal processing engine specifically designed to offer breakthrough gateway capabilities in the ISM bands worldwide.
There are many LoRa™ gateways models from different manufacturers, on LORIOT.io [4] we can find diverse models of them.
Each network gateway is listening the same set of 9 channels. Those 9 channels are not all equivalent. Some are dedicated to Adaptive Data Rate communication with end-devices and to high speed LoRa™ communication, a last one is reserved to high speed FSK (Frequency Shift Keying). Depending on the geographical area, the frequencies of these channels vary, on Tables 3 and 4 we see the frequency bands and channels for the European area.

Table 3: Bands and regulations according to the ERC Recommendation 70-03 [14].

Band number	**Frequency [MHz]**	**Duty Cycle**	**Power**
0	865.0 - 868.0	1% or LBT+AFA [5]	25 mW = 14dBm
1	868.0 - 868.6	1% or LBT+AFA	25 mW = 14dBm
2	868.7 - 869.2	0.1% or LBT+AFA	25 mW = 14dBm
3	869.4 - 869.65	10% or LBT+AFA	500 mW = 27dBm
4	869.7 - 870.0	1% or LBT+AFA	25 mW (no dutcy-cyle requirement if power < 5 mW/7 dBm)

[4] `https://www.loriot.io/gateways.html`
[5] LBT+AFA: Listen Before Talk (LBT) with Adaptive Frequency Agility (AFA).

Table 4: Channels setup.

Channel number	Frequency [MHz]	Available Spreading Factors	Band	Usage
3	867.1	SF7 to SF12	0	For Adaptive Data Rate communications
4	867.3	SF7 to SF12	0	For Adaptive Data Rate communications
5	867.5	SF7 to SF12	0	For Adaptive Data Rate communications
6	867.7	SF7 to SF12	0	For Adaptive Data Rate communications
7	867.9	SF7 to SF12	0	For Adaptive Data Rate communications
0	868.1	SF7 to SF12	1	For Adaptive Data Rate communications
1	868.3	SF6 to SF12	1	For Adaptive Data Rate communications
2	868.5	SF7 to SF12	1	For Adaptive Data Rate communications
8	868.8	SF7 to SF12	2	Supports only FSK modulation , used for high speed transfer with end-devices or between gateways

When a gateway receives a frame no matter from which end-device, it simply checks the radio PHY layer CRC and then forwards it to the network server through any IP link (Ethernet , cellular, ..).
In our case the packet is sent to the demo server provided by Semtech: `iot.semtech.com`. Through this public site, we can see all the packets received by all the gateways connected to this cloud.
The gateway does not perform any of the LoRa^{TM} protocol tasks, but rather acts as a pure bridge between the air and the IP network.
Each uplink from an end-device may be received by several gateways and though multiple copies of the same uplink frame may reach the network server. The server de-duplicates the received frames, performs the authentication, decrypts the payload and provides it to the customer.

3 LoRa^{TM} modulation

3.1 Shannon-Hartley Theorem

In information theory, the Shannon-Hartley theorem states the maximum rate at which information can be transmitted over a communications channel of a specified bandwidth in the presence of noise.
The theorem establishes Shannon's channel capacity for a communication link and defines the maximum data rate that can be transmitted within a specified bandwidth in the presence of noise interference:

$$C = B \cdot \log_2(1 + \frac{S}{N}) \tag{1}$$

Where:

- **C** = channel capacity [bps].
- **B** = channel bandwidth [Hz].
- **S** = average received signal power [W].
- **N** = average noise or interference power [W].
- **S/N** = Signal-to-Noise Ratio (SNR) expressed as a linear power ratio [dB].

If the noise is one-sided white noise, then the total noise power N in a bandwidth B is $B{\cdot}N_0$.

$$C = B \cdot \log_2(1 + \frac{S}{B \cdot N_0}) \tag{2}$$

Manipulating Equation 2 we obtain:

$$C = (\frac{S}{N_0}) \cdot (\frac{N_0 \cdot B}{S}) \cdot \log_2(1 + \frac{S}{B \cdot N_0}) \tag{3}$$

$$C = (\frac{S}{N_0}) \cdot \log_2(1 + \frac{S}{B \cdot N_0})^{(\frac{N_0 \cdot B}{S})} \tag{4}$$

We can apply the definition of number e on Equation 4, because in our cause B will be big, in the order of kHz:

$$\lim_{x \to \infty} (1 + \frac{1}{x})^x = e$$

$$C_\infty = \frac{S}{N_0} \cdot \log_2(e) = \frac{S}{N_0} \cdot 1.44 \tag{5}$$

On Equation 5 we can see the maximum information transmission rate possible for a system of given power but no bandwidth limitations. To reach this limit only the transmitted signal bandwidth needs be increased. This characteristic is the basic principle of Spread Spectrum as we see on next section.

3.2 Basic principle of Spread Spectrum

As has been noted above, by increasing the bandwidth of the signal we can compensate the degradation of the Signal-to-Noise Ratio of a radio channel.
Spread Spectrum techniques are methods by which a signal generated with a particular bandwidth is deliberately spread in the frequency domain, resulting in a signal with a wider bandwidth.
In LoRa^{TM} modulation the spreading of the spectrum is achieved by generating a chirp signal that continuously varies in frequency. An advantage of this method is that timing and frequency offsets between transmitter and receiver are equivalent, greatly reducing the complexity of the receiver design. The frequency bandwidth of this chirp is equivalent to the spectral bandwidth of the signal. The wanted data signal is chipped at a higher data rate and modulated onto the chirp signal.

3.3 Chirp Spread Spectrum

LoRa^{TM} modulation, derivative of Chirp Spread Spectrum (CSS), works by moving an RF tone around through time in a very linear way. LoRa^{TM} transmissions work by chirping, breaking the chirps in different places in terms of time and frequency in order to encode a symbol. One of the important LoRa^{TM} features is the ability to generate a stable chirp using a frac-N phase lock loop (PLL) [15]. On Figure 5 we can see the waveform of an up and down linear chirp and its frequency evolution through time.

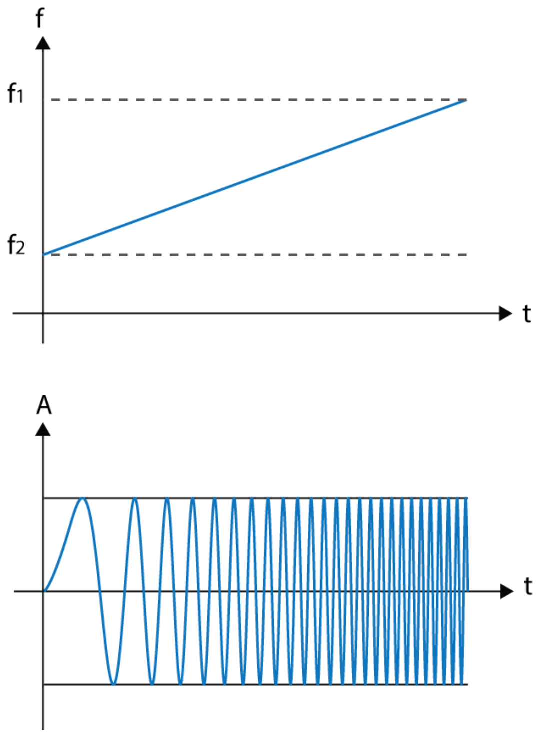

Figure 5: A linear chirp waveform; a sinusoidal wave that increases or decrease in frequency linearly through time. Left: Up chirp waveform. Right: Down chirp waveform.

Depending on the bandwidth and the Spreading Factor selected, the time of the frequency sweep (time symbol) will take more or less time. One increment on the selected Spreading Factor will duplicate the time of the symbol, so one symbol sent at SF12 will be 32 times longer than one symbol sent at SF7. However the bandwidth will be inversely proportional with time, duplicate the bandwidth will divide the symbol time by a half.
On Figures 6 and 7 we see how this two parameters (Spreading Factor and bandwidth) determine the symbol time of our transmission.

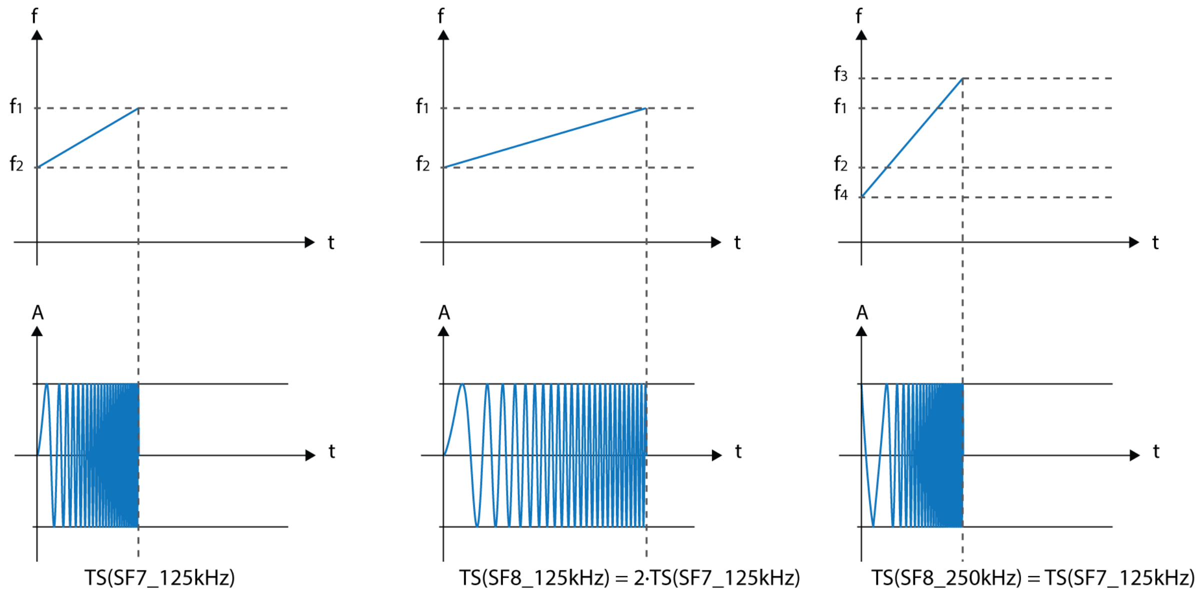

Figure 6: Relation between symbol time, Spreading Factor and the bandwidth selected.

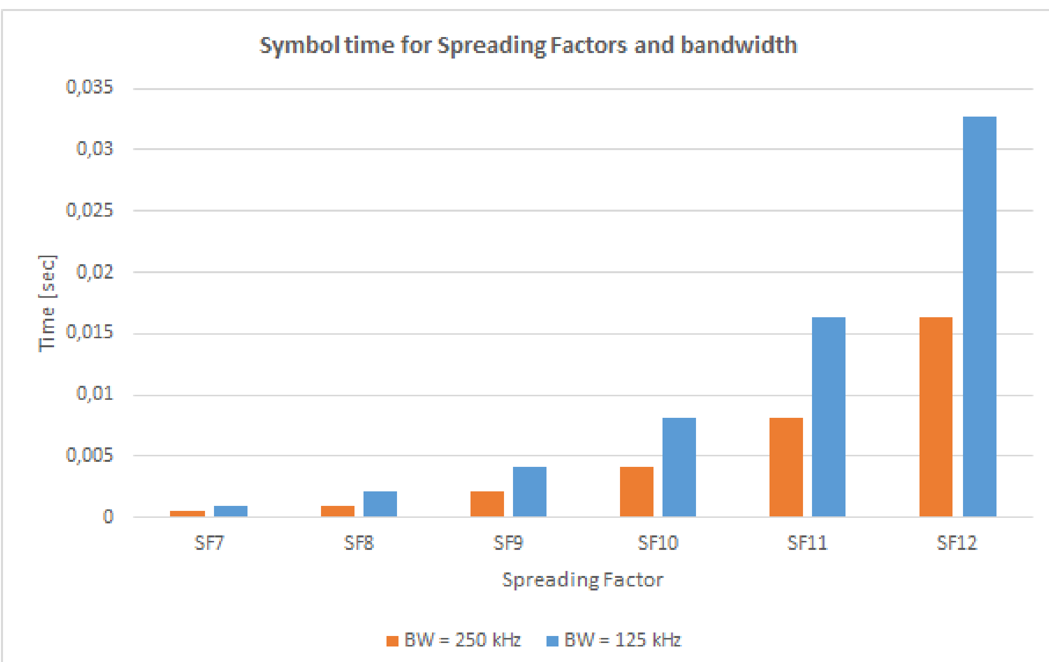

Figure 7: Symbol time duration on different Spreading Factors and bandwidths.

The fact that LoRa^{TM} transmissions jump from one place to another at a particular time might mean one bit string vs. another. It is not just simply binary, it has a lot of information you can convey (high symbol depth).
Pure Frequency Shift Keying (FSK) uses a stationary tone for some time and then jumped to another tone for a while, you would see different lines, or tones. This is called 2-ary FSK, which denotes two frequency symbols (on Figure 8 we see an example of FSK modulation with frequency jumps between f_0 and f_1).

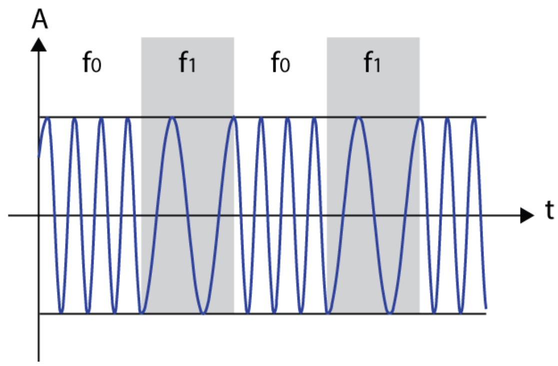

Figure 8: FSK signal in time.

M-ary FSK has multiple frequency tones which can represent even more symbols. LoRa^{TM} has taken this concept, but it does everything on a chirp. So, its getting processing gain.
On Figure 9 we see 5 LoRa^{TM} symbols, we appreciate how the frequency varies linearly in time and the frequency hops for represent different symbols, we see also how the chirp nominally covers the entire bandwidth (BW) once during one symbol time. Because it has a very distinct pattern, the LoRa^{TM} receiver can detect quieter chirps, for example below the noise floor. This feature allows to demodulate many signals with different Spreading Factors at the same time taking advantage of his orthogonality.

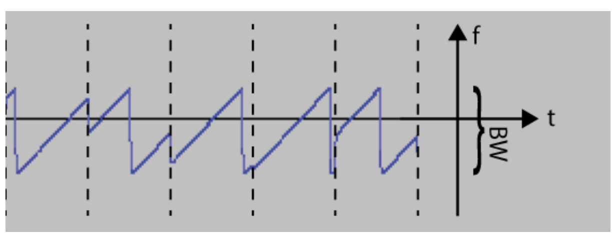

Figure 9: LoRa^{TM} signal frequency per time.[6].

On Figure 10 we see another example of the LoRa^{TM} modulation, we can see 10 up-chirps followed by 2.25 down-chirps corresponding with the 12.25 symbols of preamble (8 configurable symbols + 4.25 default symbols). The following chirps correspond to the PHY header and the PHY payload.

[6] `https://revspace.nl/DecodingLora`

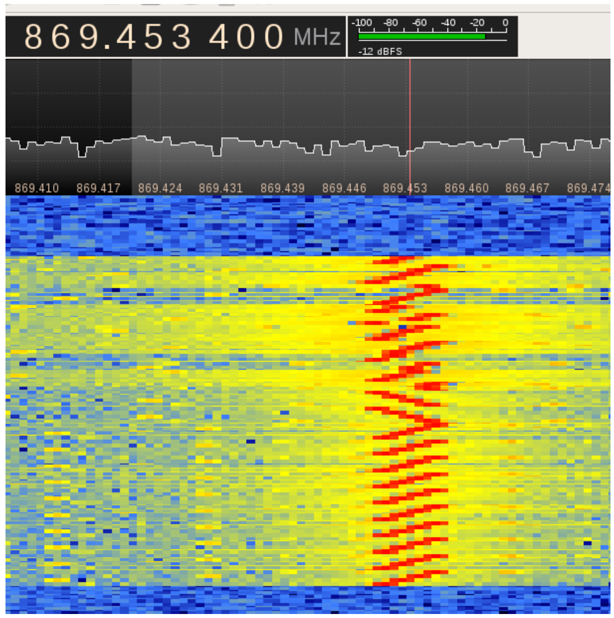

Figure 10: LoRa™ signal transmission.[7].

On LoRa™ , one symbol encodes SF bits. And as we said, one symbol is one sweep over the entire bandwidth of the LoRa™ signal. The starting frequency of this sweep can take one of 2^{SF} positions, so if you can determine the starting frequency of the sweep, you can calculate the codeword encoded in that symbol (consisting on SF bits).
The number of symbols depends in a complex way on the number of bytes you send as we will se on Section 7.7. The modulation adds a number of preamble symbols, a header, some CRC fields and also sends some extra bits so everything is always rounded to a number of bits divisible by 4.

3.4 Link budget

The link budget of a wireless system or network is a measure of all the gains and losses from the transmitter, through the propagation channel, to the target receiver. These gains and losses include system gains and losses associated with the antenna, matching networks... as well as losses associated to the propagation channel itself (either though modelling or measured data).
The link budget of a wireless network link can be expressed as:

$$P_{Rx}[dBm] = P_{Tx}[dBm] + G_{system}[dB] - L_{system}[dB] - L_{channel} - L_M[dB] \tag{6}$$

Where:

- P_{Rx} = power incident at the receiver.
- P_{Tx} = the transmitted power.
- G_{system} = system gains such as those associated with directional antennas...
- L_{system} = losses associated with the system such as feed-lines, antennas (in the case of electrical short antennas associated with many remote devices)...

[7] `https://revspace.nl/DecodingLora`

- $L_{channel}$ = losses due to the propagation channel, either calculated via a wide range of channel models or from empirical data.
- L_M = miscellaneous losses (fading margin, body loss, polarization mismatch, other losses...).

To obtain a correct demodulation on the receiver we have to introduce the term of sensitivity, the minimum input signal required to demodulate the input signal having a specified Signal-to-Noise Ratio.
The RF sensitivity on LoRa^{TM} device's depends on the Spreading Factor and the bandwidth of the modulated signal. On Table 5, according to the SX1272/73 Datasheet[16], we can see the sensitivity values for the different LoRa^{TM} signals available on ours LoRa^{TM} Motes used on the tests. We appreciate how the sensitivity decrease with the Spreading Factor in return for a longer transmission time.

Table 5: Sensitivity values for a correct demodulation on SX1272/73 transceivers.

Spreadin factor	**Sensitivity [dBm] 125 kHz**	**Sensitivity [dBm] 250 kHz**	**Sensitivity [dBm] 500 kHz**
SF7	-124	-122	-116
SF8	-127	-125	-119
SF9	-130	-128	-122
SF10	-133	-130	-125
SF11	-135	-132	-128
SF12	-137	-135	-129

Another important factor for the correct demodulation of the packet is the Signal-to-Noise Ratio. LoRa^{TM} Motes used on our tests are equipped with the WiMOD iM880 [11], we can find the resulting Signal-to-Noise Ratio required at the receiver input on Table 6.

Table 6: Required SNR at the receiver input.

Spreadin factor	**SNR [dB]**
SF7	-7.5
SF8	-10
SF9	-12.5
SF10	-15
SF11	-17.5
SF12	-20

4 LoRaWAN™ classes

LoRa™ network distinguishes between three bidirectional classes: class A, class B and class C. This three classes serve different applications and have different requirements in order to optimize a variety of end applications. The device classes trade off network downlink communication latency versus battery lifetime. In a control or actuator-type application, the downlink communication latency is an important factor.

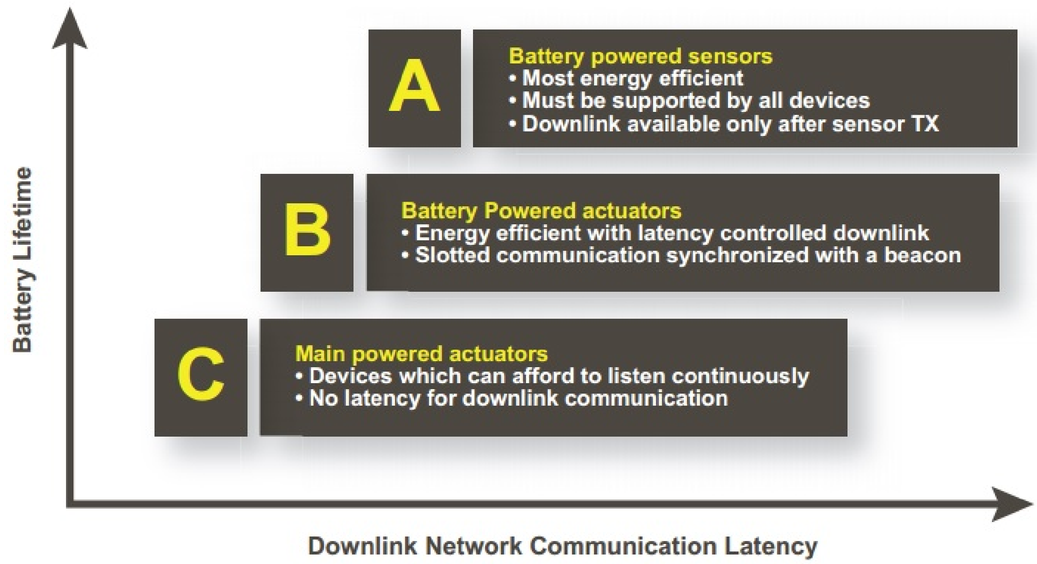

Figure 11: LoRaWAN™ classes comparative[8].

Class A: End-devices of class A allow bi-directional communications whereby each end-device's uplink transmission is followed by two short downlink receive windows.
The transmission slot scheduled by the end-device is based on its own communication needs, when the end-device needs to transmit, it will do it following the Medium Access Control Aloha. On annex A we study the limitations of this MAC protocol applied to a LoRa™ network.
This class A operation is the lowest power end-device system for applications that only require downlink communication from the server shortly after the end-device has sent an uplink transmission. Downlink communications from the server at any other time will have to wait until the next scheduled uplink is done. This class, supported by all devices, is intended for battery powered end-devices or actuators with no latency constraint. It can be useful for transmissions mainly in the uplink sense such as sensors for control temperature, traffic, metering, monitoring, mobile asset tracking...

[8] https://www.lora-alliance.org/portals/0/documents/whitepapers/LoRaWAN101.pdf

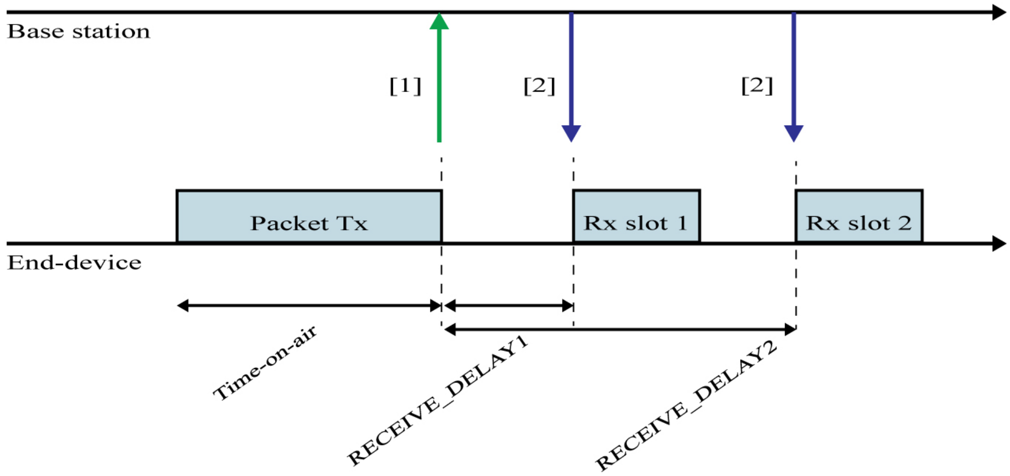

Figure 12: Class A receive slot timing.

[1] All base stations in range receive the packet.
[2] Cloud network server selects the best base station to reply to the end-device in case that the packet received needs confirmation. The reply can be received by the end-devices only in the two reception windows that it opens after each transmission. If the cloud wants to transmit some packet to the end-device it will have to wait until the next packet sent by the end-device.
The first receive window Rx slot 1 uses the same frequency channel as the uplink and a data rate that is a function of the data rate used for the uplink. Rx slot 1 opens RECEIVE_DELAY1 seconds (+/- 20 microseconds) after the end of the uplink modulation. The relationship between uplink and Rx slot 1 downlink data rate is region specific.
The Rx slot 2 window that uses a fixed configurable frequency and data rate, opens RECEIVE_DELAY2 seconds (+/- 20 microseconds) after the end of the uplink modulation. The default parameters are 869.525 MHz and DR0 (SF12, 125 kHz) for the EU863-870 regulation.

Class B: End-devices of class B allow for more receive slots. In addition to the class A random receive windows, class B devices open extra receive windows at scheduled times. In order for the end-device to open its receive window at the scheduled time it receives a time synchronized beacon from the gateway. This allows the server to know when the end-device is listening.
This class can be useful for battery powered devices where bidirectional sensors links are applicable, such as, reading a sensor with occasional control/configuration, alarm sensors with guaranteed alarm delivery.

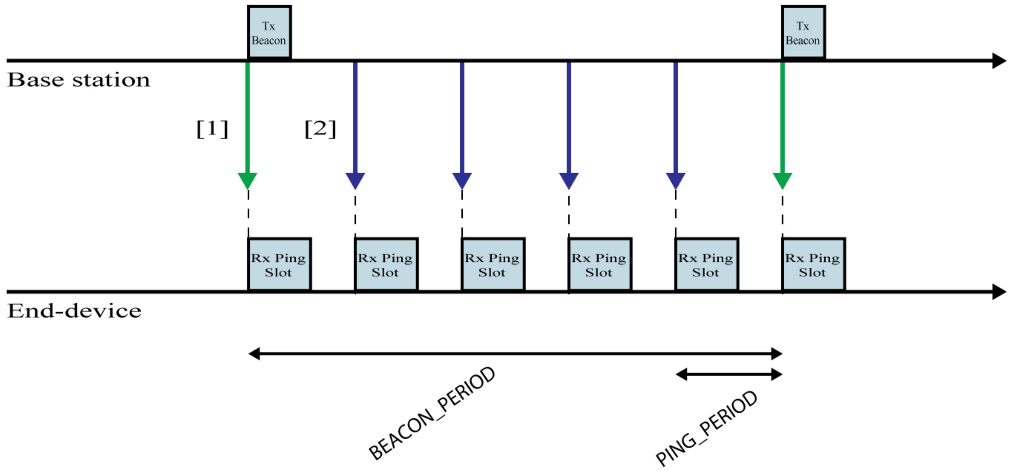

Figure 13: Class B pings and beacons slots timing.

[1] All gateways synchronously by the network server must broadcast a beacon providing a timing reference to the end-devices.
[2] End-devices open periodically receive windows called ”ping slots”, which can be used by the network infrastructure to initiate a downlink communication.

Class C: End-devices of class C have nearly continuously open receive windows, only closed when transmitting. Class C end-devices will use much more power to operate than class A or class B but they offer the lowest latency for server to end-device communication.
This class can be useful for powered devices where a downlink communication is required at any moment: industrial control, real time control of pumps/valves, residential gateways, lighting control, car engine status, car tracking...

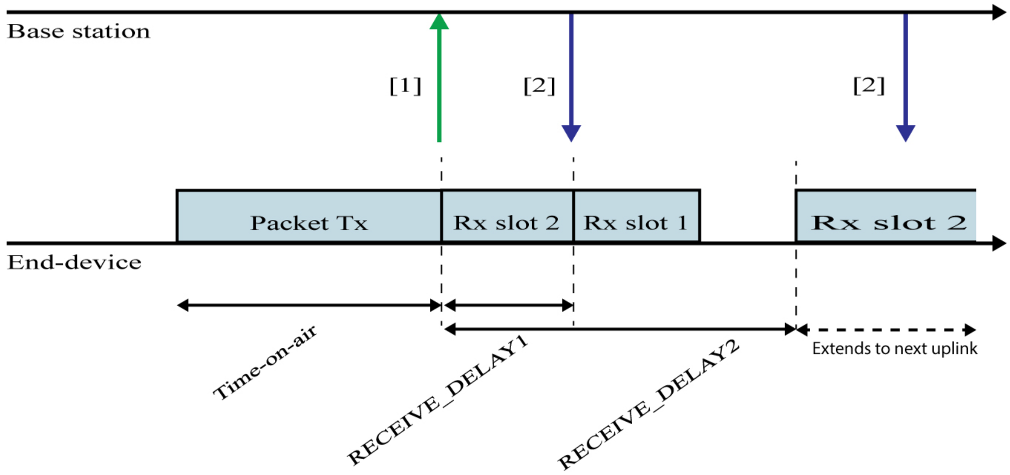

Figure 14: Class C continuous listening mode.

[1] Class C implements the same two receive windows as class A, with the difference that they do not close the second reception window until the next packet needs to be send. A short listening window is open between the end of the transmission and the beginning of the reception slot 1.
[2] Gateway can communicate with the end-devices at any time.

5 LoRaWAN^{TM} message formats

LoRa^{TM} terminology distinguishes between uplink and downlink messages. Uplink messages are sent by end-devices to the network server relayed by one or many gateways, a downlink message is sent by the network server to only one end-device and is relayed by a single gateway.
The following sections explain the packet's structure. We can find detailed information on LoRa^{TM} Datasheet [16] and LoRaWAN^{TM} Specification [17].
On section 7.7 we see how the different configurations of the messages structures modify the Time On Air of the packet.

5.1 Radio PHY layer message structure

LoRaWAN^{TM} employs two types of packet format, explicit and implicit. The explicit packet includes a short header that contains information about the number of bytes, coding rate and whether a CRC is used in the packet. The packet format is shown in the following figure. The LoRa^{TM} packet comprises three elements:

- Preamble.
- Optional header (depending on the selected mode, explicit/implicit).
- Data payload.

Table 7: Radio PHY layer.

Radio PHY layer				
Preamble	PHDR (only in explicit mode)	PHDR_CRC (only in explicit mode)	PHYPayload	CRC (optional)

Preamble
The preamble is used to synchronize the receiver with the incoming data flow. By default the packet is configured with a 12,25 symbols long sequence (8 configurable symbols + 4.25 default symbols). This is a programmable variable so the preamble length may be extended, for example in the interest of reducing to receiver duty cycle in receive intensive applications.

5.1.1 PHDR (PHY Header)

Depending upon the chosen mode of operation two types of header are available.

Explicit mode
Explicit header mode is the default mode of operation and provides information on the payload, namely: the payload length in bytes, the forward error correction code rate and the presence of an optional 2 Bytes CRC of the payload.
In this case the header is transmitted with maximum error correction code (4/8). It also has its own CRC to allow the receiver to discard invalid headers.

Implicit mode
In certain scenarios, where the payload, coding rate and CRC presence are fixed or known in advance, it may be advantageous to reduce transmission time by invoking implicit header mode. In this mode the header is removed from the packet. In this case the payload length, error coding rate and presence of the payload CRC must be manually configured on both sides of the radio link.

5.1.2 PHYPayload

Table 8: PHY Payload structure.

PHYPayload		
1 Byte	1..M Bytes	4 Bytes
MHDR	MACPayload	MIC

5.1.3 MHDR (MAC header)

The MAC header specifies the message type (MType) explained on Section 6.5, Major field specifies the format of the messages exchanged in the join procedure. RFU field are 3 bits Reserved for Future Usage.

Table 9: MAC header structure.

MHDR		
Bit# 7 to 5	Bit# 4 to 2	Bit# 1 to 0
MType	RFU	Major

MType
The LoRaWAN^{TM} distinguishes between six different MAC message types: join request, join accept, unconfirmed data up/down, and confirmed data up/down.

Table 10: MAC messages types.

MType value	Description
000	Join Request
001	Join Accept
010	Unconfirmed Data Up
011	Unconfirmed Data Down
100	Confirmed Data Up
101	Confirmed Data Down
110	RFU
111	Proprietary

5.1.4 MACPayload

The MAC payload of the data messages, also-called "data frame", contains a frame header (FHDR) followed by an optional port field (FPort) and an optional frame payload field (FRMPayload).

Table 11: MAC Payload structure.

MACPayload		
7..23 Bytes	0..1 Bytes	0..N Bytes
FHDR	FPort	FRMPayload

5.1.5 Fport (Port Field)

If the frame payload field is not empty, the port field must be present. If it is present, a FPort value of 0 indicates that the FRMPayload contains MAC commands only. FPort values 1..223 (0x01..0xDF) are application-specific. FPort values 224..255 (0xE0..0xFF) are reserved for future application extensions.

5.1.6 FHDR (Frame header)

The FHDR contains the short device address of the end-device (DevAddr), a frame control octet (FCtrl), a 2 Bytes frame counter (FCnt), and up to 15 Bytes of frame options (FOpts) used to transport MAC commands.

Table 12: Frame header structure.

FHDR			
4 Bytes	1 Byte	2 Bytes	0..15 Bytes
DevAddr	FCntrl	FCnt	FOpts

5.1.7 FCtrl (Control Field)

Table 13: Control Field structure for uplink frames.

FCtrl				
Bit# 7	Bit# 6	Bit# 5	Bit# 4	Bit# 3 to 0
ADR	ADRACKReq	ACK	RFU	FOptLen

Table 14: Control Field structure for downlink frames.

FCtrl				
Bit# 7	Bit# 6	Bit# 5	Bit# 4	Bit# 3 to 0
ADR	ADRACKReq	ACK	FPending	FOptLen

ADR and ADRACKReq

LoRa^{TM} network allows the end-devices to individually use any of the possible data rates. This feature is used by the LoRaWAN^{TM} to adapt and optimize the data rate of the end-devices. This is referred to as Adaptive Data Rate (ADR) and when it is enabled the network will optimize the fastest data rate possible. If the ADR bit is set, the network will control the data rate of the end-device through the appropriate MAC commands. If the ADR bit is not set, the network will not attempt to control the data rate of the end-device regardless of the received signal quality. The ADR bit may be set and unset by the end-device or the network on demand. However, whenever possible, the ADR scheme should be enabled to increase the battery life of the end-device and maximize the network capacity.

ACK
When receiving a confirmed data message, the receiver shall respond with a data frame that has the acknowledgement bit (ACK) set. If the sender is an end-device, the network will send the acknowledgement using one of the receive windows opened by the end-device after the send operation. If the sender is a gateway, the end-device transmits an acknowledgement at its own discretion.
Acknowledgements are only sent in response to the latest message received and are never retransmitted.

FOptsLen
The frame-options length field byte denotes the actual length of the frame options field (FOpts) included in the frame.
If FOptsLen is 0, the FOpts field is absent. If FOptsLen is different from 0, the port 0 cannot be used (FPort must be either not present or different from 0).

FPending
The frame pending bit (FPending) is only used in downlink communication, indicating that the gateway has more data pending to be sent and therefore asking the end-device to open another receive window as soon as possible by sending another uplink message.

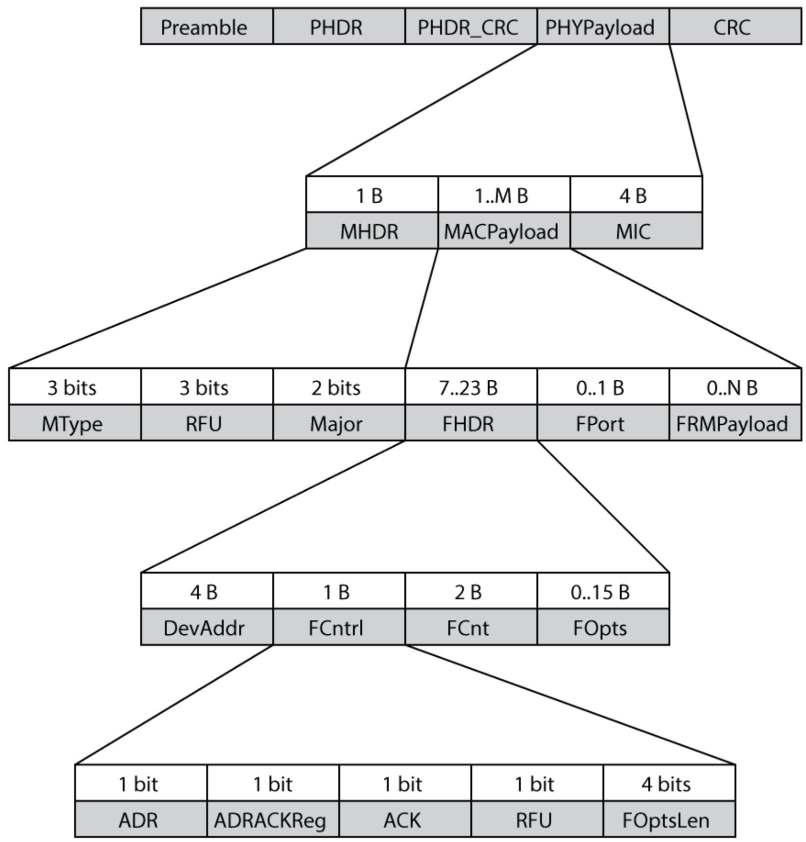

Figure 15: Radio PHY layer message structure.

5.2 MAC commands

For network administration, a set of MAC commands may be exchanged exclusively between the network server and the MAC layer on an end-device. MAC layer commands are never visible to the application or the application server or the application running on the end-device.
A single data frame can contain any sequence of MAC commands, either piggybacked in the FOpts field or, when sent as a separate data frame, in the FRMPayload field with the FPort field being set to 0. Piggybacked MAC commands are always sent without encryption and must not exceed 15 octets. MAC commands sent as FRMPayload are always encrypted and must not exceed the maximum FRMPayload length.
On Table 15 we see the MAC commands available, we can find detailed information about each command on section 5 of LoRaWAN^{TM} Specification [17].

Table 15: MAC commands.

CID	Command	Transmitted by		Short description
		End-device	Gateway	
0x02	**LinkCheckReq**	x		Used by an end-device to validate its connectivity to a network.
0x02	**LinkCheckAns**		x	Answer to LinkCheckReq command. Contains the received signal power estimation indicating to the end-device the quality of reception (link margin).
0x03	**LinkADRReq**		x	Requests the end-device to change data rate, transmit power, repetition rate or channel.
0x03	**LinkADRAns**	x		Acknowledges the LinkRateReq.
0x04	**DutyCycleReq**		x	Sets the maximum aggregated transmit duty-cycle of a device
0x04	**DutyCycleAns**	x		Acknowledges a DutyCycleReq command.
0x05	**RXParamSetupReq**		x	Sets the reception slots parameters.
0x05	**RXParamSetupAns**	x		Acknowledges a RXSetupReq command.
0x06	**DevStatusReq**		x	Requests the status of the end-device.
0x06	**DevStatusAns**	x		Returns the status of the end-device, namely its battery level and its demodulation margin.
0x07	**NewChannelReq**		x	Creates or modifies the definition of a radio channel.
0x07	**NewChannelAns**	x		Acknowledges a NewChannelReq command.
0x08	**RXTimingSetupReq**		x	Sets the timing of the of the reception slots.
0x08	RXTimingSetupAns	x		Acknowledge RXTimingSetupReq command.
0x80 to 0xFF	Proprietary	x	x	Reserved for proprietary network command extensions.

5.3 Cyclic Redundancy Check

The LoRa^{TM} modem also employs a form of Forward Error Correction (FEC) to further improve the robustness of the link, that permits the recovery of bits of information due to corruption by interference. This requires a small overhead of additional encoding of the data in the transmitted packet, the resultant additional data overhead per transmission is shown in Table 16.

Table 16: Cyclic Coding Overhead.

Coding Rate	Overhead ratio
4/5	1,25
4/6	1,50
4/7	1,75
4/8	2

This additional data overhead influences on the time on air as we can see in Equation 7 on Section 7.7. On Table 17 we see the influence of coding rate on time on air.

Table 17: Influence of CR on Time On Air (SF=12, BW=125KHz).

Coding Rate	**Time On Air [s]**
4/5	1.318
4/6	1.449
4/7	1.581
4/8	1.712

6 LoRa^{TM} Firmware

LoRa^{TM} firmwares are available for CooCox-CoIDE, Ride7 and Keil µVision on `https://github.com/Lora-net/LoRaMac-node`.
With ST-Link/v2 [18], a very low-cost professional tool to debug and program STM8 and STM32 MCUs we will be able to flash our code on LoRa^{TM} Mote end-devices with some of the main IDEs.

Figure 16: ST-Link/v2 device.

For our propose we have worked with Keil µVision IDE's. CoIDE offers us a complete free IDE, but Keil µVision on his payment version provides us more debug options very useful for our propose.
One of the advantages that we have with Keil µVision is his Debug (printf) Viewer that allows us to print sentences easily. On section B we can found the suitable configuration to flash our code using Keil µVision IDE.

6.1 LoRa^{TM} Flowchart

On the following pictures, we can see summarised flowcharts of LoRa^{TM} class A operation on the v4.0 firmware.

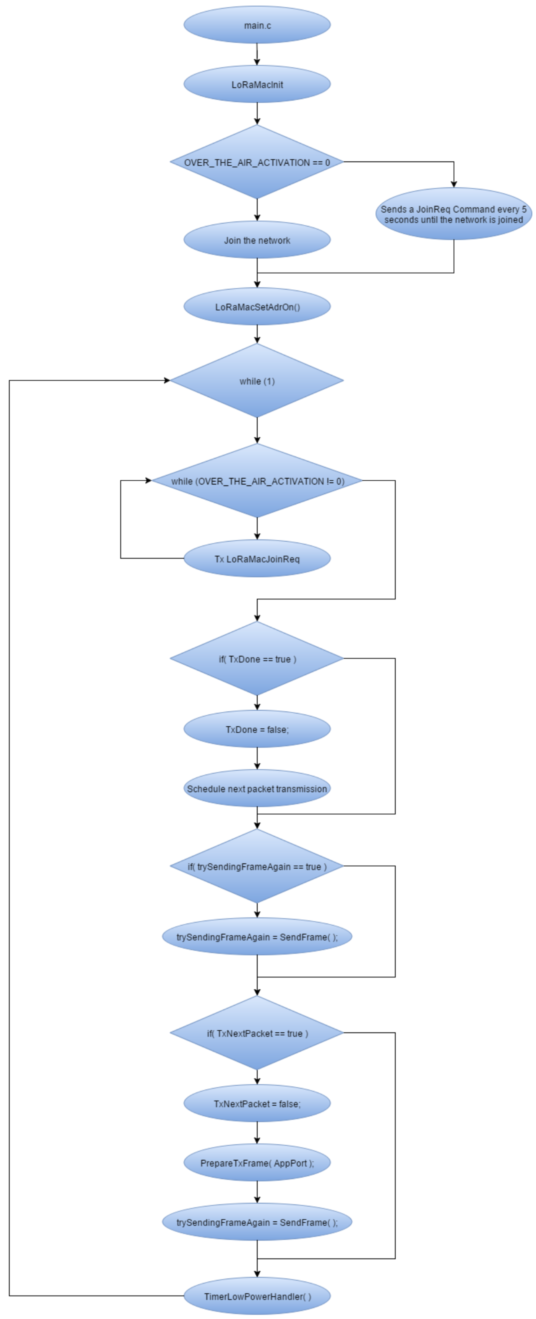

Figure 17: main.c flowchart.

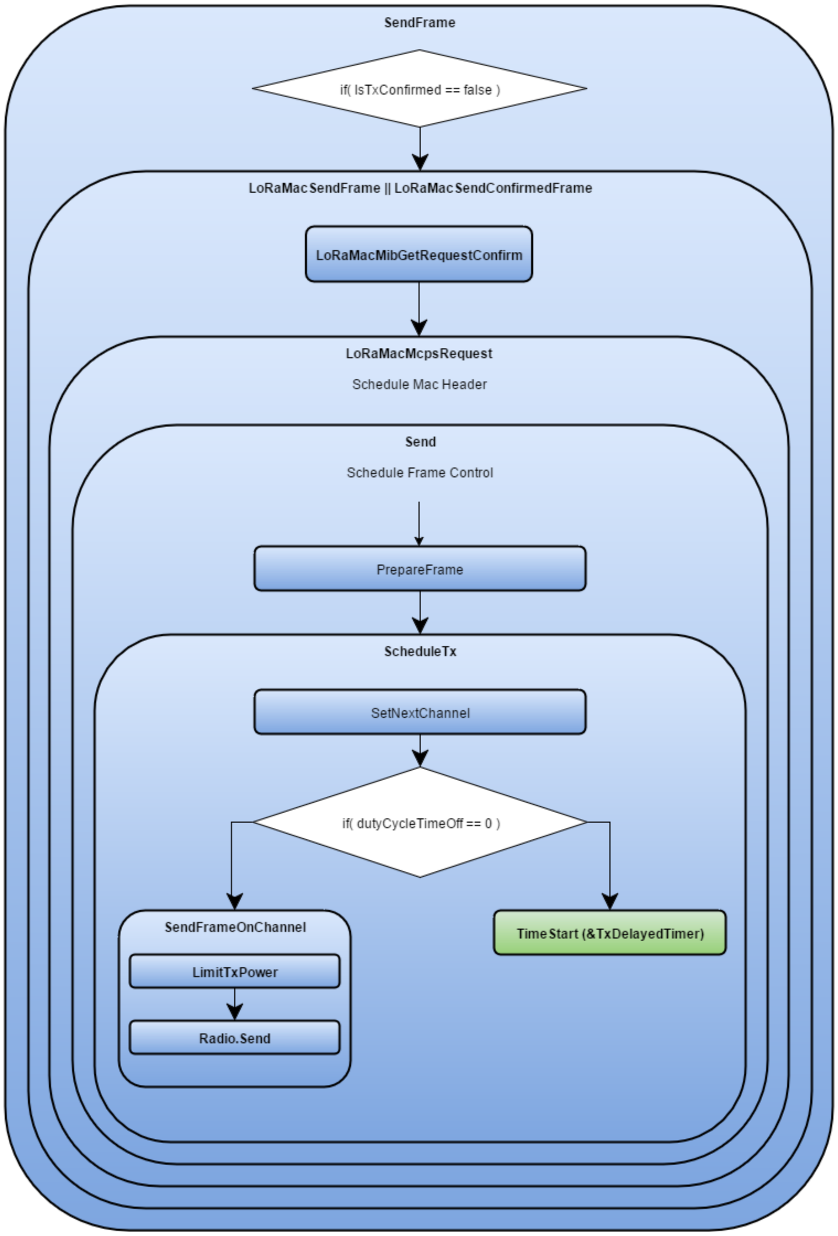

Figure 18: Send flowchart.

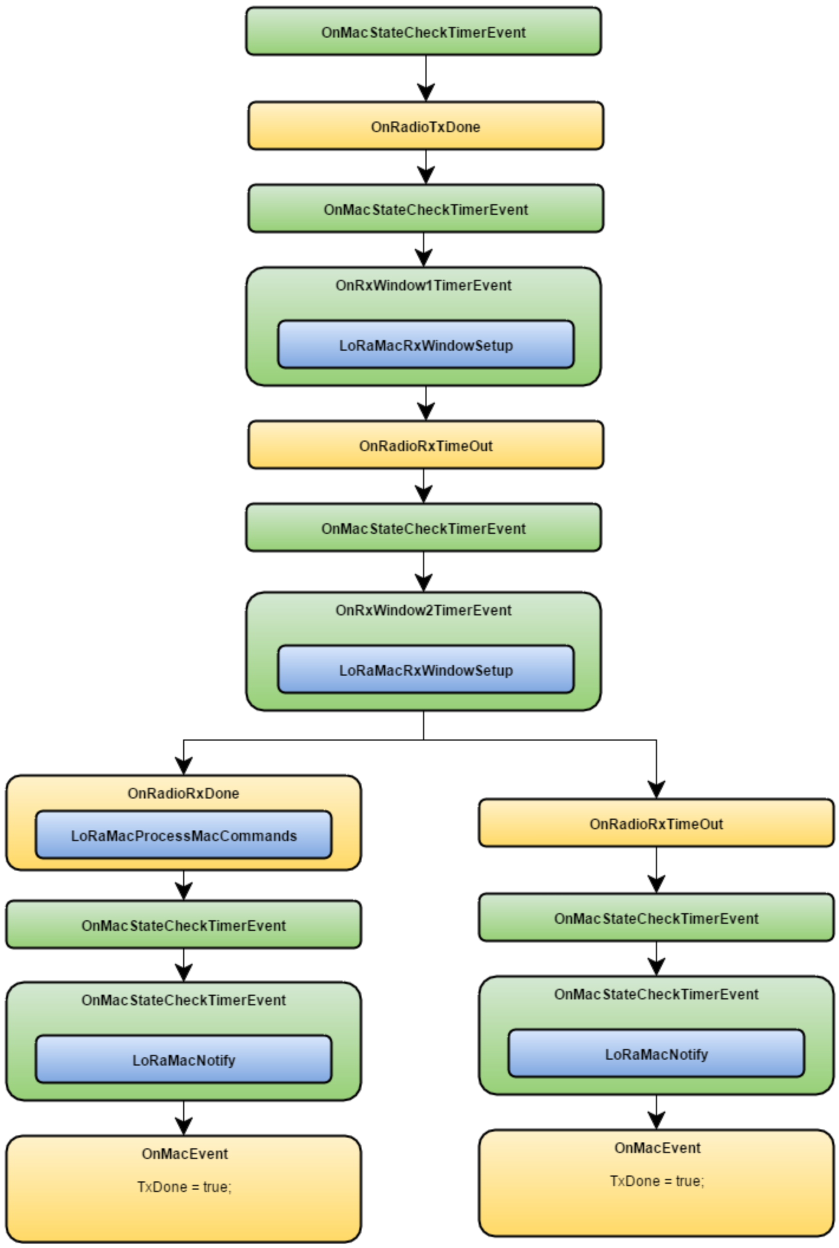

Figure 19: Reception flowchart for unconfirmed packets.

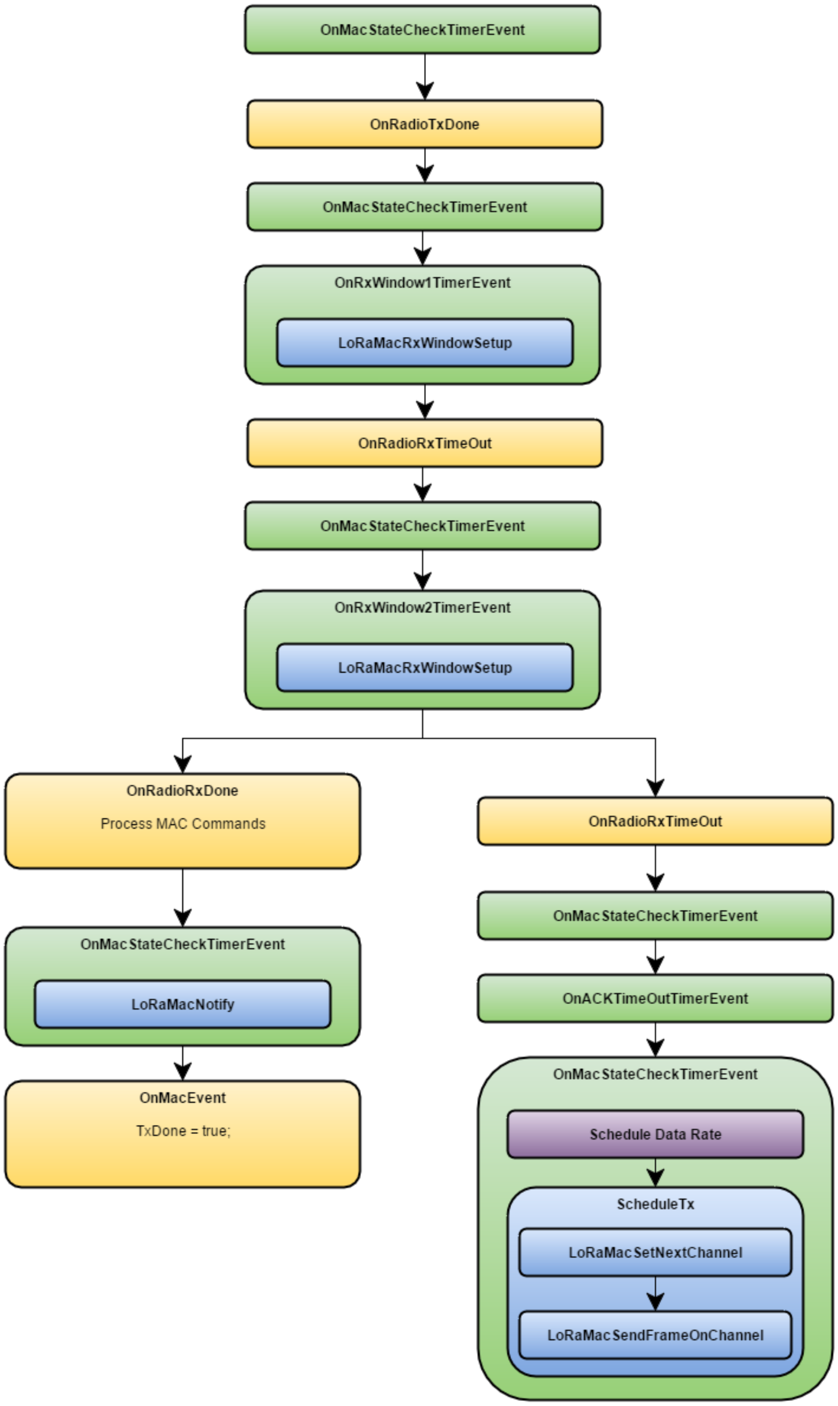

Figure 20: Reception flowchart for confirmed packets.

6.2 Timer events

Timer events functions allow to schedule functions to be executed after a timeout value. The three next functions are required to start the timer events:

timer.h

```
/*!
* \brief Initializes the timer object
* \remark TimerSetValue function must be called before starting the timer
    .
*   this function initializes timestamp and reload value at 0.
*
* \param [IN] obj           Structure containing the timer object
    parameters
* \param [IN] callback      Function callback called at the end of the
    timeout
*/
void TimerInit( TimerEvent_t *obj, void ( *callback )( void ) );

/*!
* \brief Set timer new timeout value
*
* \param [IN] obj   Structure containing the timer object parameters
* \param [IN] value New timer timeout value
*/
void TimerSetValue( TimerEvent_t *obj, uint32_t value );

/*!
* \brief Starts and adds the timer object to the list of timer events
*
* \param [IN] obj Structure containing the timer object parameters
*/
void TimerStart( TimerEvent_t *obj );
```

On LoRa^{TM} firmware we find the next important timer events:

Table 18: Timers events.

TimerEvent	Callback function	TimerStart place
TxNextPacketTimer	OnTxNextPacketTimerEvent	main
MacStateCheckTimerEvent	OnMacStateCheckTimerEvent	SendFrameOnChannel OnMacStateCheckTimerEvent
RxWindowTimer1	OnRxWindowTimerEvent	OnRadioTxDone
RxWindowTimer2	OnRxWindowTimerEvent	OnRadioTxDone
ACKTimeOutTimer	OnACKTimeOutTimerEvent	OnRadioTxDone

6.3 Adaptative Data Rate (ADR)

The Adaptative Data Rate (ADR) is a method implemented by LoRaWAN^{TM} to adjust individually the data rate of the end-devices to ensure the optimal network performance, and scale for capacity.

For this propose, end-devices placed near the gateways must use biggest data rates to allow to the farthest end-devices to use the lowest data rates. In this way, we maximize the capacity of the network, avoiding collision between packets with the same data rate, and allowing the nodes that are very edge of the link budget to use lowest data rates for maximize the distance.
To maximize both battery life of end-devices and overall network capacity, LoRa^{TM} network infrastructure can manage the data rate and output power transmission for each end device individually by implementing ADR.
On table shown on Annex C.1 we see how to configure the default Spreading Factor and how to setup the ADR.
Defining LORAWAN_ADR_ON with value 1 will set the ADR bit of the Control field explained on Section 5.1.7. In this way, the network will control the data rate of the end-device through the appropriate MAC command, in this case with the command SRV_MAC_LINK_ADR_REQ showed on Section 5.2.
When one MAC command is received by the end-device, it will run the OnRadioRxDone function to check what type of frame it has received, in case of a MAC command packet, it will call the ProcessMacCommands function who will read the values from the packet changing the future data rate, power and channels usable for the next uplinks transmissions. We can find more information about SRV_MAC_LINK_ADR_REQ on LoRaWAN^{TM} Specification [17].

6.4 Power Transmission

On LoRa^{TM} firmware we can set the default output power used by the end-devices with the variable LORAMAC_DEFAULT_TX_POWER that can be set from 20 dBm to 2 dBm, however according to ETSI regulations [19] we can not overtake 14 dBm as we have seen on Table 3.
As we explained on previous section 6.3, if ADR is enabled, the output power will be controlled by the network server, who, by the SRV_MAC_LINK_ADR_REQ packet will request to the end-device for change his output transmission power.
On Table 43 shown on Annex C.2 we see how to set the default power transmission.

6.5 Message type

The LoRaWAN^{TM} distinguishes between six different MAC message types: join request, join accept, unconfirmed data up/down, and confirmed data up/down. We can find more information about message types on LoRaWAN^{TM} Specification [17].
Data messages are used to transfer both MAC commands and application data, which can be combined together in a single message. A confirmed-data message has to be acknowledged by the receiver, whereas an unconfirmed-data message does not require an acknowledgement. Proprietary messages can be used to implement non-standard message formats that are not interoperable with standard messages but must only be used among devices that have a common understanding of the proprietary extensions.

6.5.1 Message ackowledge bit

Unconfirmed data up messages don't have to be confirmed by the gateway, this means that if an *unconfirmed data up* packet is lost, the end-device will not know and it will send next packet independently on the good or bad reception. By the other hand, when a receiver receive a *confirmed data up*, it shall respond with a data frame that has the acknowledge bit (ACK) set.
If the sender is an end-device, the network will send the acknowledgement using one of the receive windows opened by the end-device after the send operation. If the sender is a gateway, the end-device transmits an acknowledgement at its own discretion.
Acknowledgements are only sent in response to the latest message received and are never retransmitted.
As we see on the flowchart showed on Section 6.1, on Figure 20, after the second reception window there are two possibilities: the end-device receives the ACK message or not.

If end-device receives the ACK message, it will run OnRadioRxDone function where the ACK packet is processed, then OnMacEvent function will change the value of TxDone variable for schedule the next packet on the next while cycle. On the other hand, if the ACK has not been received by the end-device it will schedule the next data rate (following the explained on Section 6.5.2), it will find a new channel and it will retransmit the same packet again. The end-devices will follow this process until the ACK is received or until the maximum number of retries is reached.
On tables shown on Annex C.3 we see how configure the message type and its parameters.

6.5.2 Retransmission procedure

The number of retransmissions (and their timing) for the same message where an acknowledgement is requested but not received is at the discretion of the end-device and may be different for each end-device, it can also be set or adjusted from the network server by a MAC command.
If an end-device has reached its maximum number of retransmissions without receiving an acknowledgement, it can try to re-gain connectivity by moving to a lower data rate with longer reach. It is up to the end-device to retransmit the message again or to forfeit that message and move on.
On Table 45 shown on Annex C.3 we see how to configure the number of retransmissions.

6.5.3 Data Rate Adaptation during Message Retransmissions

When an end-device attempts the transmission of a confirmed frame toward the network it expects to receive an acknowledgement in one of the subsequent reception slots. In the absence of acknowledgement it will try to re-transmit the same data frame again. This retransmission happens on a new frequency channel, but can also happen at a different data rate (preferable lower) than the previous one.
Any further transmission uses the last data rate used. For example if an end-device sends a confirmed frame first using DR5 and has to retransmit 3 times (twice at DR5 and once at DR4), the next frame transmitted will use DR4.

Table 19: Data Rate adaptation.

Transmission nb	**Data Rate**
1	DR
2	DR
3	max(DR-1,0)
4	max(DR-1,0)
5	max(DR-2,0)
6	max(DR-2,0)
7	max(DR-3,0)
8	max(DR-3,0)

On Figure 21 we see an example of the retransmission procedure.

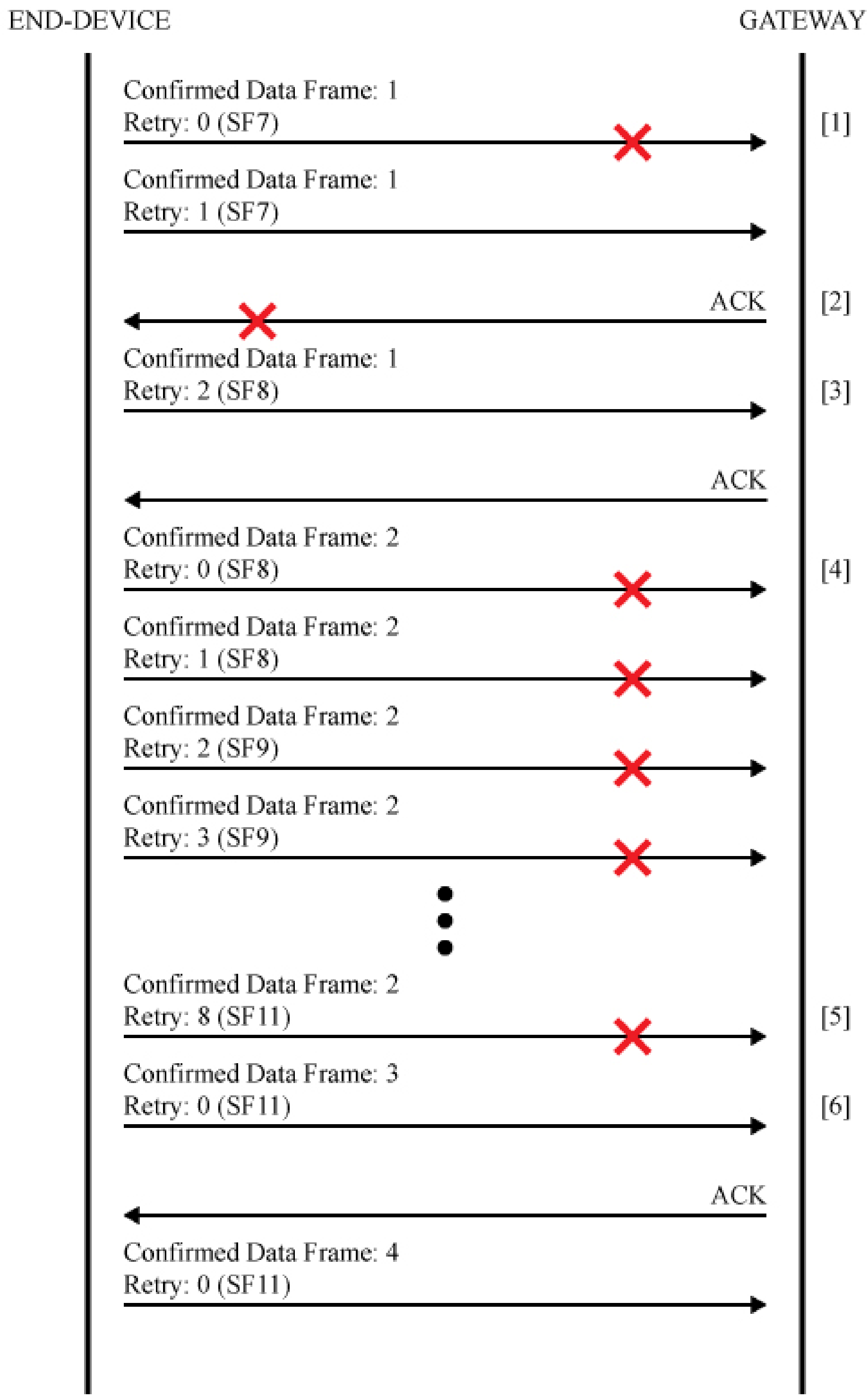

Figure 21: Data Rate Adaptation. Retransmission procedure.

[1] First confirmed frame sent by the end-device is not received by the gateway due to its distance to the gateway or because the packet has been received but it has been demodulated erroneously. As a consequence the gateway does not send any ACK because it has not read any confirmed frame.
The two reception windows opened by the end-device do not receive any ACK frame, so the end-device proceeds to transmit the same frame again and NbTrials variable is incremented.
[2] ACK is not received by the end-device due to any of the reason explained previously.
[3] End-device has not received any ACK frame, so it increments NbTrials variable and following the Adaptative Data Rate showed on Table 19 it increments the Spreading Factor to increase the possibilities to be reached by the gateway.
Data frame is received correctly by the gateway and it replies with the proper ACK frame that is received by the end-device.
[4] The end-device sends the second confirmed frame keeping the last Spreading Factor used.
[5] After 8 attempts (default maximum number of retries) the end-device has not received any ACK so it discard the frame.
[6] End-device sends the third data frame.

We have to recall that, as we have seen on Figure 20, before retransmit one frame again, the end-device runs again the LoRaMacSetNextChannel function. So the channel selected in each transmission is independent with the previous frame.

7 Testing LoRa^{TM}

We wanted to test LoRa^{TM} performance in different scenarios and configurations like:

- RSSI, SNR and PER performance in different scenarios: Line Of Sight, indoor and urban.
- PER fluctuation in time.
- Preamble performance.
- PER for different Cyclic Redudance Check.
- Performance with different packet sizes.
- Study the consecutive packets lost.
- Performance with different message types.

On the market we can find an extensive variety of LoRa^{TM} transceivers provided by different brands, most of them contain the SX1272 [8] or SX1276/77/78/79 [9] chip provided by Semtech^{TM}. Depending on the application, this RF transceivers will be adapted on different objects to act as LoRa^{TM} nodes also known as end-devices.
During our task, we have worked with four end-devices LoRa^{TM} Mote [10]. This device is a test application that provides the possibility to configure almost all parameters of the LoRaWAN^{TM} protocol, it is equipped with the WiMOD iM880A [11], a compact and low-cost radio module that operates in the unlicensed 868 MHz band and combines a powerful Cortex-M3 controller with the LoRa^{TM} transceiver SX1272.
As gateway we have used a Wirnet 868 MHz [12] provided by Kerling that has been installed on the roof of the ENSIMAG building. Through the server provided by Semtech: `iot.semtech.com` we recovered the packets arrived to the gateway for its following process.
On our first experience with LoRa^{TM} technology we decided to test LoRa^{TM} Motes on 3 different scenarios to compare its performance in 3 possible situations: indoor and urban scenarios and direct Line Of Sight with the gateway.
The tests in the different scenarios have been done in different days to minimise the channel occupation and avoid possibles collisions between our packets due to we have setup the Duty Cycle to the 100% for minimise the duration of the tests and to have the possibility of send more packets in each test.

Indoor: End-device located on a small room placed on the floor below the gateway position, at an approximately distance of 30 meters on a floor full of studies.
Line of Sight: End-device placed in a classroom in a building placed in front of the gateway position at an approximately distance of 60 meters.
Urban: End-device placed on a main floor of a house at a distance of 4,08 km from the gateway. Direct line of sign completely impossible.

Table 20: Gateway and end-devices coordinates for our tests.

	Latitude	**Longitude**	**Altitude Reference: 214 m**
Gateway	4511’35.84”N	546’7.61”E	30 m
Indoor	4511’35.58”N	546’8.42”E	25 m
L.O.S.	4511’37.68”N	546’7.75”E	25 m
Urban	4511’34.87”N	543’1.01”E	0 m

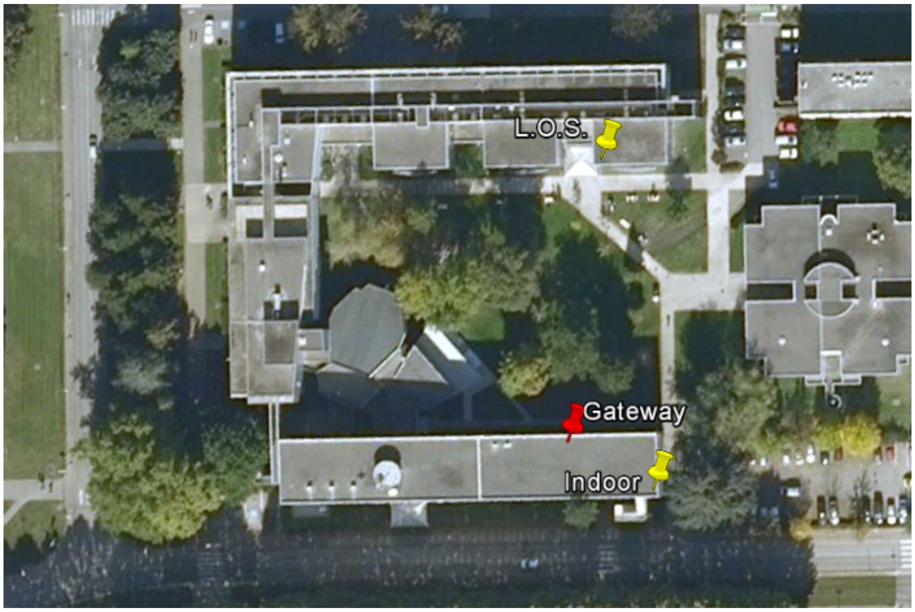

Figure 22: Positions of the end-devices in indoor and Line Of Sight scenarios.

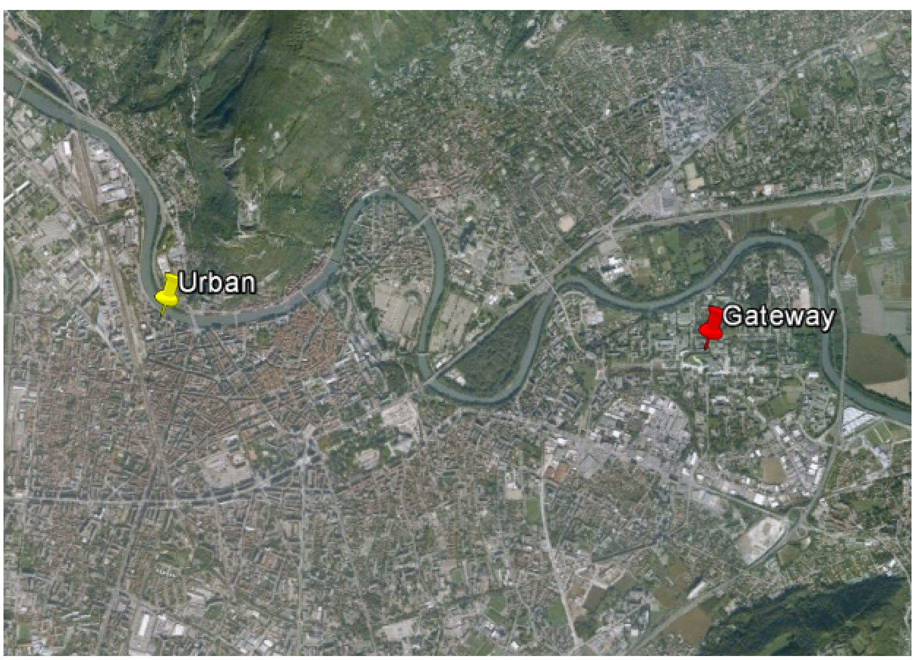

Figure 23: Position of the end-devices in urban scenario.

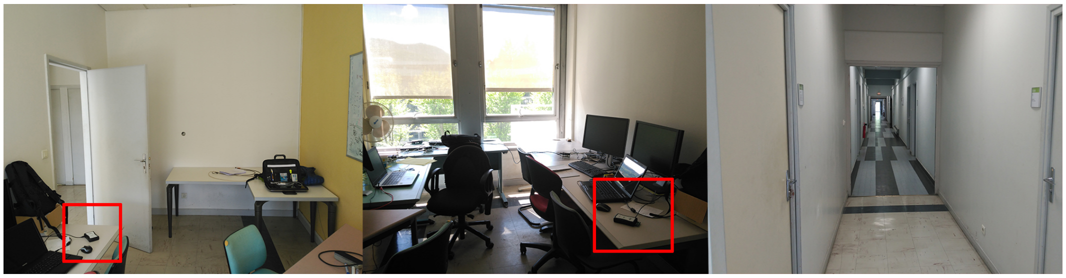

Figure 24: Indoor scenario.

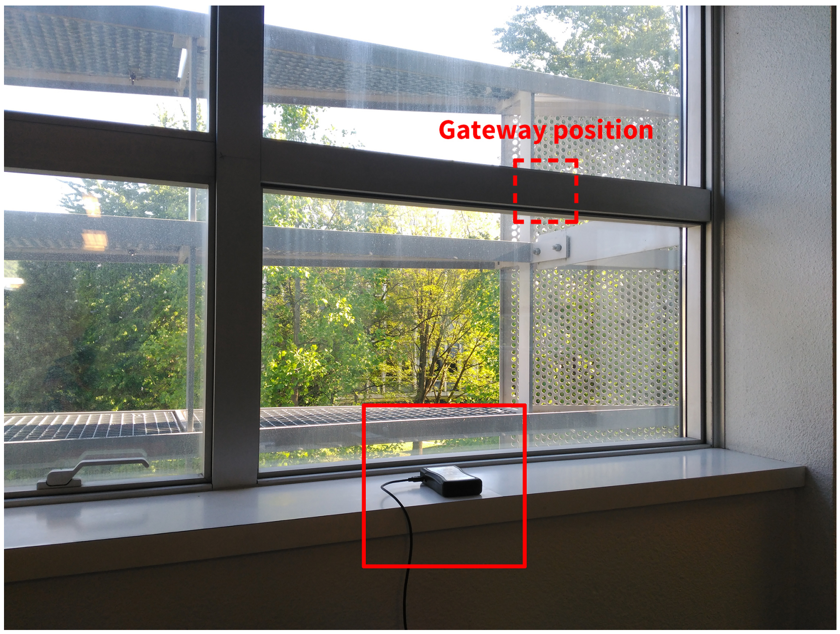

Figure 25: Line Of Sight scenario.

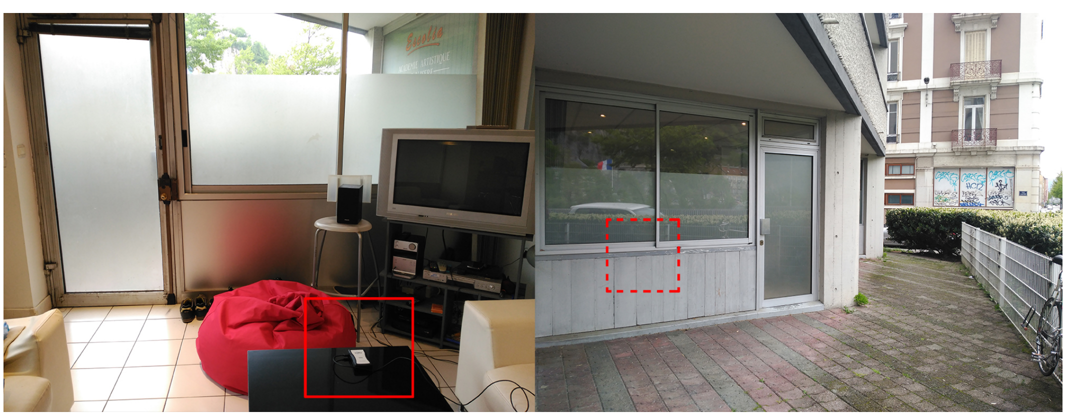

Figure 26: Urban scenario.

7.1 RSSI, SNR and PER performance

We wanted to observe the performance of LoRa^{TM} modulation in the 3 scenarios proposed, and more precisely the performance of the different Spreading Factors on long distances.
For this propose we modified the LoRaWAN^{TM} firmware to send packets over the 8 available channels with the 6 Spreading Factors that use a bandwidth of 125 kHz. Trying to minimise the random changes that can take place on the channels, we have used the next pattern to setup the transmission sequence.

Table 21: Send sequence followed by end-devices.

Sequence number	**Channel**	**Spreading Factor**
1	0	12
2	0	11
[...]	[...]	[...]
5	0	8
6	0	7
7	1	12
8	1	11

Semtech talks about a range of more than 15 km in a rural environment or 2-5 km in a dense urban environment operating in compliance with European regulatory limits [19] [9]. So one of the firsts steps was to verify this values on ours tests.
Two important parameters that will determinate the correct demodulation of the packet are:

- **RSSI:** Received Signal Strength Indicator is a measurement in dBm of the power present in a received radio signal, in our case at the Wirnet 868 MHz, also known as gateway.
- **SNR:** Signal-to-Noise Ratio indicates the ratio of signal power to the noise power measured in dB.

$$SNR_{dB} = P_{signal,dBm} - P_{noise,dBm}$$

7.1.1 RSSI for channel and spreading factor

Before realise the tests we calculated the expected RSSI value applying the link budget formula shown on Section 6.

$$P_{Rx}[dBm] = P_{Tx}[dBm] + G_{system}[dB] - L_{system}[dB] - L_{channel}[dB] - L_M[dB]$$

It has been difficult to find the gain of the antenna, and the losses of the system that must be calculated by empirical experiences. The gain of the antenna will be amongst 3 and 6 dB and the loss of the system should not overtake the 3 dB so for the calculation we compensate the gain of the system with her losses.
P_{Tx} is set to 14 dB and the theoretical channel losses can be calculate with the free space path loss formula.

$$L_{channel}[dB] = 20 \cdot \log_{10}(d[km]) + 20 \cdot \log_{10}(f[GHz]) + 20 \cdot \log_{10}(\frac{4 \cdot \pi}{c})$$

On Table 22 we see the theoretical values of channel losses, we have to recall that the free space path formula only considers the losses due to the distance without considering the obstacles, climatic attenuations, reflections...

[9] http://www.semtech.com/Press-Releases/2013/Semtech-Partners-with-IMST-On-New-Radio-Module-Featuring-Long-Distance-LoRa_Technology.html

Table 22: Channel losses for the 3 scenarios proposed for a frequency of 867.8 MHz.

	Indoor	**L.O.S.**	**Urban**
Distance [km]	0.03	0.06	4.08
$L_{channel}$ **[dB]**	60.75	66.77	103.4

Finally the expected values of RSSI will be:

Table 23: RSSI expected at the receptor.

	Indoor	**L.O.S.**	**Urban**
Theoretical RSSI [dBm]	-46.75 - L_M	-52.77 - L_M	-89.4 - L_M

Comparing the values from Table 23 with the average RSSI obtained on ours tests we get an approximately value of the miscellaneous losses due to fading margin, reflections, obstacles, attenuation due to the climatic attenuations...

Table 24: Theoretical miscellaneous losses.

	Indoor	**L.O.S.**	**Urban**
Theoretical RSSI [dBm]	-46.75 - L_M	-52.77 - L_M	-89.4 - L_M
Average RSSI of the 5 test and 8 channels[dBm]	-77,61	-78,01	-121,09
L_M [dB]	30.86	25.24	31.7

Seen the results of miscellaneous losses from Table 24 attract attention the small differences between the losses from the indoor scenario and the urban (0.84 dB), when the difference between the indoor scenario and the Line Of Sight is 5.62 dB.
On Figure 27 we see the average results of the 5 tests realised on each scenario. In terms of power, we see similar performance between the 8 channels. Comparing the results from each scenario it attract the small differences between the indoor scenario and the Line Of Sight. We expected a bigger difference due to the number of walls that the signal sent from the indoor scenario has to cross.
On Figures 28, 29 and 30, we see the Spreading Factors performance over each channel. In indoor and L.O.S. scenarios we appreciate that SF12 have a worst performance in terms of RSSI.
On Figure 30 we see the clearly difference and the real application of the Spreading Factors. We see that we have not received any data frame from SF7 and SF8, and only in one channel we have received packets from SF9. Another detail looking the RSSI values in this scenario is to compare the minimum RSSI obtained from the packets with the theoretical sensitive of the gateway, this give us a margin value in dB:

$$Margin = Sensitivity_{gateway} - RSSI_{min} = -141dB - (-124dB) = 17dB$$

This margin value allow us reduce the power transmission or increase the distance between the end-device and the gateway still obtaining a RSSI value superior to the sensitivity of the gateway.
On Figures 31, 32, 33 and 34 we show the cumulative frequency distribution and the normal cumulative distribution. These graphs represented with the packets received show us the probability to have a particular RSSI in the 3 exposed scenarios.
On Figure 32 and 33 we see for example that to obtain the best RSSI value is more recommendable use SF10 than the others. However on Figure 34 we see that the performance using the Spreading Factors 9, 10, 11 and 12 will be very similar.

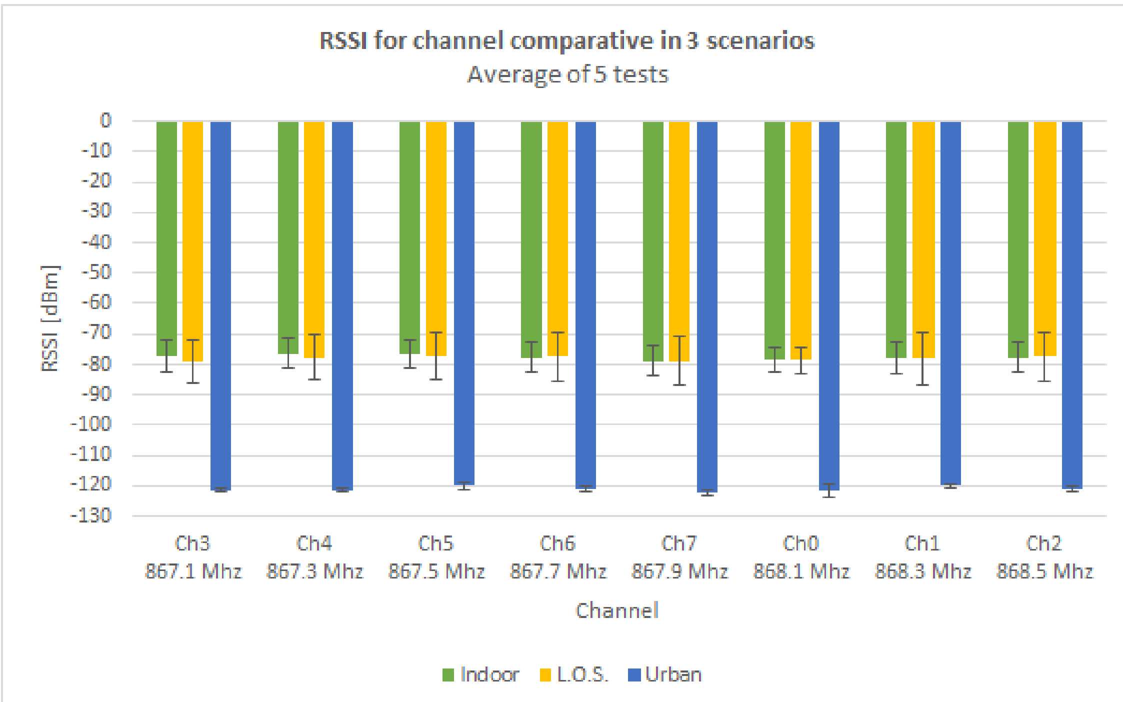

Figure 27: RSSI for channel in 3 scenarios.

Repetitions: 5	**Number of packets send per repetition:** 4800
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor, L.O.S., urban	

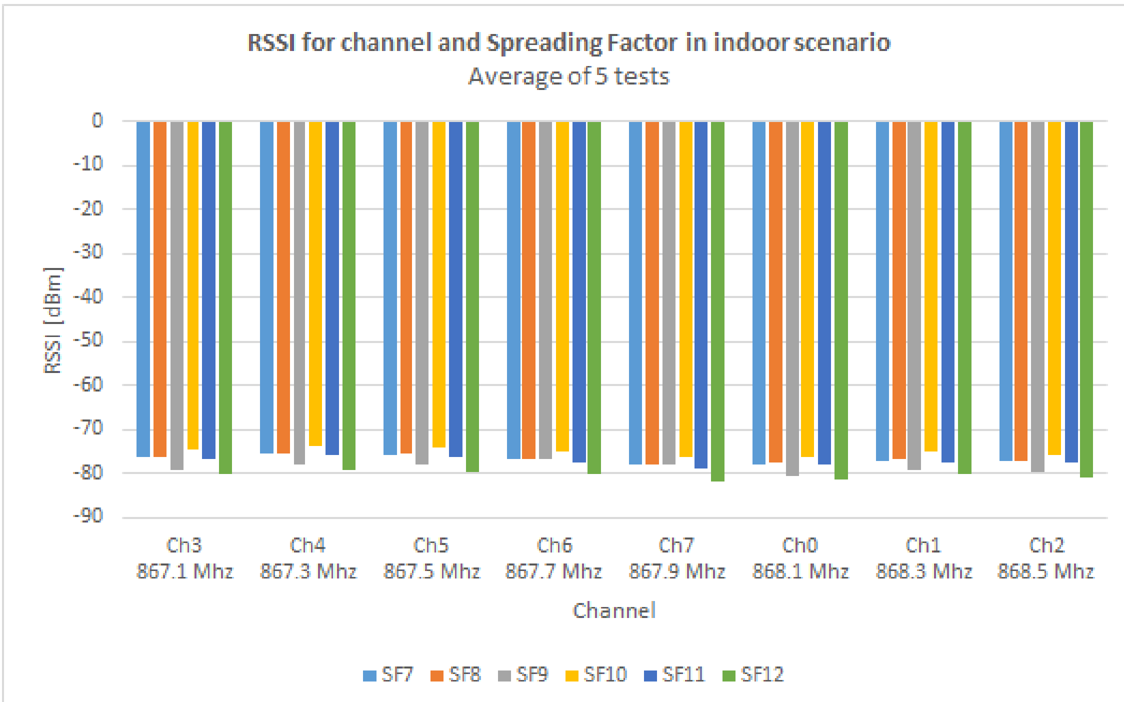

Figure 28: RSSI for channel and spreading factor in indoor scenario.

Repetitions: 5	**Number of packets send per repetition:** 4800
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor	

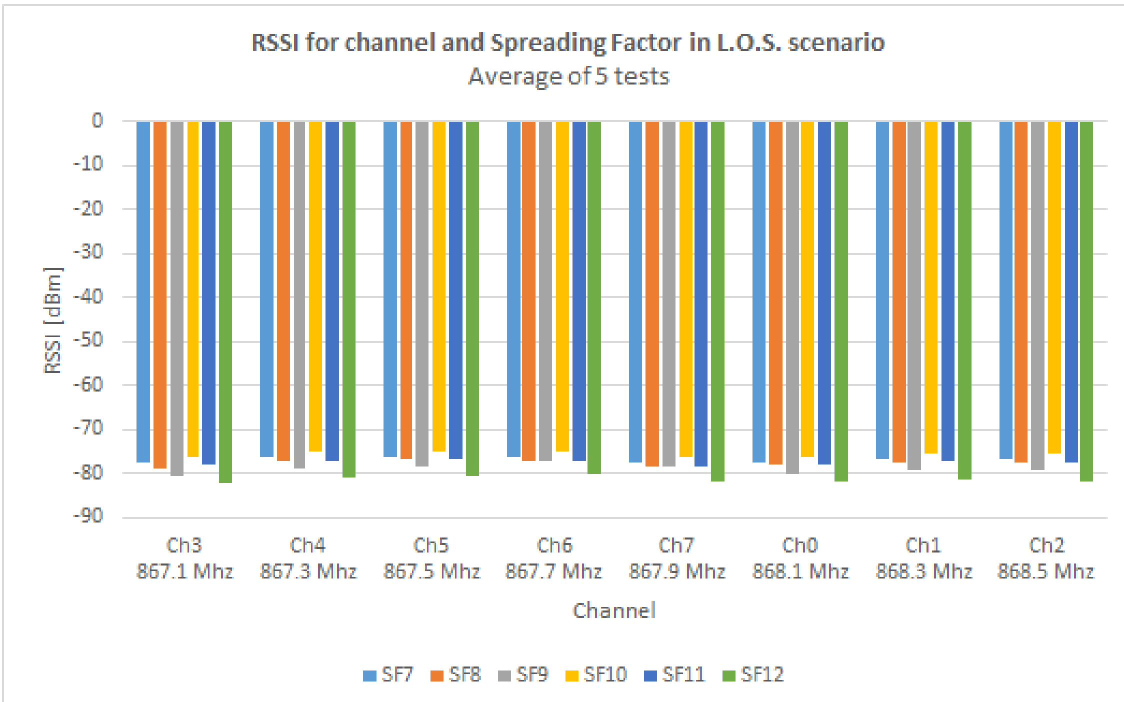

Figure 29: RSSI for channel and spreading factor in L.O.S. scenario.

Repetitions: 5	**Number of packets send per repetition:** 4800
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: L.O.S.	

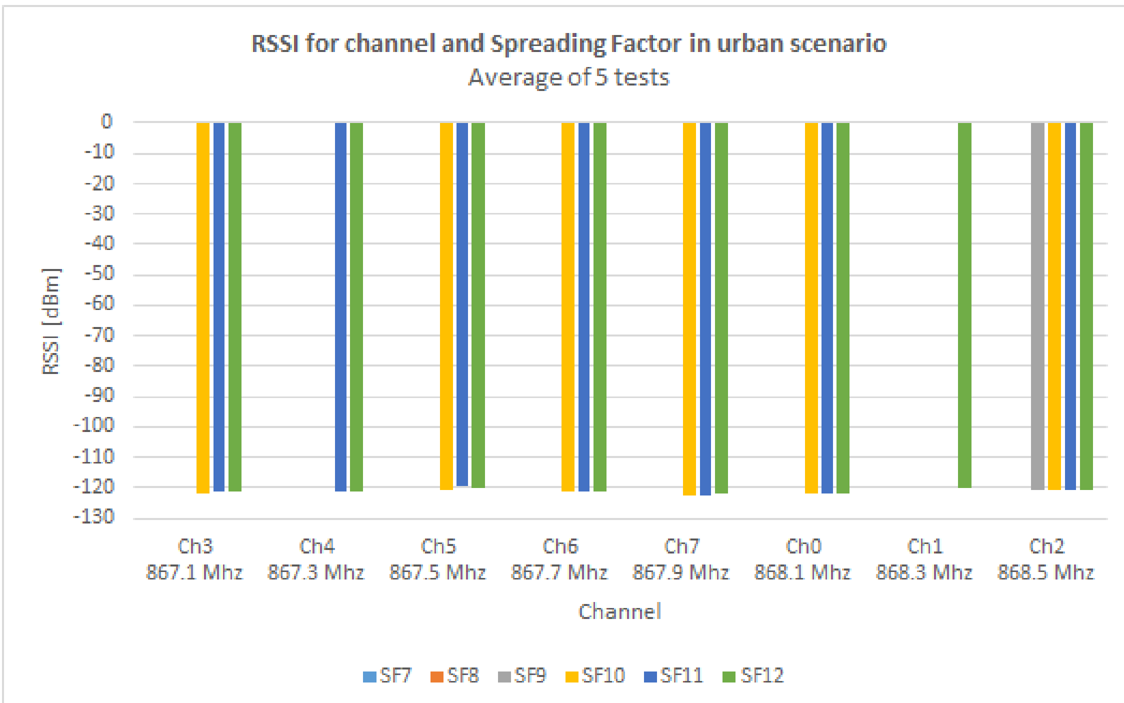

Figure 30: RSSI for channel and spreading factor in urban scenario.

Repetitions: 5	**Number of packets send per repetition:** 4800
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: urban	

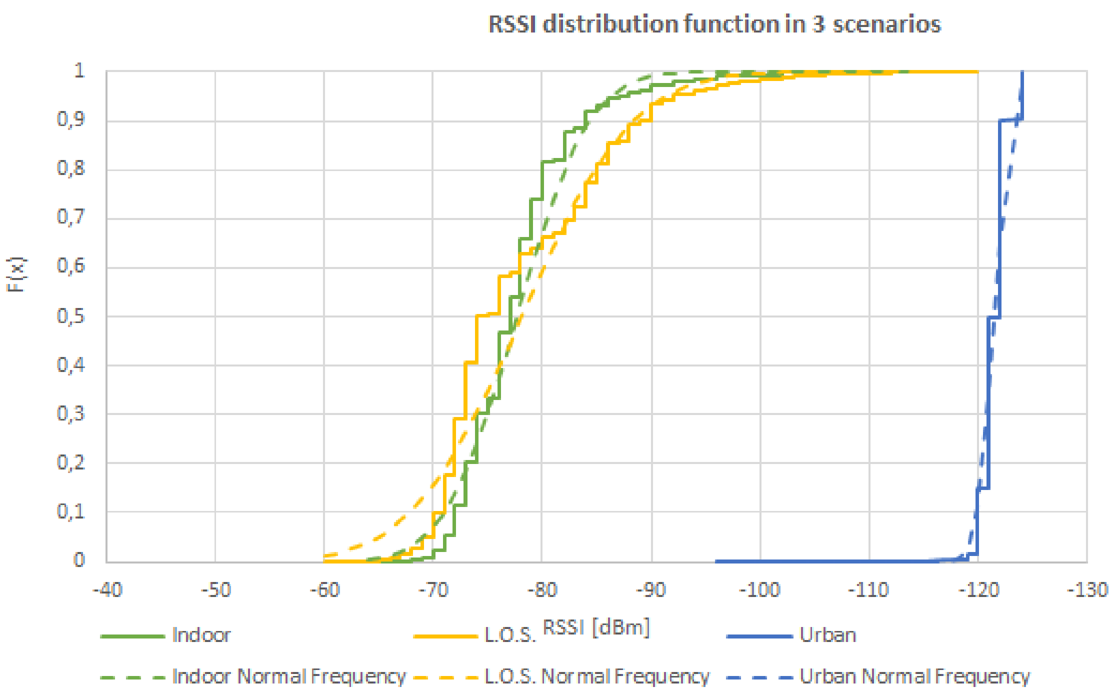

Figure 31: Cumulative Distribution Function for RSSI in 3 scenarios.

	Indoor	**L.O.S.**	**Urban**
Number of samples received	21907	22917	2399

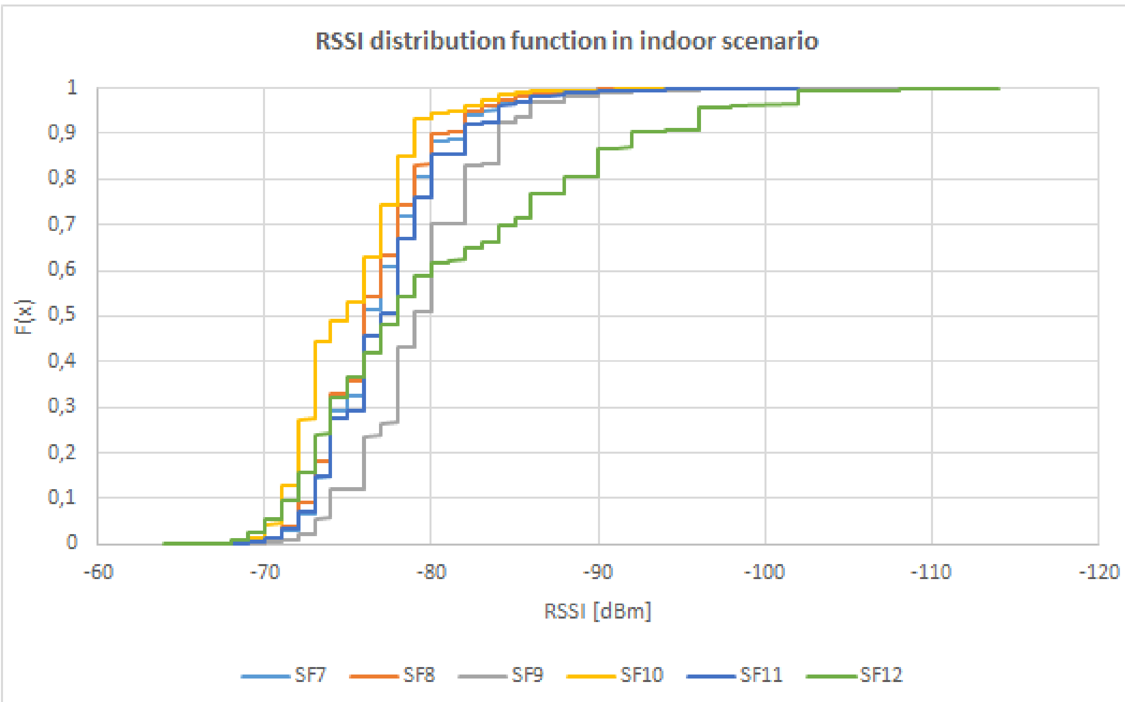

Figure 32: Cumulative Distribution Function for RSSI in indoor scenario.

	SF7	SF8	SF9	SF10	SF11	SF12
Number of samples received	3653	3660	3652	3639	3661	3642

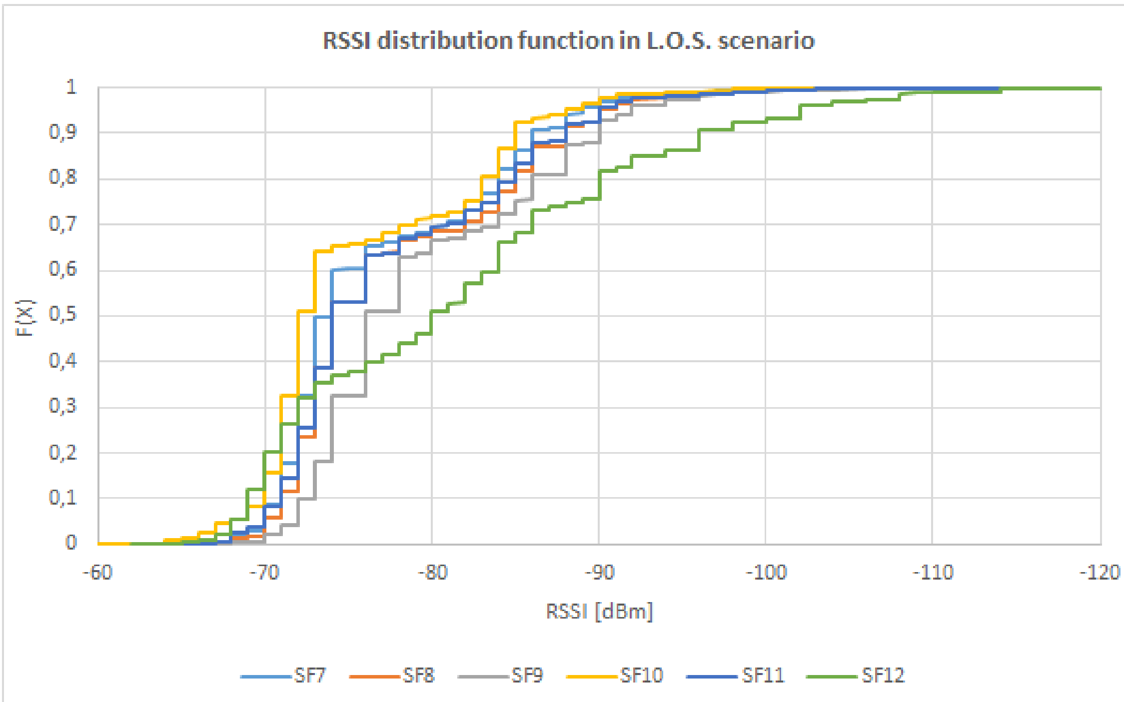

Figure 33: Cumulative Distribution Function for RSSI in L.O.S. scenario.

	SF7	**SF8**	**SF9**	**SF10**	**SF11**	**SF12**
Number of samples received	3826	3824	3817	3814	3835	3801

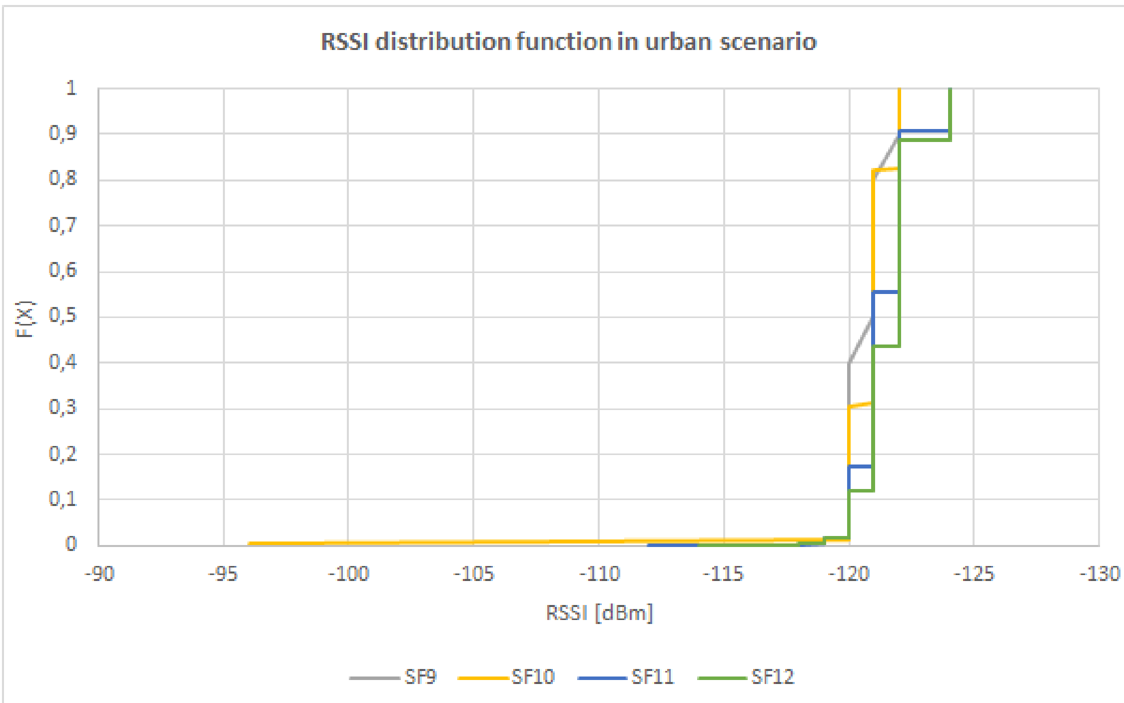

Figure 34: Cumulative Distribution Function for RSSI in urban scenario.

	SF7	SF8	SF9	SF10	SF11	SF12
Number of samples received	-	-	10	151	776	1462

7.1.2 SNR for channel and spreading factor

The results obtained with the SNR are completely unexpected. We expected to obtain best Signal-to-Noise Ratio on those Spreading Factors with the best RSSI values as the SNR definition denotes, but the results do not show this idea.

Interpreting the graphs the first thing that we can appreciate is that there are significant differences between the channels, thing that we can not appreciate on the previous RSSI graphs. This may be caused by interferences from other users that are using the same spectral band. We have to recall that we are using a public band that can be used by everyone who accomplish the rules established. Looking channel 5 on Figure 35 we also see that this channel had the worst performance in indoor and L.O.S. scenarios but in the urban one, we have obtained nearly the best performance on terms of SNR. This shows the changeable that can be these channels as we will see on the next sections.

Another interesting point that we can see on the 3 scenarios looking Figures 36, 37 and 38, is the "Gaussian" form of the SNR in relation to the Spreading Factors. The 3 scenarios show a better performance using the middle Spreading Factors than the fastest or the slowest. Even more if we compare the SNR with the RSSI obtained, looking the SNR values for SF9 we see that it has the best performance in terms of SNR, this may lead us to think that it is because it had been received with the greatest RSSI, but comparing with the previous graphs we see that is completely the opposite. SF9 is the second worst Spreading Factor in terms of RSSI.

Also we see the ability of the receiver, in this case the gateway, to demodulate correctly signals below the noise floor (negative SNR) one of the strong point of LoRa^{TM} modulation. We see this on Figures 35 and 38 where our signal is up to 19 dB below the noise floor.

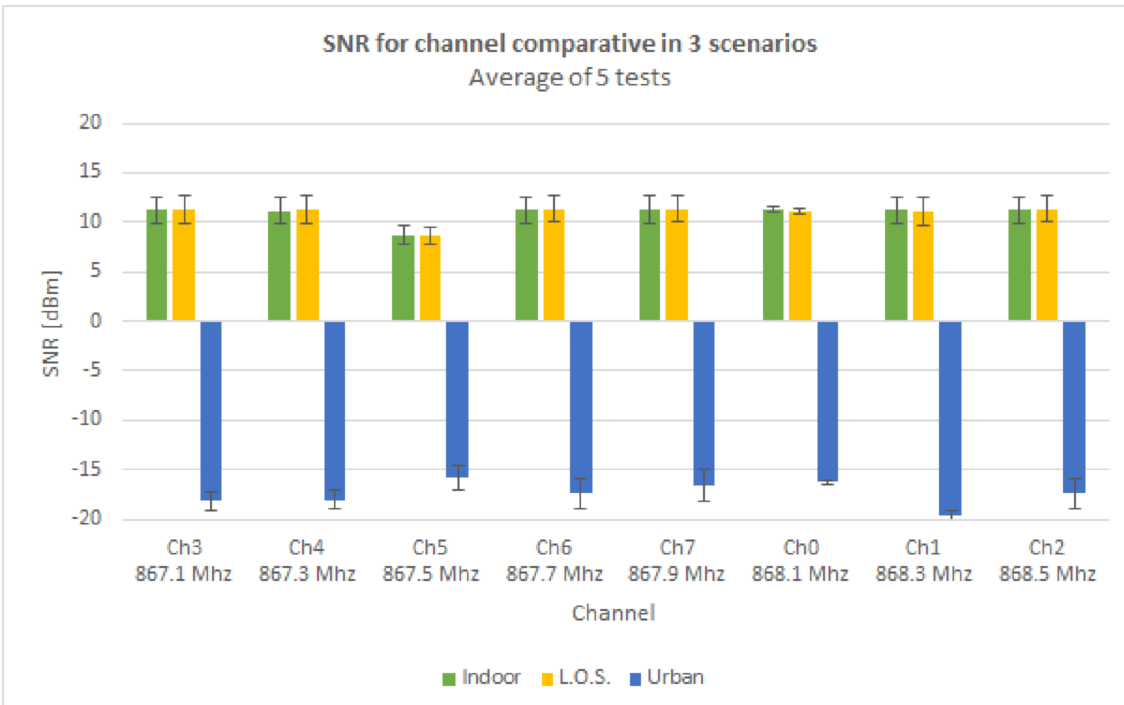

Figure 35: RSSI for channel in 3 scenarios.

Repetitions: 5	**Number of packets send per repetition:** 4800
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor, L.O.S., urban	

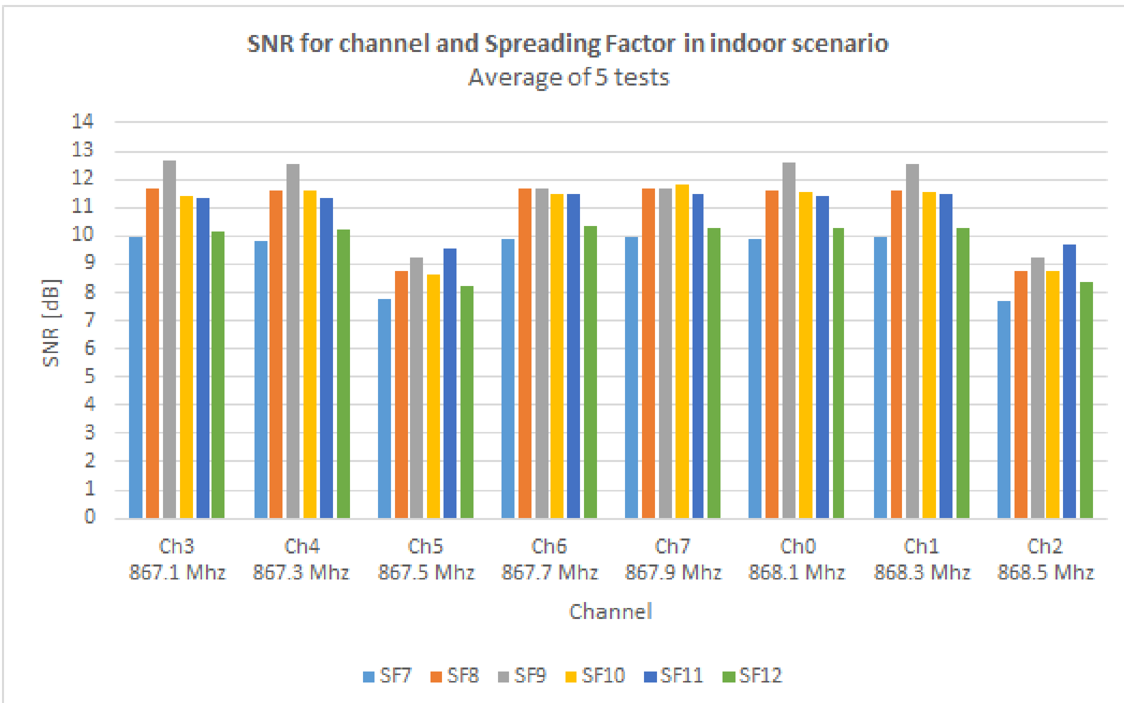

Figure 36: RSSI for channel and spreading factor in indoor scenario.

Repetitions: 5	**Number of packets send per repetition:** 4800
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor	

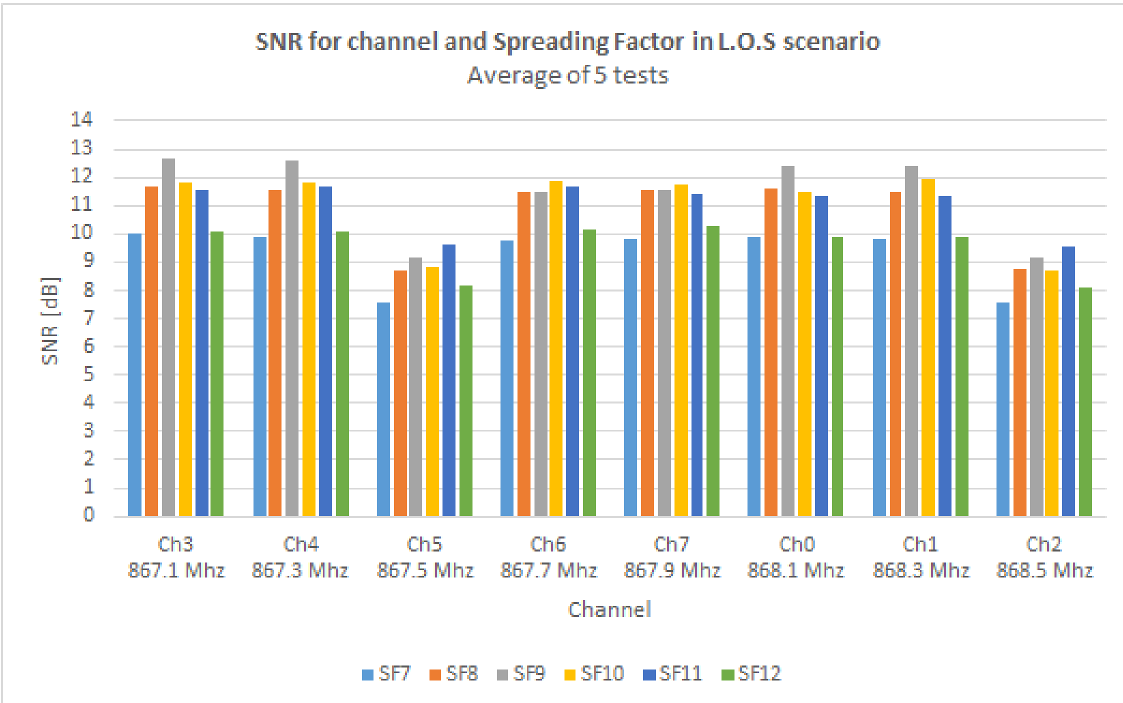

Figure 37: RSSI for channel and spreading factor in L.O.S. scenario.

Repetitions: 5	**Number of packets send per repetition:** 4800
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: L.O.S.	

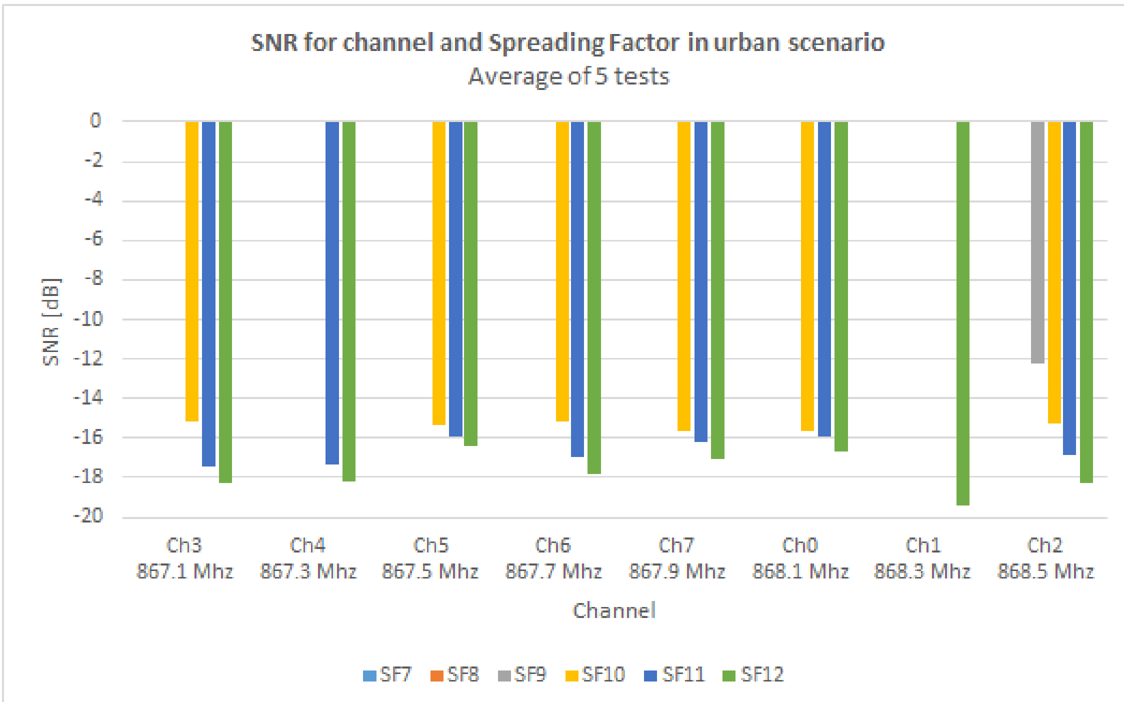

Figure 38: RSSI for channel and spreading factor in urban scenario.

Repetitions: 5	**Number of packets send per repetition:** 4800
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: urban	

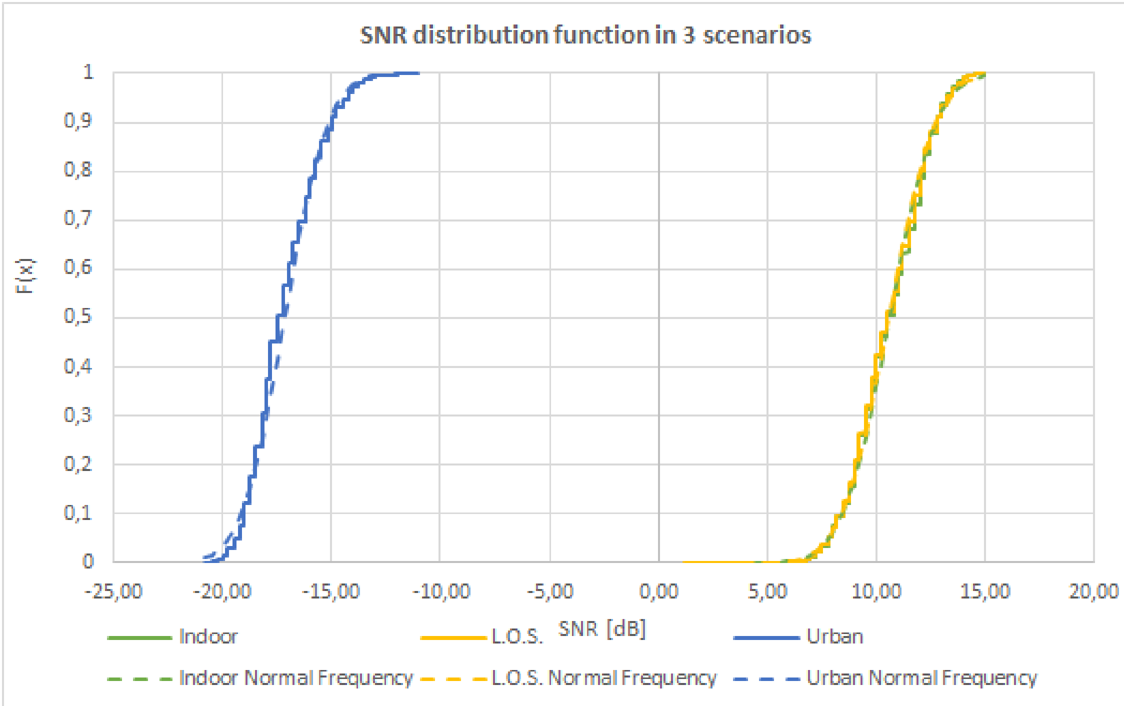

Figure 39: Cumulative Distribution Function for SNR in 3 scenarios.

	Indoor	**L.O.S.**	**Urban**
Number of samples	21907	22917	2399

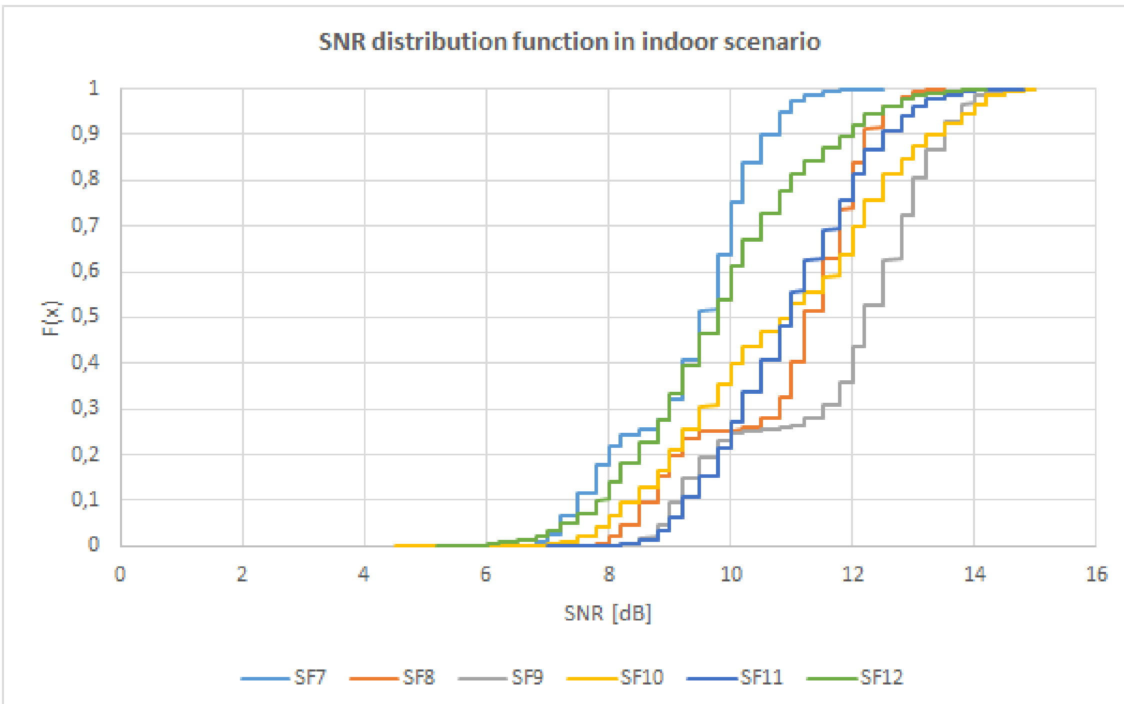

Figure 40: Cumulative Distribution Function for SNR in indoor scenario.

	SF7	SF8	SF9	SF10	SF11	SF12
Number of samples	3653	3660	3652	3639	3661	3642

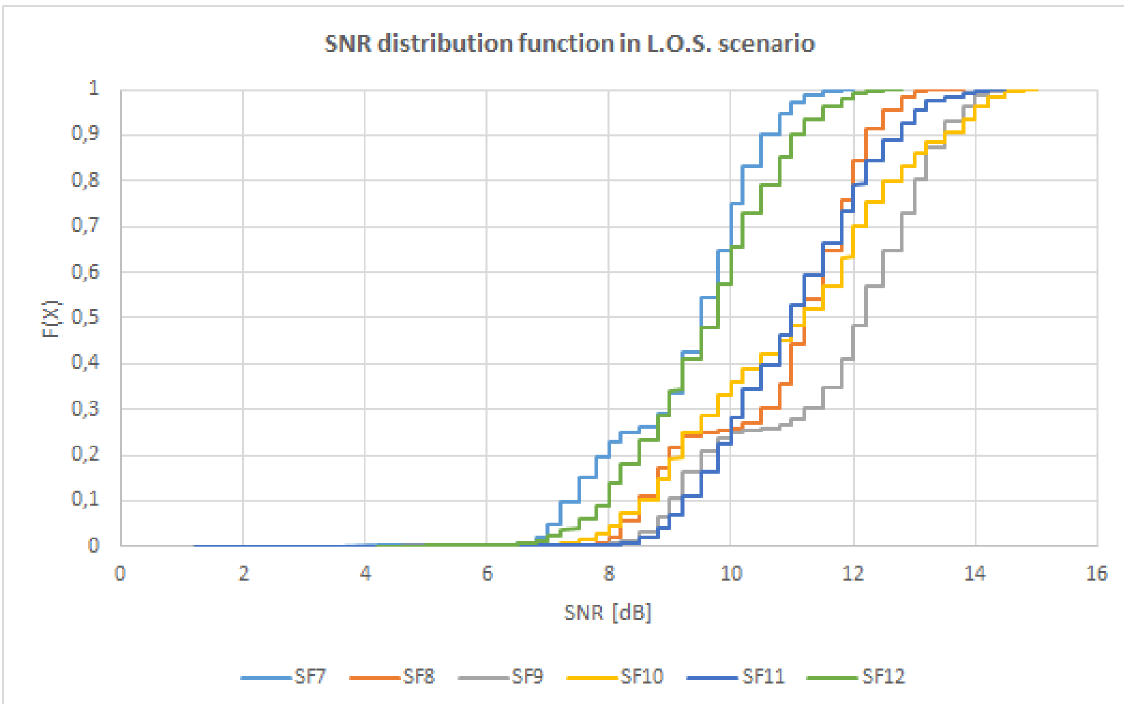

Figure 41: Cumulative Distribution Function for SNR in L.O.S. scenario.

	SF7	SF8	SF9	SF10	SF11	SF12
Number of samples	3826	3824	3817	3814	3835	3801

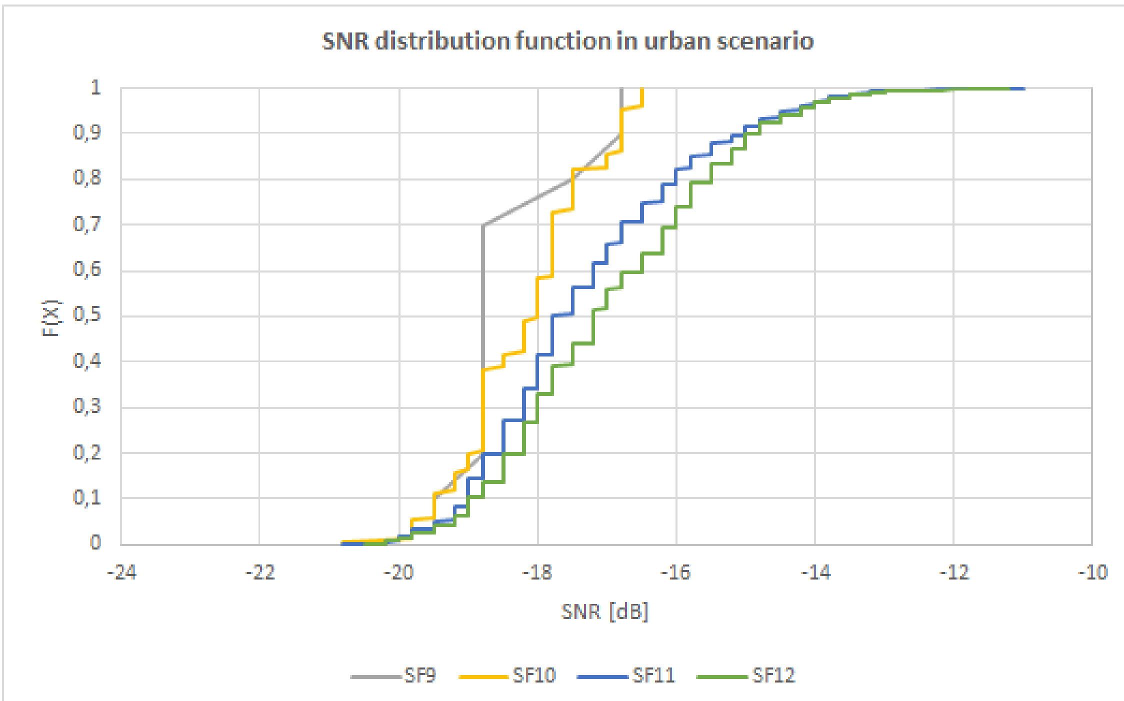

Figure 42: Cumulative Distribution Function for SNR in urban scenario.

	SF7	SF8	SF9	SF10	SF11	SF12
Number of samples	-	-	10	151	776	1462

7.1.3 PER for channel and spreading factor

Looking Figure 43 we see similar values in the indoor and L.O.S. scenarios with PER values below the 2% in all the channels used. Looking the previous graphs of RSSI and SNR we could deduce this due to its similarities performances in terms of RSSI and SNR.
However if we see its individual results on Figure 44 and 45 we see very different performance comparing the Spreading Factors between channels and it not seems to be similitude between these 2 scenarios. For example on Figure 44 SF7 has the worst PER on channel 4 but has the best PER on channel 1. Whereas on L.O.S. scenario shown on Figure 45 we appreciate an opposite performance. However the differences in PER between the different Spreading Factors do not exceed the 2%, so we are talking about small variations.
On Figure 46 that shows the PER in the urban scenario we see again the utility of the lowest data rate (biggest Spreading Factors) that allows to the signals to arrive far away. As we saw on the RSSI and SNR graphs, in urban scenario only the packets sent at SF10, SF11 and SF12 reached the gateway, with difference between SF10 and SF12 that sometimes arrive at the 50% as we see on channels 3, 6 and 7.
Another detail that we see looking Table 52 is the big differences that we can observe between the different repetition of the tests. For example looking the values in the indoor scenario we see a big difference between the test 4 and the others. The same happens on L.O.S. scenario looking test 2 results.
This was one of the reasons that led us to realise the tests shown on Section 7.1.4.

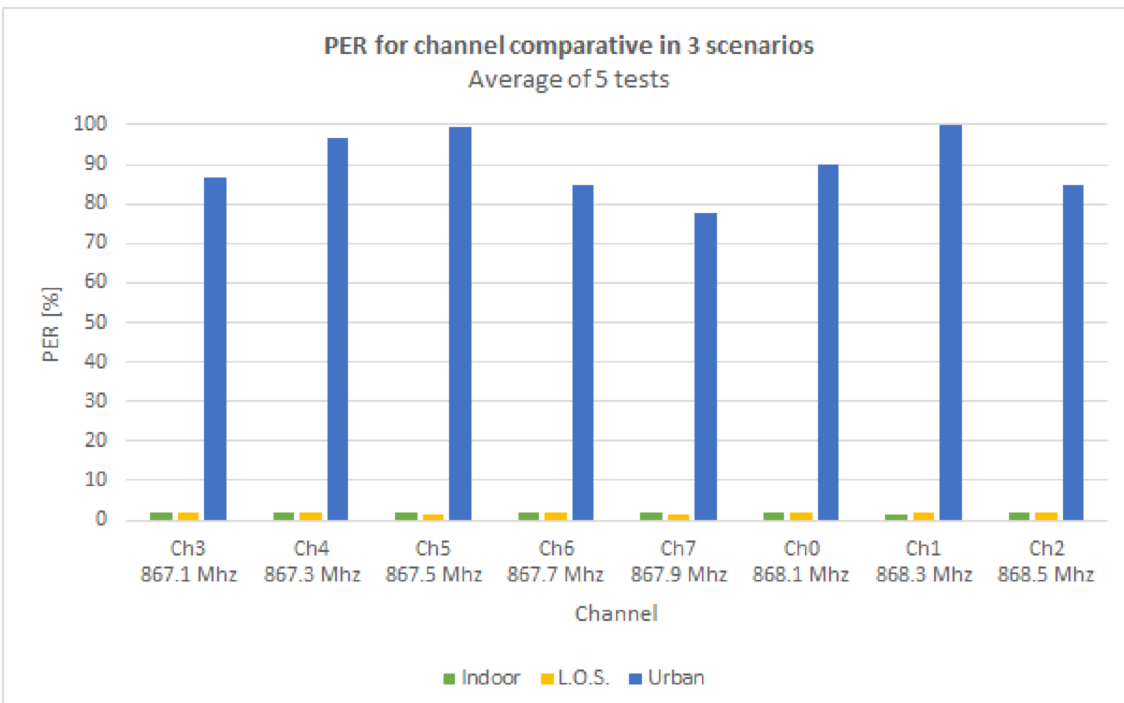

Figure 43: PER for channel in indoor scenario.

Repetitions: 5	**Number of packets send per repetition:** 4800
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor, L.O.S., urban	

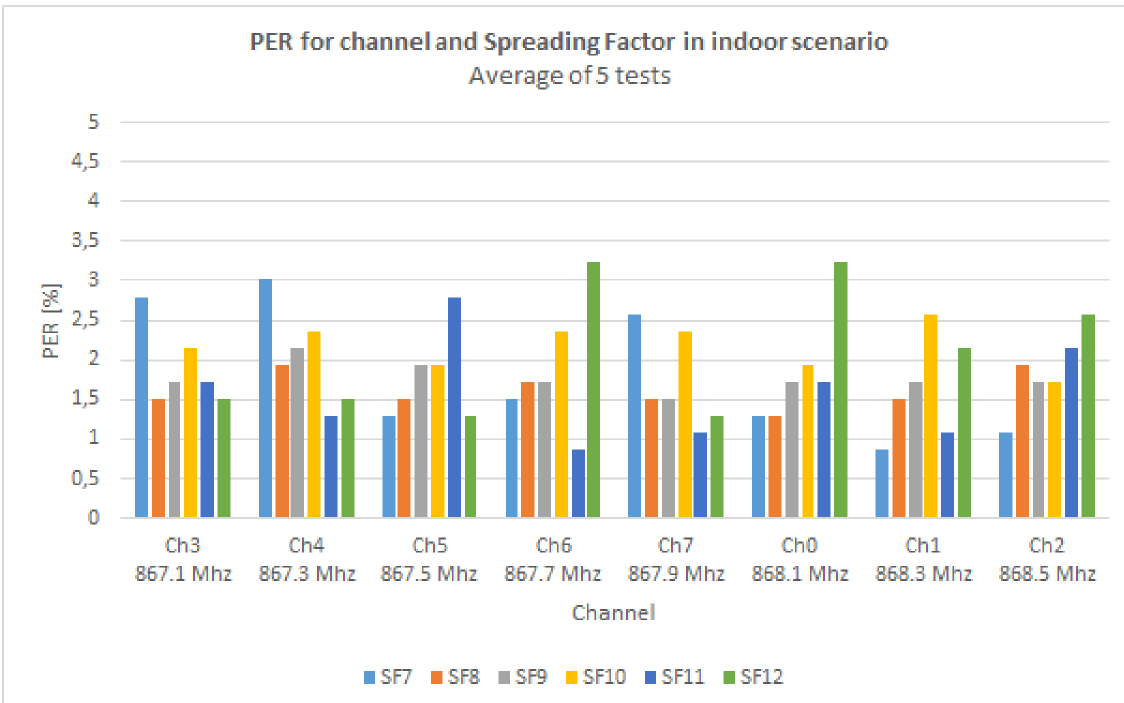

Figure 44: PER for channel and spreading factor in indoor scenario.

Repetitions: 5	**Number of packets send per repetition:** 4800
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor	

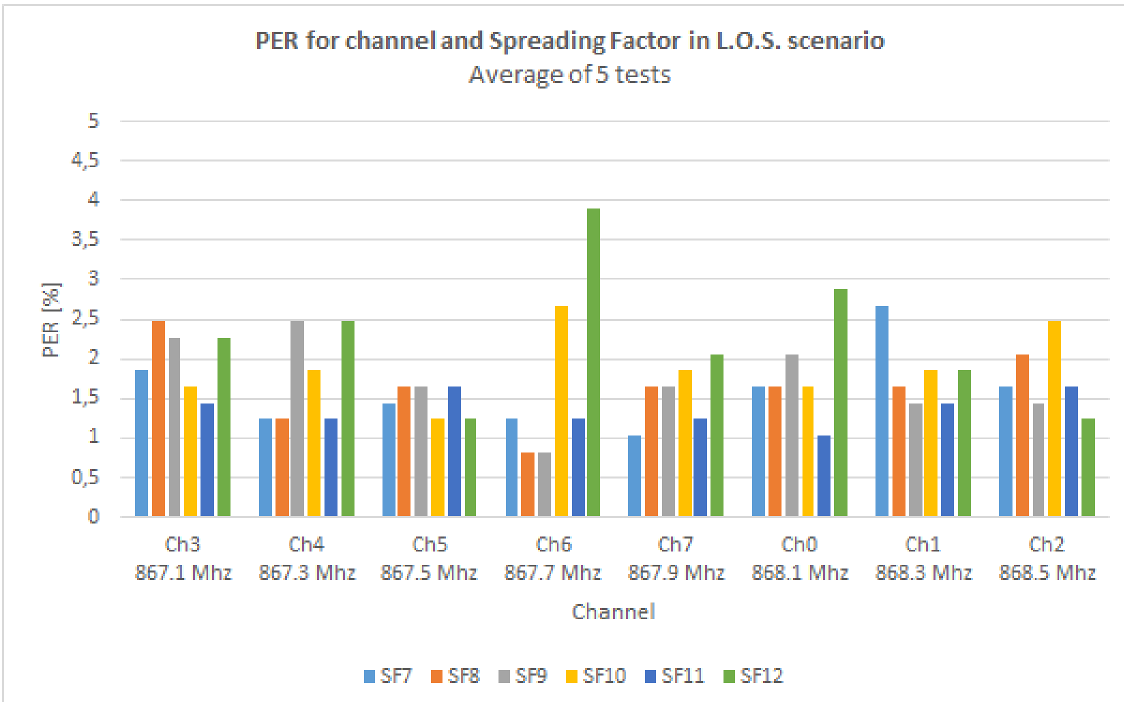

Figure 45: PER for channel and spreading factor in L.O.S. scenario.

Repetitions: 5	**Number of packets send per repetition:** 4800
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: L.O.S.	

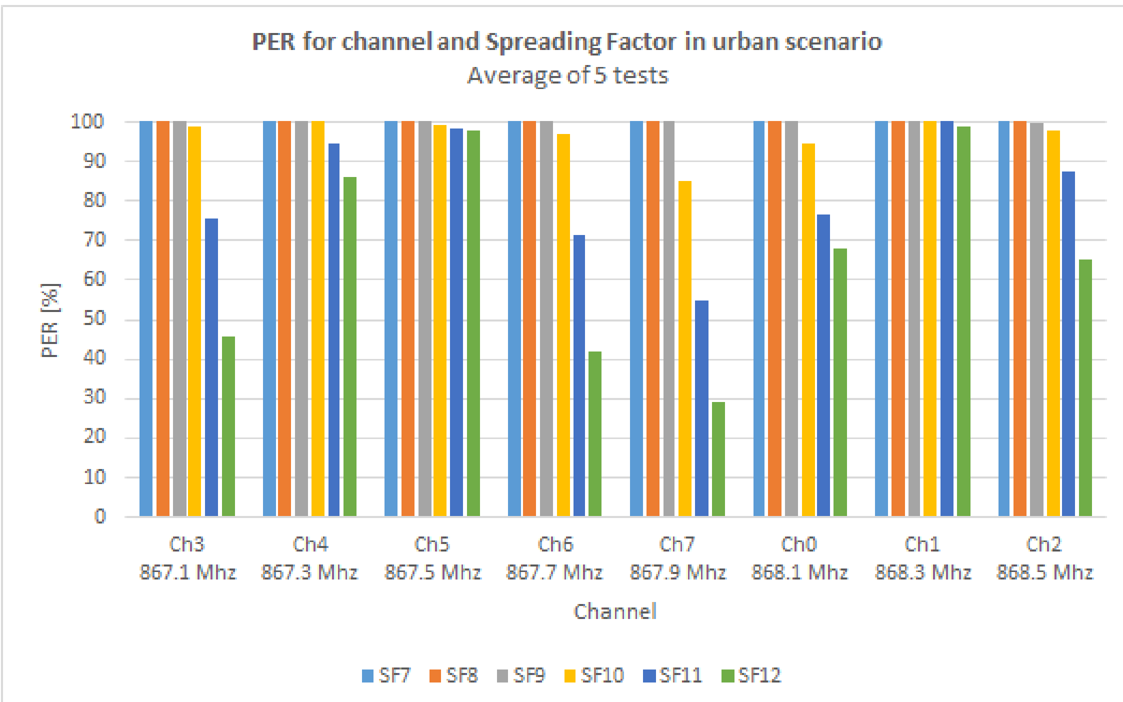

Figure 46: PER for channel and spreading factor in urban scenario.

Repetitions: 5	**Number of packets send per repetition:** 4800
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: urban	

7.1.4 PER in time

We want to study in depth the cause of the lost packets. For this propose we placed the LoRa™ Motes sending packets every 6 seconds during one week to see if there is a pattern in the lost packets, differences because of the hour of the day or if they are caused for collisions.
On the following figures we see the number of packets lost during a whole week (from Monday to Sunday). We can appreciate a lot of isolated lost packets during the day, but also we see time intervals where we have lost a lot of consecutive packets, from the order of 20, 30 or 40. Having in mind the 6 seconds between packets, this mean that it has been intervals of 2, 3 and 4 minutes where all the packets have been lost continually.
At first sight we don't appreciate a significant patron on the packets lost nor regular periods of time where we loss more packets.
However we see and important decrease of lost packets during the weekend, this leads us to think that the losses can be caused by the collisions with other packets or due to the interferences with other signals. We have to remember that we are using the shared band from 867 to 869 MHz available for everybody. One example is Sigfox's technology [2] that also works on the same band of frequencies and is being tested around l'Isère with Capturs devices [10].

Table 25: PER in time test setup.

Start: 05/11/2015 18:37:04	**End:** 16/11/2015 12:17:07
Number of days: 7	**Number of packets send per day:** 14000
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor	

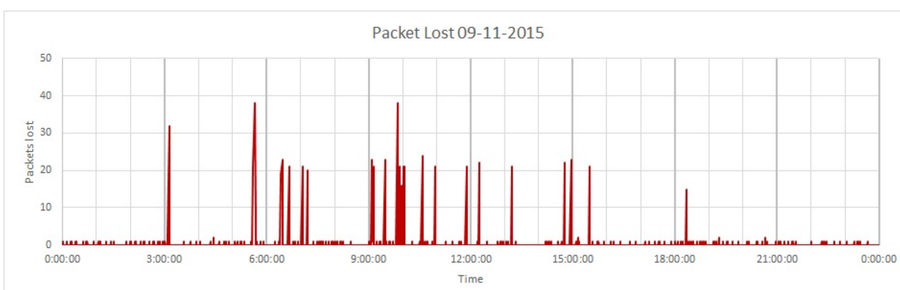

Figure 47: Packets lost on 09-11-2015.

[10] http://www.capturs.com/nos-testeurs/

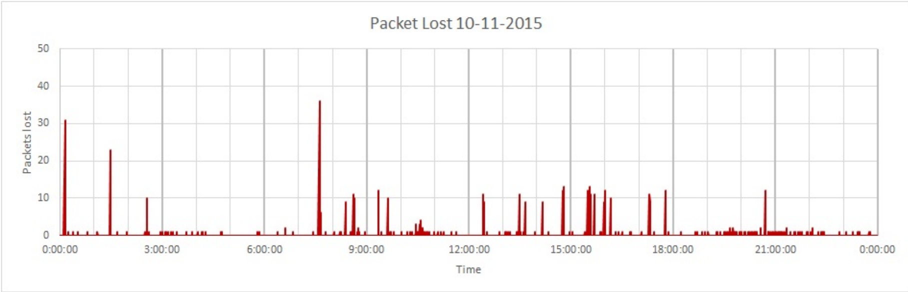

Figure 48: Packets lost on 10-11-2015.

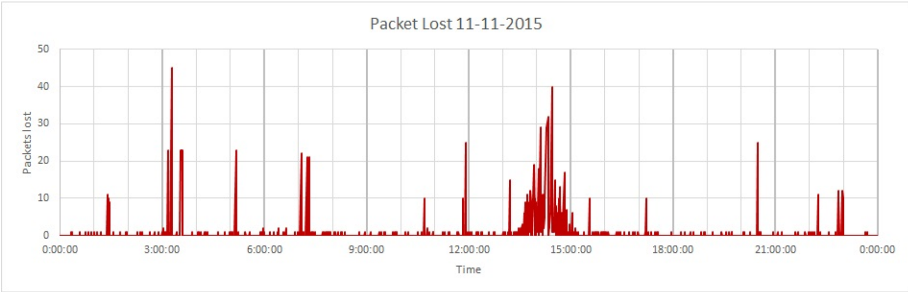

Figure 49: Packets lost on 11-11-2015.

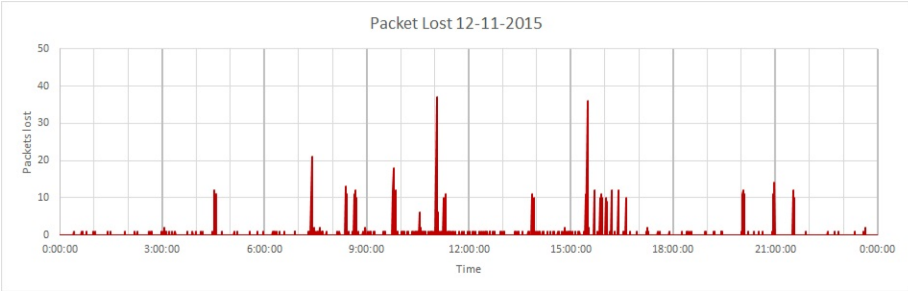

Figure 50: Packets lost on 12-11-2015.

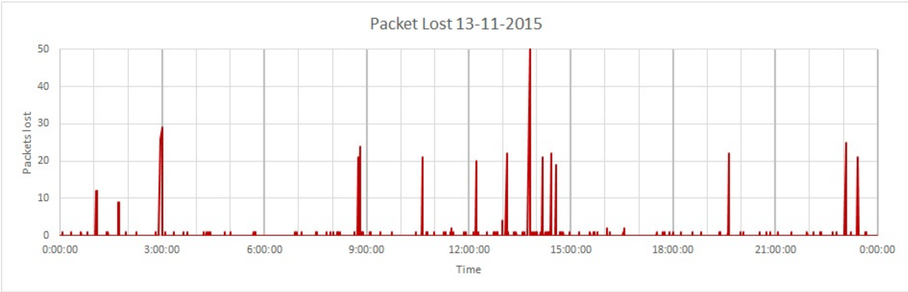

Figure 51: Packets lost on 13-11-2015.

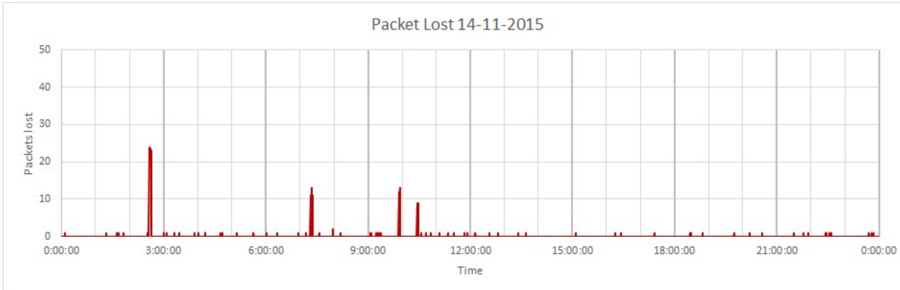

Figure 52: Packets lost on 14-11-2015.

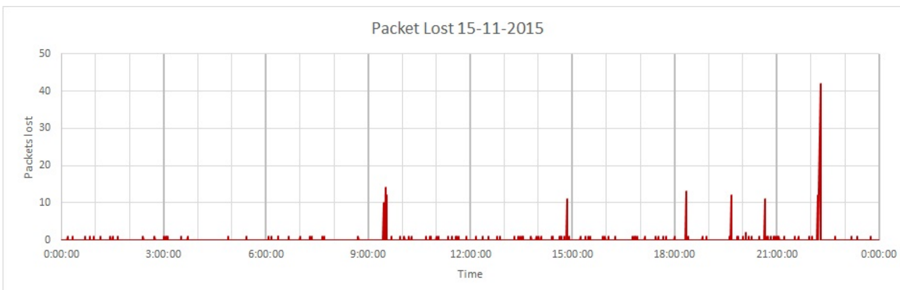

Figure 53: Packets lost on 15-11-2015.

If we compare days between weeks we see that the cause of lost packets are not deterministic (Figures 54, 55, 56 and 57). However we verify again the differences between days on weekdays and on weekend, this reaffirm the possibility that these losses are caused by interferences on the channels.

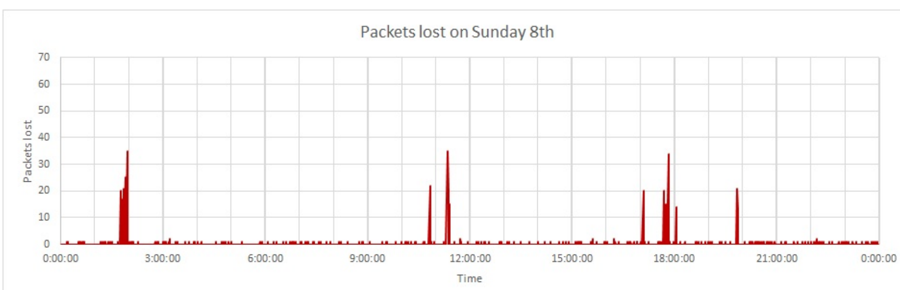

Figure 54: Packets lost on Sunday 8th.

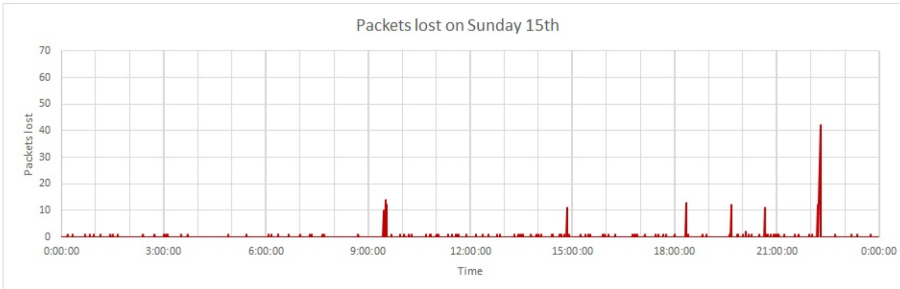

Figure 55: Packets lost on Sunday 15th.

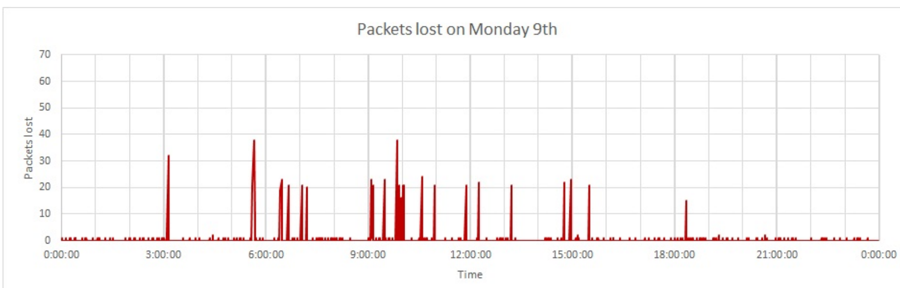

Figure 56: Packets lost on Monday 9th.

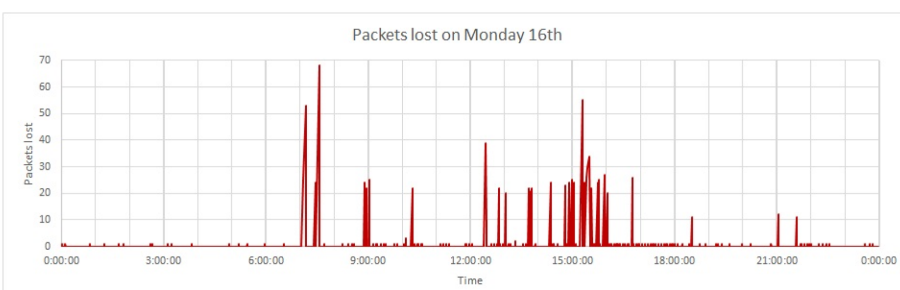

Figure 57: Packets lost on Monday 16th.

7.2 PER for payload

We want to see if the payload length affects to the PER on the communication, a longer message will increment the Time On Air of the packet incrementing the probabilities of been affected by interferences. For this propose we modified class A firmware to send messages with the following payload length in bytes: 4, 8, 12, 16, 24, 32, 51, 80, 115, 164 and 222.
According with the maximum MACPayload size length derived from the limitation of the PHY layer depending on the effective modulation rate used taking into account a possible repeater encapsulation layer. The maximum application data are showed on Table 26.
As the iot.semtech.com demonstration web page only show the first 64 bytes of payload received we modified the firmware to include in the first byte of payload the payload length of the FRM Payload field.

Table 26: Maximum application payload length

Spreading Factor	**Application Payload**
SF12	51
SF11	51
SF10	51
SF9	115
SF8	222
SF7	222

Looking Figure 58 we see a slight rise of the PER with the payload length, incident that we see very clear on the urban scenario. By the other hand on indoor and L.O.S. scenarios this incident is not so clear considering the nice PER that we obtained in the case of 222 Bytes of payload.

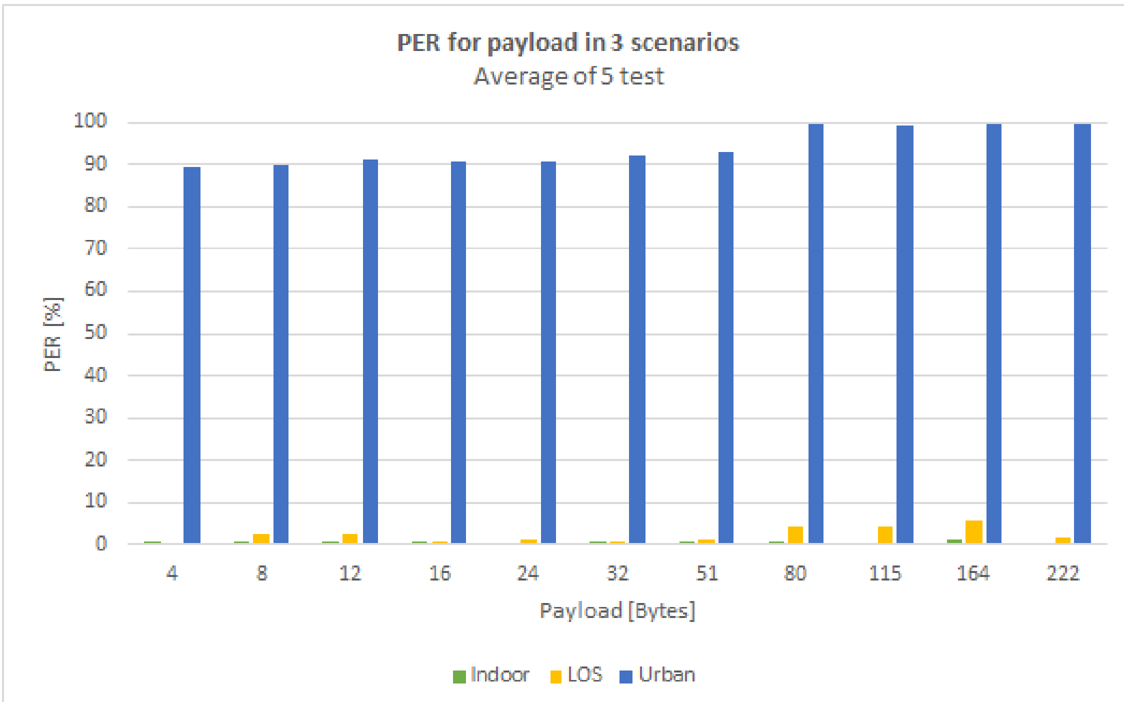

Figure 58: RSSI for channel in indoor scenario

Repetitions: 5	**Number of packets send per repetition:** 4160
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: cyclic	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor, L.O.S., urban	

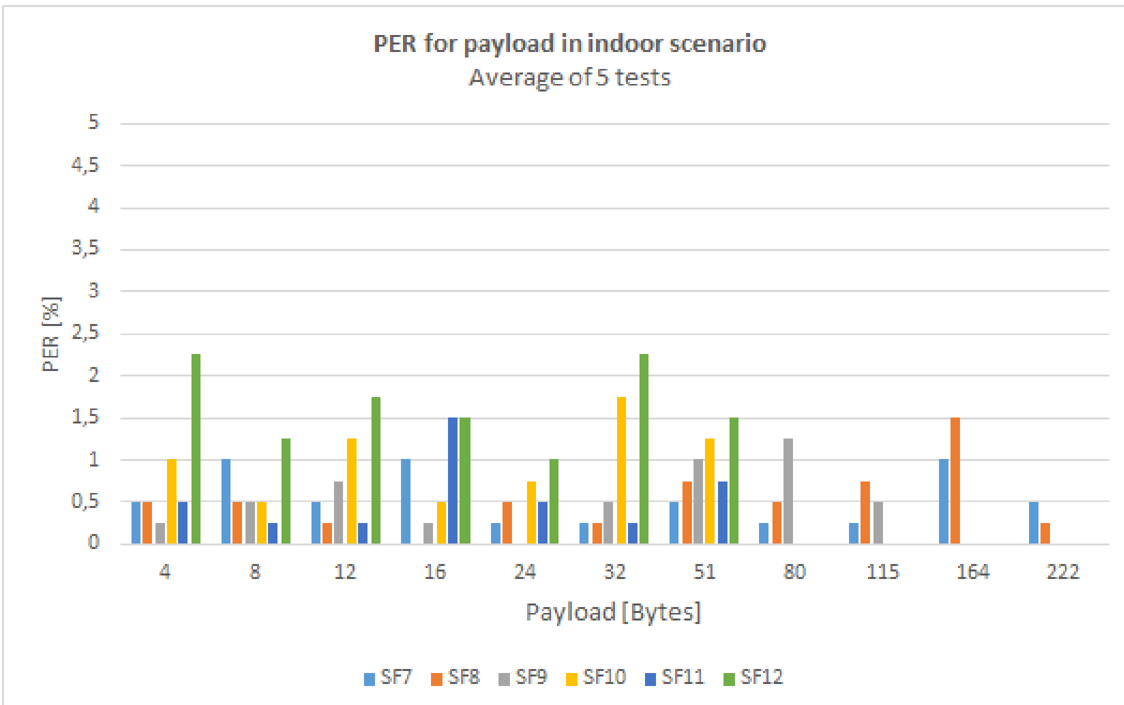

Figure 59: RSSI for channel and spreading factor in indoor scenario.

Repetitions: 5	**Number of packets send per repetition:** 4160
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: cyclic	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor	

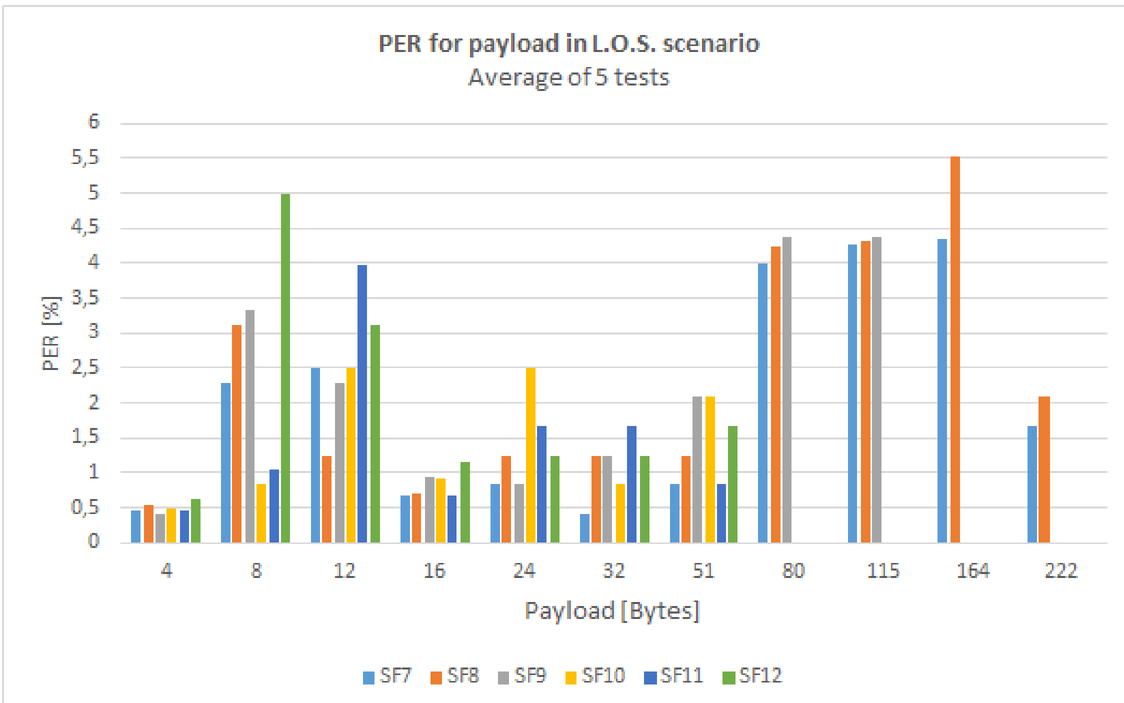

Figure 60: RSSI for channel and spreading factor in urban scenario

Repetitions: 5	**Number of packets send per repetition:** 4160
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: cyclic	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: L.O.S.	

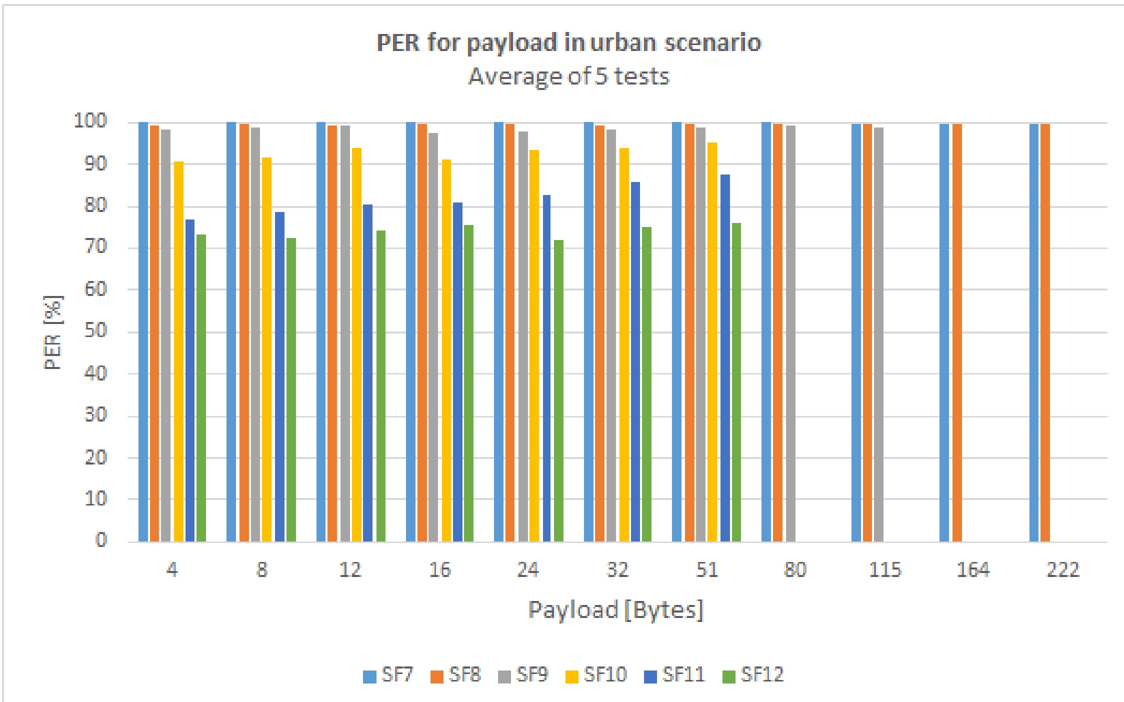

Figure 61: RSSI for channel and spreading factor in urban scenario.

Repetitions: 5	**Number of packets send per repetition:** 4160
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: cyclic	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: urban	

7.3 PER for Cyclic Redundancy Check

We wanted to test the real performance gain of Cyclic Redundancy Check applied on to the PHY Payload field, however, this happens in the presence of bursts of interference. If the radio link is likely to be subject to such interference, the use of CRC should be evaluated.
On the tests we configure the firmware to send packets using the four available Coding Rates as we explained on Section 5.3.
Before the tests we hopped to obtain best PER results using higher Coding Rates, not really in the indoor or Line Of Sight scenarios, but yes on the urban where the influences of the interferences can be higher due to the obstacles and the longer distance that the signal has to travel.
The tests realised in indoor and Line Of Sight scenarios do not show a clearly improvement in terms of PER. However in urban scenario, we appreciate a slight improvement. Comparing the average results for the urban scenario we only see an improvement near to 3% with the use of the Coding Rate 4/8 instead the faster 4/5. On the other hand if we watch the individual performance of the different Spreading Factors we notice a improvement of 4.3% for SF10, 2.2% for SF11 and 9,6% for SF12.
Results from the urban scenario 65 lead us to think that an increment on the Spreading Factor will have better performance in terms of PER than an increment on the CRC at the expense of an increment of the Time On Air of the packet. However when the Spreading Factor limit is reached it would be a good idea, increase the CRC to improve the performance. This solution is not allowed with the actual LoRaWAN^{TM} firmware and the available MAC Commands that only allow to modify the Spreading Factor, the power of the end-devices and the number of repetitions in the transmission of one frame.

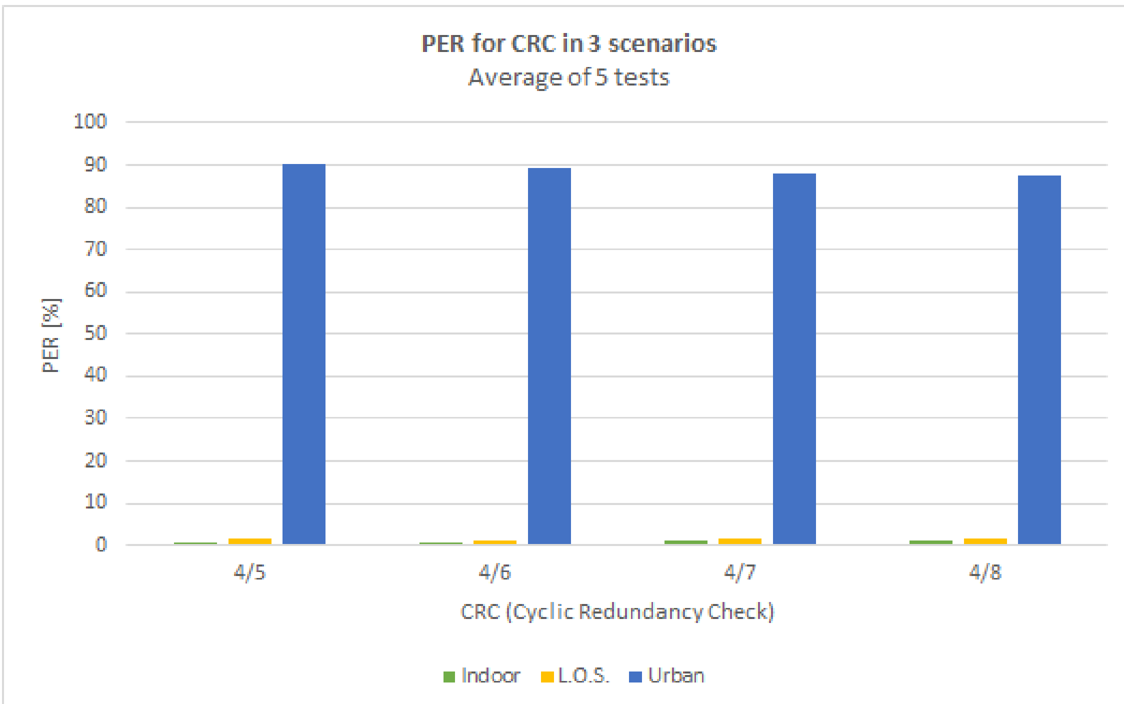

Figure 62: RSSI for channel in 3 scenarios.

Repetitions: 5	**Number of packets send per repetition:** 2304
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** cyclic
Scenario: indoor, L.O.S., urban	

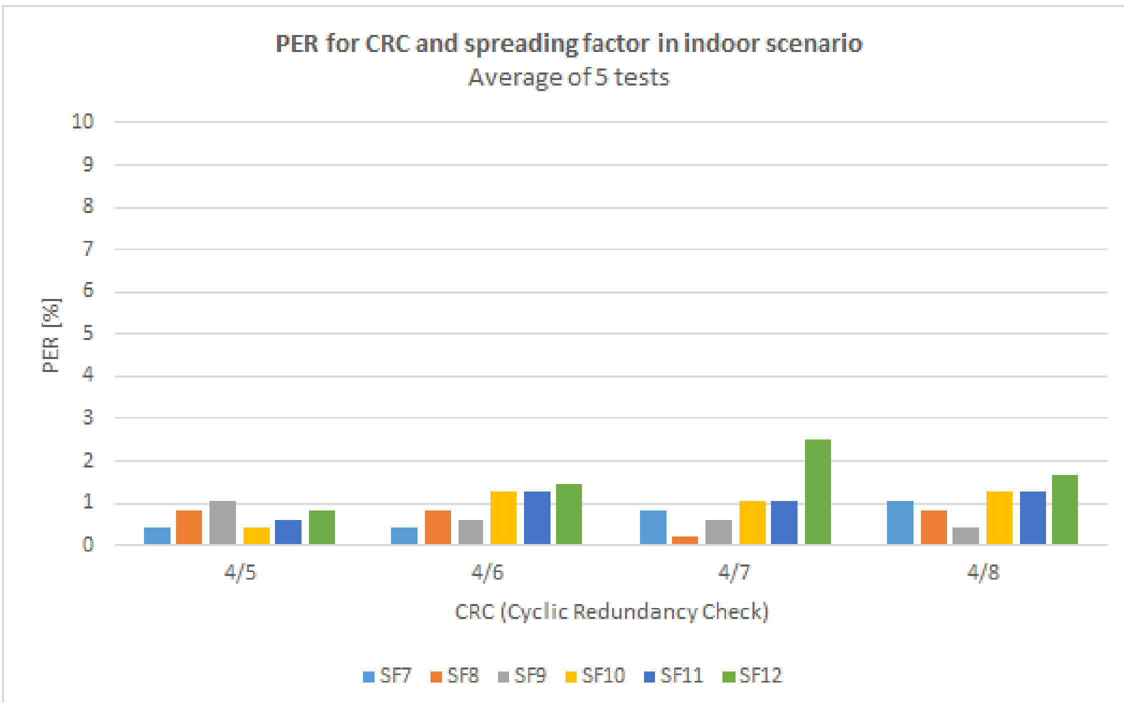

Figure 63: RSSI for channel and spreading factor in indoor scenario.

Repetitions: 5	**Number of packets send per repetition:** 2304
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** cyclic
Scenario: indoor	

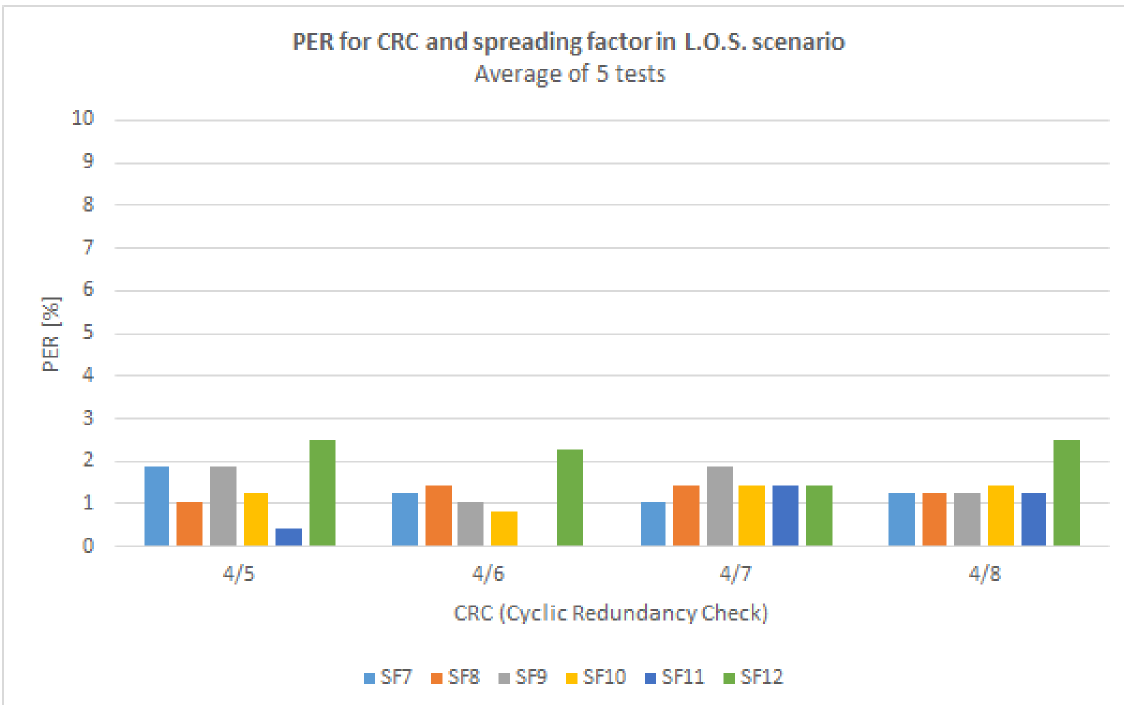

Figure 64: RSSI for channel and spreading factor in L.O.S. scenario.

Repetitions: 5	**Number of packets send per repetition:** 2304
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** cyclic
Scenario: L.O.S.	

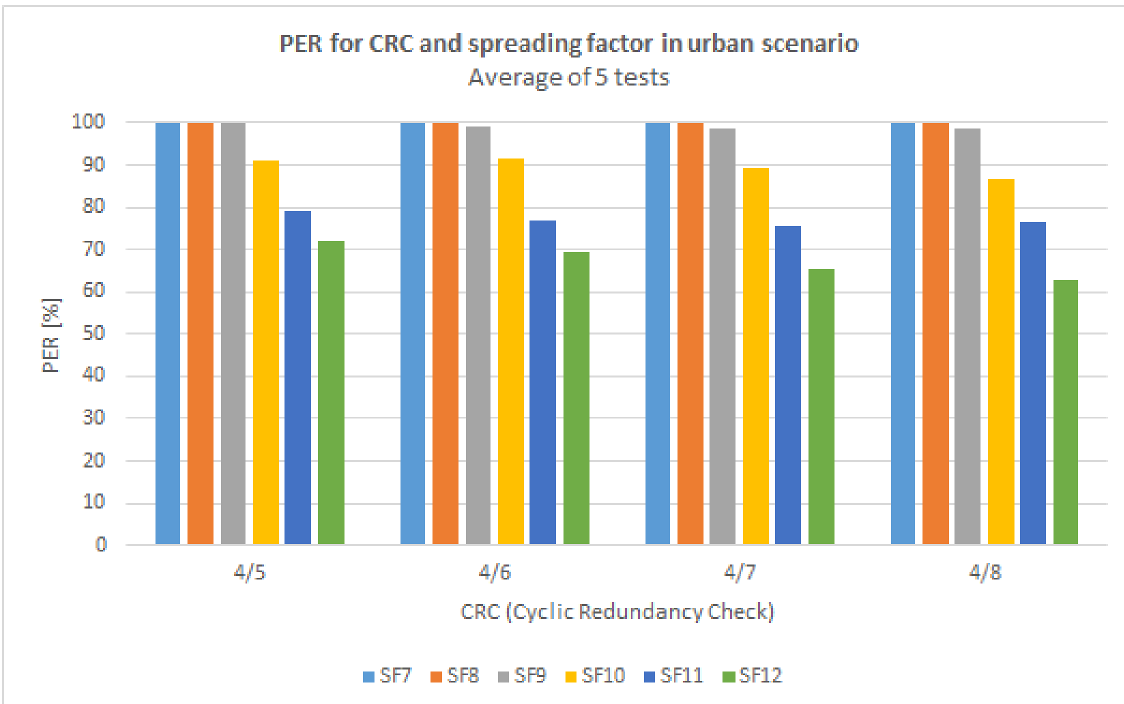

Figure 65: RSSI for channel and spreading factor in urban scenario.

Repetitions: 5	**Number of packets send per repetition:** 2304
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** cyclic
Scenario: urban	

7.4 PER for preamble length

The preamble symbols are used for synchronize transmission timing between two or more systems in our case between LoRa^{TM} Mote and the gateway. We can expect that a longer payload can reduce the PER of the comunication, but, up to what point?
To check how the preamble affects Packet Error Rate, we modified the firmware to send packets with: 5, 6, 8, 10, 12, 14 and 16 preamble symbols.
The results show that we expected, increasing the preamble we can reduce the Packet Error Rate. We can see a big fall in the PER increasing the preamble length in 1 symbol, from 5 to 6 symbols. In addition we see big differences between SF12 and the others Spreading Factors when we are using only 5 preamble symbols, the low data rate of SF12 seems to be cause.
As we have seen with the CRC tests, the improvement in terms of PER is more evident in the urban scenario, where the slowest Spreading Factors improve significantly with the increase of the preamble length. Looking Figure 69 and comparing the use of 5 symbols with 16 symbols, we see an improvement of 18.4% for SF11 and 10.6% for SF12.
On the other hand it does not seem that the increase of the preamble improve the PER with values of preamble bigger than 8 symbols.
Another interesting point, is see that in the urban scenario, the PER obtained with 16 symbols, CR of 4/5 and using SF12 (PER of 58.9%) is better than the obtained on the CRC tests (Section 7.3) with 8 symbols of preamble, CR of 4/8 and SF12 (PER of 62.7%). In this case the Time On Air of the packets is very similar 1.41 seconds and 1.44 seconds respectively.

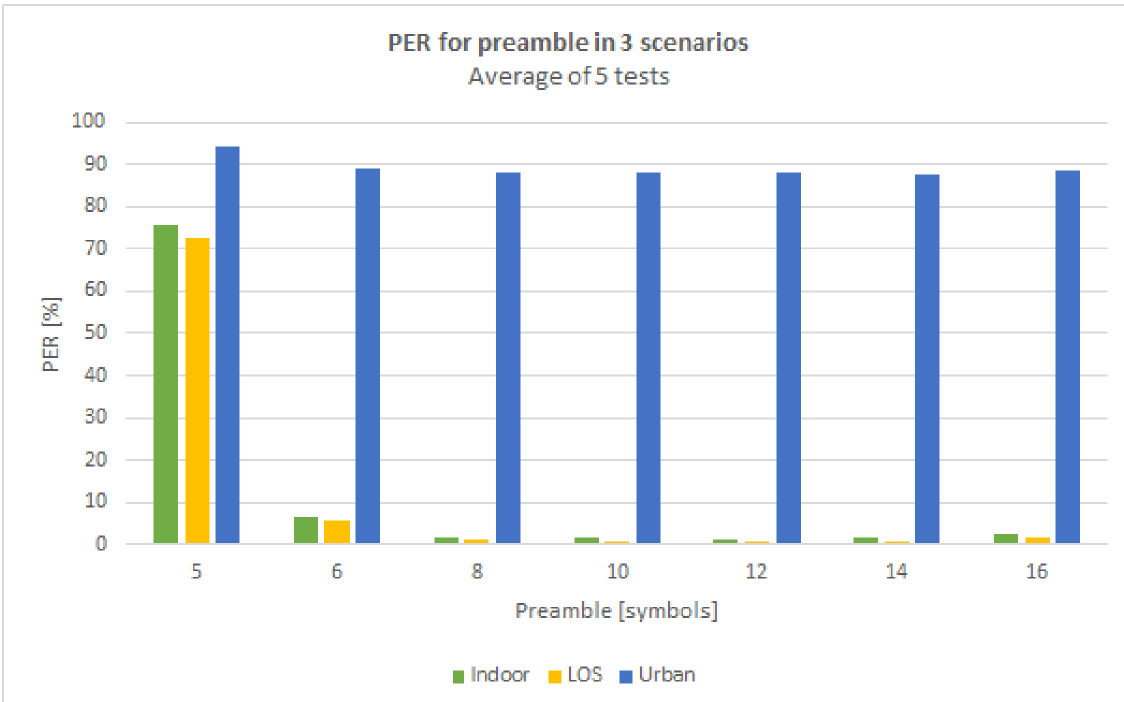

Figure 66: RSSI for channel in 3 scenarios.

Repetitions: 5	**Number of packets send per repetition:** 5376
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** cyclic
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor, L.O.S., urban	

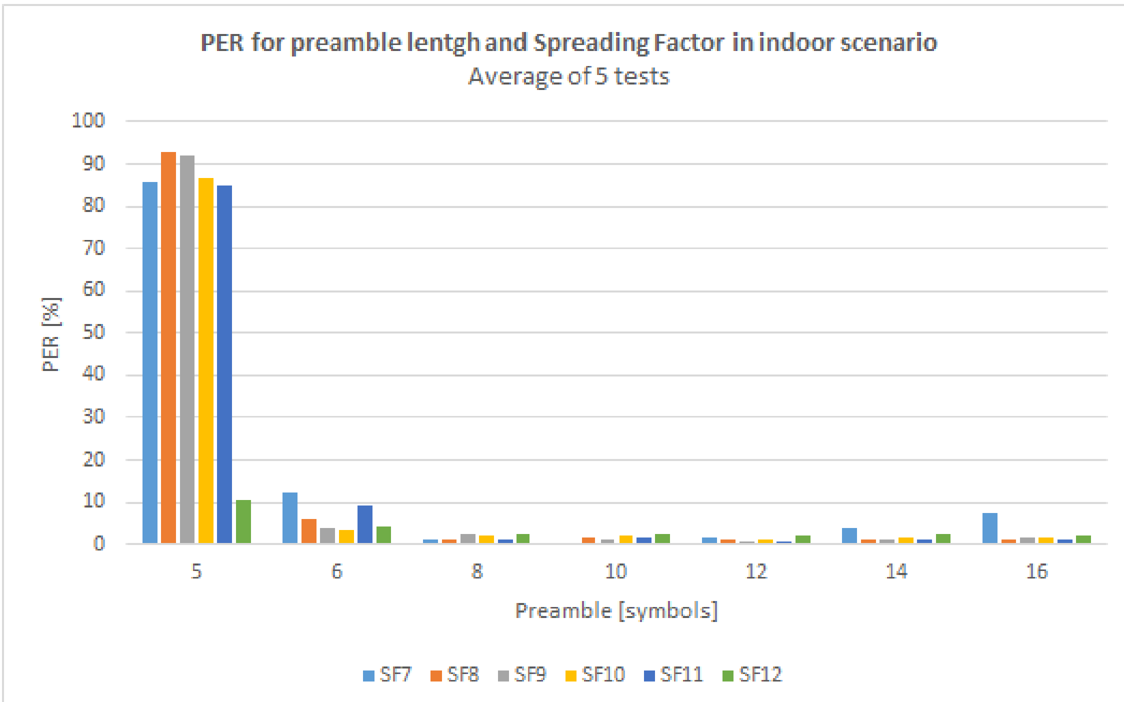

Figure 67: RSSI for channel and spreading factor in indoor scenario.

Repetitions: 5	**Number of packets send per repetition:** 5376
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** cyclic
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor	

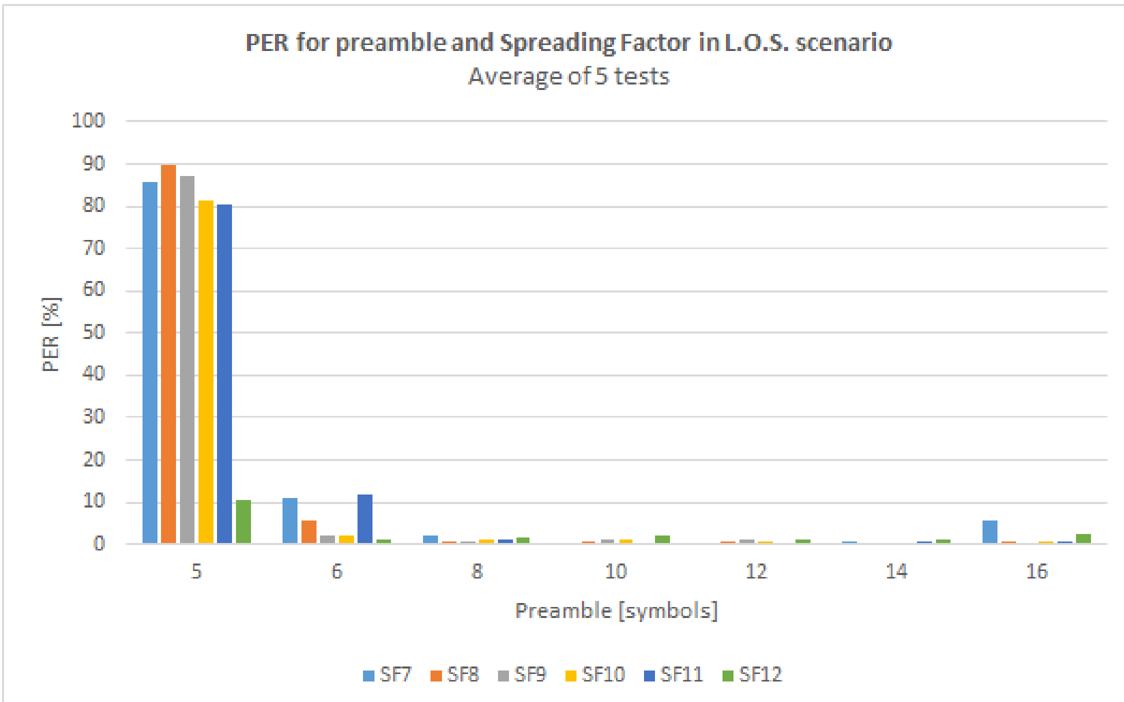

Figure 68: RSSI for channel and spreading factor in L.O.S. scenario.

Repetitions: 5	**Number of packets send per repetition:** 5376
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** cyclic
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: L.O.S.	

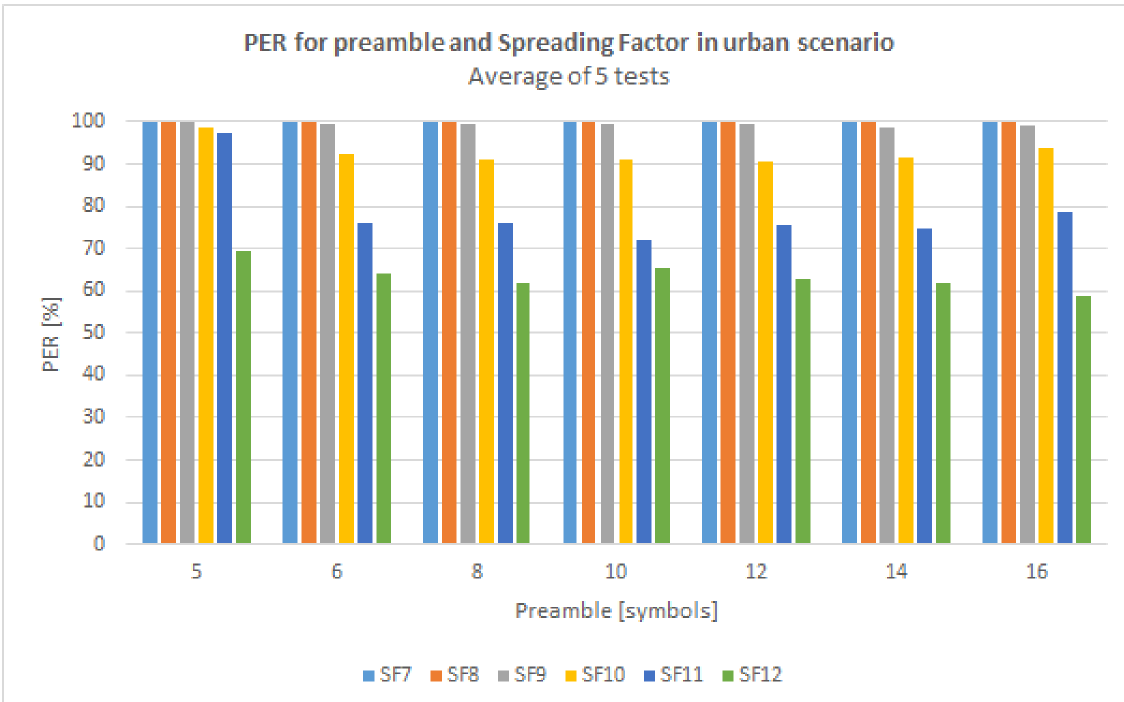

Figure 69: RSSI for channel and spreading factor in urban scenario.

Repetitions: 5	**Number of packets send per repetition:** 5376
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** cyclic
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: urban	

7.5 Indoor scenario

We tested LoRa^{TM} Motes in an indoor scenario to check if there is big differences between the Spreading Factors when they penetrate obstacles and check how the RSSI varies over them.
For these tests we place at the same time, the LoRa^{TM} Motes in 4 different floors of the ENSIMAG building (0, 1, 2 and 3), trying to place the LoRa^{TM} Motes in the same antenna's vertical and all them transmitting at 8 dBm.
The results show a nearly constant variation of the RSSI in function of the floor as we see on Figure 70, whereas the SNR keeps constant independently of the floor as we can see on Figure 72.
With the values shown on Table 57 on Annex E.6, we can calculate the attenuation due to the penetration of the signal through the floors of the building, supposing that the signal travels straight away to the antenna. This last supposition is not completely true, due to the antenna's design shown on LoRaMote USER GUIDE [20] we can not guarantee that the signal received at the gateway had only travel vertically.

$$\triangle RSSI_{3rd-2nd} = -8.6dB$$

$$\triangle RSSI_{2nd-1st} = -5.3dB$$

$$\triangle RSSI_{1st-0st} = -18.6dB$$

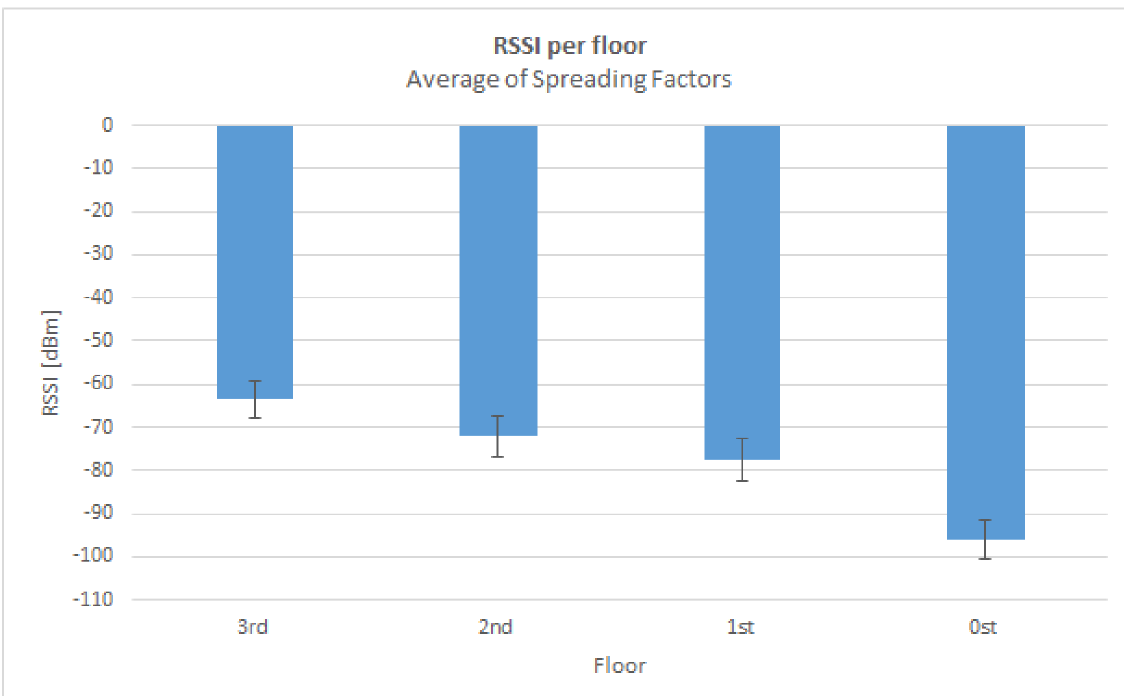

Figure 70: RSSI average in ENSIMAG floor's.

Repetitions: 1	**Number of packets send per repetition:** 2196
SF: cyclic	**Power:** 8 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor	

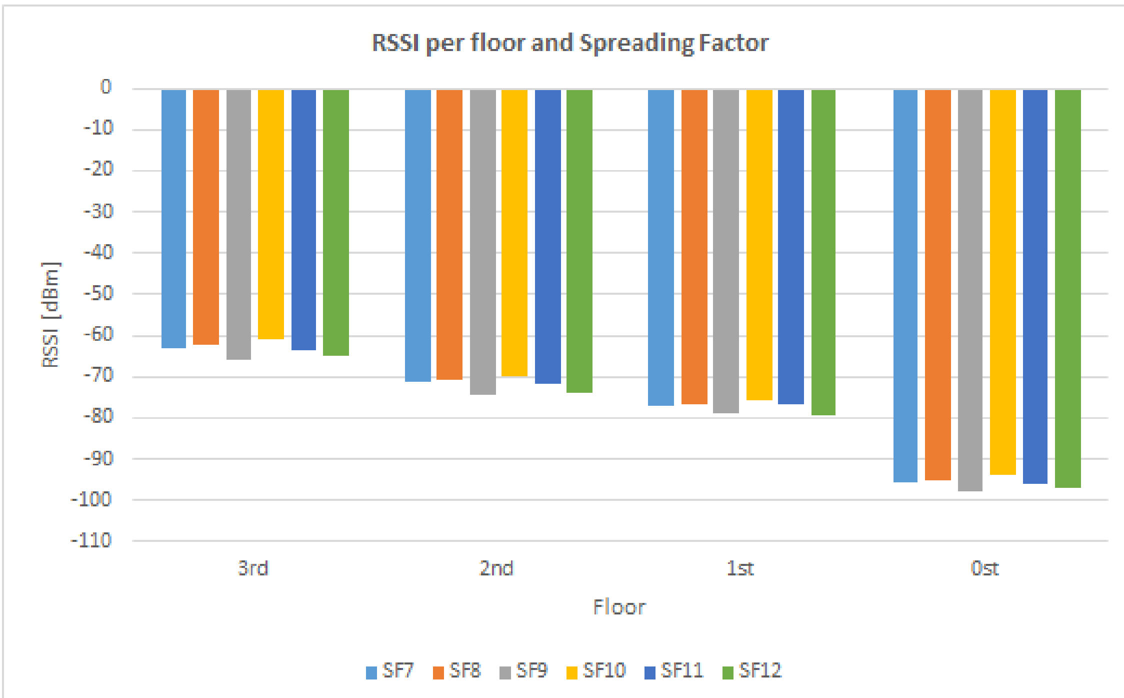

Figure 71: RSSI for Spreading Factors in ENSIMAG floor's.

Repetitions: 1	**Number of packets send per repetition:** 2196
SF: cyclic	**Power:** 8 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor	

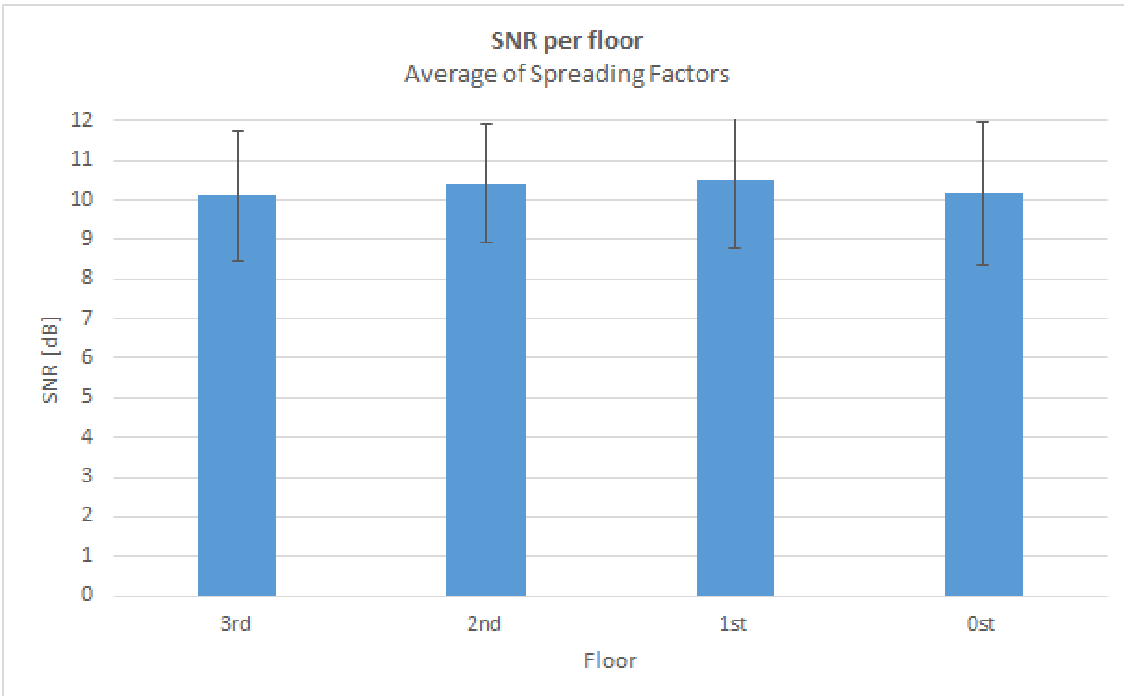

Figure 72: SNR average in ENSIMAG floor's.

Repetitions: 1	**Number of packets send per repetition:** 2196
SF: cyclic	**Power:** 8 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor	

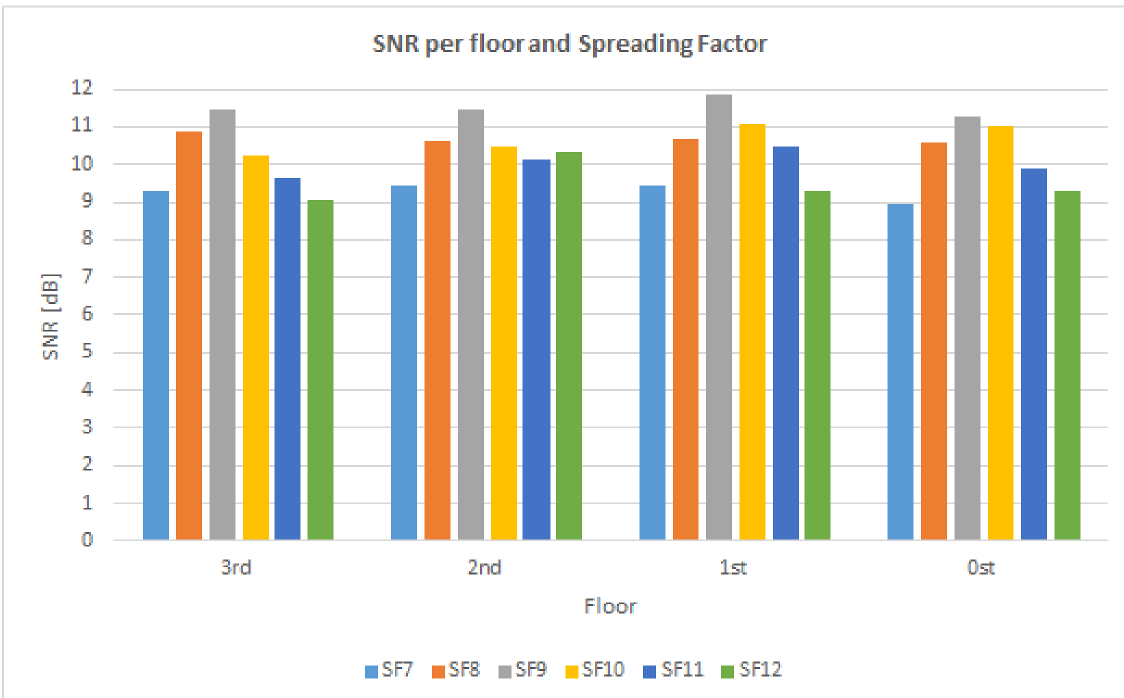

Figure 73: SNR for Spreading Factors in ENSIMAG floor's.

Repetitions: 1	**Number of packets send per repetition:** 2196
SF: cyclic	**Power:** 8 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor	

7.6 Consecutive packets lost

We check the performance of the errors in two different tests realised in different weeks and with different LoRa^{TM} Motes. Seeing the results we can see that isolated errors predominate in the two tests, and how the distribution of the errors looks similar. This isolated packets lost could be resend to try again if they arrive correctly to the demodulator. Otherwise it is also possible to have big burst of packets lost as we have seen on Section 7.1.4, in these condition would be more difficult to get the correct reception of the packet. On these cases one method could be to spread the retransmissions along over time if the packet does not have a really urgency to be send.

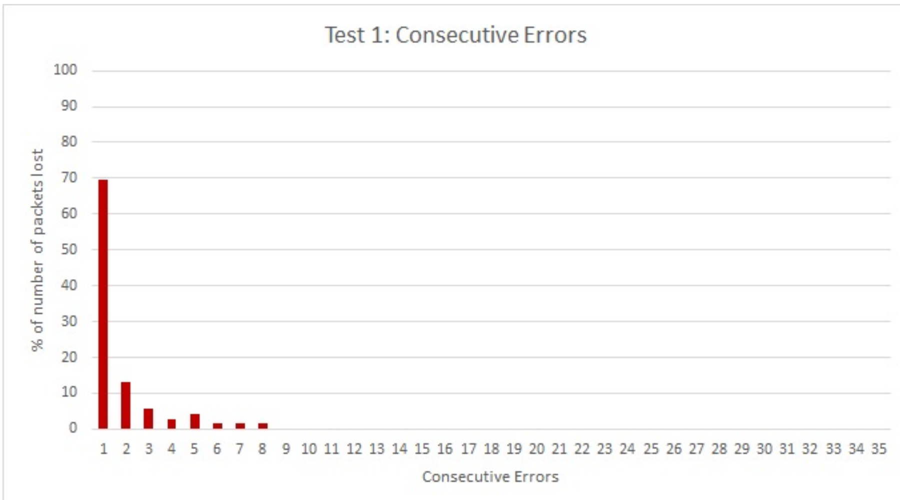

Figure 74: Test 1: Consecutive packets lost.

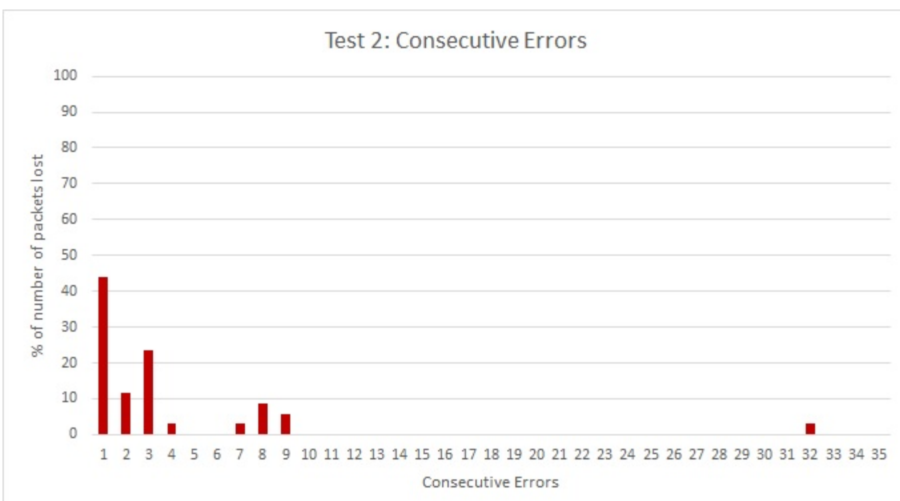

Figure 75: Test 2: Consecutive packets lost.

7.7 Maximum daily packets per LoRa^{TM} Mote and channel

We want to verify the theoretic values of maximum daily packets. Following the European telecommunication's regulations for the license-free 868 MHz band, the maximum permitted duty cycle for LoRa^{TM} 's channels is 1%.
We start with a theoretical analysis of $T_{on-the-air}$ (7) (duration of the packet transmission). This time will limit the cadence of packets because after each packet transmission, the device has to wait a time before he can send the next frame in the same sub-band.

$$T_{on-the-air} = \Big(n_{payload} + n_{preamble}\Big) \cdot T_{symbol} \tag{7}$$

As we can see, $T_{on-the-air}$ depends on three variables, number of symbols on the payload, number of symbols on the preamble and time of symbol.
$n_{symbols}$ is a variable that can be set by the user and it sets the number of symbols on the preamble.
T_{symbol} depends on the bandwidth and Spreading Factor selected for the transmission as we see on Equation 8.

$$T_{symbol} = \frac{1}{R_{symbol}} \tag{8}$$

$$R_{symbol} = \frac{BW}{2^{SF}} \tag{9}$$

The total payload length ($n_{payload}$), is a function of the Spreading Factor, the Payload Length (PL, measured in bytes), the use of explicit header or not (IH, 1 when enabled 0 when not), the use of the Cyclic Redundancy Check (CRC, 1 when enabled 0 when not), the Coding Rate (CR) and the use of the Low Data Rate Optimization (DE, 1 when used 0 when not).

$$n_{payload} = 8 + max\left[Ceil\left(\frac{8PL - 4SF + 28 + 16CRC - 20IH}{SF - 2DE} \cdot \frac{CR + 4}{4}, 0\right)\right] \tag{10}$$

It is important to see the differences between real payload data and PHY payload. The first one are the bytes of information useful for the user in Equation 10 named as PL. In the case of a LoRa^{TM} Mote, we can send frames with 16 bytes of useful data (LED, pressure, temperature, GPS position...).
PHY payload are the bytes of payload after the encapsulation.
For example one frame with 16 bytes of payload data, will be encapsulate and the PHY payload length will be 29 bytes.
The time that the LoRa^{TM} Mote has to wait before send the next packet is called T_{off} (11):

$$T_{off} = T_{on-the-air} \cdot DCycle - T_{on-the-air} \tag{11}$$

Where $DCycle$ is 100 in the case of a duty cycle of 1%.
So we can calculate the maximum daily packets per LoRa^{TM} and channel as (12):

$$Max.\ Daily\ packets = \frac{Seconds\ in\ a\ day}{T_{off}} \tag{12}$$

According to equations we can calculate $T_{on-the-air}$ for packets with variable payload, 8 symbols of preamble, CRC activated with a value of 4/5, Implicit Header deactivated, bandwidth of 125 kHz and Low Data Rate optimization activated for high spreading factors (SF11 and SF12).
The maximum application payload size length according the EU 863-870 ISM band in absence of the optional Fopt control field is given on Table 27. On Table 28 we show the theoretical values for the $T_{on-the-air}$.

Table 27: Maximum application payload.

Spreading Factor	Max. Application payload [Bytes].
SF7	222
SF8	222
SF9	115
SF10	51
SF11	51
SF12	51

Table 28: $T_{on-the-air}$.

Spreadin factor \ Payload in bytes	16	32	51	115	222
SF7	0,066816	0,090368	0,118016	0,2112	0,367872
SF8	0,121344	0,162304	0,209408	0,373248	0,64768
SF9	0,222208	0,295936	0,381952	0,672768	-
SF10	0,411648	0,54272	0,698368	-	-
SF11	0,872448	1,150976	1,511424	-	-
SF12	1,613824	2,138112	2,760704	-	-

Using Equations 11 and 12 we can obtain the maximum number of packets that we can send in a day with one LoRa^{TM} device transmitting all the packets trough the same channel.

Table 29: Theoretical maximum number of daily packets per LoRa^{TM} Mote and band.

Spreadin factor \ Payload in bytes	16	32	51	115	222
SF7	13061	9657	7394	4132	2372
SF8	7192	5377	4167	2338	1347
SF9	3927	2949	2284	1297	-
SF10	2120	1608	1249	-	-
SF11	1000	758	577	-	-
SF12	540	408	316	-	-

Next step is to compare this theoretical results with those obtained on the test with the LoRa^{TM} Motes. For this propose we modify the code of LoRa^{TM} class A to send messages of type Unconfirmed Data Up, only through one channel and changing Payload length at intervals of 50 packets.

Table 30: Case studies maximum number of daily packets per LoRa^{TM} Mote and band.

Payload in bytes / Spreadin factor	16	32	64	128
SF7	8798	7035	5155	3362
SF8	5557	4423	3149	2013
SF9	3673	2843	1917	1191
SF10	2023	1518	1064	650
SF11	964	745	492	304
SF12	553	447	-	-

7.8 Comparing unconfirmed data up and confirmed data up

On section 6.5.1 and 6.5.2 we explained the retransmission procedure for confirmed messages and the Data Rate Adaptation technique (section 6.5.3).
To compare the performance between *unconfirmed data up* and confirmed data up we put two devices LoRa^{TM} Motes, one sending confirmed data and the other one unconfirmed frames.
On Appendix C.3 we see how to define the message type.
For this test the two LoRa^{TM} Motes end-devices were placed on the same place than in the urban scenario explained previously, the gateway situated at a distance of 0,74 km from the end-devices was on a roof at a hight of 30 meters. Between them there is no direct Line Of Sight and they are separate by an urban environment as we can see in Figure 76.

Figure 76: LoRa^{TM} Motes and gateway position.

On Table 31 we see the parameters selected for these tests.

Table 31: Tests setup: Unconfirmed data up vs Confirmed data up

SF: default	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** default
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up and confirmed data up	

On Table 32 and 33 we see the number of packets sent by each end-device and the number of packets received at the gateway.
On row "Total number of packets transmitted" and concretely for *unconfirmed data* is counted the total number of packets sent, this means: packets sent for first time + possibles retransmissions. For this reason we see big differences between unconfirmed and confirmed data.
The PER values in this case are similar.

Table 32: Test 1 comparative: Unconfirmed data up vs Confirmed data up.

	Unconfirmed data up	**Confirmed data up**
Total number of packets transmitted	5934	4403
Total number of packets received	1159	633
PER	80.46	85.62

Table 33: Test 2 comparative: Unconfirmed data up vs Confirmed data up.

	Unconfirmed data up	**Confirmed data up**
Total number of packets transmitted	2062	2737
Total number of packets received	633	800
PER	69.3	70.77

On the other hand without considering the retransmissions for the PER probability, we see a big improvement on the confirmed data up messages as we see on Table 34 and 35.

Table 34: Test 1 comparative: Unconfirmed data up vs Confirmed data up without retransmissions.

	Unconfirmed data up	**Confirmed data up**
Total number of packets transmitted without considering the retransmissions	5934	802
Total number of packets received	1159	633
PER	80,46	21,07

Table 35: Test 2 comparative: Unconfirmed data up vs Confirmed data up without retransmissions.

	Unconfirmed data up	**Confirmed data up**
Total number of packets transmitted without considering the retransmissions	2062	923
Total number of packets received	633	800
PER	69.3	0,21

Figure 77: Test 1 comparative: Unconfirmed data up vs Confirmed data up without retransmissions.

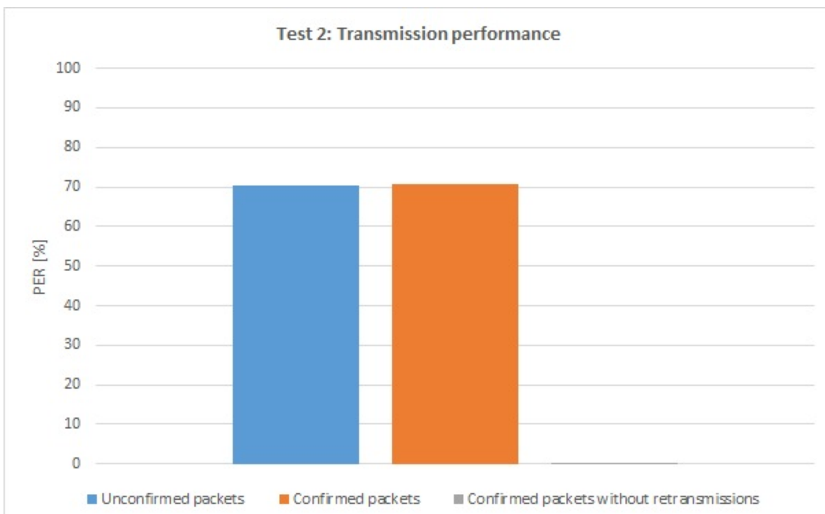

Figure 78: Test 2 comparative: Unconfirmed data up vs Confirmed data up without retransmissions.

We modified the code of the LoRa^{TM} Motes to send on the payload data the number of retransmissions done by the previous packet. In this way we can analyse the performance of the *confirmed data up* messages. On Figures 79 and 80 we can see the number of retries done by the packets, 1 retry means that the packet have been received well on the first transmission and no retransmissions have been need. Packets with more than 8 retransmission are discarded by the LoRa^{TM} Mote, we can control the maximum number of retransmissions with the function **LoRaMacSendConfirmedFrame**. It is interesting to see that in the first test only the 6.86% of the packets transmitted have arrived well on the first try. On test 2 we see a performance more expected, we can see a continuous decrease on the number of packets resend, until the point that only the 0.98% of the packets have to be send on eighth time and we have only lost the 0.22%.

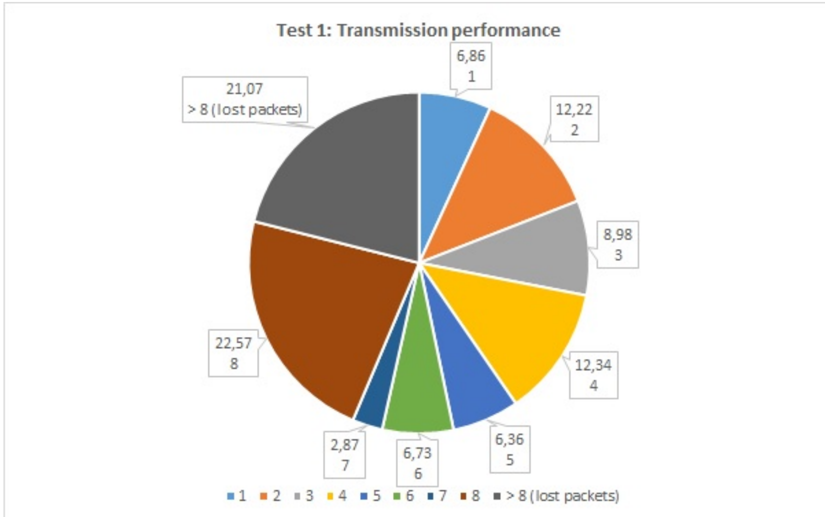

Figure 79: Test 1: Packet's number of retries.

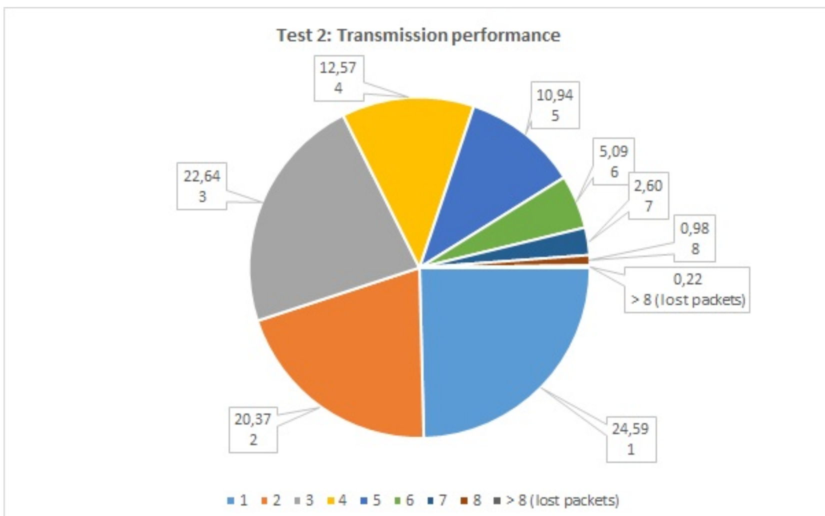

Figure 80: Test 2: Packet's number of retries.

Another interesting point is to see the data rate adaptation explained on Section 6.5.3. We can see how the spreading factor adapts to number of retransmission. If at the second try the LoRa^{TM} Mote have not received the ACK message send by the gateway, the Mote will increase the spreading factor to try that the packet reach the gateway. We can see this performance on the next table.

Table 36: Performance of the data rate adaptation.

Secuence	**Channel**	**SF**	**Number of retries**
6	869.25	SF8	4
5	867.9	SF9	6
4	868.6	SF8	3
3	869.25	SF9	5
2	869.25	SF10	8
1	868.75	SF7	2

8 Conclusion

Some of the results obtained on our tests have not been all the clarifying that we hope due to the big variation that we have observed in the different execution of the tests as we see on the tables of the Annex.
In other cases the results have not been as we expected for example on the SNR graph where one study more exhaustive on the field of the signal processing could explain the decrease of the Signal-to-Noise Ratio on the frames transmitted with the biggest Spreading Factors.
On the Cyclic Redundancy Check test we have seen that an increment on the Spreading Factor will improve the performance in comparison with the increment of the CRC at the expense of a bigger Time On the Air. Nevertheless when the slowest Spreading Factor is reached the performance of the communication can be improve with the use of higher coding rates. This option is not contemplated in the actual LoRaWAN^{TM} firmware, where the CRC keeps constant, it could be a good idea to implement some MAC command to adjust the CRC just as the network do with the data rate and transmission power. This would permit to optimize even more the network.
On the other hand, LoRa^{TM} modulation will work together with LoRaWAN^{TM} layer. In the end, the efficiency of LoRa^{TM} technology will fall back on the ability of the network server to manage intelligently the LoRaWAN^{TM} parameters of the big number of end-devices connected.
Another important aspect that will determine the quality of the communication link, will be the use of the shared ISM Band. This band will be shared between much more LPWAN technologies. The correct use of the spectrum and the compliance of the ISM regulations will be the cornerstone for the proper functioning of these LPWAN protocols.
The open code of LoRaWAN^{TM} will make easier the implementation of this technology in front others that are proprietary. At the moment there are already some projects underway. One example of this is The Things Network (TTN[11]), a crowsourced project created to extend a public LoRaWAN^{TM} network in every city. The topic of the shared spectrum it is being very commented on the community, they are proposing measures that could be taken against uncooperative devices, in example not forwarding the traffic that do not accomplish the Duty Cycle restrictions.
For a future work it would be interesting compare the spectrum utilization of the different LPWAN protocols as LoRa^{TM} , Sigfox (UNB), Ingenu (RPMA) or WAVIoT (NB-Fi).
Another interesting point to study will be how the increase of end-devices will affect the spectrum capacity and what solutions could be implemented to maximize the network performance: cooperation between network servers in the control of the LoRaWAN^{TM} parameters or which will be the consequences if we increase the number of gateways. Run some simulations regarding scaling with managed devices could give us some ideas on how to act in the different scenarios that will be present.

[11] `http://thethingsnetwork.org/`

A Limitations of ALOHA-type protocol implemented on class A

Defining:

- **N**: number of end-devices on the network.
- λ: packets send per end-devices per second.
- **T**: transmission time in seconds.
- S_{in}: arrival rate of new packets to the system per second.
- S_{out}: average number of successful packets transmissions per second.
- **G**: average number of overall transmission attempts per second.

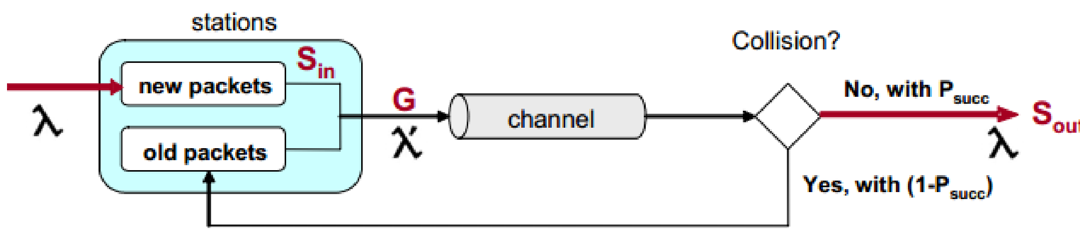

Figure 81: Aloha diagram.

If the system works under stable conditions and considering diagram of Figure 81:

$$S_{out} = S_{in} \tag{13}$$

$$S_{out} = P_{succ} \cdot G \tag{14}$$

To find P_{succ}, we consider that a frame sent at time t_0 will be transmitted successfully if no other packet attempts to transmit T_{tx} before and after t_0. On Figure 82 we can see the vulnerable period of the sent frame.

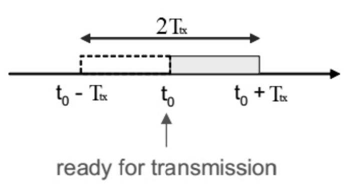

Figure 82: Vulnerable period

If frame arrivals are equally likely at any instant in time, and arrivals occur at an average rate of λ [packets per second], then we can model the arrivals like a Poisson random variable.

$$P[\text{k arrivals in T seconds}] = \frac{(T\lambda')^k}{k!} \cdot e^{-T\lambda'} \tag{15}$$

So the probability of a successful transmission is that no packets arrive in $\triangle t = 2 \cdot T_{tx}$:

$$P_{succ} = P[0 \text{ arrivals in } \triangle t \text{ seconds}] = e^{-2T_{tx}\lambda'} \tag{16}$$

Due to the traffic conservation, it must accomplish $\lambda' = \lambda$ + retries, this implies:

$$\lambda' = \lambda + \lambda'(1 - P_{succ}) \tag{17}$$

And denoting:

$$G = \lambda' \cdot T_{tx}$$
$$S = \lambda \cdot T_{tx}$$

We can rewrite Equation 17 like:

$$S = G \cdot e^{-2G} \tag{18}$$

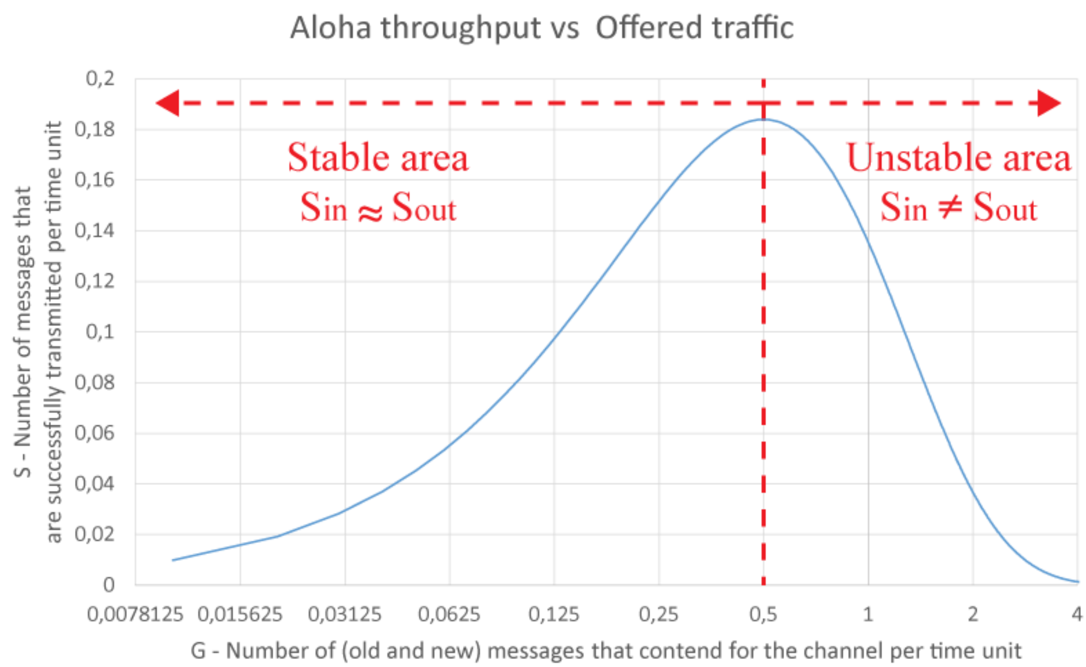

Figure 83: Successful traffic (throughput)

As we can see on Figure 83 where is graphed $S(G)$, the maximum throughput is approximately 18,4% when the load offer to the network is 0,5 packets per time unit. This means that we are missing 81,6% of the data due to the collisions between packets.
According to the equations we can calculate the maximum number of end-devices that we can connect to the network in different scenarios or the maximum daily packets send by end-device to maximize the channel goodput.

$$S(G)_{max} = S(0,5) \approx 18.4 \Longrightarrow G = 0,5 = T_{tx} \cdot \lambda \tag{19}$$

On Equation 19, λ represents all the packets sent to the network by all the LoRa^{TM} devices in the same coverage area, so we can rewrite λ as:

$$\lambda = N \cdot \lambda_i$$

Where λ_i represents the throughput of one device and N the number of devices on the coverage area. Each packet sent by one LoRa^{TM} device, will select randomly one channel from the 8 available, so the total throughput sent by one LoRa^{TM} devices per channel will be:

$$\lambda_{i\ per\ channel} = \frac{\lambda_i}{8\ channels}$$

On Table 29, we showed the maximum daily packets that one end-device can send in one day per band. We have to remember that the LoRaWAN^{TM} enforces a per sub-band duty-cycle limitation. Each time a frame is transmitted in a given sub-band, the time of emission and the Time On Air duration of the frame are recorded for this sub-band. The same sub-band cannot be used again until the T_{off} expires.
According to this limitation and considering the two available sub-bands on LoRaWAN^{TM} (Band0 from 868.1 MHz to 868.5 MHz and Band1 from 867.1 MHz to 867.9 MHz), we can consider that the maximum daily packets that one end-device can send in one day using the two available bands will be the double of the shown on Table 29.

Table 37: Maximum throughput per end-device and per channel considering packets of 16B payload, a bandwidth of 125 kHz and according to the ETSI regulations [19].

Spreadin factor	Max. daily packets per end-device [packets/day]	Max. λ_i per end-device [packets/sec]	Max. $\lambda_{i\ per\ channel}$ [packets/sec]
SF7	26122	0,302	0,037
SF8	14384	0,166	0,020
SF9	7854	0,090	0,011
SF10	4240	0,049	0,006
SF11	2000	0,023	0,002
SF12	1080	0,012	0,001

With the values from Table 37 and Equation 19 we can obtain the maximum number of end-devices, (N), that can be placed on the same coverage area (maximizing the throughput of network) in two different cases: maximizing the packets send by the end-devices and in the other case, considering 1 packet per end-device and per day.

$$N = \frac{0,5}{T \cdot \lambda_{channel}} \tag{20}$$

Table 38: Maximum number of end-device on the coverage area maximizing the throughput (number of packets per day).

Spreadin factor	**Max. daily packets per end-device** [packets/day]	**Max.** $\lambda_{i\ per\ channel}$ **[packets/sec]**	$T_{on-the-air}$ **[sec]**	**N**
SF7	26122	0,037	0,302	198
SF8	14384	0,020	0,166	198
SF9	7854	0,011	0,090	198
SF10	4240	0,006	0,049	198
SF11	2000	0,002	0,023	198
SF12	1080	0,001	0,012	198

Table 39: Maximum number of end-devices on the coverage area for 1 daily packet per end-device.

Spreadin factor	$\lambda_{i\ per\ channel}$ **= 1 packet/day [packets/sec]**	$T_{on-the-air}$ **[sec]**	**N**
SF7	$1,44676 * 10^{-6}$	0,066	5172413
SF8	$1,44676 * 10^{-6}$	0,121	2848101
SF9	$1,44676 * 10^{-6}$	0,222	1555299
SF10	$1,44676 * 10^{-6}$	0,411	839552
SF11	$1,44676 * 10^{-6}$	0,872	396126
SF12	$1,44676 * 10^{-6}$	1,613	214149

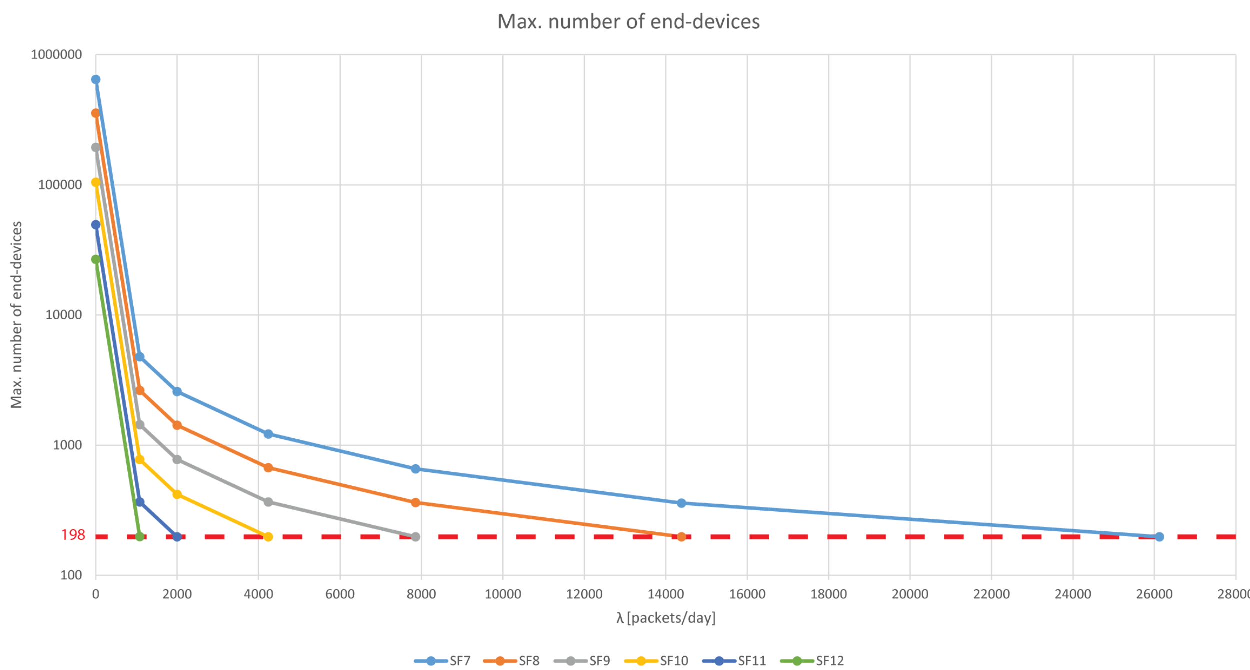

Figure 84: Maximum number of end-devices per throughput according to Equation 19.

As we see on Figure 84, if we want to maximize the number of packets sent per day, the maximum number of end-devices on the same coverage area must be 198 independently of the Spreading Factor. In this case, using the fastest Spreading Factor (SF7) we can reach 26122 packets sent per day per end-device, by the other hand using the slowest one (SF12), we reach the limit of 1080 packets per day per end-device.
The biggest difference between the number of end-devices comes when we decrease the number of packets sent per day. In the opposite case shown previously if each end-devices send 1 packet per day the number of end-devices possible on the coverage area grow considerately by a factor of 100.
We have to recall that this results are planned in a scenario with all the end-devices working on the following hypothesis:

- Class A.
- Packets with 16B of payload.
- CR = 4/5.
- BW = 125 kHz
- Same spreading factor.
- 8 channels with equal probability to be chosen.
- Considering retransmissions after each collision.
- Ignoring ACK messages.
- Respecting the 1% Duty Cycle according to the ETSI regulations [19].

B Trace and debug on Keil µVision

The following pictures shows how to configure the flash tools for our LoRa™ Mote devices. On the version 4.0 of the node firmware, we will have to add some code lines to be able to use the trace function on Keil µVision. As we explained on section 6 the trace debug will display in real time the printf-style output of the Instrumented Trace (ITM) in a terminal window: Debug (printf) Viewer (91).
To enable the printf output we will add the code showed on Table 40 at the end of utilities.c file.

Table 40: Printf output code

utilities.c

```
#ifdef __GNUC__
/* With GCC/RAISONANCE, small printf (option LD Linker->Libraries->Small
     printf
 set to 'Yes') calls __io_putchar() */
int __io_putchar( int c )
#else /* __GNUC__ */
int fputc( int c, FILE *stream )
#endif
{
return( ITM_SendChar( c ) );
}
```

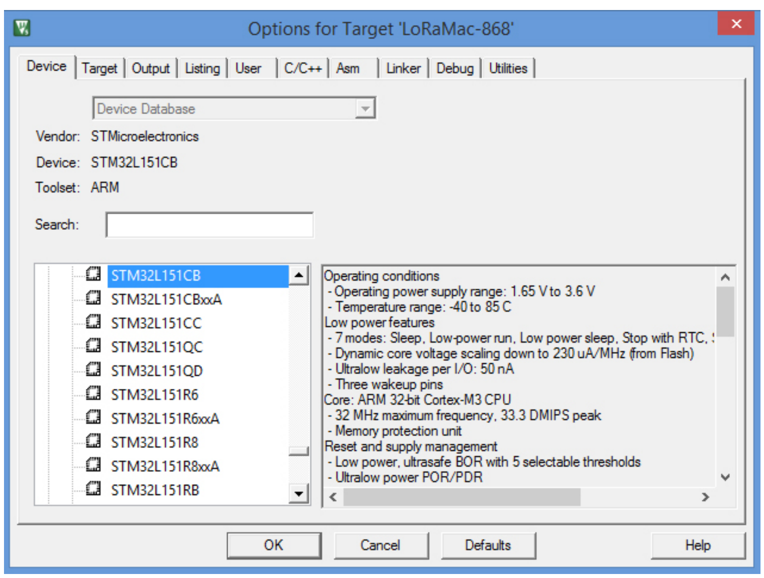

Figure 85: Device's options

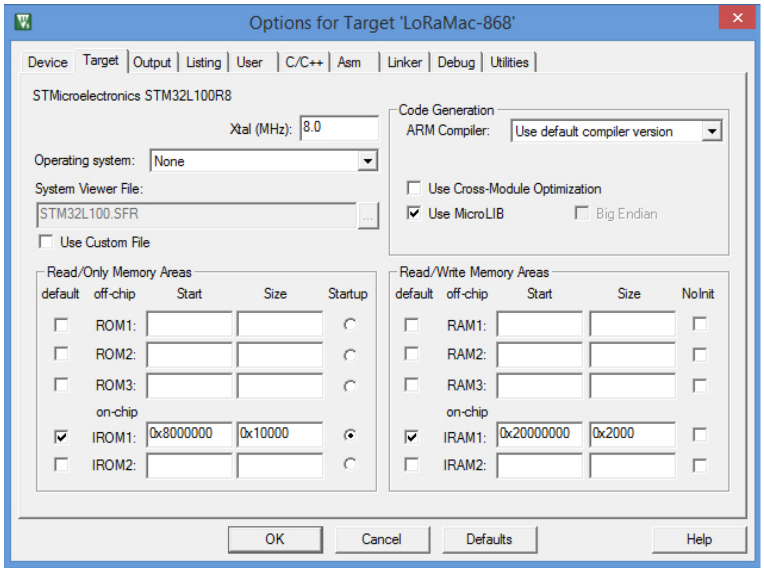

Figure 86: Target's options

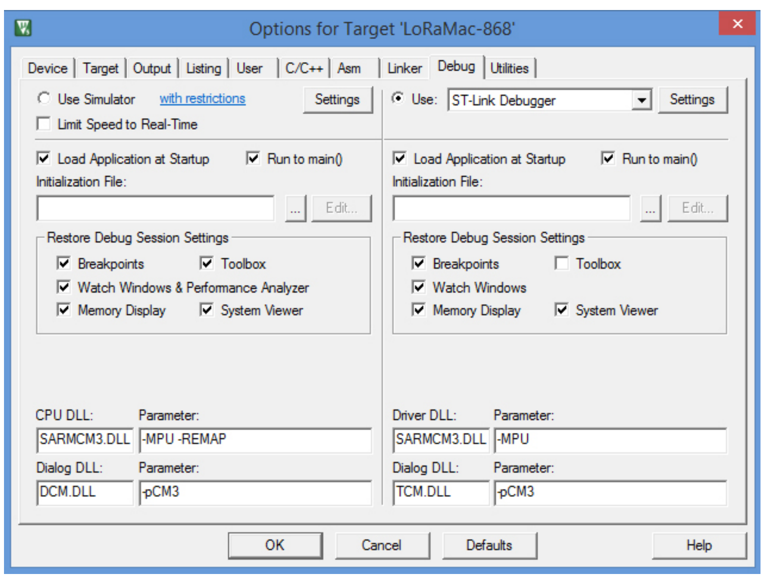

Figure 87: Debug's options

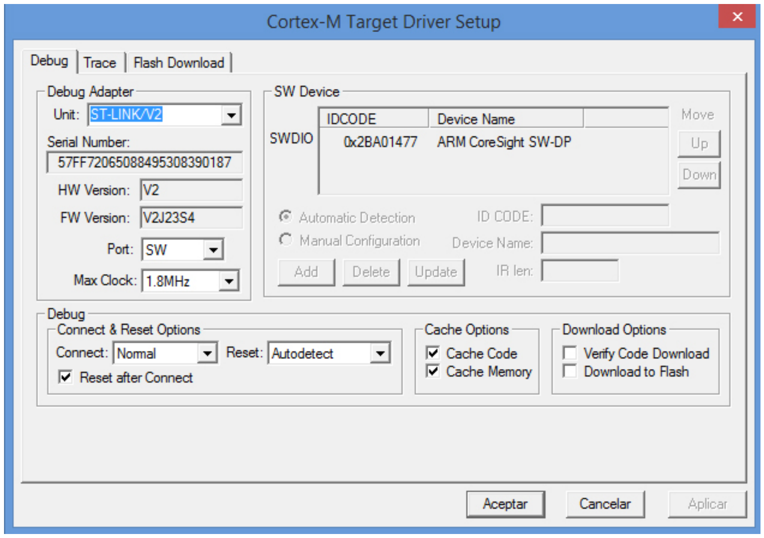

Figure 88: Target driver setup

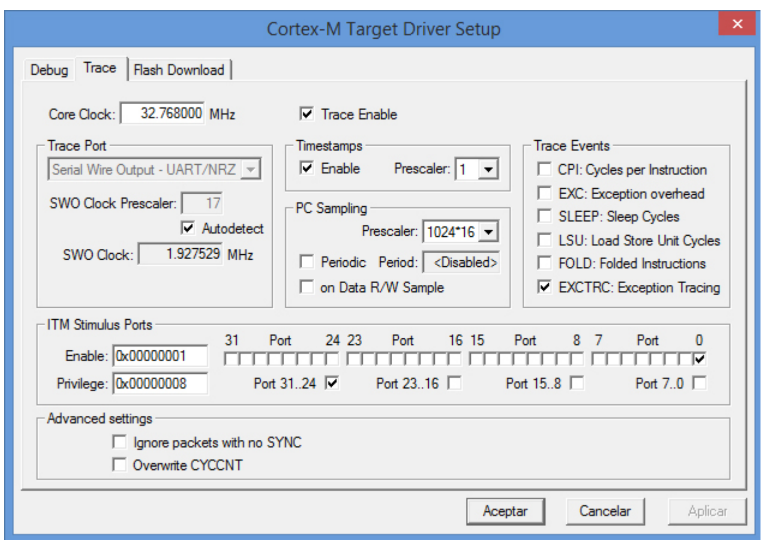

Figure 89: Target driver setup

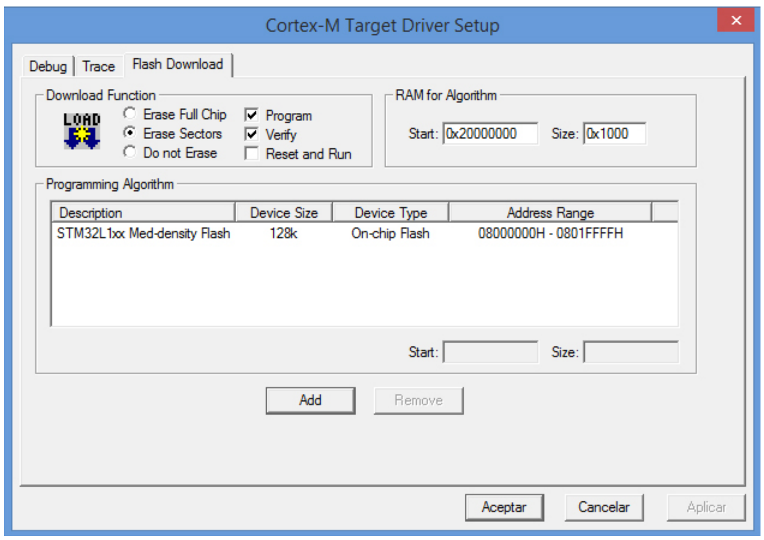

Figure 90: Target driver setup

Once configured, we can call the function printf to see it's out message on the Debug (printf) Viewer.

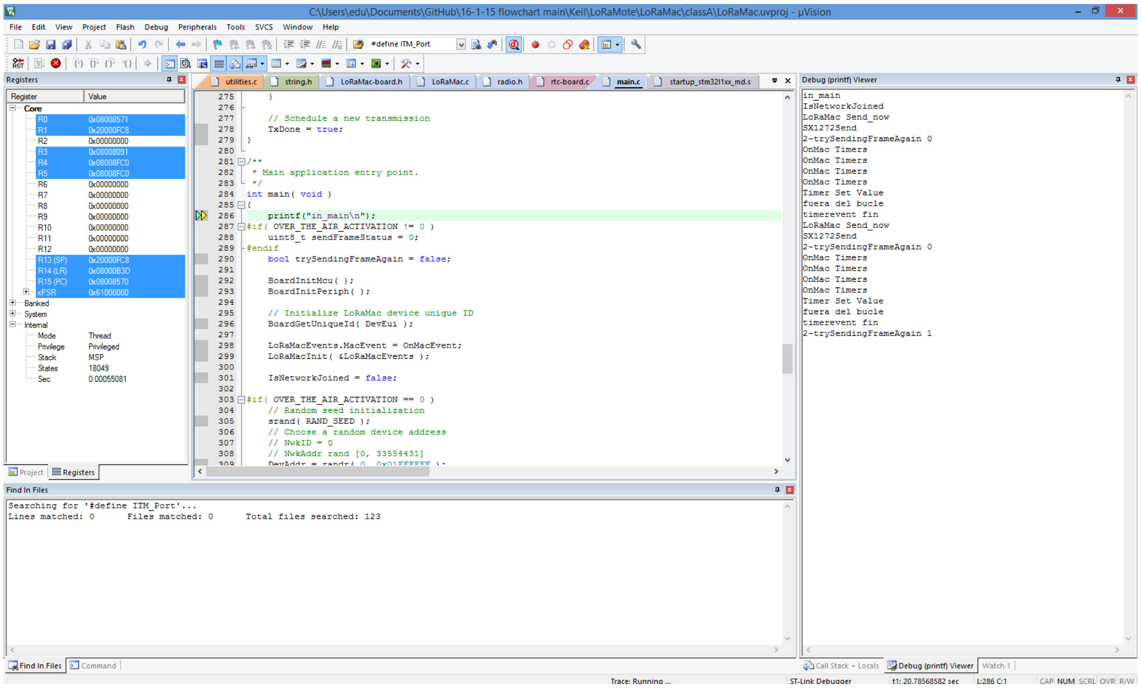

Figure 91: Target driver setup

C LoRaMAC layer. Documentation for the API

C.1 Default Spreading Factor and Adaptative Data Rate

Table 41: Confirmed messages setup.

File	**LoRaMac-board.h**
Description	Default datarate used by the node
Code	`#define LORAMAC_DEFAULT_DATARATE DR_0`
Notes	DR_i = SF(12-i) i $\in$0..7

Table 42: Confirmed messages setup.

File	**main-api-v3.c**
Description	LoRaWAN confirmed messages
Code	`#define LORAWAN_ADR_ON 1`
Notes	1 $\Longrightarrow$ ADR ON 0 $\Longrightarrow$ ADR OFF

C.2 Default power transmission

Table 43: Confirmed messages setup.

File	**LoRaMac-board.h**
Description	Default datarate used by the node
Code	`#define LORAMAC_DEFAULT_TX_POWER TX_POWER_14_DBM`
Notes	TX_POWER_i_DBM = i dBm i $\in$02, 05, 08, 11, 14, 20

C.3 Confirmed messages parameters

Table 44: Confirmed messages setup.

File	**main-api-v3.c**
Description	LoRaWAN confirmed messages
Code	`#define LORAWAN_CONFIRMED_MSG_ON false`
Notes	false $\Longrightarrow$ Unconfirmed message true $\Longrightarrow$ Confirmed message

Table 45: Confirmed messages setup.

File	**main-api-v3.c**
Description	Number of retries to receive the acknowledgement when a confirmed message has been sent
Code	`sendFrameStatus = LoRaMacSendConfirmedFrame( AppPort, AppData, AppDataSize, 8 );`
Notes	`* \param [IN] nbRetries Number of retries to receive the acknowledgement` `uint8_t LoRaMacSendConfirmedFrame( uint8_t fPort, void *fBuffer, uint16_t fBufferSize, uint8_t nbRetries );`

Table 46: Confirmed messages setup.

File	**main-api-v3.c**
Description	Maximum number of retries to receive the acknowledgement when a confirmed message has been sent
Code	`#define MAX_ACK_RETRIES 8`

D Spectrum analysis

The spectrum scan firmware provided by Semtech allows us to obtain a visual representation of the spectrum content.
We can find the spectrum scan on the firmware installed on the gateway on the directory: `/mnt/fsuser-1/forwarder_network_demo/spectrum_scan`.
The first stept to run spectrum scan is stop any packet forwarder process currently running og the gateway, we can use kill-pkt-fwd.sh script that is allocated on `/mnt/fsuser-1/forwarder_network_demo/`.
After that we go into `spectrum_scan` directory and we turn on the modem: `modem_on.sh`.
Now we are ready to run the spectral scan:

```
./util_rssi_histogram --file histogram4_n250_p1000.csv
--fmin 867000000 --fmax 869500000 --fstep 25000 -n 250 -p 1000
```

In this case we scan from 867 Mhz to 869,5 Mhz, by steps of 25 kHz, capturing 250 RSSI captures with 32 kHz ratio of capture rate.

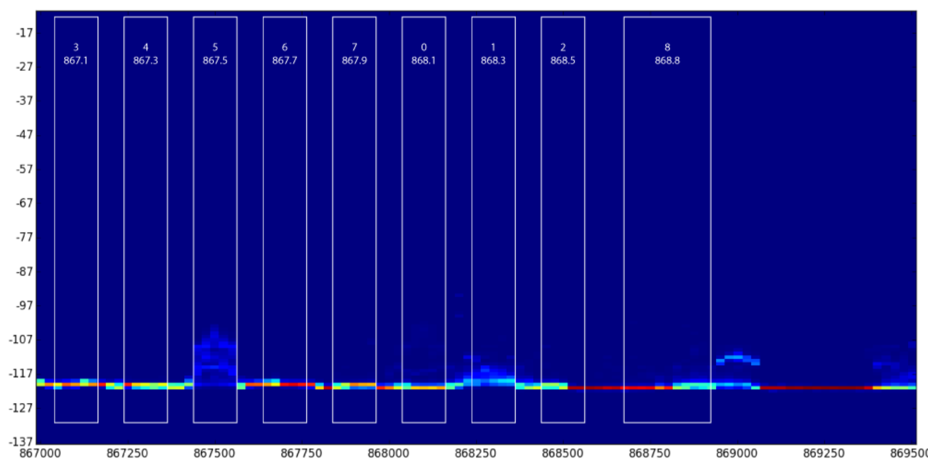

Figure 92: Scan spectrum

E Tests results

E.1 RSSI for channel and spreading factor results

Table 47: Setup for RSSI for channel tests

Repetitions: 5	**Number of packets send per repetition:** 4800
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor, L.O.S., urban	

Table 48: RSSI in dBm's for channel results in 3 scenarios

		Channel							
		Ch3 867.1 Mhz	**Ch4 867.3 Mhz**	**Ch5 867.5 Mhz**	**Ch6 867.7 Mhz**	**Ch7 867.9 Mhz**	**Ch0 868.1 Mhz**	**Ch1 868.3 Mhz**	**Ch2 868.5 Mhz**
Indoor	**Test 1 RSSI**	-78.8123	-76.5218	-75.6535	-76.5371	-79.0952	-78.9270	-79.3353	-81.2673
	Test 2 RSSI	-79.8424	-78.0629	-77.9059	-78.6092	-79.4927	-80.0210	-80.6681	-80.9473
	Test 3 RSSI	-74.9848	-74.4372	-74.9239	-76.2361	-77.6437	-77.8316	-76.1424	-76.8716
	Test 4 RSSI	-76.9276	-76.9195	-77.9120	-78.7005	-79.8796	-78.8129	-77.5018	-77.0749
	Test 5 RSSI	-76.1546	-76.1124	-76.9264	-78.3216	-78.8885	-78.0924	-75.9799	-75.3895
	Average	-77.1933	-76.3275	-76.6350	-77.6732	-78.9686	-78.6768	-77.7693	-77.6732
L.O.S.	**Test 1 RSSI**	-76.4230	-74.5254	-73.3464	-73.0845	-74.4776	-74.6567	-74.1318	-74.1712
	Test 2 RSSI	-74.0124	-73.0616	-73.3555	-73.4789	-74.2172	-73.7766	-73.4256	-73.4691
	Test 3 RSSI	-81.9731	-80.3289	-80.5920	-81.1244	-82.8622	-83.2067	-83.1286	-82.9782
	Test 4 RSSI	-86.4007	-85.6953	-85.6795	-85.7613	-86.9261	-86.2617	-84.9160	-85.2446
	Test 5 RSSI	-75.0866	-73.7210	-72.8030	-72.8851	-74.8471	-74.6319	-73.4627	-73.2085
	Average	-78.9174	-77.5913	-77.3153	-77.4303	-78.8229	-78.6682	-77.9514	-77.4303
Urban	**Test 1 RSSI**	-121.8000	-120.5000	-	-121.0794	-122.2981	-121.8547	-120.0000	-120.5000
	Test 2 RSSI	-121.5625	-121.3444	-	-121.2205	-122.4436	-121.7600	-	-120.4853
	Test 3 RSSI	-121.3883	-121.2500	-120.0000	-121.1977	-122.1707	-122.0488	-120.0000	-120.8365
	Test 4 RSSI	-121.2276	-	-	-120.7479	-121.7619	-	-120.0000	-120.8649
	Test 5 RSSI	-121.2167	-	-	-120.9600	-121.8174	-	-	-120.5000
	Average	-121.3372	-121.3148	-120.0000	-121.0323	-122.1887	-121.8288	-120.0000	-121.0323

E.2 SNR for channel and spreading factor results

Table 49: Setup for SNR for channel tests

Repetitions: 5	**Number of packets send per repetition:** 4800
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor, L.O.S., urban	

Table 50: SNR in dB's for channel results in 3 scenarios

		Channel							
		Ch3 867.1 Mhz	**Ch4 867.3 Mhz**	**Ch5 867.5 Mhz**	**Ch6 867.7 Mhz**	**Ch7 867.9 Mhz**	**Ch0 868.1 Mhz**	**Ch1 868.3 Mhz**	**Ch2 868.5 Mhz**
Indoor	**Test 1 SNR**	11.0082	11.1250	8.6385	11.2261	10.9504	11.2836	11.0000	8.6015
	Test 2 SNR	11.1901	10.8992	8.6667	11.0448	11.1290	11.0583	11.2773	8.7561
	Test 3 SNR	11.1419	11.1304	8.5400	11.0872	11.2450	11.1769	11.0930	8.6423
	Test 4 SNR	11.3475	11.3185	8.8692	11.2857	11.5854	11.2897	11.3268	8.9063
	Test 5 SNR	11.3025	11.3929	8.9161	11.4873	11.5411	11.3117	11.3410	8.7961
	Average	11.1980	11.1732	8.7261	11.2262	11.2902	11.2240	11.2076	8.7404
L.O.S.	**Test 1 SNR**	11.1387	11.2778	8.5081	11.4737	11.3053	10.9580	11.2248	8.5182
	Test 2 SNR	11.5725	11.3282	8.5906	11.3464	11.5310	11.2414	11.2649	8.7744
	Test 3 SNR	11.2484	11.3375	8.7183	11.3718	11.1088	11.0276	10.7647	8.5038
	Test 4 SNR	11.4531	11.2763	8.7708	11.2697	11.3265	11.2911	11.2192	8.7987
	Test 5 SNR	11.1136	11.2143	8.6544	11.2297	11.1793	10.9767	11.3553	8.5000
	Average	11.3053	11.2868	8.6484	11.3383	11.2902	11.0990	11.1658	8.6190
Urban	**Test 1 SNR**	-19.5000	-	-	-16.1333	-15.4068	-16.6429	-	-19.5000
	Test 2 SNR	-18.2308	-18.1333	-	-17.0645	-15.7255	-15.9787	-	-17.7500
	Test 3 SNR	-17.5000	-17.7500	-15.8333	-16.7917	-18.7273	-16.3333	-19.5000	-16.8846
	Test 4 SNR	-18.1379	-	-	-17.9333	-18.0741	-	-20.0000	-18.8235
	Test 5 SNR	-18.3000	-	-	-18.4286	-18.0000	-	-	-19.0000
	Average	-18.3337	-17.9417	-15.8333	-17.2703	-17.1867	-16.3183	-19.7500	-18.3916

E.3 PER for channel and spreading factor results

Table 51: Setup for PER for channel tests

Repetitions: 5	**Number of packets send per repetition:** 4800
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor, L.O.S., urban	

Table 52: PER for channel results in 3 scenarios

		Channel							
		Ch3 867.1 Mhz	**Ch4 867.3 Mhz**	**Ch5 867.5 Mhz**	**Ch6 867.7 Mhz**	**Ch7 867.9 Mhz**	**Ch0 868.1 Mhz**	**Ch1 868.3 Mhz**	**Ch2 868.5 Mhz**
Indoor	**Test 1 PER**	0,7843	1,1765	0,9804	2,1569	1,1765	0,5882	0,5882	0,9804
	Test 2 PER	0,8333	0,6250	0,4167	0,8333	0,6250	0,8333	0,2083	1,2500
	Test 3 PER	1,0000	0,5000	1,5000	1,1667	0,8333	2,0000	1,6667	1,3333
	Test 4 PER	5,6667	6,8333	5,3333	4,8333	4,5000	4,6667	5,0000	4,3333
	Test 5 PER	0,8333	0,6667	0,3333	0,5000	1,3333	0,8333	0,3333	1,1667
	Average	1,8235	1,9603	1,7127	1,8980	1,6936	1,7843	1,5593	1,8127
L.O.S.	**Test 1 PER**	0,5814	0,7752	0,9690	1,3566	0,5814	2,3256	0,0000	0,3876
	Test 2 PER	6,1667	5,3333	4,8333	5,0000	4,8333	4,5000	4,8333	5,5000
	Test 3 PER	1,0000	0,6667	0,3333	0,8333	0,8333	0,8333	1,5000	0,5000
	Test 4 PER	0,1667	1,0000	0,1667	0,8333	0,8333	0,6667	0,8333	0,5000
	Test 5 PER	1,8333	0,8333	1,0000	1,3333	0,8333	0,8333	1,6667	1,6667
	Average	1,9496	1,7217	1,4605	1,8713	1,5829	1,8318	1,7667	1,7109
Urban	**Test 1 PER**	98,3333	99,6667	100,0000	89,5000	65,3333	80,5000	99,8333	99,6667
	Test 2 PER	88,0952	86,6071	100,0000	81,1012	61,7560	73,9583	100,0000	89,8810
	Test 3 PER	84,6726	97,6190	96,7262	87,2024	93,8988	93,8988	99,2560	84,5238
	Test 4 PER	81,6964	100,0000	100,0000	82,2917	84,3750	100,0000	99,8512	88,9881
	Test 5 PER	82,1429	100,0000	100,0000	85,1190	82,8869	100,0000	100,0000	96,4286
	Average	86,9881	92,1131	96,7262	85,0429	77,6500	82,7857	99,5536	91,8976

E.4 PER for Cyclic Redundancy Check results

Table 53: Setup for Cyclic Redundancy Check tests

Repetitions: 5	**Number of packets send per repetition:** 2304
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** cyclic
Scenario: indoor, L.O.S., urban	

Table 54: PER for Cyclic Redundancy Check results in 3 scenarios

		Coding Rate			
		4/5	**4/6**	**4/7**	**4/8**
Indoor	**Test 1 PER**	0.1736	0.8681	1.5625	1.5625
	Test 2 PER	1.0417	0.3472	0.3472	1.0417
	Test 3 PER	0.5208	1.2153	1.7361	0.8681
	Test 4 PER	0.8681	1.9097	0.8681	0.5208
	Test 5 PER	0.8681	0.5208	0.6944	1.3889
	Average	0.6944	0.9722	1.0417	1.0764
L.O.S.	**Test 1 PER**	1.7361	0.5208	1.2153	0.6944
	Test 2 PER	0.5208	1.2153	0.5208	1.3889
	Test 3 PER	2.2569	2.0833	2.6042	2.4306
	Test 4 PER	2.4306	1.7361	2.0833	2.6042
	Test 5 PER	0.5208	0.1736	0.8681	0.3472
	Average	1.4931	1.1458	1.4583	1.4931
Urban	**Test 1 PER**	95.1389	94.2708	92.7083	92.8819
	Test 2 PER	94.6181	94.4444	93.4028	91.8403
	Test 3 PER	90.2778	89.4097	89.2361	87.6736
	Test 4 PER	84.5486	83.8542	81.5972	81.4236
	Test 5 PER	87.3264	85.7639	83.6806	83.5069
	Average	90.3819	89.5486	88.1250	87.4653

E.5 PER for preamble length results

Table 55: Setup for preamble length tests

Repetitions: 5	**Number of packets send per repetition:** 5376
SF: cyclic	**Power:** 14 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** cyclic
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor, L.O.S., urban	

Table 56: PER for preamble length results in 3 scenarios

		Preamble length [symbols]						
		5	**6**	**8**	**10**	**12**	**14**	**16**
Indoor	**Test 1 PER**	75.9115	5.3385	1.1719	1.3021	1.5625	1.3021	1.9531
	Test 2 PER	78.9063	7.6823	2.6042	1.4323	1.6927	1.6927	2.3438
	Test 3 PER	75.0000	5.9896	3.5156	3.3854	2.4740	3.2552	3.9063
	Test 4 PER	75.3906	5.9896	0.9115	0.9115	0.5208	1.4323	1.9531
	Test 5 PER	72.1354	7.5521	0.6510	0.5208	0.3906	1.6927	2.6042
	Average	75.4688	6.5104	1.7708	1.5104	1.3281	1.8750	2.5521
L.O.S.	**Test 1 PER**	73.8281	6.5104	0.7813	0.7813	0.2604	0.6510	1.6927
	Test 2 PER	70.4427	5.7292	1.8229	0.9115	0.6510	0.7813	1.6927
	Test 3 PER	72.6563	5.4688	0.7813	1.1719	0.5208	0.7813	1.9531
	Test 4 PER	71.3542	5.5990	1.5625	0.6510	1.4323	0.5208	1.9531
	Test 5 PER	74.2188	4.5573	1.4323	0.6510	0.9115	0.5208	1.9531
	Average	72.5000	5.5729	1.2760	0.8333	0.7552	0.6510	1.8490
Urban	**Test 1 PER**	94.1406	90.3646	89.8438	89.7135	88.6719	89.4531	90.4948
	Test 2 PER	94.6615	89.3229	87.3698	88.8021	88.9323	88.8021	89.8438
	Test 3 PER	94.7917	89.7135	89.3229	88.5417	89.1927	88.8021	88.0208
	Test 4 PER	95.8333	92.9688	92.7083	93.7500	93.6198	91.6667	93.0990
	Test 5 PER	95.8333	92.9688	92.7083	93.7500	93.6198	91.6667	93.0990
	Average	94.2188	88.7500	88.7500	89.1146	88.9583	87.7604	88.4896

E.6 Indoor scenario results

Table 57: Setup for Cyclic Redundancy Check tests

Repetitions: 1	**Number of packets send per repetition:** 2196
SF: cyclic	**Power:** 8 dBm
Duty Cycle: 100%	**Channel:** cyclic
Payload: 16 B	**Preamble:** 8 Symbols
Message type: unconfirmed data up	**CRC:** 4/5
Scenario: indoor	

Table 58: RSSI for indoor scenario test

	ENSIMAG's floor			
	3rd	**2nd**	**1st**	**0st**
RSSI for SF7	-63,1965	-71,3276	-76,9669	-95,6552
RSSI for SF8	-62,1603	-70,9547	-76,4878	-95,1833
RSSI for SF9	-65,6486	-74,3616	-79,1393	-97,9153
RSSI for SF10	-60,8064	-69,904	-75,7317	-94,0672
RSSI for SF11	-63,7876	-71,9048	-76,8182	-96,0769
RSSI for SF12	-64,9369	-73,8725	-79,1855	-96,9328
RSSI Average	-63,4227	-72,0542	-77,3882	-95,9718

Table 59: SNR for indoor scenario test

	ENSIMAG's floor			
	3rd	**2nd**	**1st**	**0st**
SNR for SF7	9,2800	9,4487	9,4645	8,9293
SNR for SF8	10,8718	10,6383	10,6610	10,5775
SNR for SF9	11,4831	11,4598	11,8361	11,2873
SNR for SF10	10,2561	10,4783	11,0780	11,0269
SNR for SF11	9,6667	10,1327	10,4868	9,8709
SNR for SF12	9,0556	10,3253	9,3016	9,2790
SNR Average	10,4728	10,5063	10,7599	10,4552

F Spread Spectrum principles

As we explained on Section 3.1 by increasing the bandwidth of the signal we can compensate for the degradation of the Signal-to-Noise Ratio of a radio channel.
LoRa^{TM} modulation addresses all of the issues associated with DSSS systems to provide a lowcost, low-power, yet above all robust alternative to the traditional spread-spectrum communications techniques.

F.0.1 Modulation principle of Direct Sequence Spread Spectrum

The transmitted bandwidth is determined by the chip rate sequence and by the baseband filtering. The implementation limits the maximum R_c (clock rate or chip rate). On Figure 93 we can see how the input signal is spread, the frequency bandwidth of this chip is equivalent to the final spectral bandwidth of the signal to be send.

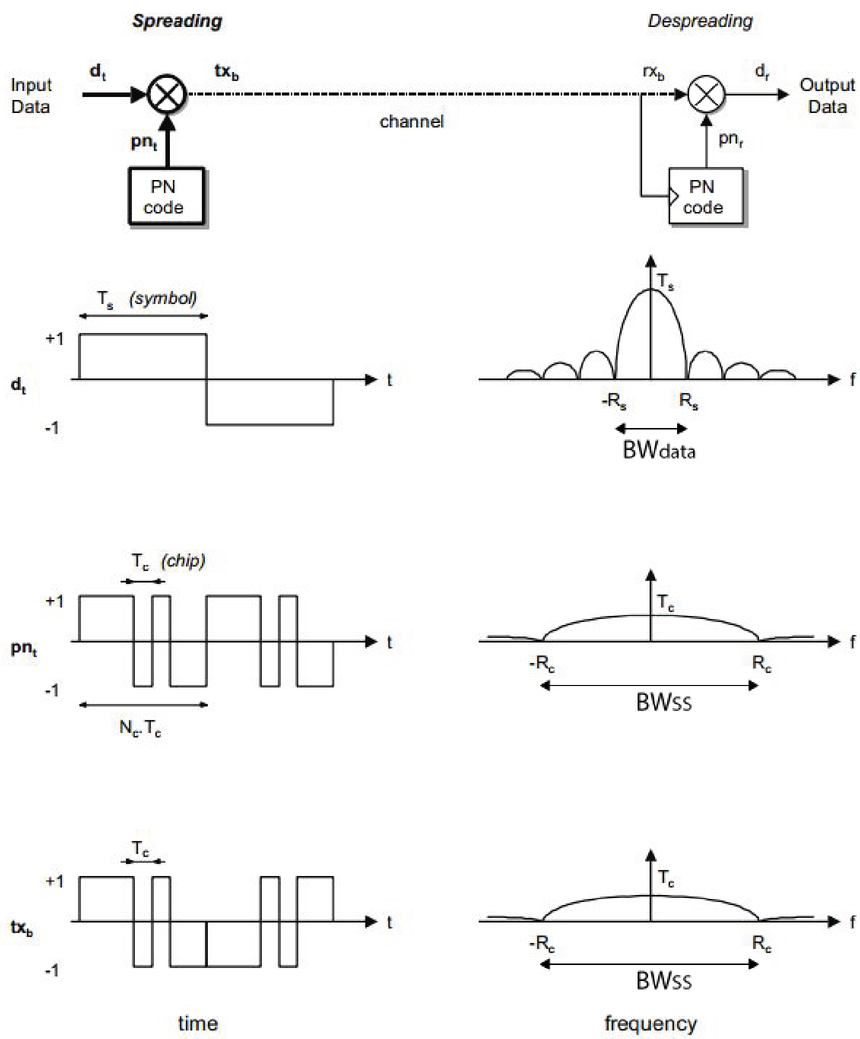

Figure 93: DSSS spreading technique[12].

[12] http://sss-mag.com/pdf/Ss_jme_denayer_intro_print.pdf

On figure 93 we can see the two inputs of our system:

- Binary data d_t (information to be send) with symbol rate $R_s = 1/T_s$.
- Pseudo-noise code (chip sequence) pn_t with chip rate $R_c = 1/T_c$ (a divisor of R_s).

We emphasise the relation between the ratio of the chip rate R_c and the data symbol rate R_s, that will give us the bandwidth expansion factor also known as the Spreading Factor. As we see on Table 2 the possibles values for LoRa$^{\text{TM}}$ are: 128, 256, ..., 4096.

$$\text{SF} = \frac{BW_{SS}}{BW_{data}} = \frac{R_c}{R_s} = \frac{T_s}{T_c} = N_c$$

As we said the transmitted data is multiplied by the chip sequence to produce the Spread Spectrum baseband signal tx_b:

$$tx_b = d_t \cdot pn_t$$

As a result, the signal d_t which has a BW_{data} have been spread over a much larger bandwidth BW_{SS}:

$$BW_{data} \simeq R_s \ll BW_{SS} \simeq R_c$$

F.0.2 Demodulation principle of Direct Sequence Spread Spectrum

To demodulate, the received signal is multiplied by pn_r, this is the same chip sequence as pn_t (the chip sequence used in the transmitter). This operation is called (spectrum) despreading, since the effect is to undo the spreading operation done by the transmitter. On Figure 94 we can see the despreading process.

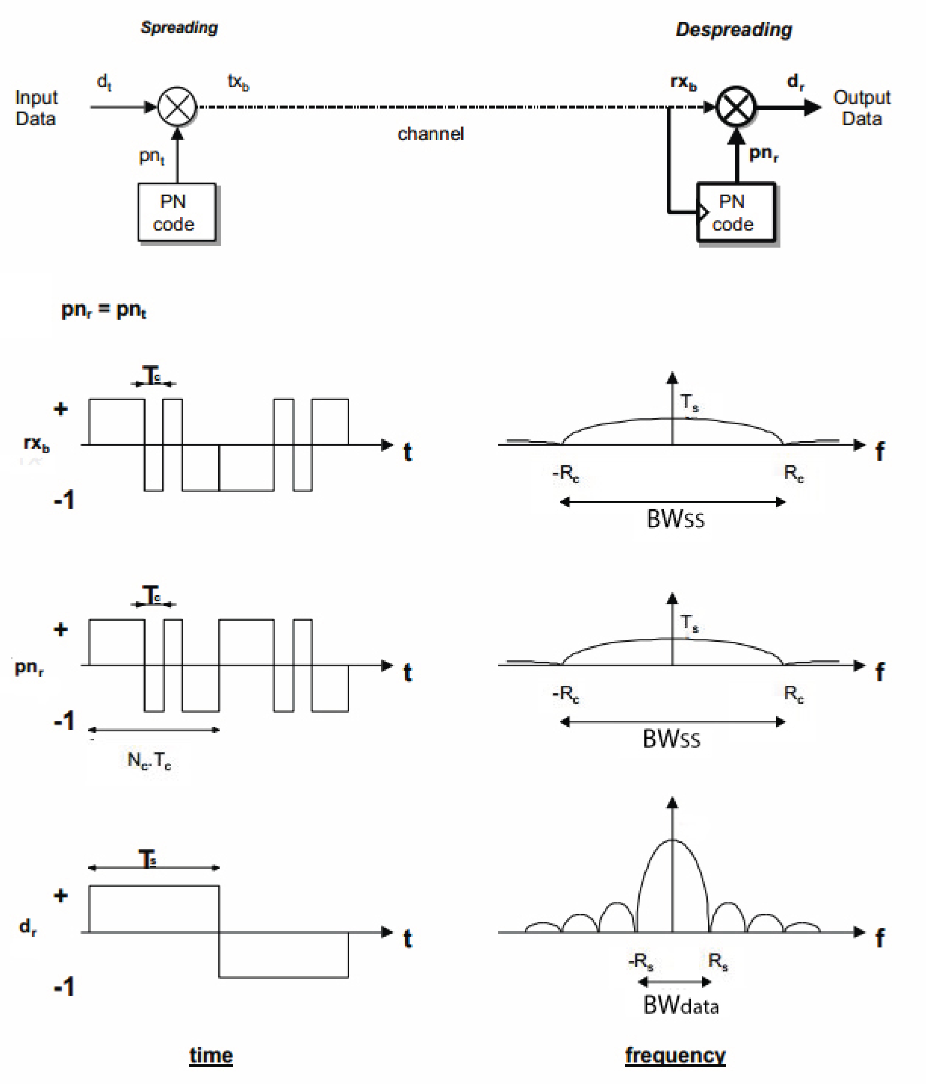

Figure 94: DSSS despreading technique[13].

The multiplied output in the receiver is then:

$$d_r = rx_b \cdot pn_r = (d_t \cdot pn_t) \cdot pn_t$$

The chip sequence, pn_t, alternates between the levels -1 and +1, so the alternation is destroyed when the chip sequence is multiplied by itself (perfectly synchronized), because:

$$pn_t \cdot pn_t = +1 \text{ for all t}$$

$$\text{autocorrelation } R_a(\tau{=}0) = \text{average } (pn_t \cdot pn_t)) = +1$$

If the chip sequence at the receiver is not synchronized properly to the received signal, the data cannot be recovered.
If the received signal is multiplied by a chip sequence different from the one used in the modulator ($pn_t \neq pn_r$), the output becomes:

[13] `http://sss-mag.com/pdf/Ss_jme_denayer_intro_print.pdf`

$$d_r = rx_b \cdot pn_r = (d_t \cdot pn_t) \cdot pn_r$$

In the receiver, detection of the desired signal is achieved by correlation against a local reference chip sequence. Therefore:

$$\text{crosscorrelation } R_c(\tau{=}0) = \text{average } (pn_t \cdot pn_r)) \ll 1 \text{ for all } \tau$$

This orthogonality property of the allocated spreading codes, means that the output of the correlation used in the receiver is proximately zero for all except the desired transmission.

F.0.3 Performance in the presence of interference

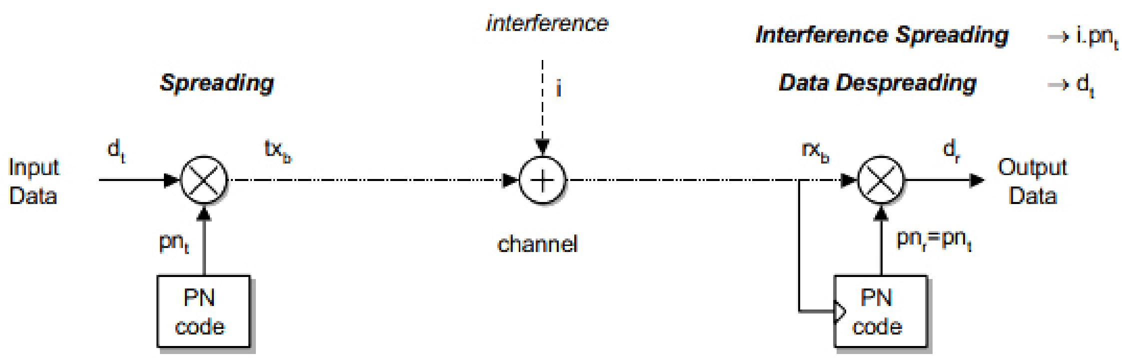

Figure 95: DSSS despreading technique in the presence of interference[14].

Acording to Figure 95, the received signal rx_b consists of the transmitted signal tx_b plus and additive interference i (noise, other users...):

$$rx_b = tx_b + \mathrm{i} = d_t \cdot pn_t + i$$

As we explained on Section F.0.2, the received signal rx_b is multiplied with the local chip sequence pn_r that is an exact replica of that used in the transmitter. So the multiplied output is therefore given by:

$$d_r = rx_b \cdot pn_t = d_t \cdot pn_t \cdot pn_t + i \cdot pn_t$$

Applying the properties of the chip sequences explained on section F.0.2, the output becomes:

$$d_r = d_t + i \cdot pn_t$$

Multiplication of the interference i by the locally chip sequence pn_t, means that the spreading code will affect the interference just as it did with the information data bearing signal at the transmitter. Noise and interference, being incorrelated with the chip sequence, increase in bandwidth and decrease in power density after the multiplier.
After despreading, the data component d_t is a narrow band signal (R_s) whereas the interference component is wideband (R_c). The effect of the interference is reduced by the spreading factor.

[14] `http://sss-mag.com/pdf/Ss_jme_denayer_intro_print.pdf`

Narrowband interference

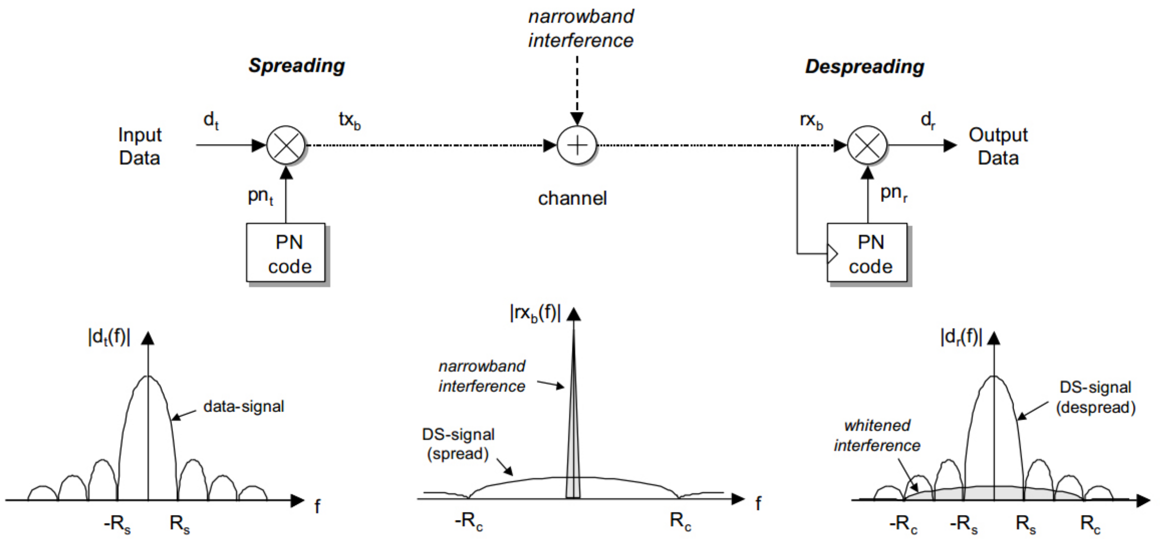

Figure 96: Narrowband interference[15].

The narrowband noise is spread by the multiplication with the chip sequence pn_r of the receiver. The power density of the noise is reduced with respect to the despread data signal. Only 1/SF of the original noise power is left in the information baseband (R_s).
The essence behind the interference rejection capability of a spread spectrum system: the usefull signal data gets multiplied twice by the chip sequence, but the interference signal gets multiplied only once.

Wideband interference
Noise or wideband interferences can be originate by:

- Multiple Spread Spectrum users. Figure 97.
- Gaussian Noise. Figure 98.

[15] http://sss-mag.com/pdf/Ss_jme_denayer_intro_print.pdf

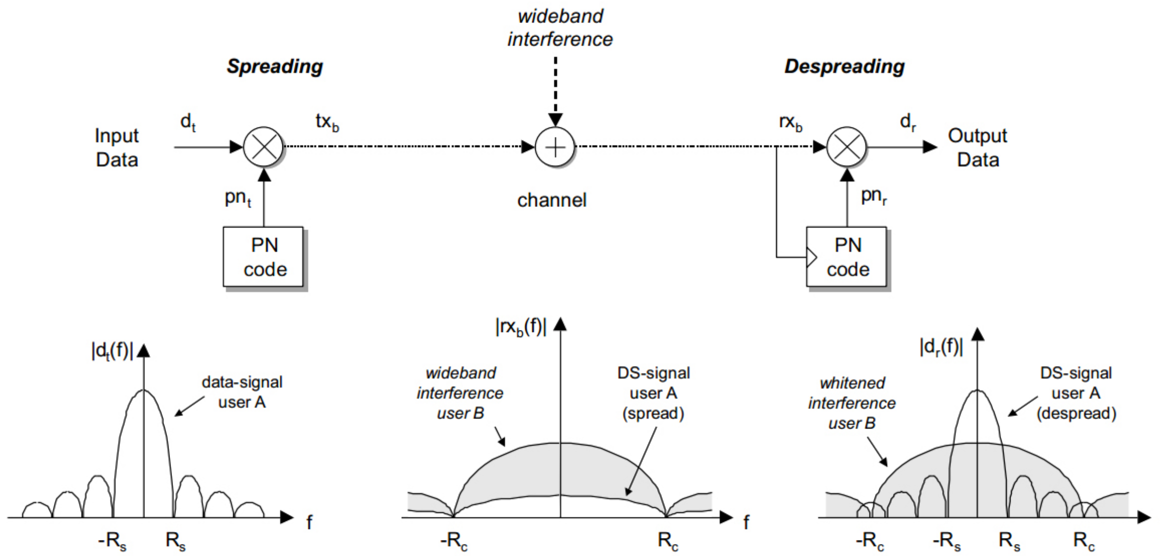

Figure 97: Wideband interference by multiply users[16].

On Figure 97, multiplication of the received data with the chip sequence of the received gives a selective despread of the data signal (smaller bandwidth, higher power density). The interference signal, coming from others users due to the shared medium, is uncorrelated with the chip sequence and is spread.

[16] http://sss-mag.com/pdf/Ss_jme_denayer_intro_print.pdf

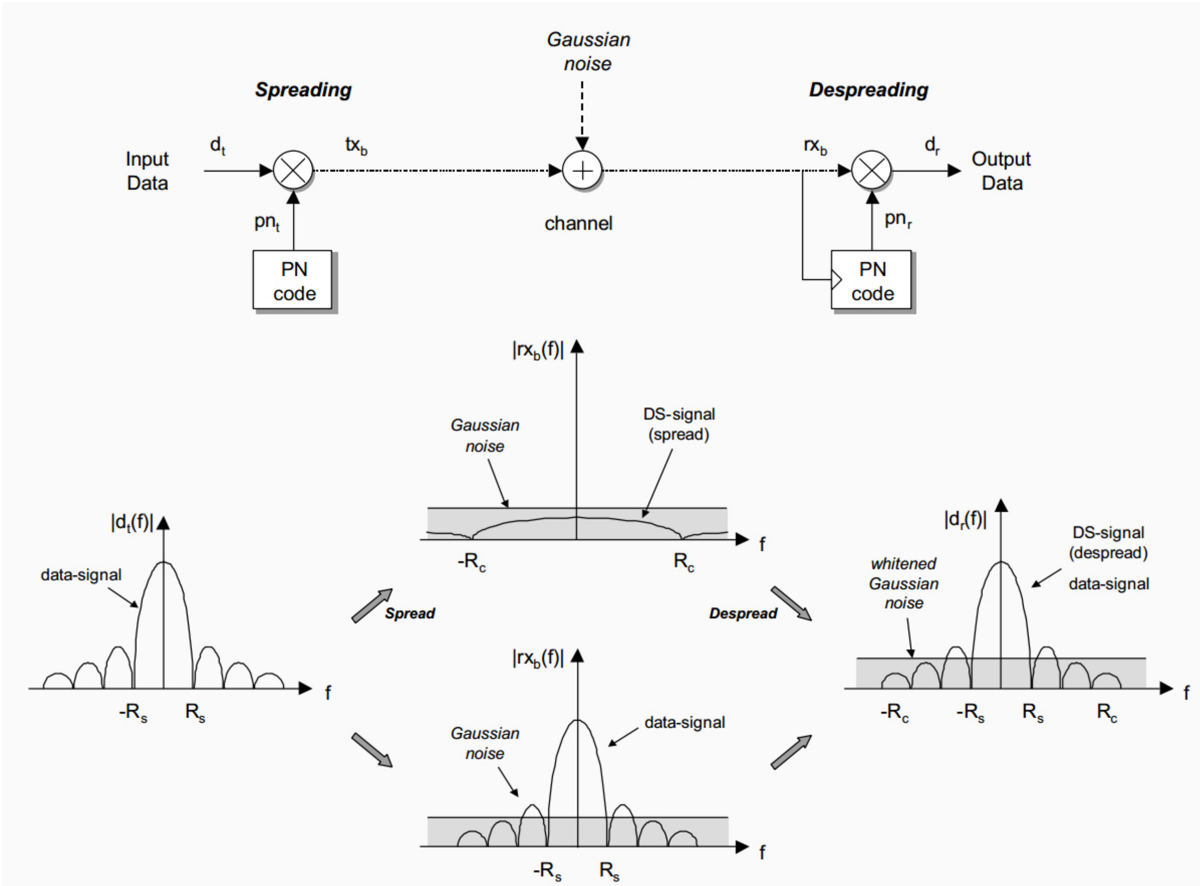

Figure 98: Wideband interference by Gaussian noise[17].

On Figure 98, there is no increase in SNR with spread spectrum. The larger channel bandwidth (R_c instead R_s) increases the received noise power with SF:

$$N_{data} = N_0 \cdot BW_{data} \rightarrow N_{SS} = N_0 \cdot BW_{SS} = N_{data} \cdot SF$$

[17] `http://sss-mag.com/pdf/Ss_jme_denayer_intro_print.pdf`

References

[1] LoRa Alliance. LoRa Alliance. https://www.lora-alliance.org/.

[2] SIGFOX. SIGFOX. http://www.sigfox.com/.

[3] NWave. NWave.

[4] On-Ramp Wireless. INGENU IS CHANGING THE IoT GAME. http://www.ingenu.com/.

[5] Telensa. Telensa. http://www.telensa.com/.

[6] LoRa^{TM} Alliance. LoRa^{TM} Alliance Members list. https://www.lora-alliance.org/The-Alliance/Member-List.

[7] Semtech Corporation. LoRa^{TM} Modulation Basics. 2015. http://www.semtech.com/images/datasheet/an1200.22.pdf.

[8] Semtech Corporation. Semtech SX1272. http://www.semtech.com/apps/product.php?pn=SX1272.

[9] Semtech Corporation. Semtech SX1276. http://www.semtech.com/apps/product.php?pn=SX1276.

[10] Semtech Corporation. Loramote end-device test application. http://iot.semtech.com/resources/LoRaMote_End-device_Test_Application.

[11] IMST GmbH. iM880A-L - Long Range Radio Module. http://www.wireless-solutions.de/images/stories/downloads/Radio%20Modules/iM880A/General_Information/iM880A_Datasheet_V1_1.pdf.

[12] Kerling. Wirnet 868 MHz. http://www.kerlink.fr/fr/produits/lora-iot-station/lora-iot-station-868-mhz.

[13] Semtech Corporation. Semtech SX1301. http://www.semtech.com/apps/product.php?pn=SX1301.

[14] Electronic Communications Committee. ERC Recommendation 70-03. http://www.erodocdb.dk/docs/doc98/official/pdf/rec7003e.pdf.

[15] Semtech Corporation. Fractional-N synthesized chirp generator, US 7791415 B2. https://www.google.com/patents/US7791415.

[16] Semtech Corporation. SX1272/73 Datasheet. 2015. http://www.semtech.com/images/datasheet/sx1272.pdf.

[17] Sornin N., Luis M., Eirich T., Kramp T., and Hersent O. LoraWAN^{TM} Specification. 2015. https://delos.imag.fr/redmine/attachments/download/204/LoRaWAN%20Specification%201R0.pdf.

[18] IMST GmbH. WiMOD iM880 Datasheet. http://www.kerlink.fr/fr/produits/lora-iot-station/lora-iot-station-868-mhz.

[19] ETSI. Final draft ETSI EN 300 220-1 V2.4.1 (2012-01). http://www.etsi.org/deliver/etsi_en/300200_300299/30022001/02.04.01_40/en_30022001v020401o.pdf.

[20] Semtech Corporation. LoRaMote USER GUIDE. 2015. http://www.semtech.com/images/datasheet/User_Guide_LoRaMote_STD.pdf.

List of Figures

List of Tables