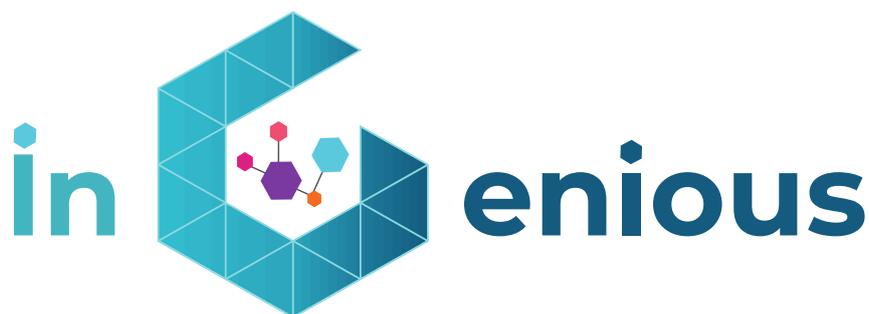




Grant Agreement No.: 957216
Call: H2020-ICT-2018-2020
Topic: ICT-56-2020
Type of action: RIA



D3.2 Proposals for next generation of connected IoT modules

Revision: v.1.0

Work package	WP 3
Task	Tasks T3.1
Due date	30/11/2022
Submission date	30/11/2022
Deliverable lead	SEQ
Version	1.0
Editors	Efstathios Katranaras (SEQ)
Authors	Efstathios Katranaras (SEQ), Olivier Marco (SEQ), Eric Poulbère (SEQ), Thomas Winięcki (SEQ), Miguel Cantero (5CMM), Manuel Fuentes (5CMM), David Martín-Sacristán (5CMM), Ivo Bizon (TUD), Roberto Bomfin (TUD), Ahmad Nimr (TUD), Raúl Lozano (UPV), Nuria Molner (UPV), Cristina Avellán (UPV)
Reviewers	Carsten Weinhold (BI), André Noll Barreto (BI), Jan Adler (BI), Miroslav Mitev (BI), Miguel Cantero (5CMM), Ivo Bizon (TUD), Nuria Molner (UPV), Efstathios Katranaras (SEQ)

Abstract	This document describes the technical work and solutions of iNGENIOUS on air interface and device connectivity for next generation Internet-of-Things (IoT) applications. In short, this deliverable targets at explaining the communication performance evaluation and analysis of IoT technologies and devices, the proposed innovations for efficient air interface and versatile communication applied to heterogeneous IoT networks, the relationship of these technical efforts to the project's architecture and use cases, as well as their exploitation potential.
Keywords	Internet-of-Things, 5G, IoT Connectivity, Cellular-IoT, Flexible HW/SW architecture

Document Revision History

Version	Date	Description of change	List of contributor(s)
V1.0	30/11/2022	EC version	See author list

Disclaimer

This iNGENIOUS D3.2 deliverable is not yet approved nor rejected, neither financially nor content-wise by the European Commission. The approval/rejection decision of work and resources will take place at the Final Review Meeting planned in July 2023, after the monitoring process involving experts has come to an end.

The information, documentation and figures available in this deliverable are written by the "Next-Generation IoT solutions for the universal supply chain" (iNGENIOUS) project's consortium under EC grant agreement 957216 and do not necessarily reflect the views of the European Commission.

The European Commission is not liable for any use that may be made of the information contained herein.

Copyright notice

© 2020 - 2023 iNGENIOUS Consortium

Project co-funded by the European Commission in the H2020 Programme		
Nature of the deliverable:		R*
Dissemination Level		
PU	Public, fully open, e.g. web	✓
CL	Classified, information as referred to in Commission Decision 2001/844/EC	
CO	Confidential to iNGENIOUS project and Commission Services	

* R: Document, report (excluding the periodic and final reports)

DEM: Demonstrator, pilot, prototype, plan designs

DEC: Websites, patents filing, press & media actions, videos, etc.

OTHER: Software, technical diagram, etc.



Executive Summary

This document describes the outcome of iNGENIOUS technical efforts on air interface and IoT devices connectivity to support Next-Generation Internet of Things (NG-IoT) scenarios.

One aspect the project explores in this document regards the performance evaluation of state-of-the-art 5G-IoT technologies. NR-based IoT connectivity is examined through system-level evaluations to understand the various KPIs that can be achieved in a factory use case environment. LTE-based IoT connectivity is examined via link-level and modem power consumption evaluations to examine performance of physical layer and modem, their dependence on network parameters, and quality of service. These evaluations are also supported by field measurements. Overall, the analysis of the performance evaluations delivers important insights and proposals for improving functionality in the domains of IoT radio connectivity and modem device design.

Furthermore, this document describes innovations that iNGENIOUS proposes for 1) efficient air interface and low complexity devices for satellite-based cellular IoT communications and 2) versatile IoT communication for realizing heterogeneous IoT networks. The former solutions are in-line with respective standardization proceedings in the area of non-terrestrial networks and have been proposed for connectivity enhancement in relevant standardization groups. The latter solutions involve innovative NG-IoT functionalities which will be implemented and showcased as part of iNGENIOUS use cases; these include flexible hardware/software architecture from the device standpoint, exploiting software defined radio platforms, and a 5G modem solution, adaptable to the needs of iNGENIOUS use cases, that will be employed to connect specific “things” to the 5G network.

The document also covers the relationship of the developments from these iNGENIOUS technical efforts to the project’s architecture and use cases, as well as their exploitation potential.



Table of Contents

1	Introduction	10
1.1	Objective of this Deliverable	10
1.2	Role of WP3 and T3.1 in iNGENIOUS	10
1.3	Relation to iNGENIOUS Architecture	12
1.4	Structure of this Deliverable	13
2	Evaluation of IoT Technologies.....	15
2.1	5G-IoT System-level Performance Analysis.....	15
2.2	NB-IoT Link-level Performance Analysis.....	28
2.3	Cellular-IoT Modem Power Consumption Analysis.....	37
3	Solutions for Efficient Cellular-IoT Air Interface	51
3.1	Impact Analysis of NTN Solutions on Device.....	51
3.2	Enhancements for NTN	58
4	Solutions for Versatile IoT Communication	66
4.1	Flexible HW/SW Architecture	66
4.2	5G Modem.....	76
5	Relation to iNGENIOUS Use Cases	79
6	Conclusion	80



List of figures

Figure 1.2-1. iNGENIOUS next generation supply chain use cases.....	11
Figure 1.3-1. Intersection of things and network layers in the iNGENIOUS cross-layer architecture.....	12
Figure 1.3-2. Device and UE domains, WP3 functionalities, and T3.1 work areas within iNGENIOUS network architecture.	13
Figure 2.1-1. Indoor Factory (InF) configurations.	16
Figure 2.1-2. Pathloss model for each InF sub-scenario.	17
Figure 2.1-3. Latency values for different distances in big-hall (left) and small-hall (right) scenarios.....	17
Figure 2.1-4. Map of the SLS scenario inside the University of Burgos.	18
Figure 2.1-5. Scenario layout for SLS.	18
Figure 2.1-6. Latency comparison mid-band (solid lines) vs. high-band (dashed lines).	21
Figure 2.1-7. Reliability comparison. Mid-band (solid lines) vs. high-band (dashed lines).	22
Figure 2.1-8. Latency comparison between mid-band and high-band for different user speeds.	23
Figure 2.1-9. Reliability comparison between mid-band and high-band for different user speeds.....	24
Figure 2.1-10. SINR (right) and SNR (left) maps for mid-band without MIMO.	24
Figure 2.1-11. SINR (right) and SNR (left) maps for mid-band with MIMO.....	25
Figure 2.1-12. SINR (right) and SNR (left) maps for high-band without MIMO.	25
Figure 2.1-13. SINR (right) and SNR (left) maps for high-band with MIMO.....	25
Figure 2.1-14. Latency and reliability values obtained for different connection densities.	26
Figure 2.1-15. Connection density for sensors in mid-band (right) and high-band (left).....	27
Figure 2.2-1. BLER vs. CNR for DS = 10 ns and different repetition number.	32
Figure 2.2-2. Gain in CNR vs. MCS for DS = 100 ns and different repetition number.	32
Figure 2.2-3. BLER vs. CNR for DS = 100 ns and different MCS cases, in-band mode vs. standalone mode.....	33
Figure 2.2-4. Schematic of the measuring system (a) and the equipment (b).	34
Figure 2.2-5. Comparison of measured RSSI values for Movistar in case of Tx0 (top) and Tx1 (bottom) antenna ports.	35
Figure 2.3-1. C-DRX states overview.....	38
Figure 2.3-2. I-DRX states overview.....	38
Figure 2.3-3. eDRX states overview.....	39
Figure 2.3-4. PSM states overview.....	39
Figure 2.3-5. Different chipset parts powered off according to usage needs.....	40
Figure 2.3-6. Leakage L and fixed cost overhead W of a device low power mode.	41



Figure 2.3-7. Modem can choose between different power modes based on expected sleep duration.....41

Figure 2.3-8. Example of a typical C-IoT modem power consumption graph..... 42

Figure 2.3-9. Power evaluations; C-DRX cycle test case (2048 ms C-DRX cycle, Good coverage, CINR>10dB).44

Figure 2.3-10. Power evaluations; I-DRX cycle test case (2.56 sec DRX cycle, Good coverage, 10dB>CINR>6dB). 45

Figure 2.3-11. Power evaluations; eDRX PTW test case (2.56 sec PTW, 1.28 sec DRX cycle, Good coverage, CINR>10dB).....46

Figure 2.3-12. Power evaluations; PSM TAU test case (180 sec TAU timer, 2.56 sec DRX cycle, Medium coverage, -3dB<CINR<6dB). 47

Figure 2.3-13. Power consumption and energy usage profile during data session with PSM – “Basic” device.....48

Figure 2.3-14. Power consumption and energy usage profile during data session with PSM – “Advanced” device.48

Figure 3.1-1. 3GPP NTN roadmap. 52

Figure 3.1-2. Mapping assessment summary of proposed NTN features with potential to affect the communication device design. 56

Figure 3.1-3. Exemplary diagram of a device architecture..... 56

Figure 3.2-1. Additional and delayed STATUS PDU. 61

Figure 3.2-2. Fast unique STATUS PDU with t-Reassembly-delay timer. 62

Figure 3.2-3. Fast unique STATUS PDU with t-Reassembly-delay timer. 63

Figure 3.2-4. Using SIB19 Target cell NTN-config for CHO.64

Figure 4.1-1. Overview of PHY modules used for baseband processing on Type-1 devices..... 67

Figure 4.1-2. Measurement setup.....70

Figure 4.1-3. Overview of PHY modules used for baseband processing on Type-2 devices..... 73

Figure 4.1-4. Throughput of GFDM modulator for different configurations of subsymbols and subcarriers.74

Figure 4.1-5. Encoder and decoder throughput performance.....74

Figure 4.1-6. Power spectral density and received symbol constellation of flexible PHY running on M³. 75

Figure 4.2-1. 5G modem..... 76

Figure 4.2-2. 5G modem architecture..... 77



List of tables

Table 1. KPIs defined in D2.1 [1] to be evaluated with SLS.	19
Table 2. System-level simulation assumptions.	19
Table 3. UEs that satisfy target latencies for each object.	21
Table 4. UEs that satisfy a reliability of 99.99%.	22
Table 5. Reliability for 10 UEs.	22
Table 6. KPIs validated with the SLS.	27
Table 7: LLS parameters.	31
Table 8. NB-IoT based device battery lifetime evaluation results for Transport and Ship use cases.	49
Table 9. Flexible PHY/MAC throughput measurements with <i>UDPtest</i>	71
Table 10. 5G modem technical specifications.	77
Table 11: Simulation environments mapped to iNGENIOUS use cases	79



Abbreviations

3GPP	3rd Generation Partnership Project
ADC	Analog-to-Digital Converter
AGV	Automated Guided Vehicle
C-IoT	Cellular IoT
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
DAC	Digital-to-Analog Converter
DL	Downlink
DRX	Discontinuous Reception
eMBB	Enhanced Mobile Broadband
(HD/FD)-FDD	(Half-duplex/Full-duplex) Frequency Division Duplex
FPGA	Field-Programmable Gate Array
GFDM	Generalized Frequency Division Multiplexing
gNB	Next Generation NodeB
GNSS	Global Navigation Satellite System
HARQ	Hybrid Automatic Repeat Request
HW	Hardware
IIoT	Industrial IoT
IMT	International Mobile Telecommunications
InF	Indoor Factory
InH	Indoor Hotspot
IoT	Internet of Things
IP	Internet Protocol
KPI	Key Performance Indicator
LoS	Line-of-Sight
LPWA	Low-Power Wide-Area
LTE	Long-Term Evolution
LTE-M	LTE for MTC
MAC	Medium Access Control Layer
MIMO	Multiple-Input Multiple-Output
mMTC	Massive MTC
MTC	Machine-Type Communications
NB-IoT	NarrowBand IoT
NG-IoT	Next-Generation IoT
NR	New Radio
NTN	Non-Terrestrial Networks
PDU	Packet Data Unit
PHY	Physical Layer
PSM	Power Saving Mode
PTW	Paging Time Window



QAM	Quadrature Amplitude Modulation
RAN	Radio Access Network
RE	Resource Element
RF	Radio Frequency
RLC	Radio Link Control
RRC	Radio Resource Control
RTT	Round-Trip Time
RX	Receive
SCS	Subcarrier Spacing
SDR	Software Defined Radio
SIM	Subscriber Identity Module
SINR	Signal-to-Interference-plus-Noise Ratio
SLS	System Level Simulation
SNR	Signal-to-Noise Ratio
SR	Scheduling Request
SW	Software
TDD	Time Division Duplex
UDP	User Datagram Protocol
UE	User Equipment
UL	Uplink
URLLC	Ultra Reliable Low Latency Communications



1 Introduction

1.1 Objective of this Deliverable

The main goal of this deliverable is to describe the technical work and solutions of iNGENIOUS regarding the air interface for internet-of-things (IoT) connectivity towards the next generation of connected IoT modules. In short, this deliverable explains the various analysis and solution *components* developed within the iNGENIOUS technical work task for IoT device evolution including: 1) performance evaluations of state-of-the-art IoT technologies, 2) suggestions and innovations, in line with standards evolution, for efficient air interface, 3) innovations and solutions for versatile communication towards heterogeneous IoT networks.

The relationship of these components to the iNGENIOUS IoT network architecture, as well as their relevance and employment, where applicable, to the realization of the iNGENIOUS use cases, are also presented.

The exploitation potential and plans for each component are summarized at the end of the respective section and further detailed in deliverable D7.3 [29].

1.2 Role of WP3 and T3.1 in iNGENIOUS

In WP3, iNGENIOUS aims to evolve both the hardware (HW) and software (SW) architectures of IoT devices, as well as their communication ability. The goal of the work package is to address current limitations and ease of adoption of devices in IoT scenarios to support several next generation supply chain use cases with diverse requirements. These use cases addressed in iNGENIOUS, plotted in Figure 1.2-1, and their definition and requirements are further detailed in Deliverable D2.1 [1].



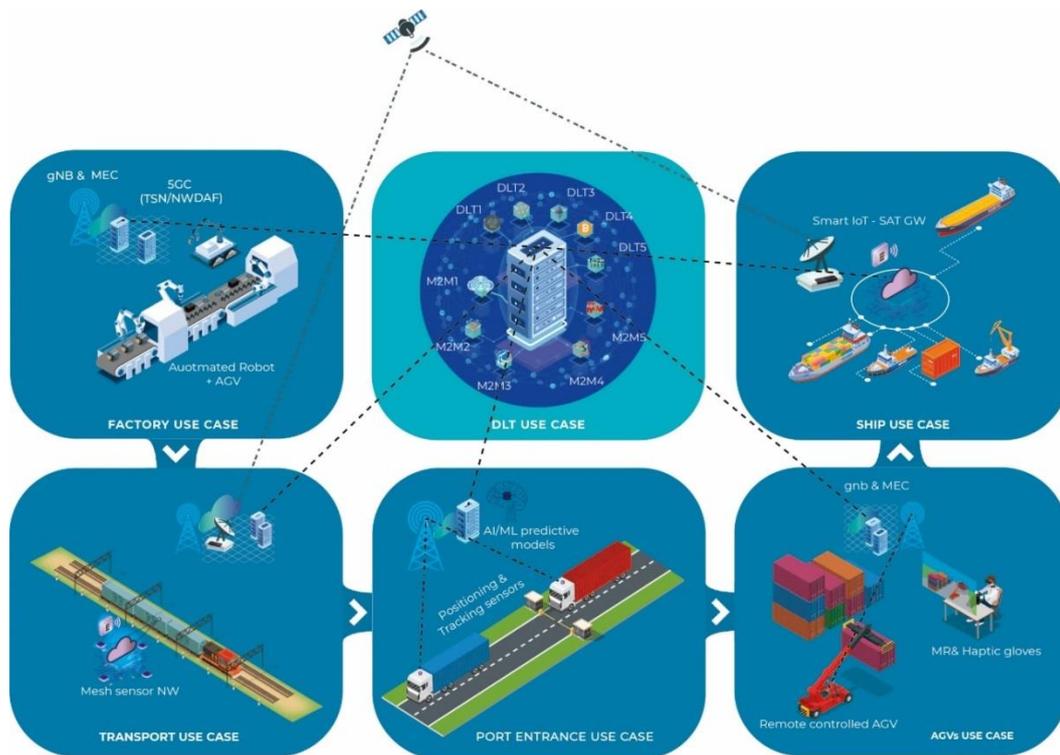


Figure 1.2-1. iNGENIOUS next generation supply chain use cases.

Accordingly, a set of objectives has been defined in WP3:

- Define and promote **evolutions of IoT connectivity**, in line with standardization frameworks, as well as to investigate innovative solutions to support the core and access technologies adopted within iNGENIOUS,
- Define **forthcoming HW and SW architectures** for immersive, tactile and more secure IoT devices, and
- Develop **proof-of-concepts** (PoC) for defined HW and SW architectures.

Essentially, three main axes of evolution are considered in WP3: connectivity, local computation, and services. Task 3.1 (T3.1) is addressing the first aspect and considers three main directions of technical work for evolution:

- Performance evaluation and analysis of IoT connectivity technologies to predict and enhance network and device performance, as well as to identify opportunities of improved radio link design for more efficient (e.g., low-power, low-cost) device connectivity.
- Air-interface enhancements to lower the cost of IoT communication (e.g., in terms of computational complexity, power consumption, latency, and flexibility), leveraging improvements discussed in the standardization bodies and investigating new innovative solutions.
- Versatile IoT communication through 1) software-defined PHY/MAC with different flexibility levels at compile-time and run-time for customization and optimization to specific IoT air-interface, as well as for supporting different traffic classes (e.g., eMBB, URLLC, mMTC), and

2) compact, flexible and customizable user equipment (UE) solution that provides 5G wireless connectivity.

T3.1 considers the requirements defined in WP2 as the basis for the research activity and coordinated closely with WP4 technical work, which focuses on radio access network (RAN) and core network (CN) aspects. Generally, the outcome of T3.1 technical work will be leveraged in prototypes to be integrated and validated into the WP6 planned trials and in research proposals submitted to 3GPP standardization body as part of WP7 efforts. This work also aspires to form a foundation for future enhancements in next-generation IoT (NG-IoT) device layer, beyond the lifetime of iNGENIOUS.

1.3 Relation to iNGENIOUS Architecture

This document describes the several analysis and solution components of iNGENIOUS for innovation within *functionalities* related to the connectivity aspect of IoT devices. These functionalities are part of the device and UE domains as captured in iNGENIOUS' network architecture, provided in iNGENIOUS deliverable D2.4 [2]. Figure 1.3-1 highlights the intersection of things and network layers within the overall architecture.

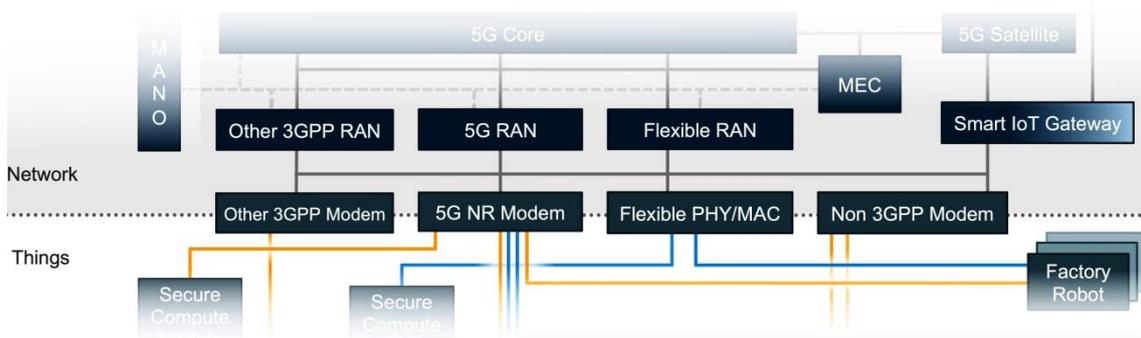


Figure 1.3-1. Intersection of things and network layers in the iNGENIOUS cross-layer architecture.

Figure 1.3-2 provides a network-centric view of the device and network components with focus on WP3 related functionalities and evolution axes. T3.1 work areas are highlighted in this figure in red to underline their relation to iNGENIOUS connectivity functionalities.

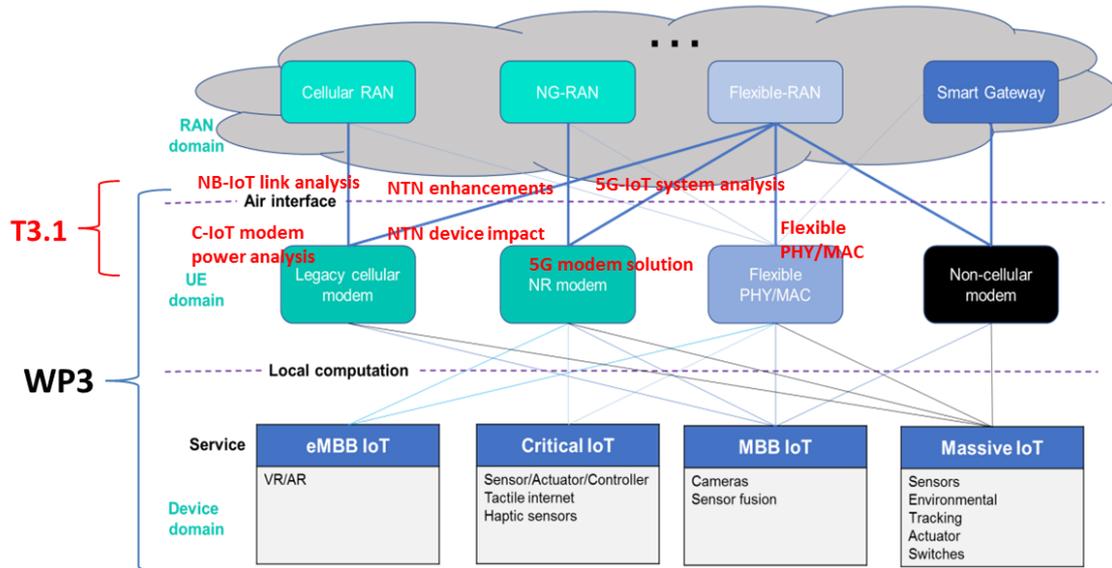


Figure 1.3-2. Device and UE domains, WP3 functionalities, and T3.1 work areas within iNGENIOUS network architecture.

1.4 Structure of this Deliverable

In accordance with the three main directions of evolution in T3.1, the work captured in this deliverable is structured as follows.

Chapter 2 captures performance evaluations and measurements of IoT state-of-the-art technologies and analyses the attained results to predict network and devices capabilities as well as opportunities for connectivity improvement:

- **5G-IoT system-level performance analysis** (Section 2.1)
- **NB-IoT link-level performance analysis** (Section 2.2)
- **Cellular-IoT modem power consumption analysis** (Section 2.3)

Chapter 3 covers air interface technology solutions, focusing on satellite-based connectivity, and presents an analysis towards IoT device-friendly features for non-terrestrial network (NTN) specification in cellular standardization framework as well as proposed solutions for efficient radio link layer design:

- **Impact analysis of NTN solutions on device** (Section 3.1)
- **Enhancements for NTN** (Section 3.2)

Chapter 4 presents the implementation developments of flexible radio architecture at UE side and 5G modem solution for realizing a heterogenous IoT network able to attain diverse and adaptable connectivity requirements:

- **Flexible HW/SW architecture at UE side** (Section 4.1)

- **5G modem solution** (Section 4.2)

In each section that covers one of the seven work components listed above, we summarize the state of the art and the importance of the efforts conducted, before explaining the innovations in detail. Each such section also includes a summary of the exploitation potential, so that the reader can better assess the relevance of the technologies and the iNGENIOUS innovations.

In addition to the brief summaries of exploitation potential, we demonstrate the applicability of the work by mapping the T3.1 work components to the iNGENIOUS use cases in Chapter 5.



2 Evaluation of IoT Technologies

This chapter presents the work within iNGENIOUS that aims to evaluate the performance of some of the current state-of-the-art technologies which are part of the iNGENIOUS project, in the context of four of the envisaged use cases: Factory, Transport, Port Entrance, and Ship. These evaluations are of paramount importance in order to predict respective network and devices performance before actual deployment as well as to identify opportunities of improved connectivity design for respective features and product development.

The performance evaluation results provided in the following sections, 2.1, 2.2 and 2.3, are based on three different simulation environments: a 5G system-level simulator developed by 5CMM, a NB-IoT link-level simulator developed by UPV, and a cellular-IoT device power model estimator developed by SEQ. For the link-level and power analysis, complementary field measurements are also performed to examine a real-life scenario deployed NB-IoT system and calibrating modem and network behaviour expectations, respectively.

2.1 5G-IoT System-level Performance Analysis

The fulfilment of technological requirements related to 5G-IoT technologies will be verified in iNGENIOUS by following a two-fold approach based on simulations and real measurements. *System-Level Simulations* (SLS) are an excellent approach for evaluating the end-to-end performance of a given network scenario and specific technology, prior to any deployment and related in-field measurements.

2.1.1 Work overview

Simulations allow us to predict network's performance before its actual deployment. The main drawback is that simulations models must be carefully selected in order to achieve sufficient accuracy so that simulation results closely match the performance achieved in real-life trials.

Trials usually generate high costs, are limited by the available human, physical and time resources, and therefore provide very valuable, but limited, amount of information. In this sense, simulations are an effective and efficient way of executing a high amount of different test variants, enabling the collection of huge amounts of data with minimal effort. Another great advantage of simulations is that technologies not yet available on the market can be evaluated.

iNGENIOUS' goal for the system-level simulation campaign is to evaluate the 5G technology. Simulation campaigns are carried out for one specific use case: "*Automated robots with heterogeneous networks*" (Factory use case), where the obtained values are compared against the requirements defined in D2.1 [1]. The simulator used is based on NS3 [4]. Specifically, a 5G-NR module called 5G-LENA [5] has been deployed and modified accordingly for this evaluation.

2.1.2 Channel model



The simulation environment is based on the indoor industrial scenario assumed for the real-world demonstration, i.e., an industrial area in the University of Burgos. This type of indoor scenario is characterized by a high clutter density with metallic elements (machinery, assembly lines, shelves, etc.). Therefore, the radio propagation conditions may differ from other indoor scenarios.

Given the increasing interest in Industrial IoT (IIoT) deployments, the 3GPP decided to develop a novel channel model for IIoT applications, different to the Indoor Hotspot (InH) channel model previously used for indoor evaluations. As a result, the 3GPP has recently defined a channel model in a new version of TR 38.901, v.16.1.0 [6], known as Indoor Factory (InF). This model is based on a set of basic assumptions such as antenna height and configurations, room size, UE distribution and speed, clutter type and density or frequency bands, among others.

In iNGENIOUS, we have studied the applicability of the 3GPP InF channel model to the University of Burgos factory scenario. Specifically, we have determined the proper parameterization of the model based on the parameters mentioned above. Note that the channel model parameters vary, for example, in mixed production areas with open spaces and storage/commissioning areas and in areas with assembly and production lines surrounded by mixed small-sized machineries.

Since the NS3 network simulator has been selected, which does not natively provide an implementation of the InF channel model, a development was carried out in the framework of iNGENIOUS, following the 3GPP specifications. The InF model specifies three basic parameters, namely, clutter density, clutter height and clutter size.

The 3GPP specification defines four different configurations that depend on two factors: clutter and antenna height. The clutter can be considered either dense or sparse, and the antenna height can be low or high. The four resulting configurations variants are therefore: Sparse-High, Sparse-Low, Dense-High and Dense-Low. These configurations are represented in Figure 2.1-1.



Figure 2.1-1. Indoor Factory (InF) configurations.

In the iNGENIOUS project, the Sparse-High (InF-SH) configuration is selected, as it is the one that better fits the considered real scenario with sparse clutter and the Next Generation NodeBs (gNB) installed in the ceiling.

2.1.2.1 InF Sub-scenarios evaluation



An analysis of each sub-scenarios of the InF channel model (InF-SL, InF-DL, InF-SH, InF-DH) is carried out in this section. The InF sub-scenarios were implemented and calibrated in NS3 in the context of iNGENIOUS [6], and in this deliverable the results of the evaluation of the sub-scenarios are shown.

The pathloss equation varies depending on the sub-scenario. In Figure 2.1-2, the results of the pathloss for different distances are shown. It is easy to see that the best sub-scenario in terms of pathloss is the InF-SH, while the worst is the InF-SL for short distances (< 23 m) and the InF-DL for long distances (> 23 m).

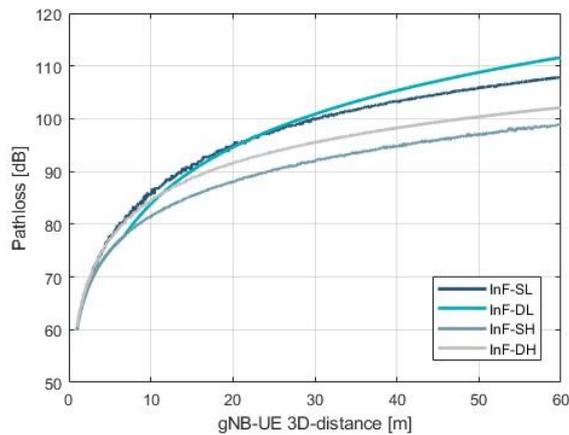


Figure 2.1-2. Pathloss model for each InF sub-scenario.

These sub-scenarios were evaluated in two layouts defined by the 3GPP [7]: the big-hall and the small-hall. These layouts were selected to observe the trends with short distances and with long distances, more concretely 20 and 50 meters inter-gNB distance for small-hall and big-hall, respectively. Simulations were run and the latency was extracted to observe the trend of this KPI for different distances. The results are shown in Figure 2.1-3. It is observed that the differences in small-hall (right) are less than in big-hall (left), where the InF-DL is clearly worst then the other sub-scenarios. In small-hall, the InF-DH and InF-DL are the worst cases in terms of latency, being both close to each other. Other conclusion is that InF-SH is the sub-scenario with best results in both layouts because of the smallest amount of industrial clutter and the largest height of gNB antennas compared to other sub-scenarios, which augments the Line-of-Sight (LoS) probability.

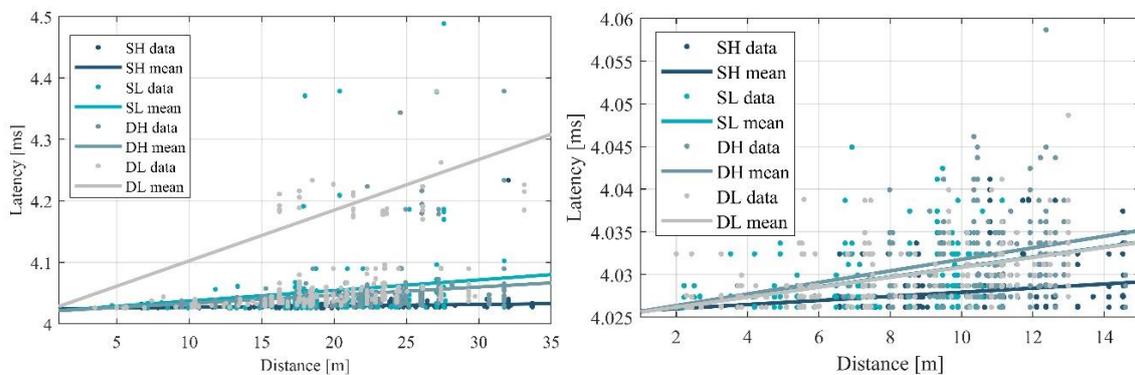


Figure 2.1-3. Latency values for different distances in big-hall (left) and small-hall (right) scenarios.



- **Sensors:** These objects are devices that produce an output signal for the purpose of sensing or measuring of a physical phenomenon and send it through the 5G network.
- **Robots:** Defined as machines that can carry out a complex series of actions automatically. In the context of the use case under investigation, they are connected automated guided vehicles (AGV) used in an industrial environment to support production lines and processes. They can be controlled or monitored through the 5G network.
- **Cameras:** Devices that transmit real-time digitally encoded video data. They can be attached to AGVs or robots and require a high data-rate.

Table 1 summarizes the KPIs and requirements coming from D2.1 that define these three types of devices and represent the objectives to reach in our SLS.

Table 1. KPIs defined in D2.1 [1] to be evaluated with SLS.

KPI	Sensors	Robots	Cameras
Throughput	0.1 Mbps	10 Mbps	25 Mbps
Reliability	99.99%	99.999%	99.999%
5G latency	1 s	10-50 ms	100 ms
Connection density	1.4 M/ km ²	10 K/ km ²	10 K/ km ²
Mobility	10 km/h		
Coverage	0.01 km ²		

Note that for cameras, D2.1 only defines a target throughput KPI. Here, the 5G latency KPI has been extracted from 3GPP TS 22.104 [8], while the rest of the KPIs for this object type have been copied from those defined for robots.

2.1.5 Assumptions

The simulations' assumptions are summarized in Table 2. For minimizing random simulation component, more than one seed are employed. The seeds do not change the simulation parameters, but the random component existing in a given simulation. This permits to extract statistically more valid results.

Table 2. System-level simulation assumptions.

Parameter	Value	
Channel model	InF SH [6]	
Room size	44 m x 23 m x 10 m	
BS antenna height	9 m	
Mobility	10 km/h (3.6 m/s)	
	Mid-band	High-band
Carrier frequency	3.5 GHz	24 GHz
Bandwidth	100 MHz	400 MHz
Numerology	1 (30 kHz SCS)	3 (120 kHz SCS)
TDD pattern	DUDU	
Flow direction	DL	
Clutter density	20%	
Clutter height	2 m	
Clutter size	10 m	



	Sensors	Robots	Cameras
Message Size	125 bytes	1250 bytes	3200 bytes
Throughput	0.1 Mbps	10 Mbps	25 Mbps
Transfer Interval	$\frac{messageSize}{throughput} * \frac{8}{1000}$		
	10 ms	1 ms	1 ms
Connection density (UE/km²)	1.4 M	10 K	10K
Connection density (UE/room)	1400	10	10

Note that, since the number of computing resources is limited, the use of the developed simulator is restricted to a specific number of UEs or resolution. The results in this deliverable include up to 40 UEs for robots and cameras, as well as 250 UEs for sensors. As it can be observed in the table above, more than 10 UEs need to be simulated for robots and cameras, and more than 1400 for sensors (see Connection density (UE/room) in Table 2). In the second case, this value was not reached due to computation limitations. In terms of resolution, there are also limitations, such as the number of 9's (i.e., the floating point precision defining the percentage of reliable communication over a large period of time) allowed when measuring reliability. The more 9's targeted, the more packets need to be simulated in an exponential way.

2.1.6 Simulation results

Simulations have been performed for both mid- and high-band frequencies, as well as for the different throughput requirements detailed in Table 1.

2.1.6.1 Latency vs. Throughput

The 5G latency is a critical KPI for most of the use cases. It indicates the network time delay between the sender and the receiver. This KPI has been evaluated in several situations or scenarios. Four different throughputs have been set, three of them associated with the devices defined in Table 1. A fourth value (1 Mbps) has been added for a more complete view of the results, as there is a big step from the sensors (0.1 Mbps) to the robots and cameras (10 and 25 Mbps respectively). Once all the throughput values are fixed, we observe the values of latency obtained for a set of users.

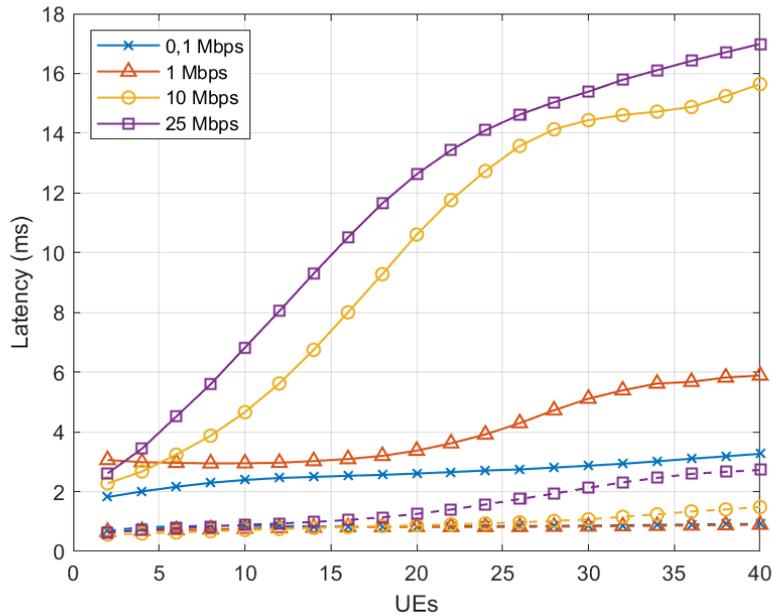


Figure 2.1-6. Latency comparison mid-band (solid lines) vs. high-band (dashed lines).

Figure 2.1-6 shows the latency obtained for the considered throughput values, for both mid-band (solid lines) and high-band (dashed lines). The simulations have been run with 20 seeds per configuration. It can be appreciated that 4 different colours are selected, each associated with a particular target throughput value.

It is observed that the more the throughput is ensured, the more latency the system introduces. The maximum latency obtained is 18 ms in a span of 40 UEs for mid-band, and 3 ms for high-band. In high-band higher numerology is employed, which results in a shorter symbol duration. This enables a faster scheduling, and thus reception, meaning less overall delay. The trend in both bands is similar, but high-band introduces much less latency than mid-band.

Table 3 provides the number of UEs that satisfies latency requirements described in Table 1 for both mid-band and high-band for all objects: Sensors (less than 1 second), Robots (less than 10 ms) and Cameras (less than 100 ms).

Table 3. UEs that satisfy target latencies for each object.

UEs that satisfy reliability (Reliability target)	0.1 Mbps (< 1 s)	10 Mbps (< 10 ms)	25 Mbps (< 100 ms)
Mid-band	>40	19	>40
High-band	>40	>40	>40

2.1.6.2 Reliability vs. Throughput

Reliability is defined as the capability of transmitting a given amount of traffic within a predetermined time duration with a particular success probability. Reliability in the SLS has been defined as the number of packets received, divided by the packets sent. This is calculated as follows:

$$Reliability (\%) = \frac{Packets\ received}{Packets\ sent} * 100$$



Figure 2.1-7 shows the reliability from 0% to 100% in axis Y on the left side, while a zoom from 90% to 100% is provided on the right part. Both graphs show mid-band results in solid lines, and high-band results in dashed lines.

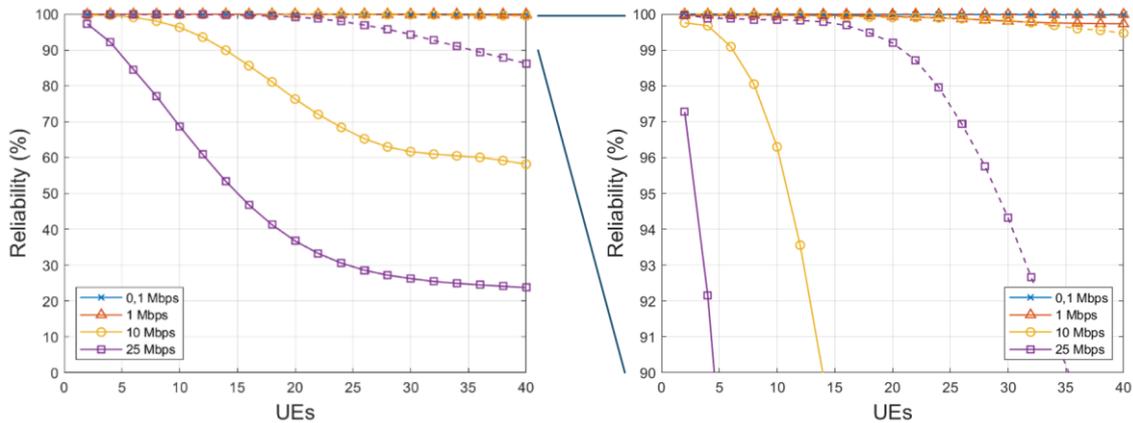


Figure 2.1-7. Reliability comparison. Mid-band (solid lines) vs. high-band (dashed lines).

It is observed that when increasing the number of users in this scenario, the reliability decreases significantly. This also occurs when increasing the throughput. For the two cases with lower data rates, the reliability is always higher than 99% for all users, while for the 10 and 25 Mbps cases, the maximum users above 99% are 7 and 0 for mid-band, and >40 and 21 for high-band, respectively.

The mid-band provides lower reliability values, meaning that the number of packets received is higher in high-band for a given number of packets sent.

All these results have been gathered in Table 4, where the number of UEs that satisfy reliability value of 99.99% is shown.

Table 4. UEs that satisfy a reliability of 99.99%.

UEs for reliability >99.99%	0.1 Mbps	10 Mbps	25 Mbps
Mid-band	14	0	0
High-band	>40	4	0

The reliability goals defined in D2.1 were 99.99% for the sensors, and 99.999% for the robots and cameras. None of them are satisfied except for the sensors. This essentially means that, with the actual network characteristics and assumptions, the requirements of reliability defined in the KPIs will not be fulfilled. In Table 5, the reliability for 10 UEs is represented for all the cases. This number of UEs has been chosen as it corresponds to the connection density KPI for robots and cameras. For sensors, further analysis will be provided in Section 2.1.6.5.

Table 5. Reliability for 10 UEs.

Reliability for 10 UEs	0.1 Mbps	10 Mbps	25 Mbps
Mid-band	99.99%	96.30%	68.71%
High-band	99.99%	99.96%	99.84%

2.1.6.3 Mobility



The mobility KPI provides information about the capability of the connected objects to move along the scenario without losing a service quality level. This KPI is a key aspect in use cases such as factories where the machinery is normally in motion.

The mobility study has been carried out for user speeds of 0, 1, 10, 30, 120 and 500 km/h, following the recommendations of IMT-2020 [9]. In this evaluation, 10 UEs with robot requirements are assumed (see Table 2). We evaluate the effect on the latency and reliability when modifying these speed values.

Figure 2.1-8 and Figure 2.1-9 represent the latency and reliability values, respectively, for the selected set of user speeds. In both cases the trend is similar. The best cases are not the 0 km/h case, but something in between: 10 km/h for the latency and 1 km/h for the reliability. This is due to the time diversity introduced by the fast fading related to such speeds. It is also observed how the results get worst when the user speed increases for 120 and 500 km/h, due to carrier shifts produced by Doppler effect.

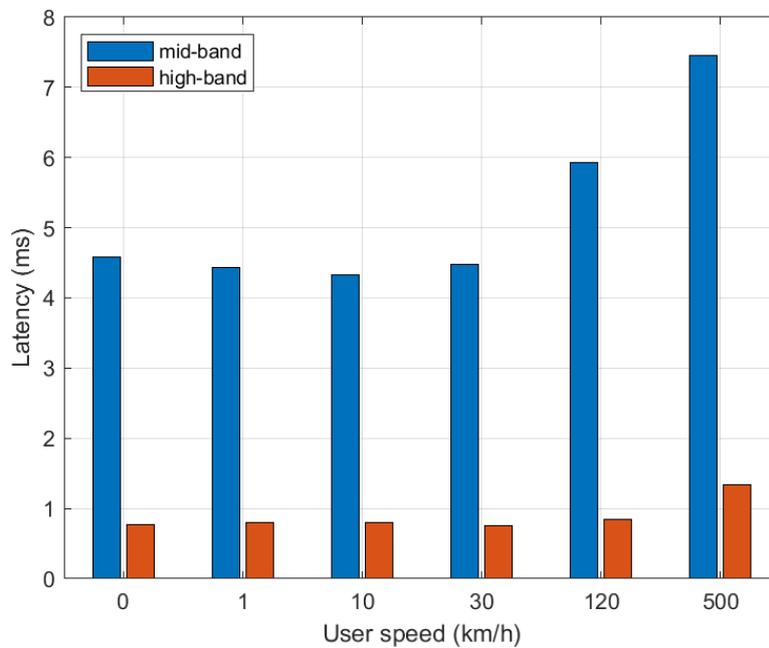


Figure 2.1-8. Latency comparison between mid-band and high-band for different user speeds.

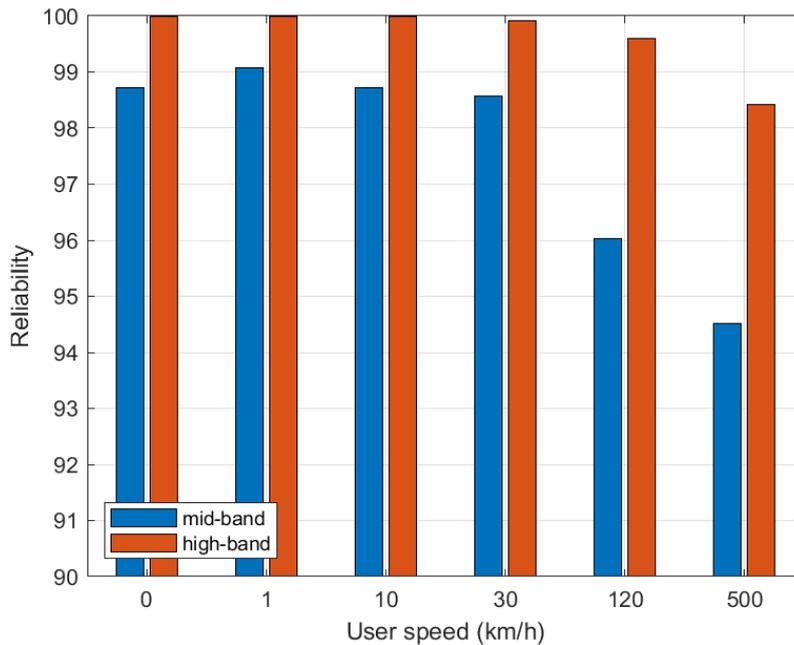


Figure 2.1-9. Reliability comparison between mid-band and high-band for different user speeds.

2.1.6.4 Coverage

The coverage KPI evaluates the service area covered by a network deployment. In D2.1, the coverage requirement was intended to cover an area of 0.01 km². However, the university of Burgos layout has been used as the scenario for the simulations, with an area of $A = b * h = 44 m * 23 m = 1012 m^2 = 0.001012 km^2$. This area will be then considered in this analysis.

To evaluate the coverage, signal-to-interference-plus-noise ratio (SINR) and signal-to-noise ratio (SNR) maps have been used. These maps calculate the SINR/SNR values in all the points across the scenario for a set of network parameters, such as the number and height of gNBs, transmission power, and channel model.

Four different cases have been evaluated in this section, corresponding to the combination of using mid-band and high-band, and single-input single-output (SISO) or multiple-input multiple-output (MIMO) configurations. Isotropic antennas have been used for mid-band and beamforming for high-band. Note that for a proper visualization of the data, different colour scales are employed. Figure 2.1-10 and Figure 2.1-11 show the results in mid-band without MIMO and with MIMO, respectively.

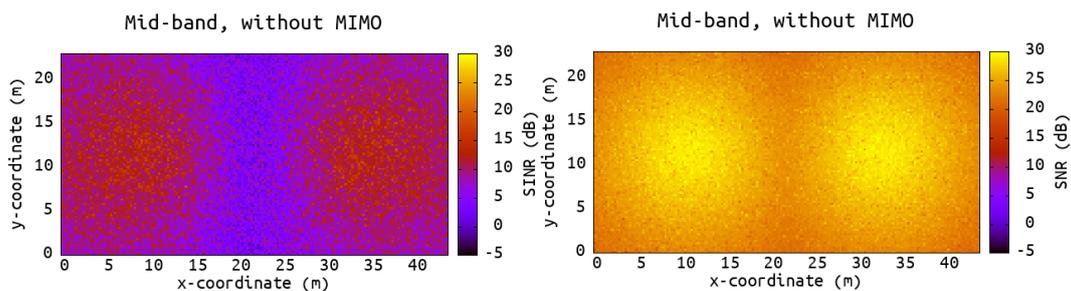


Figure 2.1-10. SINR (right) and SNR (left) maps for mid-band without MIMO.

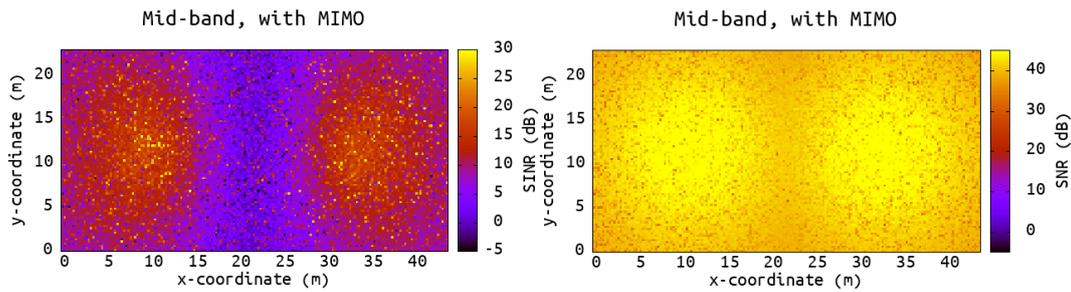


Figure 2.1-11. SINR (right) and SNR (left) maps for mid-band with MIMO.

In high-band, beamforming has been implemented, considering for the simulations that the UE is located in the right section of the map. The layout is the same as in mid-band, defined in Figure 2.1-5. The results are shown in Figure 2.1-12 and Figure 2.1-13.

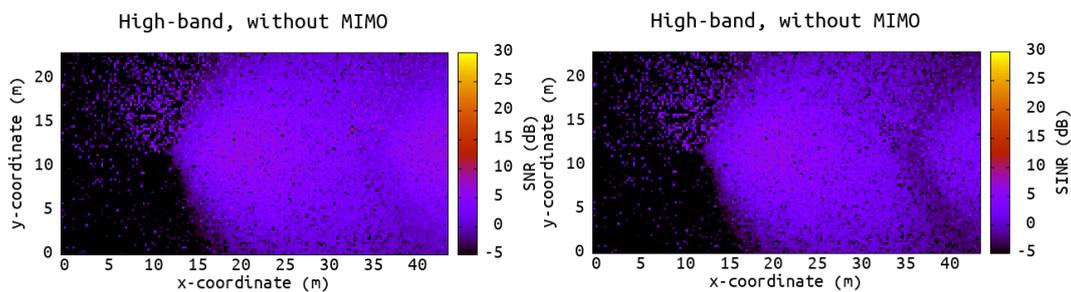


Figure 2.1-12. SINR (right) and SNR (left) maps for high-band without MIMO.

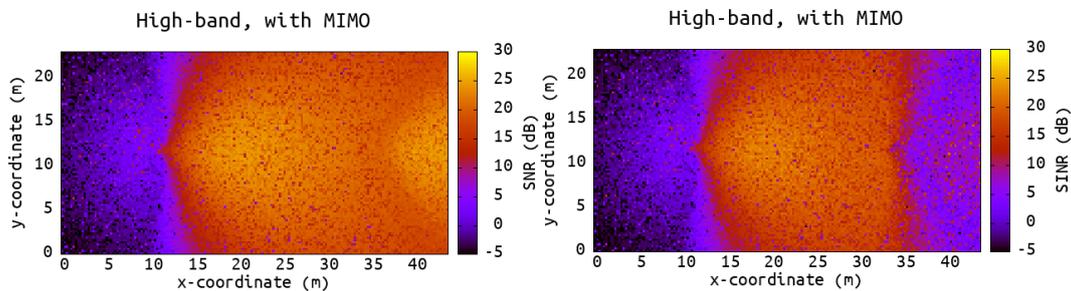


Figure 2.1-13. SINR (right) and SNR (left) maps for high-band with MIMO.

2.1.6.5 Connection density

The connection density KPI measures the density of devices that achieves a specific quality of service (QoS). The evaluation method followed has been the same as IMT-2020 [9]. In the following lines, we show the steps considered:

1. Set system user number per gNB as N.
2. Generate the user packet according to the traffic model.
3. Run non-full buffer system-level simulation to obtain the packet outage rate. The outage rate is defined as the ratio of the number of packets that failed to be delivered to the destination receiver within a transmission delay of less than or equal to 10s to the total number of packets generated in Step 2.



4. Change the value of N and repeat Steps 2-3 to obtain the system user number per gNB N' satisfying the packet outage rate of 1%.
5. Calculate connection density by equation $C = N' / A$, where the gNB area A is the scenario area. In this case, we have the following area

$$A = b * h = 44 \text{ m} * 23 \text{ m} = 1012 \text{ m}^2 = 0.001012 \text{ km}^2$$

As discussed before, there are three types of objects: sensors, robots and cameras. For robots and cameras, the connection density goal is 10K UEs per km², what corresponds approximately to 10 UEs in the evaluation scenario, calculated as follows:

$$UEs = Density * Area = \frac{10 * 10^3 \text{ UEs}}{1 \text{ km}^2} * 0.001012 \text{ km}^2 \approx 10 \text{ UEs}$$

This number of UEs has been evaluated in sections 2.1.6.1 and 2.1.6.2, where the results were explained. Figure 2.1-6 and Figure 2.1-7 show the values of 5G latency and reliability for robots and cameras for 10 UEs.

In the case of sensors, the goal is 1.4M UEs per km², what corresponds approximately to 1400 UEs in the evaluation scenario, also calculated as follows:

$$UEs = Density * Area = \frac{1.4 * 10^6 \text{ UEs}}{1 \text{ km}^2} * 0.001012 \text{ km}^2 \approx 1400 \text{ UEs}$$

Note that this amount of UEs has not been evaluated, due to computational limitations. Instead, a different evaluation has been run for this type of devices. In Figure 2.1-14, the results of the simulations are shown for up to 250 UEs, where the red plot corresponds to high-band results and the blue one for mid-band. In Figure 2.1-15 the success rate is shown.

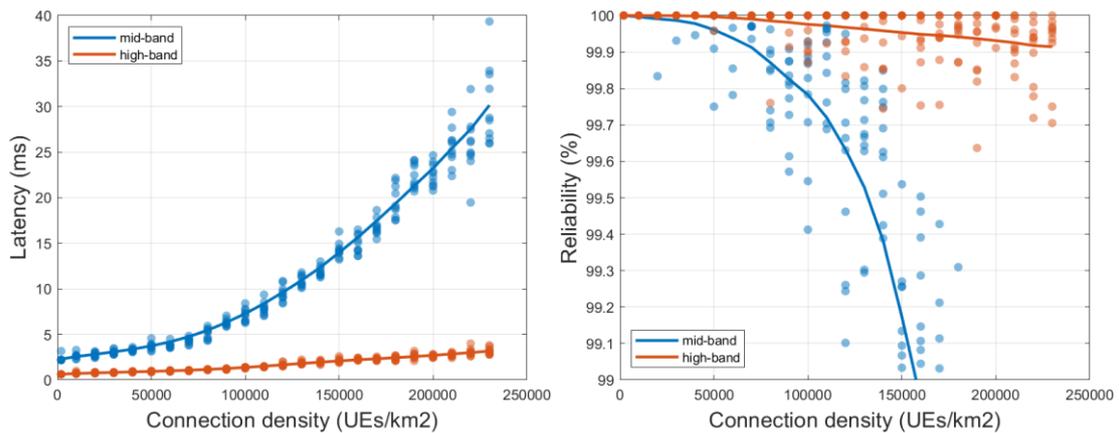


Figure 2.1-14. Latency and reliability values obtained for different connection densities.

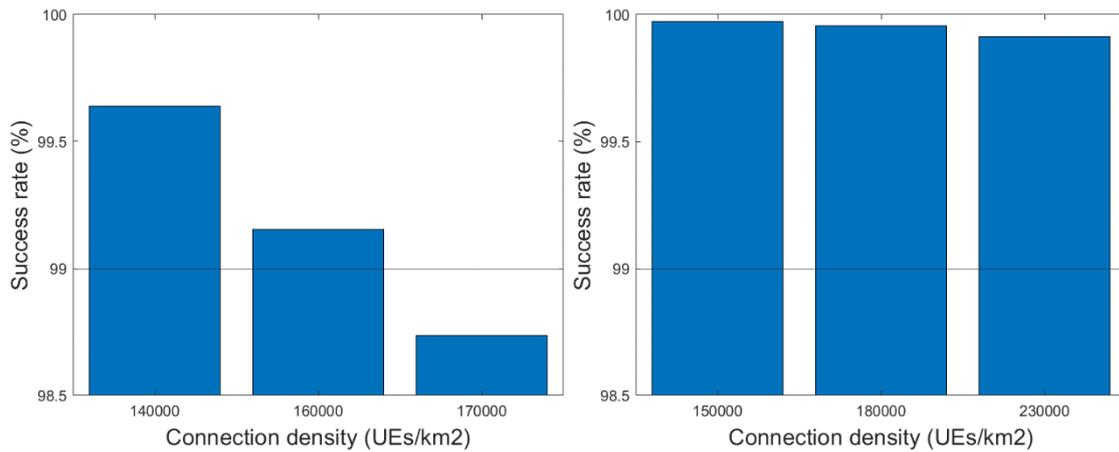


Figure 2.1-15. Connection density for sensors in mid-band (right) and high-band (left).

As shown in Figure 2.1-15, the connection density for mid-band is 160.000 UEs/km², while in high-band is more than 230.000 UEs/km². For mid-band, the value of 1.4M per km² is not satisfied. The requirements of throughput are too high for the target connection density. The more throughput, the less connection density due to the saturation of the resources available.

In the IMT-2020 analysis, the data has a message size of 32 bytes, transmitted every 2 hours per device with a bandwidth of 180kHz, which can be translated to a throughput of $0.36 \cdot 10^{-7}$ Mbps, with a connection density goal of 1M UEs/km² ([9], Section II.5.9). In iNGENIOUS, the requirement proposed for sensors is a connection density 1.4M UEs/km² with a bandwidth of 100 MHz for mid-band and 400 MHz for high-band and a data rate of 0.1 Mbps, which cannot be fulfilled in mid-band frequencies. For high-band, further analysis will be needed due to the actual limitation of the current release of the simulator. This could be a future work when new releases of the ns-3 simulator are available.

2.1.7 Summary of Achieved KPIs

Table 6 provides a summary with the values obtained for each of the KPIs defined in section 2.1.4. Different font colours are used to indicate whether the requirements are fulfilled or not. Green means that the KPI is achieved, red that it is not achieved with the actual conditions, and orange means that further analysis is needed due to simulator restrictions.

Table 6. KPIs validated with the SLS.

KPI	Sensors	Robots	Cameras
Throughput	0.1 Mbps	10 Mbps	25 Mbps
Reliability	99.99%	99.999%	99.999%
5G latency	1 s	10-50 ms	100 ms
Connection density	1.4 M/ km ²	10 K/ km ²	10 K/ km ²
Mobility	10 km/h		
Coverage	0.01 km ²		

It can be seen that almost all requirements are fulfilled. With the throughputs considered, reliability goal for sensors is achieved. In terms of latency, all values are lower than the expected ones. In terms of connection density, the goals for robots and cameras are widely fulfilled. All these results



have taken as an assumption the mobility requirement of 10 km/h. A further analysis has shown the impact of the mobility in other KPIs such as latency or reliability.

On the other hand, some KPIs goals have not been achieved. In the case of sensors, the connection density is less than 1.4M of sensors/km² in mid-band. Specifically, 160K sensors/km² is the limit. Note that more than 230K sensors/km² are permitted in high-band, although the real limit is unknown due to the limitations of the current simulator release. Reliability values for robots and cameras are not achieved. The reliability obtained with 10 UEs (connection density KPI) is 99.96% for robots and 99.84% for cameras (see Table 6), both in high-band. Although this represents a good reliability, it does not reach the 99.999% criterion established in D2.1. We recommend relaxing this KPI for the real trials and demonstrations.

Regarding coverage, an analysis for the specific scenario layout has been carried out. Good SNR results are achieved in the scenario, especially when MIMO antenna configuration is used.

2.1.8 Exploitation potential

The 5G-IoT system-level simulator has been developed by 5CMM in the framework of iNGENIOUS, as an extension of the NS3 5G-LENA module, for providing the performance analysis described above. The analysis is related to the Factory use case. This simulator will become part of their portfolio, being integrated into more elaborated planning tools. Further extensions by means of a tight collaboration with the 5G-LENA (simulator used as a basis) community are also expected.

The final information about exploitation of this product will be available in D7.3 “Final dissemination, standardisation and exploitation” [29].

Component: 5G-IoT system-level performance analysis
Relevant for use cases: Factory
Exploitation Potential: <ul style="list-style-type: none"> • 5CMM will make use of the developed system-level simulator in future projects and collaborations. Thanks to the extension for 5G-IoT technologies and the Indoor Factory channel model included, we will be able to provide extensive performance analyses for different scenarios and innovative use cases. We plan to include this simulator as part of our main portfolio, as part of our 5G Private Network Planning (PNP) tool, already available for performing extensive analyses based on 3D scenarios.

2.2 NB-IoT Link-level Performance Analysis

NB-IoT as a 5G-compliant technology has become an important asset for smart cities, smart metering and smart agriculture (waste management, detectors, parking control, low-end sensors, etc.). A comprehensive NB-IoT state-of-the-art analysis has been provided in deliverable D3.1 [10].



The overall objective of conducting a performance analysis of NB-IoT can be divided into two parts. The first part centers around building a NB-IoT Link Level Simulator (LLS) using MATLAB examining the physical layer performance and its dependence on network parameters such as the Carrier-to-Noise Ratio (CNR), the Modulation and Coding Scheme (MCS) order, the operation mode, the delay spread (DS) and the number of repetitions, in the case of transmitting user data. The second part discusses the real-life scenario of a deployed NB-IoT system in area of the Port of Valencia and includes performing field measurements in order to evaluate coverage and the RF conditions of the connection as one of the basic metrics of quality of service, by measuring parameters related to synchronization and reference signals.

2.2.1 Work overview

Link Level Simulators represent powerful tools that allow us to obtain different metrics of the system that would move closer from the theoretical margins to the results we could obtain by performing practical measurements.

In this work we focused on the repetition process and the parameters directly connected to it, which are just a few out of many possibilities the modified simulator allows us to work with. However, the main idea behind the LLS adapted for NB-IoT is to build a space in which we can try different algorithms in specific blocks of the transmission chain, modify old ones or implement new ones, and follow the response of the system through different metrics (e.g. BLER values for specific CNR). Further details can be found in [1].

As we have shown through our work, it is possible to add blocks to the chain (as is the case of a repetition combiner at the receiver side), and see how this affects the transmission by evaluating different parameter values. The obtained results can be taken into account when comparing with other solutions of repetition combining (as the receiver design is left to the manufacturer).

2.2.2 NB-IoT physical layer

NB-IoT reuses the LTE design, so unlike other IoT technologies which require the deployment of new infrastructures, NB-IoT can be easily implemented in the existing LTE infrastructure with some changes. These changes include: *(i)* a fixed transmission bandwidth of 180 kHz, *(ii)* the difference in waveform generation formula of Single Carrier Frequency Division Multiple Access (SC-FDMA), *(iii)* the resource element mapping, *(iv)* the transmission pattern of synchronization signals, and *(v)* repetitive transmissions.

As LTE is designed to provide data rates of up to 100 Mbps, NB-IoT has the goal of adapting it for very low throughput services that make up the most of IoT. Its narrow bandwidth also allows for different operation modes, based on the frequency spectrum used. In the case of in-band and guard-band modes, the NB-IoT signal occupies one Physical Resource Block (PRB) from the LTE bandwidth, thus allowing the benefits of cost savings and ease of integration, while the standalone mode occupies the bandwidth of a carrier



other than LTE, using a new frequency carrier, usually the liberated spectrum of GSM. Both Frequency Division Duplexing (FDD) and Time Division Duplexing (TDD) are supported (as of Release 15). The framing principle used in NB-IoT has been completely inherited from LTE, some differences to channel and signal mapping. The subcarrier spacing is fixed to 15 kHz in downlink, with 7 OFDM symbols per slot and normal cyclic prefix (CP) duration. Modulation techniques available are QPSK and 16QAM (as of Release 17).

This work will focus on Narrowband Physical Downlink Shared Channel (NPDSCH) as the channel used to transmit data, Narrowband Physical Broadcast Channel (NPBCH) as the channel used to transmit Master Information Block (MIB), Narrowband Reference Signal (NRS) used for channel estimation at UE side, Narrowband Primary Synchronization Signal (NPSS) and Narrowband Secondary Synchronization Signal (NSSS) as signals used for time and frequency synchronization and cell identification.

2.2.3 MATLAB-based link-level simulations

Several modifications and implementation of new features inside the MATLAB LLS - based on the Vienna 5G LLS [12]- were introduced to perform the simulations. Some of the most notable modifications are: convolutional codes for channel coding, subcarrier spacing fixed to 15 kHz, PRBs fixed both in frequency (12 subcarriers) and time (1 slot), and channel estimation procedure implemented to be done according to Cell-specific Reference Signals (CRS) i.e., NRS used only for NB-IoT and a new block allowing the repetition process has been implemented.

As LLS is able to evaluate the performance of the physical layer without instructions sent from the higher layers, it is assumed that (i) the receiver and the transmitter are synchronized, (ii) the connection is evaluated between one transmitter and one receiver, both equipped with one antenna, (iii) the duplexing method is FDD, (iv) the modulation technique is QPSK, (v) the channel model is TDL, specifically TDL-A which is a Non Line of Sight (NLOS) model with the Doppler frequency shift set to 5 Hz. The evaluated number of subframes is 30,000 per simulation, where 1 subframe is generated per iteration. A simulation iteration is initiated by specifying the MCS, the repetition number, the DS value and the starting CNR value. The CNR starts for all of the regarded cases at -10 dB, increasing with a step of 0.1 dB in case the stopping condition has not been reached. The stop condition is defined at Block Error Rate (BLER) smaller than 0.1% meaning that out of 30,000 collected subframes, less than 300 have to be erroneous in order to end the simulation process for the specified parameters. Otherwise, the CNR value is increased, and the simulations are run again. In case of a repetition number larger than 1, the total number of generated subframes is 30,000 multiplied by the number of repetitions. The repetitions are combined in the combiner and regarded as if one subframe had been sent. The average time of simulating one MCS for a specific number of repetitions and DS value was around 120 hours. Table 7 collects a summary of the parameters mentioned above.

The results have been evaluated by calculating BLER as it meticulously reflects on the RF conditions and the level of interference and results to be



one of the key metrics influencing receiver sensitivity, download throughput, and radio link monitoring indications.

The analysis is centered around the more realistic case of the TDL channel model and a repetition number of 16 and higher. The simulator allows for many different parameters to be changed as it is no longer a black box but a transparent simulation tool.

Table 7: LLS parameters.

Parameter	Value
Number of subframes	30,000
CRC value	24 bits
Modulation and Coding Scheme (MCS)	QPSK
Channel model	Tapped Delay Line (TDL) – TDL-A
Doppler frequency	5 Hz
Delay spread	10, 100, 300, 450, 1000 ns
Channel estimation model	Least Square (LS)
Demapper	Maximum Likelihood (ML)
Duplexing method	FDD
CNR start	-10 dB
CNR evaluation step	0.1 dB
Stop criteria	Block Error Rate (BLER) < 0,1%
Transmission mode	Multi-tone
Operation mode	Standalone
Subcarrier spacing	15 kHz (downlink)
Multiplexing	OFDM, 7-symbol slots (no extended CP supported)
Bandwidth	180 kHz
Number of repetitions	16, 32, 64, 128

Figure 2.2-1 depicts a graph of MCS 0 and 13 going through different repetition processes (16, 32, 64, 128 repetitions), with the x-axis representing the CNR in decibels (dB), while the y-axis represents the BLER values expressed in a logarithmic scale. The DS value is set to 10 ns and the operation mode is standalone. First observation collected from the graph is that increasing the MCS order results in higher CNR needed to obtain the same BLER. Second, that increasing the number of repetitions the curve slope gets steeper meaning the needed CNR to acquire the wanted BLER decreases i.e., the signal decoding conditions get better with each retransmission of the data bits.



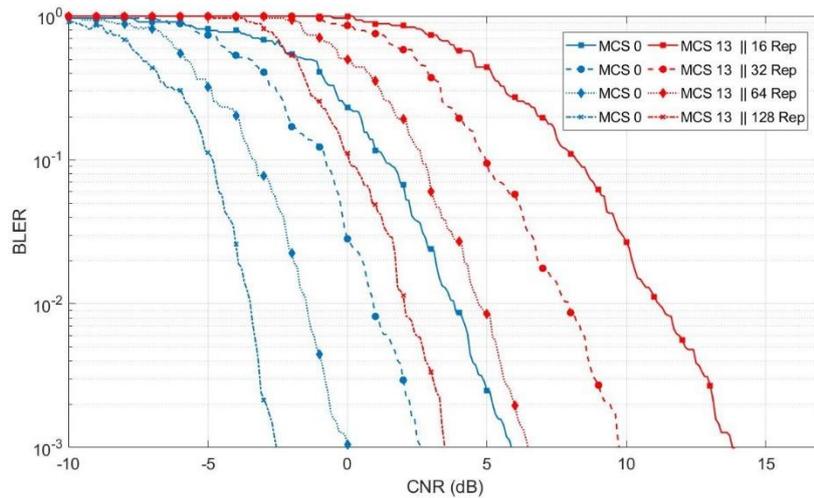


Figure 2.2-1. BLER vs. CNR for DS = 10 ns and different repetition number.

In this case, using 128 repetitions with MCS 13 results in 0.1% BLER for CNR of 3.5 dB, while using 16 repetitions with MCS 0 the CNR value goes up to 5.9 dB. Hence, we can observe that at the expense of longer processing time that is the product of increasing the repetitions number, we can send more data bits and achieve low BLER with low CNR values. However, we must consider how it will affect the spectral efficiency. Although the increase of MCS order increases the spectral efficiency, higher repetition values decrease it, so we must consider how these combinations affect the gain in spectral efficiency.

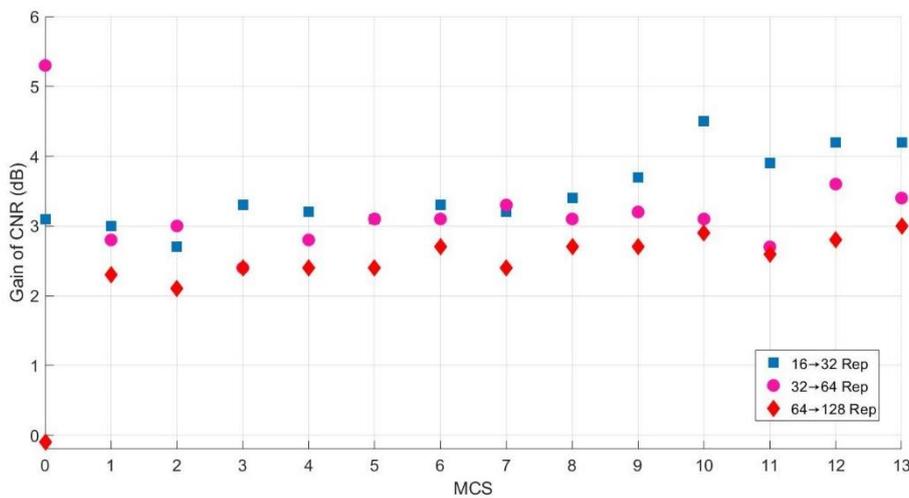


Figure 2.2-2. Gain in CNR vs. MCS for DS = 100 ns and different repetition number.

As far as the gain acquired with each increase of the repetition value is concerned, we can analyse the values shown on Figure 2.2-2, with the x-axis representing different MCSs. We can gather there is always some gain when we double the number of repetitions, however, this gain is (i) not constant, (ii) decreases as number of repetitions increases for each MCS, but (iii) tends to increase with the increase in the order of MCS.

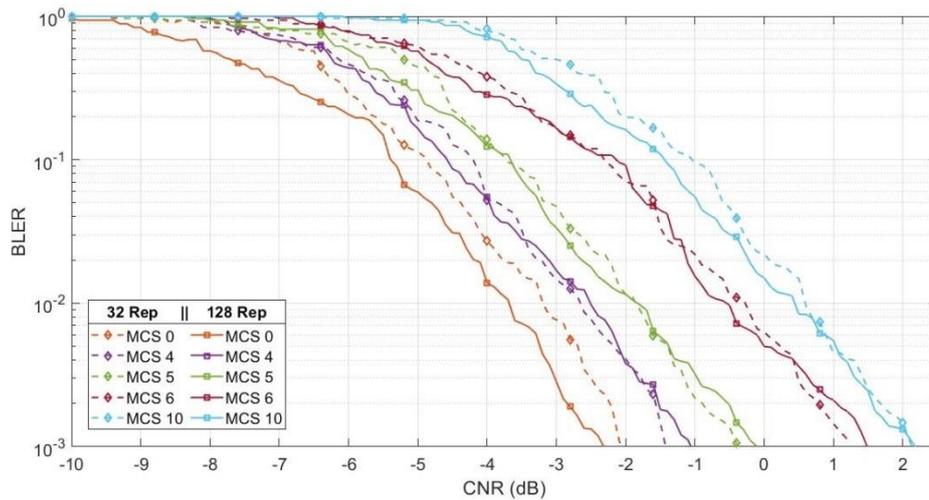


Figure 2.2-3. BLER vs. CNR for DS = 100 ns and different MCS cases, in-band mode vs. standalone mode.

Figure 2.2-3 plots the two cases of operation modes: in-band and standalone, where the DS is set to 100 ns as the recommended nominal value. The in-band mode with 32 repetitions is represented with the full line and the standalone mode with 128 repetitions is represented with the dotted line. This graph confirms the assumption of similar behaviour of one MCS in the two different operation modes (each characterized with a different repetition number). Based on it, we can conclude that the gain we get from increasing the number of repetitions 4 times in standalone mode is roughly the same as the gain we get from switching from stand-alone to in-band mode. Due to the higher number of reference signals, in-band mode shows better channel estimation performance, at the expense of a smaller number of REs available for data bits (meaning we can use MCS 10 as the highest order MCS and send 144 bits carrying information instead of MCS 13 which allows us to send 224 data bits). However, in-band mode does not require high number of repetitions, and for the same TBS offers better results than the standalone mode. It is also noted that although guard-band mode has been implemented in the simulator, its performance has not been specifically evaluated in this work, assuming it would provide similar overall results as standalone mode (same frame structure, i.e., the number of reference signals is the same, and that is what affects the channel estimation the most).

We have also analysed the effect of DS value and discovered that changes do not strongly impact the decoding of the signal as long as it is not close to the critical value, which corresponds to the CP duration of the OFDM symbol. The loss for DS smaller than 450 ns (normalized) is almost negligible, while for higher values it is noticeable more in the case of using higher repetitions (such as 64 and 128), but no higher than 3 dB, i.e., relatively small in comparison to the gain from the repetitions.

2.2.4 Field measurements

Field measurements were performed using the Rohde and Schwarz (R&S) TSME6 autonomous mobile network scanner in order to collect the signals and ROMES4 drive test software to process the data and calculate the statistics in the area of Port of Valencia. Regardless of the many bands



specified in the 3GPP releases, in Spain the NB-IoT is implemented in band number 20, with the uplink frequency range going from 832 MHz to 862 MHz and the downlink frequency range going from 791 MHz to 821 MHz. The scanner enables the measurements of power and Carrier-to-Interference-and-Noise Ratio (CINR) of the synchronization signals used for coverage measurements and different metrics regarding signal strength and CINR of the reference signals used to determine the RF conditions of the wireless connection. The measurements were performed in the outdoor area around the Port of Valencia and the schematic with the used equipment can be seen on Figure 2.2-4.

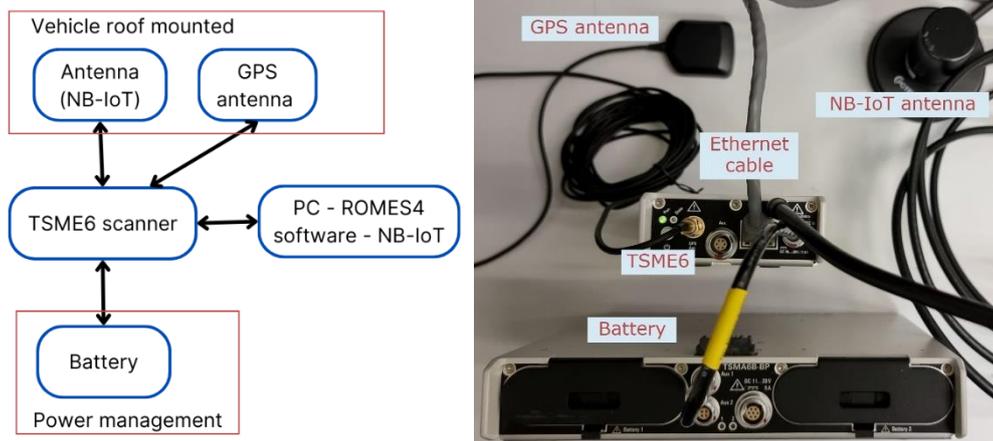


Figure 2.2-4. Schematic of the measuring system (a) and the equipment (b).

After decoding the MIB and System Information Block (SIB) broadcast messages, we obtain the network configuration related data. Based on this, we can analyse the metrics of interest for different providers (in this case 3), frequency bands (in this case only band 20) and corresponding cells.

An example of the measurements is shown on Figure 2.2-5 for one mobile provider using antenna port 0 and port 1, where RSSI represents the measured value of the entire received power including the wanted NRS and the unwanted power components present due to interference and different sources of noise. It is indicated by a negative dBm value, and the closer the measured value is to 0 dBm, the stronger is the signal, therefore, the better are the connection conditions.

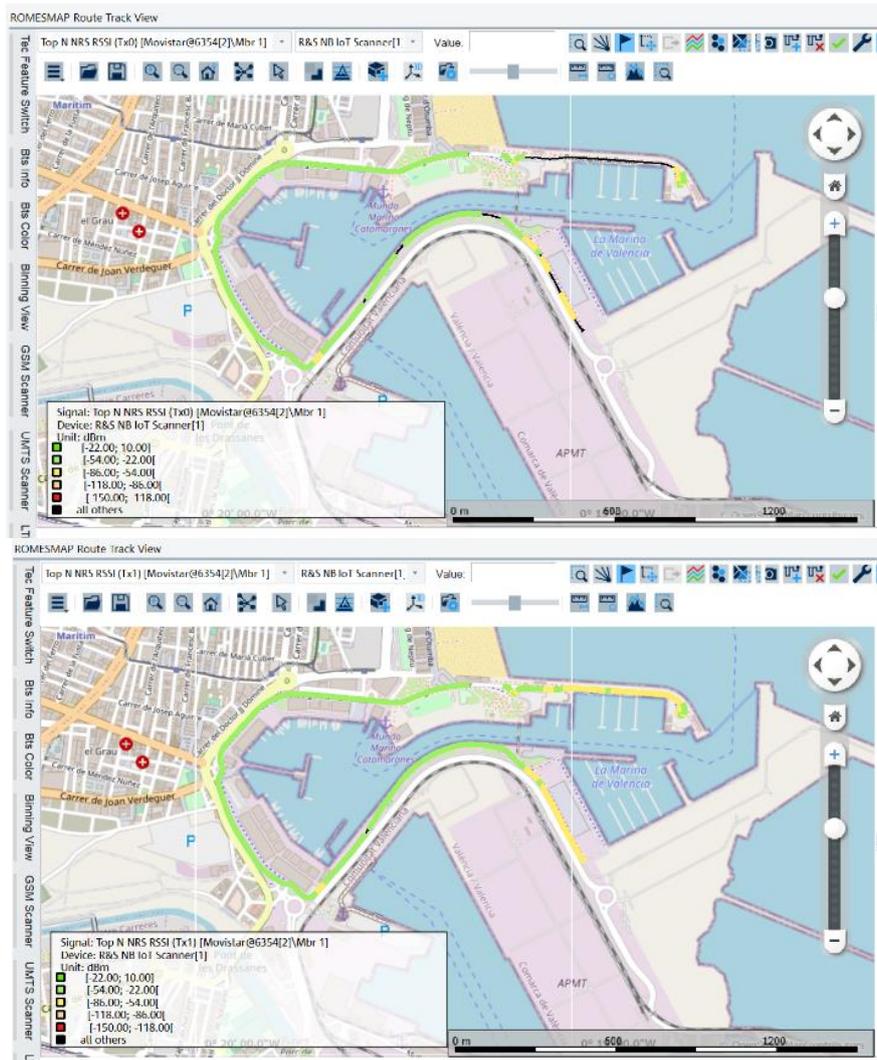


Figure 2.2-5. Comparison of measured RSSI values¹ for Movistar in case of Tx0 (top) and Tx1 (bottom) antenna ports.

We can distinguish areas in the route for the antenna port 0 (Tx0) where no signal has been detected, meaning that the conditions were too poor, i.e., the noise and the interference were too high and signal decoding was not possible. However, just by switching to the antenna port 1 (Tx1) these areas have detected and measured the RSSI value, showing rather good values (higher than -86 dBm). The reason why results differ when using port 0 or 1 is because they have different NRS positions in the frame. We can see that, even in the case of port 1, the signal strength is not the strongest, i.e., noise and interference levels are high in this area. We can only assume that they affect more the NRS of port 0 and, thus, we get poor channel estimation compared to port 1. Thus, we can observe that instead of trying different techniques to increase the CINR of the signal, the UE can take advantage of frequency/time diversity and just switch the transmission to another antenna port to obtain a sufficient CINR for a good quality service, as the signal degradation effects differ among resource elements.

¹ NRS RSSI [dBm]: Green line: [-22; 10], Lighter green line: [-54; -22], Yellow line: [-86; -54], Orange line: [-118; -86], Red line: [-150; -118], Black line: all others.

The coverage measurements performed produce the following results: regarding the sync signals power all three providers produce satisfactory values, as the measurements do not go below -86 dBm, taking into account the minimum reference sensitivity level defined for NB-IoT UEs is at -108 dBm; however, when analysing the CINR values, NPSS shows better values than NSSS (in some cases difference is 10 dB). The CINR between different providers differs as well, meaning that even though we have strong enough signals, the noise and interference levels are rather high in certain areas, and that they will undoubtedly degrade the quality of our signal. Similar results have been obtained with regards to measurements of NRS metrics. Signal power is adequate in the observed area, however certain parts show degradation due to interference and noise, which decreases the overall quality.

2.2.5 Conclusions

From the analysis performed in this work, some key conclusions are observed.

First, the repetition process strengthens the values of channel estimation, resulting in a lower CNR needed. With the appropriate combination of a high order MCS and a high enough repetition number, we can balance the spectral efficiency gain/loss with the provided CNR, whilst maintaining the same BLER. This can prove to be very useful in case of network optimization.

Second, the gain obtained from increasing the number of repetitions is not constant but decreases with each doubling of the repetition number. However, it shows higher values for high order MCSs compared to the low order MCSs. The channel conditions do not pose a loss higher than 3 dB in CNR as long as the values of the delay spread are close to the CP duration.

Furthermore, the additional reference signals found in in-band mode compensate for the high number of repetitions in the standalone case. The field measurements have shown that having a strong signal is a necessary but not sufficient condition for a permanent successful connection or good coverage. As it often happens in wireless communications, the channel conditions impact greatly the decoding of the signal, therefore additional steps must be taken in order to avoid interference and noise. One of possible solutions proved noteworthy is using another antenna port for frequency/time diversity.

2.2.6 Exploitation potential

NB-IoT performance analysis relates mainly to the Port Entrance use case where NB-IoT connectivity has been analysed and can be well applicable.

Exploitation will focus on using the developed LSS as a stepping stone for further practical experiments, especially for NTN oriented scenarios. Final information about exploitation will be available in D7.3 "*Final dissemination, standardisation and exploitation*" [29].



Component: NB-IoT link-level performance analysis

Relevant for use cases: Port Entrance

Exploitation Potential:

- **UPV:** As the link-level simulator allows many parameter variations and modifications for each block of the chain, it can be used to compare with the results obtained from the implementation of other systems as for example the implementation of an NTN channel model. The simulator can be exploited to perform further analysis and studies.

2.3 Cellular-IoT Modem Power Consumption Analysis

To understand IoT devices energy usage and assess energy-related KPIs such as battery lifetime depending on differently configured use cases, it is paramount to examine the contribution of the communication device to the energy usage of the total UE solution.

2.3.1 Work overview

A realistic power consumption model for C-IoT modem devices has been developed by SEQ in order to: *i)* obtain insight for evolution of power saving mechanisms, and *ii)* enable battery lifetime KPI assessment of typical C-IoT modem to be used within specific iNGENIOUS use cases, i.e. Transport and Ship use cases.

In this section, we first provide a brief background of the main 3GPP power saving modes that drive the energy efficiency in legacy C-IoT devices currently in market. Next, we discuss some basic principles and metrics for saving energy in device end. Finally, we present evaluation results focusing on Cat-NB devices (based on NB-IoT technology), which are relevant to Transport and Ship use cases of iNGENIOUS, and discuss the basic insights obtained.

2.3.2 C-IoT power saving modes

In the following, we provide the basic principles of the various 3GPP power saving modes that are currently used by C-IoT devices in industry.

2.3.2.1 C-DRX

Connected discontinuous reception (C-DRX) is a legacy mechanism (since initial LTE Release 8) designed basically for broadband LTE. It involves network-controlled configuration of sleep cycle and receiver ON duration. During inactivity timer period, the modem receiver is always on, thus, consumes a lot of energy. The ON duration between long cycles of sleep is the same as for the case of short cycles, e.g., device monitors scheduling messages during 10 ms ON periods once every 160 ms. With no traffic, the modem can sleep outside of ON periods (see Figure 2.3-1). It becomes evident that the C-DRX design considered no integration between applications and modem. This makes it compatible with unpredictable traffic



patterns and C-DRX offers an adaptive automatic scheme, well suited for this context. C-LoT technologies (LTE-M / NB-IoT) target primarily LPWA applications. C-DRX is not currently supported on all such networks, or it even may be bypassed (direct entry in Idle mode) using the 3GPP Release Assistance Indication (RAI) feature, which allows a device to indicate to the network that it has no other uplink data and is neither expecting to receive other downlink data. However, C-DRX can still be very useful and used for broadband-like IoT applications.

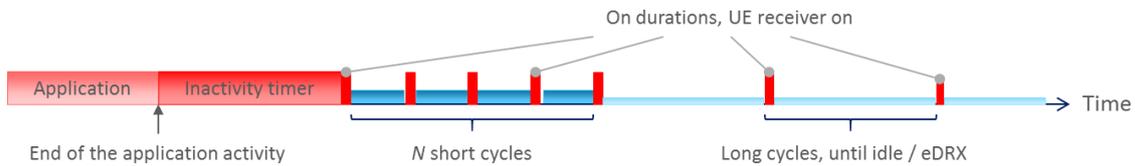


Figure 2.3-1. C-DRX states overview.

2.3.2.2 I-DRX

Radio resource control (RRC) Idle mode, or Idle DRX (I-DRX), can be triggered by the network with inactivity timer or RAI. It includes periodic paging cycles, where every paging cycle involves a wake-up cycle and an active receive period of few milliseconds (see Figure 2.3-2). Thus, during I-DRX the device is practically always reachable and there is low latency to transmit (~100ms) and to receive (on average, half the paging cycle). An example use case where RRC-Idle mode becomes useful is the communication for smartwatch application which includes: *i*) diverse data usage; *ii*) device always reachable from network (e.g. in emergency energy cut); and *iii*) low latency, critical for user experience.

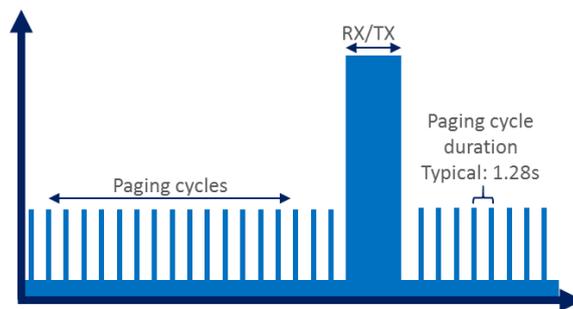


Figure 2.3-2. I-DRX states overview.

2.3.2.3 eDRX

I-DRX is used today in many LTE-based C-LoT devices but its maximum cycle period is low. It was initially 2.56 s according to first 3GPP releases, and newer releases include cases of 5.12 s and 10.24 s. This required the introduction of a new mode, extended DRX (eDRX), with higher possible cycle durations and higher flexibility (see Figure 2.3-3), to address more “static” applications, such as the smart meter, which include 1) stationary devices, possibly installed inside building/cargo or underground; 2) gas/water/electricity/etc. monitoring with periodic reporting of status; 3) readiness to send emergency data anytime (alarm); 4) device reachable from network (emergency energy



cut). eDRX allows this cycle to be expanded by several orders of magnitude, from 320 ms up to ~43 min for LTE-M, and from 10.24 s up to ~2.9 h for NB-IoT. The device in that case is still reachable, but with longer latency introduced.

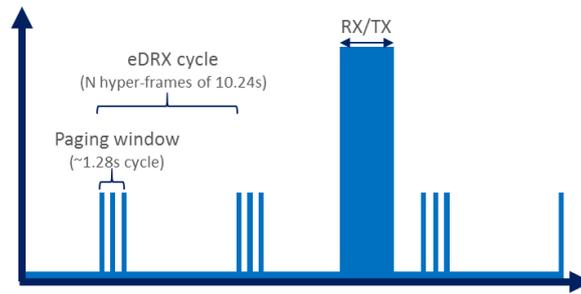


Figure 2.3-3. eDRX states overview.

2.3.2.4 PSM

Static applications, such as the asset tracker, with following characteristics: *i)* device sleeping most of the time; *ii)* periodic (slow rate) position reporting; *iii)* device does not need to be reachable but is still mobile, need to also be addressed efficiently. For that reason, the 3GPP power saving mode (PSM) was introduced in Rel.12 to enable the case of a device not being reachable while dormant and minimizes signaling with the network. Compared with the case of just disconnecting from the network, the advantage of PSM is that the device stays always registered to the network and reduces signaling and wakeup time, thus, minimizing power consumption. With PSM, the device initiates a transmission based on an application event (internal timer, sensor alarm, etc.) or a defined Tracking Area Update (TAU) timer. In that case, the device is only reachable for a time window after transmission (see Figure 2.3-4).

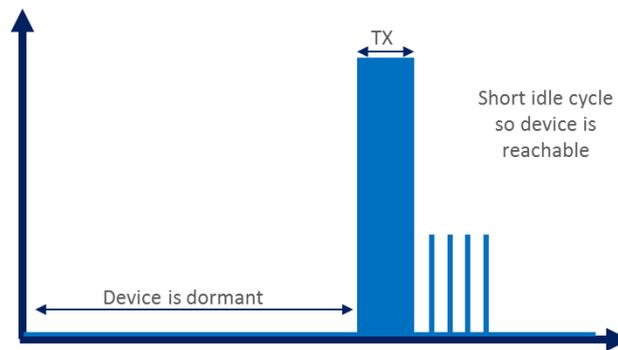


Figure 2.3-4. PSM states overview.

2.3.3 C-IoT chipset sleep modes: principles and metrics

Generally, the communication device architecture includes several parts, such as modem radio-frequency (RF) transceiver, baseband unit, internal and external memories, real-time clocks (RTC), power management unit (PMU) controlling the possibly various power supplies, subscriber identity



module (SIM) card, temperature compensated crystal oscillator (TCXO), various interfaces, etc.

Within such a complex architecture, it can be very efficient for the device that different parts can be powered off according to usage needs. For example, the modem can be actively communicating with the network (transmitting or receiving data), and in that case every part of it has to be powered (“ON”). But when a device is only processing information or monitoring control signals, there may be no need to keep the RF integrated circuit (RFIC) ON. If it can be possible to go to sleep (i.e. interrupt communications with the network) for a short time, as is the case in C-DRX or RRC Idle mode, the baseband integrated circuit (BBIC) can also stay unpowered (“OFF”).

In the extreme case of long sleeps, as for example, eDRX or PSM modes can enable, almost everything can go OFF apart from e.g., RTC or PMU. Figure 2.3-5 depicts this basic principle.

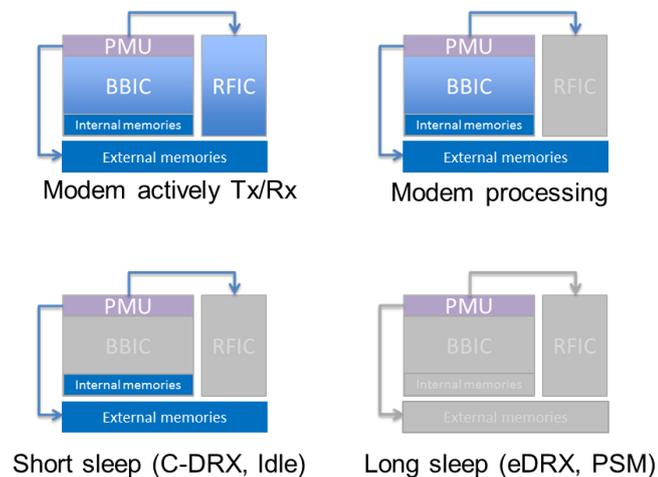


Figure 2.3-5. Different chipset parts powered off according to usage needs.

These different states, where parts of the device are OFF, can be considered as different low power (or sleeping) modes of the device. Generally, the energy consumption of a low-power mode depends on *i*) its leakage L (i.e., the power slope from initial to final state), *ii*) its fixed cost overhead W (this mostly comes from the wake-up and go-to-sleep overheads), and *iii*) the total energy E spent sleeping for a duration d , which is $E = W + L \cdot d$ (see Figure 2.3-6).

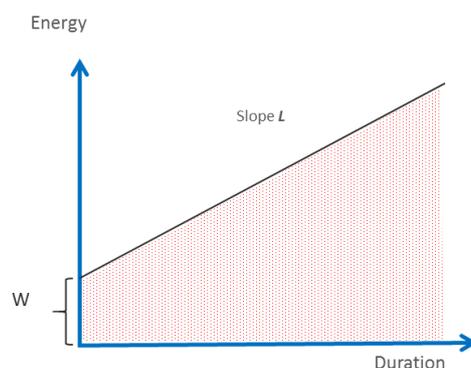


Figure 2.3-6. Leakage L and fixed cost overhead W of a device low power mode.

As can be realised, leakage and fixed overhead vary in opposite directions. This essentially means that the longer the system is shut down, the lower the leakage L is, and the more work is needed on wake-up (hence, a higher W).

From above, we can understand that the average power over a full sleep duration is equal to $P = E/d = W/d + L$. Consequently, we can observe that: *i*) the longer the sleep duration, the lower the fixed cost W impact (as the fixed cost W is amortized over time); and *ii*) a longer sleep duration enables more efficient low-power modes. As the duration increases, the modem can use lower leakage modes. This increases W , but it is amortized over time, and decreases L .

In addition, different sleep durations call for different trade-offs. For short durations, the fixed cost must be low. Then, the longer the sleep, the higher the fixed cost can be, as it is amortized over time, while the lower is the leakage we want. That calls for optimization in the sense that different low-power modes can cover a large usage duration span.

Therefore, we can have different sets of power modes ($L1, W1$), ($L2, W2$), etc. with $L1 > L2 > \dots$ and $W1 < W2 < \dots$. Then, the modem can use the best mode based on the expected sleep duration. Eventually, all the supported modes will define an optimal envelope (see example in Figure 2.3-7 example with 3 modes and 2 transition thresholds).

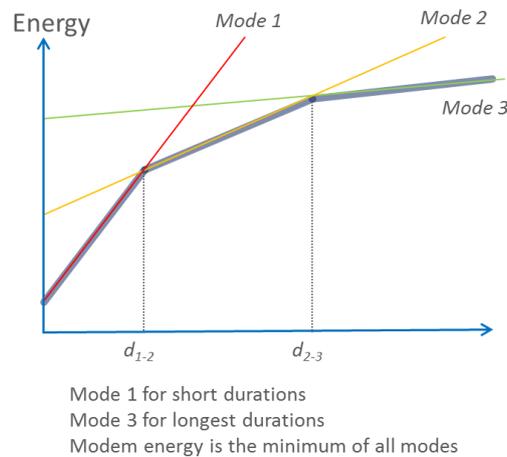


Figure 2.3-7. Modem can choose between different power modes based on expected sleep duration.

With this approach in mind, a typical C-IoT modem power consumption graph can include several low-power modes enabled within the different states appearing during a use case. Such an example is provided in Figure 2.3-8.



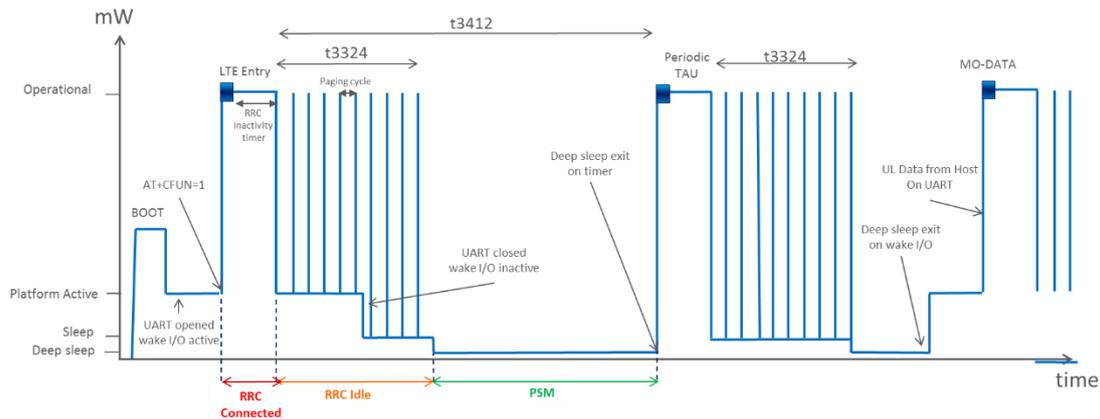


Figure 2.3-8. Example of a typical C-LoT modem power consumption graph.

2.3.4 Power consumption evaluations

A realistic NB-IoT communication module power model has been developed and validated with evaluations from tests with SEQ prototype Cat-NB devices to explore their energy usage in various states and scenarios. The target of this task is to derive insights on ways to reduce power consumption in next generations of cellular-IoT communication modules as well as to assess battery lifetime capability of compatible sensor devices.

The estimator has been consolidated according to NB-IoT technology as well as for possible network and device configurations, e.g., considering:

- Network deployment parameters, including operation mode (in-band, guard-band, standalone), scheduling of data and control blocks, system information and synchronization signals, as well as power saving mode parameters configuration, such as DRX cycle, the various C-DRX/eDRX/PSM timers involved, etc.
- Modem configuration, considering device category (e.g. NB1 or NB2), various NB-IoT uplink subcarrier modes, various standardized features that can be enabled/disabled at device such as eDRX, relax monitoring, etc.
- Device operational conditions, including various levels of coverage (from deep underground to outdoor cases) and transmit power, as well as band support.

Moreover, the following test cases and scenarios are being explored for development and analysis of modem power consumption:

- Initial cell search, from booting the device, to band scan and cell selection, to network registration.
- Various power saving modes (C-DRX, eDRX, PSM) test cases, considering the various relevant states, e.g., device wake up, cycles of paging opportunity (PO) monitoring, neighbour cell measurement, device go-to-sleep.
- Various data scenarios, considering transmission/reception of modem originated/terminated data using a transport protocol (e.g., UDP, TCP).



2.3.4.1 Insights for device power saving

After calibrating modem and network behaviour expectations according to results from tests on real Cat-NB devices in the field, several insights for communication device power saving can be obtained. With this investigation we target to generally assess C-IoT device energy behaviour and opportunities for product implementation improvement which can also be relevant in future relevant standardization topics such as IoT-NTN evolution or power saving enhancements for NR towards IoT, e.g., NR reduced capability (RedCap) devices [25].

Test case: C-DRX cycle

The typical case of power consumption during a C-DRX cycle without any optimizations considered is depicted in top part of Figure 2.3-9.

One important aspect in such scenarios, where the device can be OFF most of the time, is the leakage during deep sleep. In this C-DRX example, the device has to be one ON less than 10% of total cycle duration, rendering OFF period a significant contributor to total energy usage. Improving deep sleep power consumption efficiency, for example by implementing low leakage (yet fast accessible) memories to save connectivity related parameters and turn off modem-related parts of the device, reduces device energy usage considerably, e.g. fivefold improvement leads into ~17% reduction in total energy used per cycle in our C-DRX example scenario. Note that in I-DRX case the device has to be ON even less (~1-2% of total cycle time typically) and low power deep sleep will be even more important in that case.

Moreover, after sleep, the device wakes up to synchronize and it is common, in order to accommodate even the longest possible durations for synchronization in bad channel conditions, to implement a wake-up time margin before C-DRX ON operation. However, the device should not have to be at ON state for the whole duration of the wake-up margin in case synchronization phase finishes much faster. Employing an efficient fast sleep during wait time can give an extra energy boost, e.g. extra ~5% cycle energy reduction in our example scenario.

In bottom part of Figure 2.3-9, we illustrate the benefit on cycle total energy reduction from employing low power deep sleep and fast sleep during wait in C-DRX case.



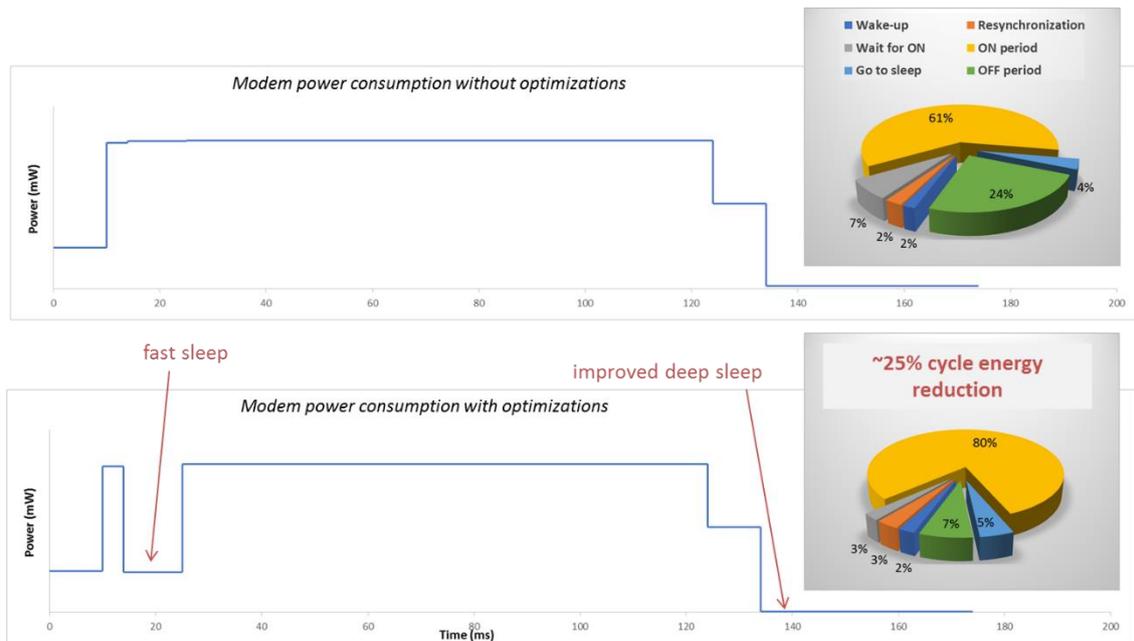


Figure 2.3-9. Power evaluations; C-DRX cycle test case (2048 ms C-DRX cycle, Good coverage, CINR>10dB).

Test case: I-DRX cycle

The typical case of power consumption during an I-DRX cycle without additional optimizations considered (from what was discussed in C-DRX case) is depicted in the upper part of Figure 2.3-10.

We first observe that having a fixed “Robust sync” implementation at device, striving to acquire all basic synchronizations signals (NPSS with period 10ms, NSSS with period 20ms) for time/frequency synchronization to the network before every paging opportunity (PO), can be inefficient. This is especially true under good channel conditions where synchronization can be faster and less power consuming by relying e.g. on narrowband reference signals (NRS) which can be found at almost every subframe and can be used to correct slight deviations on time/frequency alignment.

Moreover, in most cases of paging monitoring during I-DRX, the device has only to decode and demodulate the paging downlink control information (DCI) just to find out that no message has been scheduled by network for this device. To help this demodulation, network repeats the paging DCI over several subframes and this repetition pattern is configured based on coverage conditions. However, the device can optimise its decoding process and demodulate paging before all repetitions are received. Such early decoding early exit can save significant energy per PO cycle, reducing the most consuming phase of that scenario.

The bottom part of Figure 2.3-10 depicts the power consumption breakdown for the case where optimizations on algorithms for synchronization and paging decoding are supported. We observe that support of adaptable fast synchronization and decoding early exit can reduce the modem power consumption during such I-DRX cycles by ~35% in our example scenario.



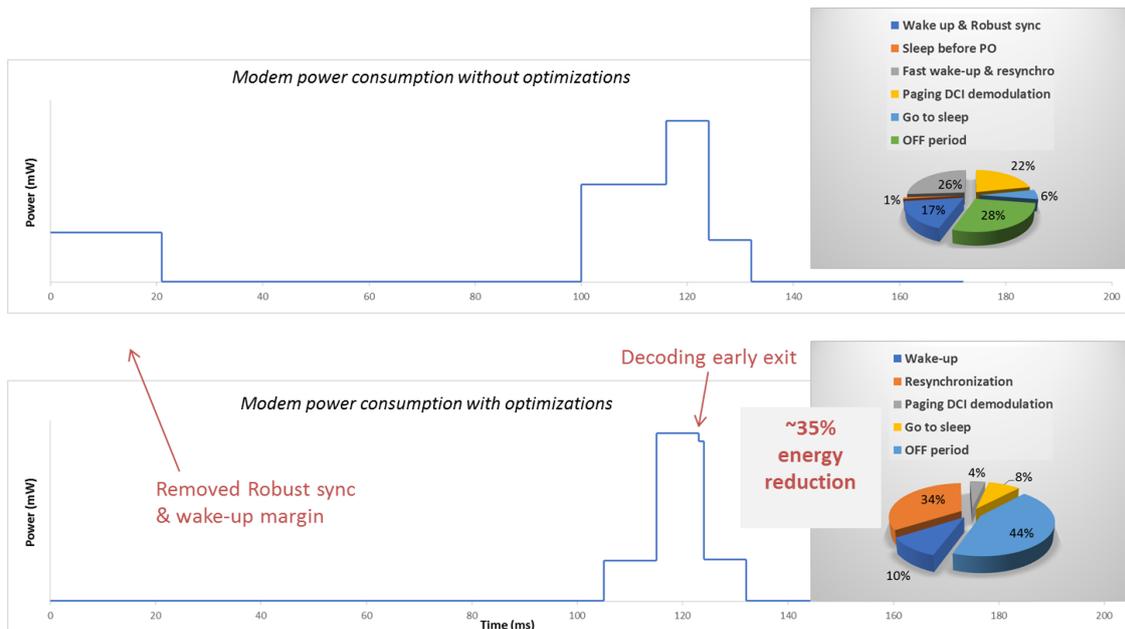


Figure 2.3-10. Power evaluations; I-DRX cycle test case (2.56 sec DRX cycle, Good coverage, 10dB>CINR>6dB).

Test case: eDRX PTW

The case of average power consumption during eDRX paging time window (PTW) without any additional optimizations considered is depicted in the upper part of Figure 2.3-11.

We first observe here that mandating intra- or inter- frequency measurements of neighbour cells at every PTW can be inefficient. This step of measurements alone may consume ~60% of the total modem consumption during PTW. However, in certain scenarios (e.g., no mobility device) there is no need to monitor neighbour cell conditions all the time, as changes (if any) will occur in larger time scales. In that case, supporting the “Relaxed monitoring” feature specified by 3GPP for IoT devices to limit the need of such measurements can be very beneficial.

In addition, having a fixed “Robust sync” implementation at device can also be inefficient, consuming ~35% of the total modem consumption during PTW. However, considering static scenario conditions, implementation in device for synchronization can rely on a faster mode, e.g., based only on NRS acquisition.

The bottom part of Figure 2.3-11 depicts the power consumption breakdown for the case where optimizations are supported. It is expected that such optimizations can reduce the modem power consumption for this scenario by ~80%.



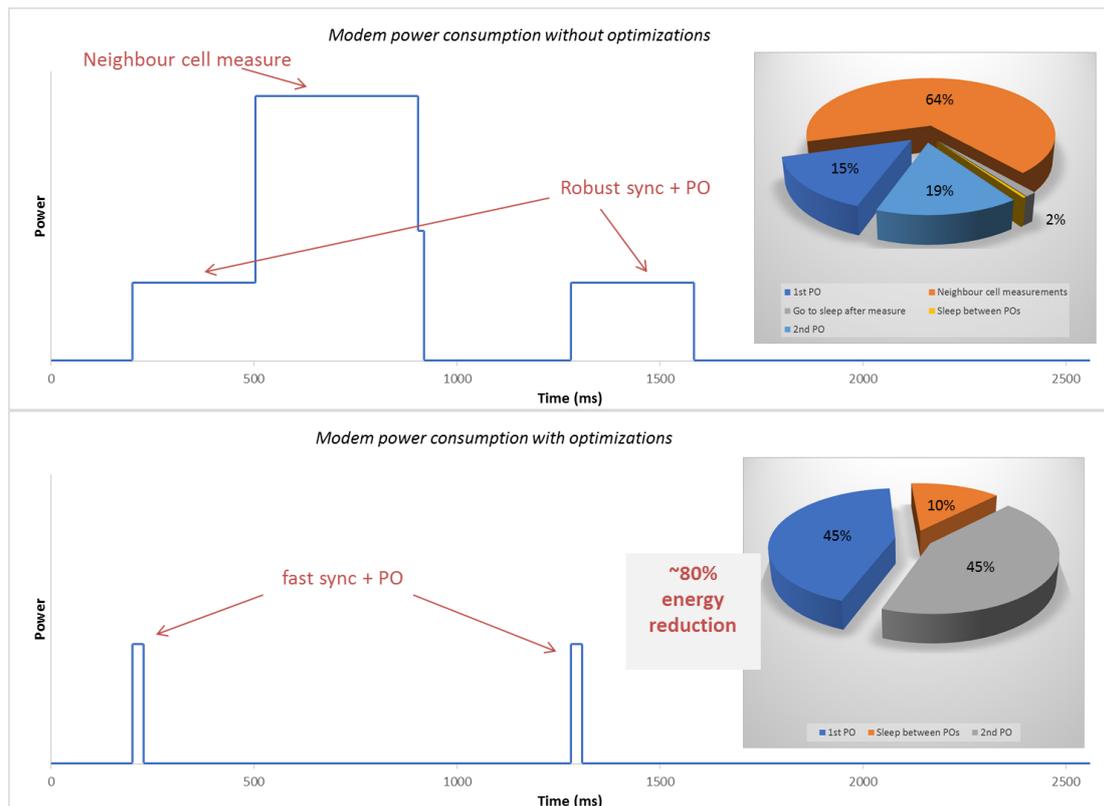


Figure 2.3-11. Power evaluations; eDRX PTW test case (2.56 sec PTW, 1.28 sec DRX cycle, Good coverage, CINR>10dB).

Test case: PSM TAU

Finally, we analyse the PSM scenario, specifically the case where the period TAU is triggered. The typical case of power consumption for this case without any additional optimizations considered is depicted in Figure 2.3-12.

One road for improvement here regards the “need” to always receive network broadcasted system information. In NB-IoT, there is a notion of modification period for basic system information which spans several tens of seconds and wherein information is not supposed to change but system information is periodically repeated with a much smaller transmission period. However, even in next modification periods, changes on such system information content occur rarely in practice. The device can take advantage of that fact and omit too often acquisition of such signals.

Furthermore, significant improvements in such scenario including data sessions can be attained if the device supports useful 3GPP standardized features such as successful acknowledgement of RRC Connection Release where device is allowed to quickly release RRC connection by considering successfully acknowledged the receipt of the RRC Connection Release message, as soon as the respective hybrid automatic repeat request acknowledgment (HARQ-ACK) has been sent to network, typically in the range of ~0.5-1 second, instead of waiting up to the default 10 seconds specified period.



In Figure 2.3-12, we illustrate the benefit of employing the aforementioned optimizations in PSM TAU test case. We observe modem power consumption can be reduced by ~70% in the example scenario.

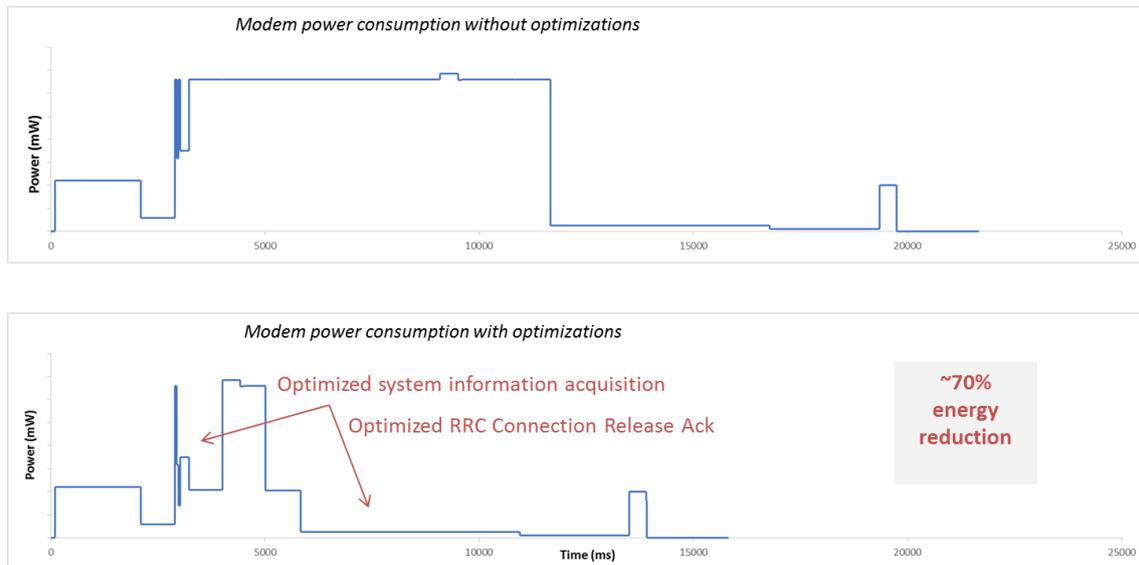


Figure 2.3-12. Power evaluations; PSM TAU test case (180 sec TAU timer, 2.56 sec DRX cycle, Medium coverage, $-3\text{dB} < \text{CINR} < 6\text{dB}$).

2.3.4.2 Battery lifetime assessment

The data application scenarios in power consumption model estimator were adapted to the traffic profiles and typical characteristics of IoT devices in iNGENIOUS Transport and Ship use cases which have been considering the use of sensors that can be supported with NB-IoT communication modules. The estimator was designed to account for changes in communication under various coverage conditions and device behaviour after modem originated data activity (i.e. simple “ON/OFF” operation, or using sleep modes with PSM and eDRX features) to understand the consumption in differently configured use cases. Furthermore, two types of cellular modem devices have been considered: 1) a “Basic” device type, representing a traditional non-power optimized NB-IoT solution seen in the market, and 2) an “Advanced” device type, representing an enhanced solution employing low power consumption implementations for wake-up and modem activity as well as the power consumption optimisation features and techniques identified in Section 2.3.4.1, such as improved sleep and resynchronization implementations, optimized system information acquisition, and support of standardized energy reduction features such as eDRX, PSM, and RRC Connection Release Acknowledgment. Figure 2.3-13 and Figure 2.3-14 illustrate the power consumption profile and the summary of the energy spent into different modem states (wake-up, attach, transmit, inactive, idle, and sleep) during a typical IoT PSM-based data session (100 bytes Tx/Rx), to highlight the difference of the two considered device types. Typical NB-IoT network configuration parameters were considered for eDRX (20.48 sec eDRX cycle, 2.56 sec PTW) and PSM (2.56 DRX cycle, 10 sec T3324 active timer) modes.



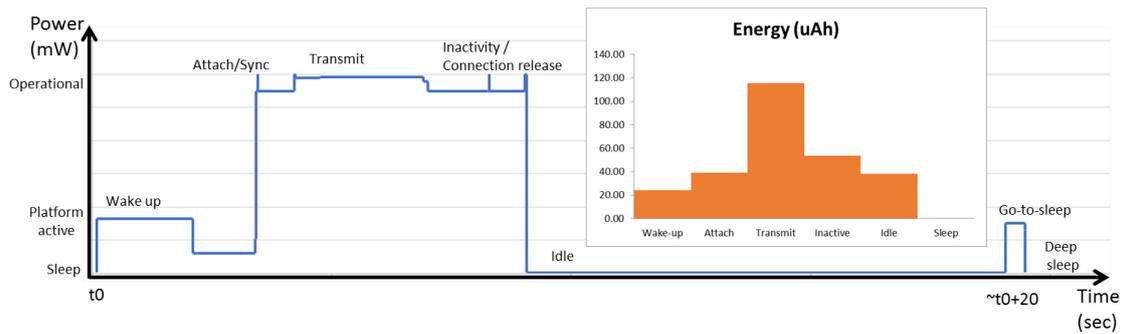


Figure 2.3-13. Power consumption and energy usage profile during data session with PSM – “Basic” device.

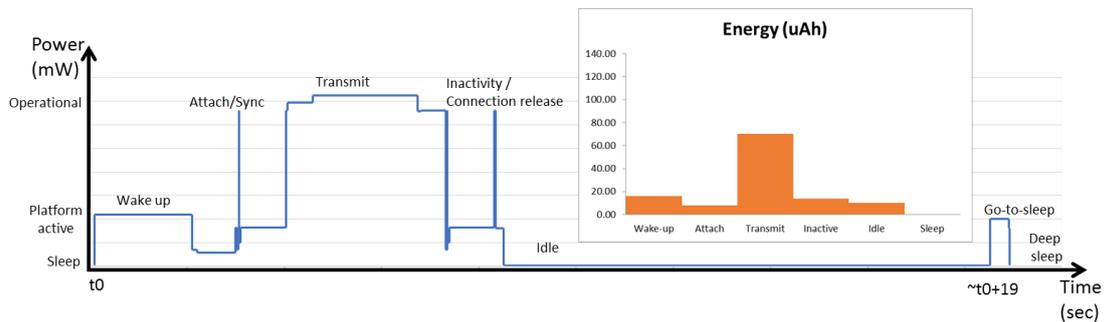


Figure 2.3-14. Power consumption and energy usage profile during data session with PSM – “Advanced” device.

We observe that the “Advanced” device can have significantly reduced energy consumption in several other traditionally consuming phases from employing identified optimisation features and techniques. In “Attach” phase, from SIM resume to acquisition and decoding of system information, a ~75% reduction can be achieved mainly due to efficient synchronization techniques. In “Inactivity” state, covering the receipt of RRC connection release message from the network and its lower layers acknowledgment from the device, a ~85% reduction can be achieved from employing fast release after last transmitted data. In ‘Idle’ state (from RRC idle preparation to the point where device saves key information in flash memory and prepares to go to deep sleep), a ~65% reduction can be achieved mainly due to optimized techniques for system information acquisition.

With the aforementioned estimator adaptations and considerations we target to provide device battery lifetime expectations for the considered sensor devices and evaluate the respective targets set in these use cases. To this end, two distinct application scenarios have been considered with the following assumed communication traffic and conditions, target lifetimes, and energy supply characteristics:

1) Rail track sensor

- a. Traffic: 30/30 Tx/Rx bytes per transaction (UDP transport protocol) in case of reporting only defects; 300 kB/30 bytes Tx/Rx per transaction (TCP protocol) in case of reporting also raw measurements. Frequency of one transaction per 30 minutes.

- b. Conditions: 7 dBm mean CINR (Outdoors coverage), ~200kbps average DL/UL throughput.
 - c. Target device lifetime: 12 years.
 - d. Energy supply: 4 AA-type batteries of 3500 mAh capacity each.
- 2) Ship track sensor
- a. Traffic: 770/30 Tx/Rx bytes per transaction (UDP transport protocol). Typical frequencies (configurable) are 1 transaction per 5 minutes / 1 hour / 1 day.
 - b. Conditions: -10 dBm mean CINR (Underground coverage), <1 kbps average DL/UL throughput.
 - c. Target device lifetime: 5 years duration at 1 transaction per day; 2.5 months duration at 1 transaction per hour; 1 week duration at 1 transaction every 5 minutes.
 - d. Energy supply: 3 AA-type batteries (non-lithium option), e.g. 3 x Asmann 5035082 AA Rechargeable Batteries 2850 mAh NiMH.

Table 8 summarizes the battery lifetime key results from the power model evaluations for the respective IoT device solutions in Ship and Transport use cases. The typical NB-IoT network configuration parameters were considered for sleep modes, as mentioned before.

Table 8. NB-IoT based device battery lifetime evaluation results for Transport and Ship use cases.

UC	Device	Operation Mode	Evaluation	Target	
Transport	Basic	ON/OFF, defect data	2 years	> 12 years	
		ON/OFF, defect & raw data	0.8 years		
		Sleep, defect data	5 years		
		Sleep, defect & raw data	1 year		
	Advanced	ON/OFF, defect data	8 years		
		ON/OFF, defect & raw data	1.2 years		
Ship	Basic	ON/OFF	3 years	> 5 years	
		Sleep	3.5 years		
	Advanced	ON/OFF	5.2 years		
		Sleep	6.5 years		
	Notes:				
	- Assuming that communication device dominates (~80%) the energy usage of the device solution.				
- Assuming 3% per year battery self-discharge rate.					

For Ship UC, we observe that the proposed power reductions allow such optimized device to generally support the respective battery lifetime targets in both operation modes, i.e. either if device is using sleep features to be in retention when not needed to transmit/receive, or if it completely powers off (which includes significant effort/energy to perform cold boot and more robust synchronization steps). It has been also observed that in case of better coverage conditions (e.g., >-3dB CINR), the lifetime targets can be easily satisfied in most scenarios. For Transport UC, we observe that the 12-year lifetime target can be only achieved with a power-optimised device using



sleep features, under generally good coverage conditions, and if the data for periodic transmission are of low size, i.e. only defect data of few tens of bytes. In case raw data of several hundreds of kilobytes size are required to be transmitted, the NB-IoT device solution is not currently viable for this use case, unless energy harvesting techniques are employed to support the re-charging of energy supply.

2.3.5 Exploitation potential

Cellular-IoT modem power consumption analysis relates mainly to the Transport and Ship use cases, but it is relevant also to Port Entrance use case where LTE-based IoT device solutions can be well applicable.

Exploitation will focus on product development for more power efficient future solutions. Final information about exploitation will be available in D7.3 “*Final dissemination, standardisation and exploitation*” [29].

Component: Cellular-IoT modem power consumption analysis

Relevant for use cases: Transport, Ship

Exploitation Potential:

- **SEQ:** Insights from analysis will be used by SEQ for internal communication and training as well as cellular IoT modem/chipset product development. With the power model estimator, SEQ can further analyse need, potential and feasibility of selected efficient solutions for low-power modem consumption (high-level, then lab, then field-tests) and plan transition to support those solutions, i.e. for internal development and implementation integrated in product, either on the SW roadmap or to be integrated in a next generation product solution.



3 Solutions for Efficient Cellular-IoT Air Interface

Air interface technology plays an important role on the overall performance of a wireless communication system. Its efficient design, considering also the limitations and capabilities of the respective communication devices, is crucial to realize the demanding requirements of IoT connectivity to the network entities. In iNGENIOUS, within its task T3.1, IoT connectivity towards contribution to the Next-Generation IoT (NG-IoT) connectivity design has been explored. Firstly, the state-of-the-art of various connectivity technologies and their respective communication devices were studied in order to identify what can be leveraged for the selected use cases, and what are the limitations of the existing solutions [10]. The ongoing 3GPP work on evolution of cellular IoT (C-IoT) technologies has been of great focus for the technical research within this task, especially of non-terrestrial networks (NTN) which play key roles in the Transport and Ship use cases. To this end, opportunities for innovation have been explored resulting in suggestions towards NTN features specification from the IoT device point-of-view (Section 3.1), as well as proposed solutions for efficient radio link layer design for satellite-based connectivity (Section 3.2).

3.1 Impact Analysis of NTN Solutions on Device

This section provides an analysis of various proposed NTN UE features and aspects for air interface enhancement arising during the respective studies from the 3GPP. We explore the potential impact of proposed NTN solutions on the hardware and software architecture design of communication devices and we provide suggestions for efficient specification evolution.

3.1.1 Work overview

Since 2018, 3GPP has been working on 5G system specification (Release 15), and its further evolution (Release 16, 17, etc.). One direction in this work has been to enable the 5G system to support NTN for NR and for the legacy IoT technologies, long-term evolution machine type communication (LTE-M) and narrow-band IoT (NB-IoT). The main objective of this 3GPP work is to address remote areas with no cellular connectivity and complement terrestrial network (TN) deployments. The current 3GPP roadmap on those so-called NR-NTN and IoT-NTN tracks is illustrated in Figure 3.1-1. More information on NTN specification activity has been provided in deliverable D3.1 [10].



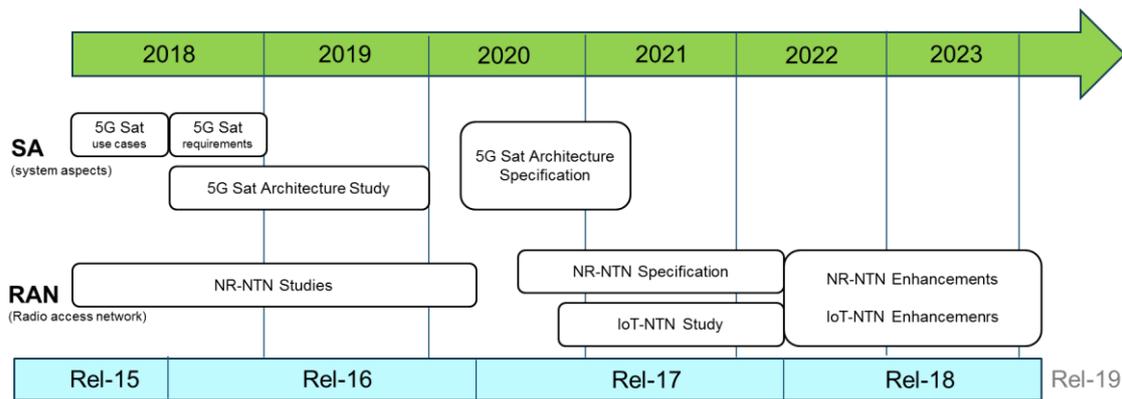


Figure 3.1-1. 3GPP NTN roadmap.

The main challenges identified for NTN work on radio communication specification arise from the following: 1) High satellite orbital height causes a high path loss and a large round-trip time (RTT); 2) Mobility of low earth orbit (LEO) satellites introduces a very high Doppler offset on the radio link, and it also inevitably requires all devices to frequently change their serving nodes.

Initially, the NR-NTN 3GPP studies, see e.g. [26], identified the issues originating in NTN due to long propagation delays, large Doppler effects, and Earth-moving cells in case of satellite without mechanically steerable beams. The features to be considered for enhancement were then shortlisted, including uplink time and frequency synchronization, extension of timers in various layers, enhancement on uplink (UL) scheduling to reduce scheduling latency, enabling/disabling of HARQ feedback, enhancement on the physical random-access channel (PRACH), and other random-access enhancements, beam management and bandwidth parts (BWP) operation, location-aware handover, enhancement to existing measurement configurations to address absolute propagation delay difference between satellites, service continuity for mobility from TN to NTN and from NTN to TN systems.

For IoT-NTN 3GPP study [14] on the other hand, the focus has been on evaluating and corroborating solutions to address the minimum essential functionalities to support LTE-M and NB-IoT over satellite, inheriting as much as possible the insights from the preceding NR-NTN work. Eventually, 3GPP objectives were shaped to focus only on the necessary enhancements of several timing relationships as well as for time and frequency synchronization. The ultimate target of this Rel-17 work was to establish basic mechanisms to manage these challenges and to provide a first set of specifications to support NTNs based on NR, NB-IoT and LTE-M.

SEQ has been monitoring and contributing to the NTN activities of 3GPP focusing initially on analyzing the various aspects and proposals for enhancement considering the potential effect from chipset device point of view, in order to provide suggestions for efficient specification evolution.

3.1.2 Potential enhancement features and impact on device architecture

In the following, we present the various key solutions discussed in 3GPP for NTN (captured within NTN study reports [12][14]), which have been identified



with potential to affect the chipset hardware design. We discuss the motivation for these proposed features, their respective arising specification design issues, and remarks from chipset design perspective. Then, we summarize our findings.

3.1.2.1 Analysis of NTN features

Configuration of timer offsets

Motivation: One important enhancement suggestion has been to introduce time offsets to enhance timing relationships, e.g., between uplink/downlink channels and signals.

Key issues: Questions that arise include 1) need of cell and/or beam-specific timers and how to configure the offsets, e.g., before or after device initial access.

Remarks: Generally, both cell-specific & beam-specific offset values can be supported by devices, but beam-specific configuration can reduce RACH process delay during initial access for UE with small RTT. It is expected to be more reliable to have such updates driven by the network, not by the UE, to ensure a more uniform behavior of the deployed UEs and the disabling of update signaling when not desirable. Moreover, the use of offsets in some cases may be problematic and requires further investigation. For example, it has been proposed to introduce an offset for dynamic slot format indicator (SFI) in TDD and half duplex FDD (HD-FDD) to combat different RTTs over the access link experienced by different users in a group. However, this may need complex chipset design to avoid cross-link interference.

Initial timing advance (TA) acquisition and frequency adjustment

Motivation: When a device is coming back from idle/inactive mode, it should be able to autonomously acquire its TA with some good accuracy level for performing timing and Doppler-shift pre-compensation.

Key issues: Two methods have been investigated: 1) geometric-based (GNSS-acquired position and network-indicated satellite ephemeris), and 2) timestamp-based (GNSS-acquired time and NW-indicated reference time).

Remarks: Regarding the power consumption impact, geometric-based methods seem to not need to use GNSS capability too often. Considering the case of DRX, GNSS timing will be needed before each random access attempt from the device, while for GNSS location that requirement may be relaxed. Regarding complexity, geometric-based option is preferable in case of device solutions with integrated modem and GNSS. However, such solutions are often not optimized in terms of positioning accuracy performance of GNSS.

TA update and frequency adjustment in connected mode

Motivation: After initial synchronization acquisition, the device stays in connected mode, and it should be able to maintain accurate time and frequency synchronization.



Key issues: An important question here is if it is possible to have only an open-loop solution, i.e., if device can update time and frequency autonomously.

Remarks: It is possibly not very reasonable to require autonomous time and frequency adjustment at device side in case of integrated modem/GNSS solution. If the GNSS radio shares the 5G radio, they probably cannot be used simultaneously due to significant interference levels from/to each other. In that case, introduction of GNSS-specific measurement gaps during radio connection may have to be introduced.

GNSS accuracy/availability requirement

Motivation: Since GNSS-based device is one of the main assumptions for Rel.17, it is essential to characterize its positioning performance and potential enhancements. Metrics considered are accuracy, which is defined as the statistical characterization of the error in time and space, and availability, which is defined as the percentage of time during which the outputs of interest regarding positioning are available to the UE.

Key issues: A question here is what the assumptions on GNSS performance requirement from the device point of view should be.

Remarks: Regarding accuracy, GNSS systems in outdoors scenarios typically provide position accuracy within a few meters (approximately 5 to 10 meters). Thus, the impact on position error may be considered negligible. However, the time-to-fix of positioning, related to the error in the time aspect of accuracy, is also an important performance parameter, as increased time-to-fix can create an even larger synchronization error. Regarding availability, in RRC connected mode, a GNSS integrated solution, i.e., RF circuit is shared between the GNSS and NR chipset, may have constraints to perform position acquisition from GNSS. Thus, introduction of GNSS-specific measurements gaps may be needed.

Enhanced HARQ process ID indication

Motivation: Increase of parallel running HARQ processes (i.e., maximum number of packets that can be transmitted concurrently and wait for receipt acknowledgement from receiver) should be supported in order to reduce long wait times due to large delays between data transmission and the respective HARQ acknowledgment.

Key issues: There is a question of whether to increase parallel running HARQ processes, and whether to make this increase an optional or mandatory UE capability.

Remarks: Generally, mandatory HARQ processes increase can be an issue for the device due to buffers scaling. This would be possibly an issue for legacy C-IoT devices to support such enhancement in IoT-NTN. For a NR Rel-15/16 device to support NR-NTN, considering restricted MIMO layers and modulation order in NTN, this should not be much of an issue, as such devices are already designed with enough memory for all component carriers.



Transmit polarization

Motivation: DL/UL transmit antenna polarization (e.g., linear, circular, adaptive, etc.) has been proposed as a solution for spatial frequency reuse for inter-beam co-channel interference and for reduced capacity impact.

Key issues: The main questions of this proposal include how to achieve common UE and network understanding on the polarization mode that is being used by device and if/how to report respective UE capabilities to the network.

Remarks: Generally, the possibility for a legacy device to support polarized antennas depends on its antenna design and implementation. A flexible UE that adapts its polarization mode may not be feasible in some cases, e.g., for low frequency bands, where the UE/antenna form factor may limit the employment of some modes. On the other hand, a positive effect of such solution (for a device that can support it) is that UE can only turn on the corresponding port according to the polarization state that a target cell employs, and keep the other turned off, consequently reducing its power consumption.

RACH enhancements

Motivation: In case pre-compensation cannot be assumed (if no GNSS-capable UE, or if GNSS performance is low), enhanced PRACH formats and/or preamble sequences will be needed.

Key issues: 3GPP tried to identify scenarios where GNSS capability cannot be pre-compensate and discussed possible enhancement options.

Remarks: Some enhancement options included the use of Gold or M-sequences. In that case, a new implementation will be required at chipset compared to legacy device design. Moreover, other enhancement options included multiple Zadoff-Chu sequences. Such solution will require a modified implementation, which, however, can be relatively simple if the existing root defined in current specification is to be considered.

3.1.2.2 Conclusions

Generally, we observe that several enhancement features discussed for NTN in 3GPP can potentially have important impact on chipset hardware and software implementation needs. In some cases caution will be needed on actual specification of the features to reduce or even completely remove such impact. Figure below provides a rough mapping assessment between the various features of interest and the chipset impact.



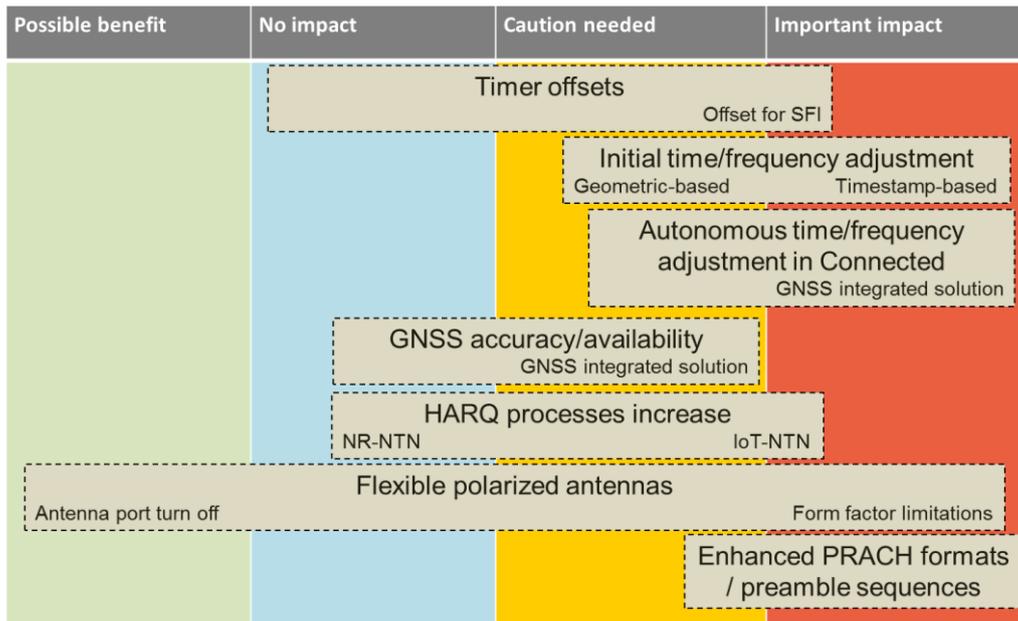


Figure 3.1-2. Mapping assessment summary of proposed NTN features with potential to affect the communication device design.

3.1.3 Overall NTN impact on device architecture design

Generally, a UE consists of several blocks and functions. A simplified architecture diagram is provided in Figure 3.1-3.

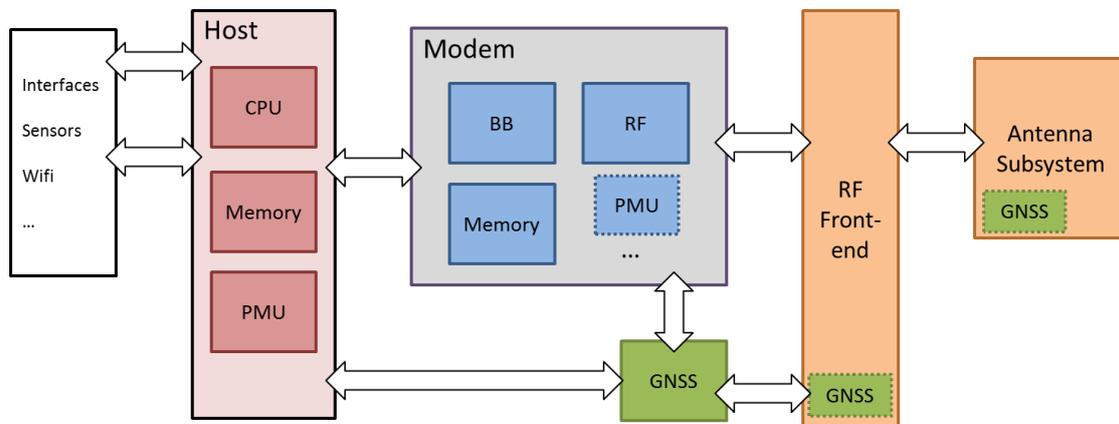


Figure 3.1-3. Exemplary diagram of a device architecture.

The user-related functions, such as connecting an IoT device through Ethernet or Wi-Fi for example, are generally driven by the host CPU, a micro-processor which has the complete control of almost everything in the device. This CPU is usually quite powerful to handle many interfaces, data management, etc. In addition, the device includes one or several modems, which provide connectivity to the network, as well as several other components, such as memories and power management unit (PMU). For cellular connectivity, such as 4G/5G, the modem is usually a separated component, and made of two main parts: a baseband (BB) and a radio frequency (RF) component. The modem is then connected to an RF front-end, which ensures the adaptation of the RF signal to the required frequency band, thus, including filtering and amplification functions, and which is



connected to the antenna subsystem. If there are positioning functions included (noted as GNSS in Figure 3.1-3), they can be separated or integrated into the modem.

Starting from this basic architecture, the important question is what has to be changed to support NTN operation:

Host CPU: most likely minimal changes required. As host CPU is in charge of controlling and driving the complete terminal, some SW changes should be expected to support some specific satellite related operations.

Modem: the required changes from hardware perspective will mainly depend on the differences between the terrestrial waveform and protocols to the satellite waveform and protocols. Since the current trend in the NTN work items of 3GPP is to minimize these differences, it is fair to expect that legacy chipset may be able to support the adaptation of the waveform and protocols from terrestrial to satellite. From software perspective, which includes the protocol stack and low level firmware, several modifications will be required. Definitely, specific protocols and procedures will have to be developed too (e.g. in MAC, RRC, etc.) to implement the 3GPP specifications for NTN. Moreover, modifications in the signal processing algorithms should be expected due to the particular nature of the satellite communication. For instance, due to the different radio channel, adaptation of channel estimation algorithms will be required. Finally, a usually overlooked aspect is the test and integration of such adaptations.

RF front-end: mandatory modifications will be needed to support the specific bands of the satellite operation. For instance, dedicated RF filters (including band-specific-filters to pass only the frequency band of interest, duplexers to split the signal into DL and UL) must be fitted alongside terrestrial bands filters to support the L-band and S-band. The design after such filtering (including signal amplifiers, mixers to down-convert signal to baseband frequencies, analog-to-digital and digital-to-analog converters (ADC/DAC), digital signal filters for compensating crude analog filtering) has to also be carefully re-designed as a whole for the device. For example, when receiving from a satellite, the frequency may experience a big Doppler shift and the device mixer may need to be calibrated according to an estimated frequency offset and the frequency of the LO (local oscillator) may need to be locked to e.g., GPS frequency, with a PLL (phase locked loop). If that is the case, the front end may need to be integrated close with the GNSS module. Finally, very low noise amplifiers (LNA) and higher power amplifiers (PA) will be needed for better receiver sensitivity and increased transmit power, respectively.

Antenna subsystem: most likely will need modifications to provide higher gain and possibly tracking functionality. Note that RF front-end and antenna subsystem are controlled by SW; hence, any modification of these subsystems to support NTN will generate changes in the SW that drives them.

As a conclusion, the adaptation of a complete terminal from terrestrial to satellite operation requires change in many subsystems of the terminal, and therefore needs complex interaction between the different providers of these subsystems. The IoT device integrator would have to coordinate the



work between the many players supplying the equipment integrated in the device (modem, front-end, antenna, power, screen, etc.) and ensure the end-to-end quality of the outcome.

3.1.4 Exploitation potential

Analysis of NTN enhancement features impact on device relates to the Transport and Ship use cases.

Exploitation will focus on utilizing the improved know-how in satellite-based connectivity for future IoT device product design and contribution to standards. Final information about exploitation will be available in D7.3 “*Final dissemination, standardisation and exploitation*” [29].

Component: Analysis of NTN enhancement features impact on device
Relevant for use cases: Transport, Ship
Exploitation Potential: <ul style="list-style-type: none"> • SEQ: This analysis improves SEQ know-how in 5G IoT space and vertical applications, modem device-oriented enhanced features, and expected impact of NTN communication on chipset architecture design. This knowledge can be used for defining future NTN-supporting product design and requirements. With this know-how basis set, SEQ can also continue contributing to relevant 3GPP studies and/or specification phases for NTN-based communication and advise accordingly on preferred features from device point of view, increasing at the same time our visibility and credibility amongst the community.

3.2 Enhancements for NTN

This section provides iNGENIOUS proposed enhancements for the radio link layer of non-terrestrial networks (NTN).

3.2.1 Work overview

During the lifetime of iNGENIOUS, 3GPP started studies and specification work related to NTN. Several issues have been brought forward from RAN groups, mainly to address procedures in legacy link layer operations that could be problematic or inefficient if employed as currently specified for NTN communication. SEQ has been monitoring these proceedings since they are very relevant to future advancement of Transport and Ship use cases of iNGENIOUS. Issues have been identified within radio link control and handover procedures in NTN, and respective solutions have been proposed for efficient procedures’ operation. The proposed specification designs can improve IoT connectivity via satellite in aspects such as latency and ease of implementation from device point-of-view.

3.2.2 Radio link control layer enhancements

During 3GPP specific work of NR-NTN within Release 17 [27], the RAN2 group agreed to extend the value range of t-Reassembly timer, to cope with longer RTT in NTN scenarios when HARQ feedback is enabled. This timer, defined in



radio link control (RLC) layer, is the period during which the network waits for HARQ retransmission from UE for the lost RLC packets. In the following, we present an investigation on possible side effects of such extension and proposed enhancements. More information can be found in SEQ proposal contribution to 3GPP [15].

3.2.2.1 Problem background

In RLC acknowledged mode (AM) of operation, the detection of a lost packet data unit (PDU) is a 2-step process: a gap detection triggers t-Reassembly timer to start, which upon expiry triggers scheduling request (SR). This process is started under two different scenarios, for which the gap detection triggers are different. These scenarios are:

1) **When a PDU of AM data (AMD) structure is placed in the reception buffer, if t-Reassembly is not running**

In this case, the RLC receiver checks if there is a gap between `RX_Next` and `RX_Next_Highest`². This means that the same gap will trigger t-Reassembly again and again, as long as the RLC receiver has not received the corresponding PDUs but continues to receive new ones. The RLC retransmissions are prioritized by the transmitter and should be received just after around 1 RTT from sending the Status Report.

Before approximately 1 RTT, these additional triggers are redundant (same gap will again and again trigger t-Reassembly and an SR, while not enough time is left for the RLC transmitter to send the retransmissions). After around 1 RTT, the additional triggers are no longer redundant, and even useful, as they indicate a problem with the retransmission (lost scheduling request or retransmission), since the RLC receiver continues to receive new PDUs.

In a legacy scenario, RTT is very small (around 8 ms in LTE), and t-statusProhibit timer, used by the receiving side of AM RLC to prohibit transmission of a STATUS PDU³, would be typically larger. In addition, t-Reassembly would be at least around a few RTTs (typically $5 \cdot \text{RTT} = 40\text{ms}$), which naturally limits the triggers (t-Reassembly can be triggered again only when not already running). Hence, in a legacy scenario, these additional triggers are never redundant.

2) **When t-Reassembly expires:**

In this case, RLC receiver checks if there is a gap between `RX_Highest_Status` and `RX_Next_Highest`. With `RX_Highest_Status` being at least equal to `RX_Next_Status_Trigger`, only “new gaps” would again trigger a t-Reassembly. There is no redundant trigger in that case.

In NTN scenario, with large RTT (up to 540 ms) and small t-Reassembly (down to 0 with HARQ disabled and no blind retransmissions), the network has only the following bad alternatives:

² `RX_Next`, `RX_Next_Highest`, `RX_Highest_Status`, and `RX_Next_Status_Trigger` are state variables existing at the receiving side of an AM RLC entity).

³ STATUS PDU is a type of PDU sent to the peer RLC layer in order to acknowledge received RLC service data units (SDU) and RLC SDU segments.



- Increase `t-statusProhibit` timer to cover the large RTT, setting it to at least 800 ms. This has the drawback to slow down the reporting of lost PDUs, i.e., it increases the overall delay of the link and, hence, reduces the throughput.
- Lower `t-statusProhibit` to values below the RTT. This has the drawback of producing redundant triggers (reporting of the same lost PDUs while RLC transmitter could not yet send them), i.e., additional redundant UL signaling (which is especially a problem in NTN where UL scheduling enhancements are needed to combat large RTTs).

In summary, the legacy lost PDU detection trigger works well for short RTT/large `t-Reassembly` (as it is the case in terrestrial networks). For large RTT and short `t-Reassembly` (which is the case in NTN with HARQ disabled), it is not well suited. This could be expected as this mechanism dates back from LTE Release 8, and was never designed to handle efficiently the NTN scenario.

3.2.2.2 Large `t-Reassembly` drawbacks

The `t-Reassembly` timer in RLC is mainly needed to allow HARQ retransmissions from lower layers, before triggering ARQ, or moving windows in un-acknowledged mode (UM). Hence, its setting will depend on whether the network intends to use HARQ feedback for downlink transfer of a given RLC bearer.

During 3GPP RAN2 specification work for NTN, it was agreed that HARQ feedback can be disabled at least on a per-process basis. In downlink, the network scheduler may dynamically decide to use “HARQ feedback enabled/disabled” processes, but we assume that:

- Large `t-Reassembly` would be configured for bearers mapped to “HARQ feedback enabled” processes. It is expected to be used for traffic requiring high reliability, e.g., signaling radio bearers (SRB). This would include application/services running over non-access stratum (NAS). This may also be used for specific traffic such as control traffic.
- Short (or null) `t-Reassembly` would be configured for bearers mapped to “HARQ feedback disabled” processes, requiring high throughput.

The following formula from [12] indicates the value required depending on RTT, maximum retransmissions and scheduling margin:

$$t\text{-Reassembly} = RTT * nr_of_HARQ_retrans + scheduling_offset$$

Given RTT of up to 541 ms in NTN, the maximum value could be quite high, compared to the maximum of 200 ms in NR.

It is expected that a common case of NTN traffic will be transmission of small data bursts (continuous transmission is not possible due to the shortage of HARQ processes relative to the large RTT). Note that, typically, the network would also poll on the last PDU of such data burst. Configuring a high timer value comes with drawbacks (as explained in section 3.2.2.1). Indeed, there is only one such timer instance, to simplify specification and implementation. In case a gap Y (of SDU(s)/segments) occurs while the `t-Reassembly` timer is



already running due to a previous gap X, the timer will be started again (to wait for recovery of Y) when the previous timer instance is either:

- stopped (X is recovered, thanks to HARQ),
- or expired (X could not be fully recovered by HARQ, i.e., lost).

This has two undesirable consequences:

- An additional STATUS PDU is triggered (to report missing Y) in case X was lost (which impacts UL scheduling).
- The STATUS PDU reporting Y is unnecessarily delayed (up to t-Reassembly value, which might be further delayed by a t-statusProhibit timer), whether X was recovered or lost.

This can be a common case, depending on HARQ operation point. For example, with 10% block error rate (BLER), the probability to have two failed transport blocks in a burst of 20 slots is around 20%.

This is described in Figure 3.2-1, in which X and Y were lost (if X was recovered while Y only is lost, the issue would be a delayed STATUS PDU). The scheduling request (SR) trigger from polling is delayed till the t-Reassembly expiry related to Y.

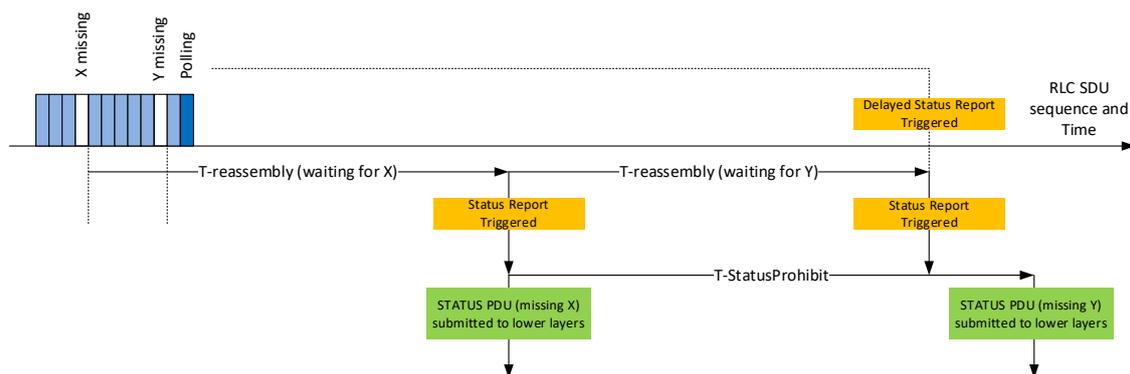


Figure 3.2-1. Additional and delayed STATUS PDU.

3.2.2.3 Proposed solutions for large t-Reassembly

In the above scenario, ideally, the t-Reassembly should be started at the end of the burst, and a single STATUS PDU should be sent for the whole downlink data burst at expiry of the timer. One approach is to have t-Reassembly triggered from the network side, e.g. along with polling, or in addition to existing triggering conditions, or to consider a different timer triggered from network side. However, this introduces complexity to the procedure design.

In our view, a simple way to better handle the above scenario would be to introduce a t-Reassembly-delay timer, which would be used to delay the start of t-Reassembly upon a missing PDU detection. The timer could also be stopped upon polling, as this typically indicates the end of a data burst, reducing useless delay. An additional advantage is that the STATUS PDU transmission could be then better predicted by the network, which can pre-schedule the uplink resources.



The t-Reassembly-delay timer would be configured to the typical length of a data burst, which in NTN scenarios might be at least one order of magnitude shorter than the value of t-Reassembly timer. This would result in the optimal behaviour illustrated in Figure 3.2-2.

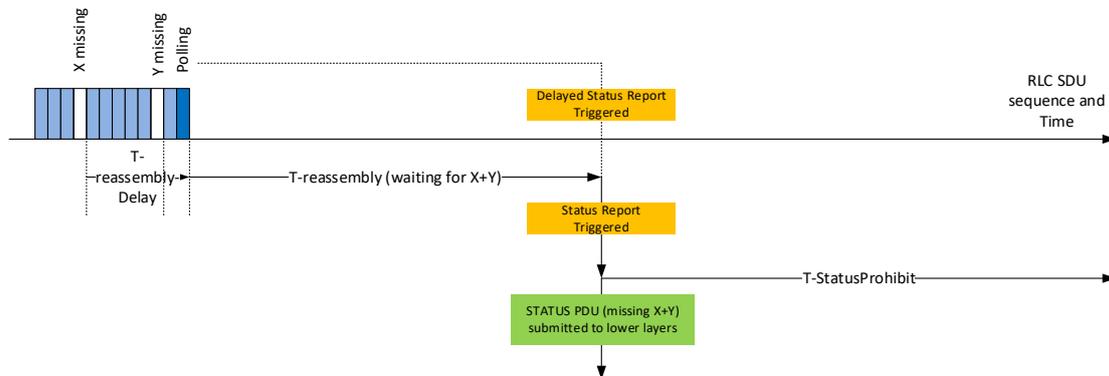


Figure 3.2-2. Fast unique STATUS PDU with t-Reassembly-delay timer.

A possible drawback of a t-Reassembly-delay timer could be that the STATUS PDU reporting missing X would be delayed by up to t-Reassembly-delay value, on top of t-Reassembly. However, considering NTN scenarios where t-Reassembly-delay timer (~data burst size) << t-Reassembly timer (~several times RTT), this drawback is insignificant.

Another approach is to introduce in NR the possibility to disable the triggering of status report upon missing PDU detection, as it was already introduced in LTE RLC for NB-IoT. However, this would only avoid the additional STATUS PDU (as the only status report trigger would be polling from network). The STATUS PDU (reporting X and Y in that case) would remain delayed by a period of up to 2*t-Reassembly.

3.2.2.4 Short/Null t-Reassembly drawbacks and proposed solution

For high-throughput bearers, network is expected to configure and use “HARQ feedback disabled” processes. It is expected that t-Reassembly is configured to a short (e.g., 10 ms) value (for instance to allow blind HARQ retransmissions), or even to 0, if the network does not plan any out-of-ordering scheduling.

Configuring a short t-Reassembly has in theory the benefit of quickly detecting/reporting missing PDUs (assuming sparse failures). However, following a missing PDU X, the RLC receiver will keep triggering SR while X is missing upon each new PDU arrival (if t-Reassembly is not running), as described in Figure 3.2-3.

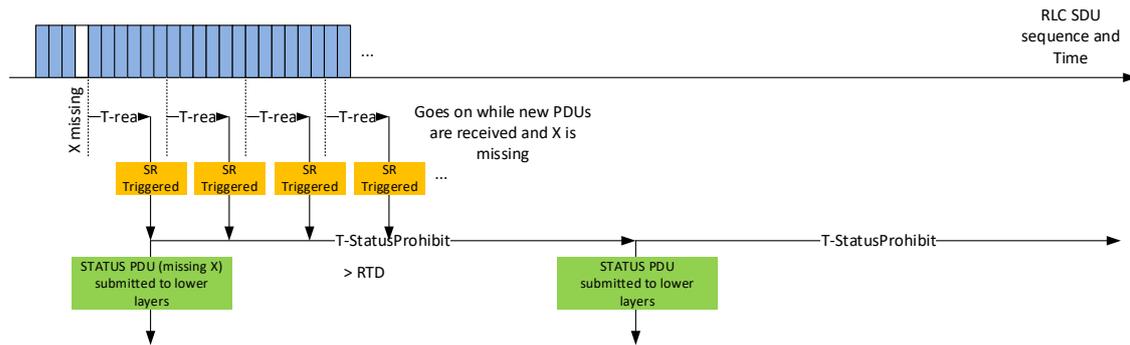


Figure 3.2-3. Fast unique STATUS PDU with t-Reassembly-delay timer.

This requires a large SR prohibit timer (at least equal to RTT) and somehow defeats the benefit of the fast detection (except for the first one, for which SR prohibit is not yet running). Instead, it might be better to rely on SR triggering by polling only, as this has the benefit that the network would always know when a SR transmission is performed (and can then preschedule resources, e.g. including scheduling blind UL retransmissions to enhance reliability of the SR).

3.2.3 Handover enhancements

RRC configuration is a very important step in establishing radio connection and understanding between UE and network. In RRC configuration, an information element called *ServingCellConfigCommon* is used by the network to configure cell specific parameters of a UE's serving cell. This element contains parameters which a UE would typically acquire from broadcasted system information when accessing the cell from idle state. In case of NTN, one such parameter that can be included in *ServingCellConfigCommon* message is *NTN-config*, including information required to access the NTN target cell, such as ephemeris (indicating position and motion of satellite) and common TA (relevant to handle timing offset for long delay) that are related to the indicated Epoch Time, i.e. the time at which this information snapshot was taken. Moreover, a new system information block is used in that case, i.e. system information block 19 (SIB19), that contains satellite assistance information.

During 3GPP specific work of NR-NTN, within Release 17, the RAN2 group has been discussing issues related to NTN configuration at handover procedure as this has been defined in earlier releases of NR [17]. In the following, we present an identified issue with handover in NTN and a proposed solution. More information can be found in SEQ proposal contribution to 3GPP [19].

3.2.3.1 Problem background

In Release 16, 3GPP introduced a new handover procedure, called conditional handover (CHO). This allows UE to decide to perform handover when certain conditions are met, instead of the legacy purely network-controlled case. In CHO, the UE starts evaluating the execution conditions upon receiving the CHO configuration, and stops once a handover is executed. Generally, in CHO case, the network has no knowledge of when the actual handover will take place. When considering NTN case, typically, the validity of NTN-config



information might be around 1 minute, while the CHO might be executed much later. Thus, it is not possible for the NW to provide a NTN-config that will be valid at the time of CHO execution in the HO message.

3.2.3.2 Proposed solution

To solve the aforementioned issue, we see the following possible options.

Time-based CHO: the network can restrain the time range in which the CHO execution can take place. The time range can currently be configured by a starting time π and a duration between 100 ms and 10 minutes, by step of 100 ms. By using a maximum time range of 10 s, the NW can configure an NTN-config with the Epoch Time falling into that range, that can be used without ambiguity by the UE. However, we see 2 main limitations with that solution: 1) time π may be restrained to some near future as the network may be limited in providing ephemeris/common TA parameters for Epochs far away in the future; 2) the maximum duration is limited to 10s as (due to Epoch Time subframe number ambiguity).

SIB19 acquisition at HO: In case no valid NTN-config is available, eventually, the only solution would be for the UE to acquire SIB19 at execution time. However, this process is currently not specified in standard, as it is assumed that all the information required by the UE is available at the time of HO execution.

Use neighbour cell assistance information from SIB19: If the target cell is part of the neighbour cells listed in SIB19, the NTN-config of the target cell will be already available to the UE and it can be kept updated by the UE reading broadcast SIB19, or by the network sending dedicated SIB19. In that case, at the point of CHO execution, the UE has an up-to-date version of NTN-config of the target cell. It would be also possible that the NW keeps sending dedicated SIB19 with the target cell NTN-config information, while not providing the information in the broadcasted version of SIB19 (for reduced broadcast overhead). This solution is exemplified in the figure below.

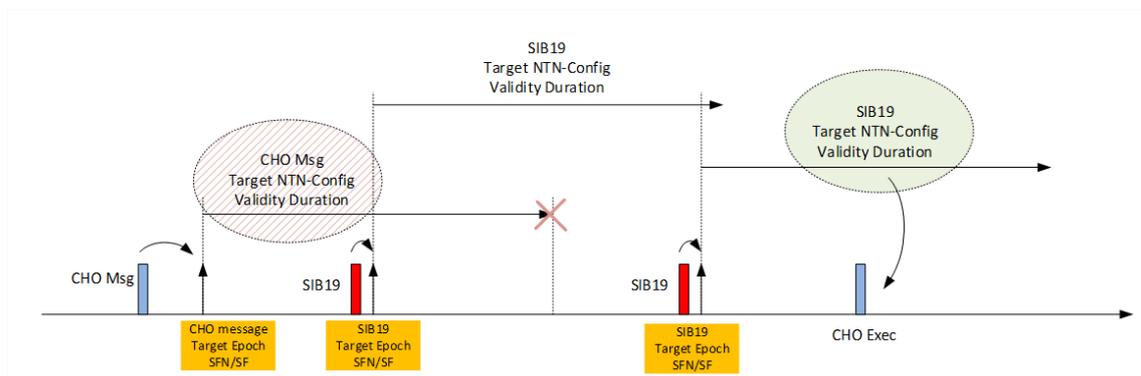


Figure 3.2-4. Using SIB19 Target cell NTN-config for CHO.

In our view, using Neighbour cell assistance information from SIB19 is the preferred solution, as it is applicable to all CHO cases and avoids handover procedure interruption. Moreover, the proposed solution could also apply to the legacy network-controlled handover case. This would simplify



specification and device implementation to keep common behaviour between CHO and legacy HO.

3.2.4 Exploitation potential

Proposed solutions for enhancement of NTN relate to the Transport and Ship use cases.

Exploitation will mainly focus on further contributions and visibility within 3GPP standards groups. Final information about exploitation will be available in D7.3 “*Final dissemination, standardisation and exploitation*” [29].

Component: Enhancements for NTN

Relevant for use cases: Transport, Ship

Exploitation Potential:

- **SEQ:** The developed solutions allow SEQ to be a useful player in 3GPP community by contributing proposals that can generally improve NTN-related technology and communications. If selected in standard specification, SEQ may plan support of the features and prepare future platforms for internal development.



4 Solutions for Versatile IoT Communication

For realizing an heterogeneous IoT network that is able to attain the requirements presented by the iNGENIOUS use cases, which covers various scenarios that may require multiple radio technologies, and adaptable radio access interfaces, a solution that exploits software defined radio platforms shall be investigated. Also, a versatile and efficiently performing 5G modem hardware device shall be employed to connect specific IoT devices to the 5G network and adapt to the needs of iNGENIOUS use cases. To this end, this chapter presents the respective implementation developments within iNGENIOUS of flexible radio architecture at device side and of 5G modem solution.

4.1 Flexible HW/SW Architecture

In this section, we present the flexible PHY/MAC implementation of TUD with emphasis in the physical layer (PHY) at device side.

4.1.1 Work overview

The goal of this implementation is to enable a dynamic network deployment with devices that have diverging needs. The TUD's PHY implementation is based on the generalized frequency division multiplexing (GFDM) waveform that can be flexibly configured depending on the users' needs.

This work has been done for two different platforms, which are called Type-1 and Type-2. The Type-1 implementation has been developed with the National Instruments (NI) Software Defined Radio (SDR) platforms, and consists of a host implementation for control, and a Field Programmable Gate Array (FPGA) implementation for real-time signal processing. Due to fast signal processing capability of the SDR platforms, the Type-1 device targets applications that require low-latency and/or moderate throughput. The Type-2 implementation has been developed for the Barkhausen Institut (BI) M³ platform, which is being designed with focus on security. This implementation has been done in Rust and will be integrated into the M³ hardware prototype of BI, as described in deliverable D3.3 of iNGENIOUS.

4.1.2 Type-1 – FPGA based Implementation

This subsection describes the implementation, and end-to-end measurements that were conducted with the Type-1 (FPGA based) platform.

4.1.2.1 Implementation

To enable the moderate throughput and low latency requirements of the Type-1 IoT devices compared to the Type-2 devices, the PHY has been implemented using the NI SDR platform taking advantage of the processing capabilities of the FPGAs. The PHY module is depicted in Figure 4.1-1, and it can be mainly divided into the host and the FPGA processing modules. The host handles the data link protocol, initializes the FPGA and the RF with the specified configuration, and controls for the execution of the PHY signal processing functions. The data transfer is done from the host to the FPGA



through host to target (H2T) first in first out (FIFO) and with the target to host (T2H) FIFO. The FIFOs are used for the buffering and the flow control between the host software and the FPGA. The baseband signal processing is deployed on the FPGA. Both the host and FPGA implementation are described in detail below.

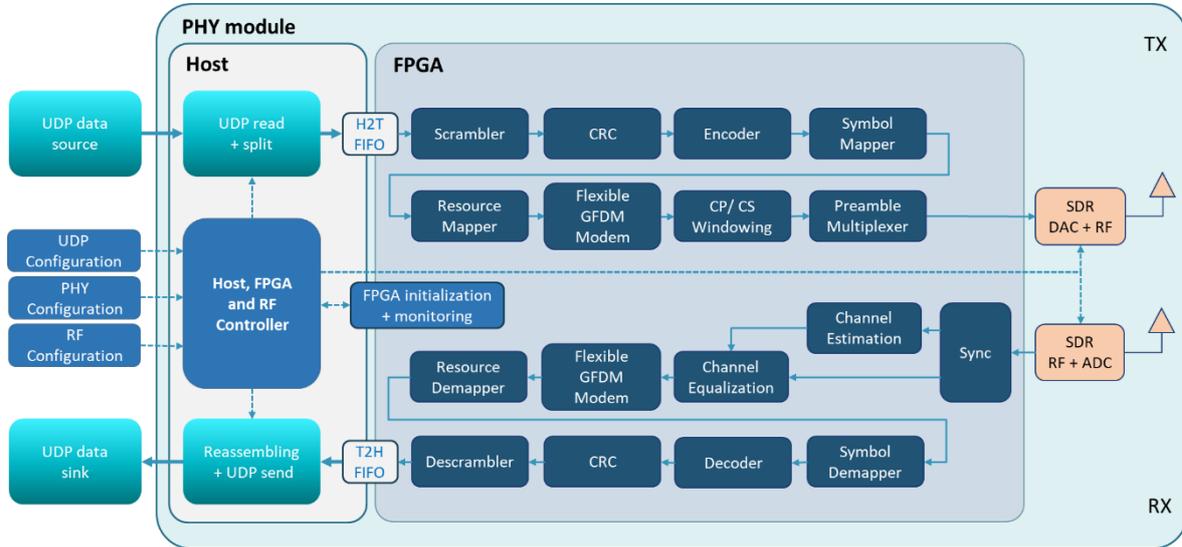


Figure 4.1-1. Overview of PHY modules used for baseband processing on Type-1 devices.

Host processing

Initialization: When starting the PHY module, the host initializes itself, the FPGA and the RF. As the first initialization step, the controller reads the input configuration for the User Datagram Protocol (UDP), the PHY and the RF. The second step is to calculate the necessary parameters for itself, the FPGA and the RF. The final step is to store the information on the host and forward it to the FPGA and the RF.

For the host and the FPGA processing, the number of bits which fit in a transmission frame $N_{\text{bitsFrame}}$ is critical. From the host perspective, it is relevant to correctly split and reassemble the UDP packets. From the FPGA perspective, this definition is needed to setup the baseband signal processing. In particular, the number of bits in a frame is calculated as:

$$N_{\text{bitsFrame}} = (N_{\text{on}} - N_{\text{pilots}})Q_m r - N_{\text{paddingEnc}} - N_{\text{CRC}} - N_{\text{overheadUDP}},$$

where N_{on} describes the number of active subcarriers. $N_{\text{pilots}} \in \{4,8\}$ pilots are used for the normalization of the QAM constellation. The number of bits which fit in a QAM constellation is given as $Q_m \in \{2,4,6\}$. A convolutional code with a code rate $r = 0.5$ is used and $N_{\text{paddingEnc}} = 8$ padding bits are utilized for the encoder. $N_{\text{CRC}} \in \{8,16\}$ is the CRC length and can be selected depending on the data length. In order to enable the transmission of long UDP packets that exceed the number of bits that the PHY can transmit in a single transmission, that is when $N_{\text{UDP}} > N_{\text{frame}}$, the host splits the UDP packet into slices which are smaller or equal than $N_{\text{bitsFrame}} - N_{\text{overheadUDP}}$. $N_{\text{overheadUDP}} = 32$ bits are used for overhead that is needed to properly reassemble the UDP packets, where one byte is used for the index of the frame, one byte for the

number of frames, and two bytes for the total number of bytes in an UDP block.

Operation: The data which should be transmitted with the PHY module can be generated by any source which is able to transmit UDP packets. UDP allows computer applications to send messages, referred to as datagrams, to other hosts on an Internet Protocol (IP) network. A simple example is a live video stream from a camera which can be transmitted to a specific IP address and port over UDP. The UDP packets are transmitted over the network to the IP address of the SDR on which the PHY module is implemented. The host reads the UDP packets from the port and splits them if necessary into smaller slices. These are stored in a queue and are transmitted to the FPGA if there is sufficient space in the H2T FIFO. In parallel, the host reads from the T2H FIFO and reassembles the UDP packets to transmit them to the predefined IP address and port over the network. The UDP stream can then be read, for example, by the VLC media player. Furthermore, the host is monitoring the FIFOs on the FPGA and the queues on the host, to detect overflows.

FPGA processing and RF

In comparison to the host processing, the FPGA implementation is pipelined to achieve moderate latency and sufficient throughput despite the complex baseband processing. For the transmitter, the incoming bits are read from the H2T FIFO. Then, the following operations are performed:

Scrambler: the bits are scrambled such that zeros and ones are equally distributed regardless of the source.

Cyclic Redundancy Check (CRC): a CRC with size 8- or 16-bits is calculated and concatenated to the scrambled bits. This allows the receiver to check whether the payload is a valid codeword or not. The small CRC of size 8 is used for the control channel where the amount of information bits is small, e.g., a few bytes, and the CRC of size 16 is used for larger packets of hundreds of bytes.

Channel Encoder: after the CRC, the bits are encoded via a convolutional encoder with generator polynomial $[133,171]_8$ in octal notation. The goal of the channel encoder is to include redundancy bits such that errors caused by the wireless channel and noise can be corrected at the receiver. The codeword size is made flexible depending on the amount of information to be transmitted by the source.

Symbol Mapper: the coded bits are then mapped onto a quadrature amplitude modulation (QAM) constellation set. The current implementation supports quaternary phase shift keying (QPSK), 16QAM and 64QAM constellation sets, which allow the transmission of 2,4, and 6 bits per symbol, respectively. This parameter can be tuned depending on the channel condition and application data rate. For example, for the control channel we employ QPSK constellation which is the most robust configuration in terms of bit error rate.

Resource Mapper: the QAM symbols are then mapped according to the active subcarriers and pilots are inserted. This step is important to select the



subcarriers to encode data, since the carriers in the edge and middle of spectrum suffer from filter attenuation and direct current (DC) components, respectively. Also, pilot symbols are inserted in this stage to estimate the channel gain and phase at the receiver side.

Modulator: the data symbols are then modulated using the GFDM waveform [20][21]. The objective of this component is to generate the discrete-time waveform that carries the symbol information. The advantage of having the GFDM model is its flexibility which allows flexible resource allocation by setting the number of subcarriers and subsymbols. In addition, filtering is also possible for spectrum shaping.

CP and CS: the cyclic prefix (CP) and the cyclic suffix (CS) are added. The CP is needed to force the channel to convolve circularly with the data, such that frequency domain equalization is possible. The CS and windowing are used to assemble consecutive waveforms without increasing the out-of-band emissions. This is necessary because a given waveform rarely begins with the same amplitude and phase with which the previous waveform has ended. In order to achieve a smooth transition between two waveforms, the CS is added, and filtering is used to align the beginning and the end of two consecutive waveforms.

DAC and RF: lastly, the signal is forwarded to the digital-to-analogue converter (DAC) and transmitted using the radio frequency (RF) module. The DAC has a resolution of 16 bits and creates the analogue signal. This signal is then forwarded to the RF module which performs the up-conversion to the carrier frequency and then transmits the signal over-the-air via an antenna. The RF properties can be defined by the user before the start of the PHY module. The bandwidth, the carrier frequency, the transmission and receive power can be set.

At the receiver side the signal is passed through the following components:

ADC and RF: the RF module performs the down-conversion of the received passband signal to the baseband signal, and the 14 bits analogue-to-digital converter (ADC) component transforms the analogue signal into discrete-time samples for the digital processing on the FPGA.

Synchronization: to determine the beginning of the frame, the synchronization is performed. The synchronization is based on the two continuous chirp signals of the preamble. A cross correlation of the received signal with the transmitted chirp is computed, as well as an autocorrelation between the received signal with a delayed version of itself which is time-shifted by the chirp length. Based on the amplitude of the cross- and the autocorrelation, the start of the frame is estimated.

Channel Estimation: both chirps of the preamble are averaged and used for the channel estimation. The channel estimator is implemented using the zero-forcing approach which can be efficiently implemented in the frequency domain. Zero-padding is applied according to the length of the payload. If N is larger than the length of the chirp, $N - N_{\text{chirp}}$ zeros will be inserted in the middle of the estimated impulse response to match the length of the payload.



Equalization: to reverse the distortions induced by the wireless channel which occur during the transmission, the payload is equalized using a zero-forcing algorithm based on the channel estimation.

Demodulator: the equalized payload is demodulated using the GFDM demodulator which results in the data symbols.

Resource demapper: the symbols which were transmitted using the active subcarrier are extracted. In addition, the pilots are averaged. Subsequently, the symbols are normalized and phase corrected according to the averaged pilots.

Symbol demapper: the normalized QAM symbols are demapped using hard-decisions to recover the coded bits.

Channel decoder: the coded bits are decoded with Viterbi algorithm to attain the received bitstream.

CRC: a CRC with the same size as the CRC generation in the transmitter is computed and compared to the received CRC to detect frame errors.

Descrambler: finally, the bit stream is descrambled according to the scrambling pattern which is used in the transmitter.

At the very last stage, the recovered bits are stored in the T2H FIFO such that the host can access this information.

4.1.2.2 Measurements

To determine the performance of the PHY module, end-to-end measurements were conducted using the tool *UDPtest* which was developed at the Vodafone Chair Mobile Communication Systems (TUD). This tool generates UDP packets, transmits them to an IP address and port and receives them from a port. The destination IP address and port can be defined, as well as the receive port on the host. The size and the time between the UDP packets can be set. The tool can measure the throughput, the latency and the frame error rate (FER), among other things. The measurement setup is visualized in Figure 4.1-2. The tool *UDPtest* is running on Device 1, which in an NI USRP-2974 [22], however, this could be any computer which is able to generate the necessary UDP throughput. The UDP packets are transmitted over an Ethernet connection to the Device 2, an NI PXIe-1082 [23] with and NI USRP-2944R [24]. This device runs the PHY transmitter according to Figure 4.1-1 and sends the signal over the wireless channel to the Device 3. This device is an NI USRP-2974 which runs the PHY receiver. The received UDP packets are forwarded over Ethernet to the Device 1, which measures the throughput, the FER and the latency.

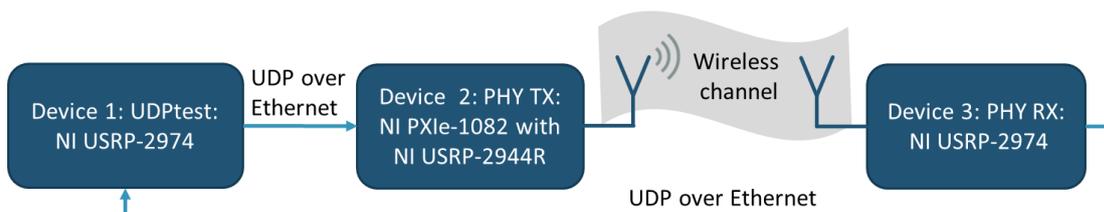


Figure 4.1-2. Measurement setup.



A wireless line-of-sight (LoS) channel was set with a distance of 4 meters in a controlled and static laboratory environment. A transmit power of 0 dBm was defined and the sample rate was fixed to $B = 20$ MHz. 3.75 GHz was used for the carrier frequency. For $N \geq 1024$, the CRC16 was applied and CRC8 otherwise. $N_{pilots} = 8$ are utilized for all payload configurations. However, a smaller number of pilots, $N_{pilots} = 4$, are used for the control channel. The time between the UDP packets was defined to be 200 μ s for $N \geq 2048$ and 100 μ s otherwise.

Without considering the host processing limitations, the throughput of the PHY is calculated based on the PHY parameters and the bandwidth as

$$\text{throughput PHY} = \frac{B}{N_{\text{frame}}} N_{\text{bitsFrame}}$$

where $N_{\text{frame}} = N_{\text{preamble}} + N_{\text{payload}}$ defines the number of samples in a frame. The number of samples in the preamble is defined as $N_{\text{preamble}} = 2N_{\text{chirp}} + N_{\text{CP}} + N_{\text{CS}}$ and for the payload as $N_{\text{payload}} = N + N_{\text{CP}} + N_{\text{CS}}$. The length of the chirp, the CP and the CS are defined as $N_{\text{chirp}} = 64$, $N_{\text{CP}} = 32$ and $N_{\text{CS}} = 15$, respectively. The overhead ratio, defining how much of the frame is used for the preamble, is crucial for the throughput of the PHY and is given as

$$\text{overhead ratio} = \frac{N_{\text{preamble}}}{N_{\text{frame}}} = \frac{N_{\text{preamble}}}{N_{\text{preamble}} + N_{\text{payload}}}$$

We propose 11 different configurations of the PHY in order to meet different application requirements. The configurations are divided into four groups, namely, H, M, L and C. Specifically, H, M and L represent the high, medium and low throughput configurations, respectively, while C is reserved for the control information. The calculated and the measured throughput for the different configurations are depicted in Table 9 where the FER = 0 was achieved for all configurations, meaning that the wireless link was reliable during these tests.

Table 9. Flexible PHY/MAC throughput measurements with *UDPtest*.

Config	N	N_{on}	K	M	QAM Order	Bytes per block	Throughput PHY (Mbps)	Overhead ratio	Throughput measured (Mbps)
H0	2048	1792	2048	1	64	666	5.87	0.08	2.94
H1	2048	1680	128	16	64	624	5.50	0.08	2.76
M0	2048	1792	2048	1	16	443	3.90	0.08	1.96
M1	2048	1680	128	16	16	415	3.66	0.08	1.84
M2	1024	896	1024	1	16	219	3.52	0.14	1.75
M3	1024	810	64	16	16	205	3.17	0.14	1.64
L0	102	896	102	1	4	108	1.73	0.14	0.86



	4		4						
L1	102 4	810	64	16	4	101	1.56	0.14	0.81
L2	512	448	512	1	4	53	1.44	0.24	0.42
L3	512	360	32	16	4	42	1.14	0.24	0.34
C0	64	52	64	1	4	4	0.21	0.61	Too few bytes for <i>UDPtest</i>

The measured throughput is computed with the tool *UDPtest* using the whole PHY module as seen in Figure 4.1-2. The difference between the measured throughput and the throughput of the PHY arises due to the limited processing speed of the host, which is not able to handle a higher UDP throughput without dropping packets.

The measured end-to-end latency is similar for all configurations ranging between 1.6 ms and 2.9 ms. In the PHY, a smaller frame size leads to a smaller latency. However, due to no significant smaller latency in the measurements, it can be concluded that the major latency comes from the host UDP processing and the UDP communications over the network.

The throughput of the configuration C0 was not measured, since the tool *UDPtest* cannot generate smaller UDP blocks than 30 bytes. However, the transmission of the control information should be robust, which is achieved with the configuration C0. This was verified by several measurements in both LoS and non-LoS wireless channels. It was observed that when the received signal is synchronized, the control information is reliably detected.

4.1.3 Type-2 – Rust based Implementation

This subsection describes the development of PHY functions written with the programming language Rust that runs on a microkernel-based operating system (OS) with a tile-based HW architecture. This hardware is currently being developed by BI and it is referred as M³. The envisioned usage of Type-2 devices lies in scenarios that require transmission of sporadic and small data packets. Nevertheless, the transmission of such information will be highly secure given its implementation using the M³ platform.

For enabling a light weight SDR PHY, the development of the required PHY chain algorithms is being carried out using Rust in a modular fashion for allowing easy upgrading, i.e., different components of the PHY chain can be individually modified.

The modules currently implemented and their structure, are illustrated in Figure 4.1-3. The computation and the data exchange are split into different classes. This helps to quickly adapt the PHY computations to new formats and diminishes the amount of transferred data.



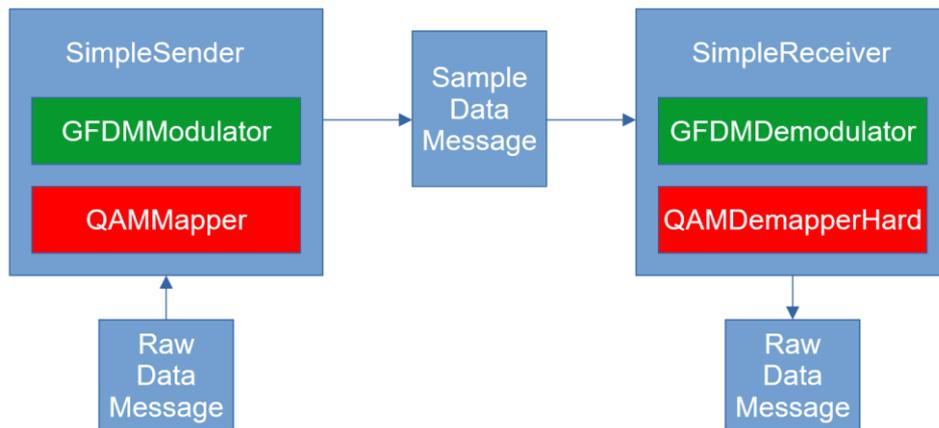


Figure 4.1-3. Overview of PHY modules used for baseband processing on Type-2 devices.

The raw data bytes are supplied to the “QAMMapper” by a class called “RawDataMessage”. After the modulation process, the signal samples are outputted in a “SampleDataMessage”. These samples could possibly be digital-to-analogue converted and transmitted via a RF front-end. On the receiver side, it is the other way around: the input are the raw signal samples supplied via a “SampleDataMessage” instance, and the output are the raw data bytes in a “RawDataMessage”. The SimpleSender and SimpleReceiver parts of the modem are built modularly of the classes responsible for QAM mapping/demapping and for the GFDM modulation. The QAM data symbols are modulated using the GFDM waveform [20][21]. The advantage of having the GFDM model is its flexibility which allows flexible resource allocation by setting the number of subcarriers and subsymbols. In the current approach, besides the limit of the machine representation of numbers, there are no restrictions on the two GFDM parameters for the number of subsymbols and subcarriers. The QAM symbols can be normalized in two ways: (i) with respect to peak power and (ii) with respect to average power. The normalization ensures that either that the peak power or the average power of the signal equals 1. The current supported QAM modulation orders are 4, 16, 256, 1024.

Figure 4.1-4 shows the current throughput performance achieved by our implementation running on a computer with an Intel Core i5-4200U processor (single thread).



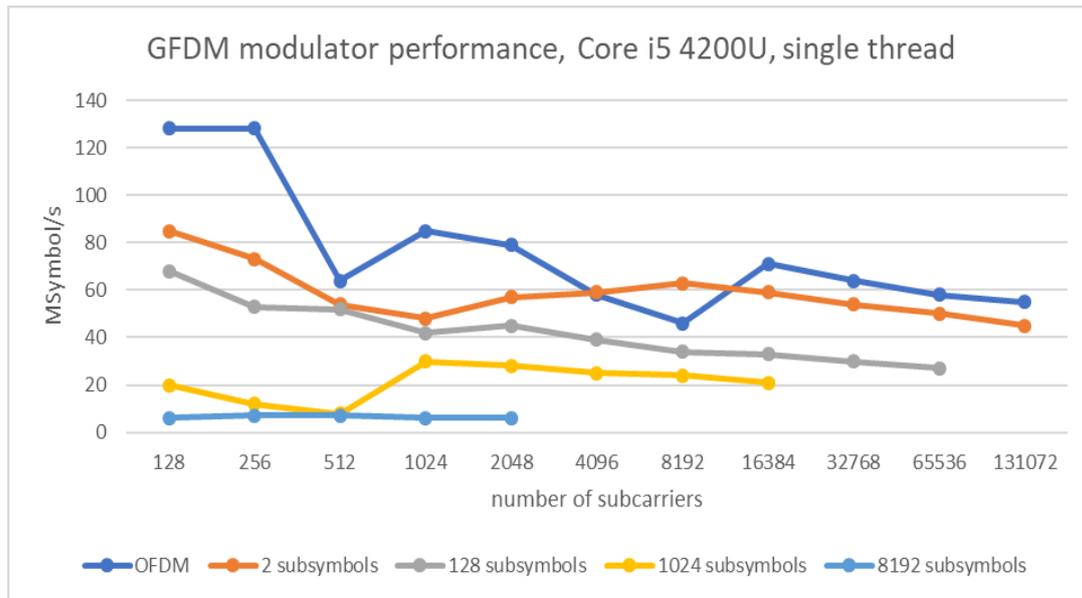


Figure 4.1-4. Throughput of GFDM modulator for different configurations of subsymbols and subcarriers.

Figure 4.1-5 shows the current encode/decode throughput performance achieved by our polar code assuming code rate equal to 1, i.e., not depend on the code rate. This performance was obtained on a computer with an Intel i5-4200U processor (single thread). The blue line, which shows an increase in performance with longer codewords, is for the Polar encoder, while the orange line is for the Polar decoder implementation. It can be seen that the current implementation of the decoder is significantly slower than the encoder. This could probably be improved by a more advanced decoder design.

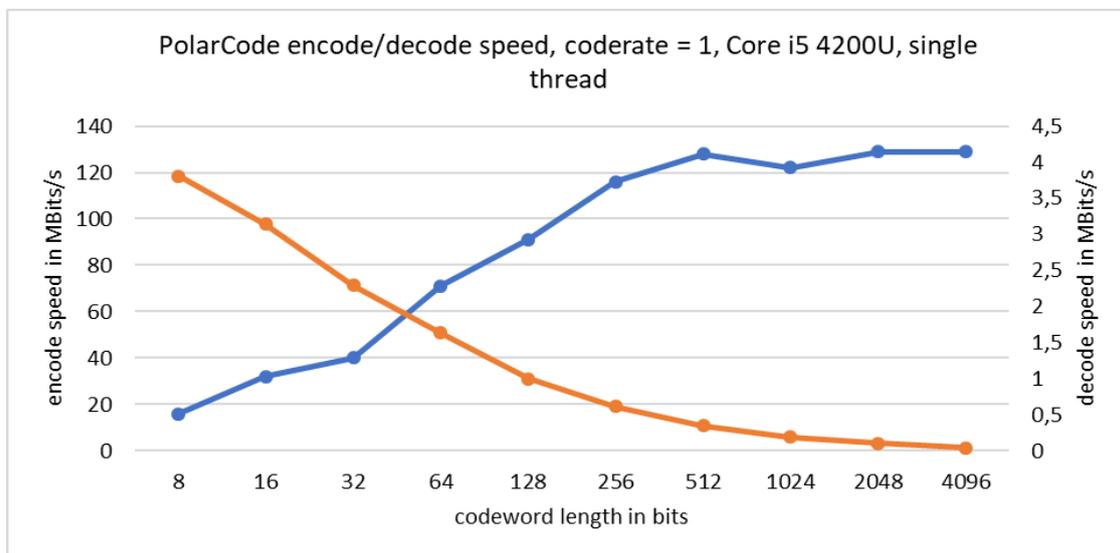


Figure 4.1-5. Encoder and decoder throughput performance.

For the QAM mapping, a performance of more than 450 MSymbol/s was reached. The hard demapping performance was about 35 MSymbol/s and the soft demapping performance could reach about 18 MSymbol/s for a QAM 256.

Performance tests of the distinct classes showed the capability of the software system for modulating and mapping the data with a speed of about 100 MSymbols/s. However, due to portability considerations and the abandonment of the *Fastest Fourier Transform in the West* (FFTW) library, these speeds are not reached in the current implementation suited for the M³ system. The next steps are a speed test of the FFTW library on the M³ system and using it with the current modulator framework. The developed PHY signal processing modules were successfully tested with over-the-air transmission with the help of a hardware-in-the-loop platform. The signal processing functions are executed on M³. Figure 4.1-6 presents the obtained power spectrum density as well as the demodulated data symbols assuming 4QAM mapping.

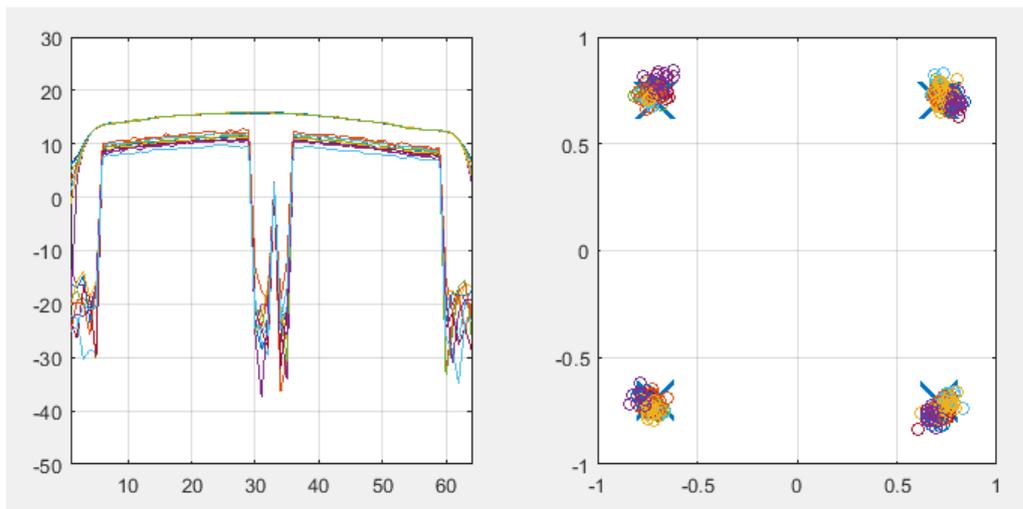


Figure 4.1-6. Power spectral density and received symbol constellation of flexible PHY running on M³.

4.1.4 Exploitation potential

Flexible PHY/MAC proof-of-concept is demonstrated through the Factory UC but it is also relevant to other use cases including diverse connectivity needs, e.g. the Transport UC.

Exploitation will focus on explorations for novel solutions in PHY/MAC design from real-time experiments. Final information about exploitation will be available in D7.3 “*Final dissemination, standardisation and exploitation*” [29].

Component: Flexible PHY/MAC (UE Side)

Relevant for use cases: Factory

Exploitation Potential:

- TUD:** The flexible PHY/MAC allows real-time experiments of new innovations in PHY/MAC, e.g., modulation waveforms and error correcting codes, under realistic channel conditions instead of using model-based simulations. With such technology, performance evaluation with hardware-in-the-loop can be easily carried out. Moreover, experiments done in realistic scenarios yield metrics that are highly correlated to the scenario under investigation, which is particularly useful for complex indoor environments such as factory plants. In addition, the custom design enables the development in a private network, such as industrial networks.



4.25G Modem

The 5G New Radio (NR) module, or 5G modem, is a UE device that is used to connect specific vertical components, such as robots, sensors, cameras, or AGVs to the network providing 5G service. The modem enables the components to communicate with the gNB wirelessly, enabling the machines to “speak 5G”.

4.2.1 Work overview

For an efficient implementation, 5G modems should be compact and integrated within the device; power efficient; simple, and usable as ‘plug and play’ devices.

Within the context of iNGENIOUS, 5G modems provided by 5CMM have been integrated into the complete end-to-end system. These modems are both NSA (non-standalone) and SA (standalone) compatible. Note that an SA deployment here refers to the architecture where 5G RAN (i.e. NR) is connected directly to the 5G core network (5GC), while NSA refers to the architecture where 5G service via NR is built over an existing 4G core network.

4.2.2 Technical features

The 5G modem is connected via Ethernet or USB to the end device and connects to the network via its integrated or external antennas. It is a powerful, versatile, and compact device designed to bring all the advantages of the new 5G RAN technology.

The modem has simplified its electronics while minimizing the power consumption and cost. An overview of the 5G modem from 5CMM is shown in Figure 4.2-1.



Figure 4.2-1. 5G modem.

The 5G modem from 5CMM consists in only one box with an ethernet and a power supply port. The electronics consist of a combination of a Raspberry Pi

4 connected to the 5G Broad, which is the name for the 5G communications module developed by 5CMM. A full integration has been made, resulting in a solution with a simple power supply. An illustration of the modem architecture is shown in Figure 4.2-2.



Figure 4.2-2. 5G modem architecture.

The 5G module can be configured via the Raspberry Pi through Network Manager/OpenWRT, and the user can access the modem via AT commands. More technical specifications of the 5G modem are shown in Table 10.

Table 10. 5G modem technical specifications.

Interfaces	
Antennas	Up to 4 external, medium and high bands Up to 2 external, low, medium and high bands
Ethernet (1Gbps)	1
(U)SIM	1
Frequency bands	
5G NR (NSA and SA)	n41/n77/n78/n79/n1/n3/n5/n7/n8/n20/n28/n38/n40
LTE-FDD	B1/B3/B5/B7/B8/B18/B19/B20/B26/B28/B32
LTE-TDD	B34/B38/B39/B40/B41/B42/B43
WCDMA	B1/B3/B5/B6/B8/B19
Data rates (Theoretical Maximum)	
5G SA Sub-6GHz	2.1 Gbps (DL)/ 900 Mbps (UL)
5G NSA Sub-6GHz	2.5 Gbps (DL)/ 650 Mbps (UL)
LTE (cat 16)	1.0 Gbps (DL)/ 200 Mbps (UL)
WCDMA	42 Mbps (DL)/ 5.76 Mbps (UL)
Mechanical	
Dimensions (H x W x D)	9 cm x 13.8 cm x 5.7 cm
Board Weight	<150 g
Electrical	
Supply voltage range	9V - 12V DC
Connector type	Molex 46999-0014
Output power	Class 3 (24 dBm +/-3 dB) WCDMA bands Class 3 (23 dBm ±2 dB) LTE bands Class 3 (23 dBm ±2 dB) 5G NR bands Class 2 (26 dBm ±2 dB) LTE B38/B40/B41/B42 bands HPUE Class 2 (26 dBm ±2 dB) 5G NR n41/n77/n78/n79 bands HPUE
Management	
5GBD management options	Remote management and Local management (Web interface)
Temperature Range	
Operating temperature	-30°C to +75°C
Extended temperature	-40°C to +85°C



The modem supports the following technical features:

- *5G native mode*: both 5G NSA and 5G SA modes are supported, including options 3x, 3a and 2. 3G/4G connectivity is additionally supported.
- 5G NR Rel-15 support.

The 5G modem has been validated against several 5G networks in D4.2 of iNGENIOUS [28], testing its performance in different conditions and under various scenarios. During the project, the 5G modem has been used in UPV campus for pre-testing of Factory use case (both commercial network and non-public network of UPV, formed by an Amarisoft gNB and CumuCore 5GC), as well as in CumuCore premises with Nokia radio equipment. The performance tests indicated RTT latency values between 10 and 20 ms, and DL throughput values up to 400-500 Mbps, depending on the bandwidth and configuration selected.

Such integration and validation work has served us towards the final trials in the Factory UC of iNGENIOUS. The 5G modem will be integrated at the University of Burgos and will be used for the KPI validation, as described in D2.1. The final results of such validation will be included in iNGENIOUS deliverable D6.3.

4.2.3 Exploitation potential

5G modem covers the AGV use case, although the integration between the AGV and the 5G modem will also be performed in the Factory use case.

Exploitation will focus on the commercialization of the components and as an integrated service. Final information about exploitation will be available in D7.3 “*Final dissemination, standardisation and exploitation*” [29].

Component: 5G Modem

Relevant for use cases: AGV, Factory

Exploitation Potential:

- **5CMM's 5G modem** has a clear exploitation route. The modem will be commercialized as a compact and configurable device for enabling 5G in any object. **5CMM's cockpit** is a solution for remote and autonomous driving using immersive devices such as MR glasses, haptic gloves, pedals, wheels, controllers, and keyboards among other devices all integrated in just one cockpit. This solution will be exploited in the market as an autonomous and remote driving solution for factories and maritime ports or terminals.



5 Relation to iNGENIOUS Use Cases

All technical work components developed within T3.1 of WP3 are relevant, in different ways, to one or more iNGENIOUS use cases as described in deliverable D4.1 [2]. Table 11 lists these components and how they map to the various use cases, also in terms of contribution.

Table 11: Simulation environments mapped to iNGENIOUS use cases

Use Cases →	Factory	Transport	Port Entrance	AGV	Ship
WP3 T3.1 Component ↓					
5G-IoT system-level performance analysis	Analysis				
5G modem	Innovation Demo			Innovation Demo	
Flexible HW/SW architecture	Innovation Demo				
NB-IoT link-level performance analysis			Analysis		
Cellular-IoT modem power consumption analysis		Analysis			Analysis
Impact analysis of NTN solutions on device		Analysis			Analysis
Enhancements for NTN		Innovation Concept			Innovation Concept

Three general options are considered for the mapping of a given work component:

- **Analysis**, for those components involving performance analysis, such as quantitative performance evaluation or qualitative impact evaluation, relevant to connectivity technologies considered in an iNGENIOUS use case.
- **Innovation Concept**, for those components involving innovative solutions which are relevant to an iNGENIOUS use case but will not be demonstrated.
- **Innovation Demo**, for those components involving innovative solutions which will be implemented and showcased as part of an iNGENIOUS use case.

Further detail of each use case description can be found in D2.1 [1].



6 Conclusion

This document describes the innovations of iNGENIOUS regarding IoT devices and communication beyond 5G. In particular, this deliverable has detailed how the innovations related to the IoT connectivity task of WP3 relate to the iNGENIOUS architecture and use cases, showing the impact of the innovations in real-world scenarios. The document has been divided in three parts, namely performance evaluations and analysis, solutions for efficient cellular IoT (C-IoT) air interface, and solutions for versatile IoT communication.

Regarding IoT technologies evaluations, the performance of some of the current state-of-the-art technologies which are part of the iNGENIOUS project are evaluated within the context of specific envisaged iNGENIOUS use cases. This evaluation is part of the iNGENIOUS target to verify technological requirements related to 5G-IoT technologies by following a two-fold approach based on simulations and real measurements. To this end, a system-level simulation campaign has been carried out for the Factory use case. The analysis concluded that almost all connectivity requirements defined at beginning of the project for this use case can be fulfilled, apart from the connection density target for sensors and the reliability target for robots and cameras, where relaxing of these KPIs for the real trials and demonstrations has been recommended. In addition, NB-IoT link-level and modem power consumption evaluation campaigns were carried out (supported by field measurements for realistic communication and devices configuration model). From link-level analysis, we get key observations for network optimisation, such as: 1) spectral efficiency (SE) gain from increased number of transmission repetitions is higher for high order modulation and coding schemes (MCS) but generally decreases as repetitions increase; 2) appropriate combination of MCS and repetitions can balance the SE gain for given signal conditions, whilst maintaining the same reliability of communication; 3) the additional reference signals in in-band mode compensate for the high number of repetitions in the standalone case; 4) use of additional antenna port makes use of frequency/time diversity to avoid interference and noise, and can be key to maintain a permanent successful connection or good coverage. From device power analysis, we identify power consumption optimisation solutions that can significantly reduce conventional modem total energy usage, e.g., up to 80%, for several scenarios of IoT communication. These insights include improved sleep and resynchronization implementations, optimized system information acquisition, and support of standardized energy reduction features such as PSM, eDRX and RRC Connection Release Acknowledgment. We finally assess the expected device battery lifetime for respective IoT sensor devices and applications in Transport and Ship use cases and we conclude that the proposed NB-IoT modem power reductions are important to ensure the lifetime targets in both use cases. While NB-IoT solutions can satisfy the 5-year lifetime target for ship tracking sensors in most scenarios, the 12-year target for rail tracking sensors target can only be achieved with power-optimised device, good cellular coverage conditions, and low data application, unless energy harvesting technology is available for battery charging.



Regarding solutions for efficient C-IoT air interface, the technical research within this WP3 task has focused on the ongoing work of the 3GPP on evolution of C-IoT technologies and specifically for non-terrestrial networks (NTN), applicable to the Transport and Ship use cases. An analysis of the various proposed NTN UE features and aspects for enhancement within 3GPP standardization body was firstly performed. We observe that several proposed enhancements can potentially have significant impact on IoT chipset implementation needs. We provide directions to avoid specification of features with significant effects to chipset design (e.g., increased number of HARQ processes running in parallel, new PRACH formats and preamble sequences to support non-GNSS-capable devices) or to reduce such impact of useful features (e.g. introduction of GNSS-specific measurements gaps to support time/frequency updates during modem activity for low-complexity GNSS integrated device solutions). Directions are also provided on what has to be changed on legacy device architecture design to support NTN operation, including modifications in: 1) RF front-end hardware, to support increased transmit power, sensitivity, and bands for satellite operation; 2) antenna subsystem hardware, to provide higher gain and tracking functionality; and 3) modem software, to support protocol and procedure changes. Moreover, proposed solutions for satellite-based connectivity are presented regarding: 1) efficient design of radio link layer (RLC) targeting to improve latency and power performance in small data burst applications; and 2) efficient design of handover procedures to improve latency and ease of implementation in devices.

Regarding solutions for versatile IoT communication, iNGENIOUS focused first on investigating a flexible PHY/MAC solution that exploits software defined radio for realizing a heterogeneous IoT network able to attain the requirements presented by the Factory use case. This use case covers various scenarios that can include multiple radio technologies and adaptable radio access interfaces, requiring dynamic network deployment with devices that have diverging needs. The implementation work was done for two different platforms: 1) a host implementation for control and a Field Programmable Gate Array (FPGA) implementation for real-time signal processing, targeting applications that require low-latency and/or moderate throughput, and 2) an implementation in Rust for the Barkhausen Institut (BI) M³ platform, designed with focus on security and sharing of hardware resources between SDR and application workload. In addition, the development of 5CMM 5G modem within the iNGENIOUS project has been presented. This device solution is employed to connect specific UEs to the 5G network, adapting to the needs of Factory and AGV use cases. The performance tests during integration and validation work have indicated satisfactory RTT latency values of 10-20 ms and DL throughput values up to 400-500 Mbps, and the device will be used for KPI validation in the project's respective final trials.



References

- [1] iNGENIOUS, Deliverable D2.1, “Use Cases, KPIs and Requirements”, March 2021.
- [2] iNGENIOUS, Deliverable D2.4, “System and Architecture Integration”. September 2022
- [3] iNGENIOUS, Deliverable D4.1, “Multi-technologies network for IoT”, May 2021
- [4] <https://www.nsnam.org/>
- [5] <https://5g-lena.cttc.es/>
- [6] A. Ramos, Y. Estrada, M. Cantero, J. Romero, D. Martín-Sacristán, S. Inca, M. Fuentes, and J. Monserrat, “Implementation and calibration of the 3gpp industrial channel model for ns-3,” in Proceedings of the WNS3 2022.
- [7] 3GPP, TR 38.901 v16.1.0, Study on channel model for frequencies from 0.5 to 100 GHz (Release 16), December 2019.
- [8] 3GPP, TS 22.104 v18.2.0, Service requirements for cyber-physical control applications in vertical domains, September 2021.
- [9] <https://5g-ppp.eu/wp-content/uploads/2020/02/5G-IA-Final-Evaluation-Report-3GPP-1.pdf>
- [10] iNGENIOUS, Deliverable D3.1, “Communication of IoT Devices”, May 2021.
- [11] M. Tutunovic, Master’s Thesis, “Performance analysis of NB-IoT using link level simulations and field measurements”, 2022, [Available Online: [UPV website](#)]
- [12] S. Pratschner and B. Tahir and Lj. Marijanovic and M. Mussbah and K. Kirev and R. Nissel and S. Schwarz and M. Rupp, “Versatile mobile communications simulation: the Vienna 5G Link Level Simulator”, EURASIP Journal on Wireless Communications and Networking, September 2018
- [13] 3GPP, TR 38.821 v16.1.0, Solutions for NR to support Non-Terrestrial Networks (NTN), June 2021.
- [14] 3GPP, TR 36.763 v17.0.0, Study on Narrow-Band Internet of Things (NB-IoT) / enhanced Machine Type Communication (eMTC) support for Non-Terrestrial Networks (NTN), June 2021.
- [15] R2-2108460, “On RLC t-Reassembly for NTN”, Sequans Communications, 3GPP TSG-RAN WG2 Meeting #116-e, November 2021.
- [16] R2-2010170, “Additional RLC and PDCP aspects for NTN”, Sequans Communications, 3GPP TSG-RAN WG2 Meeting #112-e, November 2020.
- [17] 3GPP, TS 38.331 v16.10.0, NR; Radio Resource Control (RRC); Protocol specification, October 2022.
- [18] iNGENIOUS, Deliverable D3.3, “Secure, private and more efficient HW solutions for IoT devices”, December 2022.
- [19] R2-2210729, “NTN Configuration at handover and CHO”, Sequans Communications, 3GPP TSG-RAN WP2 Meeting #119bis-e, October 2022.
- [20] N. Michailow et al., “Generalized Frequency Division Multiplexing for 5th Generation Cellular Networks,” in IEEE Transactions on Communications, vol. 62, no. 9, pp. 3045-3061, September 2014, doi: 10.1109/TCOMM.2014.2345566.
- [21] Z. Li, A. Nimr and G. Fettweis, “Implementation and Performance Measurement of Flexible Radix-2 GFDM Modem,” IEEE 2nd 5G World Forum (5GWF), 2019, pp. 130-134, doi: 10.1109/5GWF.2019.8911718.



- [22] National Instruments. "Specifications. USRP-2974. 10 MHz to 6 GHz, x86 Processor, GPS-Disciplined OCXO, USRP Software Defined Radio Stand-Alone Device," May 2019, url: <https://www.ni.com/pdf/manuals/377417c.pdf>.
- [23] National Instruments. "PXI Express. NI PXIe-1082 User Manual," March 2016, url: <https://www.ni.com/pdf/manuals/372752c.pdf>.
- [24] National Instruments. "Specifications. USRP-2944. Software Defined Radio Reconfigurable Device Specifications," July 2017, url: <https://www.ni.com/pdf/manuals/375724b.pdf>.
- [25] 3GPP, TR 38.875 v17.0.0, Study on support of reduced capability NR devices, March 2021.
- [26] 3GPP, TR 38.821 v16.1.0, Solutions for NR to support Non-Terrestrial Networks (NTN), June 2021.
- [27] 3GPP, Thales, RP-202908, Solutions for NR to support non-terrestrial networks (NTN), Work item description, December 2020.
- [28] iNGENIOUS, Deliverable D4.2, "Smart NR and NG-RAN IoT designs", January 2022.
- [29] iNGENIOUS, Deliverable D7.3, "Final dissemination, standardisation and exploitation", March 2023.

