

## Titlepage

Abstract

Report info

## Preface

Who are we?

Prerequisite for readers

Acknowledgements

## Introduction

### Motivation

Why is the topic of interest?

- New technology

- IoT development

- Massive number of devices

- 5G

### Problem analysis

What are the problem?

- Many standards

- Not really tested

- Nothing really deployed

- Unity

Why do we have the problem?

- New technology

- Money

- Competiveness

### **Problem statement!**

(To allow the companies to compete effectively, in the new IoT market a main standard communication protocol is needed. The standards in question has to be evaluated based on their supported amount of users, energy efficiency and reliability, determining which is most efficient for the IoT devices.)

## Solution analysis

What have been done before?

- Last thesis

- Some test into battery lifetime

- Theoretic calculations

What could be a solution for the problem?

- An IoT emulator

- Comparison of theoretic metrics

- Choose blindly

- Deploy a test case

What are we working with and why?

- An IoT emulator

  - Focus on massiveness and power consumption

  - Why not reliability

- Why only NB-IoT

- Focus on being able to test devices alongside the emulation (pseudo deployment)

## NB-IoT Protocol Description

### System overview

- Functionality description

### Protocol stack

- Layer responsibility

- Layer description

### Network access

- Cell search and sync

- Random access procedure

- Connection control

  - eDRX, PSM, resume

### Data transmission

- User plane

  - TBS, rep

Data plane

TBS

## System setup

### Overview

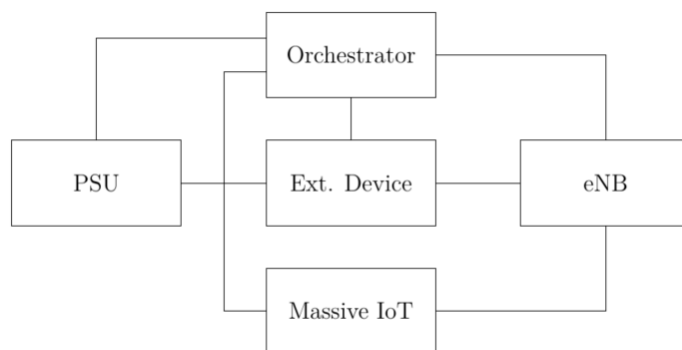
First, there should be a conceptual diagram of the setup; this should go into a description of how the emulator is actually put together.

There should be a mention of how an external device can be put into the system to be tested also due to the cellular nature of NB-IoT.

The physical connection needs to be explained, how to connect everything and where to put attenuators combiners and so forth.

There could also be a section of practical limitation due to digitalization.

Should end with differentiation of main and auxiliary components.



### eNB

Here should be mentioned that the BSE is not primary concern, therefore use of existing BSE. It should also be mentioned that it should be changeable so commercial BS can also be tested. It should end with use we use Amarisoft LTE 100 as primary and support it with UXM.

#### Amarisoft

Here should be a list of relevant features it have, and how to use them. It should also be mentioned, how we can access it from a main PC to set these features. It should be described how the core network interacts with the eNB. It should also be described where to put USIM data to allow network attach.

#### UXM

Here should be a list of relevant features it have, and how to use them. It should also be mentioned, how we can access it from a main PC to set these

features. It should be described how the core network interacts with the eNB. It should also be described where to put USIM data to allow network attach.

### **Massive IoT**

Here should be a description of the software from SRS. How the core structure of the code is and how it is expanded to accommodate multiple UEs. There should be a description of how to change key parameters in the code and how to use the system from a main PC (or if the main PC should host the MassM2M).

### **PSU**

Short description of the feature of the PSU and the limitation (can not measure and change settings simultaneously). There should also be a description of how the PSU responds to SCPI commands.

### **External device**

An explanation of why it is nice to include (possibility to test commercial devices). Some examples of commercial devices.

### **Orchestrator**

What are the function of the orchestrator? Mention the use of TAP. A list of all connections and communication protocols.

## **Performance evaluation**

Here should be an introduction of what we will test (the emulator and/or the protocol).

### **Requirements**

Here should be a list of all requirement that should be tested and criteria for passed not passed.

Requirements:

#### **Emulator**

Amount of users (CPU,RAM)

Configurability (channel, number of UEs, etc.)

Power control

#### **Massiveness**

Time to connect vs. connection request per second

Data rate vs. number of users

Spectrum use vs. number of users

Interference level vs. number of users

## Power

Energy consumption for attach.

Energy consumption vs. data rate

Energy consumption vs. coverage level

Energy consumption vs. operation mode

Energy consumption vs. number of UEs

Energy consumption vs. UE state (Connected (cDRX), eDRX, PSM, Off)

## Test setup

Here should be a description of the general setup (including figure) used in all test and a list of baseline values for all parameters. Including physical setup, BSE, UEE.

## Test procedure

Here should be a step by step procedure of all test for all requirements, maybe put tapplans in appendix.

## Results

Here should be a list of all results produced from the test. A short note should be attached to the results if the requirement is passed and if not why not.

## Discussion

What can be derived from the results?

Tradeoffs

## Conclusion

Summary of conclusion from report

What could have been done different if not?