NIST Technical Note
NIST TN 2286

Collection Methods for High-SNR I/Q Recordings of FDD LTE User Equipment Emissions

M. Keith Forsyth
Aric W. Sanders
Daniel G. Kuester
Adam Wunderlich

This publication is available free of charge from:
https://doi.org/10.6028/NIST.TN.2286

NIST NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
U.S. DEPARTMENT OF COMMERCE

NIST Technical Note
NIST TN 2286

Collection Methods for High-SNR I/Q Recordings of FDD LTE User Equipment Emissions

M. Keith Forsyth
Aric W. Sanders
Daniel G. Kuester
Adam Wunderlich
Spectrum Technology and Research Division
Communications Technology Laboratory

This publication is available free of charge from:
https://doi.org/10.6028/NIST.TN.2286

May 2024

U.S. Department of Commerce
Gina M. Raimondo, Secretary

National Institute of Standards and Technology
Laurie E. Locascio, NIST Director and Under Secretary of Commerce for Standards and Technology

Certain equipment, instruments, software, or materials, commercial or non-commercial, are identified in this paper in order to specify the experimental procedure adequately. Such identification does not imply recommendation or endorsement of any product or service by NIST, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

Publication History
Approved by the NIST Editorial Review Board on 2024-04-25

How to cite this NIST Technical Series Publication:
Forsyth M. Keith, Sanders Aric W., Kuester Daniel G, Wunderlich Adam (2024) Collection Methods for High-SNR I/Q Recordings of FDD LTE User Equipment Emissions. (National Institute of Standards and Technology, Gaithersburg, MD), NIST TN 2286. https://doi.org/10.6028/NIST.TN.2286

Author ORCID iDs
M. Keith Forsyth: 0000-0003-4018-6893
Aric W. Sanders: 0000-0002-2305-543X
Daniel G. Kuester: 0000-0003-0711-5700
Adam Wunderlich: 0000-0002-8463-9156

Abstract

This report documents collection methods for high signal-to-noise ratio (SNR) in-phase and quadrature (I/Q) radio frequency (RF) recordings of long-term evolution (LTE) uplink emissions from a commercial-off-the-shelf (COTS) handset in a fully conducted environment. A base station emulator was used to configure the handset to statically transmit over various LTE physical resource block configurations for 1.4 MHz, 3 MHz, 5 MHz, and 10 MHz uplink channels centered at 1770 MHz in LTE Band 66, which is a frequency-division duplex (FDD) band. The aim of this effort was to acquire a dataset that could be used to aid the development of computational models for emissions from real-world communication hardware. The complete dataset is publicly available at https://doi.org/10.18434/mds2-3177.

Keywords

In-Phase and Quadrature Data; Long Term Evolution (LTE); Wireless Communications.

Table of Contents

List of Tables

List of Figures

Acronyms

3GPP 3rd Generation Partnership Project
ADB android debug bridge
AWS-3 Advanced Wireless Services 3
CLI command line interface
COTS commercial off-the-shelf
CP cyclic prefix
csv comma separated value
DL downlink band
eNB evolved node B
FDD frequency-division duplex
HTTP hypertext transfer protocol
IP internet protocol
I/Q in-phase and quadrature
LAN local area network
LO local oscillator
LTE long-term evolution
NI National Instruments
OTA over the air
PRB physical resource block
RAN radio access network
RF radio frequency
RMC radio measurement channel
SCPI standard commands for programmable instruments
SMA SubMiniature version A
SMT surface mount technology
SNR signal-to-noise ratio
TDMS technical data management system
TTI transmission time interval
UDP user datagram protocol
UE user equipment
UL uplink
USB universal serial bus
VISA virtual instrument software architecture
VST vector signal transceiver

Acknowledgments

We would like to thank Dr. Duncan McGillivray and Mr. Jack Sklar for productive conversations about the RF setup as well as the output data, Mr. Aziz Kord for his insight into the configuration of the base station emulator and its quirks, and Mr. Jason Coder for his insight into LTE emissions, his knowledge of the cabled handset, and his assistance in coordinating sharing of required test equipment.

Author Contributions

Author	Roles
M. Keith Forsyth	test automation, data analysis, writing - original draft
Aric Sanders	testbed validation, visualization, writing - original draft
Daniel Kuester	testbed RF design, data analysis, writing - original draft
Adam Wunderlich	conceptualization, experimental design, writing - review & editing, supervision

1. Introduction

Assessing potential impacts of new radio frequency (RF) spectrum users on incumbent systems in a given frequency band typically requires some combination of laboratory tests, field measurements, and modeling. For such assessments, it is often necessary to collect recordings of RF emissions from real-world commercial off-the-shelf (COTS) systems that will use the band.

In this work, we present measurement methods used to collect high signal-to-noise ratio (SNR) in-phase and quadrature (I/Q) recordings of 4G long-term evolution (LTE) user equipment (UE) emissions over various prescribed physical resource block (PRB) occupancy configurations. Specifically, using a fully conducted experimental configuration with a base station emulator, we recorded emissions from a single COTS UE operating in LTE Band 66, a frequency-division duplex (FDD) band arising from the Advanced Wireless Services 3 (AWS-3) auction in 2015 [1]. Five second I/Q recordings were collected over 1.4 MHz, 3 MHz, 5 MHz, and 10 MHz LTE uplink channels, each centered at 1770 MHz, over possible PRB configurations defined in the 3rd Generation Partnership Project (3GPP) LTE standard [2]. In total, 895 unique PRB configurations were captured; see Table 4 for a summary. The dataset resulting from this effort is publicly available at https://doi.org/10.18434/mds2-3177.

The primary motivation for this data collection effort was to support the development of data-driven computational models of RF emissions from LTE UEs. In particular, a need exists for recordings of an isolated UE without the complexities of scheduling dynamics, multiple UEs, or channel propagation effects. A secondary motivation was to construct a comprehensive library of LTE captures over a wide range of spectrum occupancy states that can be used for interference susceptibility testing.

The resulting I/Q dataset can be seen as filling a gap between fully synthetic data produced by commercially available laboratory test equipment and radiated emissions collected by recent measurement campaigns in laboratory [3] and field [4] settings. The authors are unaware of a similar set of high-SNR LTE recordings that has been published to date.

2. Testbed Configuration

2.1. Overview

Data acquisition was performed with an automated RF testbed. A testbed diagram is shown in Fig. 1 and an equipment list is given in Table 1. The primary components were an LTE handset (denoted UE 1 in the testbed diagram), a base station emulator, and a vector signal transceiver (VST) configured to record I/Q data. The set-up was fully conducted with the UE located inside an RF-shielded box. The conducted experiment enabled a high dynamic range acquisition without the limitation of an over the air (OTA) RF channel. The signal fed to the modified UE was split to the Tx & COM input channels of the base station emulator, with the uplink (UL) signal split again to send to the record node. In-line attenuation was added in order to prevent high level signals from damaging any of the RF equipment.

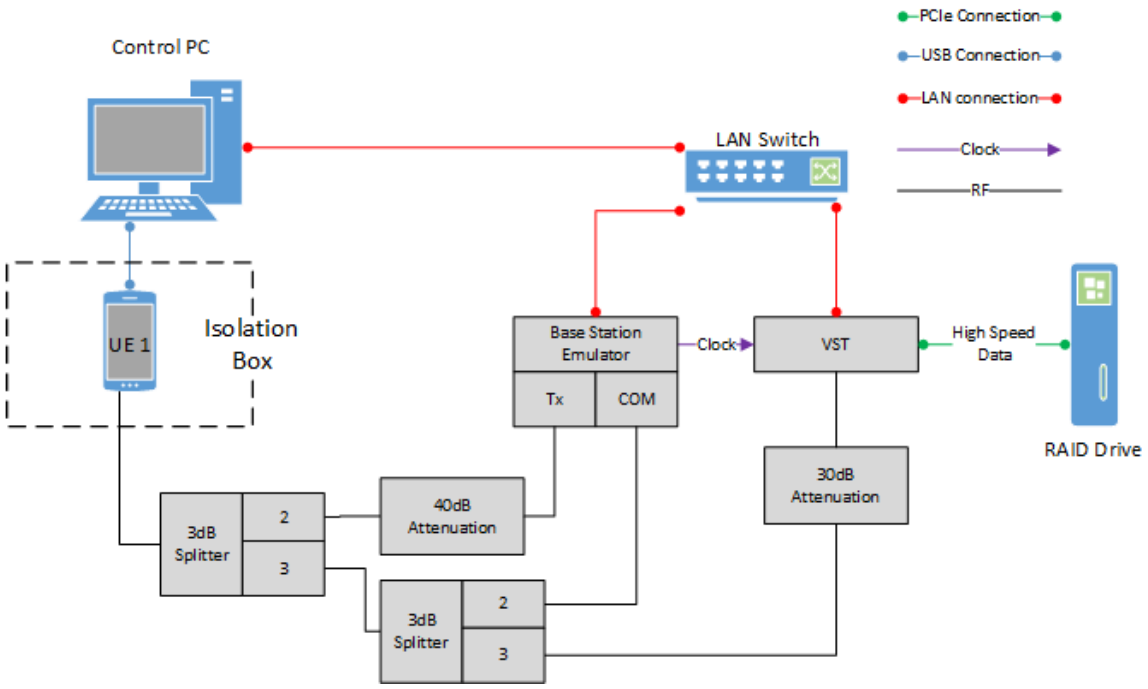

Fig. 1. Testbed Diagram. Component details in Table 1

Table 1. Equipment List

Component	Make & Model	Purpose
VST	National Instruments PXIe-5840	IQ acquisition
Base Station Emulator	Rohde & Schwarz CMW500	Receive UL transmissions
UE 1	Samsung S8	Provide UL transmissions
Isolation Box	JRE0709	Isolate UE & base station emulator

2.2. Base Station Emulator Configuration

The base station emulator was employed to create a radio access network (RAN) which provided static scheduling configurations that constrain the UL emissions of the UE. The instrument was configured to produce a FDD cell in Band 66 with an UL channel center frequency of 1770 MHz, a downlink band (DL) channel center frequency of 2170 MHz, and was set to operate in closed-loop power control mode with a nominal target power (P_0) of -20 dBm. The LTE channel bandwidth was taken to be 1.4 MHz, 3.0 MHz, 5.0 MHz, or 10.0 MHz. A typical configuration can be seen in the screenshot of the LTE signaling user interface of the base station emulator in Fig. 2.

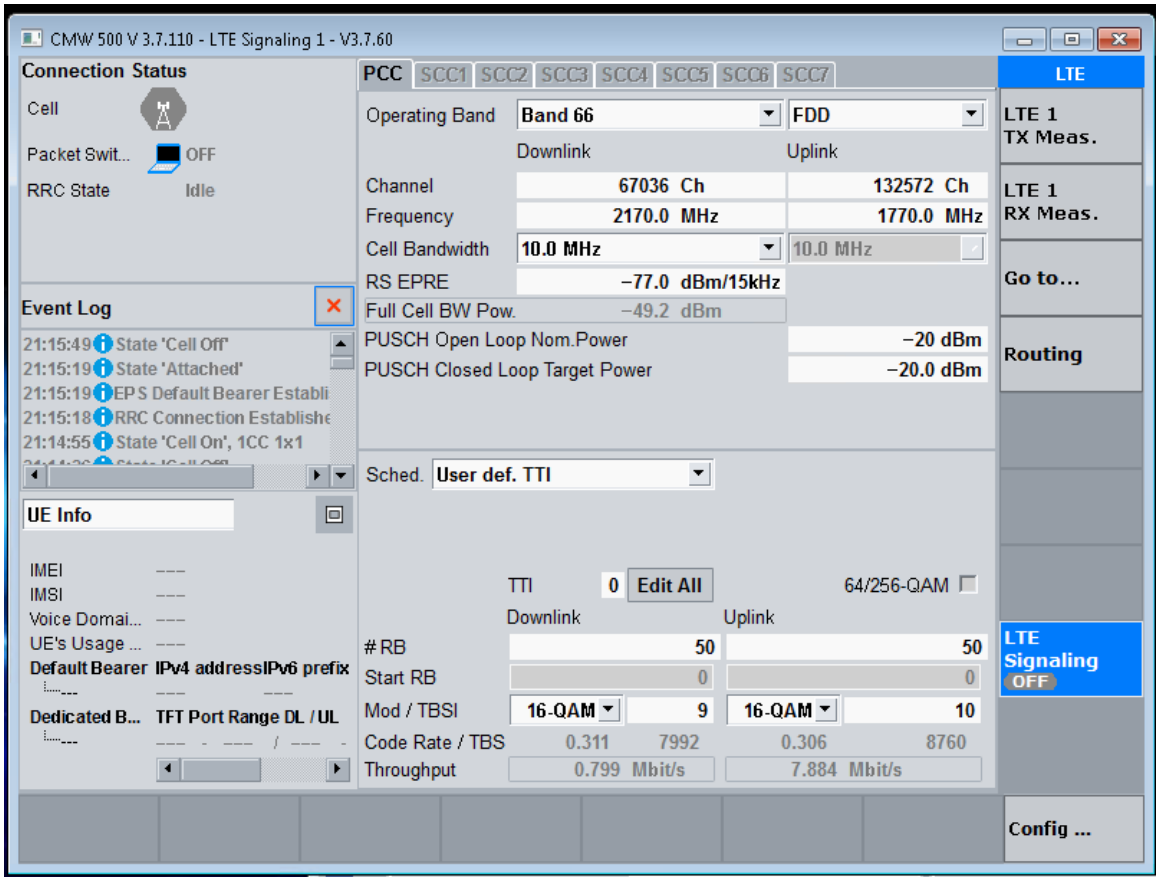

Fig. 2. Screenshot of LTE signaling user interface of base station emulator.

To ensure a robust connection with the UE, the emulator transmission was initially started with the full bandwidth of the allocation available to the UE through the radio measurement channel (RMC) scheduling mode. Once the UE was verified to be attached, the emulator mode was changed to a user-defined transmission time interval (TTI) schedule, with the starting position and number of resource blocks defined by the configuration file. To provide reference points for post-processing time-alignment, every 10th uplink TTI was left

empty. An example of this lack of scheduling in the 10th TTI for a 1.4 MHz LTE UL channel is shown in Fig. 3, with a corresponding spectrogram of the acquired I/Q waveform given in Fig. 4. Downlink channel occupancy was always configured for a full allocation in the first TTI in each frame and no allocation in the rest of the frame, e.g., see Fig. 5 for a screenshot of the user interface specifying the DL allocation for a 1.4 MHz LTE channel.

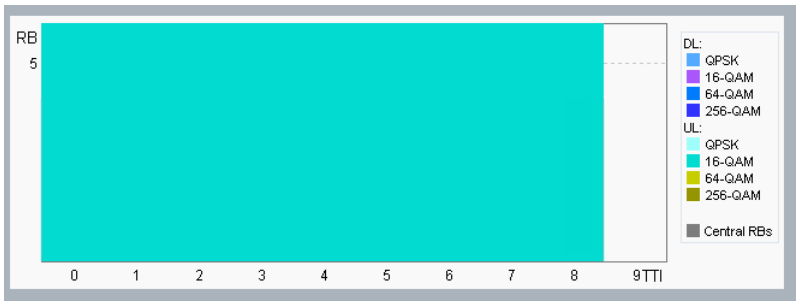

Fig. 3. Screen shot from the base station emulator user interface illustrating UL scheduling for a 1.4 MHz LTE channel.

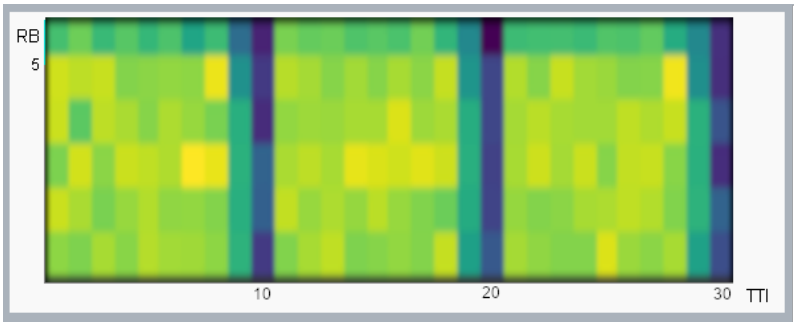

Fig. 4. Spectrogram of a 30 ms segment of recorded UL emissions for a 1.4 MHz LTE channel. Note that every 10th TTI is not allocated.

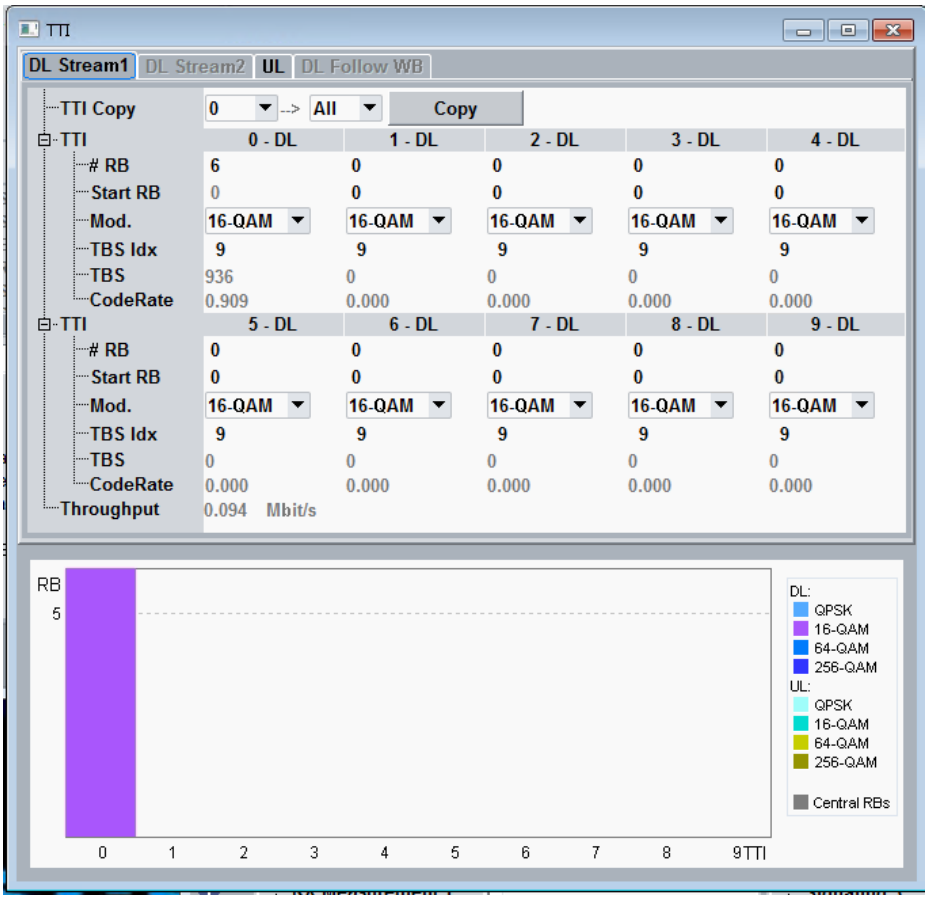

Fig. 5. Example DL configuration for 1.4 MHz LTE channel.

To ensure randomized data traffic on the RAN, the base station emulator was used in a mode of operation that allowed network traffic to transit to the application layer. An iPerf [5] server was created on the emulator at the application level that allowed the UE to push randomized data using an iPerf application on the client side; additional details are given in the next subsection.

2.3. UE Configuration

The goal of this testbed was recording UE UL transmissions. To ensure high-SNR recordings, the RF testbed was completely conducted, i.e., all connections were achieved with coaxial cables. We used a COTS UE that was previously modified by MITRE corporation staff to have a SubMiniature version A (SMA) cable connection for use in their multi-UE conducted testbed [6, Appendix F]. This coaxial connection was accomplished by disassembling the UE and disconnecting a miniature surface mount technology (SMT) connector, then replacing the back cover with a plastic shroud that housed the SMT to SMA connector. Isolation between the UE and the base station was accomplished by placing the UE in an JRE0709

isolation box that is specified to have 100 dB of isolation at 1 GHz and 95 dB of isolation at 3 GHz. [7]

Traffic volume from the UE was controlled by by the base station emulator. If the UE was not generating enough data traffic to completely fill the scheduled PRBs the emulator would command transmission of a padding sequence to fill the remainder. In order to avoid the repetitive data in the RF transmissions that was created by the padding sequence, an iPerf client was initiated on the UE that pushed data to the server on the emulator. The variable transmission rate of iPerf was controlled through the configuration file based on the size of the scheduled PRB allocation. The target data rate was chosen to fall just below the maximum throughput as calculated by the CMW based on the number of resource blocks, modulation, and code rate. For detailed description of the selected iPerf transmission rate for each scheduled condition see Table 2.

Table 2. List of Target Data Rates

Resource Blocks Allocated	Requested Data Rate (kbps)
1	130
2	259
3	389
4	518
5	648
6	778
8	1246
9	1390
10	1562
12	1879
15	2398
16	2513
18	2801
20	3146
24	3838
25	3953
27	4298
30	4817
32	5162
36	5580
40	6271
45	7193
48	7654
50	7884

2.4. VST Configuration

A VST was used to record I/Q data. The VST was physically synchronized to the other components of the testbed with a 10 MHz clock signal. Center frequency and bandwidth were configured to capture I/Q recordings for an LTE signal centered at 1770 MHz, with the sampling rate determined by the width of the configuration allocation.

The reference power was level set to -43 dBm. In order to avoid recording the local oscillator (LO) of the VST at the center frequency in the band of interest, an LO offset of -20 MHz was applied. The I/Q recordings were captured for a duration of 5 s and, due to the size of the data files, were streamed directly to a storage disk.

The acquisition rate of the VST was chosen to match the LTE channel bandwidth. These sampling rates are calculated using the number of FFT bins in a given channel and the subchannel spacing of 15kHz [2]. Details of the exact sampling rates can be found in Table 3.

Once acquisition was complete, the I/Q recordings were collated with the iPerf traffic logs by configuration number. Header information, and a description of the parameter can be found in Sec. 5.

Table 3. VST sampling rates for various bandwidth allocations

LTE Channel Bandwidth (MHz)	VST Acquisition Rate (MS/s)	FFT bins	Subcarrier Spacing (kHz)
1.4	1.92	128	15
3	3.84	256	15
5	7.68	512	15
10	15.36	1024	15

3. Testbed Automation

3.1. Overview

To capture UE UL I/Q recordings under the test conditions, a test-automation and test-network solution was developed. Testbed automation was required due to the length of the tests, complexity of individual system states, and broad test space that needed to be set and recorded. A high-level overview of the test sequence is given in Fig. 6. Test configurations were specified by a comma separated value (csv) file describing the different system states. The test sequence iterates through all test cases in the defined order read from the csv file input to the testbed. A list of all test configurations is shown in Table 4. In each test case, a configuration was applied to the hardware in the test setup, and data were acquired and recorded. A single test is defined as a single iteration of the main test loop, or the execution of the test sequence from the start of loading of configuration files to the decision point in determining if all cases have finished. Outputs from the test system included traffic logs from the UE, and the I/Q files recorded as the output of the testbed. Example visualizations of the data captures can be found in Sec. 6.

Table 4. Test configurations for the experiment. The start resource block is conditional on the number of resource blocks.

Channel Allocation (MHz)	Modulation Order	Start Resource Block	Number of Resource Blocks
1.4	QPSK, 16QAM	0, 1, 2, 3, 4, 5	1, 2, 3, 4, 5 ,6
3	QPSK, 16QAM	0, 2, 4, 6, 8, 10, 12, 14	1, 2, 3, 4, 5, 6, 8, 9, 10, 12, 15
5	QPSK, 16QAM	0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2, 3, 4, 5, 6, 8, 9, 10, 12, 15, 16, 18, 20, 24, 25
10	QPSK, 16QAM	0, 3, 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48	1, 2, 3, 4, 5, 6, 8, 9, 10, 12, 15, 16, 18, 20, 24, 25, 27, 30, 32, 36, 40, 45, 48, 50

To physically connect the equipment to the test-automation host, the testbed was networked using either universal serial bus (USB) or wired local area network (LAN) connections. Specifically, the base station emulator and VST system were connected to the host using LAN connections while the UE was connected via a USB interface. Communications with the emulator occured through virtual instrument software architecture (VISA) using standard commands for programmable instruments (SCPI). Configuration parameters and acquisition commands were communicated to the VST via a hypertext transfer

protocol (HTTP) interface. The UE was interfaced with the Control PC via android debug bridge (ADB) software which opens a remote shell on the device.

In addition to the communications connections, a 10 MHz clock signal originating from the base station emulator was routed to the VST to ensure synchronization of the LTE communications between the UE and base station emulator with the I/Q captures. Though the clocks were physically synchronized, fine-scale time-alignment was performed in a software post-processing described in Sec. 4. The testbed network is shown in Fig. 1. A combination of Python and National Instruments (NI) LabVIEW were used in order to automate various components within the testbed.

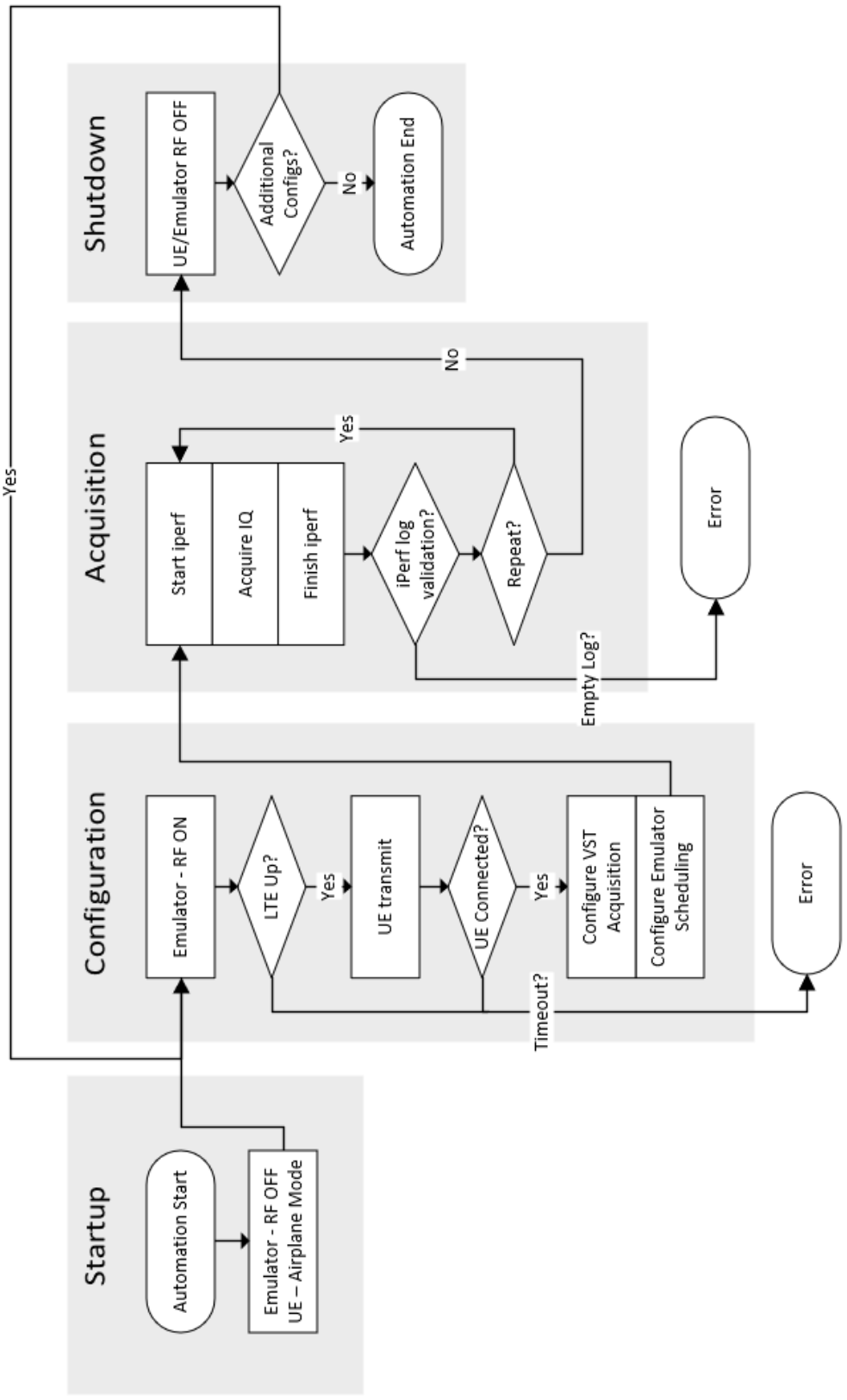

Fig. 6. High-level test sequence specification.

3.2. Python Automation

Configuration and control of the base station emulator and UE link was performed using Python version 3.7.0. The base station emulator and UE were initially placed into non-transmitting modes. From there the base station LTE cell was initiated in an RMC mode, with the automation polling the emulator to signal that the cell was stable. Once stable, the UE was moved from airplane mode to a transmitting mode and the automation again polled the emulator to signal if a connection has been made with the UE. With a confirmed link between UE and base station emulator, the emulator was then configured in a user defined TTI mode using the start PRB and the total number of PRB from the csv configuration file.

Next, the UE was automated via a terminal connection made through ADB software facilitated by a USB connection to the device. Once within the UE terminal, an application designed for generating and sending internet protocol (IP) traffic was initiated with various command line interface (CLI) arguments to create LTE traffic that was routed to a server on the control computer via the test-network. The iPerf software was used with the user datagram protocol (UDP) argument to ensure that the traffic created was UDP.

3.3. LabVIEW Automation

In addition to Python automated components, LabVIEW was used to automate the acquisition of I/Q signals of the uplink traffic. The main testbed software interfaced with the VST through a piece of executable software written in LabVIEW that included a TCP/IP server. This allowed the VST to have the pertinent settings including acquisition rate, filename and location to follow each configuration of the base station and UE pair configuration.

The control software was in the form of a typical producer-consumer structure, implying parallel loops that either produce or consume data at different rates, and passing through multiple states as the software executed. For instance, on software startup, or change of input parameters, the VST passed through a state that configured the RF hardware so that, upon issuing an acquire command, the hardware was immediately available.

The hardware was configured to acquire without receiving a trigger signal due to the timing uncertainties of creating a link between the UE and the base station emulator. Due to the lack of synchronization in the acquisition with the start transmission from the UE, time alignment of the signals after acquisition was required.

4. Catpure Postprocessing

4.1. Sample Clock Alignment

We leveraged the cyclic structure in the baseband modulation waveforms to verify that the VST sampling clock was close to the sampling clock of the UE transmitter. This step was necessary because we did not receive verification from the VST itself that its LO phase and sample clock were synchronized with the UE. In contrast, we know that synchronization is achieved between the UE and the communication tester, because it is a prerequisite to the successful communication between the devices.

We performed this testing based on cyclostationary waveform characteristics laid out in cellular physical layer specifications [2]. These provide the expected sample rates and time-indices of cyclic prefix (CP) on the baseband waveforms transmitted by the UE. Mismatch between the UE and VST sample clocks will lead to slow changes between the time-indices of the peak waveform correlation computed with the CP index locations and time-delayed CP indices. The difference in UE and VST sample clock rates can be estimated by comparing locations of the correlation peaks in radio frames near the beginning of a recording against those near the end.

Across the data set, the cyclostationary method showed that differences between the UE and VST sample clock rates were negligible. Namely, the locations of the correlation peaks at the beginning and end of the waveforms were less than one time bin index.

The result of this validation confirmed that the combination of frequency synchronization and time-base accuracy successfully resulted in synchronized sample clocks in the UE and VST.

4.2. TTI Alignment

The post-processing alignment of each waveform recording to the boundary of a TTI was facilitated by the decision to turn off scheduling of emulated emissions during the 10th TTI of each LTE frame; see Sec. 2.2. This choice minimized the RF energy present in a periodic 1ms time segment, which produced a large power differential at the boundary of the 10th TTI of one frame and the 1st TTI of the following frame, referred to as t_0.

The dynamic programming break-point estimation algorithm from the Ruptures Python library [8] was leveraged to find t_0. The TTI alignment began by fitting the break-point estimator to the first 10 ms segment of the I/Q recording, which was down-sampled by a factor of 10 to accelerate waveform processing. The break-point estimator yielded a pair of breakpoints, (B_1, B_2), with a minimum separation distance set to the number of samples in 1 ms. If the returned distance between breakpoints was greater than 1.1 ms the estimator was reapplied to the next 10 ms section of the recording. The waveform segment between the two breakpoints was taken to be the 10th TTI, which was void of RF emissions. Using the empty TTI, the maximum power of the recorded background noise was calculated.

The TTI alignment offset, t_0, was taken to be the first time step with power greater than the maximum background noise power in a window of 200 time steps centered around the second power break point, B_2. This second alignment step allowed for a more fine-grained precise estimation of the beginning of the first active TTI.

5. Dataset Description

The datasets presented here were captured in technical data management system (TDMS) format with header information that can be found in Table 5. The data is subdivided into archives by allocation size with the file structure seen in Fig. 7. Within each archive each I/Q capture is stored as a TDMS file. The individual captures are named as config_*configuration_timestamp*. In the top level directory a csv file with the various input settings to the configuration can be used to find a configuration of interest. The Python code in Listing 1 may be used to open the TDMS file and return the I/Q samples in physical units. Prior to running this code the appropriate dependencies must be installed using the commands *pip install numpy*[9] and *pip install nptdms*, where nptdms[10] is a package for reading and writing TDMS files. The output dictionary in this listing returns relevant metadata about the I/Q data along with the I/Q data itself. The dataset resulting from this effort is publicly available at https://doi.org/10.18434/mds2-3177.

Table 5. Data File Header Contents

Header Name	Type	Description
carrier_frequency	Double	The center frequency of the IQ acquisition
IQ_samples_per_second	Double	The sampling rate of the VST
reference_level_dBm	Double	The reference level that full scale of the ADC corresponds to, in dBm
number_of_seconds_to_acquire	Double	Total acquisition time if the software is to run in a finite mode.
lo_offset	Double	The frequency offset applied to the LO, typically used to move the LO outside the frequency range of interest

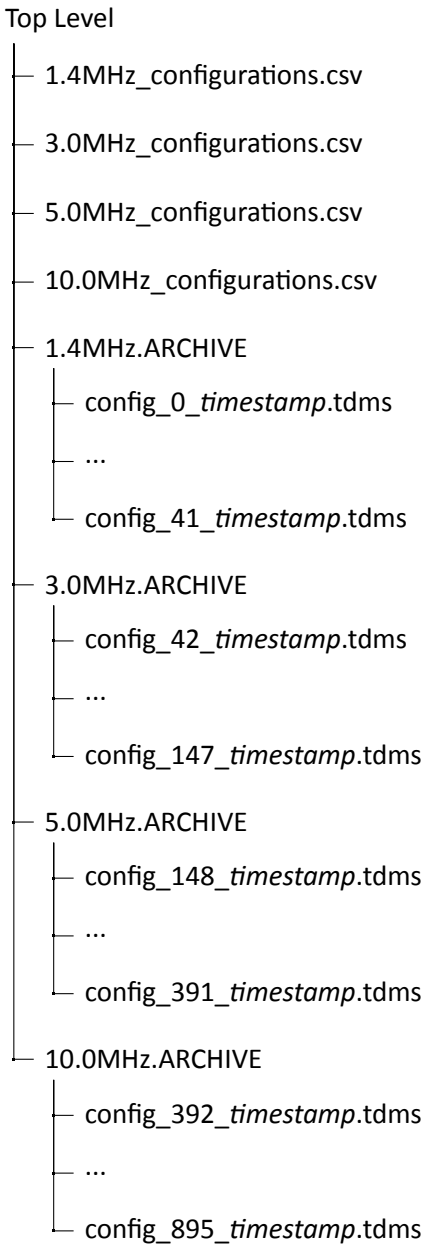

Fig. 7. Archive File Structure

```
from nptdms import TdmsFile
import numpy as np

def Tdms_to_Iq(file_path,impedance = 50, bitdepth = 2**16):
    output = {}
    tdms_file = TdmsFile(file_path)
    header = tdms_file['Header Data'].as_dataframe()
    reference_level_dBm = header["reference_level_dBm"][0]

    #convert ref level to a voltage
    reference_level_v = np.sqrt(10**(reference_level_dBm/10)*impedance)

    #determine the voltage step per bit
    voltage_step_size = reference_level_v/bitdepth

    #determine the time step per acquisition
    time_step = 1/header["IQ_samples_per_second"][0]
    iq_df = tdms_file['IQ Signal'].as_dataframe()

    #convert IQ data into complex
    iq = voltage_step_size*iq_df['I Data']+1j*voltage_step_size*iq_df['Q Data']
    output = {'bitdepth':bitdepth,
              'reference_level_v':reference_level_v,
              'time_step':time_step,
              'impedance':impedance,
              'iq':iq}
    return output
```

Listing 1. TDMS import function

6. Example Data Captures

To illustrate the character of the recordings described here, Figs. 8 through 12 contain example spectrograms for different configurations of the 1.4 MHz LTE channel bandwidth. Within these configurations are various numbers of populated resource blocks along with QPSK, and 16-QAM data presented. The additional LTE channel bandwidths, 3 MHz, 5 MHz, and 10 MHz, that were recorded are not shown here. The figures below show the maximum power as a function of frequency in the top subfigure, a time series of the mean power in the right subfigure, and in the central subfigure power in gray scale as a function of frequency and time in x, and y respectively with white being the highest power.

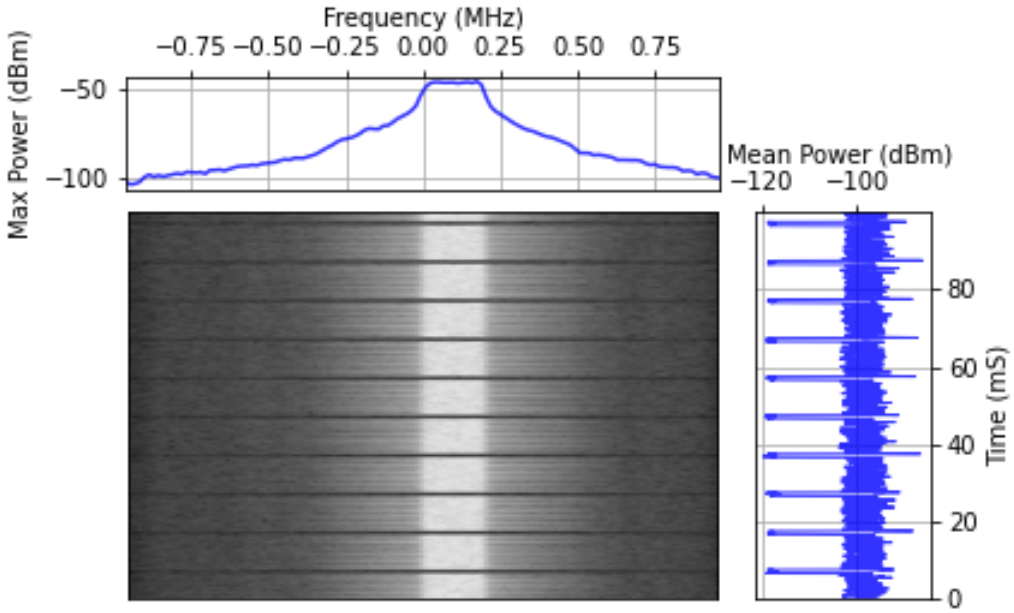

Fig. 8. Number of RB: 1, 1.4 MHz, QPSK Configuration: 15

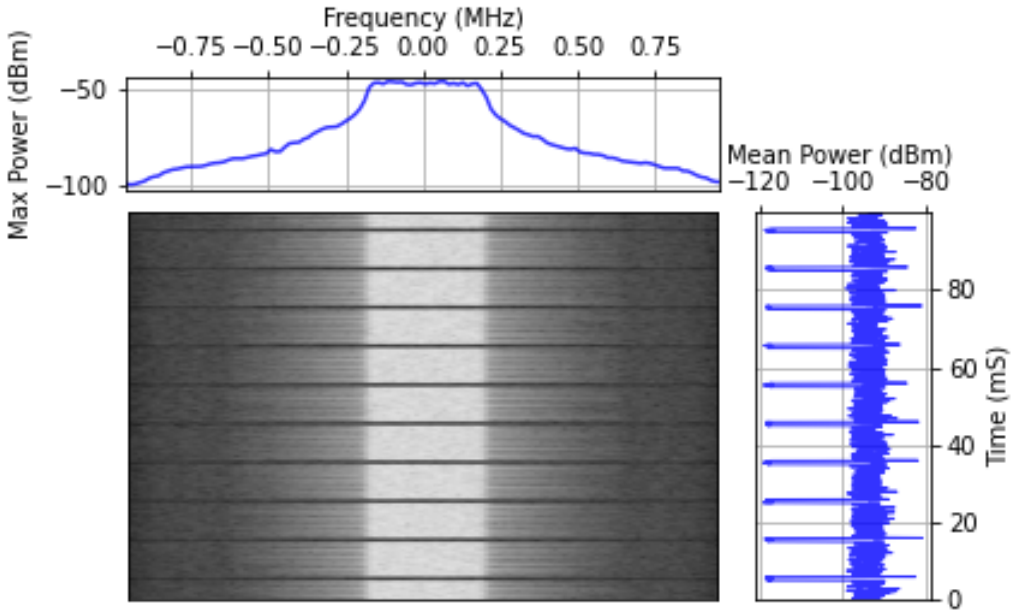

Fig. 9. Number of RB: 2, 1.4 MHz, QPSK
Configuration: 12

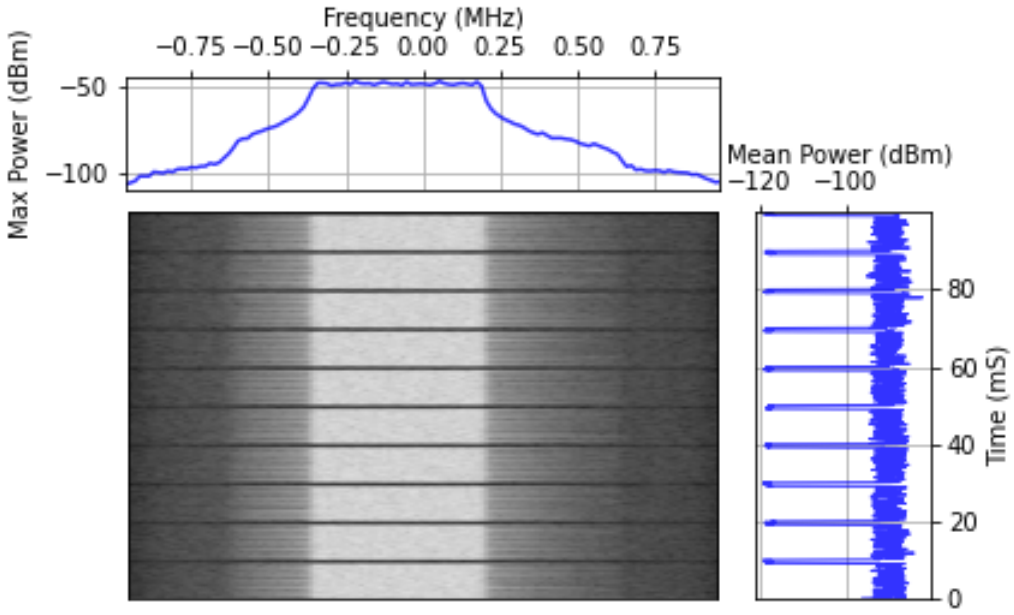

Fig. 10. Number of RB: 3, 1.4 MHz, QPSK
Configuration: 8

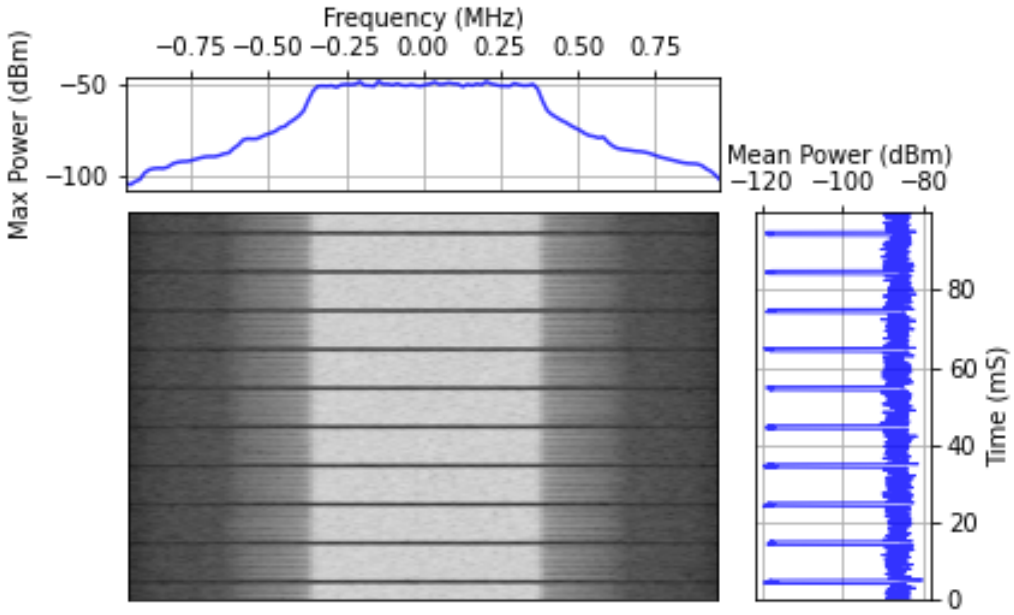

Fig. 11. Number of RB: 4, 1.4 MHz, QPSK Configuration: 9

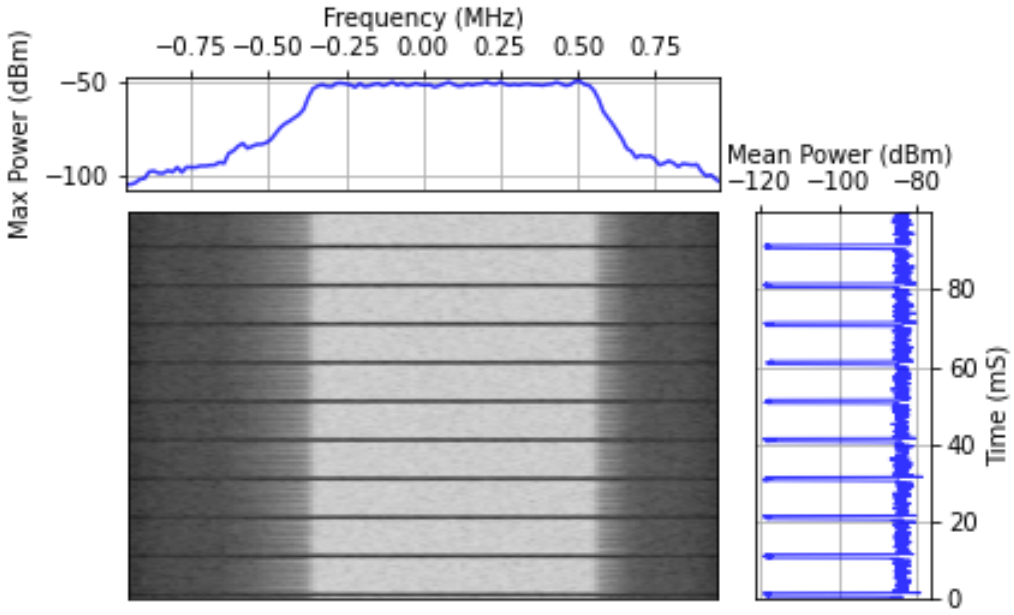

Fig. 12. 1.4 MHz, QPSK, Number of RB: 5, Configuration: 10

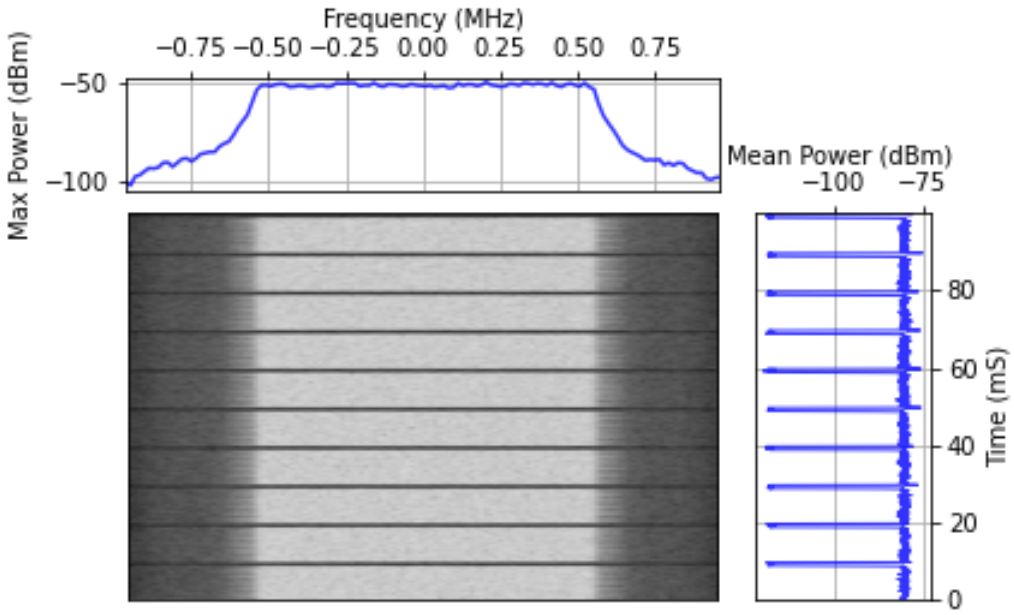

Fig. 13. 1.4 MHz, QPSK, Number of RB: 6
Configuration: 5

7. Summary

This NIST technical note presented collection methods for high-SNR recordings of LTE UL emissions with different configurations spanning various resource block allocations. This data set was intended to include the complexity of a COTS UE while controlling for other factors present in a real LTE link, such as resource scheduling and an imperfect RF channel.

The collection methods and data set described here can be viewed as intermediate in realism between a fully-synthetic, computer generated dataset and a laboratory captured dataset with a commercial evolved node B (eNB). Sec. 2 described the testbed set up with the various RF and data connections. Sec. 5 highlighted the format in which the data was collected and stored and how to load it for further use with Python. Finally, Sec. 6 provided example spectrograms for recordings. The dataset resulting from this effort is publicly available at https://doi.org/10.18434/mds2-3177.

References

[1] National Telecommunications and Information Administration (2022) AWS-3 Transition. Available at https://www.ntia.doc.gov/category/aws-3-transition.

[2] Third-Generation Partnership Project (3GPP) (2021) TS 36.211, V16.5.0, Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Channels and Modulation. Available at www.3gpp.org.

[3] Sanders A, Forsyth K, Horansky R, Kord A, McGillivray D (2021) Laboratory Method for Recording AWS-3 LTE Waveforms (National Institute of Standards and Technology), 2159. Available at https://doi.org/10.6028/NIST.TN.2159.

[4] Nelson E, McGillivray D (2021) In-Situ Captures of AWS-1 LTE for Aeronautical Mobile Telemetry System Evaluation (National Telecommunications and Information Administration), 21-553. Available at https://its.ntia.gov/publications/details.aspx?pub=3262.

[5] Dugan J, Elliott S, Mah BA, Poskanzer J, Prabhu K iPerf - The ultimate speed test tool for TCP, UDP and SCTP. Available at https://iperf.fr/.

[6] Coder J, Wunderlich A, Frey M, Blanchard P, Kuester D, Kord A, Lees M, Sanders A, et al. (2019) Characterizing LTE User Equipment Emissions: Factor Screening (National Institute of Standards and Tehnology), 2069. Available at https://doi.org/10.6028/NIST.TN.2069.

[7] JRE-Test, LLC JRE 0709-P RF Shielded Test Enclosure. Available at https://jretest.com/wp-content/uploads/2020/12/JRE0709-P.pdf.

[8] Truong C, Oudre L, Vayatis N (2020) Selective review of offline change point detection methods. *Signal Processing* 167:107299.

[9] Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, Wieser E, Taylor J, Berg S, Smith NJ, Kern R, Picus M, Hoyer S, van Kerkwijk MH, Brett M, Haldane A, del Río JF, Wiebe M, Peterson P, Gérard-Marchant P, Sheppard K, Reddy T, Weckesser W, Abbasi H, Gohlke C, Oliphant TE (2020) Array programming with NumPy. *Nature* 585(7825):357–362. https://doi.org/10.1038/s41586-020-2649-2. Available at https://doi.org/10.1038/s41586-020-2649-2

[10] Reeve A (2024) nptdms, https://github.com/adamreeve/npTDMS.