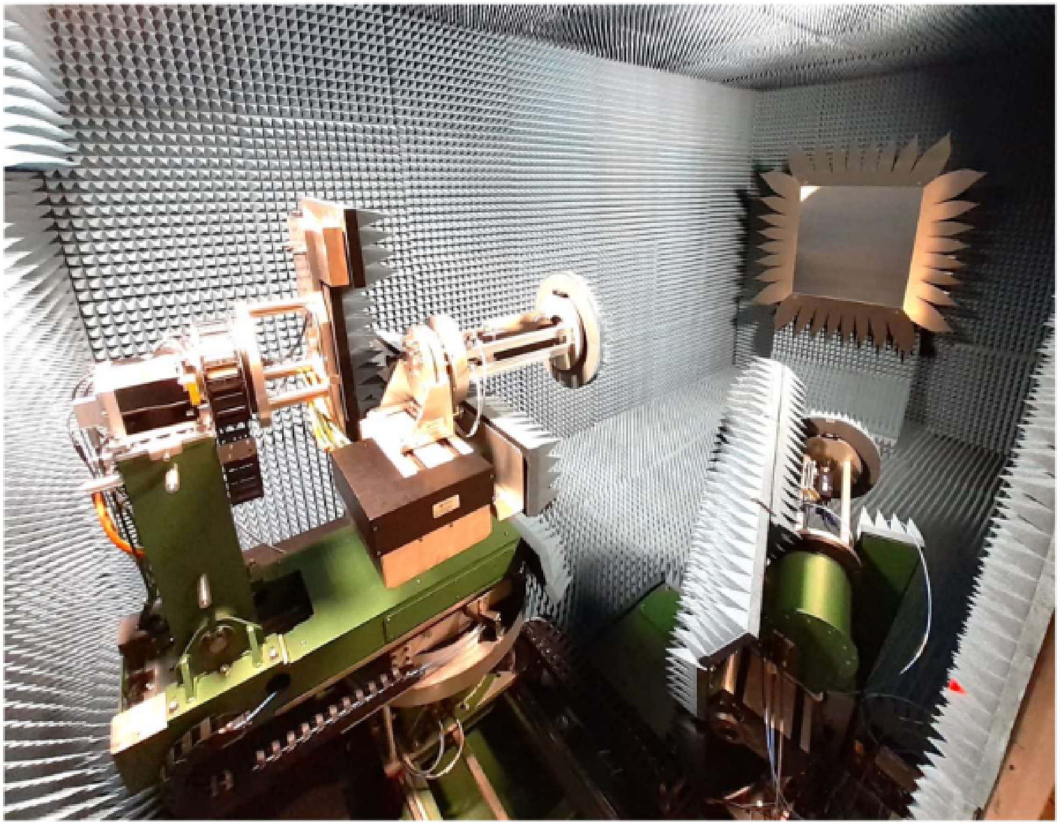


Advancing 5G Calibration Methods In The Compact Antenna Test Range

A single antenna approach for over the air testing from 18 GHz to 110 GHz

Degree project report in Electrical Engineering

Yuehong Zhou
Christian Daniel

Degree project report 2024

Advancing 5G Calibration Methods In The Compact Antenna Test Range

A single antenna approach for over the air testing from 18 GHz to 110 GHz

Yuehong Zhou
Christian Daniel

Department of Electrical Engineering
Chalmers University of Technology
& LTH Faculty of Engineering
Sweden 2024

Advancing 5G Calibration Methods In The Compact Antenna Test Range
A single antenna approach for over the air testing from 18 GHz to 110 GHz
Yuehong Zhou
Christian Daniel

Supervisors:
Peter Herrede (Radio Specialist at Ericsson) &
Aleksei Fedorov (Department of Electro and Information Technology of LTH)
Examiners:
Josip Vukusic (Department of Microtechnology and Nanoscience) &
Michael Lentmaier (Department of Electro and Information Technology of LTH)

BSc Thesis 2024

Department of Electrical Engineering
Chalmers University of Technology
SE-412 96 Gothenburg Sweden
Telephone +46 31 772 1000

Department of EIT
Lund University, LTH
Box 118 SE-221 00 Lund Sweden
Telephone +46 46 222 00 00

Cover: The Compact Antenna Test Range Chamber

Typeset in LaTeX
Gothenburg & Lund, Sweden 2024

Advancing 5G Calibration Methods In The Compact Antenna Test Range
A single antenna approach for over the air testing from 18 GHz to 110 GHz

Yuehong Zhou
Department of Electrical Engineering
Chalmers University of Technology

Christian Daniel
Department of Electrical and Information Technology
Lunds University

Abstract

This project presents a study on the calibration methodologies for OTA testing in CATR chamber, focusing on the frequency span of 18 GHz to 110 GHz. We introduce a single-antenna method, proving it more efficient and accurate than traditional multi-antenna setups. The research also delves into the challenges of high-frequency signal testing in 5G and upcoming 6G communications. These challenges include signal propagation issues and the complexity of signal processing techniques. Through a two-phase approach, our research validates the new calibration method and assesses the signal generation and measurement system's performance.

Keywords: OTA, over the air, CATR, 5GNR, FR2

Acknowledgements

First and foremost, a big thanks to our supervisor Peter Herrder at Ericsson for the opportunity, guidance, and insights. We are also grateful to the IV & C team at Ericsson Lund, for their help and support. Shoutout to the Keysight tech support for their assistance and expertise.
Special thanks to Aleksei Fedorov for his precise and adamant guidance on the report. And a thank you to Michael Lentmaier for accepting this task.
We would also like to thank our supervisor and examiner, Josip Vukusic, for his guidance and quick feedback.
Lastly, we would like to thank all our family and friends, for their understanding and encouragement during not only this project but also all those long nights.

Yuehong Zhou, Gothenburg, April 2024
Christian Daniel, Lund, April 2024

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

CATR	Compact Antenna Test Range
CW	Continuous Wave
EM	Electromagnetic
dB	decibels
DANL	Displayed Average Noise Level
DUT	Device Under Test
FSPL	free-space path loss
FR	Frequency Range
IF	Intermediate frequency
I/Q	In-phase and quadrature
LO	Local oscillator
mm-Wave	Millimeter Wave
NR	New Radio
OTA	Over the air
RF	Radio frequency
RBW	Resolution bandwidth
SNR	Signal to noise ratio

Contents

List of Figures

List of Tables

1

Introduction

1.1 Background

As mobile internet traffic surges with high-definition videos, podcasts, downloads, and uploads, mobile communication technology must advance to handle the growing demand for high data speed and therefore large bandwidth. One critical component in this advancement is the Over-The-Air (OTA) antenna, which can typically be seen outdoors on a roof or a mast. To achieve these faster speeds, 5G and 6G networks utilize frequencies up to 71 GHz and higher [1], for which the bandwidth can reach 2 GHz. This places a high demand on the whole system for accuracy and even higher for testing equipment of such a system.

5G utilizes two frequency ranges (FR): low bands (450 MHz to 7 GHz) and high bands (24 GHz to 71 GHz) [1]. Most existing 5G technologies operate in the lower bands because they offer a greater coverage area. However, higher frequency bands offer the potential for significantly faster data transfer speeds and greater capacity, pushing wireless communication technologies beyond the capabilities of the current 5G network [2, 3]. While the specific frequencies for next-generation communication 6G are still under development, they are expected to extend well beyond the current 5G range, potentially reaching as high as 300 GHz [4].

The OTA characterization are subjected to multiple tests to fully adapt to the wireless performance. Those who have requirements for this performance test are the manufacturers of common electronic products such as cell phones and smart wearable devices. Testing devices in OTA environments at higher frequencies involves complex setups with specific antennas for each frequency range, making the process both challenging and time-consuming. That adds to the complexity of the development. Ericsson, an industry leader, currently utilizes five different OTA test setups in the compact antenna test range (CATR) chamber for covering the frequency range of 24 GHz to 71 GHz.

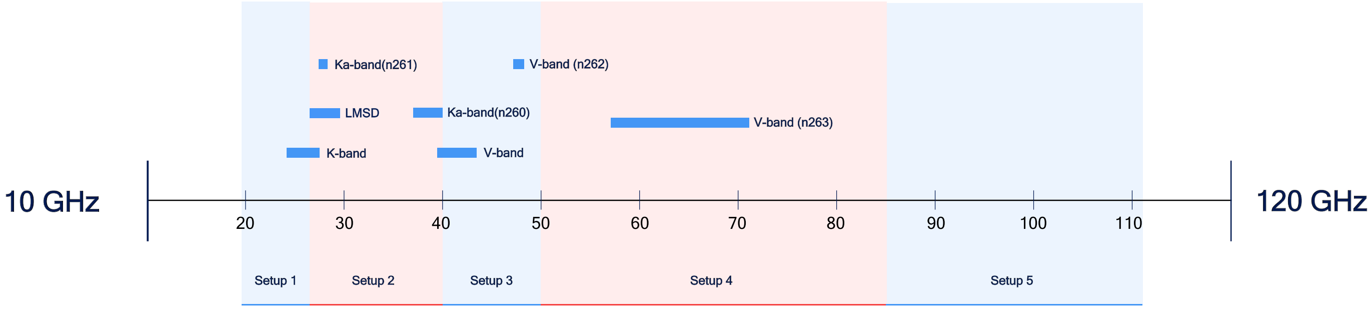


Figure 1.1: A chart that shows on top the different FR2 Bands and at the bottom the five different antenna setups in the CATR in their respective frequency range.[5]

Figure 1.1 illustrates how these five setups are distributed in the 18 GHz to 110 GHz interval. Each setup requires a specific antenna, cable, and device positioning. This segmentation emphasizes the critical need for precision and adaptability, as each new setup is meticulously calibrated to ensure high accuracy and performance. In Figure 1.1 frequency bands, represented in the blue bar, within the FR2 [1] are located within different antenna setups. These overlapping antenna setups give rise to even more human errors together with the synchronization of the calibrated systems.

1.2 Purpose and aim

The focus of this project is to utilize a new method that uses a single antenna and analyzes it by using signal-to-noise ratio among other factors. The primary objective of this project is to evaluate cutting-edge technology from Keysight for conducting frequency sweeps within the CATR chamber, covering the frequency range of 18 GHz to 110 GHz using a single antenna.

This approach eliminates the need to swap between five antennas and re-calibrate the system repeatedly (unlike the traditional method), significantly reducing measurement time. By streamlining complex operations, human error is minimized, thus improving the efficiency and accuracy of the test.

1.3 Clarification of the questions

This study aims to address the following question:

- How much does the new system differ from the normal 5-antenna approach?
- Investigate the advantages that this new way of measuring can bring.
- What is the dynamic range of the system?

1.4 Limitations

This project centers on analyzing measurements captured within the CATR chamber. While the specifics of the testing equipment will not be covered here, it is

important to acknowledge that some instruments are on temporary loan and have limited availability. Similarly, access to the CATR chamber itself is restricted to designated time frames.
Our focus remains on the measurement data obtained from the CATR. We will not go into the properties of 5G and 6G signal processing in this project.

2

Theory

This chapter aims to build the theoretical foundation of this project and presents basic information on the equipment used. It provides background knowledge about OTA testing in CATR and information to understand the content and purpose of the project.

2.1 5G New Radio

5G New Radio (NR) refers to the new radio access technology in 5G mobile communication networks, a global 5G standard. It aims to deliver higher data rates, lower latency, and a wider range of connectivity requirements. Different communication networks utilize different frequency bands to meet their specific service requirements and technical standards. 5G NR is unique in covering two different frequency ranges: FR1 and FR2. FR1 covers frequency bands below 6 GHz and is suitable for wide coverage and penetration, while FR2 uses mm-Wave bands above 24 GHz.[1] Ericsson in Lund is particularly focused on testing the capabilities within the FR2 range. This thesis concentrates on the mm-Wave frequency ranges, reflecting the advanced potential for wireless communication enabled by FR2.

Band	f (GHz)	Common name	Uplink / Downlink (GHz)	Channel bandwidths (MHz)
n257	28	LMDS	26.50 – 29.50	50, 100, 200, 400
n258	26	K-band	24.25 – 27.50	50, 100, 200, 400
n259	41	V-band	39.50 – 43.50	50, 100, 200, 400
n260	39	Ka-band	37.00 – 40.00	50, 100, 200, 400
n261	28	Ka-band	27.50 – 28.35	50, 100, 200, 400
n262	47	V-band	47.20 – 48.20	50, 100, 200, 400
n263	60	V-band	57.00 – 71.00	100, 400, 800, 1600, 2000

Table 2.1: FR 2 and their characteristics [1]

Table 2.1 summarizes the currently defined FR2 bands, their common names, the corresponding frequency range for uplink and downlink, and the available channel bandwidth. It indicates a broad frequency spectrum from 24.25 GHz to 71 GHz. That opens up new possibilities in wireless communication for high bandwidth and large data rates.

2.1.1 mm-Wave

The mm-Wave technology uses electromagnetic (EM) waves with wavelengths ranging from 1 to 10 millimeters, covering a frequency range of 30 GHz to 300 GHz.

The introduction of this band provides 5G networks with enormous bandwidth and extremely high data transmission speeds. However, mm-Wave also faces significant propagation challenges, notably severe attenuation in the atmosphere, limiting their transmission range [6].

A key limiting factor for mm-Wave radio frequency (RF) communications is the free space path loss (FSPL) for direct line-of-sight communication between two antennas [7]. The FSPL can be expressed by the following equation:

$$FSPL = \frac{P_t}{P_r} = \left(\frac{4\pi df}{c}\right)^2 \tag{2.1}$$

The FSPL represents the weakening of signal strength as it travels through free space. It is calculated as the ratio of transmitted power (P_t) to received power (P_r). This equation includes factors like distance d, carrier frequency f, and the speed of light c. The relationship between wavelength λ and frequency is given by $\lambda = \frac{c}{f}$. As the equations show, higher carrier frequencies have shorter wavelengths but experience greater attenuation. In simpler terms, this means mm-Wave signals weaken much faster over distance compared to conventional communication frequencies.

2.2 Antennas

2.2.1 Radiation pattern

Antennas radiate from the current of charges that run through them. A current produces a magnetic field [8]. A time-varying electric or magnetic field produces the corresponding magnetic or electric field respectively and forms an electromagnetic wave. The wave that propagates has its two field components orthogonally oriented and travels in the direction of the normal to the plane defined by the electromagnetic field vectors. By designing the path of the current, different radiation patterns are created. This radiation does not solely emanate from antennas, it happens in all circuits to a varying degree. Sometimes with undesirable effects, giving off emissions or having low immunity to interference from other equipment (also known as electromagnetic coupling) [9].

The most basic antenna is the dipole antenna. It is described as having isotropic radiation due to its symmetric design. However, in practice, isotropic antennas are purely theoretical [9]. The waves emitted are polarized since the charges only propagate along the rods length. To receive the transmitted EM wave with an antenna of the same design, polarization needs to be considered.

2.2.2 Gain

The gain describes how well the antenna converts input power into radio waves transmitted or received in or from a specified direction [8]. When no directions are mentioned the gain refers to the peak value of the gain. Thus it is in the direction of the main lobe. Plotting the gain as a function of direction gives us the antenna pattern. Figure 2.1 shows the radiation pattern of a dipole antenna. Due to the absence of radiation in the direction of the Z-axis the dipole antenna is not isotropic.

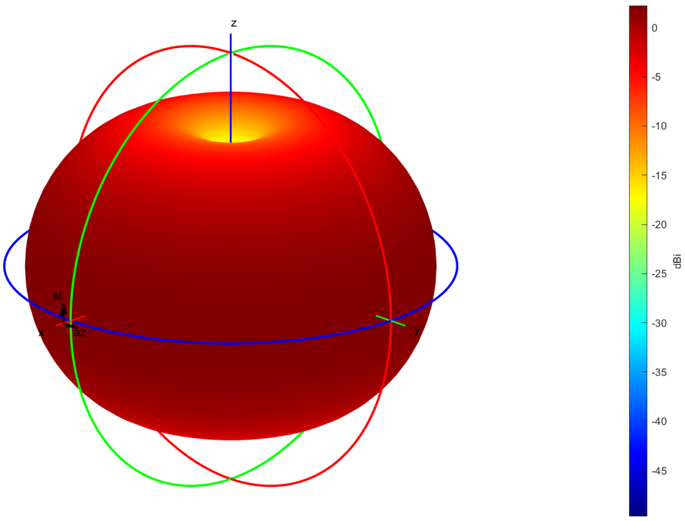


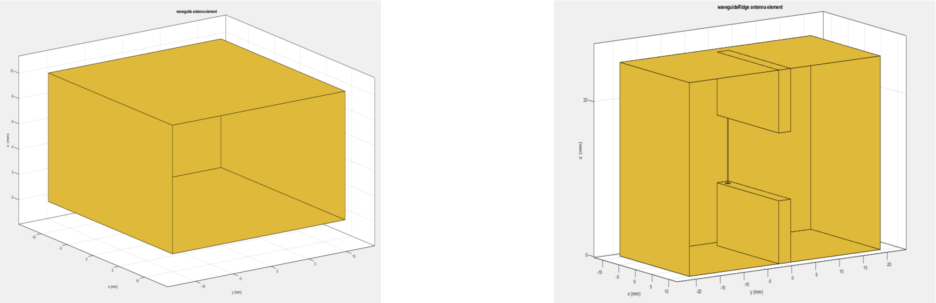
Figure 2.1: Radiation pattern from a dipole antenna, oriented on the Z axis, i.e current is running along the z-axis. Produced with Matlab Antenna Designer.

2.2.3 Polarization

Depending on the path of the current in the antenna. Different polarization occurs for each antenna, ranging from circular to horizontal or vertically polarized antennas. Circular polarizations exhibit reduced sensitivity to the orientation of the transmitting or receiving antenna. The dipole antenna is linearly polarized (due to its linearized design), in the same orientation as the rod of the dipole.

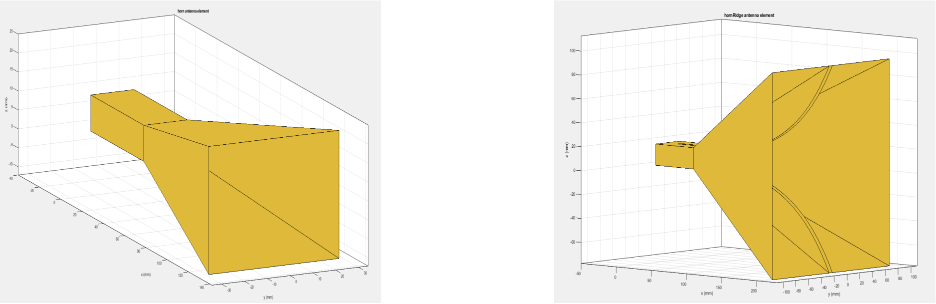
2.2.4 Horn Antenna

Starting with a waveguide shown in Figure 2.2a, Figure 2.2b shows a double ridged waveguide. The ridge horn antenna is to achieve wide bandwidth, high gain, and good impedance matching characteristics through its special ridge structure. Thereby providing excellent performance in various complex environments. When the waveguide is flared a horn antenna is created as in Figure 2.2c. Ridge horn antennas represented in Figures 2.2d have a wide bandwidth making them suitable for this test [9]. All of the antennas shown in the Figure 2.2 are linear polarization.



(a) Waveguide

(b) Waveguide with double ridges



(c) Flared horn

(d) Ridged flared horn

Figure 2.2: Pictures of different antennae produced with Matlab Antenna Designer

2.2.5 DRH110 horn antenna

The DRH110 model from RFspin, used for the project, is a horn antenna with an operational frequency range from 14 GHz to 110 GHz [10]. Figure 2.3 shows a typical pattern from a horn antenna. Comparing it to the Figures 2.1 it is clear that horn antennas are directional antennas. The front lobe dominates the radiation pattern in an intense red color that can be seen on the positive x-axis, and the back lobe on the negative side of the x-axis with a lower radiation intensity. The radiation intensity decreases significantly along the YZ-plane. The horn antenna has a focused beam primarily along the x-axis and reduced radiation in the perpendicular direction.

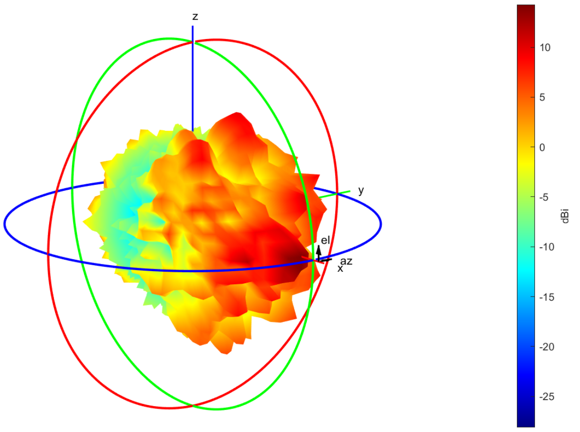


Figure 2.3: Double Ridged Horn pattern oriented on the Z axis. Obtained with Matlab Antenna Designer.

2.3 OTA test

The OTA is the propagation of information through EM waves in the air. The OTA testing is a method to validate the performance of wireless devices. The core of testing focuses on the receiving and transmitting capabilities of the antenna.

2.3.1 Near- and far-field

A key concept in OTA testing is distinguishing between the signal's near-field and far-field regions.
The interaction between electric and magnetic fields is notably complex in the antenna's radiating near-field region. It is due to significant interactions with adjacent electronic components. This complexity decreases with distance from the antenna, where radiation effects intensify and the radiation pattern evolves towards its stable far-field configuration [11].
In the far-field region, the EM wave propagates approximately as a plane wave. This propagation region is optimal for antenna performance testing, given that wireless communications typically span large distances. The transition to the far field is defined when the distance from the antenna exceeds $\frac{2D^2}{\lambda}$, with D representing the antenna's maximum dimension and λ the wavelength [8].
This far-field region ensures a homogeneous test environment for the measurements. This uniformity guarantees that measurements accurately reflect the antenna's performance in a standardized and repeatable manner.

2.3.2 CATR chamber

A CATR chamber is an approach to the OTA test, where it simulates a far-field scenario in a limited space. By using a parabolic reflector, it transforms the spherical waves emitted from the signal gain horn antenna into plane waves. This setup allows for precise far-field testing in a physically confined environment, ensuring that devices under test perform as expected in real-world conditions. Figure 2.4 illustrates the basic configuration in the CATR chamber.

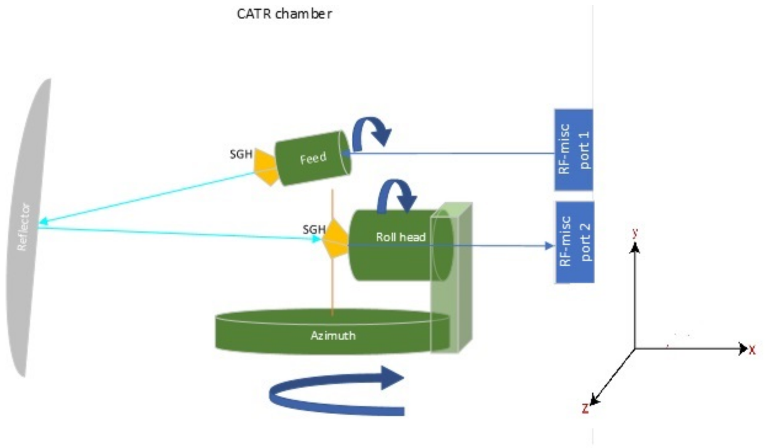


Figure 2.4: Illustration of the CATR chamber by Peter Herder.

The device under test (DUT) is fixed on the roll head that can roll 360 degrees. The roll head is mounted on the azimuth, which rotates around the Y-axis for plus and minus 200 degrees. It can move a few centimeters forward and backward. The feed is the same as the roll it can be rolled 360 degrees and can be moved forward and backward.
The antenna transmits a wavefront that travels towards the reflector antenna. This reflector converts the spherical wave into a plane wave, feeding it to the receiving antenna. This process works in both directions, allowing the device under test to receive a plane wave transmitted through the reflector from the feed antenna.

2.4 Instruments

This section details the advanced instruments utilized to test and measure the system's performance, highlighting their capabilities and applications.

2.4.1 Keysight signal analyzer and V3050A frequency extender

The Keysight N9042B UXA signal analyzer is a high-performance device that tests the real performance of mm-Wave innovations in 5G. It has a frequency range of 2Hz to 50GHz. Connected to the V3050A frequency extender, the analyzer extends its capability to assess signal power across frequencies up to 110GHz [12]. The configuration of the connection between the analyzer and the extender is illustrated in Figure 2.5.

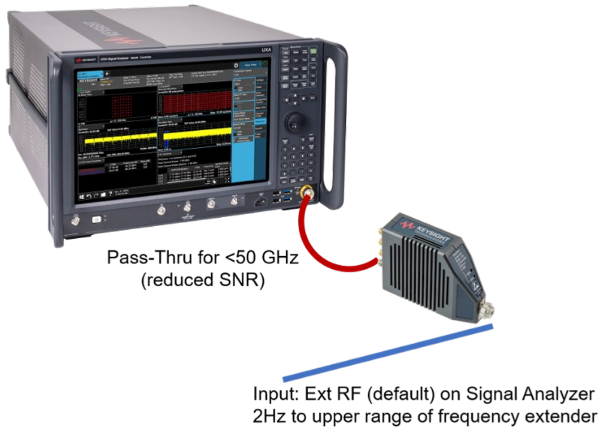


Figure 2.5: Configuration UXA (on the left) and V3050A (to the right). [12]

UXA is on the left side in Figure 2.5, the red line is a Pass-Thru cable that connects to the V3050A extender. The signals below 50GHz are sent directly to the UXA. For frequencies above 50GHz, the signals are handled by the V3050A, and then passed to the UXA for further processing. In addition to the Pass-Thru cable the V3050A has two other ports, intermediate frequency (IF) out and local oscillator (LO) in, these will be connected to the IF in and LO out on the signal analyzer. This setup enables the handling of frequencies above 50 GHz. A USB-C to C cable acts as a power delivery for the extender and a data transmitter, connecting the two devices.

2.4.2 Keysight signal generator and frequency extender

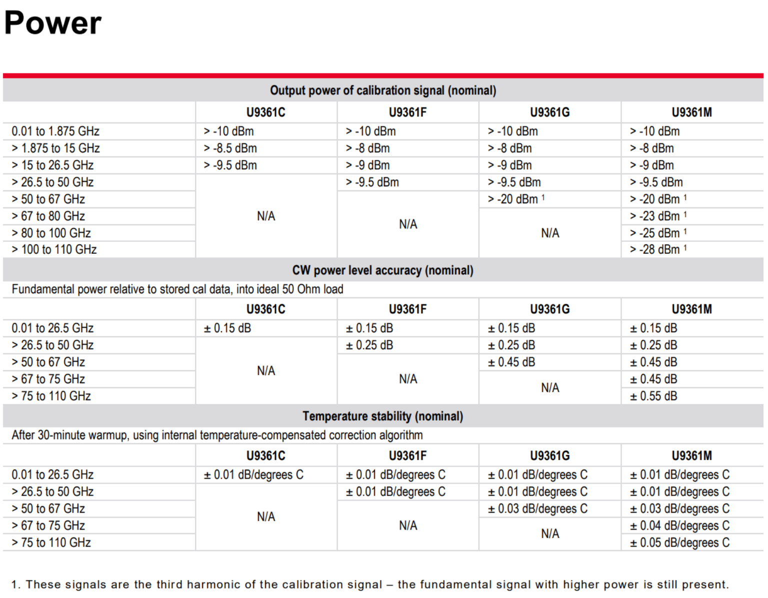
The Keysight M9484C VXG vector signal generator produces various signals up to 54 GHz, ranging from single-tones to modulated ones. It is designed to help to characterize the next generation of communication technologies. The VXG with the V3080A frequency extender can extend the frequency range to 110 GHz [13].

2.4.3 Keysight RCal Receiver Calibrator

The Keysight RCal Receiver Calibrator is a hardware, as Figure 2.6, designed to calibrate and test receiver performance in communication systems. That provides precise measurements and calibration signals to ensure that receivers can accurately receive and process signals. The RCal is capable of transmitting a range of standardized test signals. These signals evaluate receiver sensitivity, selectivity, path loss, and other key performance parameters [14].

Figure 2.6: The RCal calibrator [14]

Calibrators are typically used to calibrate losses in transmission cables. The RCal connects to the analyzer with a USB-C cable and reference cable. The calibration process involves connecting one end of the transmission cable to the analyzer and the other end to the calibrator's RF output. The calibrator emits a known signal, allowing the analyzer to determine the difference between the received signal and the calibrator's output, thus calibrating for phase and magnitude. The specifications of the output power from RCal are shown in Figure 2.7.



Power

Output power of calibration signal (nominal)				
	U9361C	**U9361F**	**U9361G**	**U9361M**
0.01 to 1.875 GHz	> -10 dBm	> -10 dBm	> -10 dBm	> -10 dBm
> 1.875 to 15 GHz	> -8.5 dBm	> -8 dBm	> -8 dBm	> -8 dBm
> 15 to 26.5 GHz	> -9.5 dBm	> -9 dBm	> -9 dBm	> -9 dBm
> 26.5 to 50 GHz	N/A	> -9.5 dBm	> -9.5 dBm	> -9.5 dBm
> 50 to 67 GHz		N/A	> -20 dBm [1]	> -20 dBm [1]
> 67 to 80 GHz			N/A	> -23 dBm [1]
> 80 to 100 GHz				> -25 dBm [1]
> 100 to 110 GHz				> -28 dBm [1]

CW power level accuracy (nominal)

Fundamental power relative to stored cal data, into ideal 50 Ohm load

	U9361C	**U9361F**	**U9361G**	**U9361M**
0.01 to 26.5 GHz	± 0.15 dB	± 0.15 dB	± 0.15 dB	± 0.15 dB
> 26.5 to 50 GHz	N/A	± 0.25 dB	± 0.25 dB	± 0.25 dB
> 50 to 67 GHz		N/A	± 0.45 dB	± 0.45 dB
> 67 to 75 GHz			N/A	± 0.45 dB
> 75 to 110 GHz				± 0.55 dB

Temperature stability (nominal)

After 30-minute warmup, using internal temperature-compensated correction algorithm

	U9361C	**U9361F**	**U9361G**	**U9361M**
0.01 to 26.5 GHz	± 0.01 dB/degrees C	± 0.01 dB/degrees C	± 0.01 dB/degrees C	± 0.01 dB/degrees C
> 26.5 to 50 GHz	N/A	± 0.01 dB/degrees C	± 0.01 dB/degrees C	± 0.01 dB/degrees C
> 50 to 67 GHz		N/A	± 0.03 dB/degrees C	± 0.03 dB/degrees C
> 67 to 75 GHz			N/A	± 0.04 dB/degrees C
> 75 to 110 GHz				± 0.05 dB/degrees C

1. These signals are the third harmonic of the calibration signal – the fundamental signal with higher power is still present.

Figure 2.7: Specifications for RCal [14].

Figure 2.7 details various parameters such as frequency range, output power levels, and associated uncertainties. It provides essential data like the nominal output power across different frequency bands and the expected variability, which are critical for verifying that the RCal performs within specified tolerances.

2.5 SCPI

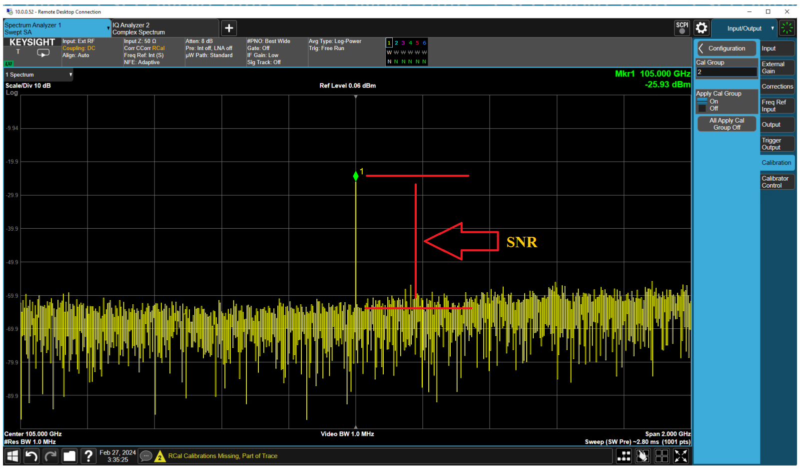
The instruments were placed in various positions and it would be beneficial to control and sync all equipment from one terminal. In such cases, the Standard Commands for Programmable Instruments (SCPI), a standardized command language, simplifies the control over measurement instruments and testing equipment. It is pivotal for setting measurement parameters and data acquisition. This standardization is crucial for ensuring consistent command interpretation and execution, leading to more reliable and reproducible results. Ericsson leverages an internal Python-based framework that utilizes SCPI for seamless communication with various instruments, thereby automating the testing and measurement process.

2.6 Dynamic range

Dynamic range, a crucial metric in signal analysis, is the range of signal strength that can be accurately measured as received or transmitted by an antenna. The minimum signal that can be detected is affected by the system's noise level. A lower noise floor allows for the detection of weaker signals, thereby extending the lower end of the dynamic range. The largest signal strength that the system can handle without introducing unacceptable distortion, is the upper limitation of the dynamic range. The nonlinear distortion is a particular concern, as defined by the compression point. Therefore the two key factors in dynamic range, are noise level, and compression point. These factors are discussed in detail in this section [15].

2.6.1 Signal-to-noise

The signal-to-noise ratio (SNR) is the ratio between the desired signal level and the level of background noise. A ratio higher than one indicates a stronger signal than the noise. SNR plays a crucial role in signal quality. With a high SNR, the signal is clear and easy to detect, like a strong radio signal compared to faint static. Conversely, a low SNR means the signal is weak and prone to corruption by the background noise, making it difficult to distinguish the signal from the noise.



Input: Ext RF
Coupling: DC
Align: Auto
Input Z: 50 Ω
Corr CCorr RCal
Freq Ref: Int (S)
NFE: Adaptive
Atten: 8 dB
Pre: Int off, LNA off
µW Path: Standard
#PNO: Best Wide
Gate: Off
IF Gain: Low
Sig Track: Off
Avg Type: Log-Power
Trig: Free Run
Scale/Div 10 dB
Ref Level 0.06 dBm
Mkr1 105.000 GHz
-25.93 dBm
Cal Group
2
Apply Cal Group
On
Off
Center 105.000 GHz
#Res BW 1.0 MHz
Video BW 1.0 MHz
Span 2.000 GHz
Sweep (SW Pre) ~2.80 ms (1001 pts)
Feb 27, 2024
3:35:25
RCal Calibrations Missing, Part of Trace

Figure 2.8: A screenshot of the UXA that shows the signal as a green dot in the middle and background noise (DANL) towards the bottom.

There are various ways to calculate SNR, depending on how signal and noise measurements are made. In this project, the focus is solely on signal strength. Therefore, the SNR will be calculated in decibels (dB) as illustrated in Figure 2.8. It is defined by the difference between the signal level and the average noise level, directly measurable from the signal analyzer.

$$SNR_{dB} = P_{signal,dBm} - P_{noise,dBm} \tag{2.2}$$

There are various methods to improve SNR, such as increasing signal strength, reducing the noise level, and filtering the signal. Adjusting the analyzer's settings is critical. These adjustments can significantly influence noise levels.
Settings such as resolution bandwidth and internal attenuation have a direct impact on the noise level. They are capable of altering the SNR significantly, thereby optimizing the accuracy and clarity of signal detection.

2.6.1.1 Resolution bandwidth

Resolution bandwidth (RBW) or IF bandwidth is the filter that appears after the signal has been mixed with LO. The IF filter is a bandpass filter that is used as the

"window" for detecting signals. Its bandwidth is the so-called RBW of the analyzer and can be changed via the front panel of the analyzer.
The noise generated in the analyzer is random and increases with the frequency. This noise is commonly referred to as Displayed Average Noise Level (DANL). The total noise power passing through the filters is determined by the bandwidth of the filters. The following equation describes a relation between RBW and DANL.

$$\Delta_{dBm} DANL = 10 \lg \frac{BW_2}{BW_1} \tag{2.3}$$

Where BW_1 is the before adjustment, and BW_2 is the RBW after adjustment. This equation indicates that increasing the RBW (i.e., $BW_2 > BW_1$) results in an increase in DANL, reflecting a higher noise floor while decreasing the SNR.
Additionally, the choice of RBW significantly impacts the sweep time of the spectrum analyzer [17]. The relation can be presented by the following equation:

$$\text{SweepTime} = \frac{K(SPAN)}{RBW^2}, \quad 2 \leq K \leq 3 \tag{2.4}$$

The factor K here is typically a constant that adjusts for the characteristics of the filter used in the analyzer. "For the synchronously-tuned near-Gaussian filters, used in many analog analyzers, the value of K is in the 2 to 3 range" [17]. The inverse square relationship between the sweep time and the RBW indicates that decreasing the RBW significantly increases the sweep time.
It is a trade-off between sweep time and resolution in spectrum analysis. A smaller RBW enhances the SNR but requires a longer sweep time, impacting the overall throughput of measurements. Conversely, increasing the RBW decreases the sweep time, allowing quicker sweeps but at the cost of reduced resolution and lower SNR.

2.6.1.2 Internal Attenuation

The primary function of the internal attenuator is to prevent the overloading of the analyzer by reducing the signal's amplitude, This reduction ensures that the signal remains within the acceptable input range of the instrument. Moreover, the choice of attenuation affects the SNR [17].
When the internal attenuation is increased, the input signal decreases. However, a corresponding increase in the gain of the spectrum analyzer's amplifier can compensate for this attenuation. This automatic gain adjustment ensures a true representation of the received signal for observation and data analysis. However, this gain adjustment not only amplifies the signal but also amplifies the DANL.
The signal remains stationary on the display as the internal attenuation changes, while the DANL goes up or down. As the attenuation increases, the noise level is also amplified due to the corresponding increase in amplifier gain. This amplification leads to a reduction in the SNR. Conversely, with smaller attenuation, the amplifier operates at a lower gain, thereby amplifying less noise and helping to maintain a higher SNR. It is advantageous when processing weaker signals or in noise-sensitive applications. However, caution is required to prevent overpowering the input signal, which could lead to overloading the analyzer.

2.6.2 1 dB Compression Point (P1dB)

Ideally, the input signal and output signal will be linear. The gain of the output signal is constant for the input signal. In practice, after the input signal power increases to a certain level, the gain of the output signal will start to decrease. The 1 dB compression point is the point when the output power drops in addition by 1 dB compared to the linear relationship with the input power, as shown in Figure 2.9. This point indicates that the system begins to lose linearity significantly, i.e., the output signal begins to deteriorate in linearity with respect to the input signal, resulting in distortion [19].

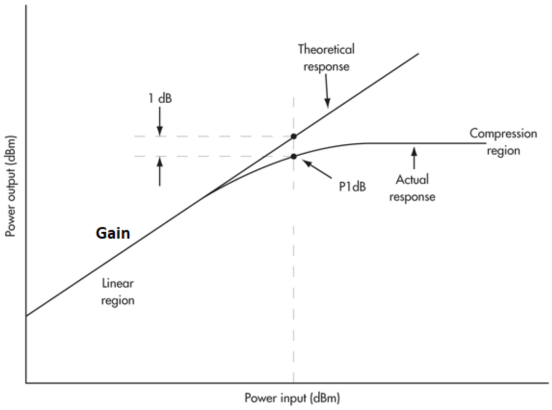


Figure 2.9: The P1dB point [20]

P1dB also indicates a measure of the upper limit of the linear operating range and dynamic range of an amplifier or antenna system. That helps determine the maximum input signal power that the measurement system can handle.

3

Methods

The evaluation of signal OTA in CTRA chamber, from 18 to 110GHz with horn antenna, was realized in two phases. Phase 1 was to test and verify the new calibration method. That moves the reference plane, where the signal characteristics are measured, to the point at which devices or products would be located. The second phase was to generate the signal and measure the P1dB of the system. This phase assessed the dynamic range of the test system, specifically its capacity to handle the power. Prior to these phases a sanity check using RCal calibrator and V3050A extender was performed to build the basics of the calibration and instrument.

Prior to Phase 1: Sanity Check

To confirm a functional calibration before system evaluation, a sanity check was conducted using RCal and V3050A extender. This ensured the calibration produced the desired outcome for the subsequent measurements.
First, the RCal was used under normal conditions for calibrating a transmission line. The RCal was connected to the V3050A via two mm-Wave cables with an adapter of female-to-female in between. Then an attempt was made to calibrate the loss in air over a short distance without antenna. A simple OTA test was facilitated by a small air gap of about 2 mm between the instruments without any cables or adapter.

3.1 Setup

At phase 1, the calibration step, the antenna was mounted with the green 3D printed bracket, as illustrated in the left picture of Figure 3.1. The bracket also serves as the holder for RCal.

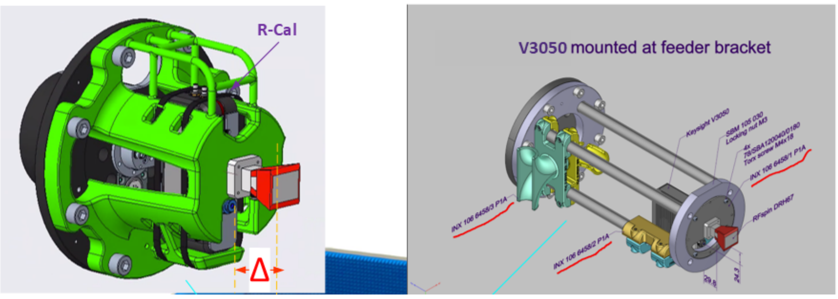


Figure 3.1: Antenna mounted with the 3D printed bracket

In the picture on the right side in Figure 3.1, the fixture on the front is the holder for V3050A/ V3080A extenders. The fixture on the back of the image is the cable holder for the extender, to ensure the cable does not become entangled when the roll head or feed rotates in the CATR. The CAD sketches for holders were provided by Ericsson and were printed using PolyLactic Acid material.

3.2 Phase 1

In phase 1, the UXA signal analyzer was placed outside the CATR chamber at position C, as shown in Figure 3.2. The Figure shows the main equipment in the CATR chamber, along with the positioning and connections of the measuring instruments. The RCal was placed between the horn antenna and the UXA, and fastened in the feed. The UXA is also connected to the V3050A extender which was fastened on position B, the roll.

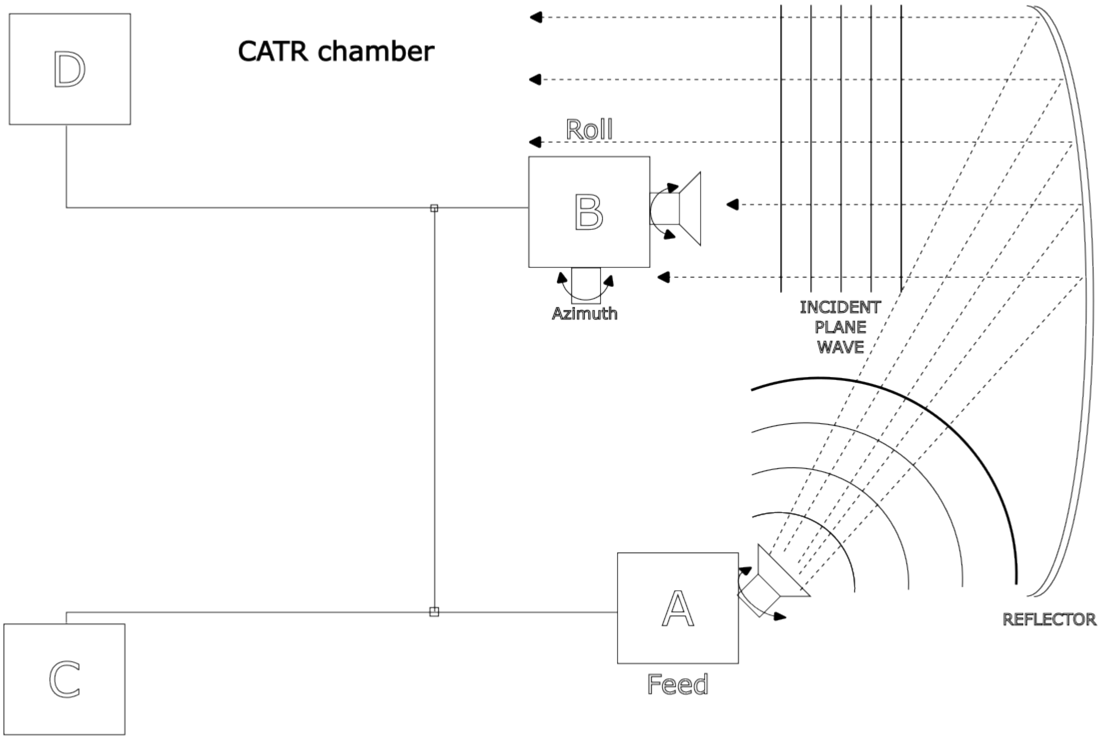


Figure 3.2: Schematic description of a CATR chamber

The RCal transmits a known signal through a DRF 110 Spin horn antenna, which travels OTA onto the reflector and then towards the receiving horn antenna mounted on position B. So the signal is received by the V3050A and into the signal analyzer at C position via the cable. The measured value on the received signal after the calibration should closely match the output power of the RCal. This calibration moves the reference plane to the feed position.

The calibration process, covering frequencies from 18 GHz to 110 GHz by 1 GHz increment each step, is implemented through a Python script in Appendix A.1. This script used the SCPI interface to manage the operations of the instruments. The flowchart illustrated in Figure 3.3 of the code details the process for calibration in spectrum mode and I/Q mode in the UXA and recording the measurement and data.

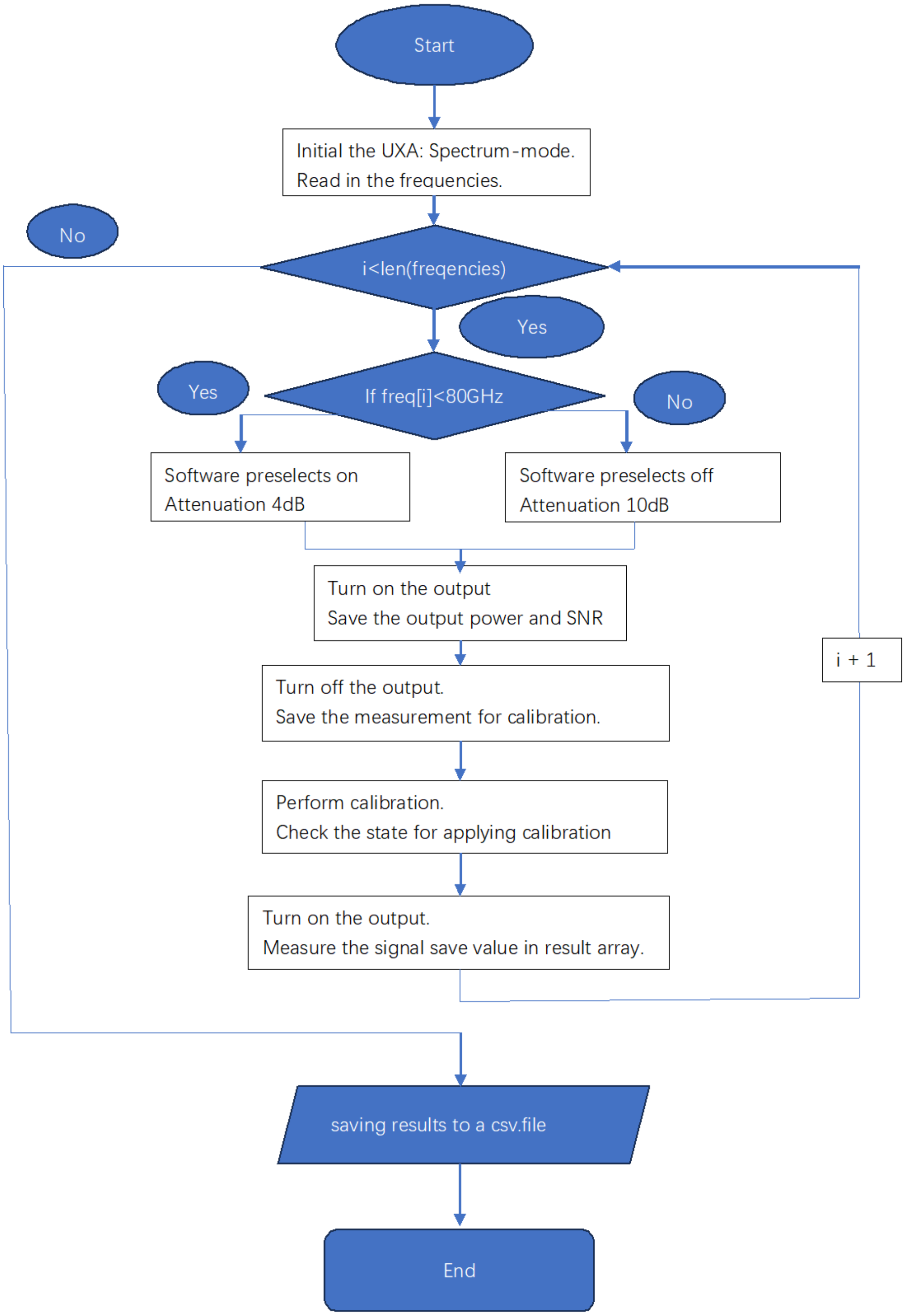
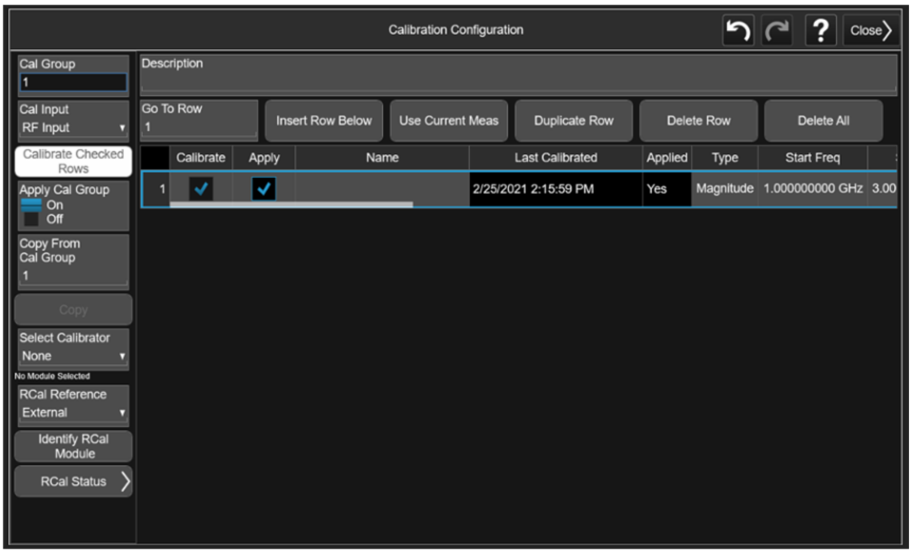


Figure 3.3: Flowchart of measurement script for phase 1

For different frequencies, specific settings, such as RBW and attenuation, are adjusted to maintain the SNR around 20 dB, which was a requirement by Ericsson. The power before calibration and the SNR have also been recorded.
The signal analyzer, when connected with RCal, can automatically calibrate all gains and losses between two measurement points, the transmitting and receiving points. Calibration was processed in the UXA and Figure 3.4 shows the window of the calibration configuration.



Cal Group
1

Cal Input
RF Input

Apply Cal Group
● On
○ Off

Copy From Cal Group
1

Select Calibrator
None
No Module Selected

RCal Reference
External

Description

Go To Row
1

	Calibrate	Apply	Name	Last Calibrated	Applied	Type	Start Freq	
1	☑	☑		2/25/2021 2:15:59 PM	Yes	Magnitude	1.000000000 GHz	3.00

Figure 3.4: Calibration window in the UXA

The window includes parameters and controls for the calibration. Start with the "Use Current Meas" function to save the measurement for calibration. The calibration process includes the following steps. First, the "Calibrate" box must be checked for the corresponding row. Then, use the "Calibrate Checked Rows" to initiate the calibration process. After calibration is complete, the correction needs to be applied by checking the “Apply” column for the corresponding row.
To ensure that the program has applied the calibration successfully, the status displayed in the ’Applied’ column must be verified. This basically shows if the calibration has been applied or not with ’Yes’ or ’No’. The script processes all the calibrations for frequencies one by one to save the status of each calibration and signals power after the process. This is a double confirmation that the calibration was successful or not.

3.2.1 Resolving Setup Challenges in the CATR Chamber

Initially, the signal analyzer was positioned outside the CATR, with USB cables and SMA cables connected to an adapter on the CATR wall, which in turn were forwarded inside the CATR to the V3050A extender located at position B and RCal positioned on A from Figure 3.2. Due to unsatisfying results together with some error messages from instruments, an investigation was made. The culprit was found to be the adapter (power shell) that sends and amplifies USB signals through the CATR. Consequently, the original setup was no longer viable. To circumvent the power shell the instruments were placed inside the CATR since both the extenders (3050A and 3080V) and RCal need a USB connection. Placing the signal analyzer inside the CATR resolved the issue without any further errors.

3.3 Phase 2

In phase 2, the signal source changes from RCal to the VXG generator. For this setup, the V3050A extender is kept in the same place at roll as shown in Figure 3.5, and connected to the UXA signal analyzer.

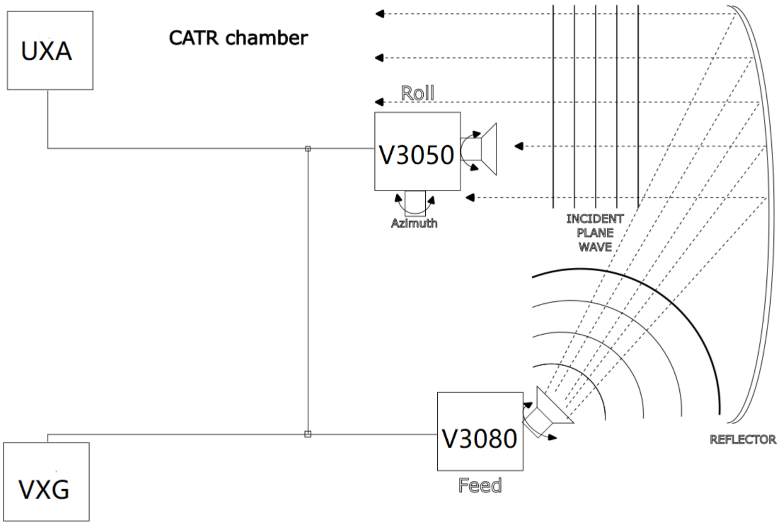


Figure 3.5: The setup of phase 2 in the CATR chamber

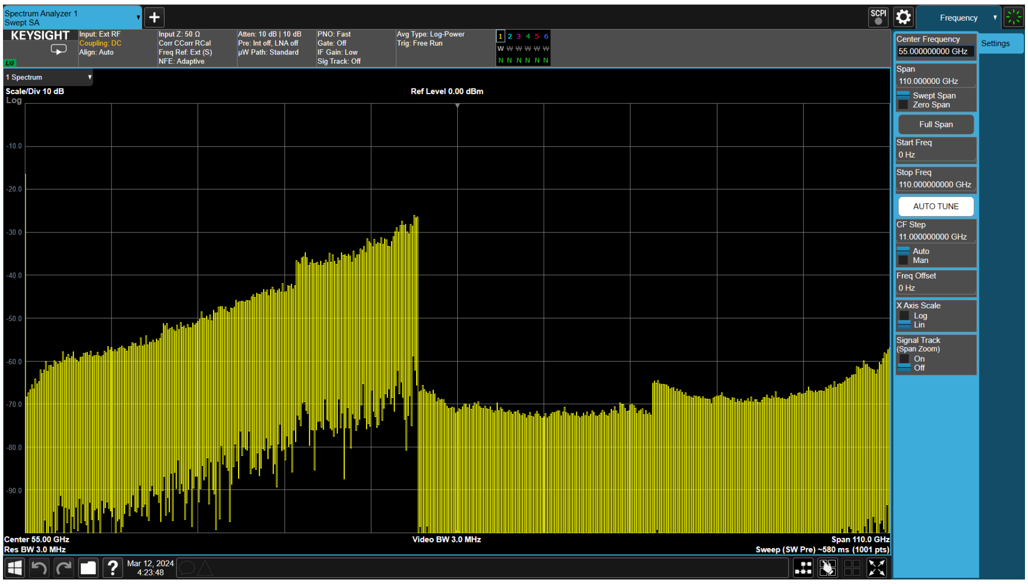
Instead of RCal the V3080A extender is placed on the feed and connected to the signal generator. This configuration allows for the generation of signals up to 110 GHz, facilitating the testing of both CW and NR 5G signals.
To utilize the calibration from Phase 1 and measure the system's P1dB, a new Python script was developed, provided in Appendix A.2. It tests frequencies ranging from 20.5 GHz to 110 GHz in 10 GHz increments. The analyzer needs to be used with the same calibration settings as when it was previously calibrated. The frequency for the VXG signal generator must then be selected first to generate the corresponding signal.
The script evaluates the calibration for accuracy and saves the measurement data to compare the output power set by the signal generator. The signal generator is then gradually swept from -30 dBm to 20 dBm and then compared to the receiver power. In order to perform a compression point measurement. After that, a linear fit is performed to find a linear relationship between input power and received power. The difference between the fitted line and the actual measured received power is then calculated. The point where this difference is closest to -1dB, is the P1dB point. Finally, the P1dB of all tested frequencies can be plotted.

4

Results & Discussion

The key results obtained from phases 1 & 2 will be presented and discussed in this chapter.

4.1 Calibration



Input: Ext RF
Coupling: DC
Align: Auto

Input Z: 50 Ω
Corr CCorr RCal
Freq Ref: Ext (S)
NFE: Adaptive

Atten: 10 dB | 10 dB
Pre: Int off, LNA off
μW Path: Standard

PNO: Fast
Gate: Off
IF Gain: Low
Sig Track: Off

Avg Type: Log-Power
Trig: Free Run

Scale/Div 10 dB
Ref Level 0.00 dBm

Center 55.00 GHz
Res BW 3.0 MHz
Video BW 3.0 MHz
Span 110.0 GHz
Sweep (SW Pre) ~580 ms (1001 pts)

Center Frequency
55.000000000 GHz

Span
110.000000 GHz
Swept Span
Zero Span

Start Freq
0 Hz

Stop Freq
110.000000000 GHz

CF Step
11.000000000 GHz
Auto
Man

Freq Offset
0 Hz

X Axis Scale
Log
Lin

Signal Track
(Span Zoom)
On
Off

Figure 4.1: Picture of the whole frequency span of 10 MHz to 110 GHz on the UXA, with the DANL in yellow.

Figure 4.1 shows the whole frequency span from 10 MHz to 110 GHz with DANL that the UXA, paired with the V3050A extender, can display. There is a clear disruption at 50 GHz, this is where the V3050A take over to analyze the signal and send that data to the UXA.
Figure 4.2 below illustrates the results from the sanity check in the frequency range from 18 GHz to 110 GHz. The yellow curves represent the power before calibration, while the blue curves show the data after calibration. It can be observed that the calibration was relatively smooth and the calibrated powers were essentially the same in both tests.

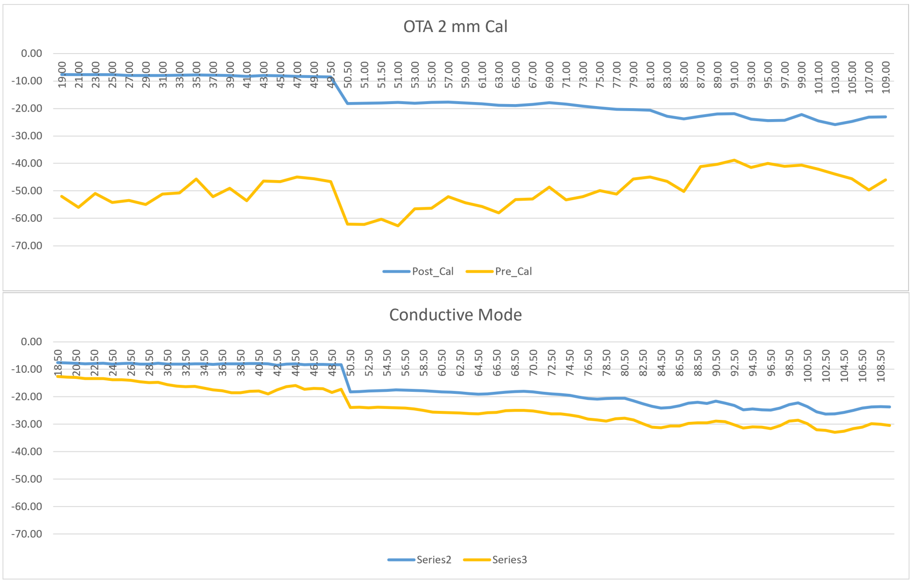


Figure 4.2: Two calibrations; one in conductive and another with an air gap of 2mm, spanning the frequency range from 18 to 110 GHz

4.1.1 Spectrum mode

In phase 1 the UXA has checked the whole frequency range with the CW signal and has been calibrated successfully in spectrum mode. The CW signals power before and after the calibration and SNR from 20 GHz to 110 GHz are summarised in Figure 4.3 below.

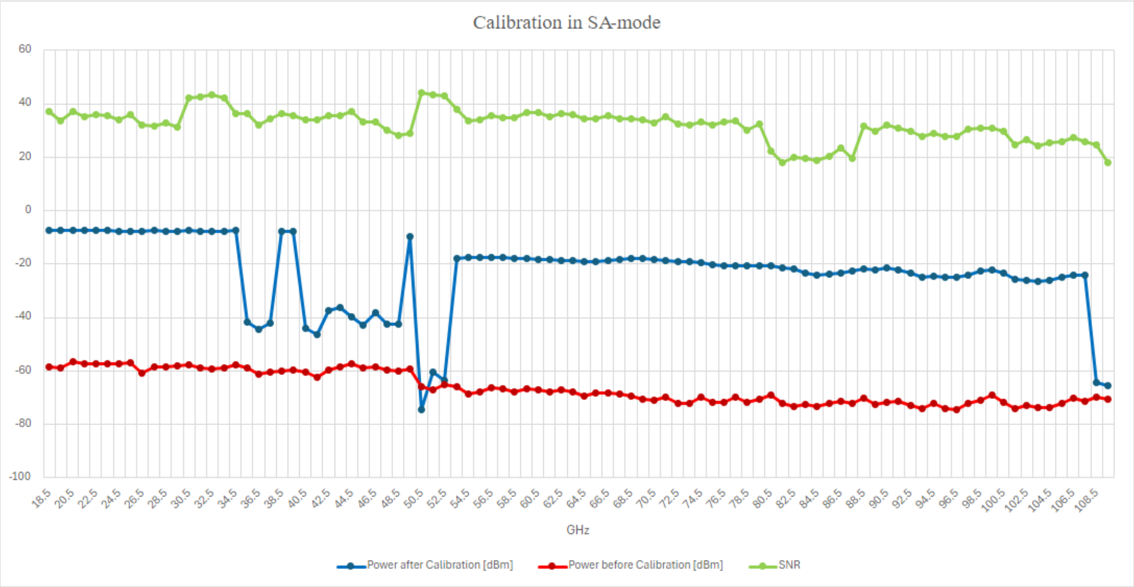


Figure 4.3: The power before and after calibration and SNR, across the frequency range from 20 GHz to 110 GHz.

Three data series are presented in Figure 4.3. The red line is the signal's power before calibration. This curve is relatively smooth but has a general downward

trend. As the frequency gets higher, the power level gradually decays, because of FSPL.
The calibrated power level of the signals is represented by the blue line, showing significant fluctuations in power level at different frequencies. It should be the same as the reference value for RCal's output power. But there are multiple peaks and deep valleys. From 35 GHz to 45 GHz, the calibrated power values are raised a bit and do not reach the reference value. Around 50 GHz and 110 GHz the power level are in line with the pre-calibration power. Other than those, most frequencies achieved successful calibration with the power levels aligning with the Rcal reference output power.
The green curve label with SNR. The value overall is better than 20 dB over all frequencies. The curve also shows moderate fluctuations as the frequency changes, with a similar pattern as the calibrated power curve, but with smaller fluctuations. For the troublesome range, the RBW was very small, thereby barely sustaining the SNR around 20 dB. The signal power is weak at this frequency range. After 80 GHz, the attenuation was changed to 10 dB, which had definite effects on the SNR. But overall, judging from Figure 4.3, no direct relation between SNR and calibration results can be seen.

4.1.2 I/Q mode

Figure 4.4 below illustrates the calibration results in I/Q mode in the frequency range from 24.45 GHz to 71 GHz. Same as in the spectrum mode, it shows the power and SNR of the signal before and after calibration.

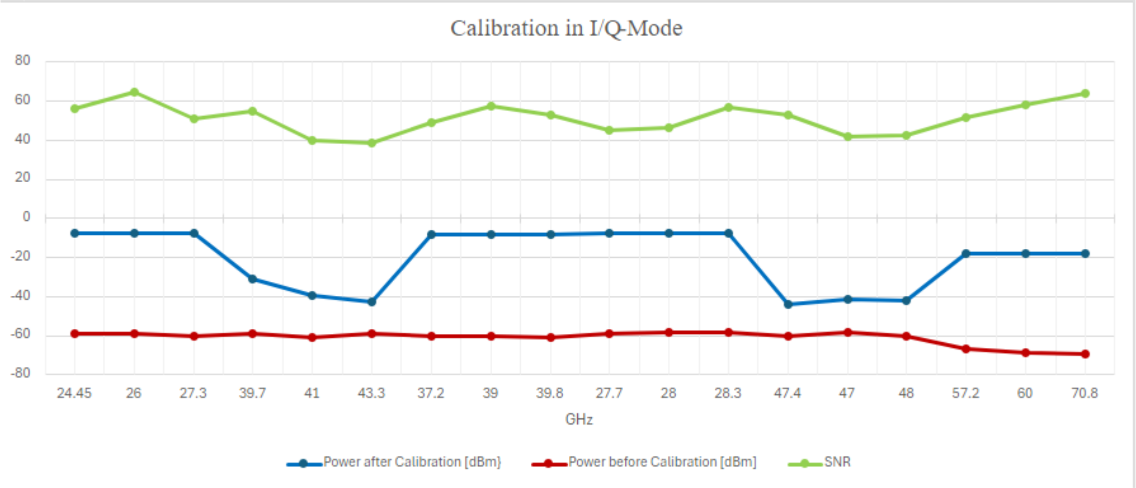


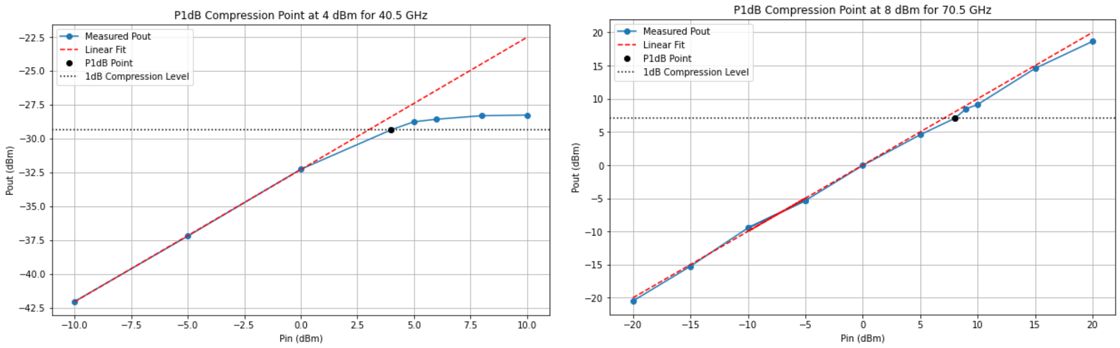
Figure 4.4: Calibration in I/Q mode with SNR, power before and after the calibration between 20 GHz to 110 GHz.

The green curve represents the SNR, which fluctuates in the upper half of the graph and appears to vary between 40 dB and 60 dB. The red curve shows the power before calibration, with an overall trend that is relatively flat and consistent. The signal power after calibration, which represents the blue line, starts near 0 and then fluctuates dramatically, with several significant dips. These are around 39 GHz to

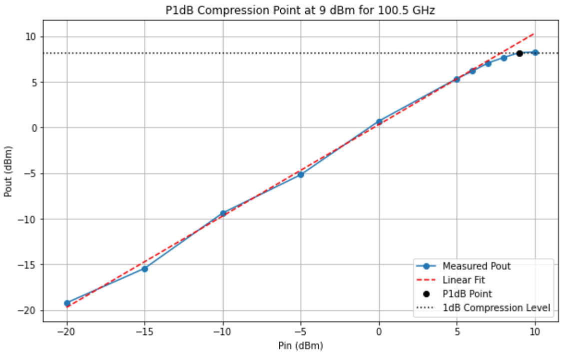
50 GHz, and the calibrated power is not as good as expected. But it is the same as the results from the spectrum mode of measurement.

4.2 Maximal output power

In phase 2, the resulting P1dB points were important indicators of system linearity, signal processing capacity, dynamic range, and system performance characteristics. The compression points were measured at some frequencies. Such as in Figure 4.5a and Figure 4.5c, the compression point is readily identifiable at 40.5 GHz and 110.5 GHz. But, as shown in Figure 4.5b, there is no obvious compression point in the 70.5 GHz example. The compression point on this frequency of the system may be out of the measurement range. As a whole most compression points below 50 GHz were found. Whereas when moving into the upper half of the frequency range somewhere is difficult to define if it is a real compression point or not.



(a) Compression Points at 40.5 GHz. **(b)** Compression Points at 70.5 GHz.



(c) Compression Points at 100.5 GHz.

Figure 4.5: P1dB at different frequency

The compression points from 20.5 GHz to 100.5 GHz have been summarized in Figure 4.6 below. There is a clear trend that the compression points decrease within the span of the UXA without the header, in other words, frequencies below 50 GHz.

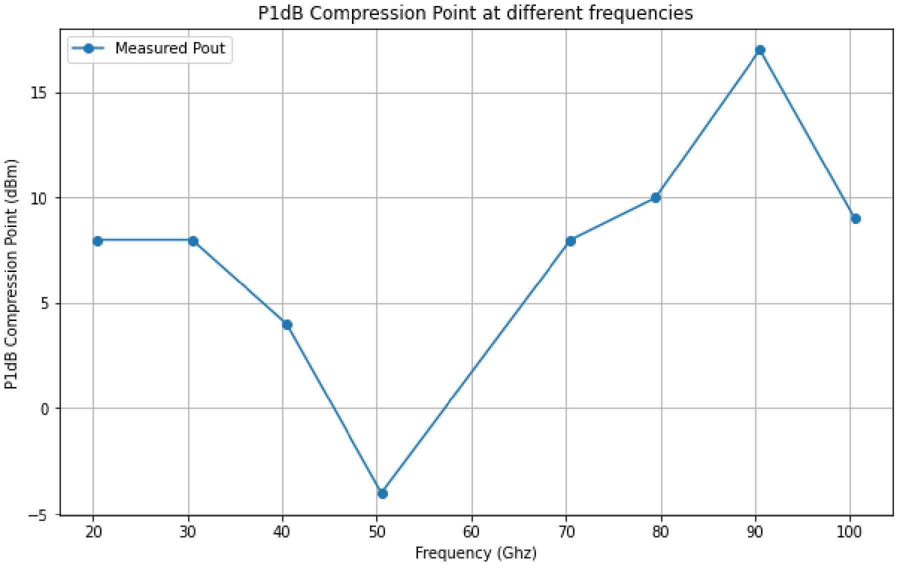


Figure 4.6: Compression Points from 20.5 GHz to 100.5 GHz.

In the higher frequency ranges where both frequency extenders are active, the compression points that are found may not always be the true compression points. This is because frequency extenders can introduce additional non-linearities and distortions, which may affect the accuracy of the measurements. It is also possible that the compression point is not within the measurement range due to the maximum power that can be produced by the VXG signal generator, which represents the maximum input power of the measurement system.

5

Conclusion

In this chapter, the project's main results will be concluded, together with areas of further research and improvement for the future.

5.1 Achievements

The primary focus of the task is to test new avenues of calibration. The testing from 18 GHz to 110 GHz was done with an overall success.
This setup significantly improves the efficiency of the calibration. In the wide frequency range, only one calibration is required to move the reference plane to the device under test. Furthermore, the entire process is automated, by scripts to perform the calibration process. The method reduces the need for multiple antennas and frequent calibrations, thereby reducing the time and cost of testing.
Traditionally, when multiple antennas are used for measurements, the precision can be reduced. This is simply because different measurement systems have different properties and data that have to be considered. The new method demonstrated that, even though RCal is not designed for OTA calibrations, it did indeed work.

5.2 Further development

There were unstable signals in some frequency ranges, that impacted the system's accuracy and reliability. The RCal faced challenges around 40 GHz and the dynamic range was not clearly defined at several frequencies. There may be unidentified factors that need to be taken into consideration in the system. They can be influenced by the CATR chamber environment and the performance of the instruments. Especially V3080A was an early prototype, but will probably be improved which will help with some of the issues. Addressing these imperfections requires a focused effort to refine the test system. Improving the test system will provide more efficient technical support for product testing and future development of communication technology. The RCal calibration was overdue, post calibration showed no remarks.

Bibliography

[1] "5G NR frequency bands", 3GPP. `https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3284` (Accessed: 12 April 2024).

[2] Ericsson, 'What, why and how: 5G carrier aggregation,' Ericsson Blog, 24 June 2021. [Online]. Available: https://www.ericsson.com/en/blog/2021/6/what-why-how-5g-carrier-aggregation. (Accessed: 12 April 2024).

[3] Ericsson, 'Leveraging the potential of 5G millimeter wave,' Ericsson, [Online]. Available: https://www.ericsson.com/en/reports-and-papers/further-insights/leveraging-the-potential-of-5g-millimeter-wave. (Accessed: 12 April 2024).

[4] Ericsson, '6G Spectrum,' Ericsson, [Online]. Available: https://www.ericsson.com/en/6g/spectrum. (Accessed: 12 April 2024).

[5] Ericsson, 'H. Peter, private communication, Mar. 2024' at Ericsson

[6] B. T. Jijo, S. R. M. Zeebaree, R. R. Zebari, M. A. M. Sadeeq, et al., "A Comprehensive Survey of 5G mm-Wave Technology Design Challenges," in Asian Journal of Computer Science And Information Technology, vol. 8, no. 1, pp. 1-20, Apr. 2021. DOI: 10.9734/AJRCOS/2021/v8i130190 (Accessed: 12 March 2024).

[7] "Free-Space Path Loss" Encyclopedia, HandWiki. 26 October 2022. Available at: https://encyclopedia.pub/entry/31362 (Accessed: 13 March 2024).

[8] C.A Balanis. Antenna theory: Analysis and design. en. 4th ed. Hoboken, NJ: Wiley-Blackwell, 2016.

[9] Thomas A Milligan. Modern antenna design. en. 2nd ed. Hoboken, NJ: John Wiley & Sons, 2005.

[10] RFspin, "Double Ridged Horn Antenna" 2024. [Online]. Available at: https://www.rfspin.com/product/drh110/. (Accessed: 10 April 2024).

[11] Castañer M.S. and Foged L.J., "2.5.2.2 Near-Field to Far-Field", in Post-Processing Techniques in Antenna Measurement. Institution of Engineering and Technology (The IET), 2019.[Online]. Available at: `https://search.ebscohost.com/login.aspx?direct=true&db=edsknv&AN=edsknv.kt0124J704&site=eds-live&scope=site` (Accessed: 12 March 2024).

[12] Keysight, "V3050A Signal Analyzer Frequency Extender"[Online]. Available at: `https://www.keysight.com/us/en/assets/3122-2019/data-sheets/V3050A-Signal-Analyzer-Frequency-Extender.pdf` (Accessed: 20 March 2024).

[13] Keysight, "M9484C VXG and V3080A Vector signal generator and frequency extender"[Online]. Available at: `https://www.keysight.com/us/en/assets/3122-1445/data-sheets/M9484C-VXG-and-V3080A.pdf` (Accessed: 20 March 2024).

[14] Keysight, "U9361 RCal Receiver Calibrator" [Oniline]. Available at: `https://www.keysight.com/us/en/assets/3120-1426/technical-overviews/U9361-RCal-Receiver-Calibrator.pdf` (Accessed: 20 March 2024).

[15] Keysight, "Network Analyzer Dynamic Range - Understanding and Improving" [Online]. Available at: `https://www.keysight.com/us/en/assets/7018-06837/application-notes/5980-2778.pdf` (Accessed: 20 March 2024).

[16] National Instruments, "Dynamic Range," NI RFSA Documentation, National Instruments, 2023. [Online]. Available at: `https://www.ni.com/docs/en-US/bundle/ni-rfsa/page/dynamic-range.html.` (Accessed: May 1 2024).

[17] "Spectrum Analysis Basics", [Online]. Available at: `https://www.keysight.com/nl/en/assets/7018-06714/application-notes/5952-0292.pdf` (Accessed: 20 March 2024).

[18] "DRH110 Datasheet", [Online]. Available at: `https://www.rfspin.com/wp-content/uploads/2024/02/DRH110-%E2%80%93-RF-SPIN.pdf` (Accessed: 20 March 2024).

[19] "What's The Difference Between The Third-Order Intercept And The 1-dB Compression Points?", [Online]. Available at:https://www.electronicdesign.com/resources/whats-the-difference-between/article/21799714/whats-the-difference-between-the-third-order-intercept-and-the-1-db-compression-points (Accessed: 14 March 2024).

[20] "What is P1dB? - everything RF", [Online]. Available at: `https://www.everythingrf.com/community/what-is-p1db` (Accessed: 14 March 2024).

A

Appendix

A.1 Appendix 1 - Measurement script for phase 1

Measurement script 1

```
import os
import sys
import csv
import time
import logging
import vf_core
from pathlib import Path
from enum import Enum

# Where should we save results
if sys.platform == ’win32’:
    results_path = r’C:\test_results’
else:
    results_path = Path(r’~/test_results’).expanduser()

results_file = r’results.csv’
# Create directories if they don’t exist
if not os.path.exists(results_path):
    os.makedirs(results_path)

# Specify the frequencies to use in our test function

# CSV results file header row
title = [’Band’, ’frequencies [Hz]’, ’Power [dBm]’,
        ’Power before Cal[dBm]’,
         ’start_frequencies [Hz]’, ’stop_frequencies [Hz]’,
         ’Cal applied?’,
         ’SNR’, ’RBW’, ’sweeptime’, ’Cal row’]

# Result handling. Make a timestamp to use for file name
ts = time.strftime(’%Y-%m-%d_%H_%M_%S’)
results_file = ’{}_{}’.format(ts, results_file)
results_file = os.path.join(results_path, results_file)


# Read the file
file_path = r’C:\Users\Desktop\TestCWFreq.csv’
```

```python
frequencies = []
start_frequencies = []
stop_frequencies = []
band = []
RBW = []
with open(file_path, 'r', newline='')as file:
    reader = csv.reader(file, delimiter=',')
    # Skip first row of the document
    next(reader)
    for row in reader:
        frequencies.append(row[7].replace(",", "."))
        start_frequencies.append(row[3].replace(",", "."))
        stop_frequencies.append(row[4].replace(",", "."))
        RBW.append(row[5].replace(",", "."))
        band.append(row[0])

def my_test(instruments, comp, freqs=frequencies, results_file=None,
            start_freqs=start_frequencies, stop_freqs=stop_frequencies):
    """
    Demo test for Validation Framework.

    Doing result file handling, instrument communications etc.
    Raises
    ------
    EnvironmentError
        If instrument is missing.

    """
    try:
        # Fix for code completion to work for instruments in Spyder
        from framework.station.autogen_instruments import Instruments
        instruments = Instruments()

        # Open file for appending results data, create
        only if results_file
        # exists
        wr = None
        if results_file is not None:
            fh = open(results_file, 'a', newline='')
            wr = csv.writer(fh, delimiter=',')
            # Create a csv writer object for writing CSV

        total_results = []  # Keep measured data
        # Add header to file only if it does not exist
        total_results.insert(0, title)

        uxan9042b = instruments.keysight_uxan9042b
        uxan9042b.setTimeout(800000)

        # SA mode with Cal Group 1
        uxan9042b.saMode()
```

```
        uxan9042b.output.setCalSource('RCM1')
        uxan9042b.output.Calgroup(1)
        uxan9042b.output.delete_all_cal_rows()
        time.sleep(10)
        uxan9042b.amplitude.setSignalPath('FULL')

        uxan9042b.output.delete_all_cal_rows()
        time.sleep(10)

        for i in range(len(freqs)):
            results = []
            Cal_row = len(freqs)-i
            freq = float(freqs[i])
            uxan9042b.frequency.setFrequencyCenter(freqs[i])
            uxan9042b.frequency.setFrequencyStart(start_freqs[i])
            uxan9042b.frequency.setFrequencyStop(stop_freqs[i])
            # Set on the output and the output Frequency
            uxan9042b.output.setOutputFrequency(freqs[i])
            uxan9042b.output.setOutputState(1)
            time.sleep(5)

            # Set the RBW and attenuation
            # for different settings for frequency
            # above 80G and below
            uxan9042b.frequency.setResolutionBw(RBW[i])
            if freq < 50e9:
                uxan9042b.amplitude.setExtendAttenuationValue(4)
                uxan9042b.amplitude.setAttenuationValue(4)
            else:
                uxan9042b.amplitude.setSoftwarePreselection(1)
                if freq > 80e9:
                    uxan9042b.amplitude.setSoftwarePreselection(0)
                    uxan9042b.amplitude.setExtendAttenuationValue(10)
                    uxan9042b.amplitude.setAttenuationValue(10)

            sweeptime = uxan9042b.sweep.getSweepTime()
            sweeptime = float(sweeptime)
            if sweeptime > 1000:
                sweeptime /= 1000
            time.sleep(sweeptime*2.5)

            # save the value before cal
            uxan9042b.marker.setMarkerXvalue(1, freqs[i])
            time.sleep(5)
            y1 = uxan9042b.marker.getMarkerYvalue(1).strip()
            uxan9042b.marker.setmarkermode(1, "delta")
            time.sleep(2)
            uxan9042b.marker.findMarkerNextPeak(1)
            time.sleep(5)
```

```
            y2 = uxan9042b.marker.getMarkerYvalue(1).strip()
            uxan9042b.marker.setmarkermode(1, "position")

            # Turn off the output and save the
            # measurement for calibration
            uxan9042b.output.setOutputState(0)
            time.sleep(sweeptime*2.5)
            uxan9042b.output.setOutputAplCalGroup(1)
            uxan9042b.output.setOutputCurrentMeas(1)
            time.sleep(5)

            # calibration check
            uxan9042b.output.set_Cali_Row(1, 'ON')
            uxan9042b.output.setOutputCaliChecked()
            time.sleep(5)

            # apply calibration and save the data in the
               array
            uxan9042b.output.apply_Cal_row(1, 'ON')
            appliedcheck = uxan9042b.output.check_calstate(1)
            # check the state
            while appliedcheck == '---':
                time.sleep(5)
                appliedcheck = uxan9042b.output.
                   check_calstate(1)

            # set on the output
            uxan9042b.output.setOutputState(1)
            time.sleep(sweeptime*2.5)
            if sweeptime < 10:
                time.sleep(10)

            uxan9042b.marker.setMarkerXvalue(1, freqs[i])
            time.sleep(5)
            x = uxan9042b.marker.getMarkerXvalue(1).strip()
            y = uxan9042b.marker.getMarkerYvalue(1).strip()

            print(appliedcheck)
            if appliedcheck == 'Yes':
                results.extend([band[i], x, y, y1,
                   start_freqs[i],
                                stop_freqs[i], appliedcheck,
                                   y2,
                                RBW[i], sweeptime, Cal_row])
                total_results.append(results)
            else:
                results.extend([band[i], x, y, y1,
                   start_freqs[i],
                                stop_freqs[i], appliedcheck,
                                   y2,
                                RBW[i], sweeptime, Cal_row])
                total_results.append(results)
                print('apply calibration failed')
            uxan9042b.output.apply_Cal_row(1, 'OFF')
            uxan9042b.output.set_Cali_Row(1, 'OFF')
            uxan9042b.output.setOutputAplCalGroup(0)
```

```
        uxan9042b.output.setOutputState(0)

        print('open new window for IQ mode')

        # //////////// I/Q mode ////////////////////////
        # Cal Group 2
        # read the file for the IQ mode
        file_path = r'C:\Users\Desktop\Book1.csv'
        iqfreqs = []
        iqstart_frequencies = []
        iqstop_frequencies = []
        with open(file_path, 'r', newline='')as file:
            reader = csv.reader(file, delimiter=',')
            # Skip first row of the document
            next(reader)
            for row in reader:
                iqfreqs.append(row[7].replace(",", "."))
                iqstart_frequencies.append(row[3].replace(",", "."))
                iqstop_frequencies.append(row[4].replace(",", "."))

        # open a new window,set to the iq mode- with spectrum
        # complex
        uxan9042b.iqm.Addscreen()
        uxan9042b.iqModeNr()
        uxan9042b.iqm.selcomplexmode()
        # seleted the calue for the IFpath and Cal group
        uxan9042b.iqm.setifpath(255e6)
        uxan9042b.output.Calgroup(2)
        uxan9042b.output.SetCalInput('EXTERNAL')
        # uxan9042b.iqm.SetCaltype('COMB')
        uxan9042b.iqm.setspacing(1e6)
        # clear
        uxan9042b.output.delete_all_cal_rows()
        time.sleep(10)
        total_results.append(['iqsignal'])
        uxan9042b.amplitude.setSignalPath('FULL')

        #  measurement
        for i in range(len(iqfreqs)):
            iqresults = []

            uxan9042b.amplitude.setExtendAttenuationValue(4)
            uxan9042b.amplitude.setAttenuationValue(4)

            Cal_row = len(iqfreqs) - i
            # Set Center Frequency and span
             uxan9042b.frequency.setFrequencyCenter(iqfreqs[i])
            uxan9042b.iqm.setSpan(250e6)
            uxan9042b.iqm.setBW(1000)
            uxan9042b.iqm.setautoscaling(2, 1)

            # Set on the output and the output Frequency
            uxan9042b.output.setOutputFrequency(iqfreqs[i])
```

```
            uxan9042b.output.setOutputState(1)
            time.sleep(5)
            uxan9042b.iqm.setautoscaling(2, 1)

            # keep the value before Cal.
            uxan9042b.iqm.setMarker(1, iqfreqs[i])
            # uxan9042b.iqm.setmarkerfunction(1, 'bandpower')
            time.sleep(5)
            y1 = uxan9042b.iqm.getmarkerY(1).strip()
            uxan9042b.iqm.setautoscaling(2, 0)
            uxan9042b.iqm.SetMarkermode(1, "delta")
            time.sleep(2)
            uxan9042b.iqm.setMarker(2, (float(iqfreqs[i]) - 30e6))
            time.sleep(5)
            y2 = uxan9042b.iqm.getmarkerY(1).strip()

            uxan9042b.iqm.SetMarkermode(2, 'OFF')
            uxan9042b.iqm.SetMarkermode(1, "position")

            # use currentmeasurement setting for calibration
            uxan9042b.output.setOutputState(0)
            time.sleep(10)
            uxan9042b.output.setOutputAplCalGroup(1)
            uxan9042b.output.setOutputCurrentMeas(1)
            time.sleep(5)

            uxan9042b.output.set_Cali_Row(1, 'ON')
            uxan9042b.output.setOutputCaliChecked()
            time.sleep(5)

            # apply calibration, and check the state of it
            uxan9042b.output.apply_Cal_row(1, 'ON')
            appliedcheck = uxan9042b.output.check_calstate(1)
            while appliedcheck == '---':
                time.sleep(5)
                appliedcheck = uxan9042b.output.check_calstate(1)

            uxan9042b.output.setOutputState(1)
            time.sleep(15)
            uxan9042b.iqm.setautoscaling(2, 1)
            uxan9042b.iqm.setMarker(1, iqfreqs[i])
            # uxan9042b.iqm.setmarkerfunction(1, 'bandpower')
            time.sleep(2)
            x = float(uxan9042b.iqm.getmarkerX(1))
            y = float(uxan9042b.iqm.getmarkerY(1))
            print(appliedcheck)
            if appliedcheck == 'Yes':
                iqresults.extend([band[i], x, y, y1, 0,
                                 0, appliedcheck, y2,' ',
                                 ' ', Cal_row])
            else:
                iqresults.extend([band[i], x, y, y1, 0,
                                 0, appliedcheck, y2, 0, 0, Cal_row])
```

```
                print('apply calibration failed')

            total_results.append(iqresults)
            uxan9042b.output.apply_Cal_row(1, 'OFF')
            uxan9042b.output.set_Cali_Row(1, 'OFF')
            uxan9042b.output.setOutputAplCalGroup()
            time.sleep(10)
            uxan9042b.output.setOutputState(0)
        print(total_results)

        # Write results to file
        if wr is not None:
            for row in total_results:
                wr.writerow(row)
            fh.close()  # Close results file

        return total_results  # Return to caller for post processing

    except IOError:
        log.exception(f'Could not open results file: {results_file}, '
                      'for writing, error: ')
    except Exception as e:
        log.exception(e)


if __name__ == '__main__':
    # =============================
    # Setup Validation Framework
    # ============================
    # Set debug=False in vf_core.start_framework below to enable instrument
    # communication, otherwise fake calls are used (for testing/demo)
    instruments, comp = vf_core.start_framework(
        sc_path=__file__, debug=False, log_file=None, log_level=logging.DEBUG,
        no_compensation=True)
    log = logging.getLogger(__name__)
    # ==============================

    my_test(instruments, comp, frequencies, results_file,
            start_frequencies, stop_frequencies)

    print('\nScript done!')
```

A.2 Appendix 2 - Measurement script for Phase 2

Measurement script 2

```
import os
import sys
import csv
import time
import logging
import vf_core
from pathlib import Path
import matplotlib as plt
import numpy as np

# import pandas as pd

# Where should we save results
if sys.platform == ’win32’:
    results_path = r’C:\test_results’
else:
    results_path = Path(r’~/test_results’).expanduser()

results_file = r’results.csv’
# Create directories if they don’t exist
if not os.path.exists(results_path):
    os.makedirs(results_path)

# Specify the frequencies to use in our test function


# CSV results file header row
title = [’frequencies [Hz]’]

# Result handling. Make a timestamp to use for file name
ts = time.strftime(’%Y-%m-%d_%H_%M_%S’)
results_file = ’{}_{}’.format(ts, results_file)
results_file = os.path.join(results_path, results_file)


# Read the file
file_path = r’C:\Users\Desktop\resultsTestCWFreq.csv’
frequencies = []
BW=[]
Cal_row=[]
RBW=[]
with open(file_path, ’r’, newline=’’)as file:
    reader = csv.reader(file, delimiter=’,’)
    # Skip first row of the document
    next(reader)
    for row in reader
        frequencies.append(row[2].replace(",", "."))
        BW.append(row[0])
        Cal_row.append(row[10])
```

```
        RBW.append(row[8])

def my_test(instruments, comp, freqs=frequencies,Cal_row=
   Cal_row,
            results_file=None,):
    """
    Raises
    ------
    EnvironmentError
        If instrument is missing.

    """
    try:
        # Fix for code completion to work for instruments in
        #Spyder
        #from framework.station.autogen_instruments import
           Instruments
        instruments = Instruments()

        # Open file for appending results data, create only
        #if results_file
        # exists
        wr = None
        if results_file is not None:
            fh = open(results_file, 'a', newline='')
            wr = csv.writer(fh, delimiter=',')
            # Create a csv writer object for writing CSV

        total_results = []  # Keep measured data
        # Add header to file only if it does not exist
        total_results.insert(0, title)
        measurede_freq = []
        p1db_pout_total = []

        # assigned the instrument
        uxan9042b = instruments.keysight_uxan9042b
        vxg9848c = instruments.keysight_vxg9848c
        uxan9042b.setTimeout(800000)

        # IQ mode with Cal Group 2 same setup as in Cal
        uxan9042b.saMode()
        uxan9042b.output.Calgroup(1)
        uxan9042b.amplitude.setSignalPath('FULL')

        for i in range(len(freqs)):
            results = []
            freq = freqs[i]

            #Set frequency for vxg
            vxg9848c.Setfreq(freq)
            vxg9848c.SetBW('FR2 100')
            vxg9848c.GenerateWave(1)
            vxg9848c.iqmod(0)
            # Set the RBW and attenuation for UXA should be
               same
            #as phase 1
```

```
            uxan9042b.frequency.setFrequencyCenter(freqs[i])
            # Set the RBW and attenuation
            # for different settings for frequency
            # above 80G and below)
            uxan9042b.frequency.setResolutionBw(RBW[i])
            if freq < 50e9:
                uxan9042b.amplitude.setExtendAttenuationValue(4)
                uxan9042b.amplitude.setAttenuationValue(4)
            else:
                uxan9042b.amplitude.setSoftwarePreselection(1)
                if freq > 80e9:
                    uxan9042b.amplitude.setSoftwarePreselection(0)
                    uxan9042b.amplitude.setExtendAttenuationValue(10)
                    uxan9042b.amplitude.setAttenuationValue(10)
            time.sleep(5)

            #Select and apply the corresponding calibration row
            uxan9042b.SelCalrow(Cal_row[i])
            uxan9042b.output.apply_Cal_row(1, 'ON')
            appliedcheck = uxan9042b.output.check_calstate(1)
            while appliedcheck == '---':
                time.sleep(5)
                appliedcheck = uxan9042b.output.check_calstate(1)

            #A temp empty array for keep the output power in the loop
            pouttemp=[]
            pintemp=[]
            delta=[]
            # for analyzer the P1dB
            # Change the output power from the vxg in range -30dBm to
            # 20dBm with 0.5 stepsize
            for p in range(-30, 21, 0.5):
                # set the ouput amplitude turn on the output
                #without modulation
                vxg9848c.SetAmplitude(p)
                vxg9848c.SetRFOutput('ON')

                #Change to trace mode for a stable value in UXA
                uxan9042b.trace.SetAverage('ON')
                uxan9042b.trace.RestartAverage('ON')
                time.sleep(10)
                #record value
                uxan9042b.marker.setMarkerXvalue(1, freq)
                pout1 = uxan9042b.marker.getMarkerYvalue(1)
                pouttemp.extend(pout1)
                pintemp.extend(p)
```

```
            # end of p loop

            # Linear fit (in the linear region)
            # Assuming the linear region is between Pin =
            # -30 dBm and Pin = 20 dBm
            fit_region = pintemp <= 20
            coefficients = np.polyfit(pintemp[fit_region],
                        pouttemp[fit_region], 1)
            linear_fit = np.poly1d(coefficients)

            # Extend the linear fit line for visualization
            fit_line = linear_fit(pintemp)

            # Find the P1dB point
            # Calculate the difference between the linear fit and
            # the actual Pout
            delta = fit_line - pouttemp
            # The point where this difference is closest to
            # -1dB is the P1dB point
            p1db_index = np.argmin(np.abs(delta - 1))
            p1db_pin = pintemp[p1db_index]
            p1db_pout = pouttemp[p1db_index]

            # Plotting
            plt.figure(figsize=(10, 6))
            plt.plot(pintemp, pouttemp, 'o-',
                    label='Measured Pout')
            plt.plot(pintemp, fit_line, 'r--', label='Linear Fit')
            plt.plot(p1db_pin, p1db_pout, 'ko',
                    label='P1dB Point')
            plt.axhline(y=p1db_pout, color='k', linestyle=':',
                    label='1dB Compression Level')
            plt.title('P1dB Compression Point at %s' %freq)
            plt.xlabel('Pin (dBm)')
            plt.ylabel('Pout (dBm)')
            plt.legend()
            plt.grid(True)
            plt.show()

            #insert first column with current freq
            pouttemp.insert(0,'freq, pout,')
            pintemp.insert(0,'freq, pin')
            delta.insert(0,'' ', delta')
            # append row to keep the pin and pout values
            results.append(pintemp, pouttemp, delta,
                           ['P1dB', p1db_pin, p1db_pout])
            measurede_freq.append(freq)
            p1db_pout_total.append(p1db_pout)
            total_results.append(results)
            #turn of the Calrow for next measurement
            uxan9042b.output.apply_Cal_row(1, 'OFF')
            uxan9042b.output.set_Cali_Row(1, 'OFF')
            #10GHz jump
```

```
            i = +10
        # end of i loop

        #Plot the P1dB for tested signals
        plt.figure(figsize=(10, 6))
        plt.plot(measurede_freq, p1db_pout_total, 'ko',
                label='P1dB Points')
        plt.title('P1dB Compression Points over
                  the frequency range')
        plt.xlabel('Freuency (Hz)')
        plt.ylabel('Pout (dBm)')
        plt.legend()
        plt.grid(True)
        plt.show()

        time.sleep(10)
        print(total_results)

        # Write results to file
        if wr is not None:
            for row in total_results:
                wr.writerow(row)
            fh.close()  # Close results file

        return total_results
        # Return to caller for post processing

    except IOError:
        log.exception(f'Could not open results file: {
           results_file}, '
                      'for writing, error: ')
    except Exception as e:
        log.exception(e)


if __name__ == '__main__':
    # =======================================
    # Setup Validation Framework
    # =======================================
    # Set debug=False in vf_core.start_framework below to
       enable
    #instrument
    # communication, otherwise fake calls are used (for
    # testing/demo)
    instruments, comp = vf_core.start_framework(
        sc_path=__file__, debug=False, log_file=None,
        log_level=logging.DEBUG,
        no_compensation=True)
    log = logging.getLogger(__name__)
    # ========================================

    my_test(instruments, comp, frequencies, Cal_row,
       results_file)

    print('\nScript done!')
```

B

Appendix

B.1 Figures for phase 2

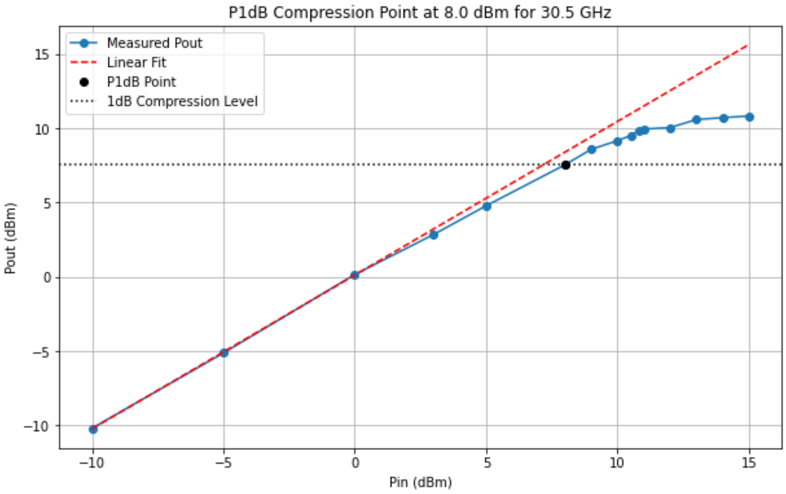


Figure B.1: Compression Points found at 30.5 GHz

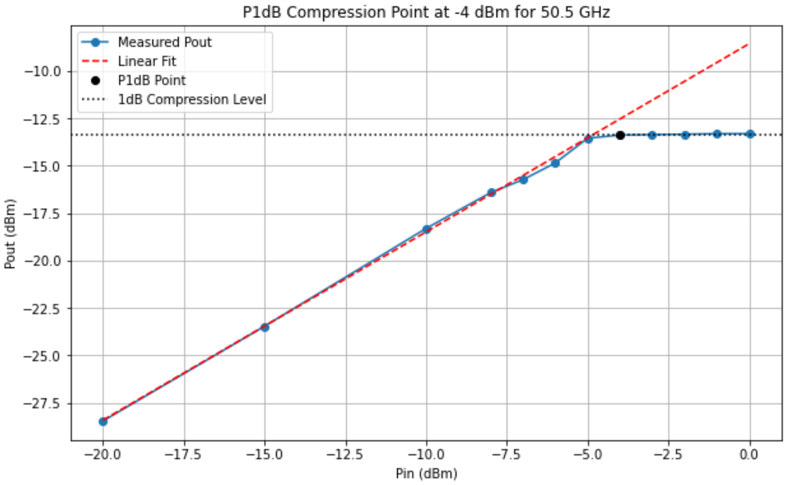


Figure B.2: Compression Points found at 50.5 GHz

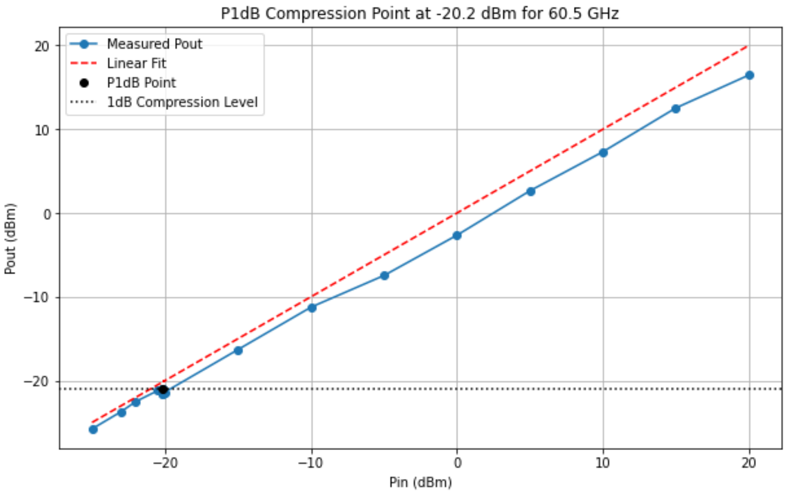


Figure B.3: Compression Points found at 60.5 GHz

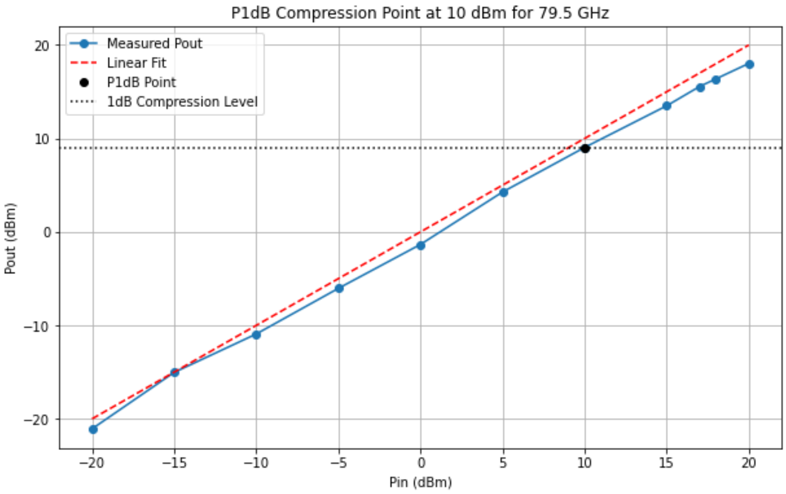


Figure B.4: Compression Points found at 79.5 GHz

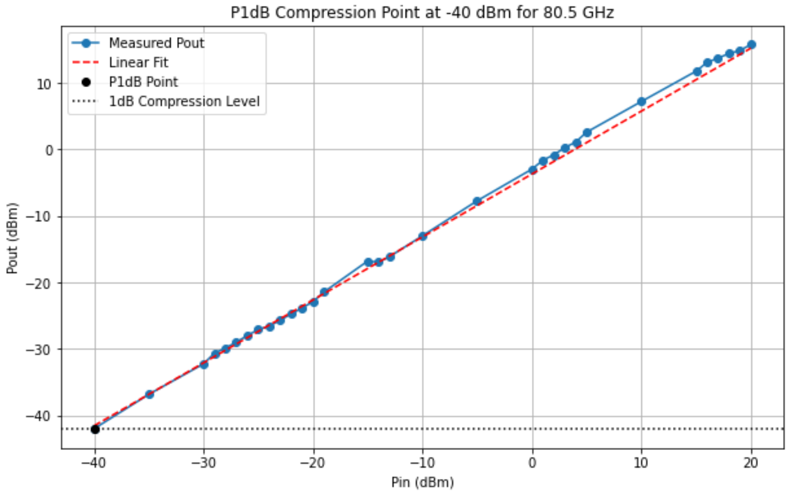


Figure B.5: Compression Points found at 80.5 GHz

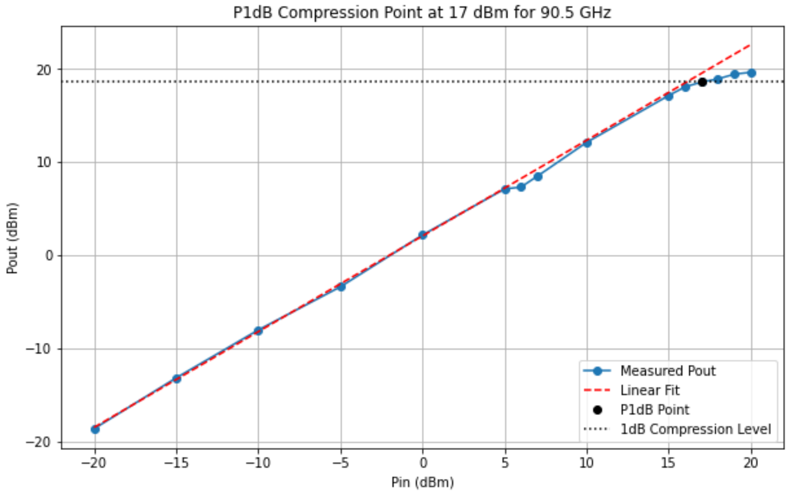


Figure B.6: Compression Points found at 90.5 GHz

Department of Electrical Engineering
Chalmers University of Technology
SE-412 96 Gothenburg Sweden
Telephone +46 31 772 1000

Department of EIT
Lund University, LTH
Box 118 SE-221 00 Lund Sweden
Telephone +46 46 222 00 00

CHALMERS
UNIVERSITY OF TECHNOLOGY

LUND
UNIVERSITY