

SDR for Metrology at Microwave and Submillimeter-Wave Frequencies

A
Dissertation
Presented to
the faculty of the School of Engineering and Applied Science
University of Virginia

in partial fulfillment
of the requirements for the degree

Doctor of Philosophy

by

Dustin Widmann

May 2025

APPROVAL SHEET

This
Dissertation
is submitted in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy

Author: Dustin Widmann

This Dissertation has been read and approved by the examining committee:

Advisor: Robert Weikle

Advisor:

Committee Member: Scott Barker

Committee Member: Arthur Lichtenberger

Committee Member: Neal Magee

Committee Member: Harry Powell

Committee Member: Michael Cyberey

Committee Member: Christopher Moore

Accepted for the School of Engineering and Applied Science:



Jennifer L. West, School of Engineering and Applied Science

May 2025

Abstract

This thesis explores the practicality of using commercial off-the-shelf (COTS) software-defined radio (SDR) hardware to implement metrology equipment, specifically a vector network analyzer (VNA). Although SDRs are flexible general purpose instruments, their performance typically falls short of dedicated measurement instruments. A software framework has been developed to facilitate high performance collection and processing of SDR data, interface with scientific libraries, and interchangeably connect to various radio frequency (RF) instruments. A low frequency prototype was constructed and thoroughly tested to assess its performance metrics and to identify limitations when using SDR as the underlying backend, enabling future projects to avoid pitfalls and optimize performance. Furthermore, SDR has been used as a back-end for submillimeter wave (sub-mmWave) experiments utilizing VNA frequency extenders. This work not only demonstrates the current feasibility of using SDRs as metrology instruments, particularly VNAs, but also outlines strategies for future advancements. To my knowledge, this is the most detailed characterization of a SDR-based VNA performed yet and is the first to describe using a SDR as a backend for sub-mmWave VNA measurements.

To my parents.

Acknowledgements

This work was made possible by the tremendous amount of support, mentorship, and encouragement I have received along the way.

I am deeply grateful to the University of Virginia, its School of Engineering and Applied Science, and the Electrical and Computer Engineering department. Its excellent professors and staff, thought-out classes and research opportunities nurtured my curiosity and forced me to adapt by demanding excellence.

I thank my research group for their ideas, debates, open ears, and even the occasional moment of clarity, especially in the more chaotic times. Without the many technical debates with the likes of Eller, Scott, Cyb, and Richard, I would probably still be stuck in the mud. Their *questions* sometimes provided the best insight I could ask for.

I am especially grateful to my advisor, Bobby Weikle, whose guidance has frequently guided me into useful ideas and struck a balance between being demanding and patient. Without his confidence in my work it may never have come to fruition.

To my committee, thank you for your time, critique, and expectations. This document is much better because of your influence.

To my friends and family, thank you for helping me stay grounded, reminding me why what I am doing matters, and tolerating that I am going AWOL, whether for work or for recovery. The patient positivity of my parents, long walks and talks with Chris, and the love and encouragement of Erin have helped keep me productive and sane.

Finally, I gratefully acknowledge the sponsorship and support of Azbil North America and the National Ground Intelligence Center. In addition, I would like to thank Virginia Diodes for loaning me equipment and devices. Without these contributions, this research would not have been possible.

Every person and institution mentioned here has left its own mark. No impact, however subtle, has gone unnoticed.

Table of Contents

Abstract	iii
Dedication	iv
Acknowledgements	v
List of Tables	ix
List of Figures	x
Acronyms	xv
Glossary	xviii
1 Introduction	1
2 Radio Frontend Characterization: Receivers, Transmitters, and Measurement	
Reliability	4
2.1 Brief Overview of Radio Performance Metrics	6
2.2 Noise and Noise Figure	8
2.3 How to Measure a Receiver's Noise Figure	9
2.4 Characterization of the X310 + SBX-120 Radio	9
2.4.1 What does the "rx_gain" setting do?	10
2.4.2 What does the "sampling rate" setting do?	10
2.4.3 Determine the maximum input level.	11
2.4.4 Characterizing the Output Level and Distortion of an SDR Transmitter	14
2.4.5 Determine the Noise Figure and Noise Variance	17
2.4.6 Measurement Stability	25

2.4.7	Measurement Repeatability	30
2.5	Chapter Conclusion	32
3	DSP Methods for Signal Analysis	34
3.1	Sampling and Frequency Transform Fundamentals	34
3.1.1	Fourier Analysis and Signal Representation	34
3.1.2	Sampling and the Discrete-Time Fourier Transform	35
3.1.3	The Discrete Fourier Transform and the Fast Fourier Transform	35
3.2	Spectral Leakage and Windowing	36
3.3	Measuring A Tone In Noise	42
3.3.1	Time domain analyses - Lock-in Amplifier	42
3.3.2	Vector Lock-In Detection - Extracting Phase Information	43
3.3.3	Amplitude Extraction and Measurement Noise Floor	43
3.3.4	Comparing Time-Domain vs. Frequency-Domain approaches	45
3.4	Implementation for a SDR, VNA	46
3.4.1	Time-Domain method implementation details	46
3.4.2	Frequency-Domain method - A naive implementation and why it still worked	47
3.4.3	Frequency-Domain method - How to do it well	48
3.4.4	DSP Approach to Measurement	50
3.5	Chapter Conclusion	53
4	VNA Background	55
4.1	Two-Port Networks and Network Parameters	55
4.2	Signal Flow Graphs	58
4.3	General VNA topology	59
4.4	Measurement Errors	61
4.5	Error Modeling and Calibration	66
4.6	Chapter Conclusion	69
5	SDR-based VNA Prototype	70
5.1	Radio and Frontend Hardware	70
5.2	Brief History of This Project's SDR-based VNA Software	76
5.3	Early experiments, TwinRX quirks	79
5.4	The UBX-160 radio, quirks, and power leveling	83

5.5	Later experiments with the SBX-120 radio	87
5.6	SDR Power, Sampling Rate, Measurement, and Sweep Approaches	89
5.7	Techniques to Compensate for Erroneous Data	92
5.8	VNA Residual Error Analysis	93
5.9	Accuracy with respect to Return Loss	99
5.10	VNA Residual Error Comparison	105
5.11	Chapter Conclusion	107
6	Frequency Extension	108
6.1	Overview	108
6.2	Interfacing with VNA Frequency Extension Modules	110
6.3	Upconverter Communications Experiment	111
6.4	SINAD-Based Characterization of Performance	114
6.5	SDR-based VNA at Sub-millimeter Wave Frequencies	117
6.6	VNA Measurements	118
6.7	Chapter Summary	124
7	Conclusions and Suggestions for Future Work	126
	References	129
A	Software Architecture	132
A.1	SDRVNA	132
A.2	RMS (Radio Metrology Software)	134
A.3	AERIS	136
A.4	Chapter Summary	138

List of Tables

3.1	Window function performance summary, data from [18]	40
4.1	Reflectometer error case with absorptive DUT	62
4.2	Reflectometer error case with reflective DUT	62
4.3	Reflectometer error case with reflective DUT (cont)	63
5.1	Hardware used in SDR-based VNA prototype	73
5.2	Residual error analysis results and comparisons	105
6.1	Power levels at various points in the transmit chain	112
6.2	Equipment used in the upconversion and VNA extension chain	113
6.3	SINAD measurements at different stages of the measurement chain	116
A.1	Feature comparison between AERIS and other frameworks	137

List of Figures

2.1	Superheterodyne Receiver Architecture[6]	5
2.2	TX Input Power Needed to Saturate ADC as a function of rx_gain setting and center frequency	12
2.3	Relative full scale received power as a function of input power; compared with manufacturer-measured gain level (extracted from published plots at rx_gain = 32.5 dB); Both as a function of center frequency)[13]	13
2.4	UBX-160 transmitter power [dBm] as a function of tx_gain and center frequency . .	14
2.5	UBX-160 third-order intercept point (IP3) as a function of frequency, for select values of tx gain. Extracted from manufacturer published plots [13]	15
2.6	SBX-120 output power as a function of frequency at select values of tx_gain. Data extracted from manufacturer published plots. [14]	15
2.7	SBX-120 IP3 as a function of frequency at select values of tx_gain	16
2.8	Noise figure (NF) as a function of center frequency, with sampling rate of 100 MHz .	18
2.9	NF as a function of center frequency, at select gains and sample rates	19
2.10	NF as a function of rx gain, at select frequencies with sampling rate of 100 MHz . .	20
2.11	NF as a function of rx gain, at selected sampling rates and frequencies	21
2.12	Standard deviation of the estimated power spectral density (PSD)as a function of rx_gain, median across frequency values,and normalized to $N = 1$ samples	22
2.13	Standard deviation of the estimated PSD as a function of rx_gain, at selected sampling rates and center frequencies, normalized to $N = 1$ samples.	23
2.14	Standard deviation of the estimated PSDas a function of center frequency, median taken across rx_gain values, normalized to $N = 1$ samples.	24
2.15	Standard deviation of the estimated PSDas a function of center frequency, at selected sampling rates and rx_gain values, normalized to $N = 1$ samples	25

2.16 Short-time fourier transform (STFT) derived a-wave (top) and b-wave (bottom) magnitudes.	26
2.17 STFT magnitude ratio of reference (a-wave) and measured (b-wave) channels across time at select frequencies	27
2.18 STFT phase difference of reference (a-wave) and measured (b-wave) channels across time at select frequencies	28
2.19 Interpolated Phase Noise PSD as measured at select frequencies	29
2.20 Normalized magnitude difference between ref and meas channel repeatability violin plot of SDR measured with a reflective load	30
2.21 Normalized phase difference between ref and meas channel repeatability violin plot of SDR measured with a reflective load	31
3.1 Collecting a finite number of samples is equivalent to multiplying by a rectangle function in the time domain	36
3.2 Small alignment errors between signal frequency and sinc frequency result in large spectral leakage	37
3.3 Spectral leakage for a Rectangular window with several frequency offsets	38
3.4 Spectral leakage for a Hamming window with several frequency offsets	39
3.5 Spectral leakage for a Hann window with several frequency offsets	39
3.6 Spectral leakage for a Blackman-Harris window with several frequency offsets	40
3.7 Goertzel algorithm performance Vs fast Fourier transform (FFT) for one bin	49
3.8 Early, naive approach to processing the IQ data	50
3.9 Example IQ Data	51
3.10 Revised DSP Block Diagram	51
3.11 Multiplying by complex exponential produces frequency shift	52
3.12 Phase Difference Sometimes Takes Significant Time to Settle	53
4.1 A two port network with transmission lines at the ports. Figure copied from [22]	56
4.2 General Two-Port Signal Flow Graph. Figure from [9]	58
4.3 Dual Directional Coupler Diagram. Figure from [25]	59
4.4 Switch-based VNA with Two Receivers. Figure from [22]	60
4.5 Directional couplers with imperfect coupling	61
4.6 Return loss measurement error as a function of DUT return loss and coupler directivity. Figure from [26]	64

4.7	Forward Measurement Error as a function of return loss and coupler directivity. Figure from [26]	65
4.8	Directivity and Source Match Error as a function of DUT return loss, source port match, and coupler directivity. Figure from [27]	66
4.9	VNA One Port Error Model – Error box diagram and SFG. Figure from [30]	67
4.10	VNA One Port Error Model - Linear System of Equations. Figure from [30]	68
4.11	12-Term Forward Error Model SFG. Figure from [30]	68
4.12	12-Term Reverse Error Model SFG. Figure from [30]	69
5.1	Low Frequency SDRVNA Prototype with COTS hardware, early revision	70
5.2	Block diagram of low frequency VNA prototype hardware	71
5.3	VNA frontend fixture design, including clamp-like structures on the corners to prevent unintended cable movement	74
5.4	Low Frequency SDR-based VNA prototype with COTS hardware, later revision	74
5.5	Low frequency \gls{sdr}-based VNA assembly and wiring diagram	75
5.6	RF switch controller wiring diagram	76
5.7	RF switch controller	77
5.8	Frontend Nominal S-Parameters	78
5.9	Uncalibrated phase-difference between open-circuit and short-circuit calibration standards, measured with UBX-160 transmitter, and TwinRX receivers	79
5.10	Uncalibrated magnitude difference between reflective and absorptive calibration standards measured with UBX-160 transmitter and TwinRX receivers	80
5.11	TwinRX intermediate frequency (IF) deviation from expected value	81
5.12	TwinRX a-wave and b-wave measured magnitudes of a short-circuit calibration standard	81
5.13	TwinRX measured phase-difference between open-circuit and short-circuit calibration standards, without calibration, after upgrading to UHD 4	82
5.14	Standard deviation of magnitude for TwinRX measurements of a reflective load – y-scale is 0-2dB, x-scale is 0-6 GHz	83
5.15	Standard deviation of phase for TwinRX measurements of reflective load – y-scale is 0-5 degrees, x-scale is 0-6 GHz	83
5.16	UBX-160 measurement of magnitude of matched load calibration standard, without calibration applied	84
5.17	UBX-160 power output as a function of frequency and tx_gain	85

5.18	UBX-160 measurement of matched load calibration standard, without calibration applied, with (red) and without (blue) transmit power leveling applied.	86
5.19	UBX-160 Radio – phase difference between open-circuit and short-circuit calibration standards, without calibration applied – Phase (degrees) plotted with respect to frequency in 100s of MHz	86
5.20	UBX-160 two-port measurement of 10dB attenuator	87
5.21	SBX-120 two-port measurement of 10 dB attenuator	88
5.22	SBX-120 measurement of delayed short, compared with the same measurement by a commercial VNA	88
5.23	Example measurement utilising the "a-wave masking" approach described above . . .	93
5.24	Shorted airline block diagram. Figure from [34]	94
5.25	Measurement of a precision airline terminated with a matched load calibration standard. This is used to calculate the effective directivity figure of merit (FOM)	95
5.26	Measurement of a precision airline terminated with a short-circuit calibration standard, with the attenuation and phase of the airline de-embedded and the residual directivity subtracted. This is used to calculate the effective source match FOM . .	96
5.27	Measurement of a precision airline terminated with a short-circuit calibration standard, after de-embedding the airline, subtracting the residual directivity, and also subtracting the residual source match. This is used to determine the effective reflection tracking FOM	97
5.28	Measurement of a precision airline connected between the two measurement ports, with this plot displaying the return loss looking into the device from port 1, after de-embedding the airline model, subtracting the residual directivity and residual source match. This figure is used to determine the effective load match FOM	98
5.29	Measurement of a precision airline connected between the two measurement ports, with this plot displaying the insertion loss between port 1 and port 2, after de-embedding the airline model, subtracting the residual directivity and residual source match and load match errors. This figure is used to determine the effective transmission tracking FOM	99
5.30	Return Loss measurement error as a function of directivity and return loss.	100
5.31	Projected return loss measurement error of the SDR-based VNA COTS frontend as a function of return loss at several frequencies	101

5.32	Predicted return loss limit as a function of frequency for different values of return loss measurement error	102
5.33	One-port measurement by the SDR-based VNA of a 5 dB attenuator terminated with a short-circuit.	103
5.34	One-port measurement by the SDR-based VNA of a 10 dB attenuator terminated with a short-circuit.	103
5.35	One-port measurement by the SDR-based VNA of a 15 dB attenuator terminated with a short-circuit.	104
5.36	One-port measurement by the SDR-based VNA of a 20 dB attenuator terminated with a short-circuit.	105
6.1	VNA frequency extender block diagram. Figure from [41]	110
6.2	Equipment setup for SDR to VNAX upconverter	111
6.3	SDR with VNA extender experimental setup block diagram	112
6.4	Signal power and bandwidth at each stage	113
6.5	GNU Radio program for measuring SINAD	114
6.6	Example SINAD plot	115
6.7	Block diagram for SDR-backend VNA at sub-mmW frequencies	117
6.8	1-inch thru magnitude	119
6.9	1-inch thru phase in degrees	119
6.10	1-inch thru phase in degrees, after de-embedding 1.18-inches of phase	120
6.11	1-inch thru phase difference between ZVA and SDR	120
6.12	Attenuator Return Loss Magnitude dB	121
6.13	Filter Return Loss, Magnitude dB	122
6.14	Attenuator Insertion Loss Magnitude dB	123
6.15	Filter Insertion Loss Magnitude dB	123
A.1	Libraries and plugin-system block diagram	135

Acronyms

ADC analog-to-digital converter. 1, 5–8, 11, 16, 19, 32, 89, 108, 111, 127

API application programming interface. 10, 134

COTS commercial off-the-shelf. iii, xii, xiii, 1, 2, 6, 70, 74, 100, 101, 105–107, 111, 124, 126, 127

DAC digital-to-analog converter. 1, 108, 111

DC direct current. 48, 52, 79, 115

DFT discrete Fourier transform. 35, 36, 47–49, 52, 54, 90

DR dynamic range. 7, 126

DSP digital signal processing. 2, 3, 12, 34, 46, 54, 118, 126, 132, 133

DTFT discrete-time Fourier transform. 35, 36, 54

DUT device under test. 9, 26, 67, 79

ENBW equivalent noise bandwidth. 37, 41, 44, 45, 54

ENR excess noise ratio. 9, 17

FFT fast Fourier transform. xi, 17, 26, 29, 36, 44–46, 48, 49, 52, 54

FM frequency modulation. 114

FOM figure of merit. xiii, 3, 6, 7, 32, 93, 95–99

FPGA field-programmable gate array. 10, 80, 82, 83, 91, 107, 133

HPF high-pass filter. 46

I/Q in-phase and quadrature. 5, 7, 26, 43, 46, 50, 76, 77

IBW integration bandwidth. 44, 46, 71, 89, 90, 109, 124

IC integrated circuit. 108, 127

IF intermediate frequency. xii, 4, 5, 7, 26, 81, 90, 109, 110, 113, 114, 117, 118, 124

IP3 third-order intercept point. x, 7, 14–16

LIA lock-in amplifier. 42, 43, 45–48, 54, 91

LNA low-noise amplifier. 6, 10

LO local oscillator. 4–6, 17, 25, 31, 32, 43, 71, 110, 112, 113, 117, 124, 127

LPF low-pass filter. 42, 43, 47, 111

MDS minimum detectable signal. 6, 7

meas measured or b-wave. 46, 50, 52, 110

mmWave millimeter wave. 3, 108, 109, 124

NBFM narrowband frequency modulation. 114, 116

NEB noise equivalent bandwidth. 44, 45

NF noise figure. x, 6, 8–10, 16–21, 23, 25, 32, 46, 89, 124, 126

OCXO oven-controlled crystal oscillator. 25

P1dB 1-dB compression point. 7

PLL phase-locked loop. 31, 71, 107, 127

PSD power spectral density. x, xi, 8, 11, 22–25, 29

QAM quadrature amplitude modulation. 5, 116

RBW resolution bandwidth. 46

ref reference or a-wave. 46, 50, 52, 110

RF radio frequency. iii, xii, 4–6, 26, 32, 34, 45, 46, 71, 72, 75–77, 108–110, 117, 124, 133, 135

RFIC radio frequency integrated circuit. 1

RMS root mean square. 29

SDR software-defined radio. iii, ix, xi–xiv, 1–3, 6, 10, 11, 14, 17, 25, 30–32, 46, 70, 73, 74, 76, 83, 89, 90, 93, 99–101, 103–111, 114–118, 120, 124–128, 132, 133, 135, 138

SFG signal flow graph. 58, 69

SINAD signal-to-noise-and-distortion ratio. 114–116, 124

SNR signal-to-noise ratio. 8, 9

STFT short-time fourier transform. xi, 26–28

sub-mmWave submillimeter wave. iii, 3, 109, 116, 124–126

SWaP size, weight, and power. 6

THD total harmonic distortion. 7

VNA vector network analyzer. iii, ix, xii–xiv, 2, 3, 31, 34, 46, 48, 54, 67, 70, 72–75, 83, 89, 90, 93, 94, 99–101, 103–107, 109–112, 114, 117, 118, 124–127, 132–135, 138

VSA vector signal analyzer. 46

WBFM wideband frequency modulation. 116

Chapter 1

Introduction

SDRs have revolutionized radio over the past few decades, driven by advancements in computing power, data bus speeds, analog-to-digital converters (ADCs), digital-to-analog converters (DACs), compact radio frequency integrated circuits (RFICs) Initially conceived for communications engineering, SDRs integrate these components in a single device and have proven valuable for a variety of research applications extending beyond their traditional scope. Although they are very versatile, COTS SDRs still have limitations. This research aims to quantify some of these limitations and, where possible, to offer solutions.

The term SDR broadly encompasses any radio system that can be controlled or reconfigured through software. However, in common use, “SDR” refers more specifically to generic, general purpose, user-reconfigurable devices, and this is how I will apply the term in this thesis. These systems offer a gateway to a broad frequency spectrum and integrate the traditional analog functions of filtering, mixing, amplifying, and digitizing into a unified hardware package. They typically provide robust computer interfaces, prewritten firmware, and software libraries that facilitate interaction with minimal effort. However, the rapid evolution of SDR technology also leads to products that have a comparatively short lifespan before obsolescence. Creating a radio frequency device that incorporates these advanced features from scratch is, therefore, impractical without substantial resources.

One of the most significant benefits of general-purpose SDRs is their accessibility and adaptability. However, their adaptability is also their greatest weakness, because it means that they are inherently not optimized for any particular task. Therefore, more specialized instruments can generally outperform them. There are a range of possible selection criteria depending on the user's needs, including channel count, phase coherence, channel isolation, measurement repeatability, and system stability. The price of SDRs ranges dramatically from as little as \$30 to well above \$100,000 for extremely high-end systems. Open-source frameworks such as Pothos and GNU Radio provide easily reconfigurable digital signal processing chains that can accommodate a range of uses, especially for communications-centric tasks. Otherwise, it is possible to interface with radios by developing custom software limited only by the hardware capabilities and the developer's imagination, although this can be a time-consuming endeavor.

This project explores VNA measurements as a means to explore the limitations of SDR as a metrology instrument. The metrology focus and VNA measurements in particular stem from the desire to use it as back-ends for more complicated measurements like plasma imaging without resorting to large and expensive laboratory-grade instruments for the task. This thesis is focused on radio and VNA characterization; however, many of the principles will be carried over to other use cases. Chapter 3 describes the relevant digital signal processing (DSP) techniques, while Chapter 4 details the operational principles of VNAs. The former includes a refresher on sampling, windowing, frequency transforms, and the DSP implementation used in this project. The latter discusses scattering parameters, measurement error, calibration, and verification. The VNA application aims to delineate the limits of the SDR performance for metrology-centric applications, and address the dynamic range, phase coherence, measurement stability, repeatability, and more.

Recent literature, inside the past few years, has seen several implementations of SDR-based VNA applications. For example, Myriad RF published a project that allows a LimeSDR to perform one-port VNA measurements [1], although this implementation was particular for the LimeSDR and was not directly compatible with other platforms. Another research project [2] utilized an Analog Devices Pluto SDR with a COTS front-end implemented with MATLAB scripts and showed promising results. Furthermore, a Malaysian team demonstrated a SDR-based VNA with a BladeRF SDR with a custom VNA front end and achieved superior dynamic range for medical imaging [3]. Another group tackling medical imaging at Dartmouth [4] also achieved good performance and found

a novel way to phase lock multiple radios together utilizing a spare channel. More recently, another group created an SDR prototype utilizing a LimeSDR that seemed to perform well [5].

This research extends beyond these studies by meeting the following core objectives:

1. The procedure for developing a system prototype is described at length, including characterization of the SDR's performance (and, to some extent, quirks) (see Chapter 2), the relevant DSP for performing the measurements (see Chapter 3), VNA techniques (see Chapter 4), and a more complete characterization yielding the typical VNA FOMs, which are then compared with commercial instruments (see Chapter 5).
2. A robust software framework has been developed to facilitate maximum flexibility and high performance, integrate with existing toolkits, and enable control of laboratory instruments (see Chapter 5 and Appendix 1).
3. The SDR can be used as a backend to perform millimeter wave (mmWave) and sub-mmWave communications and metrology experiments (see Chapter 6).

Afterward, a brief discussion of the results and future work is provided (see Chapter 7).

Chapter 2

Radio Frontend Characterization: Receivers, Transmitters, and Measurement Reliability

Radio frequency (RF) receivers are fundamental to modern communications, RADAR, and measurement systems. Their role is to convert an input RF signal into a usable baseband (BB, 0 Hz, DC) or intermediate frequency (IF) signal. There are various RF receiver architectures, each offering trade-offs in performance, complexity, and cost. Some types of RF receivers are:

- Homodyne receivers—Primarily of historical interest, homodyne receivers "regenerate" the carrier of a modulated signal using a positive feedback amplifier that is phase-synchronized with the incoming signal. The amplifier mixes with the carrier, reproducing the baseband signal. It is rather sensitive to phase noise and is rarely used today.
- Heterodyne receivers— In heterodyne mixing, a local oscillator (LO) is mixed with the RF signal, resulting in a IF signal, and an additional stage mixes it down to the baseband. This architecture improves filtering, selectivity, and image rejection, but is more complicated and expensive.

- Direct conversion receivers– These receivers mix directly to baseband. They’re compact and cost-effective, but they’re prone to direct current (DC) offsets from self-mixing and LO leakage, and they also suffer from relatively poor image rejection.

Many modern receivers are direct conversion architectures, but high-performance systems still rely on heterodyne designs for superior filtering and interference rejection. Today, almost all radios use quadrature modulation, transmitting and receiving signals in an orthogonal pair of in-phase and quadrature (I/Q), enabling advanced modulation schemes like quadrature amplitude modulation (QAM). A heterodyne receiver can be seen in Figure 2.1 [6]. Observe that it mixes down in two steps and thus has two IF frequencies (a high IF and a low IF).

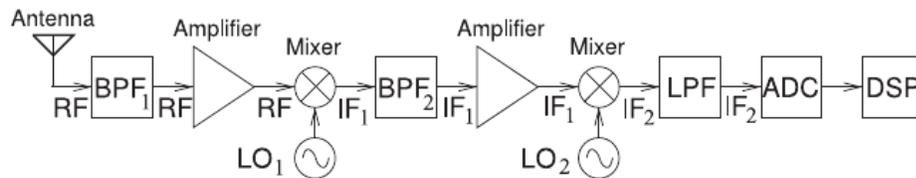


Figure 2.1: Superheterodyne Receiver Architecture[6]

RF receivers consist of several key subsystems:

- RF Frontend– Essentially, this includes all components up to the first mixer, often including the antenna (if present).
- Frequency Conversion (Mixing)– Downconverts the signal from RF to IF or baseband using mixers and LO.
- IF and Baseband Processing– Filtering and amplification at IF or baseband improves selectivity and gain.
- Analog-to-digital converter (ADC)– Digitizes the signal for digital signal processing.

They can contain numerous components:

- Antenna– Converts over-the-air electromagnetic waves into circuit signals. It also acts as a filter.
- Preselector filters– bandpass filters that reject out-of-band signals and noise.

- Protection Circuits/Attenuators– Prevent damage from strong signals.
- Low-noise amplifier (LNA)– Critical for sensitivity, as it sets the noise floor while also providing gain.
- Frequency Mixer(s): non-linear devices optimized for frequency translation (may introduce gain or loss).
- LO– Generate signals for heterodyning (mixing).
- Image Rejection Filters – Remove image frequencies before mixing.
- Anti-aliasing filters– suppress out-of-band signals before the ADC to prevent aliasing.

A receiver is considered software-defined radio (SDR) if it is reconfigurable via software. SDRs minimize analog processing, digitizing signals early and performing processing in software. Most modern radios incorporate SDR principles, but commercial off-the-shelf (COTS) SDRs refer specifically to fully user-reconfigurable systems. The design of the radio receiver is driven by trade-offs, including sensitivity, dynamic range, linearity, and noise performance, as well as size, weight, and power (SWaP), and cost. The following sections will provide a brief introduction to RF receiver performance metrics, especially noise, which should be considered in more detail as it is pertinent to this work.

2.1 Brief Overview of Radio Performance Metrics

In assessing the performance of transceivers, a variety of metrics are employed, each tailored to the radio’s specific operational requirements. The significance of these metrics varies depending on the application’s requirements. This section delineates several of the key figure of merits (FOMs), particularly those relevant to fundamental operations, which aligns with the focus of this work.

The receiver’s sensitivity primarily concerns its ability to discern between signal and noise. As such, the most important FOM is noise figure (NF), which quantifies the noise added by the receiver. The NF sets the limit on another FOM: the minimum detectable signal (MDS). The MDS is the weakest signal that the receiver can reliably detect under a specific configuration. [6]

Furthermore, the maximum input level marks the upper limit of signal strength that a receiver can handle before the risk of damage to the analog components or saturation of the ADC, which would

begin to “clip” the signal. This threshold underpins the receiver dynamic range (DR), defined as the ratio between the maximum input level and MDS. For digital receivers, the instantaneous DR is also relevant and is a function of the ADC’s resolution. It is the broadest range of input powers that can be observed simultaneously without reconfiguration and can be calculated as $6.02 \cdot N$ dB for an N-bit ADC converter.[7]

Linearity is assessed to ensure that output signals are proportional to input signals without distortion that could complicate signal processing. Key linearity metrics include third-order intercept point (IP3) and 1-dB compression point (P1dB). IP3 is measured with a two-tone test in which the signal level of the unwanted $2f_2 - f_1$ modulation product is compared to that of the desired tones (f_2 and f_1) with increasing input power; the former increases 3 dB for each 1 dB of the latter, so they can eventually be shown to be intercepted (though this is often outside the usable range of the device). It is an indicator of where the intermodulation products will produce significant interference and be of interest. The P1dB indicates the input level where the gain falls 1 dB short of what it would be with a linear amplifier and marks the beginning of the compression region of the device. Both of these FOMs are primarily related to amplifiers and affect both receivers and transmitters. [6]

There are also other sources of unwanted signal distortion. One such source of distortion is harmonic distortion, which is characterized by the total harmonic distortion (THD). The THD FOM is a ratio of the fundamental to the sum of all harmonics. Outside of multiplier design, the harmonics are usually just wasted power, although they can sometimes be put to creative use. Frequency mixers produce both a sum and a difference frequency. When downconverting, usually only one or the other is desired. In superheterodyne receivers, the other is easily filtered out in the IF stage. [6] However, direct conversion receivers have to take a different approach. They use image rejection mixers. The basic premise of image rejection mixers is that the input signal is split and phase shifted, e.g., by a 90° hybrid coupler, where both paths are mixed and then fed into a second 90° coupler. The desired signal interferes constructively (0° phase difference). In contrast, the image interferes destructively (180° phase difference). This cancellation is not perfect because the I/Q are not precisely 90° out of phase at all frequencies, and the amplitudes of the two paths are not exactly the same, either. The extent to which it is canceled is the FOM for image rejection, which is called the image rejection ratio. [8]

2.2 Noise and Noise Figure

Electronic measurements have various sources of noise, each arising from different physical mechanisms. Key noise sources include thermal (also known as Johnson-Nyquist noise), shot, flicker, phase, and quantization noise, as well as aperture jitter.

Thermal noise often sets the fundamental noise floor and arises from the random movement of charge carriers. $N_0 = k_B T$ gives its power spectral density (PSD). At room temperature (290 K), this corresponds to -174 dBm/Hz. In a bandwidth B , the total thermal noise power is $N = k_B T B$. Shot noise occurs in non-linear elements as a result of the discrete nature of charge carriers. Its current noise spectral density is $i_n^2 = 2qI[\text{A}^2/\text{Hz}]$, which can be converted to power, resulting in $P = 2qIR^2$. Flicker noise, or 1/f noise, dominates at low frequencies and arises from material defects and trap states in semiconductors. Its power spectral density scales as $S(f) \propto \frac{1}{f^\alpha}$. Phase noise is caused by random phase fluctuation about a carrier that originates from oscillators. It is described as $L(f) = 10 \log_{10} \left(\frac{P_{\text{SSB}}(f)}{P_{\text{carrier}}} \right)$. Quantization noise arises from the finite resolution of ADCs and can be approximated by $SQNR = 6.02N + 1.76\text{dB}$, where N is the bit resolution of the ADC. Aperture jitter is caused by timing uncertainty of the ADCs sampling clock. Its impact on the signal-to-noise ratio (SNR) is $SNR_{\text{jitter}} = 20 \log_{10} (2\pi f_{\text{in}} \sigma_t)$. [6]

When trying to characterize the noise of a system, it is apparent that a simplified way to characterize the noise added by a component is needed. Perhaps, by referring to each component's total added noise, there could be a way to sum the sources of noise across components to capture the total system noise. This characterization can be accomplished by converting the noise of each component to a noise factor (or equivalent noise temperature), which references the noise added by the component to the thermal noise floor. Then, by using Friis' gain equation, the effective system noise can be calculated as:

$$F_{\text{tot}} = F_1 + \frac{F_2 - 1}{G_1} + \frac{F_3 - 1}{G_1 G_2} + \dots$$

where F_n refers to the noise factor of component n , and G_n refers to the gain of component n . The NF can then be computed from the noise factor as:

$$NF = 10 \log_{10} (F)$$

The NF – a figure that captures the total noise incident in a measurement—can then be used to make design decisions that optimize the needed SNR for the measurements. [9]

2.3 How to Measure a Receiver’s Noise Figure

Depending on the application, there are a few different ways to measure NF. One option is to use a noise figure meter. It is a very straightforward way to measure NF accurately. However, this approach tends to require that the device under test (DUT) have an input and an output. Another way that is well suited to measuring devices like amplifiers is the “Gain Method.” While a receiver makes use of amplifiers, it is not an amplifier, so this is not the appropriate method. A third way is the Y-Factor method. This method entails measuring the DUT both with and without a source of excess noise. [10] This generalized approach is well suited to characterizing receivers, so it will be discussed at length.

The only extra needed to conduct a y-factor measurement is a source of excess noise. If you were feeling industrious, you could make your own by passing current through a resistor, but the different levels need to be calibrated, and you have to be able to connect it to your receiver’s input port. As such, it is easier to buy a calibrated, connectorized noise source. Commercial noise sources generally specify an excess noise ratio (ENR) in dB. They are commonly available with 15 dB ENR or 6 dB ENR. If the NF you are trying to measure is less than ~15dB, using a noise source of 6 dB ENR will provide better accuracy. [11]

The technique is fairly simple: the device is first measured with the noise source off (or with the port terminated) and then measured again with the noise source powered on. From these data, you can calculate the noise figure as follows: [12] $Y_{dB} = P_{hotdB} - P_{colddB}$ $Y = 10^{\frac{Y_{dB}}{10}}$ $ENR = 10^{\frac{ENR_{dB}}{10}}$
 $NF = 10 \cdot \log_{10}\left(\frac{ENR}{Y-1}\right)$

2.4 Characterization of the X310 + SBX-120 Radio

Several experiments were performed to characterize this transceiver, which has been used throughout most of this work. Similar steps could be followed to characterize a different radio. Although the

measurements themselves are simple, it will take a moment to review the data and make sense of them.

2.4.1 What does the "rx_gain" setting do?

This radio, like many other SDRs, has an application programming interface (API) setting for the `rx_gain`. The SBX-120 allows for adjustments from 0.0 dB to 37.5 dB. If you were to adjust the gain of an actual amplifier, e.g. by adjusting the bias, the results would likely be nonlinear or may adjust differently over wide bandwidths. In order to obtain relatively flat and consistent gain adjustments, many receivers, including this one, adjust an attenuator instead. If you look at the figure in the previous section, you will see that the SBX-120's attenuator is placed after the first LNA. The attenuator follows the amplifier because if the attenuator came first, it would kill the noise figure of the system. Why then does the attenuator come before the second amplifier, instead of after, which would reduce its NF? This configuration provides improved dynamic range, particularly considering the case of strong input signals. Observe that the attenuation range of the digital step attenuator used in the SBX-120 is 0 to 31.5 dB. From where then does the other 6 dB of `rx_gain` come? The additional gain will be discussed later when considering the results of the measurements.

2.4.2 What does the "sampling rate" setting do?

The X310 actually uses only one sampling clock rate, 200 MHz (not entirely true; a second master clock, 184 MHz, is also available, which is a multiple of the optimal rate for cellular reception). Rather than utilizing multiple clocks, it instead performs decimation in the digital domain (e.g., in the field-programmable gate array (FPGA)), so if N samples are captured at sampling rate $f_{s_{\text{target}}}$, then the radio actually captures $N \cdot \frac{f_{s_{\text{master}}}}{f_{s_{\text{target}}}}$ samples at $f_{s_{\text{master}}}$ sample rate and then performs decimation in the digital domain.

2.4.3 Determine the maximum input level.

Determining the maximum input level to the radio is valuable. The receiver uses amplifiers, but it is not one itself. As such, the goal is not to look for a loss of linearity (e.g. compression, intermodulation distortion), but to find the input level required to saturate ADC.

An experiment was conducted to characterize the maximum input level (in dBm) as a function of center frequency and receiver gain. A minimal GNURadio flowgraph was developed to allow manual adjustment of the center frequency and receiver gain, while displaying the PSD using a Blackman-Harris window. A signal generator was configured to produce a single-tone output, with the tone frequency selected to fall within a spectral bin centered in the SDR's frequency range to minimize leakage. For each configuration of center frequency and gain, the input amplitude was initially set to a low level and gradually increased until signs of ADC saturation were observed. Saturation was identified by the appearance of spurious periodic signals throughout the spectrum. The maximum input to the first component in the amplification chain is specified as +13 dBm; To avoid damage, this limit was not exceeded. The measured results are presented in the figure below. Lower gain settings correspond to proportionally lower output levels and exhibit similar trends. Notably, the input level required to saturate the ADC increases with center frequency. Additionally, increasing the gain setting by approximately 2.5 dB results in an upward shift of the corresponding measurement curve by a similar amount.

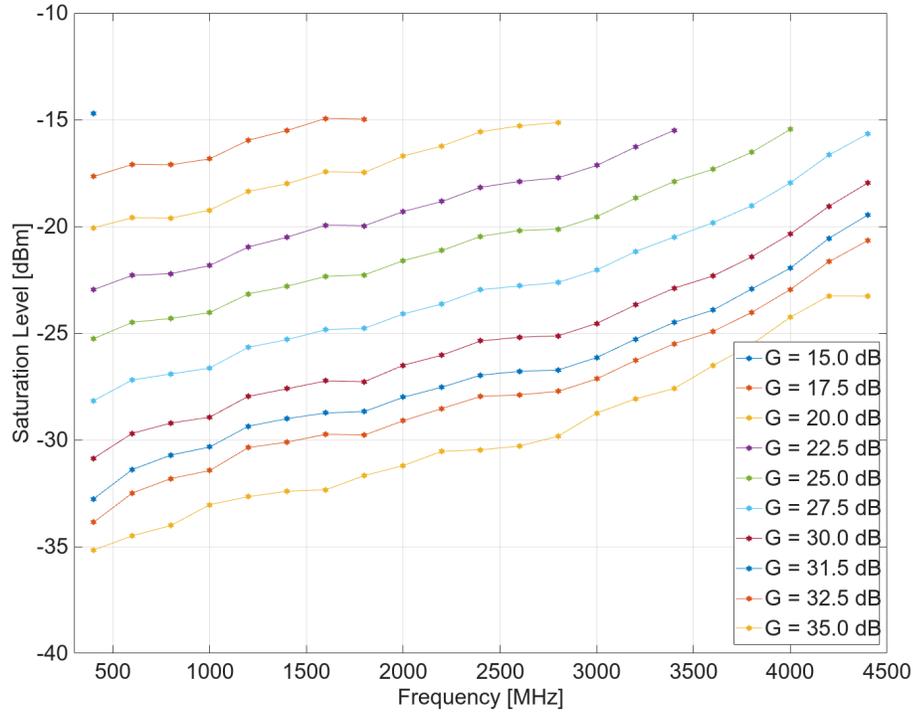


Figure 2.2: TX Input Power Needed to Saturate ADC as a function of rx_gain setting and center frequency

The results shown in Figure 2.2 indicate that the radio's digital signal processing (DSP) does not apply implicit compensation for the configured rx_gain. When the receiver gain is added to the measured values, the resulting curves align closely across all gain settings, with only minor variance. This suggests that automatic normalization is not applied in the signal chain. To illustrate the discrepancy between the measured signal level and the actual input level (in dBm), Figure 2.3 presents the deviation of the full-scale value received from the input power level as a function of frequency, averaged across gain settings where appropriate to reduce variance. Manufacturer-provided performance data, extracted from published plots, is also included for comparison (denoted "Ettus Data"). This reference curve represents the specified total gain at an rx_gain setting of 32.5 dB, offset by the configured gain (i.e., 32.5 dB was subtracted to make this curve). Although an offset of approximately 2–3 dB is observed, the general trend is consistent with the measured data. These results demonstrate that achieving a frequency-independent measurement requires compensating for frontend gain/loss variation, as well as explicitly subtracting the configured receiver gain from measured magnitudes.

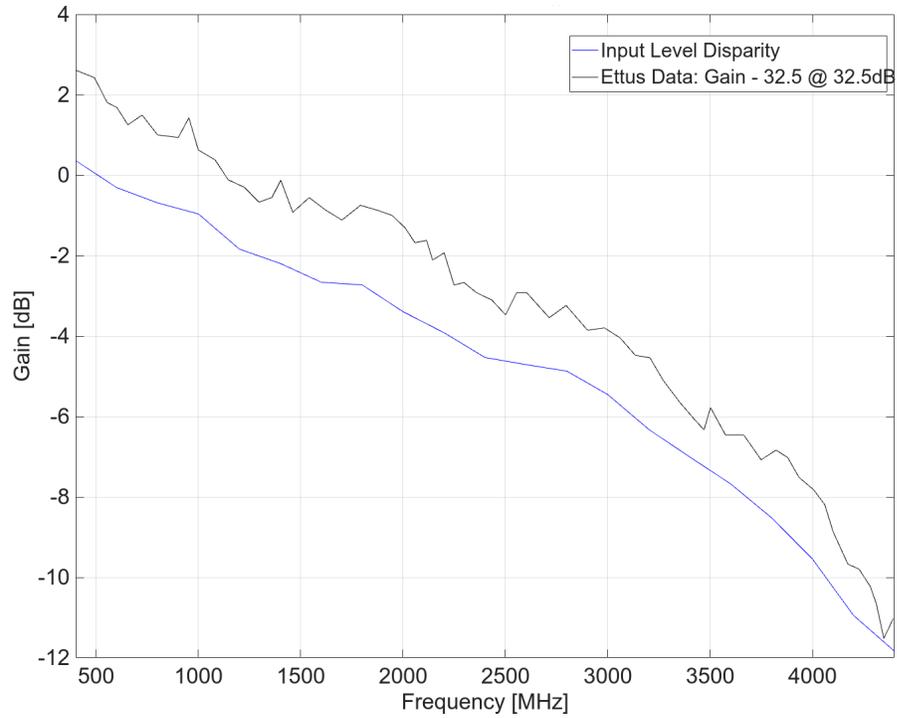


Figure 2.3: Relative full scale received power as a function of input power; compared with manufacturer-measured gain level (extracted from published plots at rx_gain = 32.5 dB); Both as a function of center frequency)[13]

2.4.4 Characterizing the Output Level and Distortion of an SDR Transmitter

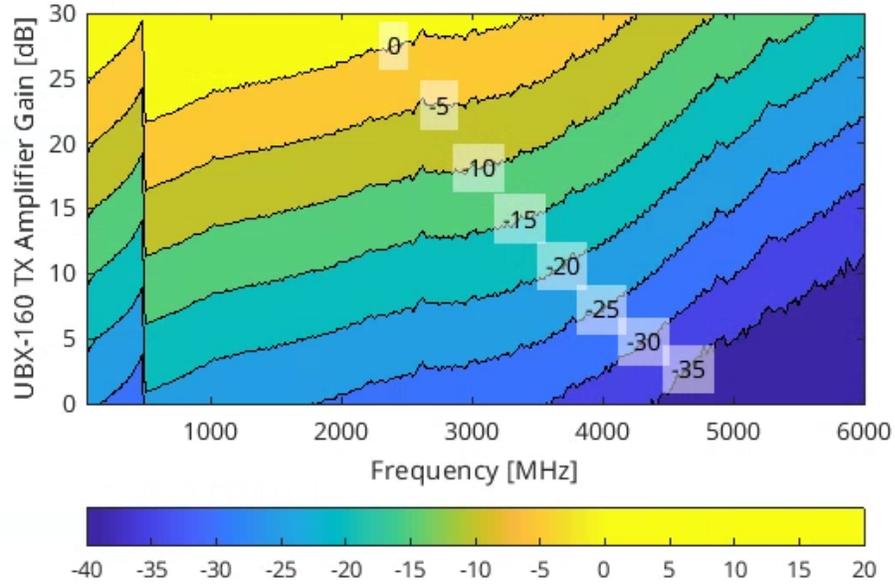


Figure 2.4: UBX-160 transmitter power [dBm] as a function of tx_gain and center frequency

It is also necessary to characterize the output level and distortion characteristics of the transmitter, particularly when the SDR is intended for use as a signal source. Figures Figure 2.4 and Figure 2.5 present the output power and IP3 performance, respectively, for the UBX-160 transmitter across frequency for various tx_gain settings. The output power data in Figure 2.4 was obtained through direct measurement, while the IP3 data in Figure 2.5 was extracted from manufacturer documentation [13]. As expected, output power increases with tx_gain. The frequency-dependent behavior exhibits a notable decline in output power at higher frequencies, with a variation exceeding 20 dB across the frequency range at most gain settings. The IP3 trend is broadly similar; however, at tx_gain = 30 dB and at lower frequencies, the IP3 approaches the output power level, indicating that intermodulation distortion may become significant under these conditions and should be considered when operating at high gain and low frequency.

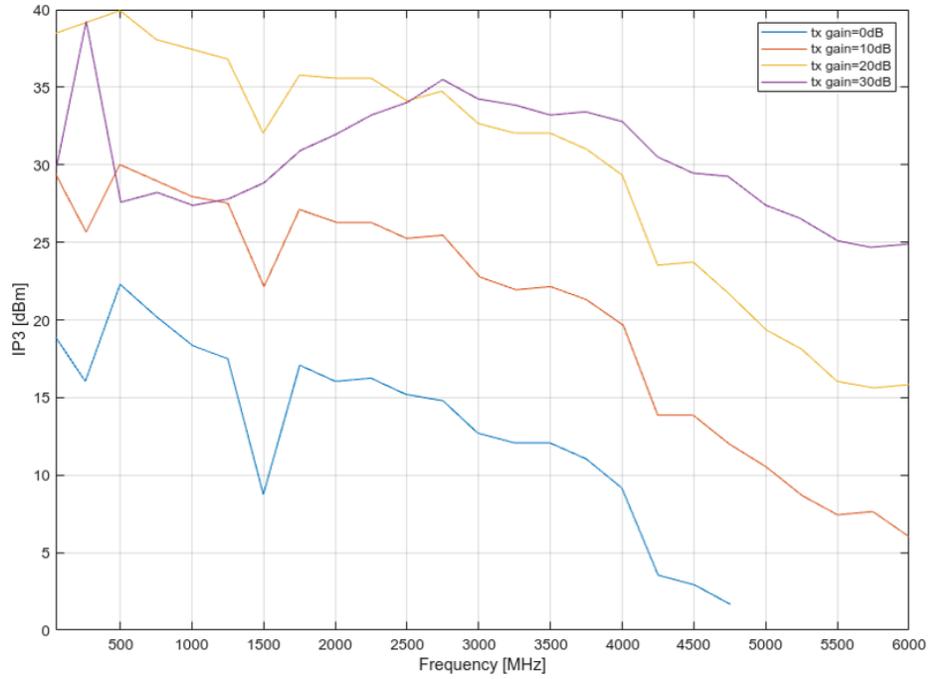


Figure 2.5: UBX-160 IP3 as a function of frequency, for select values of tx gain. Extracted from manufacturer published plots [13]

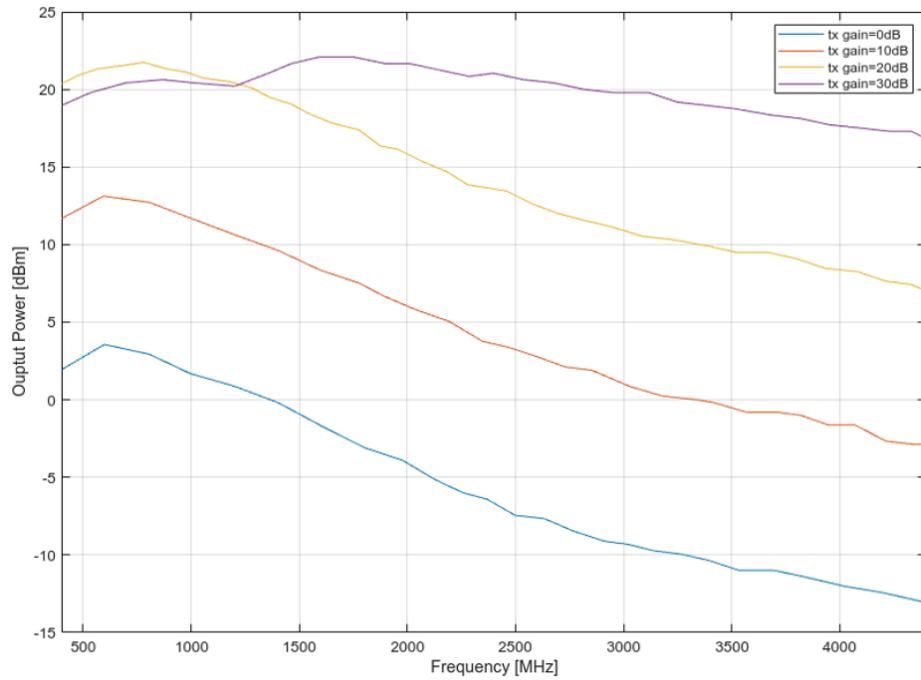


Figure 2.6: SBX-120 output power as a function of frequency at select values of tx_gain. Data extracted from manufacturer published plots. [14]

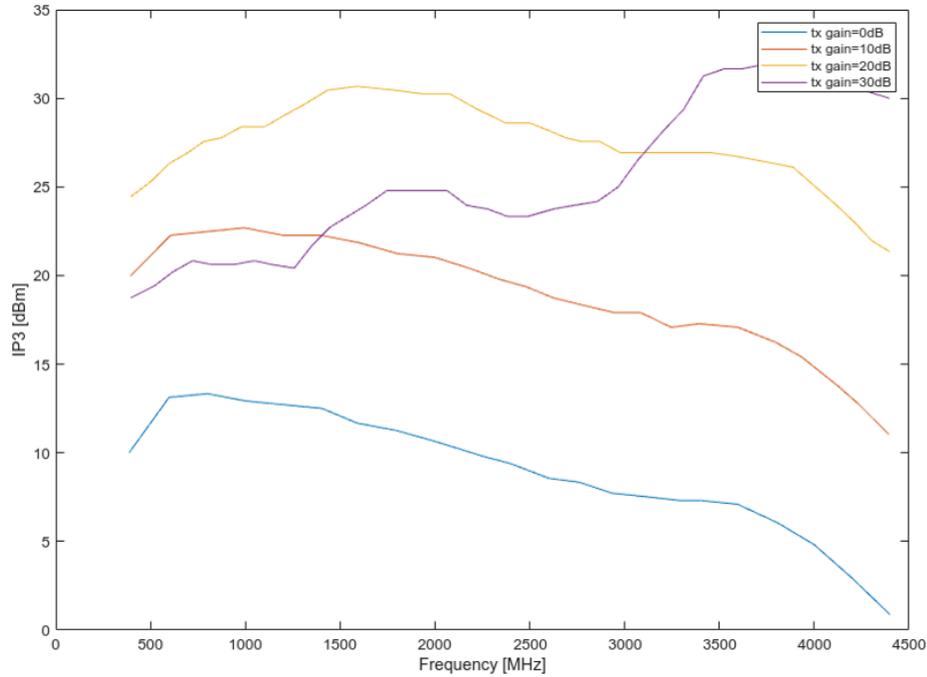


Figure 2.7: SBX-120 IP3 as a function of frequency at select values of tx_gain . Data extracted from manufacturer published plots. [14]

Following the transition from the UBX-160 to the SBX-120 radio during the course of the project, the output characteristics of the SBX-120 were also examined. The data, extracted from manufacturer-provided documentation [14], exhibit similar trends in output power to those observed with the UBX-160, as shown in Figure Figure 2.6. However, the SBX-120 demonstrates saturation behavior at approximately 20 dBm, particularly at lower frequencies. The IP3 performance, depicted in Figure Figure 2.7, is notably inferior to that of the UBX-160, with IP3 values approaching the output power level at gain settings of 20 dB or higher. This trend persists across the frequency range, with further degradation in IP3 performance evident at higher frequencies.

Minimally sufficient leveling of the output power can be achieved by adjusting the transmit gain as a function of frequency and applying a fixed attenuator. A target output power in the range of -25 dBm (at higher frequencies) to -40 dBm (at lower frequencies) is typically appropriate to optimize the signal level delivered to ADC when using the maximum rx_gain to minimize NF (as discussed in the next section). However, this approach does not account for additional frontend losses, which may amount to 20–25 dB depending on the system configuration. In such cases, a revised output power

target of approximately -20 dBm (at lower frequencies) to 0 dBm (at higher frequencies) is more appropriate. The recommended method is to first set the transmit gain so that the output power exceeds the target level by approximately 3 dB—where possible—and then fine-tune the amplitude of the digital waveform provided to the SDR. Although interpolation may yield acceptable results, calibration should be performed at the intended operating levels to ensure optimal accuracy. The implications of this power adjustment strategy are examined further in Chapter 6.

2.4.5 Determine the Noise Figure and Noise Variance

Some data are available for the expected NF levels of the SBX-120 daughterboard; however, the conditions under which they were measured are not specified. Because of this, I performed my own characterization. I used the y-factor technique described in the previous section, using a high-quality matched termination as a cold source and a 15 dB ENR noise source for a hot source. In addition to sweeping the frequency, the `rx_gain`, sampling rate, and number of samples were swept.

In order to process each data point:

- The data was windowed with a Hann ($\alpha = 2$) window, and its' amplitude was scaled appropriately (windowing is discussed in detail in Chapter 3.)
- The fast Fourier transform (FFT) of the data was taken, and the results were scaled appropriately
- The magnitude squared of the FFT data was computed
- The top and bottom 5% of data were trimmed to remove outliers (which includes peaks due to distortions, like LO leakage, which can be quite significant)
- The noise power per bin is calculated as the average of the remaining bins' powers.
- The noise power density is calculated by dividing the average power per bin by the bin width ($w_{bin} = \frac{f_s}{N}$)
- Similarly, the standard deviation is calculated as the standard deviation across the bins remaining after trimming and scaled to its per Hz value.

The noise characteristics data requires quite a lot of plots. This is because depicting 4- or 5-dimensional data with 2- or 3-dimensional plots is difficult. I have trimmed the number of plots and

their contents to show only essential trends in the data without conveying too much information in any single plot.

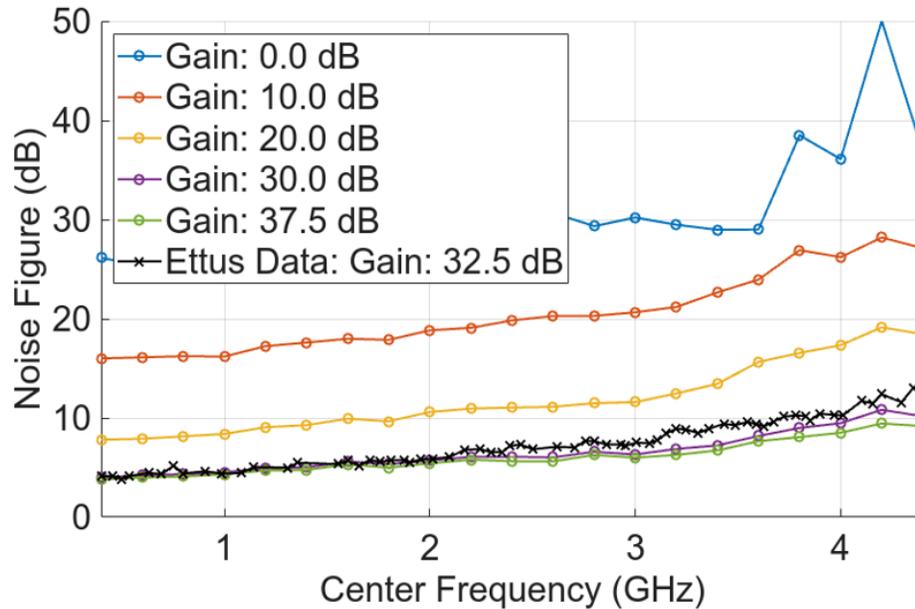


Figure 2.8: NF as a function of center frequency, with sampling rate of 100 MHz

Observe Figure 2.8 above, in which the NF is plotted with respect to frequency over the range 400 MHz to 4.4GHz (the range of the SBX radio), with separate lines corresponding to different rx gain values – 0, 10, 20, 30, and 37.5 dB, and also another line, representing manufacturer data at a rx_gain of 32.5 dB. As expected from Friis’ noise equation, the noise figure is drastically worse with lower values of rx_gain, which correlates with higher attenuation values immediately following the radio’s first amplifier. The manufacturer’s data trend line falls within a few dB the measured frontend loss. Increasing the gain above 30.0 dB does not significantly improve the NF; this will be discussed with the other plots. Furthermore, the accuracy/repeatability is poor at high noise figures (low gains).

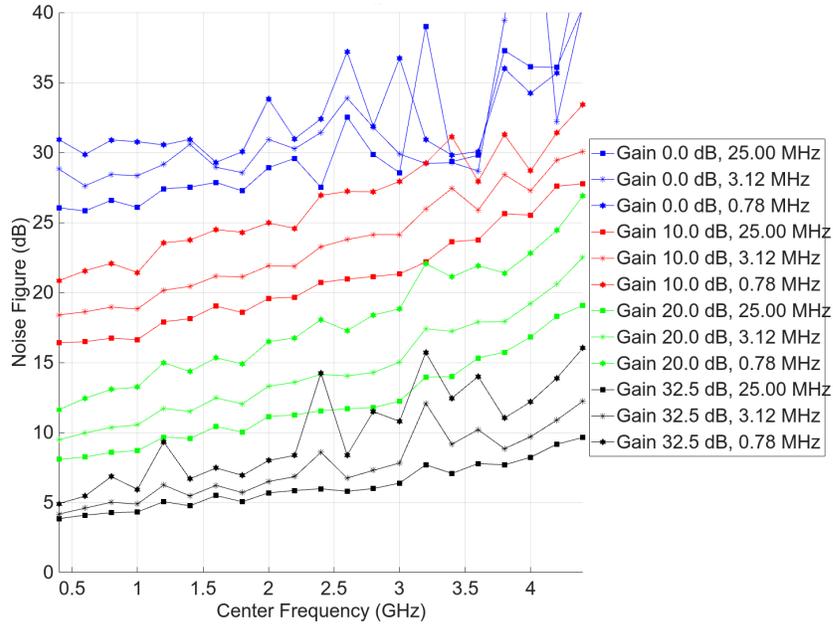


Figure 2.9: NF as a function of center frequency, at select gains and sample rates

In Figure 2.9, it can be observed that the NF increases with decreasing sampling rate, although the effect is less pronounced at the highest gains / lowest noise figures. Otherwise, the trend is reasonably stable, with a relatively fixed difference from one sampling rate to another. The apparent dependence of NF on the sampling rate is interesting because the data was not captured at this rate in analog; all data were captured at the same ADC rate and then digitally decimated down to the target rate shown. The NF for the highest sampling rates, not depicted here, showed little, if any, difference from one another or the highest rate depicted in this plot.

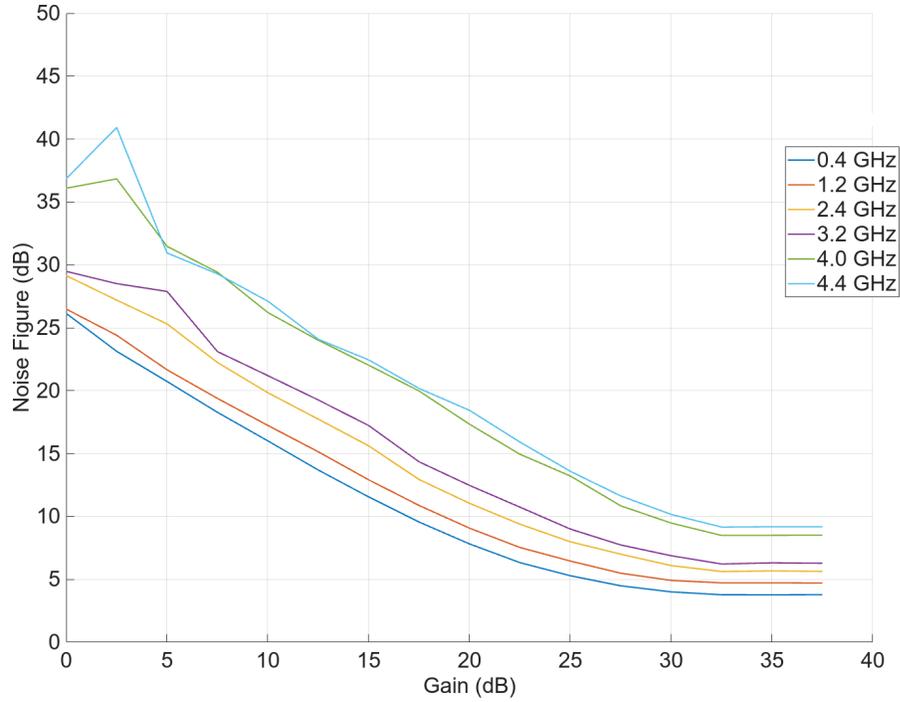


Figure 2.10: NF as a function of rx gain, at select frequencies with sampling rate of 100 MHz

The trend of NF with respect to gain is more stable and clear. Gain values up to 31.5 dB contribute to lower NFs, which aligns with the maximum digital attenuator setting of 31.5 dB. At higher values, the noise figure is flat, suggesting that further increases in gain could be applied digitally instead of analog. One trend that warrants further discussion is that the noise figure seems to be appreciably worse for the higher frequencies, particularly at lower gains.

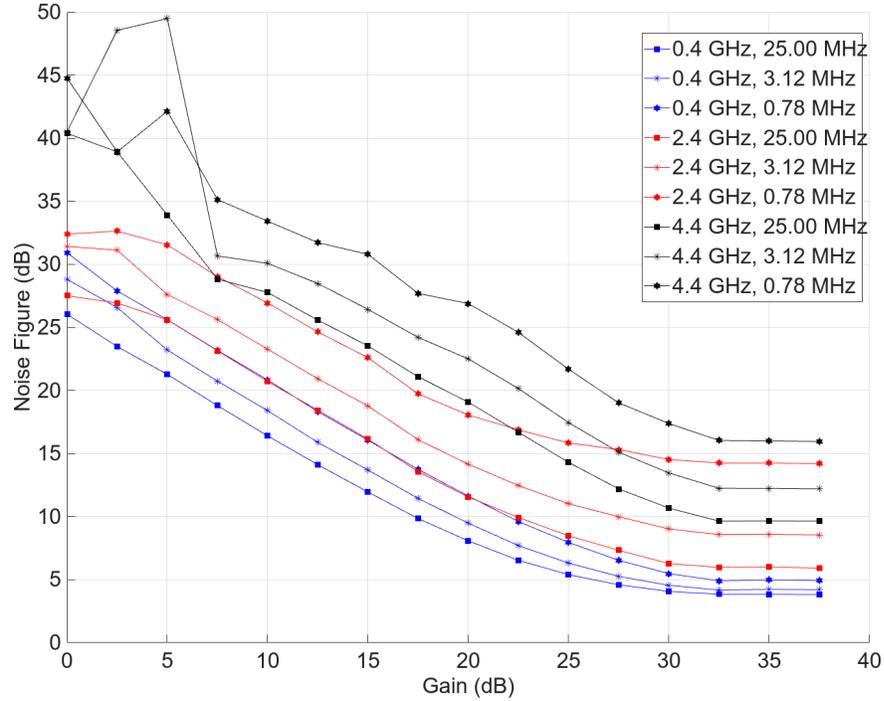


Figure 2.11: NF as a function of rx gain, at selected sampling rates and frequencies

The data observed at selected frequencies and sampling rates exhibit trends consistent with previous findings. Lower sampling rates tend to exacerbate the noise figure at higher frequencies, while their impact is minimal at the lowest frequency. This behavior is consistent with the frequency-dependent loss characteristics of the frontend, which exhibit minimal attenuation at low frequencies and up to 13 dB loss at the upper end of the frequency range (4.4 GHz).

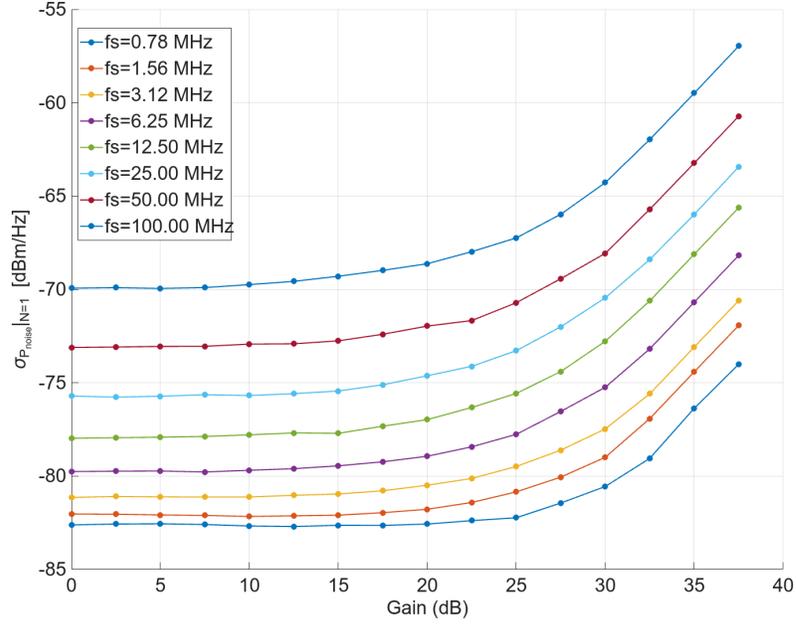


Figure 2.12: Standard deviation of the estimated PSDs as a function of rx_gain, median across frequency values, and normalized to $N = 1$ samples

The normalized standard deviation of the estimated PSD, $\sigma_{P_{\text{noise}}|N=1}$, is depicted in Figure 2.12, with the median taken across frequencies. Observe that $\sigma_{P_{\text{noise}}|N=1}$ is relatively flat at low gains and gradually begins to rise at higher gains. At a glance, $\sigma_{P_{\text{noise}}|N=1}$ appears much worse for the higher sampling rates, but this is because it has been normalized to $N = 1$ samples. If, instead, it were normalized to a fixed duration (e.g., $N = 1$ second), the curves would nearly align within a few decibels. Furthermore, the approximately uniform spacing between the sampling rate curves across all gains suggests that the dependence of $\sigma_{P_{\text{noise}}|N=1}$ on sampling rate is largely independent of gain.

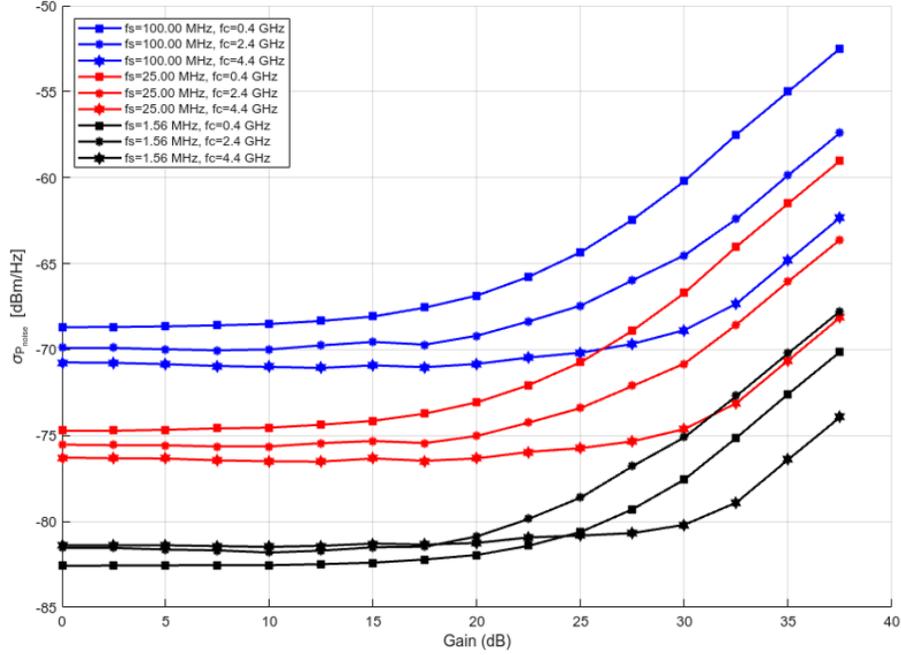


Figure 2.13: Standard deviation of the estimated PSD as a function of rx_gain, at selected sampling rates and center frequencies, normalized to $N = 1$ samples.

The plot above, Figure 2.13, shows that $\sigma_{P_{\text{noise}}}|_{N=1}$ seems to be somewhat lower at higher frequencies and that with respect to frequency, the values diverge significantly at higher gains. Again, I suspect that, as with the NF, the uncompensated loss in the receiver with respect to frequency such that there is more loss at higher frequencies is again responsible for this effect. Looking at any of these lines on a frequency-by-frequency basis, you see lines that follow the same trend, with a fixed shift as in the previous plot, suggesting again that the dependence of $\sigma_{P_{\text{noise}}}|_{N=1}$ on sampling rate is independent of the gain. The trend with respect to frequency will be easier to observe in the following plot. However, you can see that in the high-frequency high-gain case, the $\sigma_{P_{\text{noise}}}|_{N=1}$ is worse.

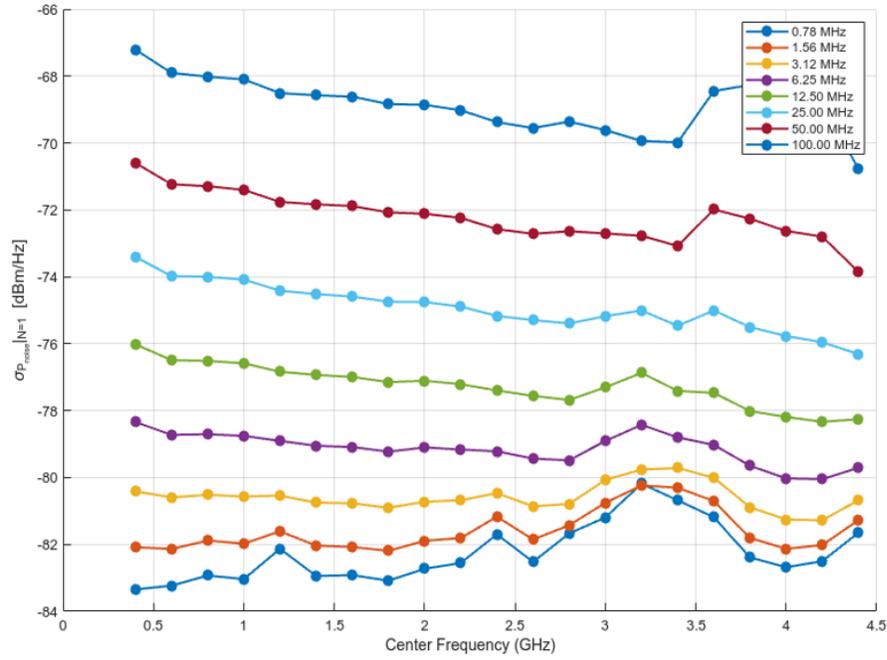


Figure 2.14: Standard deviation of the estimated PSDs as a function of center frequency, median taken across rx_gain values, normalized to $N = 1$ samples.

You can see in Figure 2.14, where the $\sigma_{P_{\text{noise}}|N=1}$ is shown as a function of center frequency, median across the rx_gain values, that again, altering the sample rate produces a fixed offset in $\sigma_{P_{\text{noise}}|N=1}$, except at the lowest sampling rates. Compared to the trend with respect to gain, the dependence on center frequency is relatively weak, with values typically not varying by more than a few dB.

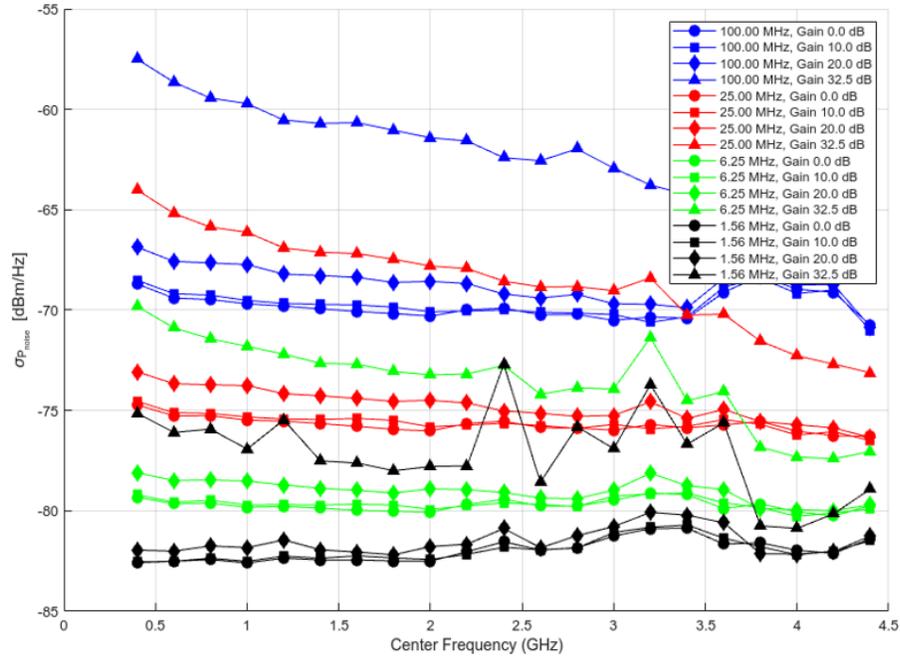


Figure 2.15: Standard deviation of the estimated PSDs as a function of center frequency, at selected sampling rates and rx_gain values, normalized to $N = 1$ samples

The standard deviation can be seen for select gains and sampling rates in Figure 2.15 above. The trends at gains of 20 dB or lower are relatively flat with respect to frequency. However, jumping from 20dB to 32dB gain significantly raises the NF. An increase of $\sim 8\text{dB}$ $\sigma_{P_{noise}}|_{N=1}$ should still be worth the $\sim 4\text{-}8$ dB improvement in NF, though.

2.4.6 Measurement Stability

After tuning the receiver to the desired frequency and waiting a short time for the LO to settle, the SDR exhibits excellent short-term stability. This stability can best be demonstrated with a concrete example. The following measurements were performed using a high-quality 10 MHz external reference derived from a laboratory-grade, oven-controlled crystal oscillator (OCXO).

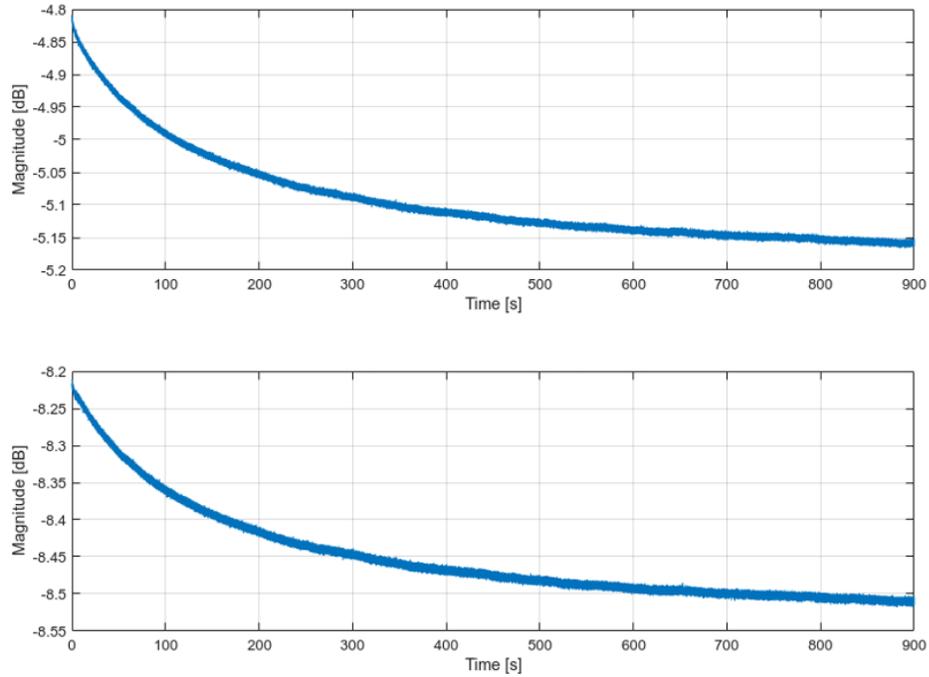


Figure 2.16: STFT derived a-wave (top) and b-wave (bottom) magnitudes.

In the experiment, a continuous 15 minute I/Q capture was recorded at a carrier frequency of 1 GHz, with a IF offset of 128 kHz and a sample rate of 781.25 kS/s. The processing techniques used are only briefly described here, but will be described in more detail in Chapter 3; unfamiliar readers may need to skip ahead. The attached DUT was a high quality

RF short. Rather than employing a single FFT on the entire dataset, a short-time fourier transform (STFT) approach was used: the data is segmented into many small, essentially time-localized segments. [15] In this case $N = 1024$ samples were used per segment with a 50% segment overlap, yielding an effective frame rate of $\frac{781.25kS/s}{1024*0.5} = 1525.6\frac{S}{s}$. In order to minimize the spectral leakage, each segment was mixed so that the tone of interest was centered on an FFT bin. After windowing, an FFT was computed, and the magnitude of the relevant frequency bin was extracted as a function of time. The approach of capturing many FFT values for a single frequency with respect to time is similar to taking a 1-bin-wide slice of a waterfall plot with respect to time. Figure 2.16 above illustrates the observed magnitude of both the measured signal and the reference signal plotted across time. Both show a similar downward drift, likely due to analog frontend settling or perhaps

other effects. Fortunately, the drift is common to both channels; therefore, taking the ratio of the measured signal to the reference signal significantly detrends the magnitude and phase data.

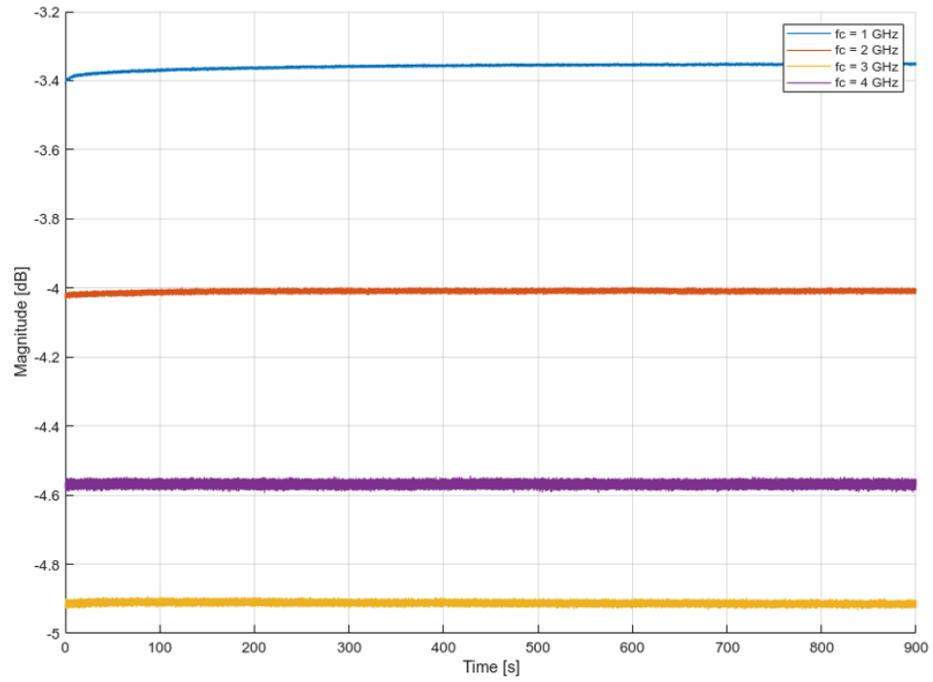


Figure 2.17: STFT magnitude ratio of reference (a-wave) and measured (b-wave) channels across time at select frequencies

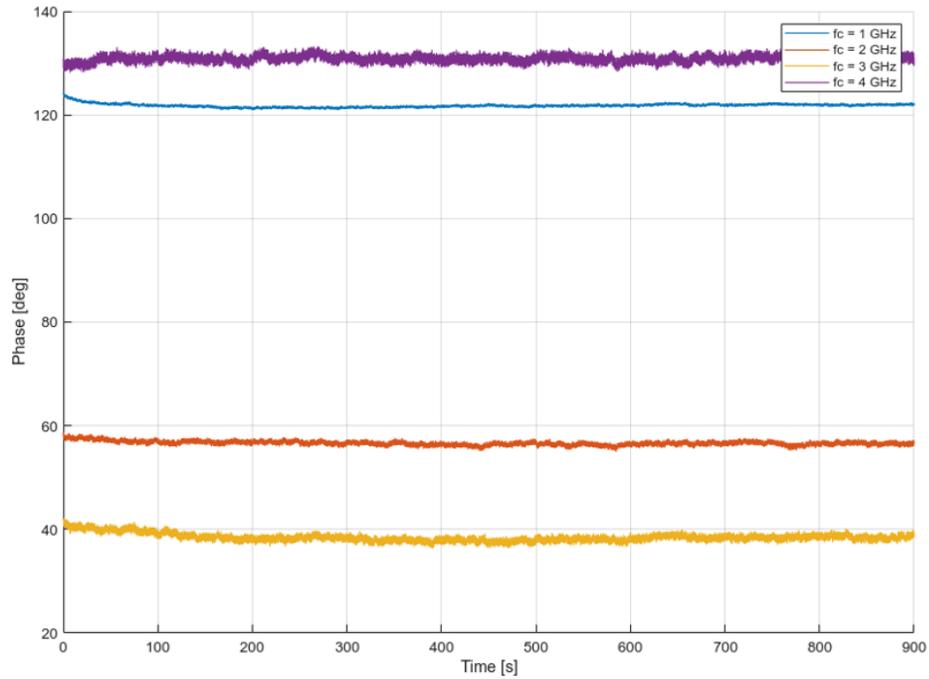


Figure 2.18: STFT phase difference of reference (a-wave) and measured (b-wave) channels across time at select frequencies

The resulting plots of magnitude ratio and phase difference (measured / reference) are shown for four different frequencies: 1, 2, 3, and 4 GHz in figures Figure 2.17 and Figure 2.18 above. After initial transients (visible on the left side of the plot), both magnitude and phase appear steady but noisy, and more so at higher frequencies. The remaining variations are consistent with the hypothesis that phase noise is the dominant source of noise in this data.

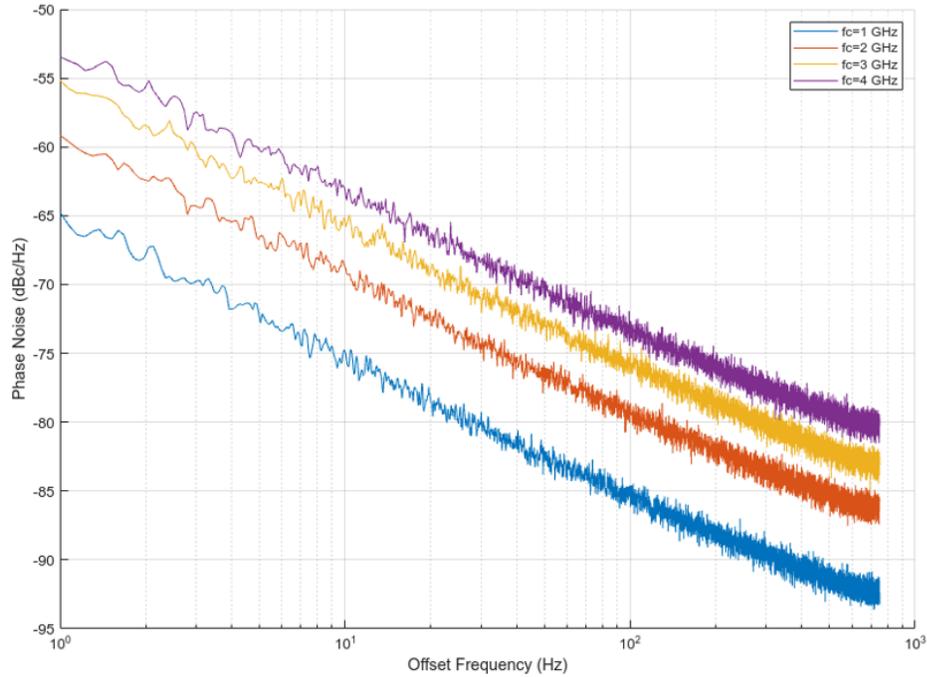


Figure 2.19: Interpolated Phase Noise PSD as measured at select frequencies

The ratio samples were first converted to linear quantities as the first step in extracting the phase noise spectrum from the data. Next, the time-domain signal was DC-centered by subtracting the mean, windowed, and transformed using an FFT to obtain the power spectral density. The result is the single-sideband phase noise spectrum, $L(f)$, where f represents the noise offset frequency, ranging from ~ 1 Hz up to half the frame rate (~ 763 Hz in this case). Figure 2.19 above shows this extracted phase noise spectrum. Observe that the phase noise decreases with increasing offset frequency, as expected. Also, each doubling of the carrier frequency leads to an increase in phase noise of approximately 6 dB, which is consistent with theory: phase noise in dBc/Hz increases with $20 \log_{10}(f_{\text{carrier}})$.

From the phase noise spectrum $L(f)$, one can calculate the root mean square (RMS) timing jitter using the following relationship: [16] $\sigma_{\text{jitter}} = \frac{1}{2\pi f_{\text{carrier}}} \cdot \sqrt{\int_{f_1}^{f_2} 2L(f)df}$ where:

- $L(f)$ is the phase noise in linear units (not dBc/Hz)
- f_{carrier} is the signal's carrier frequency
- The factor of 2 accounts for integration over both sidebands (positive and negative frequencies)

For this dataset, using an integration range of 1 to 763 Hz, all measured signals (1 GHz to 4 GHz) produced essentially identical jitter values of approximately 228 femtoseconds. This indicates that the system’s phase-noise performance is flat when viewed as timing jitter, as expected for a high-quality reference oscillator.

The above measurements show that, after tuning and locking, the SDR system, when used in conjunction with a precision reference, exhibits excellent short-term phase stability, but there is a wrinkle

2.4.7 Measurement Repeatability

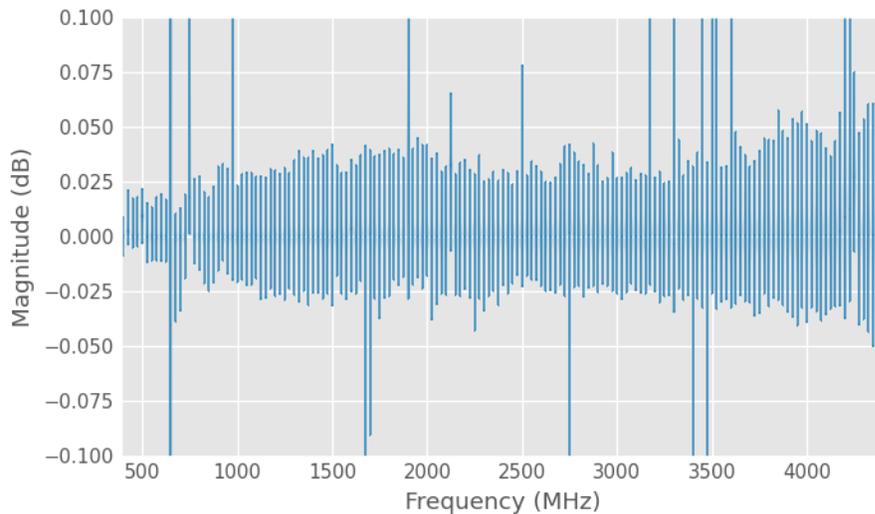


Figure 2.20: Normalized magnitude difference between ref and meas channel repeatability violin plot of SDR measured with a reflective load

However, the repeatability of measurements is somewhat of a thorn in the side of SDR. Observe that in Figure 2.20 above the magnitude measurements are typically relatively repeatable, with magnitude variations ranging from about 0.025 dB at lower frequencies to 0.050 dB at higher frequencies. Magnitude variations of .050 dB or less are excellent; however, there are a great number of outliers in the data. Likewise, note that in Figure 2.21 below, which tracks the phase difference between the reference and measured channels with a short-circuit termination, there is a similar trend, ranging

from about 0.25 deg variability at 400 MHz to about 2.5 degrees at 4400 MHz. The linear trend of increasing phase variability is expected as frequency increases. However, phase variation on the order of +/-2.5 degrees is problematic for instruments that require a great deal of precision, such as vector network analyzers (VNAs). This is because, as device return loss is increased, observed variations of magnitude and phase have increasingly significant effects. Commercial VNAs typically claim better than 1 degree measured phase error even at high return loss, suggesting very high measurement repeatability. A study could be performed, with either synthetic or commercial network analyzer data, with random perturbations applied to both the calibration and dut measurement data, in order to fully characterize the calibration degradation caused by poor magnitude and phase repeatability.

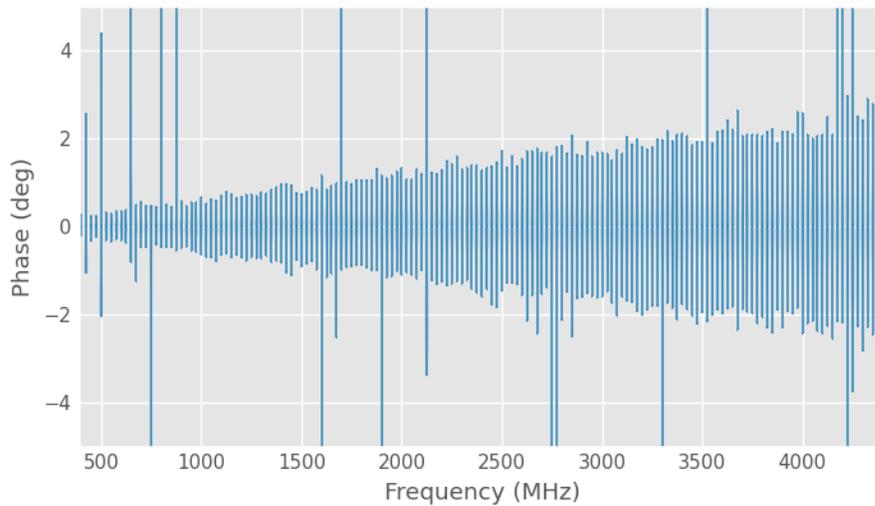


Figure 2.21: Normalized phase difference between ref and meas channel repeatability violin plot of SDR measured with a reflective load

The substantial phase-difference variation between supposedly coherent channels across re-tunes with the SBX-120 is most likely an architectural problem. The Analog Devices ADF4351 LO and phase-locked loop (PLL) synthesizer chips supposedly have excellent repeatability, but the utilization of two of them on separate boards locked to a common 10 MHz reference leaves more room for problems to occur. Any source of timing skew can lead to significant phase error – even a timing difference of 0.631 ps will result in a phase difference of 1° at 4.4 GHz. The correction for this architectural problem is to share not just the phase reference, but also the LO and PLL. This should also help

reduce the worst-case outliers, though it may not eliminate them. The outliers in the data will be discussed in more detail in Chapter 5.

2.5 Chapter Conclusion

This chapter explored the architecture and performance of RF receivers, with an emphasis on practical considerations for SDR-based systems. It began with an overview of the characteristics of radio architectures, particularly the direct conversion and superheterodyne receivers. The implications of quadrature signals and the modularity of SDR design were highlighted, establishing a baseline for understanding modern SDR-based systems.

Subsequent sections focus on quantifying receiver performance through standard FOMs, especially NF and sensitivity. The Y-factor technique was identified as well-suited to characterizing receivers, and the methodology of the y-factor technique was discussed at length.

The chapter shifted from theory to practice in an empirical characterization of the Ettus Research X310 SDR with an SBX-120 daughterboard. Experiments were discussed, and the system's behavior was analyzed across different gain settings, sampling rates, and center frequencies. The input power required to saturate the radio's ADC was identified as a function of receiver gain and center frequency. The noise figure of the system was characterized as a function of receiver gain, center frequency, and sampling rate.

The performance and stability of the system were analyzed next. Short-term stability was shown to be excellent when using a high-quality external reference oscillator, with both amplitude and phase measurements showing minimal drift over time. However, repeatability between tuning events was found to be limited by variations in the LO settling, leading to measurement variability on the order of a few dB or several degrees. Future chapters will discuss the necessity of statistical post-processing to achieve acceptable measurement repeatability and accuracy.

Taken together, the results presented in this chapter offer both theoretical grounding and practical insight into the characterization of modern SDRs. The combination of architectural analysis, perfor-

mance quantification, and experimental validation provides a foundation for subsequent system-level evaluations of like systems.

Chapter 3

DSP Methods for Signal Analysis

3.1 Sampling and Frequency Transform Fundamentals

Contemporary radio frequency (RF) Receivers and vector network analyzers (VNAs) are highly dependent on digital signal processing (DSP) to extract useful information from signals. To understand how signals are processed in the digital domain, we must first establish the mathematical foundations that allow continuous signals to be represented in the frequency domain.

3.1.1 Fourier Analysis and Signal Representation

A periodic function $x(t)$ with period T can be represented as a Fourier series, decomposing the signal into an infinite summation of complex exponentials:

$$x(t) = \sum_{n=-\infty}^{\infty} C_n e^{j2\pi n f_0 t}$$

where C_n are the Fourier coefficients and $f_0 = 1/T$ is the fundamental frequency. [17]

For aperiodic signals, this framework extends to the Fourier transform [17] by taking the limit as the period approaches infinity:

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft} dt$$

This representation expresses any signal as a continuous spectrum of frequency components.

3.1.2 Sampling and the Discrete-Time Fourier Transform

In practical systems, continuous-time signals are sampled to obtain discrete-time sequences. Sampling with a fixed interval T_s yields the following.

$$x[n] = x(nT_s)$$

According to the Nyquist-Shannon Sampling Theorem, a band-limited signal can be perfectly reconstructed if the sampling rate f_s satisfies:

$$f_s > 2f_{max}$$

where f_{max} is the highest frequency component present in the signal, sampling below this rate results in aliasing, a phenomenon in which high-frequency out-of-band components fold into the observable spectrum.

The discrete-time Fourier transform (DTFT) provides a frequency-domain representation of discrete-time signals: [17]

$$X(e^{j2\pi f}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j2\pi fn}$$

Although the DTFT is conceptually useful, it requires an infinite number of samples and is therefore not computable in practice.

3.1.3 The Discrete Fourier Transform and the Fast Fourier Transform

The discrete Fourier transform (DFT) is used to process finite-length signals. For a sequence $x[n]$ of length N the DFT is given by:

$$X[k] = \sum_{n=0}^{N-1} x[n]e^{-j2\pi kn/N}, k = 0, 1, \dots, N - 1$$

The DFT provides a discrete approximation of the DTFT, assuming that the input signal is periodic with period N . [17] Direct computation requires $\Theta(N^2)$ operations, which is computationally expensive. The fast Fourier transform (FFT), which uses the Cooley-Tukey algorithm, reduces this complexity to $\Theta(N \log(N))$ operations by exploiting symmetry and periodicity in a divide-and-conquer approach, making spectral analysis much more efficient.

3.2 Spectral Leakage and Windowing

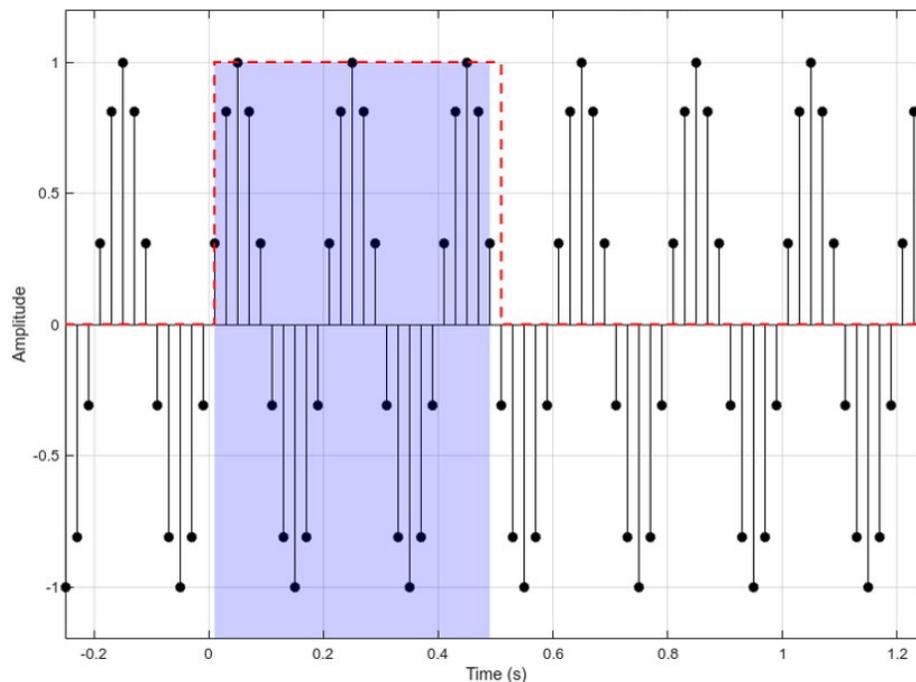


Figure 3.1: Collecting a finite number of samples is equivalent to multiplying by a rectangle function in the time domain

The transformation of an infinite-duration signal into a discrete finite-length sequence, with duration T and number of samples N , is equivalent to multiplying the signal by a rectangular window in the time domain, as in 3.1. By the convolution property of the Fourier transform, this time-domain multiplication is equivalent to a frequency-domain convolution. [17] The Fourier transform of a rectangle is a sinc function that introduces significant sidelobes. When the signal frequency is well

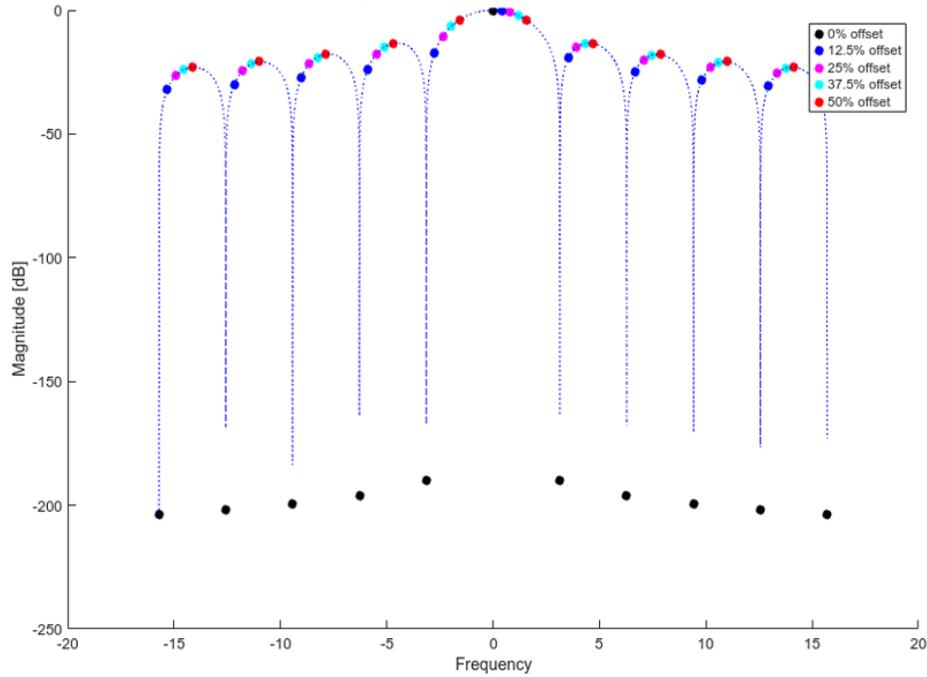


Figure 3.2: Small alignment errors between signal frequency and sinc frequency result in large spectral leakage

aligned, the main lobe of the sinc function is centered on the bin with maximum amplitude, and all other bins are in the sinc function nulls. However, misaligned signals leak significant energy into the sidelobes, an effect called **spectral leakage**. The sinc function is very sensitive to misalignments, so even small misalignments result in considerable leakage.

Observe Figure 3.2, which shows an unnormalized sinc function. The blue dotted line represents the continuous function ($sinc(x) = \frac{\sin(x)}{x}$), while the black dots indicate properly aligned points where leakage is minimal (π spacing). The remaining points illustrate various offsets ranging from 12.5% to 50%. A 50% misalignment represents the worst-case scenario, the main lobe sees maximum scalloping loss and it aligns with all of the side-lobe peaks. You can see that even small misalignments result in a significant shift of energy into the sidelobes, with any misalignment at all resulting in near-worst-case leakage.

Spectral leakage can be mitigated by applying a function that tapers the signal's endpoints. The soft transition reduces the magnitude of the sidelobes in the frequency domain. However, tapering in this manner introduces other trade-offs. It results in an increased width of the main lobe, which increases the equivalent noise bandwidth (ENBW), and reduces frequency resolution. Each window

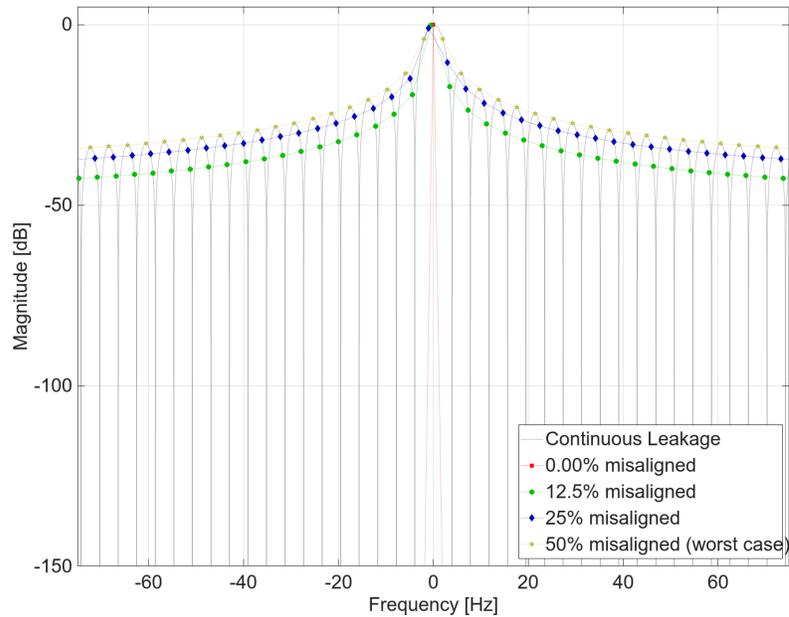


Figure 3.3: Spectral leakage for a Rectangular window with several frequency offsets

embodies a trade-off between the width of the main lobe, the amplitude of the first sidelobe, and the rate of sidelobe roll-off. The choice of window must be guided by the application requirements, such as minimizing leakage from close-in interference, improving resolution, or reducing integrated noise. [18]

The four figures above (Figure 3.3, Figure 3.4, Figure 3.5, and Figure 3.6) depict the Fourier transform of a signal with various degrees of misalignment from 0% to 50%, for four different windowing functions: Rectangular, Hamming, Hann($\alpha = 2$), and Blackman-Harris(4-sample). The dark gray line represents the continuous-time Fourier transform of the windowed signal, while the other lines are sampled with different sampling times represented by different markers. You can make several observations from these plots [18] :

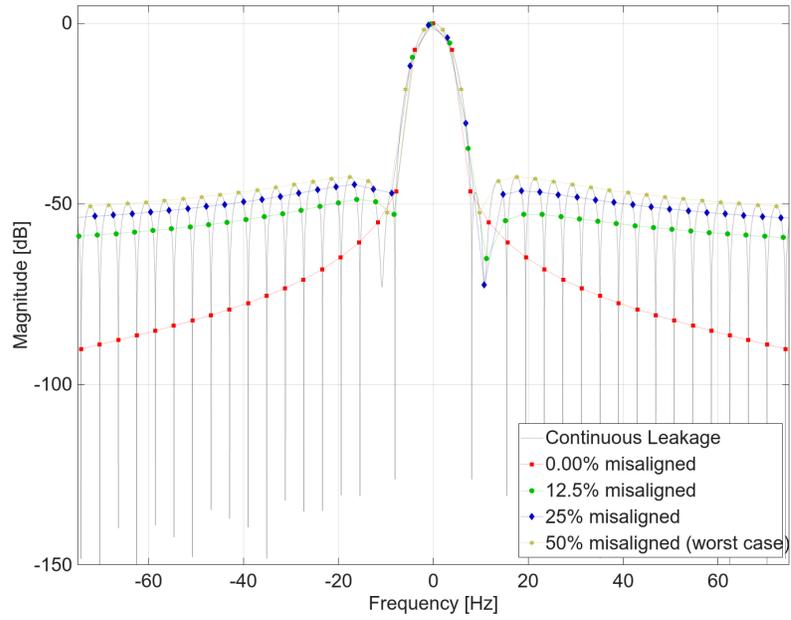


Figure 3.4: Spectral leakage for a Hamming window with several frequency offsets

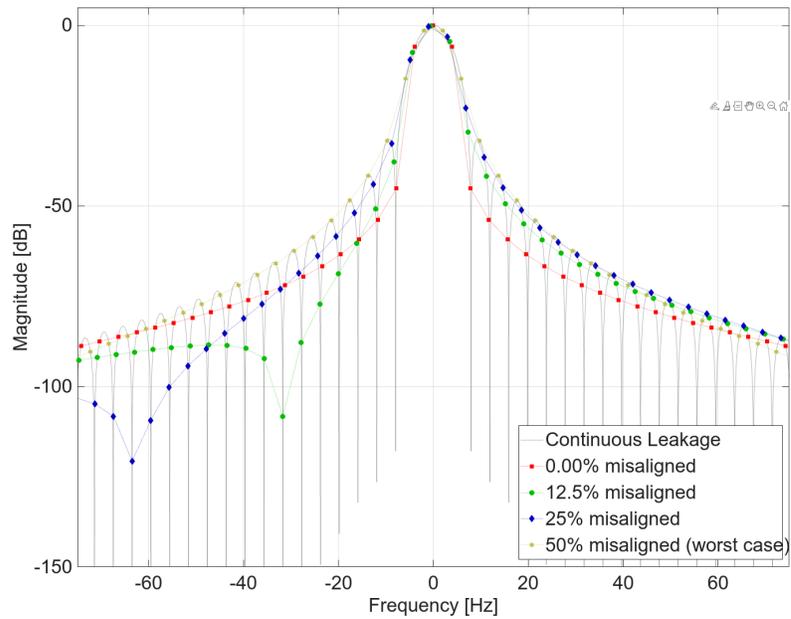


Figure 3.5: Spectral leakage for a Hann window with several frequency offsets

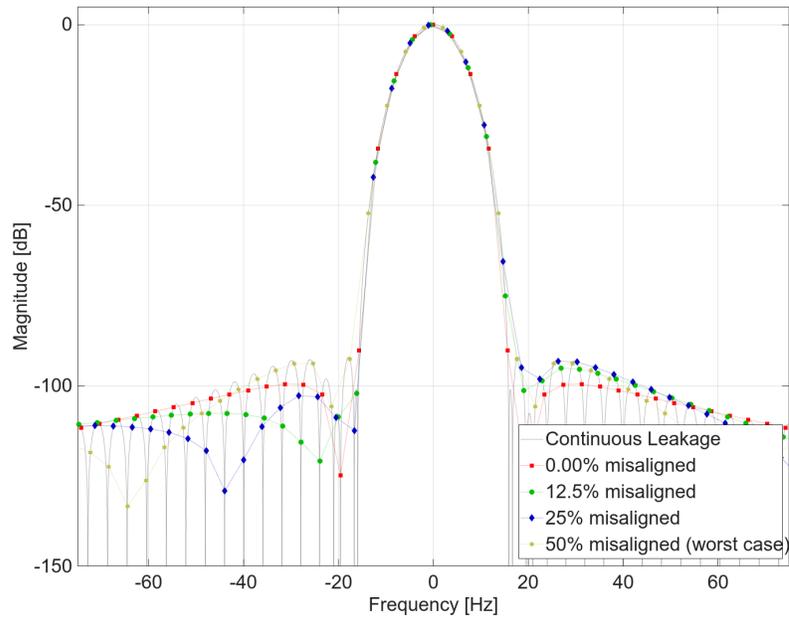


Figure 3.6: Spectral leakage for a Blackman-Harris window with several frequency offsets

Table 3.1: Window function performance summary, data from [18]

Window	First Sidelobe	Sidelobe Rolloff	ENBW
Rectangular	-13 dB	-6 dB/octave	1.00
Hamming	-42 dB	-6 dB/octave	1.36
Hann	-32 dB	-18 dB/octave	1.50
Blackman-Harris	-92 dB	-18 dB/octave	2.00

- The Rectangular window has a high first sidelobe peak and slow sidelobe roll-off rate. If the signal is perfectly aligned with the bin, there will be no leakage; otherwise, the leakage will be the worst of all windows.
- The Hamming window has a much lower first sidelobe amplitude and a slow roll-off rate, but it has a moderately narrow main lobe.
- The Hann window's first side-lobe amplitude is somewhat higher than the Hamming window but exhibits a much steeper roll-off and has a slightly wider main lobe.

- The Blackman-Harris window has an excellent first-sidelobe level and fast rolloff, but a much wider main-lobe width.

When analyzing a signal with noise, this increase in main lobe width means a higher ENBW— that is, more noise energy is captured in the signal bin. It also reduces frequency resolution. However, this trade-off is often worthwhile due to the substantially reduced spectral leakage, which prevents unwanted contributions from distant frequencies from contaminating the signal bin.

Numerous windowing functions are available, each with its own trade-offs. The rectangular window is the most accurate when the signal is perfectly aligned, but outside of contrived examples, it will generally have the worst performance because of severe spectral leakage. The Hamming window offers a reasonable balance, with moderate main lobe width, decent roll-off, and manageable sidelobe levels.

Each window function makes different trade-offs. For example:

- The generalized Hann window sacrifices the main lobe width in exchange for faster roll-off.
- The Dolph-Chebyshev window eliminates roll-off entirely in favor of maintaining a fixed, low sidelobe level without sacrificing as much main lobe width.

The Hann window is arguably the most general purpose window, with an acceptable first-sidelobe level, fast roll-off, and moderately good main lobe width. Its fast rolloff results in a lower integrated noise power, which is preferred.

Other windows, such as Dolph-Chebyshev, may offer better worst-case performance due to a lower first sidelobe level and narrower main lobe, which reduces the ENBW and improves frequency resolution. However, this is attained by sacrificing rolloff resulting in higher integrated noise power.

Ultimately, the optimal windowing function for a measurement is driven by requirements and signal conditions.

3.3 Measuring A Tone In Noise

When analyzing a signal, the first step is to determine the type of data being analyzed and the information that needs to be extracted from it. Is the signal time harmonic or does it change over time? Is it modulated, wideband, narrowband, or a tone? What parameters need to be measured — amplitude, phase, bandwidth, center frequency, or something else? Answering these questions helps narrow the scope of the analyses.

3.3.1 Time domain analyses - Lock-in Amplifier

Let us first consider determining the amplitude of a stationary sinusoid, one of the simplest measurements, but one that still requires a nuanced approach. The optimal instrument for this task is the lock-in amplifier (LIA), a phase-sensitive detector designed to extract signals buried in noise.

The classic analog LIAs uses a coherent reference to mix the signal down to the 0 Hz baseband, a process mathematically described as:

$$A_{RF}A_{LO}\cos(2\pi f_{RF} + \phi_{RF})\cos(2\pi f_{LO} + \phi_{LO})$$

Using the product-to-sum identities from trigonometry, you can show that this is equivalent to

$$\frac{1}{2}A_{RF}A_{LO}[\cos(2\pi(f_{RF} - f_{LO}) + (\phi_{RF} - \phi_{LO})) + \cos(2\pi(f_{RF} + f_{LO}) + (\phi_{RF} + \phi_{LO}))]$$

By assuming a coherent reference with unit amplitude ($A_{LO} = 1$, $\phi_{LO} = \phi_{RF}$, and $f_{LO} = f_{RF}$) it simplifies to

$$\frac{1}{2}A_{RF} + \cos(4\pi f_{RF} + 2\phi_{RF})$$

Afterward, applying a low-pass filter (LPF) removes the high frequency component, leaving the DC term: $\frac{1}{2}A_{RF}$.

Thus, the original signal amplitude can be recovered. However, without a coherent reference, the measured magnitude will vary with phase alignment, and in the worst case, the detected amplitude will be zero. [19]

3.3.2 Vector Lock-In Detection - Extracting Phase Information

A variation on this technique is to make a vector measurement by splitting the signal into in-phase and quadrature (I/Q) components. The analog approach to this task is to: 1. **Splitting the signal into two paths**, with one path being delayed by 90 degrees. 2. **Mix both signals with the local oscillator (LO)**. 3. **Using the I/Q component amplitudes to reconstruct complex magnitude and phase**. Mathematically, assuming the LO has both unit amplitude and phase ($A_{LO} = 1$, $f_{LO} = f_{RF}$, and $\phi_{LO} = 0$):

$$I = A_{RF} \cdot \cos(2\pi f_{RF} + \phi_{RF}) \cdot \cos(2\pi f_{RF}) \quad Q = A_{RF} \cdot \cos(2\pi f_{RF} + \phi_{RF}) \cdot \sin(2\pi f_{RF})$$

Which, after using the product/sum trigonometric identities and applying a LPF – simplifies to

$$I = \frac{1}{2} A_{RF} \cdot \cos(\phi_{RF}) \quad Q = \frac{1}{2} A_{RF} \cdot \cos(\phi_{RF} - 90^\circ)$$

From these, the magnitude and phase are simple to calculate:

$$A_{RF} = 2\sqrt{I^2 + Q^2} \quad \phi_{RF} = \text{atan2}(Q, I) \quad (\text{where } \text{atan2} \text{ is the four-quadrant arctangent})$$

This approach also loosens the requirement for a coherent reference. If the signal frequency is known with significant precision, then LIA can internally generate an LO signal at the same frequency. If it is locked to a common reference, then it will also provide a stable phase measurement. Otherwise, the magnitude measurement will still be accurate (provided that the frequency is stable), but the phase will slowly drift over time. The use of a coherent reference in conjunction with an LIA has one other significant advantage: it can recover the magnitude and phase even if the signal is varying over time. [20].

3.3.3 Amplitude Extraction and Measurement Noise Floor

Regardless of whether a single channel or I/Q measurement is used, the amplitude can be extracted in multiple ways.

Time Domain Averaging

Averaging in the time domain allows the mean value to converge on the true DC value, driving down both the noise floor level and variance. The effective measurement integration bandwidth (IBW) is related to the integration time (τ) as $B_{avg} = 1/\tau$.

The total noise power within this bandwidth is given by: $P_{noise} = 10 \cdot \log_{10}(k_B T B_{NEB}) + NF$ where:
* k_B is Boltzmann's constant * T is the temperature in Kelvin, often taken to be 290 Kelvin under normal conditions * B_{NEB} is the noise equivalent bandwidth (NEB) in Hz * NF is the receiver's noise figure in dB

kT is the thermal noise power spectral density, and at room temperature, $kT = -174$ dBm/Hz. For example, with $B = .01$ Hz and $NF = 5$ dB, the measurement noise floor is: $P_{noise} = -174 + 10 \cdot \log_{10}(.01) + (5) = -189$ dBm.

Such a low-noise floor enables the detection of signals deeply buried in noise. However, the trade-off is a long integration time – in this case:

$$\tau = 1/B = 100s.$$

Frequency Domain Processing

Alternatively, processing can be done in the frequency domain. In this case:

1. Frequency-shifting the signal to baseband is optional, but strongly desirable because this ensures that the presumably strongest signal is centered on a frequency bin)
2. The signal should be windowed to minimize spectral leakage.
3. The FFT is computed and acts as a filter bank.
4. The bin index corresponding to the signal is extracted as: $k = \frac{f_s}{N} \cdot \frac{f_c}{f_s} = \frac{f_c}{N}$
5. The NEB is calculated as: $B_{NEB} = \frac{f_s}{N} \cdot ENBW$

Here, ENBW is a property of the windowing function used, which is specifically related to the window's main lobe width.

For example, assuming constraints $f_{s-\min} = 781.25\text{kHz}$, $N_{\text{fft}-\max} = 4\text{M}$, and $ENBW = 1.5$ (for $\text{Hann}(\alpha = 2)$):

- This yields a NEB of
- $B_{\text{NEB}} = \frac{781.25\text{kHz}}{4\text{M}} \cdot 1.5 = 279\text{mHz}$

In order to match the noise floor of the time-domain approach example above, the bandwidth must be reduced by a factor of ~ 28 by:

- Increase the number of samples by 32x and decimate by the same.
- The new effective sample rate is 24.4 kHz, yielding:
- $B_{\text{NEB}} = \frac{24.4\text{kHz}}{4\text{M}} \cdot 1.5 = 8.7\text{mHz}$

Implementation with a moderate FFT size of 2048 – e.g., to ease a hardware implementation – would require further decimation by $D = \frac{N}{2048} = 2048$, reducing the sample rate and, by extension, the Nyquist bandwidth: $f_s = \frac{24.4\text{kHz}}{2048} = 11.9\text{Hz}$

3.3.4 Comparing Time-Domain vs. Frequency-Domain approaches

Each method has distinct advantages and trade-offs:

- Time-domain averaging (LIA):
 - achieves a lower noise floor with less processing power
 - analyzes only one frequency point at a time
- Frequency-domain processing (FFT):
 - analyzes N frequencies at once
 - has a longer integration time due to the ENBW of the windowing function.

The choice of time-domain averaging (classic LIA-style processing) versus frequency-domain analysis (FFT-based processing) is a fundamental trade-off in RF measurement. Although both methods can

extract amplitude and phase from a noisy signal, they differ in terms of sensitivity, flexibility, and spectral resolution.

Unlike LIAs, which was discussed in previous sections, a vector signal analyzer (VSA) is designed for general-purpose broadband analysis RF. They effectively function as a high-speed digitizer with real-time DSP. Their broadband nature lends them usefulness for applications requiring modulated signal analysis, broadband spectral monitoring, and multi-tone measurements. Their broadband design comes at the cost of higher receiver noise figures (NFs), wider IBWs, and increased computational complexity compared to LIAs; however, their configurable resolution bandwidth (RBW) also enables them to have a “zero-span” mode feature where they mimic an LIA single-tone measurement, though they will not beat a purpose-built LIA. In most respects, with respect to the base instrument, commercially available software-defined radios (SDRs) are very similar to VSAs.

3.4 Implementation for a SDR, VNA

To implement a VNA using an SDR, one needs to use two receiving channels, one for the reference or a-wave (ref) wave and one for the measured or b-wave (meas) wave (see Chapter 4 for more details). For now, assume that there are two phase-matched channels. The SDR will output two I/Q signals. In the previous sections, how to extract the magnitude and phase of a tone in the presence of noise was discussed, for which two main methods were considered, a time-domain averaging method and a frequency-domain FFT method.

3.4.1 Time-Domain method implementation details

For each channel (ref and meas), the following steps should be followed:

- if using a direct conversion architecture, apply a high-pass filter (HPF) (to remove DC offset)
- Combine the I/Q components into a complex signal
- Mix the signal down to baseband

- Use a moving average signal with two circular buffers of fixed length (one for I and the other for Q)

The software will then take the complex ratio $S_{ij} = \frac{b_i \text{ (meas)}}{a_j \text{ (ref)}}$ where S_{ij} is the scattering parameter from port j to port i (and is explained in detail in the next chapter). The scattering parameter is the power ratio between the measured and reflected wave. If $i = j$, this is a reflection coefficient; otherwise, it is a transmission coefficient.

This project did not use the time domain approach, but it can be quite effective, especially for long-duration low sample-rate measurements, like a LIA. The moving average window acts as a LPF for the signal and also fixes the integration time. By adjusting the length over which the signal is averaged, the integration time is adjusted, which allows for a trade-off between noise performance and signal tracking speed.

3.4.2 Frequency-Domain method - A naive implementation and why it still worked

The initial prototype demonstrated that the ratio measurements exhibit a high degree of tolerance to suboptimal conditions. In this configuration, the signal frequency was not aligned with a bin center DFT and no windowing was applied. The sampling parameters – specifically, the number of samples and the sample rate – were selected heuristically to yield visually consistent results, rather than through rigorous optimization. Quantitative evaluation metrics such as noise floor, variance, or worst-case error were not systematically analyzed at this stage.

One might mistakenly argue, for instance, that in the following equation

$$\frac{w(t) \cdot s_2(t)}{w(t) \cdot s_1(t)} = \frac{s_2(t)}{s_1(t)}$$

That the windowing function cancels out in the ratio and is therefore unnecessary. This expression is mathematically valid under certain assumptions, and the ratio remains unaffected by uniform scalloping loss across both signals, but such an approach does not account for spectral leakage. The absence of windowing can thus introduce significant distortions. In particular, the method worked

only because it relied on a ratio measurement; absolute measurements under the same conditions would likely have failed to produce meaningful results.

3.4.3 Frequency-Domain method - How to do it well

First, the receiver should be well characterized (see Chapter 2). Then, the appropriate trade-offs between measurement speed and noise (both mean level and variance) can be made using the methods introduced earlier in this chapter. Then:

- Frequency-shift the signal to baseband by multiplying by a complex exponential to reduce spectral leakage.
- Window the data to reduce spectral leakage (not just of the signal, but the noise and distortion, too)
- Calculate the DFT of the relevant bin using Goertzel's method, or by taking the FFT and selecting the correct bin.

Frequency shifting or mixing to the center frequency bin is necessary. Of the two, frequency-shifting by multiplying by a complex exponential is preferred. Using the frequency-shifting property of the FFT allows one to avoid mixing with direct current (DC) offsets or image frequencies in direct conversion architectures, or mixing with harmonics or intermodulation distortions, also. The shifting to the baseband moves the frequency to the center of the frequency bins, which enables one to get an accurate measure of the amplitude of the signal. Otherwise, the signal is offset from the peak (by scalloping loss) and the measured value will be lower by up to a few dB. Still, it is optional if you are measuring a ratio. Windowing is also important, as it reduces the impact of nearby noise or distortion signals. For this use case, a $\text{Hann}(\alpha = 2)$ window is a good place to start.

A faster alternative to computing the FFT for the full spectrum is to use Goertzel's algorithm to calculate DFT for a single bin. Rather than using FFT to compute the value of the bin, a faster option is to use Goertzel's algorithm, assuming that you only need one or a few bins (as may typically be the case for VNAs and LIAs). The performance difference can be seen in Figure 3.7 below, which shows the relative performance difference, assuming that only one Goertzel bin needs to be computed:

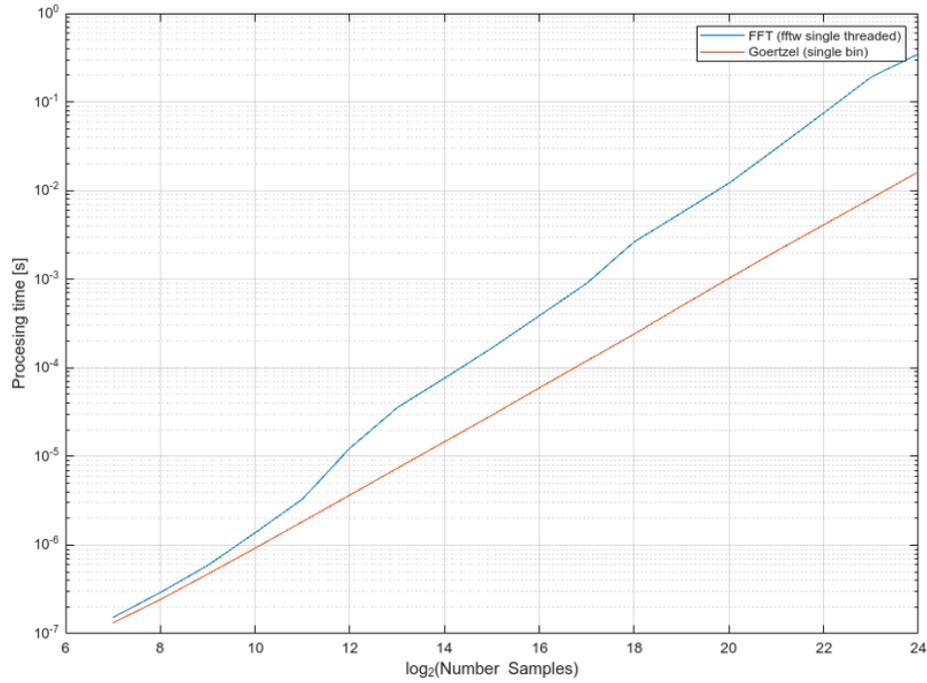


Figure 3.7: Goertzel algorithm performance Vs FFT for one bin

The red line on the plot represents a simple, unoptimized C++ implementation of Goertzel’s algorithm for a single bin. As expected, its performance scales linearly. If more than one bin needs to be computed, the complexity is $\Theta(M * N)$ operations, where M is the number of bins and N is the DFT size. The blue line on the plot is a single-threaded FFT implemented by FFTW [21] the “Fastest Fourier Transform in the West,” a highly optimized FFT implementation. Because FFT takes a divide-and-conquer approach, its performance scales as $\Theta(N \log_2(N))$ operations. You can see that, especially for larger N (2^{16} samples or more), the time difference can become quite significant, with the FFT taking ~ 1 ms and Goertzel’s finishing in tens of microseconds. With larger N , the difference becomes even more pronounced.

3.4.4 DSP Approach to Measurement

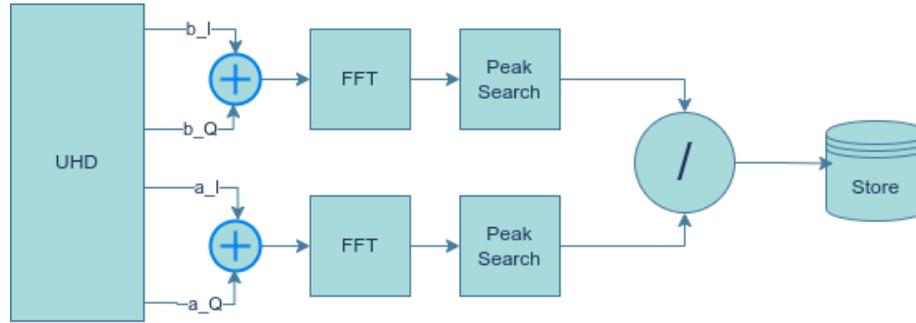


Figure 3.8: Early, naive approach to processing the IQ data

The incoming signals are complex: $a_I + ja_Q$ and $b_I + jb_Q$, but what does this actually look like? Observe in Figure 3.9 below what the I/Q data of the ref and meas waves look like in the time domain, each a simple tone – a complex exponential: $A \cdot \exp(j2\pi f_{IF}t + \phi)$ – looks like in the time domain (note: it is actually a complex signal because it can be a positive or negative frequency and is not symmetric). Note that the imaginary-valued components of the signals should ideally have the same amplitude as the real-valued components and should ideally delay the signal 90° . The I/Q components might not have the same amplitude as each other due to subtle differences in the analog hardware. In this example, you can see that the a-wave amplitude is about 0.8 compared to the b-wave at 0.7, and that the b-wave lags the a-wave by about 30° .

If there is no significant signal present at the image frequency, which will usually be the case for VNA measurements, one can take some small shortcuts with the processing of the data. The imaginary portion of the wave can be discarded, leaving behind a real-valued (symmetric) signal. This likely results in noise and images folding across $f = 0$. In direct conversion receivers such as the SBX-120, there can be an image frequency component as strong as -20 dBc, which is typically the strongest spurious signal present. Because mixing with a real-valued signal causes both the signal and its image to fold to the target frequency, this means that processing should consider the full complex data, even though the I/Q data, at a glance, seem redundant. Therefore, it is actually preferable to mix with a complex exponential. Rather than creating a sum and a difference (as is the case with real mixers), the complex exponential shifts all frequency data without creating images.

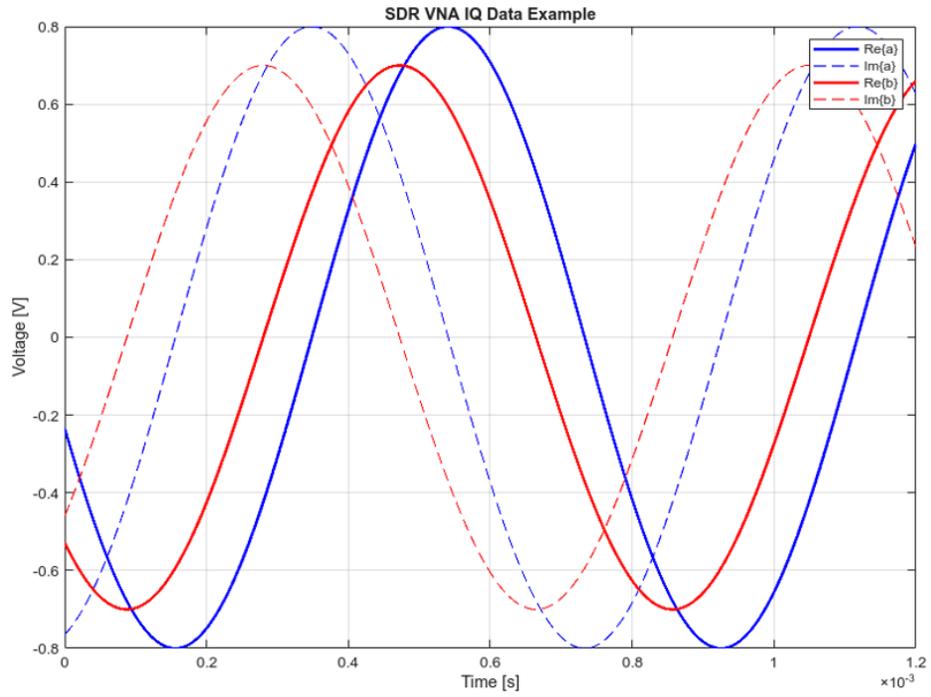


Figure 3.9: Example IQ Data

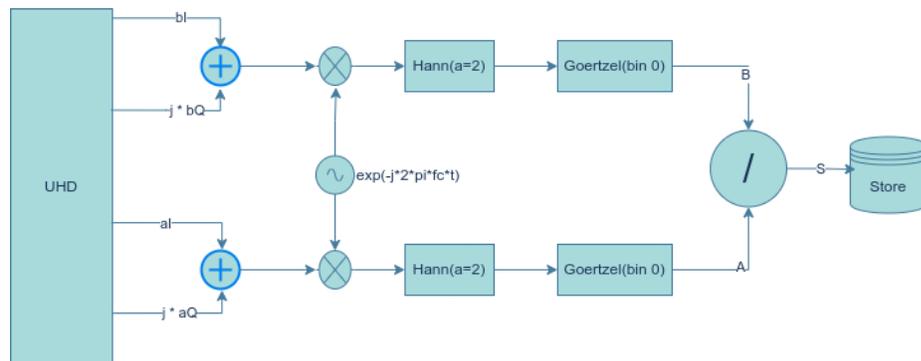


Figure 3.10: Revised DSP Block Diagram

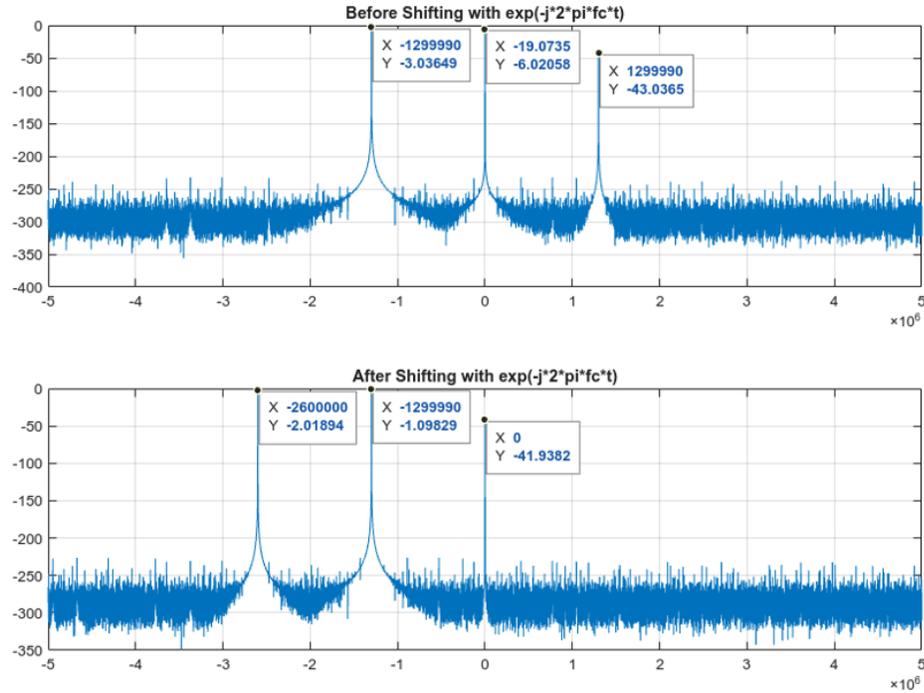


Figure 3.11: Multiplying by complex exponential produces frequency shift

A revised block diagram is shown in Figure 3.10 above. It shifts the entire frequency spectrum to the left by f_c (as depicted in Figure 3.11 below). The signal is then windowed, and the DFT is computed for a single bin (DC) using the Goertzel method. Finally, the ratio between the meas and ref signals is calculated and stored for later use. Compared to the previous approach, this is improved because it suppresses leakage and eliminates scallop loss. It is also robust against interferers because it utilizes frequency shifting instead of mixing by a real sinusoid.

Assuming that the signals are fully settled, this approach works quite well. See Figure 3.12 below and observe that this long-time measurement (900 seconds) took approximately 140 seconds to converge on a phase value. It may be possible to track this “settling drift” and characterize it, and perhaps find a way to prevent it or at least detect it. In principle, tracking drift could be done by utilizing a short-time frequency transform and tracking how long the phase takes to converge. Unfortunately, with a drift rate of about 0.02 degrees / second, it is not feasible in practice to track the phase with a sufficiently large FFT to suppress the phase noise sufficiently to observe the drift in any short period. If the behavior were consistent, it could be accounted for, but thus far, the behavior seems to be sporadic, resulting in occasional phase errors up to a couple of degrees.

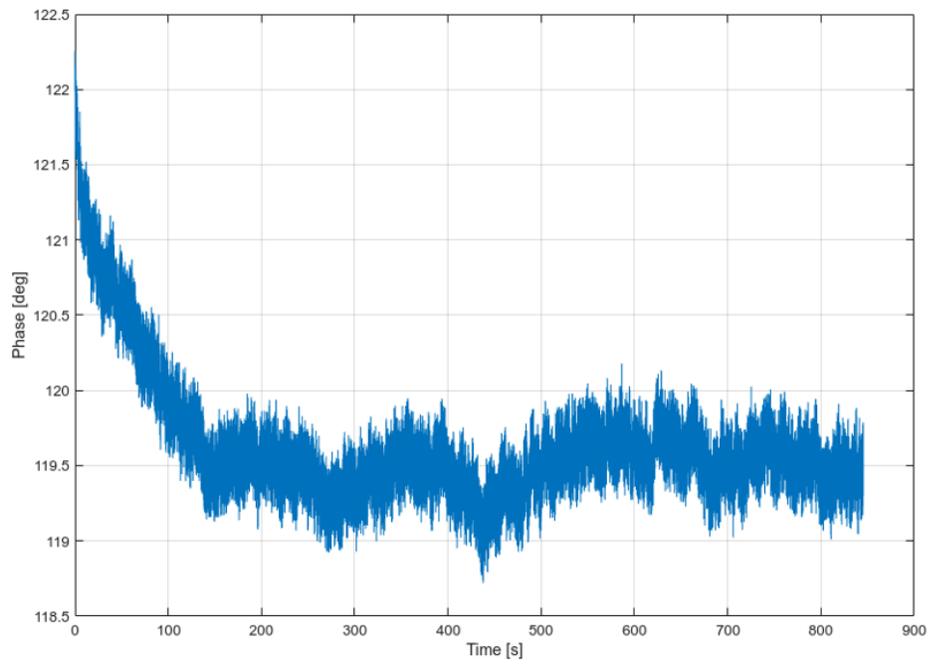


Figure 3.12: Phase Difference Sometimes Takes Significant Time to Settle

3.5 Chapter Conclusion

This chapter explores the DSP techniques essential for LIAs and VNAs. It began with a foundational discussion on Fourier analysis detailing how continuous signals are transformed into the frequency domain. Next, the critical role of sampling in DSP is covered, including the importance of the Shannon-Nyquist sampling theorem and its role in preventing aliasing. The chapter progresses to an elaborate discussion of the theoretically useful DTFT, the practically realizable DFT, and the extremely efficient FFT algorithm for computing the DFT.

The topic of spectral leakage and scalloping loss is covered in detail, including windowing to mitigate leakage and mixing to a bin center to prevent scalloping loss. The tradeoffs of various windowing functions are discussed, including their widths of the main lobe (and their effect on ENBW), the levels of the side lobe and the rate of sidelobe roll-off. A brief discussion followed, discussing practical guidance on how to choose windowing functions to meet specific needs.

The latter part of the chapter shifts focus to practical implementation techniques for measuring tones in the presence of noise. The chapter introduces a candidate time-domain technique (untested) and a frequency-domain technique. For the latter, the pitfalls of a naive implementation were discussed, followed by how to do it correctly. Then, some considerations for optimizing this technique for speed and practicality were discussed.

Chapter 4

VNA Background

Vector Network Analyzers (VNAs, in older literature, often called Automatic Network Analyzers or ANAs) are valuable tools for testing, characterizing, and troubleshooting radio frequency (RF) electronics. This chapter discusses the underlying theory of VNA operation, including two-port networks, scattering parameters, flow graphs, and calibration. The chapter then discusses hardware considerations for practical VNA design.

4.1 Two-Port Networks and Network Parameters

One of the most valuable concepts for characterizing systems is the concept of the "black box." Given an excitation, a system will respond characteristically. A one-port network (one-port) is the most straightforward black-box device, as it only has one pair of terminals. The incident wave is either absorbed or reflected (perhaps with a phase shift). In a two-port network (two-port), there are four possibilities: the reflection from the first port, the transmission from port 1 to 2, the transmission from port 2 to 1, and the reflection from the second port. Although an N port network may exhibit coupling between all ports, it can be fully characterized using sequential two-port measurements by appropriately terminating the other ports, which is how multiport VNA measurements are usually performed. [22]

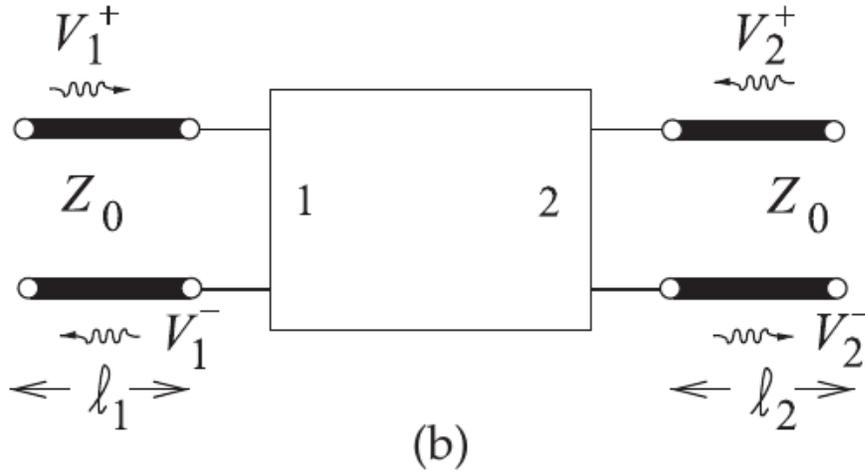


Figure 4.1: A two port network with transmission lines at the ports. Figured copied from [22]

In circuits, the relationship will generally discuss the incident and reflected voltage and current waves, measured at the port, such that $V_n = V_n^+ + V_n^-$ and $I_n = I_n^+ - I_n^-$ at the n^{th} port plane. They are related to each other through the impedance Z by the impedance matrix:

$$\begin{bmatrix} V_1 \\ V_2 \\ \dots \\ V_N \end{bmatrix} = \begin{bmatrix} Z_{11} & Z_{12} & \dots & Z_{1N} \\ Z_{21} & & & \dots \\ \dots & & & \dots \\ Z_{N1} & \dots & \dots & Z_{NN} \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ \dots \\ I_N \end{bmatrix}$$

where $Z_{ij} = \frac{V_i}{I_j} |_{I_k=0 \text{ for } k \neq j}$. The impedance Z_{ij} can be found by driving the port j with current I_j and measuring the open circuit voltage V_i at the port i when the open circuits terminate all other ports. The impedance matrix above fully characterizes the "black box" device under a given set of conditions. There are numerous equivalent representations of network parameters other than impedance parameters that, depending on the conditions, may be easier to use, manipulate, or measure, including admittance, hybrid, ABCD, scattering, and transfer parameters, among others. [9]

When dealing with practical devices, especially at microwave frequencies, it can be difficult to measure both the voltage and the current directly. Directly measuring impedance or admittance matrices also necessitate terminating the unused ports with open-circuits or short-circuits, respectively, which can be a problem when measuring active devices. Additionally, realizing a high-quality short-circuit

or open-circuit is actually harder than providing a good-quality matched load. As such, it can be more useful to measure scattering parameters instead, which assume matched loads at the unused ports and require measuring the reflected and incident magnitude and phase of the traveling voltage waves, which refer to power flow rather than absolute voltage or current. The parameter S relates the incident voltage wave to the reflected voltage wave, resulting in the scattering matrix (or S matrix): [22]

$$\begin{bmatrix} V_1^- \\ V_2^- \\ \vdots \\ V_N^- \end{bmatrix} = \begin{bmatrix} S_{11} & S_{12} & \cdots & S_{1N} \\ S_{21} & & & \vdots \\ \vdots & & & \vdots \\ S_{N1} & \cdots & \cdots & S_{NN} \end{bmatrix} \begin{bmatrix} V_1^+ \\ V_2^+ \\ \vdots \\ V_N^+ \end{bmatrix}$$

(equivalently, $\mathbf{V}^- = \mathbf{S}\mathbf{V}^+$ where the bold terms are understood to be matrices). Here, a specific element is determined as $S_{ij} = \frac{V_i^-}{V_j^+} |_{V_k^+ = 0 \text{ for } k \neq j}$. S_{ij} is found by driving the port j with an incident voltage wave V_j^+ and measuring the transmitted or reflected wave V_i^- at the port i . The waves at all other ports are set to zero, which means that they should be terminated with matched loads to prevent reflections back into the DUT. [9]

The above formulation assumes that the reference impedance at each port is the same (usually normalized to 50Ω), which is often but not always the case. Scattering parameters can be generalized by introducing the diagonal impedance matrix Z_0 and the root power waves at the n^{th} port – a_n and b_n , such that $\mathbf{a} = \mathbf{Z}_0^{-1/2}\mathbf{V}^+$ and $\mathbf{b} = \mathbf{Z}_0^{-1/2}\mathbf{V}^-$, where a is referred to as the forward wave, reference wave or simply a wave, and b is called the backward wave, measured wave, or simply the b wave. The generalized S-parameters can then be written as $\mathbf{b} = \mathbf{S}\mathbf{a}$. [22].

The relations between different network representations (e.g., impedance parameters, S -parameters) are well understood, so converting from one form to another is trivial. Different network representations have different uses; for example, converting to $ABCD$ parameters makes cascading two-port networks trivial.

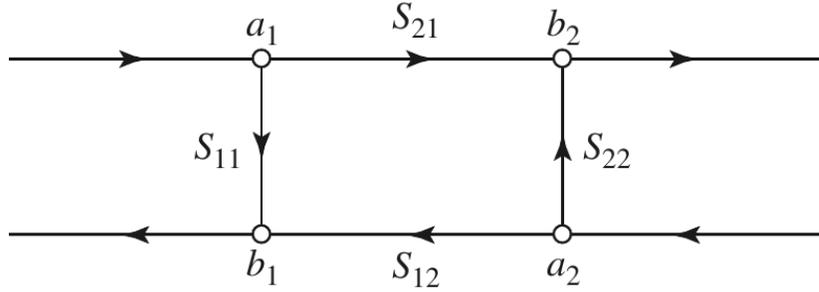


Figure 4.2: General Two-Port Signal Flow Graph. Figure from [9]

4.2 Signal Flow Graphs

Another useful concept related to networks is the signal flow graph (SFG). SFGs are directed graphs consisting of nodes and edges. Each port of a network has two nodes a_i and b_i , such that incident waves (a-waves) are incident on the a nodes and reflected waves are incident on the b nodes. Observe Figure 4.2, depicting the SFG for a general two-port network.

SFGs are useful for two reasons. First, they provide a visualization of the signal flow throughout the system. Second, they can be a powerful tool for analysis. SFGs can be solved iteratively by decomposition, in which they are iteratively simplified by following simple rules relating to series connections, parallel connections, self-loops, and splitting. They can also be solved directly by applying Mason's rule, which is shown in the equation below. [9].

$$G = \frac{y_{out}}{y_{in}} = \frac{\sum_{k=1}^N G_k \Delta_k}{\Delta}$$

$$\Delta = (-1)^0 + (-1)^1 \sum L_i + (-1)^2 \sum L_i L_j + (-1)^3 \sum L_i L_j L_k \dots$$

Where G is the complete gain between input node y_{in} and output node y_{out} , N is the total number of forward paths between y_{in} and y_{out} , G_k is the path gain of the k th forward path between y_{in} and y_{out} , Δ_k is the cofactor value of the k th forward path that does not include loops that touch the path, Δ is the determinant of the graph, L_i is the loop gain of each closed loop, $L_i L_j$ is the product of the loop gains of any two nontouching loops. [23] Solving Mason's gain equation by hand is quite laborious and potentially error prone. Fortunately, there are computer scripts that can quickly solve it if a declarative description of the signal flow graph is provided. [24]

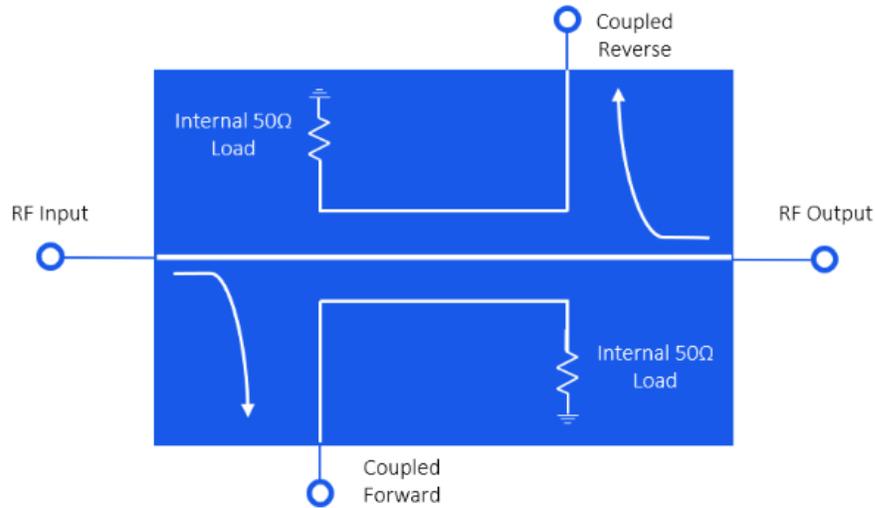


Figure 4.3: Dual Directional Coupler Diagram. Figure from [25]

4.3 General VNA topology

As was alluded to in the section on network parameters, especially S-parameters, VNAs measure data by wave separation. That is, they provide a signal excitation and then measure the forward and reverse power waves. The separation of forward and reverse power waves is achieved by using directional couplers. The coupled port outputs from the directional couplers are passed to the device's receivers for processing.

Directional couplers are three-port devices (four internal ports) with an input port, an output port, a coupled port, and (terminated) isolation port. The signal path from the input to the output has some insertion loss. The coupling value from the input to the coupled port is typically a fixed/designed value and can be as much as 20 dB below the input. The isolation port's termination is tuned to be as precise a match as possible because it plays a significant role in the directivity of the coupler. Directivity is a measure of how much of the reverse wave signal couples into the coupled port. Directional couplers will be discussed in greater detail later in this chapter. [25]

Bi-directional couplers and dual-directional couplers are also available. Bi-directional couplers have a symmetric design, with the main ports connected on one path and the coupled ports connected on another. A coupled line coupler with each of the four ports left open to use would be an example of this. The disadvantage of this is that mismatches in either of the coupled ports will easily propagate to the other coupled ports. Dual-directional couplers remedy this by effectively cascading

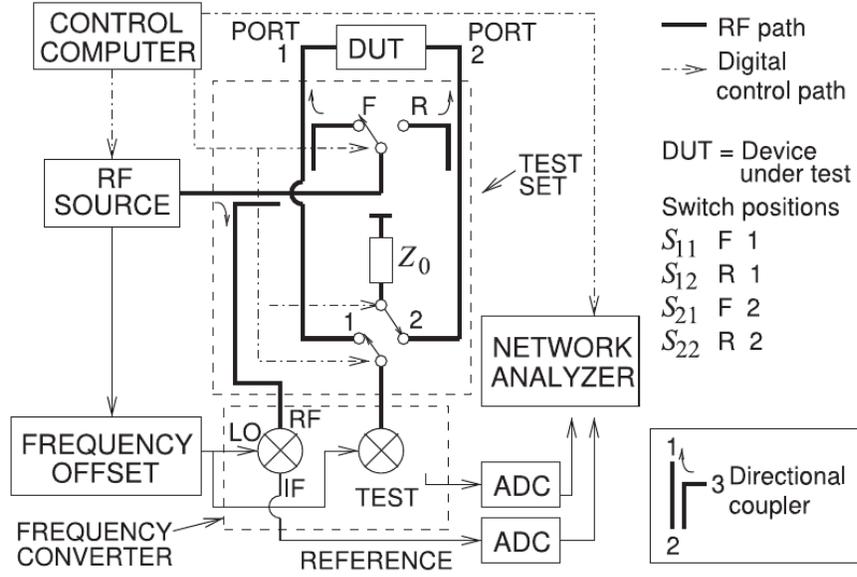


Figure 4.4: Switch-based VNA with Two Receivers. Figure from [22]

two directional couplers in a back-to-back configuration such that the coupled ports are independent of each other, with the other two internal ports terminated by a precision-matched load. A dual directional coupler can be seen in Figure 4.3. [25]

The full topology for a switch-based two-receiver VNA is shown in Figure 4.4 [22]. The directional couplers selectively couple forward or backward traveling waves. This topology allows a single RF source to be shared between two or more ports via a switch. A directional coupler can be placed in front of the transmit path switch so that the a-wave path is shared between ports; with this approach, a N port VNA can be constructed with only $N + 1$ couplers and two switches, one for the transmit path and one for the receive path. The primary disadvantage of this approach is that certain error terms cannot be measured with this topology, particularly the switching terms, which require measuring things like S_{11} with an excitation at port 2. The two-receiver VNA can be adjusted to achieve full generality while still sharing the RF source by including three switches – one for the a-wave, one for the b-wave and one for the RF source – using $2N$ directional couplers; this project utilized this approach and is shown in Chapter 5.

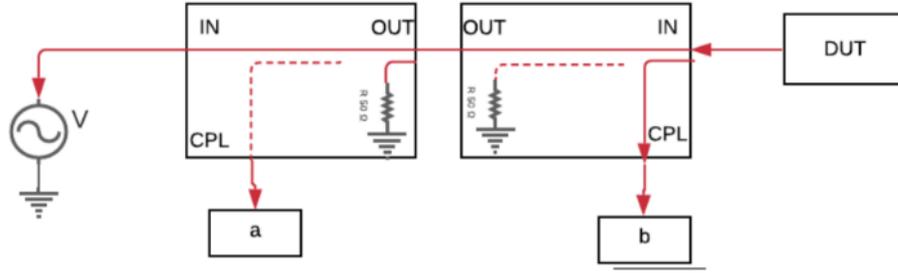


Figure 4.5: Directional couplers with imperfect coupling

4.4 Measurement Errors

One aspect that has not been discussed yet is why a VNA performs vector (complex) measurements as opposed to scalar measurements. The reason is that it provides the full (complex) characterization of devices under test (DUTs) and because it enables a complex calibration.

Detailed error models will be discussed later in this section, but before diving into that, let us intuitively consider some of the common sources of measurement error. In the previous section, directional couplers were discussed and their directivity was mentioned but not formalized. An ideal directional coupler would provide infinite directivity – backward coupling would not be allowed. In real directional couplers, the directivity is finite, with the best directional couplers achieving a directivity of better than 40 dB. Directivity is then an FOM for how well a coupler can discriminate forward from backward waves. It is defined positively assuming logarithmic values as $D = S_{31} + S_{21} - S_{32}$ where S_{31} is the coupling ratio, S_{21} is the insertion loss, and S_{32} is the isolation. As such, the best directional couplers need to have excellent isolation, moderate coupling factor, and low insertion loss.

The error introduced by the imperfect directivity of real directional couplers is intuitive once visually considered. For example, assume that the voltage source is applying a zero-dBm signal to the system. At the first coupler, a small amount of signal is coupled to the a-wave port, a tiny amount leaks to the isolation port(-D), and the mainline signal incurs some insertion loss. At the second coupler, which is oriented backwards, a small amount of signal is coupled to the isolation port, a tiny amount is leaked to the b-wave port, and the mainline signal again incurs some insertion loss. At this point, the signal reflects off the DUT and starts going the other way. On the second pass through the second directional coupler, a small amount of signal is coupled to the b-wave port, a tiny amount

leaks to the isolation port, and the mainline signal incurs some insertion loss. At the first directional coupler, a small amount of signal is coupled to the isolation port, a tiny amount leaks to the a-wave port, and the mainline again experiences some insertion loss. The signal levels at each of these stages are considered in Table 4.1 below, neglecting the reflections of the a, b and isolated ports, assuming that they are well matched. All quantities are expressed as losses relative to a zero dBm input signal in a logarithmic (dB) scale for notational convenience. A numerical example is provided by assuming a coupling factor (C) of 16 dB, 2 dB insertion loss (L), 30 dB directivity (D), and a DUT with matched load of 36 dB (Γ_{DUT}).

Table 4.1: Reflectometer error case with absorptive DUT

Stage	Main signal	a-wave port	b-wave port
1st pass 1st coupler	$L = 2$	$C = 16$	0
1st pass 2nd coupler	$2L = 4$	$C = 16$	$L + D = 32$
2nd pass 2nd coupler	$3L + \Gamma_{DUT} = 42$	$C = 16$	$L + D = 32$
			$2L + \Gamma_{DUT} + C = 56$
2nd pass 1st coupler	$4L + \Gamma_{DUT} = 44$	$C = 16$	$L + D = 32$
		$3L + \Gamma_{DUT} + D = 72$	$2L + \Gamma_{DUT} + C = 56$

In the above case, the main source of error was directivity, with a leakage signal at the b-wave port of about -32 dB, while the desired signal level at that port was -56 dB. Also, by the time the mainline signal returns to the source, it is sufficiently attenuated so that any re-reflected signals are negligible. However, what if the matched load DUT was replaced with a flush short ($\Gamma_{DUT} = 0$ dB)? The case of a highly reflective DUT is considered in Table 4.2.

Table 4.2: Reflectometer error case with reflective DUT

Stage	Main signal	a-wave port	b-wave port
1st pass 1st coupler	$L = 2$	$C = 16$	0
1st pass 2nd coupler	$2L = 4$	$C = 16$	$L + D = 32$
2nd pass 2nd coupler	$3L + \Gamma_{DUT} = 6$	$C = 16$	$L + D = 32$
			$2L + \Gamma_{DUT} + C = 20$

Stage	Main signal	a-wave port	b-wave port
2nd pass 1st coupler	$4L + \Gamma_{DUT} = 8$	$C = 16$	$L + D = 32$
		$3L + \Gamma_{DUT} + D = 36$	$2L + \Gamma_{DUT} + C = 20$

In the case of the flush short load in this example, the desired signal (-20 dB) was about 12 dB higher than the strongest error signal (-32 dB). However, note that the reflected mainline signal incident on the source port is at -8 dB, so a re-reflection from the source must be considered. For example, let us assume a relatively poor source match of $\Gamma_{SRC} = 10$ dB and observe the results in Table 4.3.

Table 4.3: Reflectometer error case with reflective DUT (cont)

Stage	Main signal	a-wave port	b-wave port
3 rd pass	$5L + \Gamma_{DUT}$	$C = 16$	$L + D = 32$
1 st coupler	$+\Gamma_{SRC} = 20$	$3L + \Gamma_{DUT} + D = 36$	$2L + \Gamma_{DUT} + C = 20$
		$4L + \Gamma_{DUT} + \Gamma_{SRC} + C = 34$	
3 rd pass	$6L + \Gamma_{DUT}$	$C = 16$	$L + D = 32$
2 nd coupler	$+\Gamma_{SRC} = 22$	$3L + \Gamma_{DUT} + D = 36$	$2L + \Gamma_{DUT} + C = 20$
		$4L + \Gamma_{DUT} + \Gamma_{SRC} + C = 34$	$5L + \Gamma_{DUT} + \Gamma_{SRC} + D = 50$
4 th pass	$7L + 2\Gamma_{DUT}$	$C = 16$	$L + D = 32$
2 nd coupler	$+\Gamma_{SRC} = 24$	$3L + \Gamma_{DUT} + D = 36$	$2L + \Gamma_{DUT} + C = 20$
		$4L + \Gamma_{DUT} + \Gamma_{SRC} + C = 34$	$5L + \Gamma_{DUT} + \Gamma_{SRC} + D = 50$
			$6L + 2\Gamma_{DUT} + \Gamma_{SRC} + C = 38$
4 th pass	$8L + 2\Gamma_{DUT}$	$C = 16$	$L + D = 32$
1 st coupler	$+\Gamma_{SRC} = 26$	$3L + \Gamma_{DUT} + D = 36$	$2L + \Gamma_{DUT} + C = 20$
		$4L + \Gamma_{DUT} + \Gamma_{SRC} + C = 34$	$5L + \Gamma_{DUT} + \Gamma_{SRC} + D = 50$
		$7L + 2\Gamma_{DUT} + \Gamma_{SRC} + D = 54$	$6L + 2\Gamma_{DUT} + \Gamma_{SRC} + C = 38$

For the example of a highly reflective load, the error incident on the a-wave port after reflecting off the DUT, then source, and then coupling was worse than that caused by reflecting off the DUT

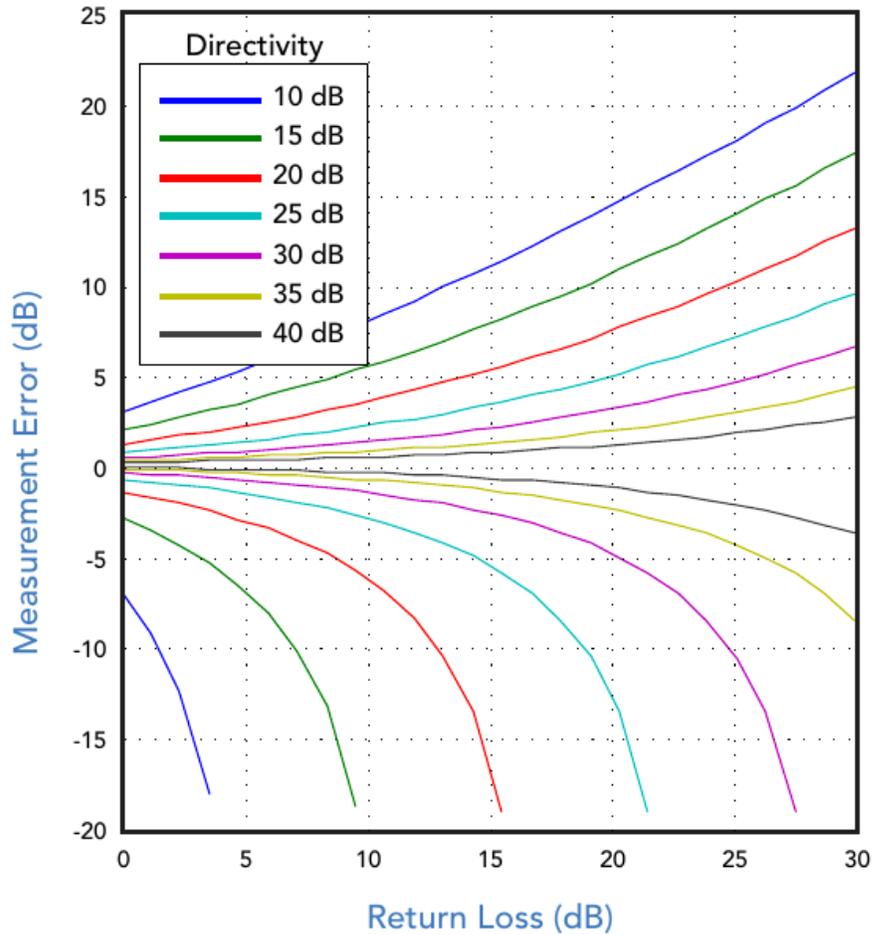


Figure 4.6: Return loss measurement error as a function of DUT return loss and coupler directivity. Figure from [26]

and then backward coupling. Another error term, the load match, can be considered in the case of transmission measurements between two ports. As the process is nearly identical, it is left as an exercise to the reader.

The impact of the coupler’s performance on measurement accuracy is well characterized and depends greatly on the directivity, which itself depends on the coupling factor and isolation and, to a lesser extent, on the insertion loss. [26] The worst-case return loss errors are shown in Figure 4.6. Observe that if the directivity is not about 15 dB better than the return loss being measured, the raw error becomes quite excessive. The error values are not always at the worst case; they rotate around the true value according to the equation below:

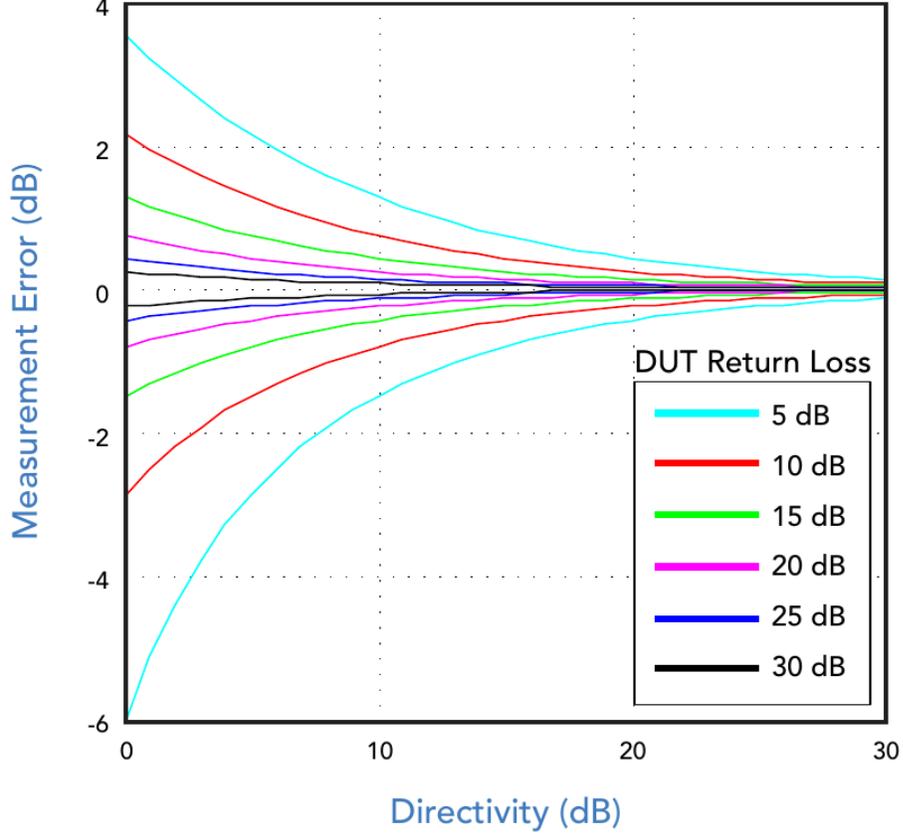


Figure 4.7: Forward Measurement Error as a function of return loss and coupler directivity. Figure from [26]

$$\begin{aligned}
 RL_m = RL + IL + 10 \log_{10} & \left(10^{\frac{-IL-RL}{10}} + 10^{\frac{-D}{10}} + 2 \cdot 10^{\frac{-IL-RL-D}{20}} \cos \theta \right) \\
 - 10 \log_{10} & \left(10^{\frac{-IL-RL-D}{10}} + 1 + 2 \cdot 10^{\frac{-IL-RL-D}{20}} \cos \theta \right)
 \end{aligned}$$

In contrast, error for forward power measurements, as shown in Figure 4.7, are generally very small – even for very small return losses – provided the directional coupler provides has at least 15 dB directivity.

The source match error can be taken into account as in Figure 4.8 [27]. Observe that at high DUT return loss, the directivity error dominates and that at low DUT return loss, the source match error dominates.

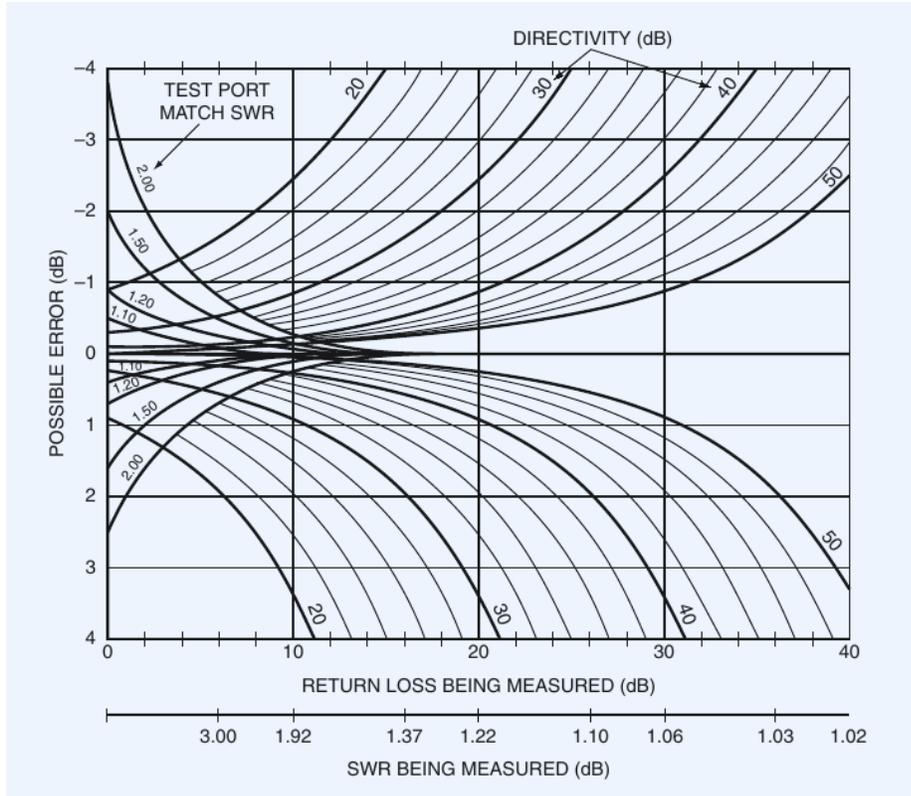


Figure 4.8: Directivity and Source Match Error as a function of DUT return loss, source port match, and coupler directivity. Figure from [27]

There are various other sources of errors in measurements. Some systematic error terms are the directivity, source match, and load match discussed above, as well as tracking error terms and crosstalk. [28] Some terms of random error include thermal drift, random noise, repeatability of the connector, and others. [29]

4.5 Error Modeling and Calibration

The procedure discussed above, considering the directivity and source match errors, could be codified as a signal flow graph, as described in the section of the same name. In fact, that is almost the complete model for error in a calibration of a one-port and is depicted in Figure 4.9 [30]. As you can see, the error box has a directed edge for directivity e_{00} ; source matching e_{11} and a reflection tracking term ($e_{10}e_{01}$), which absorbs the other sources of error not accounted for by directivity and port matching. Observe that, as in the example presented in the Measurement Errors section, the

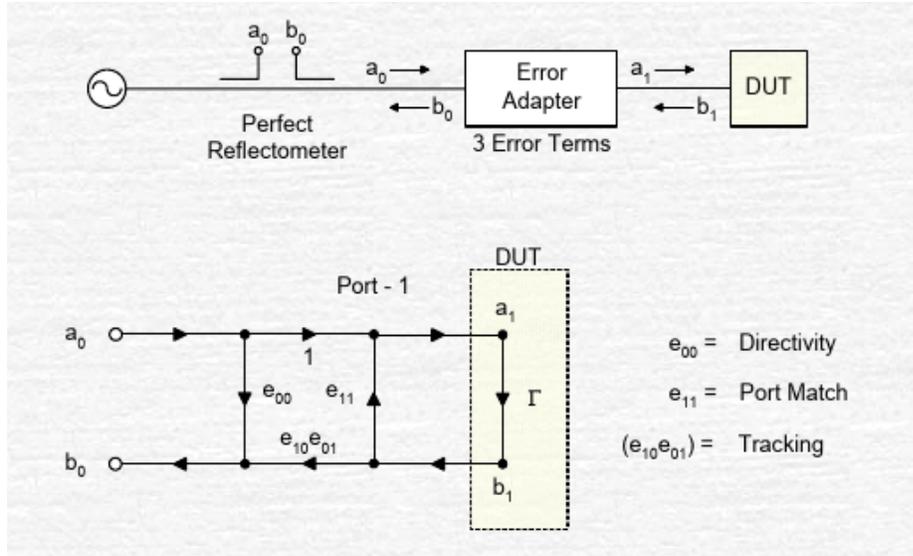


Figure 4.9: VNA One Port Error Model – Error box diagram and SFG. Figure from [30]

error signal can loop due to directivity or port match errors. Observe that this error model has three unknown terms. By measuring three known loads, called calibration standards, these unknowns can be found by solving a system of three equations, as shown in Figure 4.10. The most common set of standards for coaxial measurements is to use a short-circuit ($\Gamma = -1$), open-circuit ($\Gamma = +1$), and matched load ($\Gamma = 0$).

Two-port vector network analyzers (VNAs) are more complicated to model. The most common model is the 12-term error model Figure 4.11 and Figure 4.12, which includes terms for directivity, source match, load match, reflection tracking, transmission tracking, and crosstalk (leakage). A more complete model, the 16-term error model, includes additional leakage paths, including switch leakage, signals reflecting from the device under test (DUT) that leak to the transmission port, common-mode inductance, and others. The other error terms characterized by the 16-term model should be quite small, assuming that the switch provides high isolation.

The calibration procedure for the two-port is similar to that of the one-port but with some extra steps. First, a one-port calibration is performed at each port, characterizing the error terms at port 1 (directivity e_{00} , source match e_{11} , and reflection tracking $e_{10}e_{01}$) and port 2 (directivity e_{33} , source match e_{22} , and reflection tracking $e_{23}e_{32}$). Next, a transmission measurement in both directions with both ports terminated with matched loads is performed to measure the crosstalk error terms (e_{03} and e_{30}) directly. Finally, the transmission tracking error terms ($e_{10}e_{32}$ and $e_{23}e_{01}$) are characterized

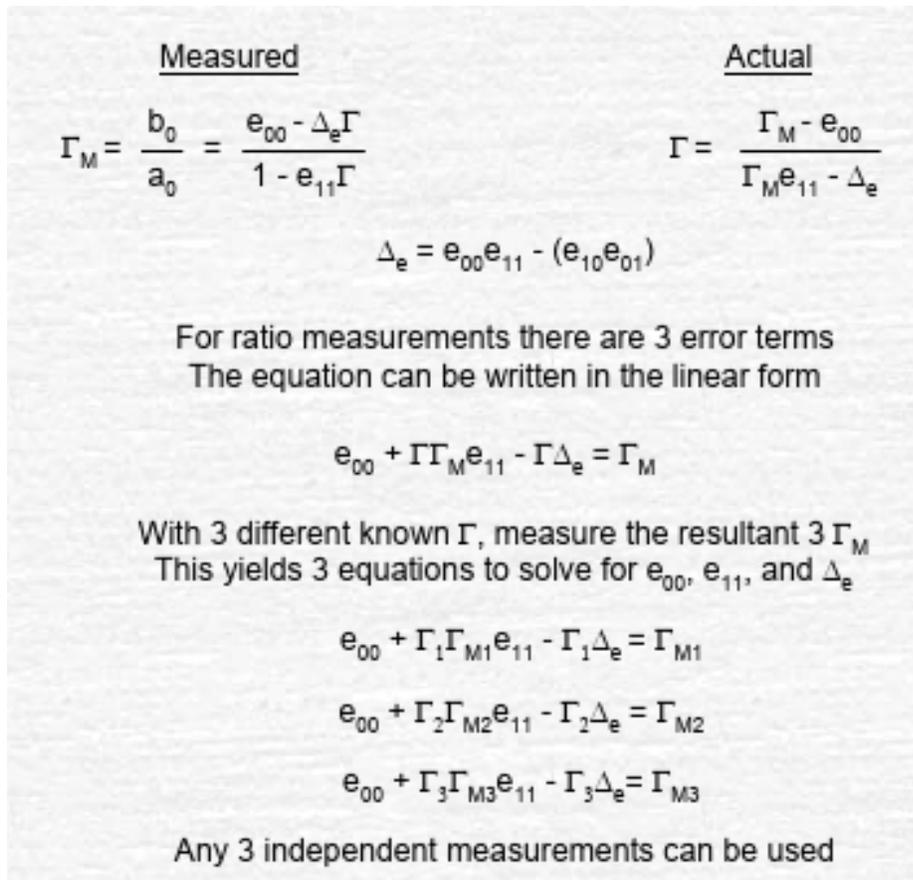


Figure 4.10: VNA One Port Error Model - Linear System of Equations. Figure from [30]

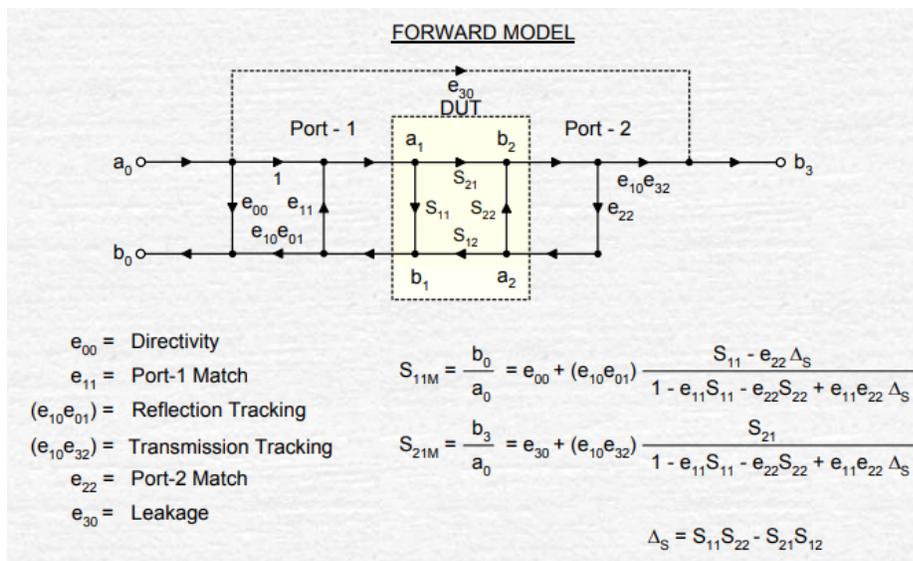


Figure 4.11: 12-Term Forward Error Model SFG. Figure from [30]

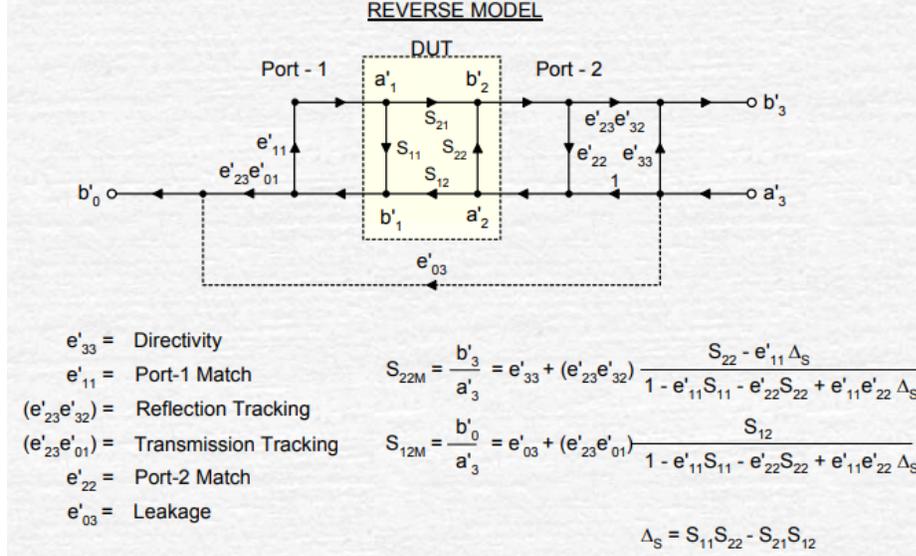


Figure 4.12: 12-Term Reverse Error Model SFG. Figure from [30]

by connecting ports 1 and 2 together. The most common version of this calibration scheme uses a short-circuit, open-circuit, matched load, and a well-characterized (preferably flush) through. [30]

No calibration procedure is perfect. Chapter 5's section on VNA Residual Error Analysis discusses residual errors that remain after calibration in the context of their measurement.

4.6 Chapter Conclusion

This chapter has presented the fundamental principles behind vector network analysis, including two-port network theory, definitions of scattering parameters, and SFG representations. The practical implications of VNA components and topology were discussed, particularly for directional couplers. The sources of systematic error were discussed and a few sources of random error were mentioned. An emphasis was placed on the importance of good coupler directivity and good source match.

The chapter concluded with an examination of the most common error models used in VNAs, including the one-port and 12-term two-port models, followed by a discussion of calibration standards and procedures that are used to correct measurement errors. These theoretical foundations and modeling approaches are necessary to understand the performance and limitations of VNA measurements and form the basis for the implementation and analyses described in Chapter 5.

Chapter 5

SDR-based VNA Prototype

5.1 Radio and Frontend Hardware

An software-defined radio (SDR) could be proposed to be used as a flexible platform for performing vector network analyzer (VNA) measurements. A low-frequency prototype utilizing commercial off-the-shelf (COTS) hardware was devised to demonstrate this functionality at frequencies up to about 6 GHz. It was to be a fairly conventional VNA in every respect, with the goal being ease of comparison and apples-to-apples comparison wherever possible. As such, there are two coherent detectors



Figure 5.1: Low Frequency SDRVNA Prototype with COTS hardware, early revision

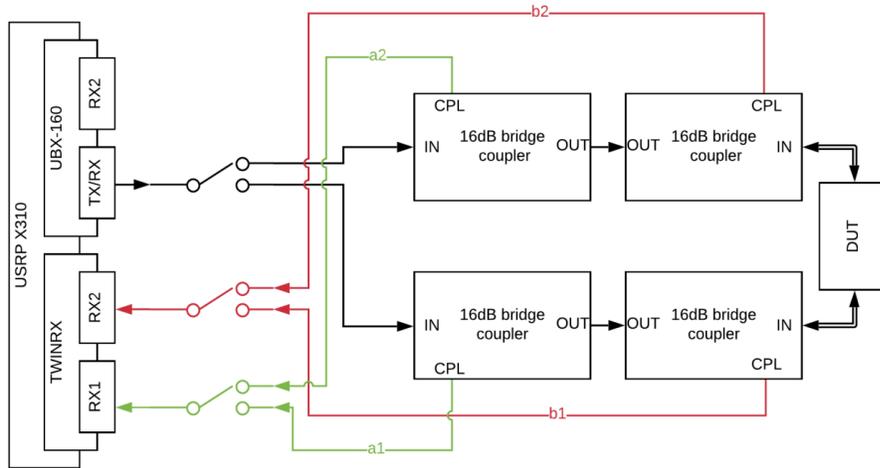


Figure 5.2: Block diagram of low frequency VNA prototype hardware

(one for the a-wave and the other for the b-wave) connected to back-to-back directional couplers (which are easier to find than high-directivity dual-directional couplers) and radio frequency (RF) switches. This allows sharing of the detectors and transmitter. Sharing the detectors results in slower measurement sweeps (it requires N^2 measurements per frequency instead of only N measurements), but it reduces the complexity and cost of the project.

Observe Figure 5.1, an early revision of the completed prototype, and Figure 5.2, which is an early block diagram of the same. This approach required 3 single-pull double throw RF switches and 4 directional couplers (two per port).

Ettus Research X310 radios were used because they were on hand and provide good performance. The TwinRX daughterboard was initially selected for receiving, because – on paper – it has much better performance than the other daughterboards that are available for the X310. It is a two-stage superheterodyne receiver that covers frequencies from 10 MHz to 6 GHz with 80 MHz integration bandwidth (IBW). In addition, it is the only radio frontend for the X310 that allows local oscillator (LO) sharing (the others synthesize signals from a common reference, but with separate LOs and phase-locked loops (PLLs)). The UBX-160 was initially selected as a transmitter because it covers the same frequency range; but was replaced by the SBX-120 daughterboards later, for reasons that will be apparent later in this chapter. Each of these three boards are recommended for phase-coherent operation by their manufacturer.

Arguably, the most important part of the frontend is the wave separation apparatus. As discussed in Chapter 4, the coupler directivity is one of the most important attributes for VNA measurement accuracy. As such, directivity greater than 40 dB is strongly preferred. A higher coupling factor is also desirable because it should reduce specific error terms. Lower insertion loss is desirable and can improve accuracy, but it is a secondary concern. Two Mini-Circuits, ZHDC-16-63-S+ directional bridges, were used per port, which provide a nominal directivity of about 30 dB, but the worst case is that the spec is 24 dB, so it may be worth binning them. Marki Microwave offers a similar coupler that is typically >38 dB directivity, but its specification allows 27 dB worst case, so they may also need to be binned – that is, buy more than you need and only use the best ones.

RF switches are advisable for making a two-port (or N-port) analyzer so that all hardware, except directional couplers and cables, can be shared between ports. The RF switches should be of the absorptive type, and the switching speed and isolation should be prioritized. Three single-pull N-throw switches are needed (where N is the number of ports): one for the transmit path, one for the a-wave path, and one for the b-wave path. If the selected switch has poor isolation, it can be improved by cascading a single-pull single-throw isolation switch. Low insertion loss is desirable but not critical. An easy interface to control from a computer is preferred, but anything that can be automated will suffice. Be aware that mechanical switches are relatively slow and can have significant flyback when turned off. This project used three MSP2TA-18-12BM+ switches, which provide isolation of 80-100 dB in this frequency range, 0.25 dB insertion loss, and a relatively slow switching speed of 20 ms. The control signal / input power is 12 V / 0.3 A, with fairly severe flyback (-50 V) necessitating well-protected control circuitry. A better choice, although more expensive, would have been the solid-state switch U2C-1SP2T-63VH. It provides excellent isolation of >100 dB, a very fast switching speed of 2 μ s, and can be controlled by USB, I2C, or SPI.

For cabling, semi-rigid cables are preferable, as they should be best at resisting movement and vibrations, though working with them can be rather cumbersome. Hand-flex cables are a good second choice and are easier to work with. High-quality test cables are nice, but not strictly necessary in the frequency range addressed with this prototype. Most of the measurements taken with this prototype used inexpensive and flexible test cables. For a permanent fixture, fully rigid cables might maximize performance.

Table 5.1 below shows a list of parts that were acquired for this project:

Table 5.1: Hardware used in SDR-based VNA prototype

Manufacturer	Identifier	Quantity
Ettus Research	USRP X310 SDR Platform	2
Ettus Research	TwinRX Daughterboard (radio frontend)	1
Ettus Research	UBX-160 Daughterboard (radio frontend)	2
Ettus Research	SBX-120 Daughterboard (radio frontend)	2
Mini Circuits	MSP2TA-18-12+ SPDT Mechanical RF Switch	3
Mini Circuits	ZHDC-16-63-S+ Directional Bridge	4

Also necessary is a SMA/3.5mm/2.92mm calibration kit. Throughout most of this work, an old Hewlett-Packard 85052D economy calibration kit was used. The calibration kit should preferentially match the connector being used for the measurements; this project used SMA. This kit provides a polynomial model for the standard definitions. Better results can be obtained by utilizing fully characterized standards, e.g., ones that provide measured S parameters performed with a regional or vendor laboratory.

For verification, coaxial airline standards are needed also; their length should be long enough so that several periods of the resultant standing wave are visible within the operating bandwidth. In this frequency range, an airline at least 30 cm long is desirable, but they can be hard to find with SMA connectors, so - for this work - 15 cm airlines were used instead. The 15 cm airlines produce a 1 GHz standing wave, so only four iterations of the standing wave were visible. There is also a trade-off between beaded or beadless airlines. Beaded airlines are more durable and highly repeatable, but their return loss (~ 35 dB) is lower than that of beadless airlines. Beadless airlines require skilled hands and are difficult to use, but if assembled correctly, they can achieve a superior return loss (~ 50 dB), which is highly desirable for verification standards.

The assembled prototype needs to have a rigid structure. Any movement of the cables introduces a random error in your calibration or measurement. Great care should be taken to make sure that the couplers, switches, and associated cables cannot move. The flexible test cables at the end are a necessary evil, but when precision is necessary, minimizing their movement as much as possible can be helpful. Observe that in Figure 5.1 at the beginning, the components are fixed to a 3D printed plastic plate, but the plate was not attached to the radio; as a result, it was prone to moving around slightly. These slight movements made calibration impossible. In Figure 5.4 and Figure 5.3, notice that not only is all the hardware screwed down to a metal plate, but the plate is also clamped to the side of the radio using L-brackets, machine screws, set screws and Loctite (to act as a stopper),

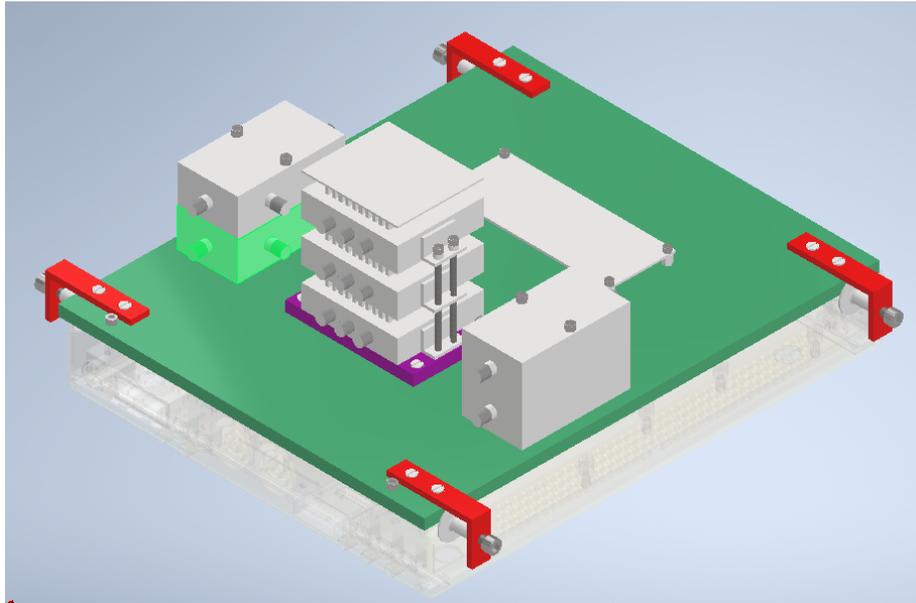


Figure 5.3: VNA frontend fixture design, including clamp-like structures on the corners to prevent unintended cable movement

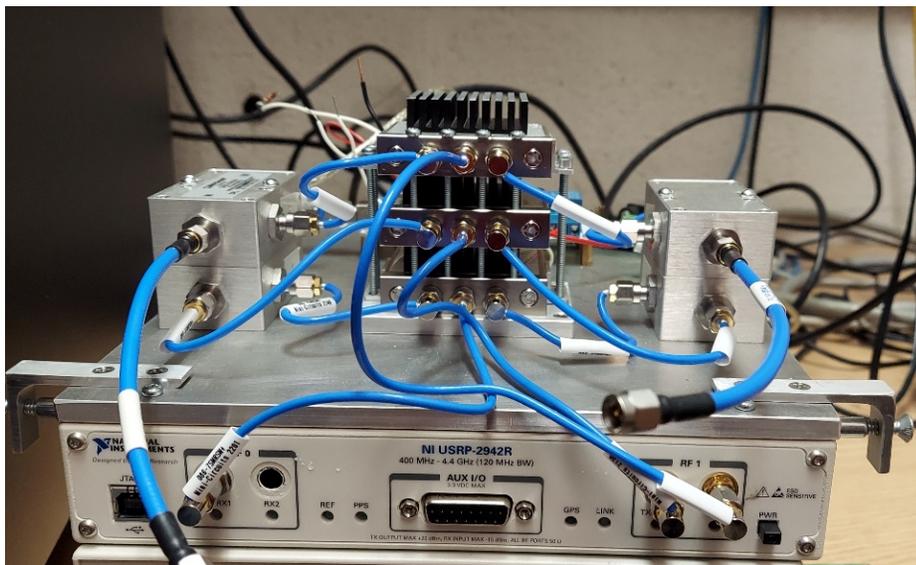


Figure 5.4: Low Frequency SDR-based VNA prototype with COTS hardware, later revision

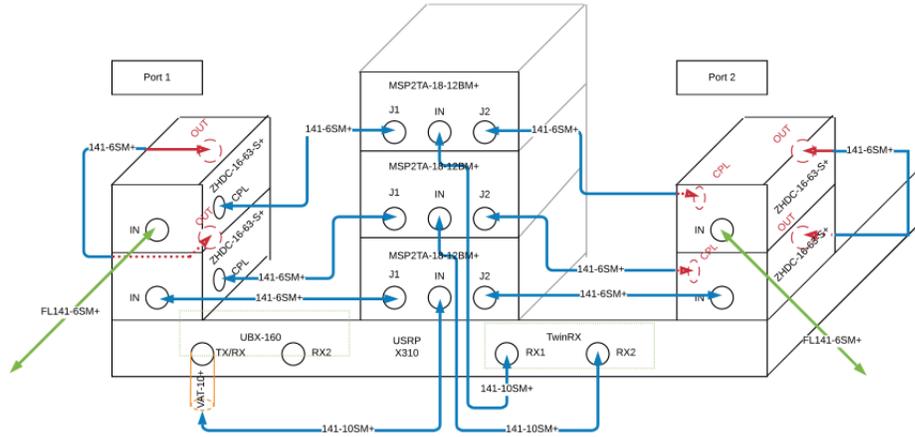


Figure 5.5: Low frequency $\backslash\text{gls}\{\text{sdr-based VNA assembly and wiring diagram}$

and weld nuts. The machine screw pushes the weld nut into the sides, which makes it clamp on the two surfaces (radio and plate) with enough force to keep them fixed together.

See Figure 5.5 for a (somewhat outdated) assembly/wiring diagram. Assembly and wiring are straightforward and mainly consist of turning screws and connecting wires. Do torque all SMA connections in order to obtain accurate phase measurements. The pair of couplers on the left is port one and the pair on the right is port two. The bottom couplers are for the a-wave and the top couplers are for the b-wave. This stack of switches from bottom to top is a-wave, transmit, and b-wave, respectively.

Given the use of mechanical switches in this prototype, a means of controlling the switches was devised, as shown in Figure 5.6 and Figure 5.7. Ultimately, it did not work for long because it did not adequately compensate for the flyback when the switches were turned off. A resilient diode such as a 1N4007 placed in parallel with the switch may have helped. In the end, in the interest of robustness, simplicity and functionality, GPIB-controlled bench-top power supplies were used instead.

After the prototype was assembled, it was characterized as a whole, as a 5-port network (TX, A, B, Port 1, and Port 2) with RF switches in each permutation of the on-off state. The nominal levels are summarized in Figure 5.8. Indigo cells are direct paths through the frontend; light blue boxes are coupling paths; yellow cells are directivity error paths; green cells are return loss paths, and red cells are paths limited by isolation. You will also see that the TX to B path appears as isolated too;

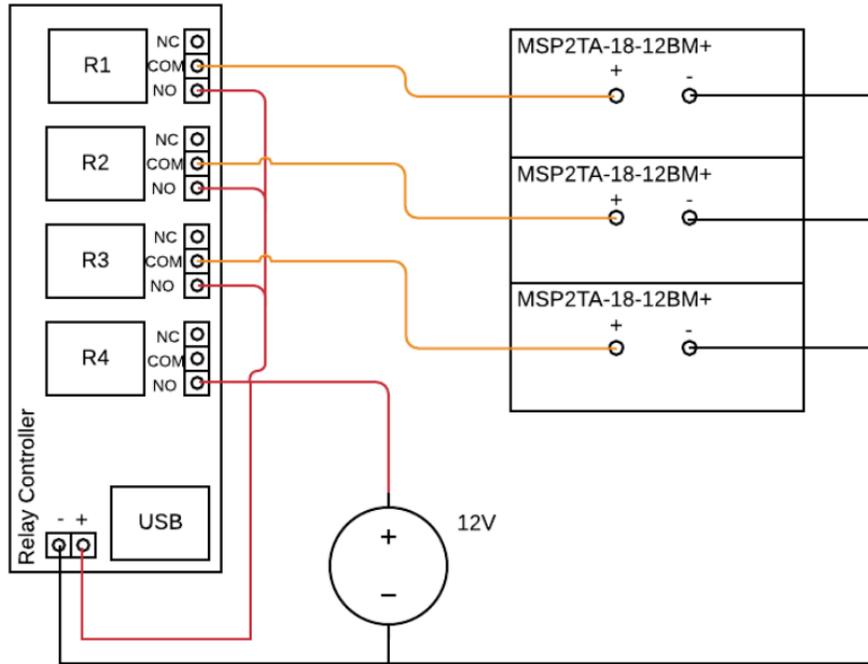


Figure 5.6: RF switch controller wiring diagram

this is because S-Parameters assume matched loads terminate the ports; were it terminated by, e.g., a short, the result would be similar to the TX to A case.

5.2 Brief History of This Project's SDR-based VNA Software

The software for controlling the experiment was continually developed to meet the growing needs of the project. Much of the focus was on the architecture, enabling it to more easily automate different kinds of sweeps and control different kinds of instruments, like signal generators, power supplies, and power meters, for e.g., power calibration sweeps. This is explained in Appendix 1. Discussion in this chapter will be limited to aspects pertaining specifically to controlling the SDR and extracting S-Parameters from the in-phase and quadrature (I/Q) data.

In the early days of the project, samples were captured using Ettus Research's "rx_samples_to_file" script, which dumps received I/Q data into a file. The I/Q data files were processed with MATLAB, and the results were consistent with expectations. Re-tuning. There are numerous disadvantages

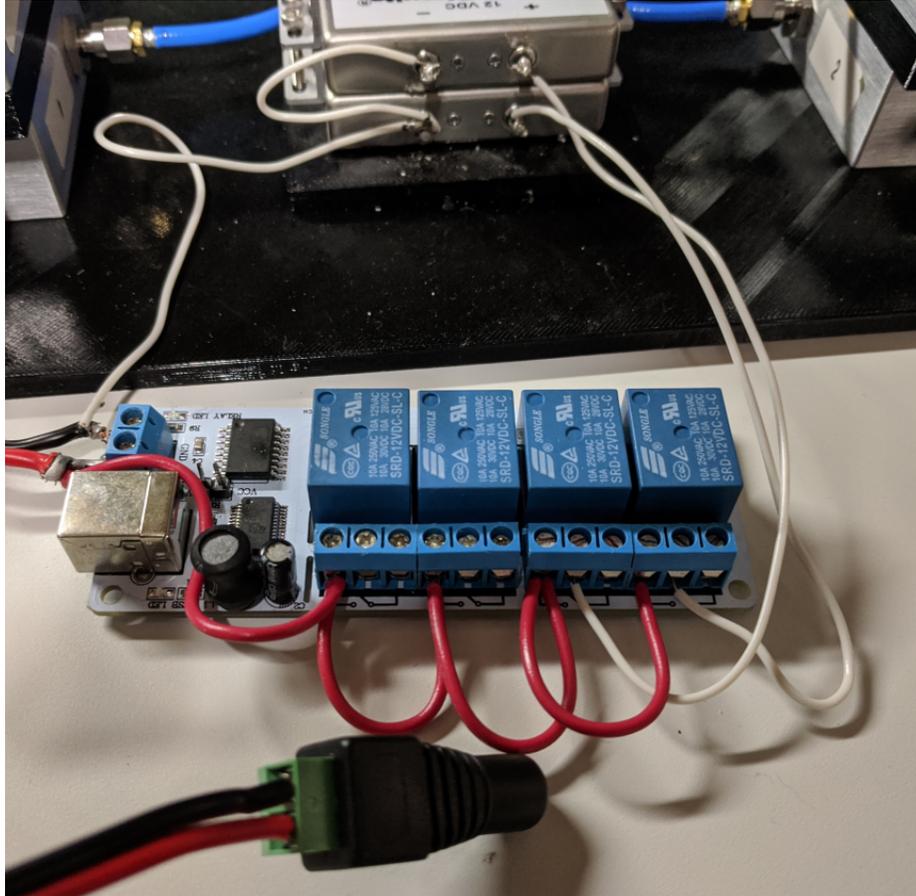


Figure 5.7: RF switch controller

to this approach. First, at each measurement point, the radio is completely reinitialized. As such, sweeps are executed at a rate of about 400 points per hour. Second, I/Q data are enormous, and storing them requires a large amount of disk space.

A Python script variation was created to address these issues and more. Utilizing a GNURadio "top block" script internally, it was able to process an entire sweep and save the results to a touchstone file— the typical file format used for storing S-Parameter data. Sweeps were still rather slow. There were also unresolved issues including random 90-degree phase jumps and random crashes.

Moving the project codebase to C++ greatly improved the measurement speed, with the new bottleneck being the switching and tuning speeds of the hardware, but stability was still a sore point
....

from	to	sw=s11	sw=s12	sw=s21	sw=s22
TX	TX	< -15 dB	< -15 dB	< -15 dB	< -15 dB
TX	A	~ -16 dB	~ -16 dB	~ -16 dB	~ -16 dB
TX	B	< -40 dB	< -80 dB	< -80 dB	< -40 dB
TX	P1	~ -4 dB	~ -4 dB	< -80 dB	< -80 dB
TX	P2	< -80 dB	< -80 dB	~ -4 dB	~ -4 dB
A	TX	~ -16 dB	~ -16 dB	~ -16 dB	~ -16 dB
A	A	< -40 dB	< -15 dB	< -15 dB	< -15 dB
A	B	< -80 dB	< -80 dB	< -80 dB	< -80 dB
A	P1	< -40 dB	< -40 dB	< -80 dB	< -80 dB
A	P2	< -80 dB	< -80 dB	< -40 dB	< -40 dB
B	TX	< -40 dB	< -80 dB	< -80 dB	< -40 dB
B	A	< -80 dB	< -80 dB	< -80 dB	< -80 dB
B	B	< -15 dB	< -15 dB	< -15 dB	< -15 dB
B	P1	~ -16 dB	< -80 dB	~ -16 dB	< -80 dB
B	P2	< -80 dB	~ -16 dB	< -80 dB	~ -16 dB
P1	TX	~ -4 dB	~ -4 dB	< -80 dB	< -80 dB
P1	A	< -40 dB	< -40 dB	< -80 dB	< -80 dB
P1	B	~ -16 dB	< -80 dB	~ -16 dB	< -80 dB
P1	P1	< -15 dB	< -15 dB	< -15 dB	< -15 dB
P1	P2	< -80 dB	< -80 dB	< -80 dB	< -80 dB
P2	TX	< -80 dB	< -80 dB	~ -4 dB	~ -4 dB
P2	A	< -80 dB	< -80 dB	< -40 dB	< -40 dB
P2	B	< -80 dB	~ -16 dB	< -80 dB	~ -16 dB
P2	P1	< -80 dB	< -80 dB	< -80 dB	< -80 dB
P2	P2	< -15 dB	< -15 dB	< -15 dB	< -15 dB

Figure 5.8: Frontend Nominal S-Parameters

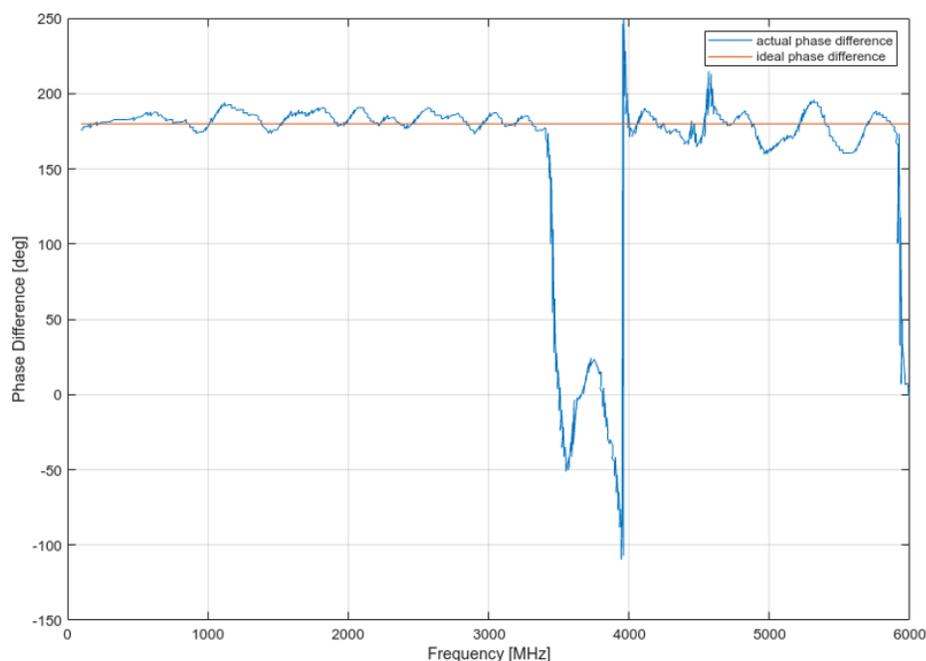


Figure 5.9: Uncalibrated phase-difference between open-circuit and short-circuit calibration standards, measured with UBX-160 transmitter, and TwinRX receivers

5.3 Early experiments, TwinRX quirks

The first functional experiments focused on looking at two metrics for the measurements. Firstly, the phase difference between a measured open-circuit and a short-circuit should be approximately 180° ; recall that this is how the calibration kit opens and shorts are specified. In addition to being open or short at direct current (DC), they maintain a fixed phase difference of 180° throughout their usable frequency range, give-or-take about 2°). Second, the magnitude difference between a reflective standard and a matched load should be as large as possible, which plays a role in the measurement of devices with high return loss.

The measured phase difference between an open-circuit standard and a short-circuit standard can be seen in Figure 5.9, and the magnitude difference can be seen in Figure 5.10. Below about 3.4 GHz, the phase is relatively stable at nearly 180° , and again above about 4 GHz; but in between, the phase difference drops to near zero, which is the worst-case scenario for the phase difference, as this means that the system cannot resolve the difference between two device under test (DUT) with completely opposite phase. The magnitude difference highlights problems in the same frequency range, with the magnitude difference between the reflective standards and the matched load also dropping to near

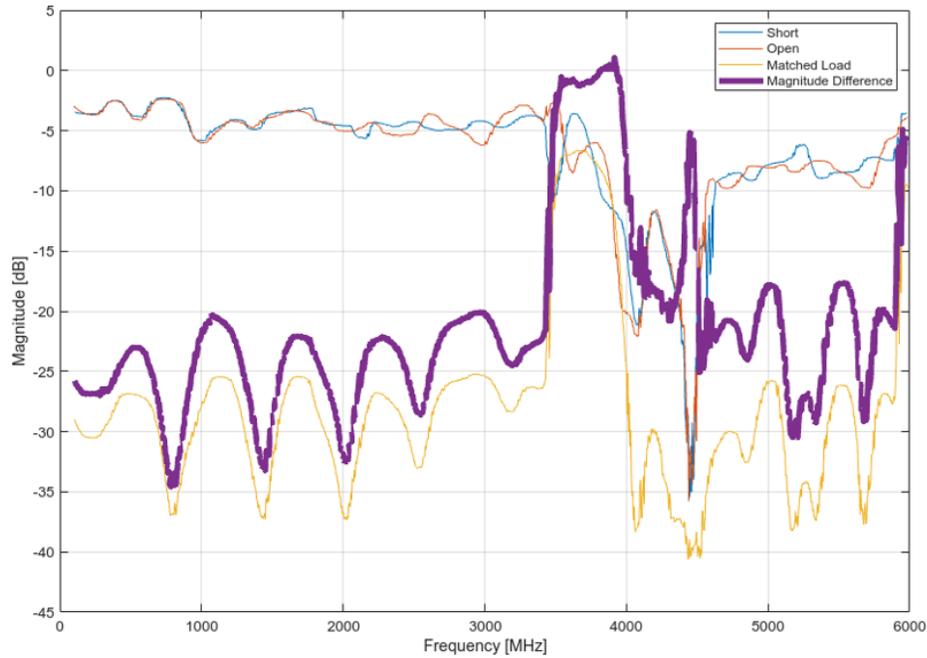


Figure 5.10: Uncalibrated magnitude difference between reflective and absorptive calibration standards measured with UBX-160 transmitter and TwinRX receivers

zero, which is the worst-case scenario for the magnitude difference, as this means that the system cannot resolve the difference between a device with low return loss and one with high return loss. Additionally, there is another point near 4.4 GHz that likewise seemed to be problematic.

A single coherent script was devised to greatly improve measurement speed by using a single, continuously running program. Keeping the radio initialized between sweep points greatly accelerated measurements. Before this, measurements were made by repeated calls to a program from a shell script, which moved at a glacial pace of about 400 measured data points per hour. Unfortunately, this uncovered issues with coherent tuning (a 90° phase ambiguity) and hardware crashes (field-programmable gate array (FPGA) exceptions) became commonplace. The latter had already been seen very occasionally, but less frequently, because the measurements were very slow. Based on advice from other users of the TwinRX receiver, the program was rewritten in C++ to obtain lower-level control over the hardware. The phase ambiguity was eventually resolved by tuning the receiver *twice* for each measurement, which was necessary because the command queue buffer of the X310 is too short to hold all the commands needed to do phase synchronous tuning.

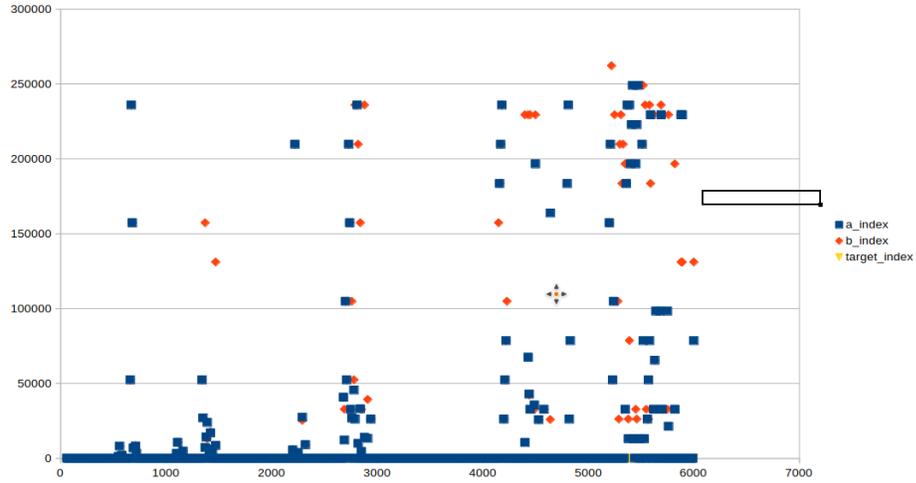


Figure 5.11: TwinRX intermediate frequency (IF) deviation from expected value

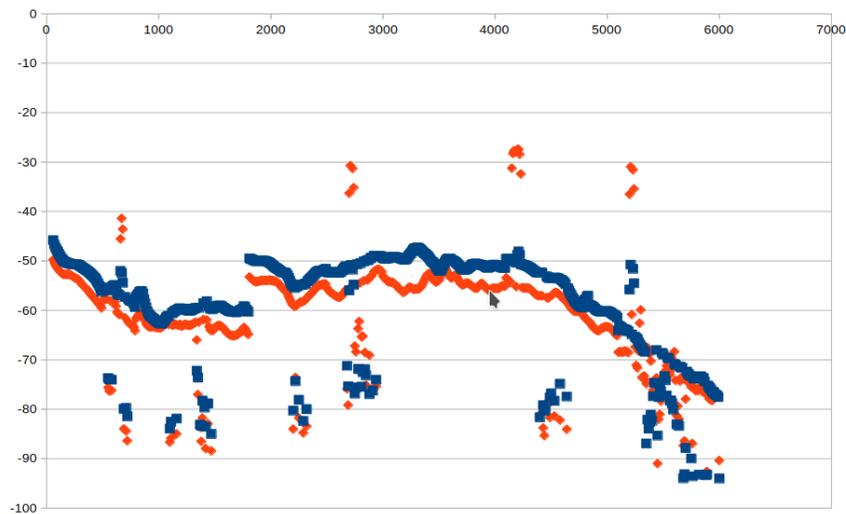


Figure 5.12: TwinRX a-wave and b-wave measured magnitudes of a short-circuit calibration standard

After completing the rewrite of the software in C++, the oddities of the previous data were investigated. It was found that the strongest signal present in the bandwidth was not necessarily the tone of interest, either on the a-wave, the b-wave, or both. The deviation of the tone from the expected IF frequency is depicted in Figure 5.11. The a-wave and b-wave magnitudes of a measured short-circuit are depicted in Figure 5.12 below. These data illustrate that the scope of the problems observed previously is wider than previously suggested, partly because the denser sweep allows more to be seen and partly because the a-wave and b-wave values are being observed directly instead of their ratio.

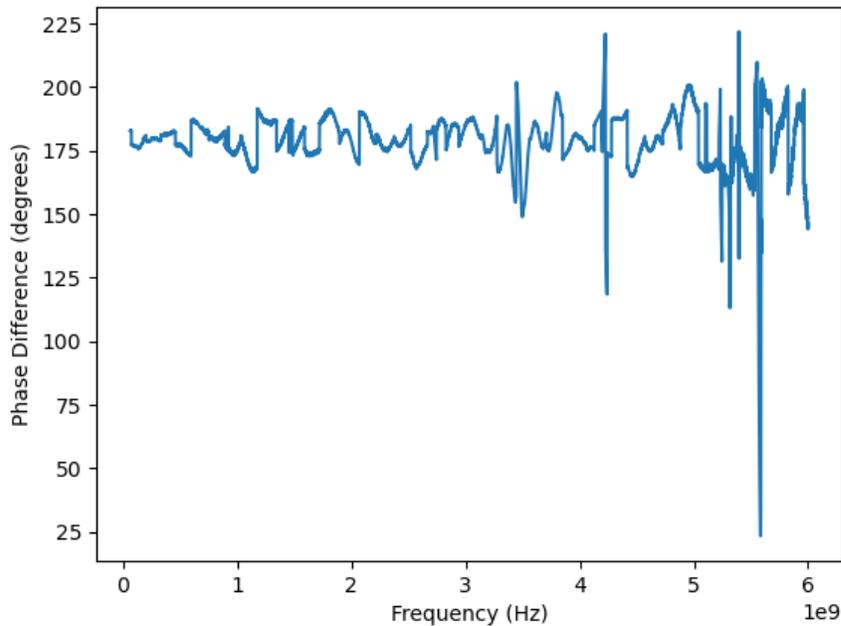


Figure 5.13: TwinRX measured phase-difference between open-circuit and short-circuit calibration standards, without calibration, after upgrading to UHD 4

Some of the issues were partially resolved by switching to UHD [31] version 4 when it became available. The phase difference between an open-circuit and a short-circuit are depicted in Figure 5.13 and are much improved after the upgrade to UHD 4. In Figure 5.14 and Figure 5.15, the pink regions represent areas of invalid data. Apart from the problematic areas, the magnitude and phase repeatability were shown to be reasonable, with standard deviations typically less than 0.5 dB for reflective loads and phase repeatability better than 1° at low frequencies.

Much later, the TwinRX issues were revisited and it was discovered that having a TwinRX and an UBX-160 in the same chassis is not supported by the manufacturer; however, the TwinRX problems were still present without the UBX-160. The frequency dead zones were checked and rechecked to ensure that the issues were repeatable. Currently, it is unknown whether the observed issues were the result of defective hardware. The combination of frequency dead zones paired with FPGA instability made it evident that change was needed. The natural solution was to try another radio. Later experiments switched to using the UBX-160 radios instead.

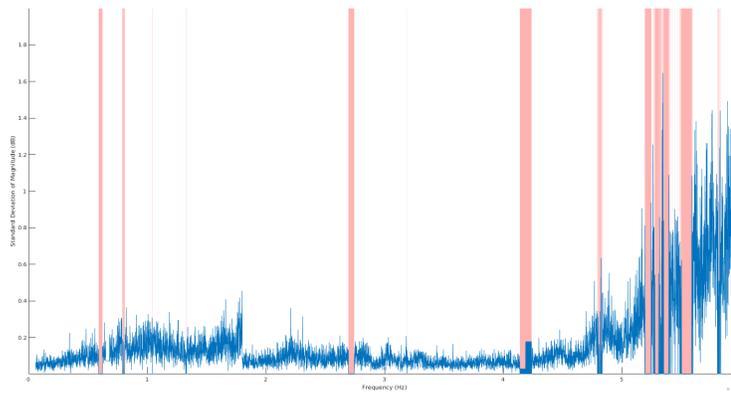


Figure 5.14: Standard deviation of magnitude for TwinRX measurements of a reflective load – y-scale is 0-2dB, x-scale is 0-6 GHz

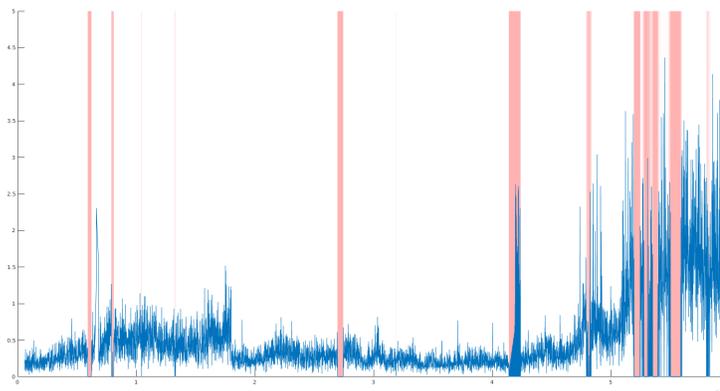


Figure 5.15: Standard deviation of phase for TwinRX measurements of reflective load – y-scale is 0-5 degrees, x-scale is 0-6 GHz

5.4 The UBX-160 radio, quirks, and power leveling

The measurements with the UBX-160 took place months later, after significant modifications to the software architecture and SDR-based software VNA to accommodate other instruments and new kinds of data sweeps. The initial data captured with the UBX-160 looked promising and did not show frequency dead zones or instability of the FPGA. Observe that in Figure 5.16, which shows the measured magnitude of a matched load prior to calibration, the magnitude of the matched load above 2500 Mhz was fairly high. This means that the system could barely distinguish between reflective and absorptive loads.

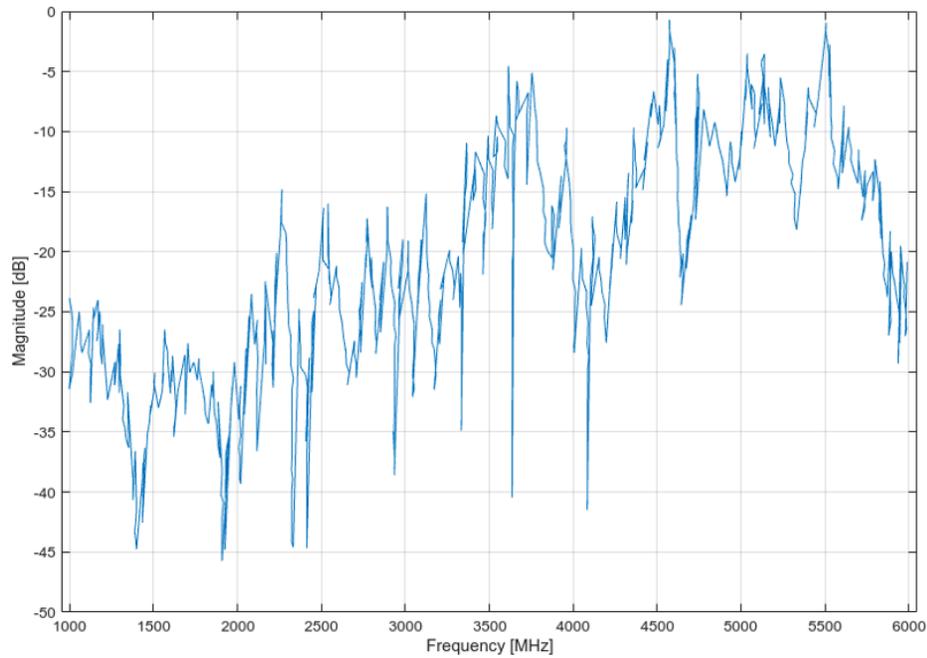


Figure 5.16: UBX-160 measurement of magnitude of matched load calibration standard, without calibration applied

The initial hypothesis for the poor magnitude response was that the measured b-wave amplitudes approached the measurement noise floor, which, after further investigation, was found to be true. This topic, including the characterization and compensation of transmitter power variability, is discussed in more detail later in the latter sections of Chapter 2. Observe the figure below, a MATLAB contour plot that visualizes the power magnitude as a function of frequency and tx_gain. A power meter sensor with an input range of -30 to +25 dBm, was used to measure the total output power at each frequency and tx_gain setting. At any given gain level, about 20 dB more power was available at low frequencies compared to high frequencies. There is a discontinuity at 500MHz where the UBX-160 switches component chains.

Compensation for the transmit power, in order to level the power across the frequency spectrum, greatly improved the magnitude response above 2 GHz. The power variation with respect to frequency was reduced to $\sim .05$ dB. In theory, power leveling should not be necessary because only ratios are measured. However, if the signal approaches the measurement noise floor, then the lack of power leveling will change the results in practice. 5.18 shows a comparison of the magnitude measured before and after compensating for transmission power levels. However, it did not compensate for receiver power differences.

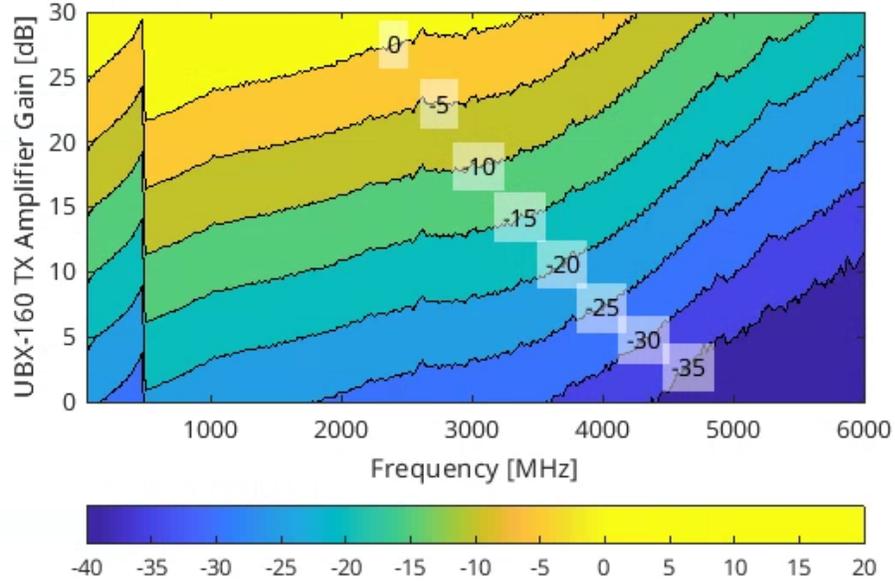


Figure 5.17: UBX-160 power output as a function of frequency and tx_gain

Observe that the frequency start point of this plot is 1 GHz instead of ~ 50 MHz. The UBX-160 radios are not capable of phase-synchronous operation above and below 1GHz at the same time; switching between the two requires a complete reinitialization of the radio.

The corrected magnitude of the UBX-160 radio, which lacked dead zones and instability, was indeed very promising. Once again, the phase coherence was checked by inspecting the phase difference between the open- and short-circuit standards. Observe that in Figure 5.19 below, the absolute value of the phase, this time in the low band (below 1 GHz), is typically flat at about 180° ; evidently, even flatter than was the case for the usable regions of the TwinRX. While a non-ideal phase difference between the open- and short-circuit standards is compensated by calibration, the closer the raw level is to being flat to begin with, less correction is needed. However, also observe that there were regions where the radio evidently failed to tune coherently. It can be seen in the figure that many iterations were plotted, with each representing a different run, and that the coherence issues only seemed to occur at specific frequencies.

Despite some apparent issues with phase coherence using the UBX-160 radio, the first two-port measurements were soon performed. Observe that in Figure 5.20, apart from being relatively noisy, the results look more or less as expected for a 10 dB attenuator. There is a relatively flat 10 dB

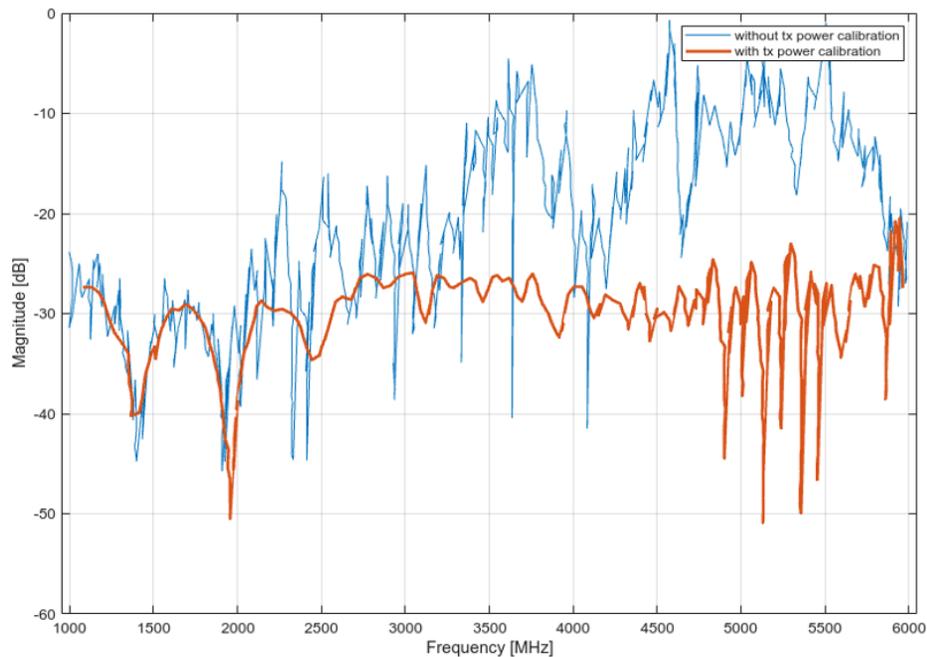


Figure 5.18: UBX-160 measurement of matched load calibration standard, without calibration applied, with (red) and without (blue) transmit power leveling applied.

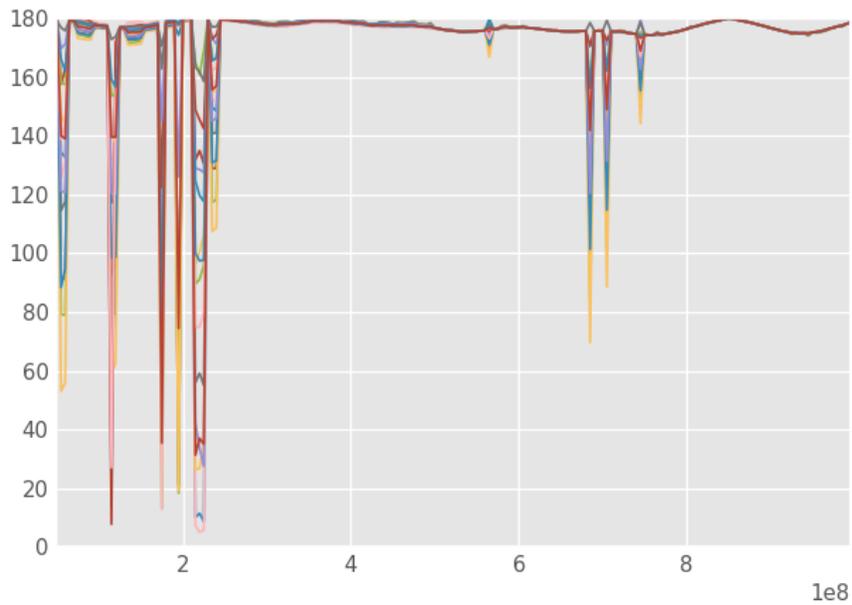


Figure 5.19: UBX-160 Radio – phase difference between open-circuit and short-circuit calibration standards, without calibration applied – Phase (degrees) plotted with respect to frequency in 100s of MHz

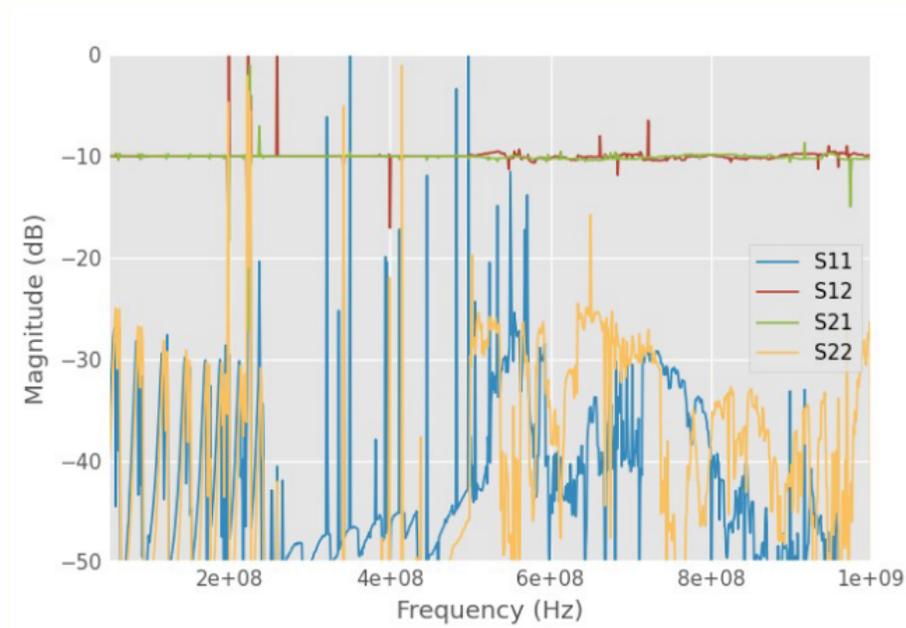


Figure 5.20: UBX-160 two-port measurement of 10dB attenuator

insertion loss (lines S_{12} and S_{21}), and the return loss at either port exceeds 25 dB (lines S_{11} and S_{22}).

Rather than investing more effort into another problematic radio, another radio was selected for future work, this time the SBX-120.

5.5 Later experiments with the SBX-120 radio

The SBX-120 radio remedied most of the serious issues that had been observed thus far. The SBX-120 radio was relatively free from quirks. It required sacrificing some frequency range, but its stability and ease of use more than compensated for it. It was free from the frequency dead zones and FPGA instabilities of the TwinRX radio and also the phase coherence issues of the UBX-160 radio. The first measurement taken with this radio was a repeat of the 10 dB attenuator measured by the UBX-160 radio depicted in Figure 5.20.

Observe that in Figure 5.21, most of the noise of the measurement observed with the UBX-160 Radio is gone and also that the return loss appears to be somewhat lower, which suggests a more

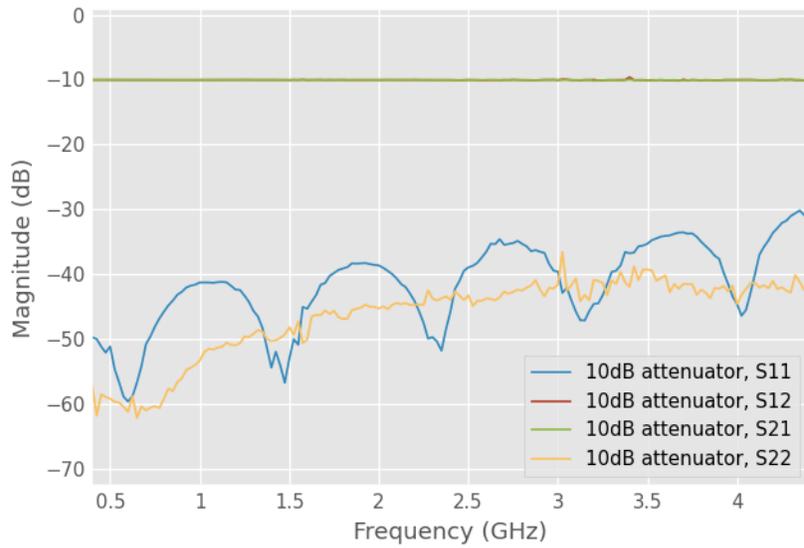


Figure 5.21: SBX-120 two-port measurement of 10 dB attenuator

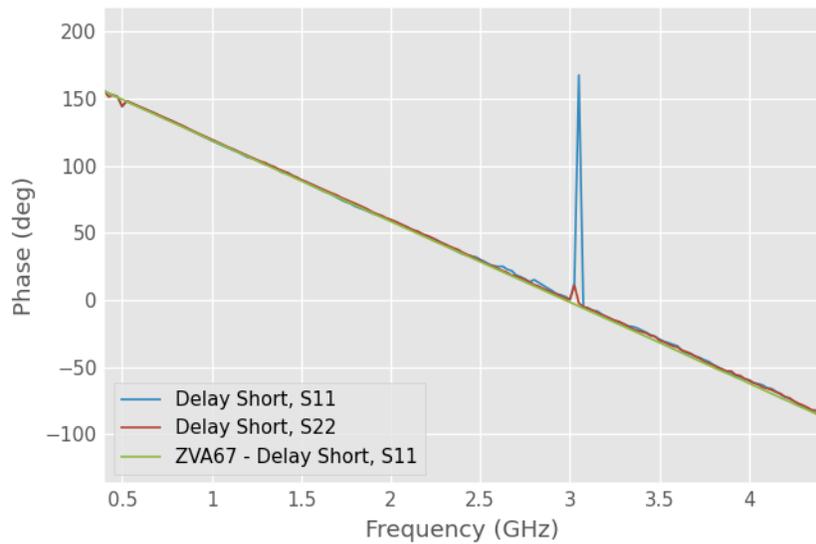


Figure 5.22: SBX-120 measurement of delayed short, compared with the same measurement by a commercial VNA

accurate calibration. The subsequent measurement taken with the SBX-120 was of a delayed short. Observe in Figure 5.22 below that, apart from a couple of outliers in the data, the data are similar to a measurement done with a commercial analyzer, the R&S ZVA67.

Outliers like these in the data can be remedied to some extent, and the best ways to do so will be considered in the following sections.

5.6 SDR Power, Sampling Rate, Measurement, and Sweep Approaches

Several considerations must be taken when creating the application(s) to interact with the SDR, sweep, measure, and process the data. This section distills the knowledge gained during the project. The approaches taken and numerous alternatives are discussed here.

The first consideration to be taken into account is the power received. An effort should be made to ensure that the same level of power is incident on the receiver's ADC at all frequencies. On the surface, the choice of whether to use a higher input power with a lower gain or vice versa looks unimportant; however, using a higher gain should be preferred because it allows the system to operate with a much lower noise figure (NF), perhaps 20 dB lower, which will greatly reduce the measurement noise. As such, the highest receiver `rx_gain` setting should be used to minimize NF. This necessitates the use of a lower excitation power.

The next consideration has to do with how much transmit power is needed. It is preferable to avoid saturating the analog-to-digital converter (ADC). Backing the power off from saturation by about 5-10 dB is preferable. Ensuring that the power level is optimized will not impact most measurements, but can be particularly beneficial when measuring isolation and switch terms. Other measurements are more likely to be limited by the leakage of the couplers.

The sampling rate and number of samples to collect also warrant careful consideration. This will also tie in with the measurement sweep. At least with the Ettus Research X310, the ADC is sampling at its' full rate, 200 MHz, at all times. Downsampling occurs in the digital domain, not by adjusting the actual clock. The full data rate is too high for practical processing, but that does not mean that it is not possible to take advantage of the full sampling rate when making measurements. VNA. The Ettus X310 allows for "digital tuning" within the analog bandwidth. The significance of this is that one can tune the analog center frequency to f_c and then digitally re-tune within IBW to various offsets, allowing one to capture many data points with a single analog tune. Instead of

re-tuning potentially hundreds of times per sweep, it should be possible to cover an entire sweep with the SBX-120 using only 40 analog re-tunes – 100 MHz steps, because the manufacturer advises against tuning all the way to the edge of the analog bandwidth because the IF anti-aliasing filter may somewhat attenuate it. Digital tuning is instantaneous, so using the IBW of the SDR will drastically speed up the sweep, though it is still not anywhere near the same speed as a commercial VNA. This technique can in principle be adapted to any radio, not just those that support this feature. However, the IBW would be limited to the maximum tolerable sampling rate and instead of digitally re-tuning the frequency shift would be implemented by multiplying the signal by a complex exponential in the software instead. Recall from Chapter 3 that for consistent results the signal of interest must be aligned with the center of a discrete Fourier transform (DFT) bin to prevent scalloping loss. A simple algorithm for tuning in a sweep of this manner follows:

```
void tune_rx(target_freq)
{
    static double rf_freq = 0;
    static double analog_ssbw =
    0.9 * RADIO_ANALOG_BANDWIDTH / 2;

    double dsp_freq =
    target_freq - current_rf_freq;

    if (
    dsp_freq < -1 * analog_ssbw ||
    dsp_freq > analog_ssbw )
    { // analog tuning required
        rf_freq =
        std::round((target_freq + analog_ssbw) / REFERENCE_FREQUENCY)
        * REFERENCE_FREQUENCY;

        if (rf_freq > RADIO_MAX_FREQUENCY)
        {
            rf_freq = RADIO_MAX_FREQUENCY;
            std::cerr <<
```

```

        "requested frequency was beyond the maximum frequency, tuning to " <<
        rf_freq << std::endl;
    } else if (rf_freq < RADIO_MIN_FREQUENCY)
    {
        rf_freq =
            std::round(( RADIO_MIN_FREQUENCY + analog_ssbw )
                / REFERENCE_FREQUENCY ) * REFERENCE_FREQUENCY;
        std::cerr <<
            "requested frequency was beneath the minimum frequency, tuning to " <<
            rf_freq << std::endl;
    }
} else
{ // can digitally re-tune
    dsp_freq = target_freq - rf_freq;
}

/*
 * API interaction for handling the actual tuning here
 */
}

```

It is also important to consider the other consequences of the sampling rate. At higher sampling rates, data processing is more intensive, so a balance must be achieved. It is possible to optimize the sampling rate and the number of samples to achieve a target measurement noise floor level and noise variance, as was discussed in the context of lock-in amplifiers (LIAs) in Chapter 3. In practice, the sampling rate used most frequently in this work was 6.25 MS/s with 131072 samples. This was probably overkill, but provided a good safety margin.

The last primary consideration is the means of getting the data and how to safely store it once it has been acquired. Receiving bursts of measurement data by issuing timed commands for tuning and bursting data could be a viable way to maximize the data rate, assuming that the FPGA's command queue can keep up. This project has not taken that route. First, this way would be pretty complex, and second, great effort has been made to keep this project's software hardware-agnostic.

The other way, and the one used by this project, is to continuously stream data from the radio, throwing away samples until they are needed. The reception should be in a dedicated thread. Once received, the data should be stored in an appropriate buffer that tracks the size of the data, and provides mutexes for safety.

5.7 Techniques to Compensate for Erroneous Data

There has always been some incidence of erroneous data received, sometimes in one measurement channel, sometimes the other, or sometimes both, with the three radios used in this work. Differentiating erroneous data from good data can be difficult but is not always necessary. The main ways that have been devised for coping with the erroneous data fall into two basic categories: threshold the data against a known good data range (especially with the reference channel) or correct the data statistically.

In the first approach, in which the data are thresholded, the a-wave value is compared to known good measurements of the a-wave under a variety of conditions (e.g., short, open, match, thru) and a threshold is applied. If the measurement deviates too far, then it is marked for remeasurement. The data in Figure 5.23 below show the concept and the results: the dashed lines are the thresholds, the dotted lines were the minima and maxima – observe that the samples occasionally fall outside the threshold range and would therefore have been marked for remeasurement. This approach only catches about half of all errors because errors seem to be equally likely to occur in either channel, and this method can only test the reference channel.

The first approach corrected only for reference data. It can be supplemented by statistics, but finesse is required in how the statistics are applied. Simple averaging can not cope well with extreme outliers, even with large averaging factors. The next thought may be to apply the median on a measurement-by-measurement basis; and this is what was tried first before moving towards more complicated methods. Unfortunately, when the median is applied on a measurement-by-measurement basis, even subtle errors in the measurement of the calibration standards could result in large outliers in the calibration. The next thought was to perform M calibrations, where M was the number of measurements for each calibration standard, perform M calibrations, and then apply the median to each calibration coefficient. This worked a bit better, but when applied to measurements it still

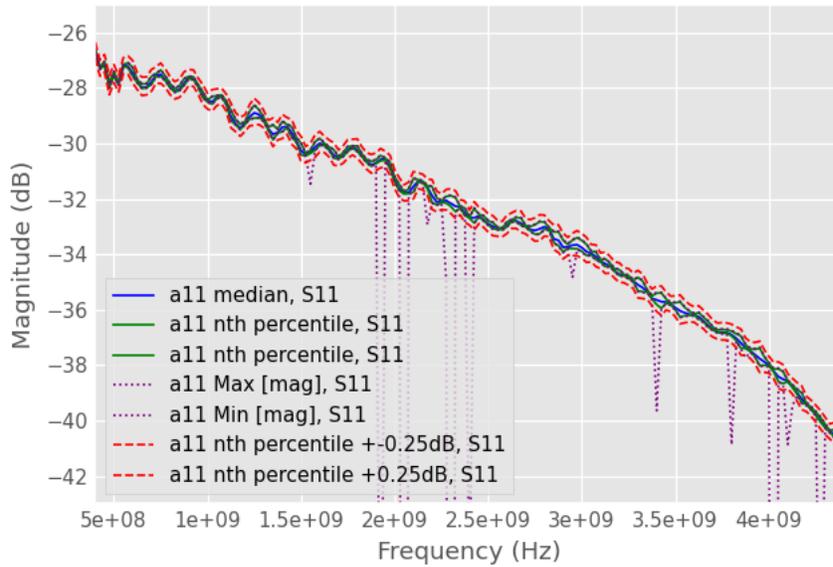


Figure 5.23: Example measurement utilising the "a-wave masking" approach described above

resulted in outliers in the data as often as not. The approach that ultimately corrected the problem most consistently is a little more involved. N subtle variations (e.g., differing pin depth, return loss, etc.), within manufacturer-specified tolerances, and then N calibrations are performed and the N calibrations are all written to disk. Later, when the calibrations are applied, the N calibrations are applied to the M measurements, again selecting each measurement at random. Then the software takes the median of the ensemble average. This value has proven to be more stable, though sometimes there are still residual errors in the data. This method is also quite slow, especially for dense sweeps or large values of N .

5.8 VNA Residual Error Analysis

VNA figure of merits (FOMs) were briefly discussed in Chapter 4. The most important ones are, arguably, those related to effective calibration accuracy (residual errors after calibration): effective directivity, effective source matching, effective reflection tracking, effective load match, and effective transmission tracking. Accuracy at different levels of return loss is also important. VNA datasheets may also specify linearity, speed, and other metrics. This characterization of SDR-based VNA is primarily focused on the residual errors, but also touches on the measurement error with respect to return loss.

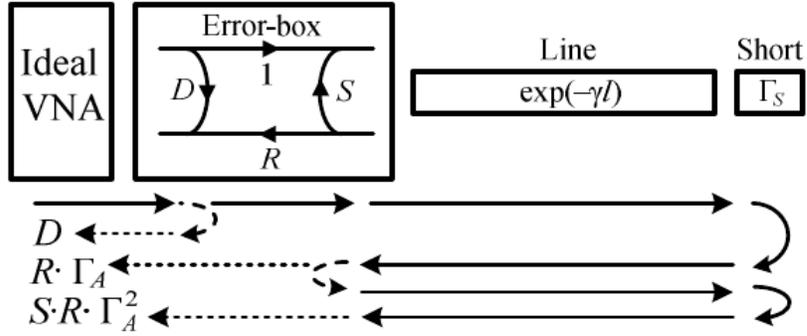


Figure 5.24: Shorted airline block diagram. Figure from [34]

The ripple method is a traditional amplitude-only technique to verify the accuracy of the calibration of VNAs [32] [33]. This technique makes use of precision coaxial airlines – standards that exhibit minimal loss, have precise lengths, and excellent return loss. Ideally, the calibrated measurement of these devices resembles that of any other through-device, but inaccuracies in the calibration result in a standing wave that remains observable even after calibration. The airline length should be selected so that multiple cycles of the standing wave can be observed within the measurement bandwidth. The period of the round-trip standing wave can be calculated as $f = c_0 / (2l\sqrt{\epsilon_R})$ where c_0 is the speed of light in vacuum, l is the length of the airline, and ϵ_R is the relative permittivity – approximately 1.0006 for air. The airline used in this work was 14.99 cm long, so the period of the standing wave was $f = 299792458 / (0.30\sqrt{1.0006}) = 999.68$ MHz, a round trip delay of about one ns. The main benefit of airlines is their exceptional return loss, typically ~35-50 dB, which minimizes the amount of unwanted reflections during verification. The following paragraphs and figures document the residual error analysis of one VNA, the R&S ZVA40, using its internal frontend. Applying the ripple technique to extract all residuals constitutes residual error analysis. The interaction between the source, airline, and a termination is depicted in Figure 5.24. The equations shown in the following paragraphs were derived by summing incident waves, where higher order waves are found by simply cascading transmission (i.e. of the airline) and reflection (i.e. of the source and load ports) terms.

The first step in this technique is to measure the airline terminated by a precision-matched load. The airline's exceptional return loss helps ensure that there are no undesirable reflections, and the precision-matched load should ideally absorb any incident energy. In this case, any incident energy on the b-wave port can be assumed to be due to residual directivity error. Mathematically, this looks like $\Gamma_{ac} \cong \delta + \Gamma_l e^{-2\gamma l}$ where Γ_{ac} is the measured ("actual") reflection coefficient, δ is the residual directivity vector, Γ_l is the reflection from the load, γ is the complex propagation constant, and l is

the length of the airline. For a matched load, $\Gamma_l \approx 0$, so $\Gamma_{ac} \cong \delta$. The residual directivity vector can be approximated as the radius of the circle being traced out in the complex plane, or equivalently as the difference between the minima and maxima of the ripple on a Cartesian plot, $|\delta| \cong \frac{s_\delta}{2}$ where s_δ is the magnitude difference of the minima and maxima. See Figure 5.25 below; observe that the difference between the minima and maxima is about .007, so the effective directivity can be calculated as $20 \log_{10} (.006694/2) = 49.5$ dB.

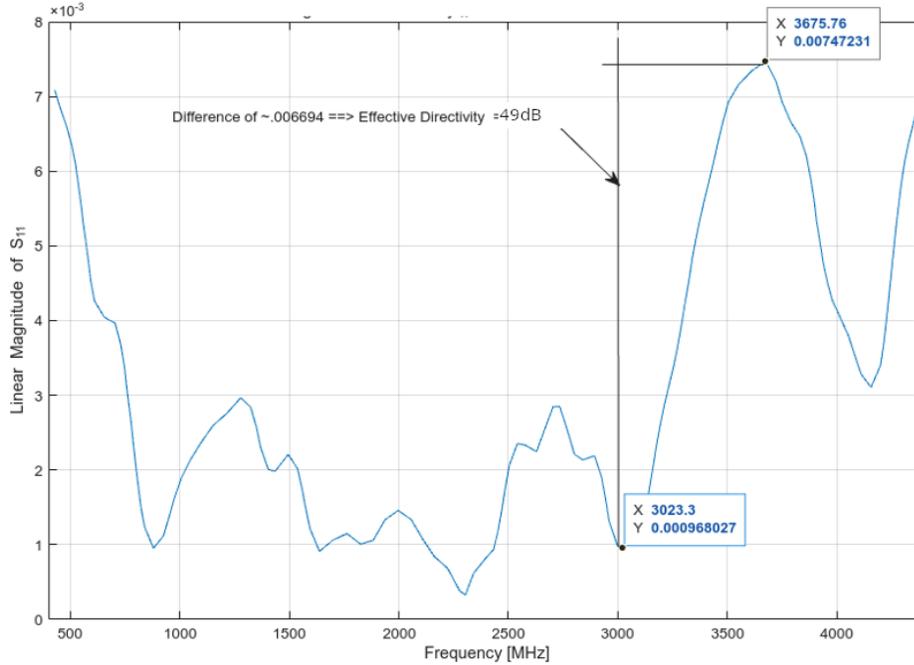


Figure 5.25: Measurement of a precision airline terminated with a matched load calibration standard. This is used to calculate the effective directivity FOM

The next step is to measure the airline terminated with a short. Assuming that the effective directivity is appreciably better than the effective source match (which is typical), the residual directivity error should have minimal effect on the source match, and what effect it has can theoretically be subtracted out, leaving the airline itself. Then, the airline attenuation and phase can be de-embedded, leaving as pure a version of the effective source match as possible. Mathematically, this looks like $\Gamma_{ac} = \delta - e^{(-2\gamma l)} + \mu e^{(-4\gamma l)}$, where $|\mu| \approx \frac{s_\mu}{2} \cdot \frac{1}{|\Gamma_{ac}|^2}$ is the residual reflection coefficient at the source port and s_μ is the difference of the ripple maxima and minima in the shorted airline measurement. The first term is again the residual directivity, the second term represents a single round trip, and the third term represents a second round trip after reflecting off the source. Ideally, these could

be separated in the time domain by gating. Unfortunately, this does not work well without substantial measurement bandwidth – The 4 GHz bandwidth used in these measurements limits the time-domain resolution to about 250 ps. Another way to isolate the desired term is to first de-embed the airline model and subtract the residual directivity (as effectively measured by the airline match). After de-embedded the airline including its attenuation, $|\Gamma_{ac}|^2 \approx 1$, which after simplifying leaves $\Gamma_{ac} = \mu e^{(-4\gamma l)}$, so $|\Gamma_{ac}| = |\mu|e^{(-4\gamma l)}$. The ripple in the shorted measurement then gives the effective source match. See Figure 5.26 below, which also includes a low-pass filtered version of the signal, which is useful for making it easier to pin down the value of the source match. Note that the low-pass filtered signal diverges near the right edge of the plot, which is unfortunate but unavoidable. In this case, the in-band ripple was about .0229, so the effective source match is $20 \log_{10} (.0229/2) = 38.8$ dB

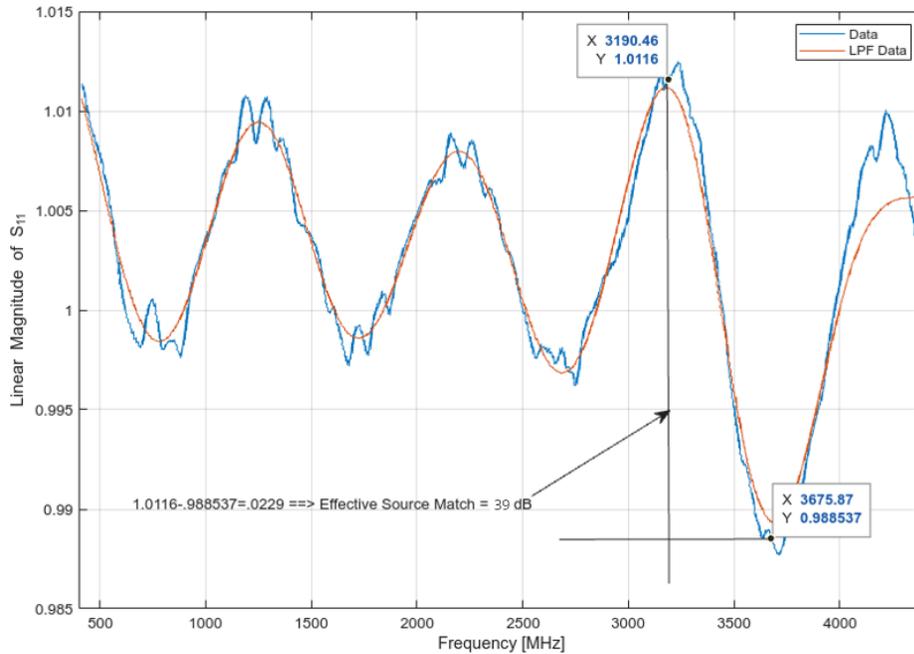


Figure 5.26: Measurement of a precision airline terminated with a short-circuit calibration standard, with the attenuation and phase of the airline de-embedded and the residual directivity subtracted. This is used to calculate the effective source match FOM

The remaining measurement error after removing the effects of residual directivity and source match is the effective reflection tracking error. The effect can be removed by subtracting the effective source match contribution from the shorted airline measurement. The source match contribution can be

removed by applying a high-pass filter to the data, which removes the airline’s very long-standing wave. The effective tracking error is again half the diameter of the ripple. As shown in Figure 5.27, this is $0.0039/2$, so $20 * \log_{10} (1 - \frac{.0039}{2}) = 0.017dB$, which seems a little low.

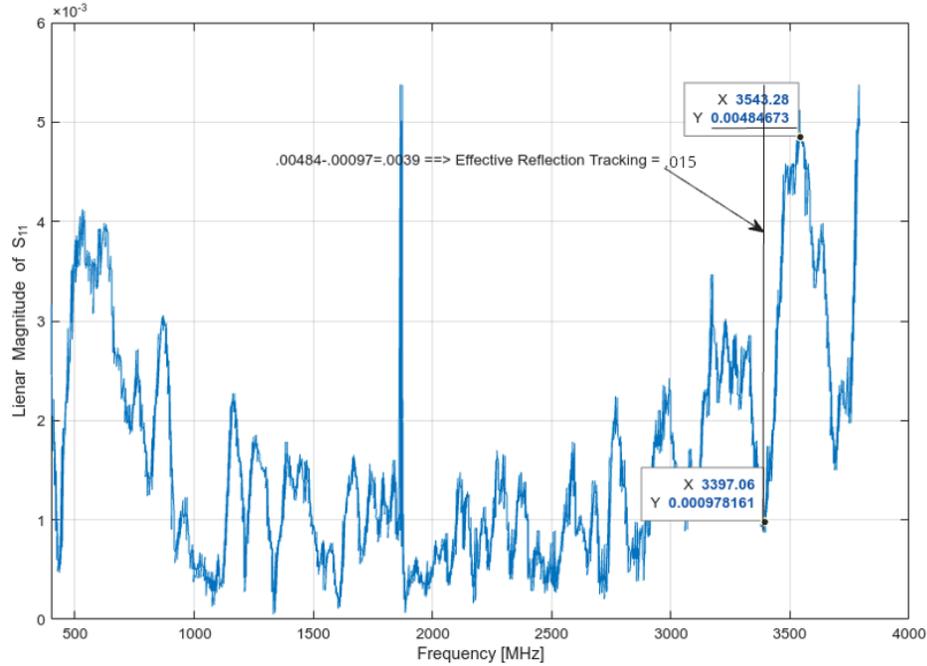


Figure 5.27: Measurement of a precision airline terminated with a short-circuit calibration standard, after de-embedding the airline, subtracting the residual directivity, and also subtracting the residual source match. This is used to determine the effective reflection tracking FOM

Next, the effective load match is determined by observing the return loss of a 2-port airline through measurement. The effective load match characterization is analogous to the effective source match characterization, except that instead of the source reflecting off a short at the end of the airline, it reflects off the load port. Mathematically, we have $\Gamma_{ac} = \delta - \rho e^{(-2\gamma l)} + \rho \mu e^{(-4\gamma l)}$ where ρ is the residual reflection coefficient at the load port due to load match error. After subtracting the directivity error and de-embedding the airline model, this simplifies to $\rho = \frac{s_\rho}{2(\mu-1)}$. In this case, as seen in Figure 5.28, $s_\rho = 0.01188$, so the effective load match is $20 \log_{10} (|\frac{.01188}{2(.0229-1)}|) = 44.4dB$

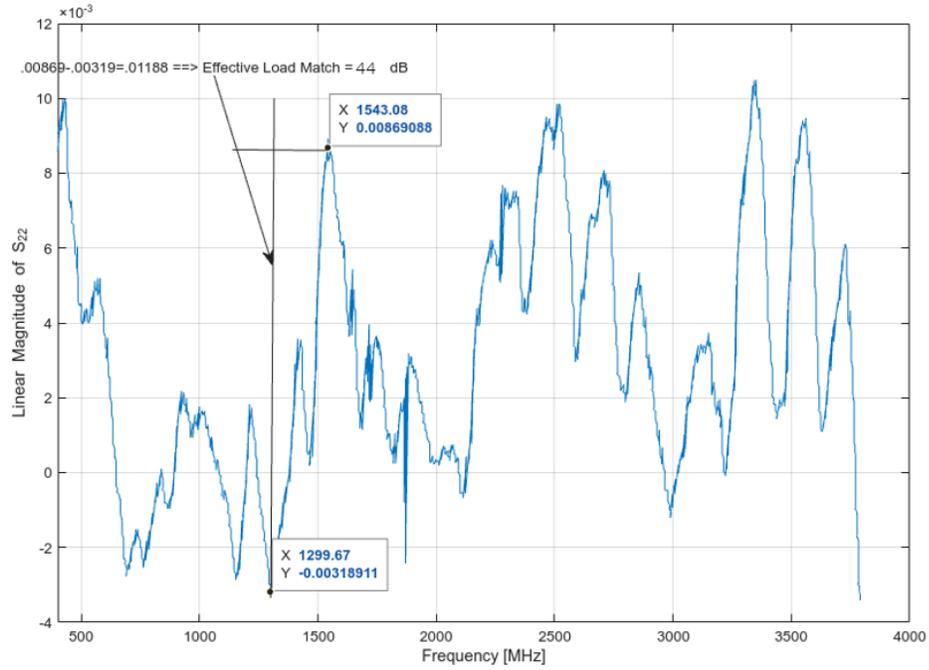


Figure 5.28: Measurement of a precision airline connected between the two measurement ports, with this plot displaying the return loss looking into the device from port 1, after de-embedding the airline model, subtracting the residual directivity and residual source match. This figure is used to determine the effective load match FOM

Determining the effective transmission tracking term is the final step in this residual error analysis and concerns the residual error in S_{21} after correcting for the load match error. The load match error can be removed by applying a high-pass filter to remove the long-period frequency ripple caused by the airline after subtracting the effective directivity and de-embedding the airline model. (or equivalently, subtracting the ripple identified by low-pass filtering during the load match analysis). The observed ripple in Figure 5.29 below is 0.04536, so the calculated ripple is $20 \log_{10} (.04536/2) = 0.2$ dB.

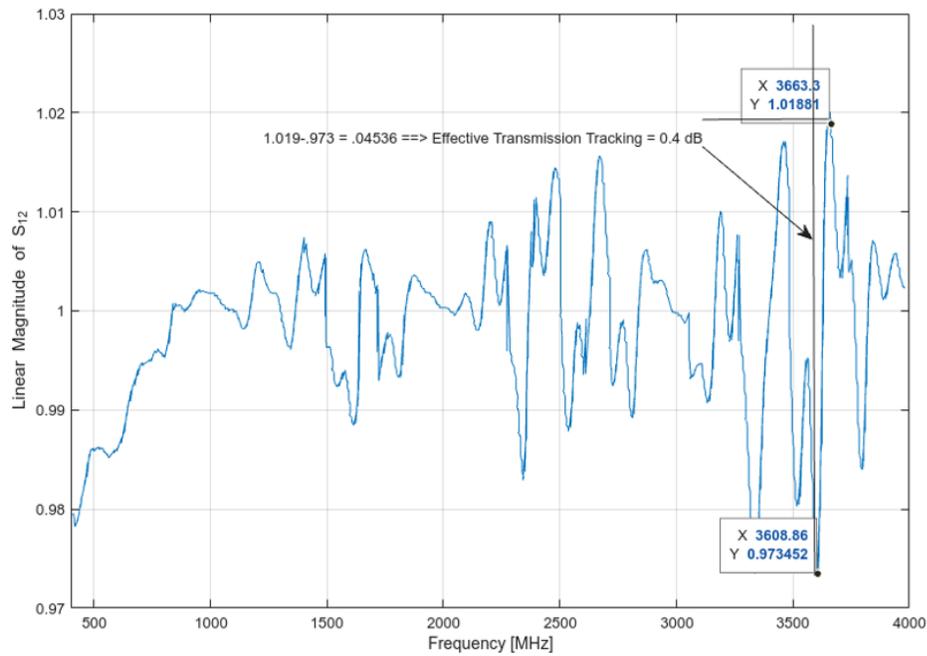


Figure 5.29: Measurement of a precision airline connected between the two measurement ports, with this plot displaying the insertion loss between port 1 and port 2, after de-embedding the airline model, subtracting the residual directivity and residual source match and load match errors. This figure is used to determine the effective transmission tracking FOM

5.9 Accuracy with respect to Return Loss

Another main figure quoted in VNA data sheets is the accuracy with respect to return loss. The raw, uncorrected accuracy is hugely dependent on the quality of the couplers used in the frontend. As such, the results with this frontend compared to the one in a commercial analyzer should be expected to be inferior. However, this is remedied to an extent by the VNA calibration – both in terms of accuracy and repeatability. Here, the SDR as a backend is expected to fair much worse, doing perhaps not much better than the uncorrected values, because of the poor phase repeatability, as was discussed at the end of Chapter 2.

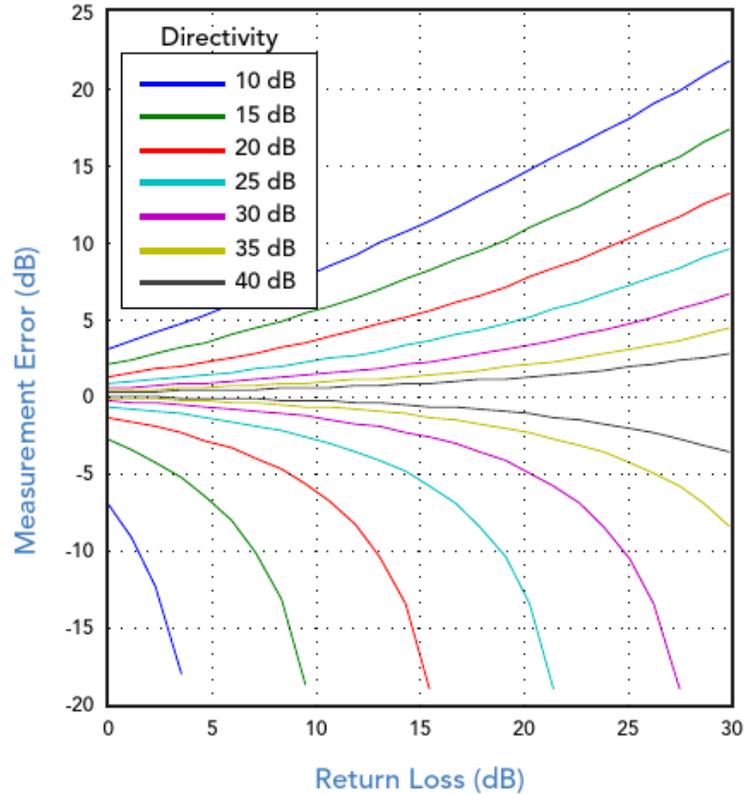


Figure 5.30: Return Loss measurement error as a function of directivity and return loss.

Recall that the raw return loss of a reflectometer is a function of the directivity and insertion loss of the pair of couplers. Figure 5.30 [26] shows the expected return loss measurement error as a function of the actual return loss and the coupler directivity, assuming a low insertion loss of 1 dB. In order to accurately measure return loss, it is evident that the directivity of the couplers should be about 10 dB greater than the return loss being measured, which keeps the error within about ± 5 dB. The expected return loss measurement error for the COTS frontend used with the SDR-based VNA of this project is depicted in Figure 5.31 below, which were determined graphically using the equations used to derive the figure, but using the insertion loss and directivity values of this frontend as a function of frequency. The values shown in the plot suggest that measurements of return loss beyond 15-25 dB may be difficult to achieve with this frontend.

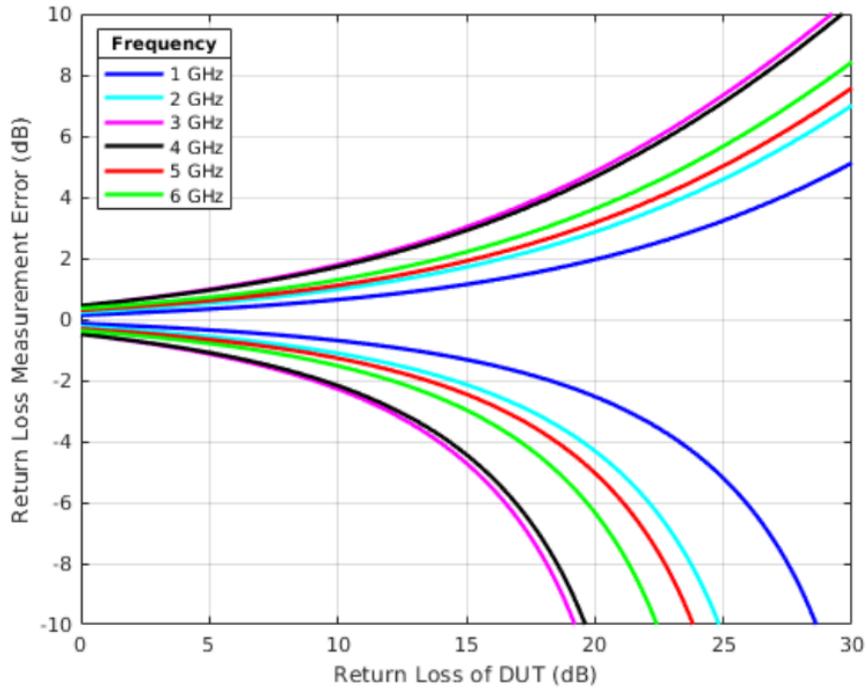


Figure 5.31: Projected return loss measurement error of the SDR-based VNA COTS frontend as a function of return loss at several frequencies

The limit at which a return loss can be measured and with what error across frequency for the SDR-based VNA COTS frontend before taking calibration into account is depicted in Figure 5.32 below. It can be observed that the limit with minimal error is about 5-10 dB return loss, with acceptable measurement error is about 20 dB, and that much beyond 25 dB error is likely to dominate. To an extent, this error can be corrected; however, it requires highly repeatable magnitude and phase measurements, which the SDR increasingly struggles with as the frequency is increased, so it can be predicted that the calibration using the SDR might not be able to adequately compensate for the directivity error, but the same measurement with a more repeatable receiver, like a commercial VNA, might not display the same problems.

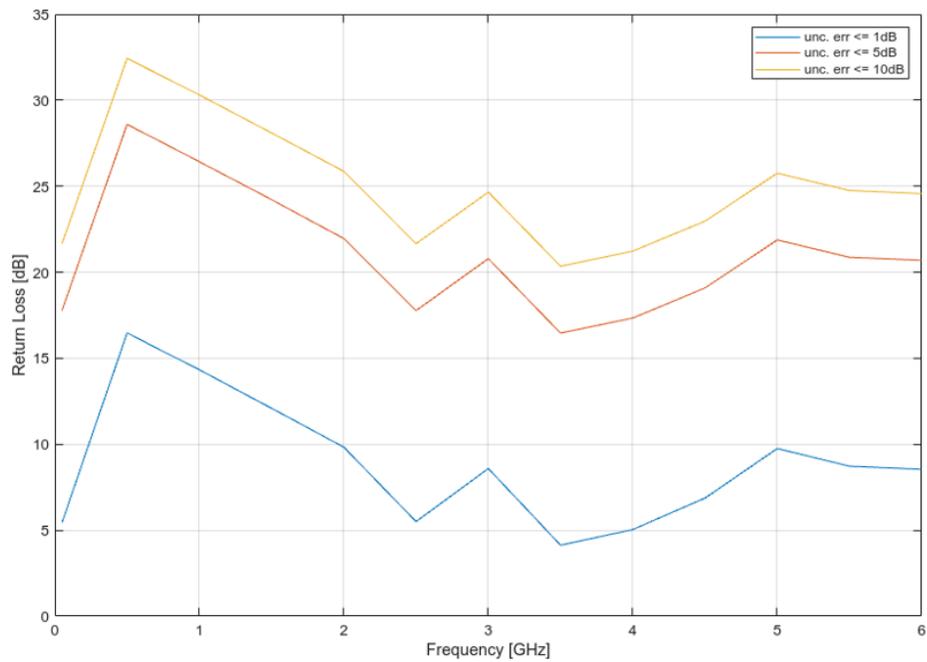


Figure 5.32: Predicted return loss limit as a function of frequency for different values of return loss measurement error

The plots below show measured return loss of a shorted attenuator, such that a 5 dB attenuator would present a 10 dB round trip return loss. The calibration in these plots exhibited problems above 2 GHz, but below that frequency some observations can still be accurately accounted for. For the 5 dB attenuator (10 dB return loss) in Figure 5.33 below, observe that the magnitude error is within about 0.5 dB.

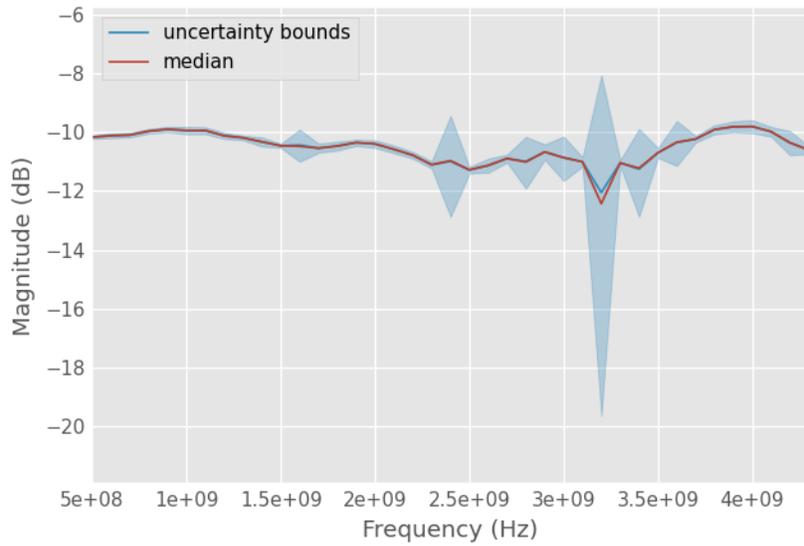


Figure 5.33: One-port measurement by the SDR-based VNA of a 5 dB attenuator terminated with a short-circuit.

In the case of a measured 10 dB shorted attenuator (20 dB return loss), shown in Figure 5.34 below, the error slipped to a little over 1 dB.

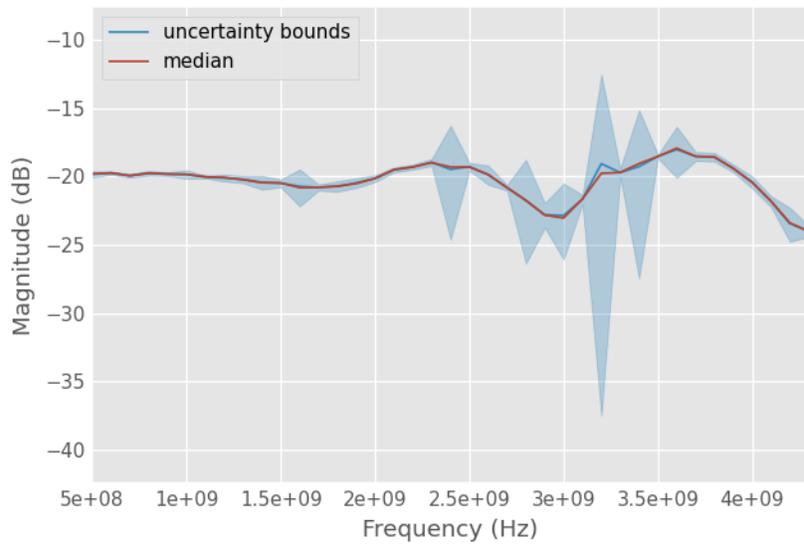


Figure 5.34: One-port measurement by the SDR-based VNA of a 10 dB attenuator terminated with a short-circuit.

The increase to a 15 dB attenuator (30 dB return loss), which is depicted in Figure 5.35, increased this error to about 2 dB.

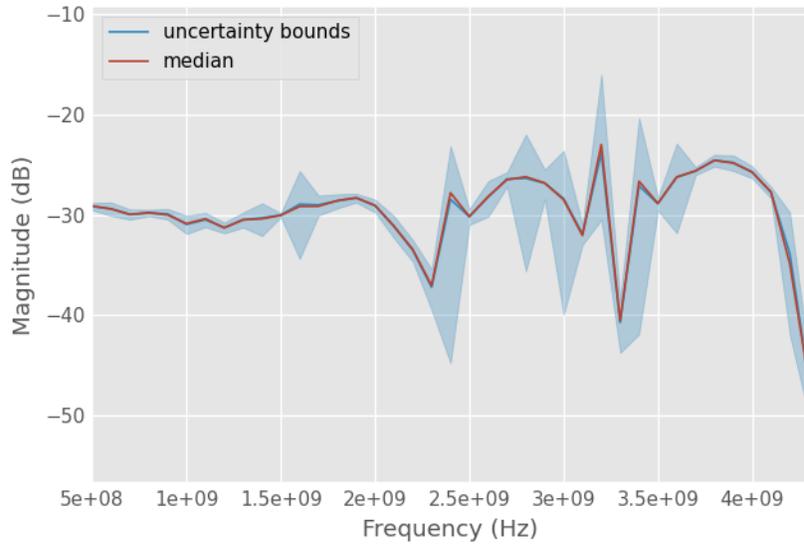


Figure 5.35: One-port measurement by the SDR-based VNA of a 15 dB attenuator terminated with a short-circuit.

Increasing the attenuator to 20 dB (40 dB return loss) – shown in Figure 5.36 below – resulted in a much higher error of nearly 10 dB.

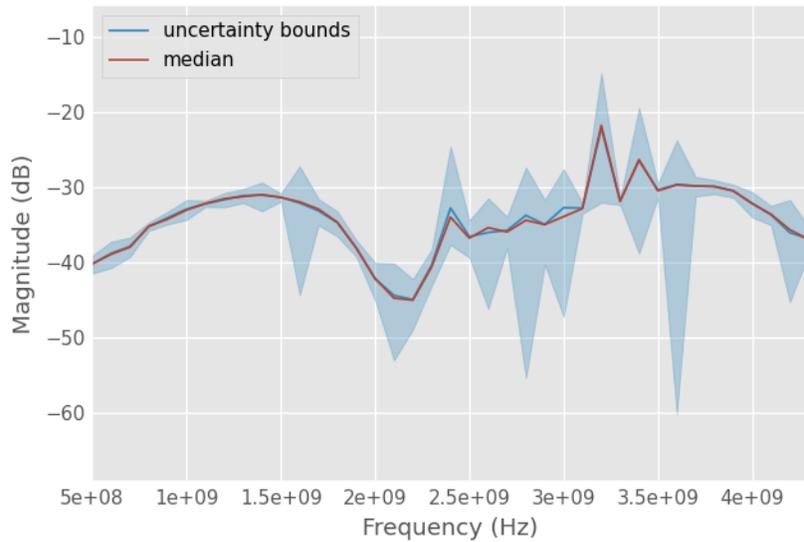


Figure 5.36: One-port measurement by the SDR-based VNA of a 20 dB attenuator terminated with a short-circuit.

The results depicted in the attenuator measurements above are indicative of about a 10-15 dB improvement in measurable return loss with a typical short-open-load calibration with the SDR-based VNA. While an improvement, this compares poorly with commercial instruments like the R&S ZVA40, which can claim accuracy to within 1dB down to 80dB return loss.

5.10 VNA Residual Error Comparison

Table 5.2: Residual error analysis results and comparisons

FOM (all units dB)	ZVA Spec.	ZVA × ZVA	ZVA × COTS	SDR × COTS
Eff. Directivity	36-42 typ.	49	48	46
Eff. Source Match	36-42 typ.	39	33	28
Eff. Reflection Tracking	0.05-0.1 typ.	0.015	0.030	0.150
Eff. Load Match	36-42 typ.	44	35	35
Eff. Transmission Tracking	0.05-0.01 typ.	0.4	3.2	0.45

In Table 5.2, compares the residual calibration error of a commercial VNA, the R&S ZVA40, with the SDR backend. It was also possible to measure using the COTS frontend with the ZVA40, which helps isolate which is suggestive of which problems are related to the frontend COTS hardware.

Unfortunately, the attempt to do the reverse and measure using the SDR and the ZVA frontend was unsuccessful. It might be possible, but not simply.

The parameters measured with the ZVA, compared to datasheet specifications, mainly seemed reasonable. The effective directivity seemed somewhat high, but this may be because the measurement was performed primarily at relatively low frequency (0.4 - 4.4 GHz). Likewise, the effective reflection tracking also beat specification. The effective transmission tracking, which has the most room for error (especially considering cascading error when doing ripple analysis), was slightly out of specification. R&S's laboratory probably uses time-domain gating, or more advanced techniques [32] [33] and 100x averaging and NIST traceable standards to characterize their equipment with exacting accuracy. The simpler approach taken in this work may not achieve the same level of accuracy and precision.

The one-port parameters of the ZVA with its own frontend compared to the ZVA measured with the COTS frontend compared favorably, exhibiting minimal difference except a slightly raised reflection tracking term; however the two-port parameters suffered. There was a large error in the transmission tracking term, possibly due to a problem in the calibration, else maybe due to the long cables used to connect it to the frontend, and the load match term was also worse by about 10 dB.

The residual directivity of the SDR with the COTS frontend fared almost as well as the ZVA, however the source match and the reflection tracking error somewhat worse. The effective load match with the SDR matched that of the ZVA using the COTS frontend, and the tracking error was comparable to the ZVA using its own frontend. This comparison required neglecting significant outliers in the residual data, but is suggestive that the SDR-based VNA could be a strong performer if the measurements were more repeatable.

It should be possible to effectively isolate the couplers from the transmitter and receivers, which is very helpful if there are long cables leading into the couplers or if the ports are poorly matched, by placing precision attenuators at the source input, a-wave, and b-wave ports on the directional couplers. The slight loss of power can easily be remedied by adjusting the transmit power. Precision attenuators can have nominal return loss of 30 dB or better, which should be sufficient to prevent reflections from the attenuator itself. The attenuator value should optimally be chosen such that $\Gamma_{eff} = S_{11}^{att} = \Gamma + 2S_{21}^{att}$. This can effectively improve the raw port matching to 30 dB or better.

5.11 Chapter Conclusion

This chapter presented the design, implementation, and evaluation of an SDR-based VNA prototype using COTS hardware. Several hardware and software iterations were explored, with each stage addressing challenges related to phase coherence, measurement repeatability, and calibration accuracy.

Initial testing with the TwinRX daughterboard revealed significant limitations, including frequency dead zones and FPGA instability during tuning. These led to the selection of the UBX-160, which offered improved stability but still exhibited inconsistent phase coherence at certain frequencies and could not operate across its full bandwidth without reconfiguration. Ultimately, the SBX-120 was chosen for its superior stability and repeatability, at the cost of reduced frequency range.

To mitigate repeatability limitations, techniques such as a-wave masking and ensemble-based calibration averaging were developed. SDR parameters were tuned for consistent performance across frequency without sacrificing precision or efficiency.

Residual error analysis confirmed that the SDR-based VNA, while not competitive with commercial instruments in absolute terms, performed well for a low-cost platform. The dominant limitation was shown to be phase repeatability, restricting the effectiveness of complex calibration techniques. With architectural improvements, such as introducing a shared PLL across measurement channels, the system could achieve substantially better performance. Further gains in precision, range, and measurement speed are possible by upgrading the directional couplers and switching elements.

Chapter 6

Frequency Extension

6.1 Overview

Since their inception in the mid-1990s [35] [36], software-defined radios (SDRs) have become ubiquitous tools for accessing the radio spectrum. Their flexibility, reconfigurability, and relatively modest cost have made them attractive for a wide range of applications – including spectrum sensing, implementation of signal processing techniques, investigation of radio frequency (RF) system performance [37], vector network analyses (see Chapter 1) and imaging. SDRs entered the mainstream with the emergence of highly integrated transceiver integrated circuits (ICs) that combined analog-to-digital converters (ADCs), digital-to-analog converters (DACs), and reconfigurable signal-processing hardware in a single chip [38]. SDRs are sold as ready-to-use packages with the relevant hardware, firmware, and software, enabling agile development and iteration of a digitally controllable radio frontend. Affordable SDRs with frequency coverage up to about 6 GHz have been commercially available for years. State-of-the-art units can reach up to 18 GHz or so; however, they are quite expensive. This stagnation in maximum frequency is not surprising. SDRs have historically been driven by communications applications, most of which occur below 6 GHz – with exceptions such as satellite communications and newer 5G UWB bands, neither of which are the primary design targets of SDR platforms. Nevertheless, research has demonstrated that SDRs operate at millimeter wave (mmWave) frequencies through the use of solid-state RF frontends and frequency conversion chains

[39]. The potential for SDRs to operate at mmWave and submillimeter wave (sub-mmWave) wavelengths could potentially open the door to a broader set of applications, including different RADAR applications, spectroscopy, imaging, medical diagnostics, and other applications. [40]. However, doing so typically introduces a host of technical challenges, particularly with regard to frequency generation, signal power, and signal integrity.

The conventional method for generating sub-mmWave signals is to use frequency multiplier-based upconverters or mixers. This process requires the use of nonlinear components that, unfortunately, also introduce compression, harmonic distortion, and intermodulation effects. Another consideration is that frequency multipliers multiply the phase, noise, and operating bandwidth of the input signal in addition to the frequency. Moreover, frequency multipliers scale not only frequency, but also the phase, bandwidth, and noise. As such, the bandwidth of the upconverted signal may need to be constrained so that it does not exceed the receiver's integration bandwidth (IBW). An alternative is to use narrowband direct-generation sources such as gun diodes. This option is somewhat narrower in bandwidth and may be challenging to control sufficiently via software. It is also possible to generate sub-mmWave frequencies using laser heterodyning techniques, although generating frequencies this way may be a project in its own right.

Higher-end laboratory RF instruments also share some of the limitations associated with SDR frequency range and performance. Achieving multiple decades of bandwidth often requires either highly specialized components or the ability to switch between bands using different hardware paths. SDRs have a more general purpose design which aims to support wide IBW in their transmit and receive chains. Large IBW is useful, especially for communications-centric work, but can actually be a hindrance in some respects. Instruments like LIAs and VNAs, by contrast, will typically have relatively narrow IBW and can afford to follow their mixing stages with tight intermediate frequency (IF) filters, which reduces the frontends effective noise bandwidth. vector network analyzers (VNAs) are known for supporting a very broad frequency range – often limited by the connector and interface technologies used – but even these instruments rely on external frequency extension modules when operating deep into the mmWave and sub-mmWave regimes. These experiments aimed to investigate whether SDRs, when combined with such extension modules, can be similarly extended into these higher frequency domains and to understand the associated tradeoffs.

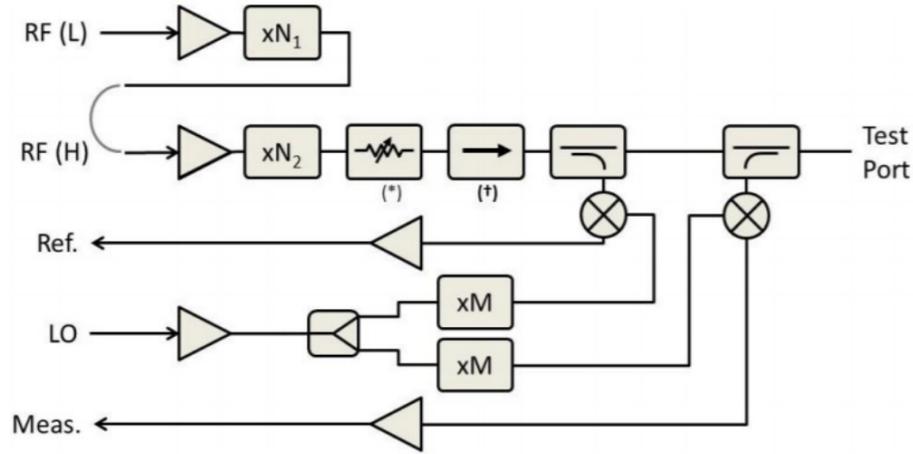


Figure 6.1: VNA frequency extender block diagram. Figure from [41]

6.2 Interfacing with VNA Frequency Extension Modules

Significant strides have been made in interfacing SDRs with VNA frequency extension modules, but the process is non-trivial. These extenders were designed to interface with VNAs and, as such, typically require input RF and local oscillator (LO) input signals of 9-18 GHz with a nominal input power of +0-10 dBm. These frequency and power values are well within the reach of most VNAs but beyond the reach of most SDRs. The two IF output signals – reference or a-wave (ref) and measured or b-wave (meas) – are within the range of 100-1000 MHz.

A representative block diagram of a VNA frequency extender is shown in Figure 6.1 below. Observe that the RF and LO are inputs to multiplication chains. The RF path arrives at the test port via back-to-back directional couplers while the LO drives the mixers connected to the coupled outputs of the directional couplers, producing the REF and MEAS outputs at the IF. The relationship between the RF, LO, and IF are described by the equation $f_{RF} \cdot N - f_{LO} \cdot M = f_{IF}$. In this equation, N and M are the multiplication factors of the RF and LO multiplier chains, respectively. The IF is typically fixed – often 279 MHz – due to poor gain flatness. $N \cdot f_{RF}$ is the desired test frequency, and f_{LO} is computed accordingly to satisfy this equation.

Because most SDRs cannot directly generate signals at 9-18 GHz with sufficient output power, two practical strategies exist for using them with VNA frequency extension modules. First, when the desired waveform is simple (e.g., a pure tone or low bandwidth simple modulation), one can bypass the SDRs transmit chain entirely and use an RF signal generator as the source instead. In

this arrangement, the SDR can still be used for signal reception, effectively operating as a high-performance commercial off-the-shelf (COTS) ADC. Second, one can upconvert the SDR output to the necessary drive frequency and power by using a mixer or multiplier upconversion chain. Both approaches were used at different stages of this work, depending on experimental constraints and goals.

6.3 Upconverter Communications Experiment

A custom upconverter was assembled using COTS components to meet the frequency and power requirements of the VNA frequency extension modules. The goal was to translate the SDR's output signal – centered initially at 450 MHz – to a frequency within the extenders' input range (in this case, 12.15 GHz) while maintaining sufficient power and spectral purity. This setup is illustrated in Figure 6.3 below.

A 450 MHz tone was transmitted using a USRP X310 with an SBX-120 daughterboard [42] [14]. Because SDRs rely on DACs for waveform synthesis and typically operate into compression when near the upper threshold of their output power, significant harmonic content is often present. A Mini-Circuits ZXLF-K551 low-pass filter (LPF) was used to suppress the harmonic content, effectively attenuating the 2nd, 3rd, and 4th harmonics by 5.5 dB, 22 dB, and 15 dB, respectively. The filtered fundamental tone power was measured at 16.3 dBm.

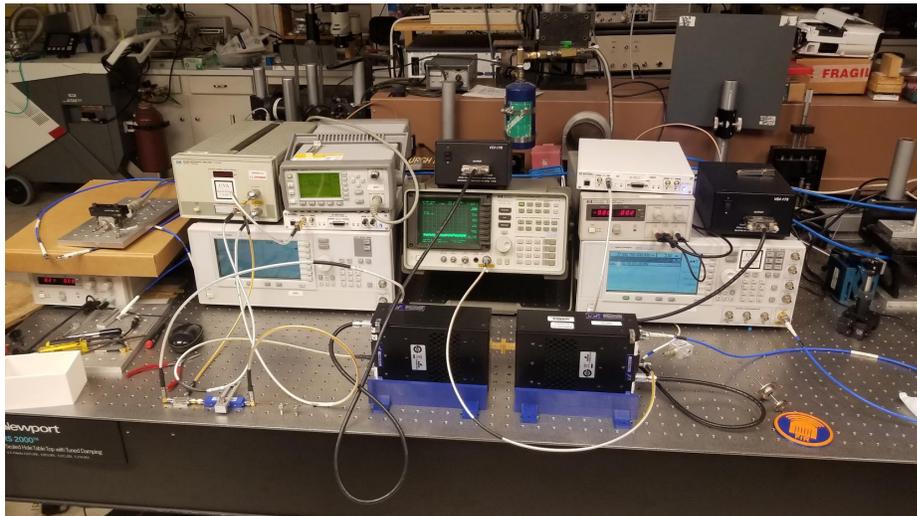


Figure 6.2: Equipment setup for SDR to VNAX upconverter

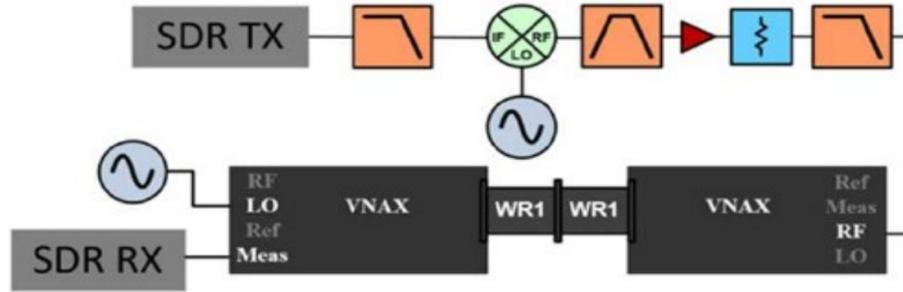


Figure 6.3: SDR with VNA extender experimental setup block diagram

The filtered signal was then mixed to 12.15 GHz using an Avantek DBX-186 mixer, with an Agilent E8247C signal generator providing the LO. The unwanted image signal was suppressed using a cavity bandpass filter (Mini-Circuits ZVBP-13R1G-S+). The upconverted signal was then amplified using an HP 8349B power amplifier, attenuated to control drive level, and filtered again (Mini-Circuits ZXLF-K133+) before being delivered to a VDI WR-1.0 VNA frequency extension module at 2 dBm input power. The extender’s multiplication factor of 81 moved the signal center frequency to 984 GHz. A summary of the power measurements at each stage is provided in Table 6.1.

Table 6.1: Power levels at various points in the transmit chain

Position	Tone Power	Total Power
At SDR TX	17.17 dBm @ 450 MHz	15.9 dBm
After LPF	16.3 dBm @ 450 MHz	17.3 dBm
After Mixer	-4 dBm @ 12.15 GHz	-0.93 dBm
After BPF	-4.83 dBm @ 12.15 GHz	-4.1 dBm
After Amplifier	11.5 dBm @ 12.15 GHz	14.84 dBm
After Attenuator	1.5 dBm @ 12.15 GHz	4.84 dBm
After LPF	-1 dBm @ 12.15 GHz	3.17 dBm

The complete set of test equipment used in this experiment is summarized in Table 6.2.

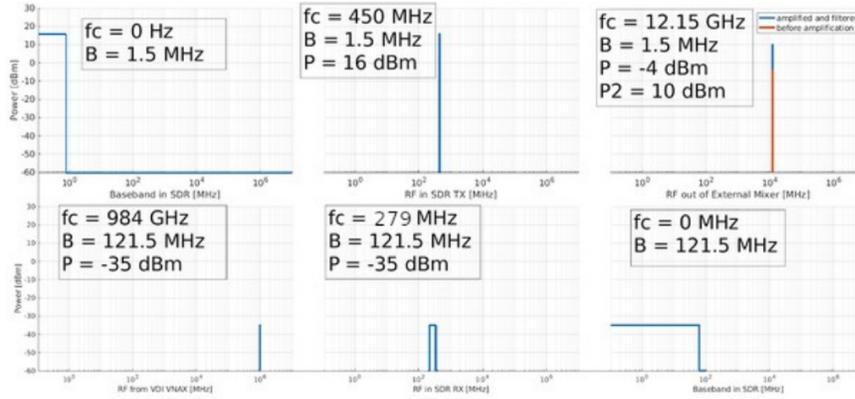


Figure 6.4: Signal power and bandwidth at each stage

Table 6.2: Equipment used in the upconversion and VNA extension chain

Device	Description
NI/Ettus X310 w/ SBX-120	TX up to 20 dBm max, 400 MHz to 4.4 GHz
Mini-Circuits ZXLF-K551 LPF	Passes 0-550 MHz, stopband to 35 GHz, 50 dB rejection
Agilent 8247C Signal Generator	TX to 20 GHz, used for upconversion mixer LO
Avantek DBX-186 Mixer	IF DC-7 GHz, RF/LO 6-18 GHz, 6 dBm P1dB, 11 dB conversion loss
Mini-Circuits ZVBP-13R1G-S+ BPF	Passes 11.7-14.5 GHz, stopband to 22 GHz
HP 8349B Power Amplifier	>10 dB small signal gain, 22 dBm P1dB
Mini-Circuits ZXLF-K133+ LPF	Passes DC-13 GHz, stopband to 40 GHz
VDI WR-1.0 VNA extender	2 dBm RF/LO input, ~ -30 dBm IF output, upconverts to 750-1100 GHz
Agilent 8257D Signal Generator	TX to 50 GHz, used for VNA extender downconversion LO

A companion VDI WR-1.0 frequency extension module provided the downconversion LO signal, set to achieve a 279 MHz IF. At the time, our research group did not yet know the extenders' IF range, so 279 MHz was chosen as a common best practice. Otherwise, a higher frequency (perhaps ~ 500 MHz) may have been used so that the second SBX-120 in the first X310 could have been used instead of a second X310.

Figure 6.4 above illustrates the signal’s spectral content at each stage: baseband (top-left), post-SDR upconversion to 450 MHz (top-middle), post-upconversion to 12.15 GHz (top-right), post-frequency extension to 984 GHz (bottom-left), downconverted to IF (bottom-middle), and finally back to baseband (bottom-right). Notably, the signal bandwidth after frequency extension was substantially wider – a natural consequence of the frequency multiplication, which scales the signal bandwidth in addition to the frequency, in this case, by a factor of 81.

6.4 SINAD-Based Characterization of Performance

A signal-to-noise-and-distortion ratio (SINAD) measurement was performed to verify system functionality and assess performance at each stage of the frequency upconversion chain [43]. Frequency modulation (FM) was selected as a modulation scheme for this experiment because it is relatively immune to compression and phase distortion – both of which are present in abundance due to the saturated amplifiers, mixers, and multipliers used in the extenders. FM also offers a well-established test methodology for SINAD measurement.

The measurement flowgraph, shown in Figure 6.5, was implemented in GNU Radio [44]. A 1 kHz sinusoidal tone was used as a modulating signal, driving a narrowband frequency modulation (NBFM) modulator block. The signal was then transmitted through the upconverter chain, passed through the VDI WR-1.0 VNA frequency extension modules, and finally received by the SDR backend.

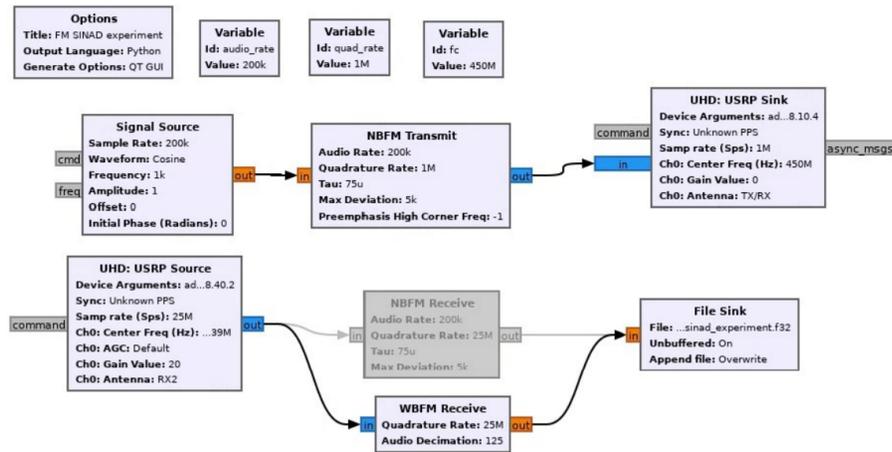


Figure 6.5: GNU Radio program for measuring SINAD

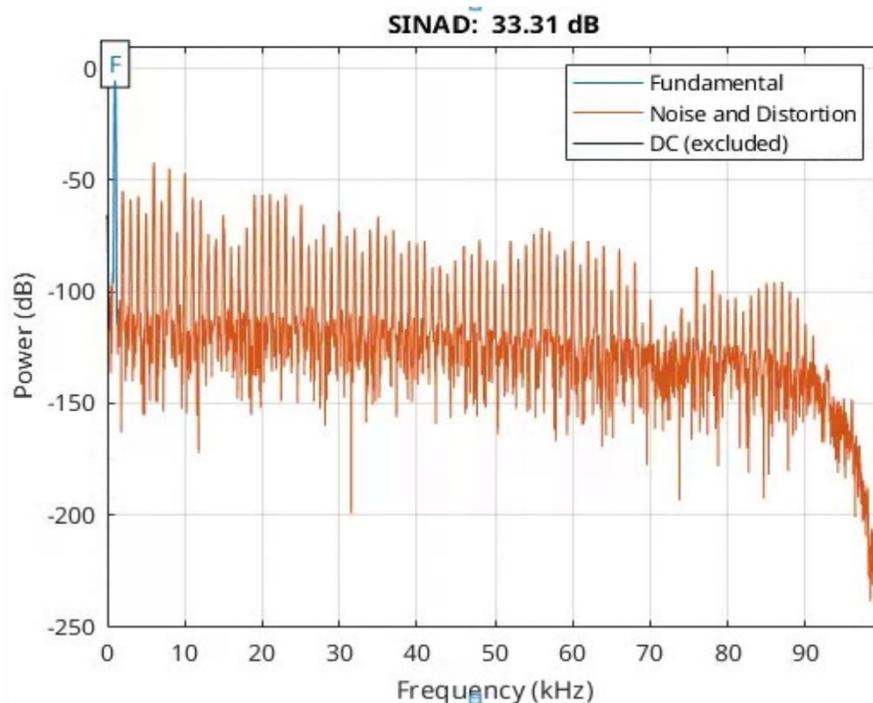


Figure 6.6: Example SINAD plot

The received signal was demodulated and analyzed using MATLAB's built-in `sinad()` function [45] to quantify the SINAD. This function estimates SINAD by identifying the strongest non-direct current (DC) spectral component as the fundamental and excluding it and its harmonics from the noise floor calculation. A modified periodogram was used to display the spectral content, as shown in Figure 6.6. The fundamental is highlighted in blue and is visible on the left-hand side of the spectrum.

A series of measurements were taken to evaluate how each additional element of the frequency chain – upconversion, frequency extension, and downconversion – affected the system's performance. These experiments included:

1. A synthetic baseband loopback test without hardware
2. A standard SDR loopback test at 450 MHz.
3. A test where the signal was first upconverted to 12.15 GHz, then immediately downconverted.
4. A full path test using WR-5.1 extenders (~145.8 GHz).
5. A full path test using WR-1.0 extenders (~984 GHz).

In the full extension tests, the transmitted modulation remained NBFM; however, the wideband frequency modulation (WBFM) receiver block had to be used for demodulation to compensate for the frequency extender’s bandwidth multiplication.

The quantitative results are shown in Table 6.3. As expected, SINAD degrades progressively as more nonlinear components are added due to worsening distortion. Both even and odd signal harmonics were visible in all cases. The observed anomalies in standard deviation are likely a consequence of the bandwidth-broadening effects introduced by frequency multiplication or perhaps the usage of the WBFM receiver block instead of the NBFM receiver block.

Table 6.3: SINAD measurements at different stages of the measurement chain

Configuration	Mean SINAD [dB]	Standard Deviation
Loopback (450 MHz)	39.13	4.44
No Extenders (12.5 GHz)	33.11	5.30
WR-5.1 Extenders (145.8 GHz)	27.87	0.128
WR-1.0 Extenders (984 GHz)	17.90	0.62

These results confirm that although distortion increases with each non-linear stage, signal integrity is sufficient even when converting to sub-mmWave frequencies. However, it also revealed that a relatively complicated experiment setup is required to ensure the correct frequency range, sufficient output power, and reasonable spectral purity. With narrowband experiments like this one it is not a problem, but achieving broad bandwidth would prove more difficult.

Other experiments that were performed in conjunction with this used digital modulation techniques such as quadrature amplitude modulation (QAM). The initial experiments were confounded and failed. The primary reason for this failure was that the multipliers also multiply the phase of the signal. By characterizing the nonlinear trend of the amplitude and phase multiplication before transmission, it is possible to apply pre-distortion to the signal to obtain a signal that can be correctly demodulated. This has been tried before, in analog, by VDI themselves [46]. The analog circuitry should be replaceable by an SDR with digital corrections instead of analog phase shifting.

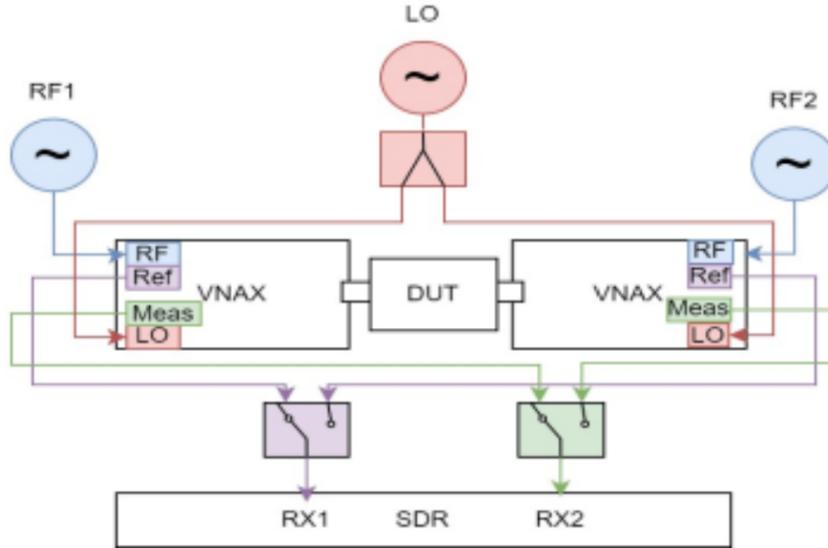


Figure 6.7: Block diagram for SDR-backend VNA at sub-mmW frequencies

6.5 SDR-based VNA at Sub-millimeter Wave Frequencies

The second major experiment performed in conjunction with the VNA frequency extension modules were calibrated VNA measurements using the SDR as a backend. In this architecture, signal generators were used as transmitters for simplicity. However, in principle, they are not strictly required, provided that the SDR can supply the appropriate frequency and power to drive the extenders directly or if a broadband upconversion chain, not dissimilar from the narrowband one described in the previous section, is employed.

Each VNA frequency extension module requires RF and LO inputs and produces two IF outputs, REF and MEAS, as described in previous sections. In this setup, the LO signal was shared with the two extenders via a resistive power splitter, and a separate signal generator independently generated each RF. The REF and MEAS IF outputs were multiplexed to the SDR using RF switches (Mini-Circuits MSP2TA-12-18+). An Ettus X310 equipped with two SBX-120 daughterboards was used for reception, with an IF of 500 MHz.

The measurement flow using the SDR with the frequency extenders mirrored the approach taken with the low-frequency prototype discussed in Chapter 5. The transmitter abstraction was configured to point to Signal Generator objects, and logic was introduced to handle the frequency conversion logic

transparently. Although the signal generators were tuned at each frequency step, the SDR remained fixed at the chosen IF. The digital signal processing (DSP) followed the same steps as described in Chapters 3 and 5.

6.6 VNA Measurements

The measurements described here were performed first using a Rohde & Schwarz ZVA67 VNA and then repeated using the SDR-based system. Calibration followed standard procedures for waveguide-based measurements and included the following waveguide standards: short, quarter-wave delayed short (QDS), matched load, and flush through. Three components were characterized: a waveguide attenuator, a bandpass filter, and a 1-inch through section. Each component was measured multiple times with both backends, with disconnects and reconnects between measurements to evaluate repeatability. Care was taken to ensure minimal cable movement – though some movement is inevitable – and the waveguide screws were properly torqued to reduce phase error.

The calibrated S-parameter measurements are shown below, with red traces corresponding to the Rohde & Schwarz ZVA67 and blue traces to the SDR-based system.

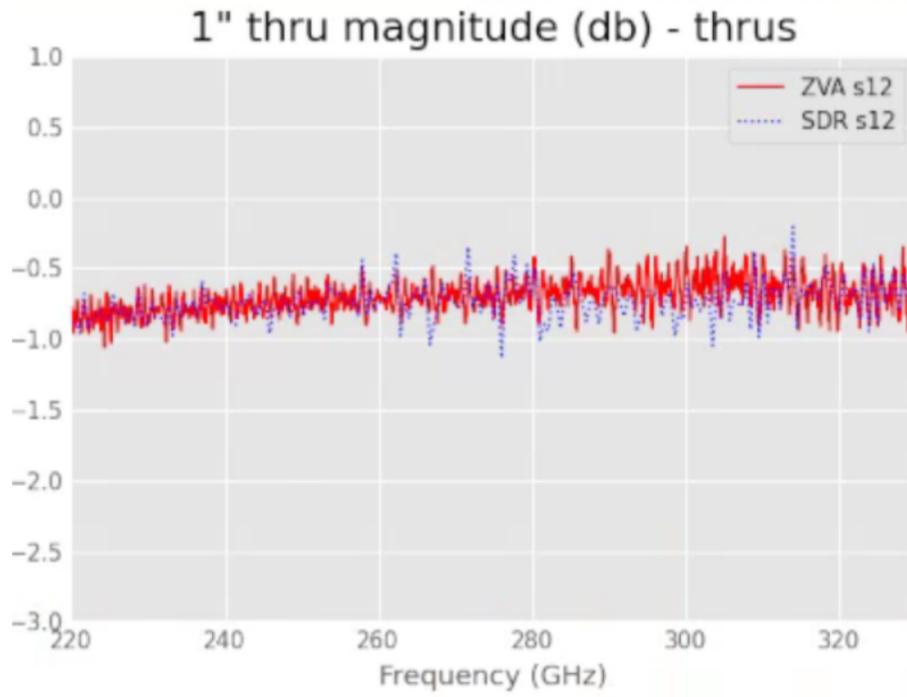


Figure 6.8: 1-inch thru magnitude

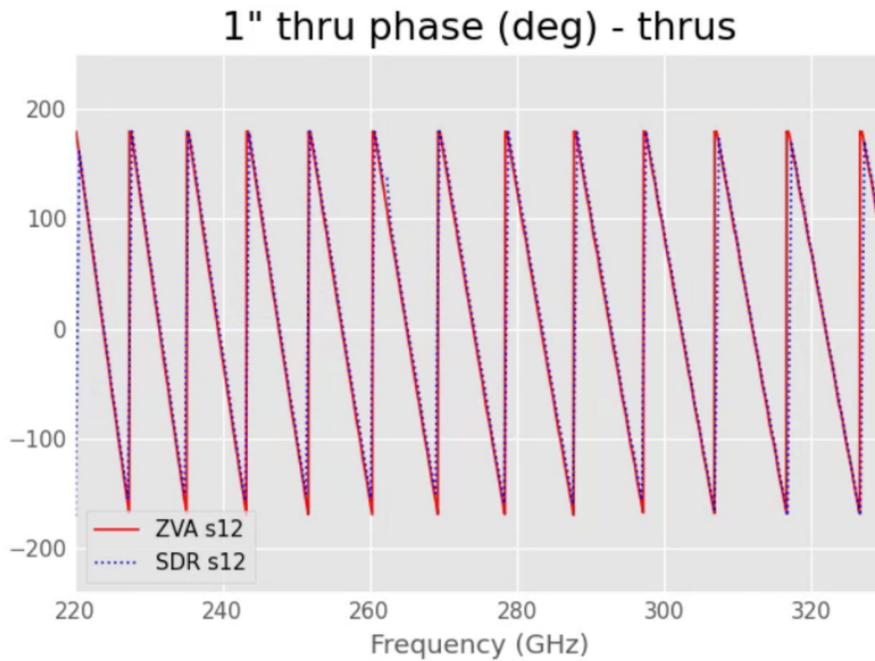


Figure 6.9: 1-inch thru phase in degrees

Figure 6.8 and Figure 6.9 show the S_{12} magnitude and phase response of the 1-inch rectangular waveguide thru section, respectively. Strong agreement is evident between the ZVA and SDR measurements in both magnitude and phase.

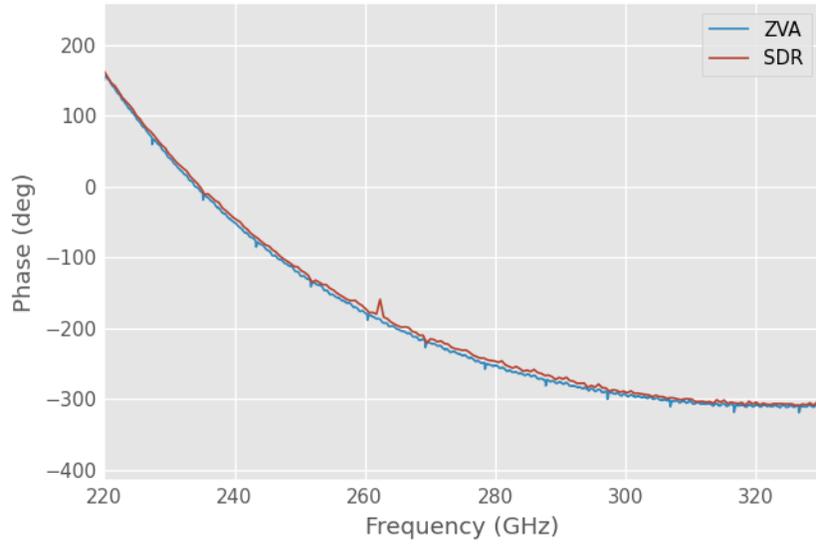


Figure 6.10: 1-inch thru phase in degrees, after de-embedding 1.18-inches of phase

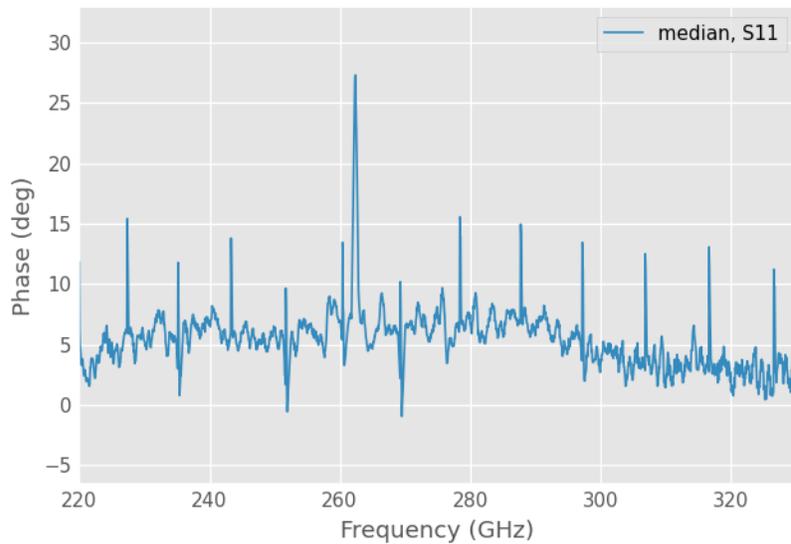


Figure 6.11: 1-inch thru phase difference between ZVA and SDR

Observe that in Figure 6.10 after embedding most of the phase of the through, there is some evidence of uncorrected dispersion in the 1-inch thru measurement. Figure 6.11 shows the direct phase difference between the ZVA and SDR measurements. The average phase difference was fairly flat at about 5-degrees, but the difference is quite noisy, and there are also some possibly periodic outliers in the data. The necessity for de-embedding more than 1-inch of waveguide might be due to extra length from the flange. The 5-degree difference could be due to a slight offset or misalignment during measurement or calibration (around $10\mu\text{m}$ probably).

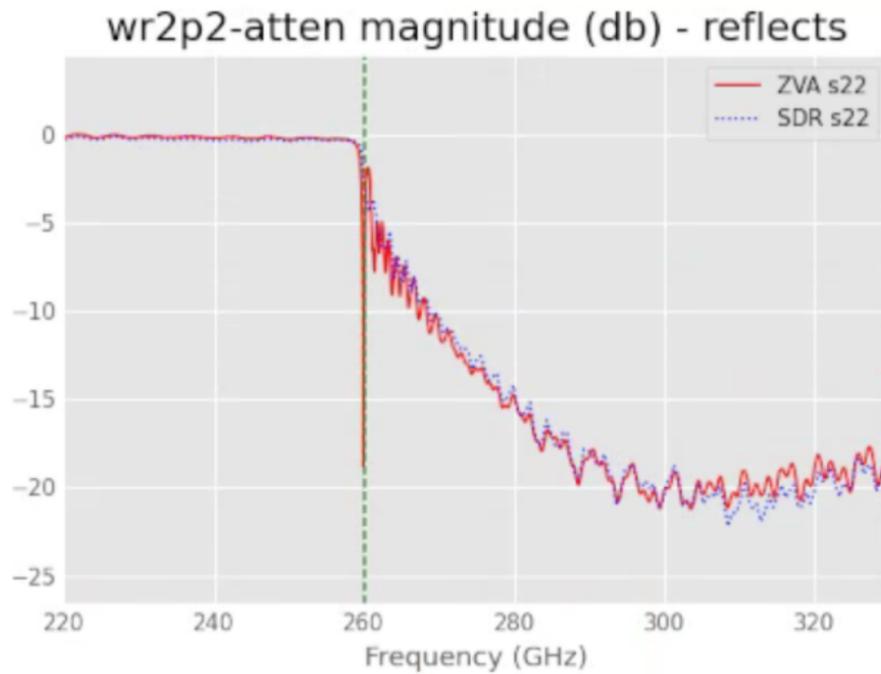


Figure 6.12: Attenuator Return Loss Magnitude dB

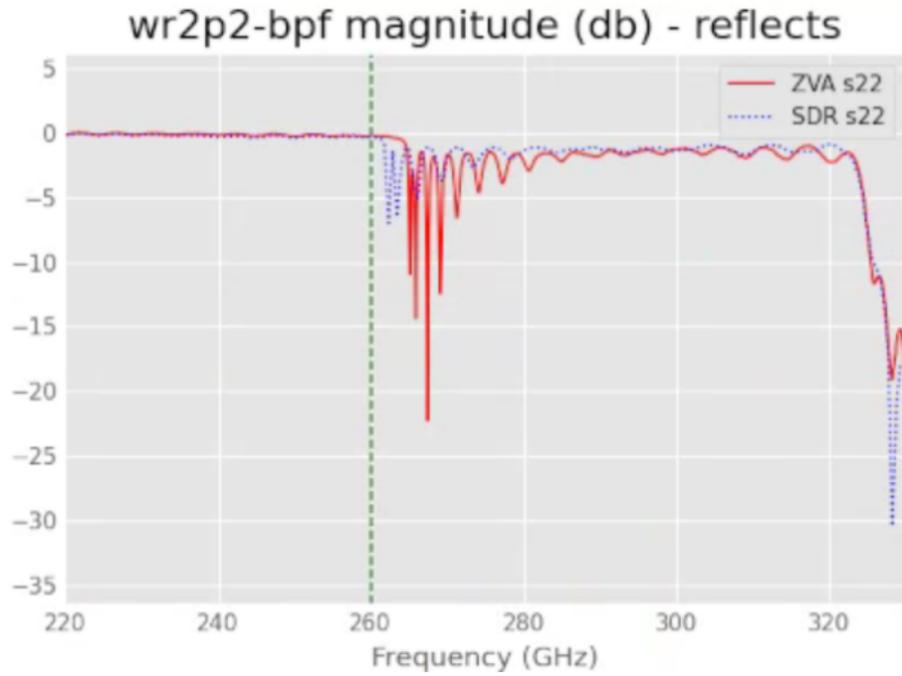


Figure 6.13: Filter Return Loss, Magnitude dB

Next, Figure 6.12 and Figure 6.13 show the return loss measurements for the waveguide attenuator and bandpass filter, respectively. Both instruments capture the overall behavior, with some ripple near the filter’s cutoff frequency appearing more prominent in the ZVA results, likely because of the use of a denser frequency sweep.

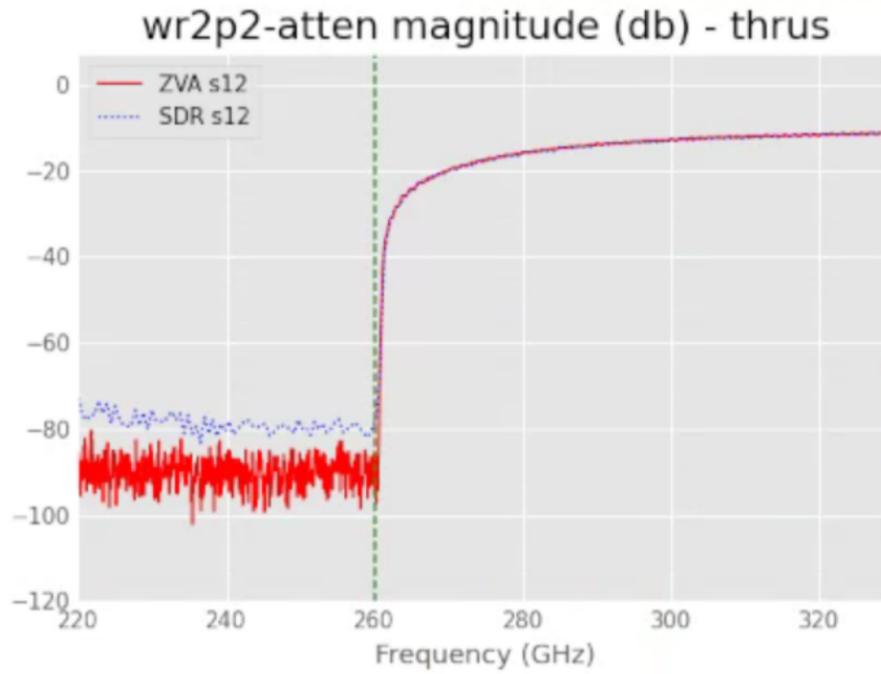


Figure 6.14: Attenuator Insertion Loss Magnitude dB

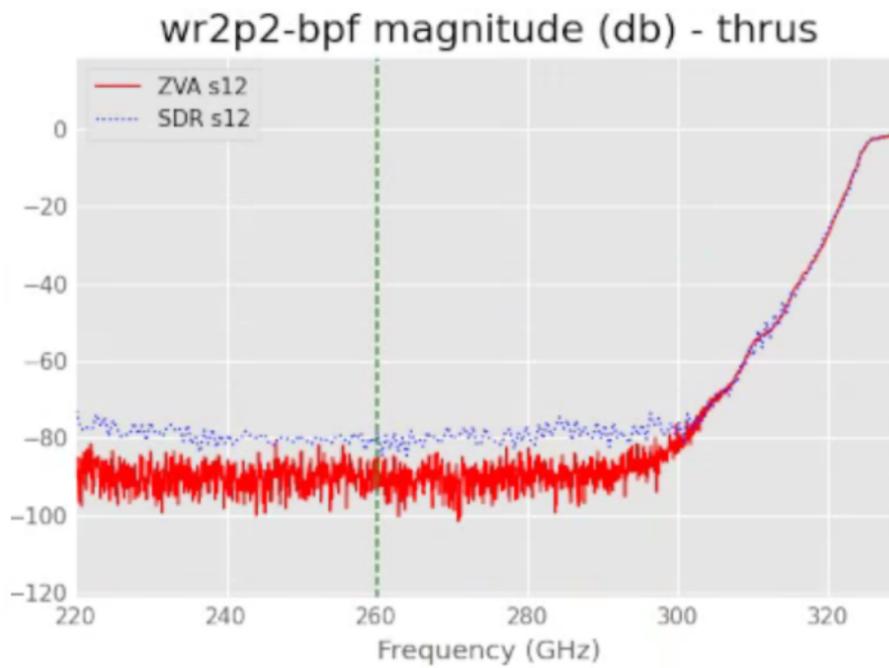


Figure 6.15: Filter Insertion Loss Magnitude dB

Finally, Figure 6.14 and Figure 6.15 show insertion loss measurements. Here, a key limitation of the SDR-based system is evident: its effective noise floor was approximately -80 dB, while the ZVA achieved around -90 dB. This limitation may stem from insufficient power reaching the SDR MEAS port, which could be improved by applying the dynamic range optimization strategies outlined in Chapters 2 and 5. Alternatively, degraded calibration accuracy – perhaps due to poorer measurement repeatability – could also contribute to the worsening of the noise floor.

6.7 Chapter Summary

This chapter explored the challenges and methodologies involved in extending the capabilities of SDRs to mmWave and sub-mmWave frequencies by using VNA frequency extension modules. While SDRs are traditionally limited in terms of frequency range, their utility for high-frequency measurements can be explored by pairing them with frequency extension modules.

The chapter began by reviewing the architectural and performance limitations of SDRs, contrasting them with specialized laboratory instruments such as VNAs. VNAs typically have very narrow IBW and provide metrology-grade performance, which helps enhance their low system noise figure (NF). SDRs are, by their nature, broadband devices, which limits their ability to sweep quickly when performing VNA measurements.

Initial experiments utilizing the frequency extenders demonstrated upconversion of the SDR transmitted signal to sub-mmWave bands using a COTS-based frequency upconversion chain. SINAD was used as a quantitative measure of performance, particularly distortion, which degrades signals. The measurements confirmed the feasibility of transmitting and receiving modulated signals at frequencies up to 1 THz using frequency extenders and an SDR. Notably, this experiment also revealed challenges, such as bandwidth and phase multiplication due to the frequency multipliers in the frequency extenders – highlighting the need for digital pre-distortion if using digital modulation formats.

The chapter also explored performing vector network analysis using the extenders with the SDR as a backend receiver. A system architecture was constructed in which the high-frequency RF and LO signals were sourced from signal generators. In contrast, the REF and MEAS IF signals

were routed to the SDR for digitization and processing. Using this system, calibrated S-Parameter measurements were obtained for several waveguide components and compared to results from a commercial VNA. The SDR-based measurements showed excellent agreement with the VNA, with the primary limitation being a slightly higher effective noise floor.

Overall, this chapter demonstrated that SDR-based VNA measurements at sub-mmWave wavelengths are feasible and can be accurate and repeatable when properly designed and calibrated. These findings pave the way for future research and experimentation using SDRs at sub-mmWave wavelengths.

Chapter 7

Conclusions and Suggestions for Future Work

This dissertation has explored the feasibility and performance of commercial off-the-shelf (COTS) software-defined radio (SDR)-based vector network analyzers (VNAs), including their extension to submillimeter wave (sub-mmWave) frequencies. The primary contributions of this work to the field of test and measurement are as follows:

- A procedure was established for the development of a low-frequency SDR-based VNA prototype. This includes selection and characterization of the radio to maximize raw dynamic range (DR) and minimize noise figure (NF), test set component selection methodology, practical mechanical considerations, a robust digital signal processing (DSP) approach for measurement extraction (even under imperfect conditions), and calibration and verification techniques. The system was benchmarked against a commercial VNA. To the author's knowledge, this represents the most comprehensive characterization of an SDR-based VNA to date.
- A flexible and extensible software framework was developed to support this system, integrating control of laboratory instruments with a high-level, declarative measurement architecture.
- The SDR was successfully interfaced to commercial VNA frequency extenders. This effort demonstrated that calibrated measurements at sub-mmWave frequencies are feasible using an SDR backend, marking the first known demonstration of such a system.

- Several critical conclusions were drawn regarding the use of COTS SDRs in VNA systems. These include the importance of advanced synchronization or shared local oscillator (LO) and shared phase-locked loop (PLL) for repeatability, the necessity of power leveling, and the observation that radio firmware stability may outweigh published specifications in determining practical capabilities.

The results presented herein may prove useful for future researchers investigating SDR-based VNA architectures, particularly those interested in high-frequency applications. The architecture supports a high degree of flexibility, enabled by a task-oriented, declarative sweep framework. Moreover, limitations in tuning repeatability and phase stability were identified as a critical weakness in current SDR platforms for VNA use. This limitation could be further characterized by introducing randomized phase errors of comparable magnitude into commercial VNA measurements and observing the resulting degradation in effective dynamic range. Such analysis may also inform broader classes of RF measurements.

One area for potential future work is the application of SDR-based VNAs to in-situ plasma monitoring, potentially in the role of a process endpoint detector. If appropriate resonance conditions can be identified for the plasma of interest, it may be possible to detect etch completion by observing the presence or absence of particular gases via their scattering signatures. The extension of this concept to a three-dimensional measurement using multipoint architectures—enabled by dense RF switching networks and additional couplers—could enable real-time process optimization in semiconductor fabrication.

Another area for further study is a comparative analysis of emerging VNAs integrated circuits (ICs) and SDRs. The landscape of available hardware has evolved considerably since the inception of this work. For example, the Analog Devices ADL5960 integrates a VNA frontend into a compact package, though it lacks integrated synthesizers or analog-to-digital converters (ADCs), and may thus benefit from pairing with a high-performance SDR. Likewise, newer SDRs may offer improved synthesizer designs with superior tuning speed and phase repeatability, directly addressing limitations identified in this work.

While the software developed herein has not yet been publicly released, it is available upon request. Future public distribution is anticipated. It is hoped that both this research and its supporting

software will facilitate continued innovation in measurement science and the broader adoption of SDR-based instrumentation.

Bibliography

- [1] *LimeSDR Made Simple Part 9: It's a VNA Too*. [Online]. Available: <https://myriadrf.org/news/limesdr-made-simple-part-9-vna/> (visited on 12/08/2022).
- [2] A. Raza, A. Jabba, D. A. Sehrai, H. Atiq, and R. Ramzan, "SDR Based VNA for Characterization of RF Sensors and Circuits," in *2021 1st International Conference on Microwave, Antennas & Circuits*, Dec. 2021, pp. 1–4. DOI: 10.1109/ICMAC54080.2021.9678273.
- [3] J. Marimuthu, "Design of wideband microwave frontend for microwave-based imaging systems," doi: 10.14264/uql.2016.955, PhD Thesis, The University of Queensland, 2016.
- [4] P. Meaney, A. Hartov, S. Bulumulla, *et al.*, "A 4-channel, vector network analyzer microwave imaging prototype based on software defined radio technology," en, *Review of Scientific Instruments*, vol. 90, no. 4, p. 044 708, Apr. 2019, ISSN: 0034-6748, 1089-7623. DOI: 10.1063/1.5083842. [Online]. Available: <https://pubs.aip.org/rsi/article/90/4/044708/283211/A-4-channel-vector-network-analyzer-microwave> (visited on 03/26/2025).
- [5] M. I. Vidotto, F.E. Veiras, and P. A. Sorichetti, "Software defined radio for vector network analysis: Configuration, characterization, and calibration," *Measurement*, vol. 189, 2022. DOI: 10.1016/j.measurement.2021.110468.
- [6] M. Steer, *Radio Systems* (Microwave and RF Design). 2019, vol. 1, Doi: <https://doi.org/10.5149/9781469656915> Steer, ISBN: 978-1-4696-5691-5.
- [7] Walt Kester, *Fundamentals of Sampled Data Systems*. [Online]. Available: <https://www.analog.com/media/en/training-seminars/design-handbooks/data-conversion-handbook/chapter2.pdf>.
- [8] Ian Poole, *What is an Image Rejection Mixer?* [Online]. Available: <https://www.electronics-notes.com/articles/radio/rf-mixer/image-rejection-mixer.php> (visited on 04/21/2025).
- [9] D. M. Pozar, *Microwave engineering*, 4th ed. Hoboken, NJ: Wiley, 2012, OCLC: ocn714728044, ISBN: 978-0-470-63155-3.
- [10] *Noise Figure Measurement Methods and Formulas*, 2003. [Online]. Available: <https://www.analog.com/en/resources/technical-articles/noise-figure-measurement-methods-and-formulas--maxim-integrated.html> (visited on 03/14/2025).
- [11] *Noise Figure Measurement Accuracy: The Y-Factor Method*. [Online]. Available: <https://www.keysight.com/us/en/assets/7018-06829/application-notes/5952-3706.pdf> (visited on 03/14/2025).
- [12] Dr. Steve Arar, *Explore the Y Factor Method for Noise Figure Measurement*, Jun. 2023. [Online]. Available: <https://www.allaboutcircuits.com/technical-articles/explore-the-y-factor-method-for-noise-figure-measurement/> (visited on 03/14/2025).
- [13] Ettus Research, *UBX - Ettus Knowledge Base*. [Online]. Available: <https://kb.ettus.com/UBX> (visited on 04/01/2021).
- [14] Ettus Research, *SBX - Ettus Knowledge Base*. [Online]. Available: <https://kb.ettus.com/SBX>.

- [15] DSP Related, *Short-Time Fourier Transforms*. [Online]. Available: https://www.dsprelated.com/freebooks/sasp/Short_Time_Fourier_Transform.html (visited on 04/22/2025).
- [16] Analog Devices, *Clock (CLK) Jitter and Phase Noise Conversion*, Dec. 2004. [Online]. Available: <https://www.analog.com/en/resources/technical-articles/clock-clk-jitter-and-phase-noise-conversion.html> (visited on 04/21/2025).
- [17] B. P. Lathi, *Signal processing and linear systems* (Oxford series in electrical and computer engineering), eng. New York: Oxford Univ. Press, 1998, ISBN: 978-0-19-521917-3.
- [18] Frederic J. Harris, "On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform," *Proceedings of IEEE*, vol. 66, pp. 51–83, Jan. 1978. DOI: 10.1109/PROC.1978.10837.
- [19] Stanford Research Systems, *About Lock-In Amplifiers*. [Online]. Available: <https://www.thinksrs.com/downloads/pdfs/applicationnotes/AboutLIAs.pdf>.
- [20] S. Bhattacharyya, R. N. Ahmed, B. B. Purkayastha, and K. Bhattacharyya, "Implementation of Digital Lock-in Amplifier," *Journal of Physics: Conference Series*, vol. 759, p. 012096, Oct. 2016, ISSN: 1742-6588, 1742-6596. DOI: 10.1088/1742-6596/759/1/012096. [Online]. Available: <https://iopscience.iop.org/article/10.1088/1742-6596/759/1/012096> (visited on 04/12/2025).
- [21] *FFTW*. [Online]. Available: <https://www.fftw.org>.
- [22] M. Steer, *Networks* (Microwave and RF Design), 3rd ed. 2019, vol. 3, Doi: <https://doi.org/10.5149/9781469656953>. Steer, ISBN: 978-1-4696-5695-3.
- [23] Geeks By Geeks, *Mason's Gain Formula in Control System*. [Online]. Available: <https://www.geeksforgeeks.org/masons-gain-formula-in-control-system/> (visited on 04/21/2025).
- [24] Rob Walton, *Mason.m*, Jul. 2016. [Online]. Available: <https://www.mathworks.com/matlabcentral/fileexchange/22-mason-m> (visited on 04/21/2025).
- [25] *Directional Couplers: Their Operation and Application*, Oct. 2020. [Online]. Available: <https://blog.minicircuits.com/directional-couplers-their-operation-and-application/> (visited on 04/11/2025).
- [26] Marki Microwave, *Directivity and VSWR Measurements*, May 2010. [Online]. Available: <https://markimicrowave.com/technical-resources/white-papers/directivity-vswr-measurements/> (visited on 04/21/2025).
- [27] Anritsu, *Reflectometer Measurements - Revisited*, Apr. 2000. [Online]. Available: https://maximinstruments.com/pdf_files/ANRITSU_S332C_Application_Note.pdf (visited on 04/21/2025).
- [28] D. Rytting, "ARFTG 50 year network analyzer history," in *2008 71st ARFTG Microwave Measurement Conference*, Atlanta, GA, USA: IEEE, Jun. 2008, pp. 1–8. DOI: 10.1109/ARFTG.2008.4633319. [Online]. Available: <https://ieeexplore.ieee.org/document/4633319/> (visited on 03/26/2025).
- [29] EURAMET, *Guidelines on the Evaluation of Vector Network Analysers (VNA)*, Mar. 2018. [Online]. Available: https://www.euramet.org/Media/news/I-CAL-GUI-012_Calibration_Guide_No._12.web.pdf (visited on 04/21/2025).
- [30] Doug Rytting, *Network Analyzer Error Models and Calibration Methods*. [Online]. Available: http://emlab.uiuc.edu/ece451/appnotes/Rytting_NAModels.pdf (visited on 04/21/2025).
- [31] Ettus Research, *UHD*. [Online]. Available: <https://github.com/EttusResearch/uhd>.
- [32] Reichel et al, "METHOD OF MEASURING THE EFFECTIVE DIRECTIVITY AND/OR RESIDUAL SYSTEM PORT IMPEDANCE MATCH OF A SYSTEM-CALIBRATED VECTOR NETWORK ANALYSER," 7,076,382.

- [33] T. Reichel, H. Jager, and J. Schubert, *A new approach for the verification and calibration of vector network analyzers*, 2005. [Online]. Available: http://resource.npl.co.uk/docs/networks/anamet/members_only/meetings/23/thomas_reichel.pdf.
- [34] Aleksandr A. Savin, “A novel factor verification technique for one-port vector network analyzer,” Nuremberg, Germany: IEEE, Dec. 2013, ISBN: 978-2-87487-031-6. DOI: 10.23919/EuMC.2013.6686590.
- [35] J. Mitola, “The software radio architecture,” *IEEE Communications Magazine*, vol. 33, no. 5, pp. 26–38, May 1995, ISSN: 01636804. DOI: 10.1109/35.393001. [Online]. Available: <http://ieeexplore.ieee.org/document/393001/> (visited on 03/27/2025).
- [36] J. Mitola, “Software radios-survey, critical evaluation and future directions,” in *[Proceedings] NTC-92: National Telesystems Conference*, Washington, DC, USA: IEEE, 1992, pp. 13/15–13/23, ISBN: 978-0-7803-0554-0. DOI: 10.1109/NTC.1992.267870. [Online]. Available: <http://ieeexplore.ieee.org/document/267870/> (visited on 03/26/2025).
- [37] M. Nayak, U. Bhanja, D. Parida, D. Dash, and K. D. Sa, “A real time implementation of spectrum sensing system using software defined radio,” in *2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, Kerala State, Kannur, India: IEEE, Jul. 2017, pp. 603–607, ISBN: 978-1-5090-6106-8. DOI: 10.1109/ICICICT1.2017.8342631. [Online]. Available: <http://ieeexplore.ieee.org/document/8342631/> (visited on 03/27/2025).
- [38] A. M. Wyglinski, D. P. Orofino, M. N. Ettus, and T. W. Rondeau, “Revolutionizing software defined radio: Case studies in hardware, software, and education,” *IEEE Communications Magazine*, vol. 54, no. 1, pp. 68–75, Jan. 2016, ISSN: 0163-6804. DOI: 10.1109/MCOM.2016.7378428. [Online]. Available: <http://ieeexplore.ieee.org/document/7378428/> (visited on 03/26/2025).
- [39] O. Abari, H. Hassanieh, M. Rodreguiz, and D. Katabi, “Poster: A millimeter wave software defined radio platform with phased arrays,” in *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*, New York City New York: ACM, Oct. 2016, pp. 419–420, ISBN: 978-1-4503-4226-1. DOI: 10.1145/2973750.2985258. [Online]. Available: <https://dl.acm.org/doi/10.1145/2973750.2985258> (visited on 03/26/2025).
- [40] T. S. Rappaport, Y. Xing, O. Kanhere, *et al.*, “Wireless Communications and Applications Above 100 GHz: Opportunities and Challenges for 6G and Beyond,” *IEEE Access*, vol. 7, pp. 78 729–78 757, 2019, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2019.2921522. [Online]. Available: <https://ieeexplore.ieee.org/document/8732419/> (visited on 03/26/2025).
- [41] *Vector Network Analyzer Extension Modules (VNAX)*. [Online]. Available: <https://vadiodes.com/en/products/vector-network-analyzer-extension-modules> (visited on 04/01/2021).
- [42] Ettus Research, *X300/X310 - Ettus Knowledge Base*. [Online]. Available: <https://kb.ettus.com/X300/X310> (visited on 04/01/2021).
- [43] D. Widmann, C. Moore, M. B. Eller, *et al.*, “SDR for Metrology and Communications at Submillimeter Wavelengths,” in *2021 46th International Conference on Infrared, Millimeter and Terahertz Waves (IRMMW-THz)*, Chengdu, China: IEEE, Aug. 2021, pp. 1–2, ISBN: 978-1-7281-9424-0. DOI: 10.1109/IRMMW-THz50926.2021.9567077. [Online]. Available: <https://ieeexplore.ieee.org/document/9567077/> (visited on 03/26/2025).
- [44] *GNURadio*. [Online]. Available: <https://github.com/gnuradio/gnuradio> (visited on 04/01/2021).
- [45] MathWorks, *MATLAB*. [Online]. Available: <https://www.mathworks.com/products/matlab.html> (visited on 04/21/2025).
- [46] A. Darwish, J. Qiu, E. Viveiros, H. A. Hung, and J. Hesler, “Efficient linear transmission of complex waveforms at 216 GHz using nonlinear multiplier chains,” in *2016 IEEE MTT-S International Microwave Symposium (IMS)*, San Francisco, CA: IEEE, May 2016, pp. 1–3, ISBN: 978-1-5090-0698-4. DOI: 10.1109/MWSYM.2016.7540046. [Online]. Available: <http://ieeexplore.ieee.org/document/7540046/> (visited on 04/10/2025).

Appendix A

Software Architecture

This appendix documents the architectural evolution of the software platform developed to support measurement tasks in this research. Initially designed as a vector network analyzer (VNA) control system, it has since grown into a generalized testing and automation framework. The project began as a minimalist BASH script, invoking calls to a GNU Radio-based Python script to perform vector network measurements. Over time, it evolved into something more comparable to OpenTAP, LabVIEW, or TANGO. It is not that script anymore, nor is it merely SDRVNA anymore – it has become a fledgling test-and-measurement framework in its own right.

A.1 SDRVNA

The rationale behind the architecture is tightly bound to the project's history. At first, the sole focus of the software was to perform VNA measurements using an software-defined radio (SDR). The simplest way to do this is to call one of the UHD example programs "rx_samples_to_file" over and over again in a loop in a shell script. This approach had simplicity in spades, but it was almost unbearably slow and did not have a good way to facilitate logic or calibration. Additionally, stability problems were observed when reinitializing the radio repeatedly.

The first logical step in the evolution of the SDRVNA was to leverage GNU Radio to perform the digital signal processing (DSP) on the fly and save complex a-wave, b-wave, and s-parameter values to file. However, this still functioned much the same way; GNU Radio was called on each iteration

of a loop in a shell script and sweeps were still agonizingly slow due to the need to re-initialize the radio for every sweep point. This approach displayed stability issues in addition to the slow sweeping speed, and it likewise did not provide the logic for adequately storing and calibrating with s-parameters. It served as a proof-of-concept for the feasibility of utilizing an SDR for performing vector network measurements; however, the measurements needed to be faster to be practical.

What followed next was a Python script that integrated a GNU Radio exported "topblock.py" script. It provided functions for tuning the frequency on the fly, allowing for many measurements to be performed without needing to re-initialize the radio. Finally, the sweep speeds had become a little more practical. However, it still showed issues with phase repeatability and field-programmable gate array (FPGA) stability, which were ultimately traced to the TwinRX daughterboard. This radio's command queue buffer is too short to ensure phase-coherent tuning. Data processing speeds were a bottleneck that prevented anything remotely close to real-time speeds. GNU Radio's producer/consumer-style DSP chain built on streaming primitive data types seemed overly restrictive. It also does not seem to be inherently well adapted to a task-oriented, parameter sweep application. The combination of performance, stability, repeatability, and flexibility issues made it necessary to migrate to a platform that provides the low-level access needed to address stability and repeatability problems while also addressing the desire for flexibility and near-real-time performance.

A rewrite of SDRVNA using C++ and the Qt framework soon followed. Most stability issues were eventually worked around (discussed in Chapter 5), and measurement sweep speed was significantly increased, making further experimentation more practical. This iteration also included a simple GUI for a one-port VNA implementation that allowed for one-port calibration and provided a real-time data display. In many senses, this "quick-and-dirty" application was a dirty hack, but it worked. The problem was that it was not scalable. There was no sane way forward for it to switch ports, use alternate signal sources, perform power calibrations, perform two-port calibrations, perform overdetermined calibrations, and a range of other problems, like integrating with power meters, spectrum analyzers, power supplies, and signal generators among others. Thus began the imagining of two things – a means of integrating another library for handling radio frequency (RF)/microwave data and an architecture that freely supports the integration of various instruments.

A.2 RMS (Radio Metrology Software)

The first step towards advancing the SDR-based VNA was to integrate more powerful software for data manipulation and calibration. One of the most powerful tools available for processing VNA data is METAS' VNA tools. While its UI has a great deal of functionality, it would be very challenging to embed into this architecture. The Python-based scikit-rf package is another powerful tool initially designed by a past student in our research group, Alex Arsenovich. Despite being a Python package, it was still possible to make direct use of it by first painstakingly creating a set of static bindings for the library using pybind11. This approach gave the program full access to nearly all of scikit-rf's extensive list of features. Additionally, scripts for processing data can be prototyped in Python and then easily translated and implemented here. Unfortunately, the nature of static bindings like this is that they are tightly tied to a specific version of the scikit-rf application programming interface (API), making the implementation somewhat fragile and requiring high maintenance. Debugging problems in the bindings tends to be complex and opaque. The performance of the application using these bindings also leaves much to be desired.

The second primary architectural requirement was instrument control. It was at this point that the software architecture at large gravitated towards becoming a plugin-based architecture, with plugin interfaces for transmitters, receivers, SDRs, power meters, port switchers, instruments, spectrum analyzers, power meters, and power supplies – things that I needed pursuant to my research with the SDRs. These are intended to provide abstractions; for example, a signal generator is a transmitter, but so is an SDR. Since they share a standard interface, if the plugins implementing these interfaces are specified in a way that allows them to be loaded dynamically, the applications utilizing those interfaces can switch from one plugin implementing that interface to another without even needing to recompile. The 'instruments' interface provides an abstraction for VISA interfaces and is used by most of the plugins.

Most of the functionality for the project also shifted to a shared library. The original SDRVNA GUI was scrapped because the way it was implemented could not easily be scaled to meet the expanding needs of the project. The project, at this point, had shifted from one purely focused on vector network analyses to one that could control various lab instruments seamlessly. As such, the name for the project, SDRVNA, seemed to narrow a description, leading to the project being renamed "RMS" - the Radio Metrology Software.

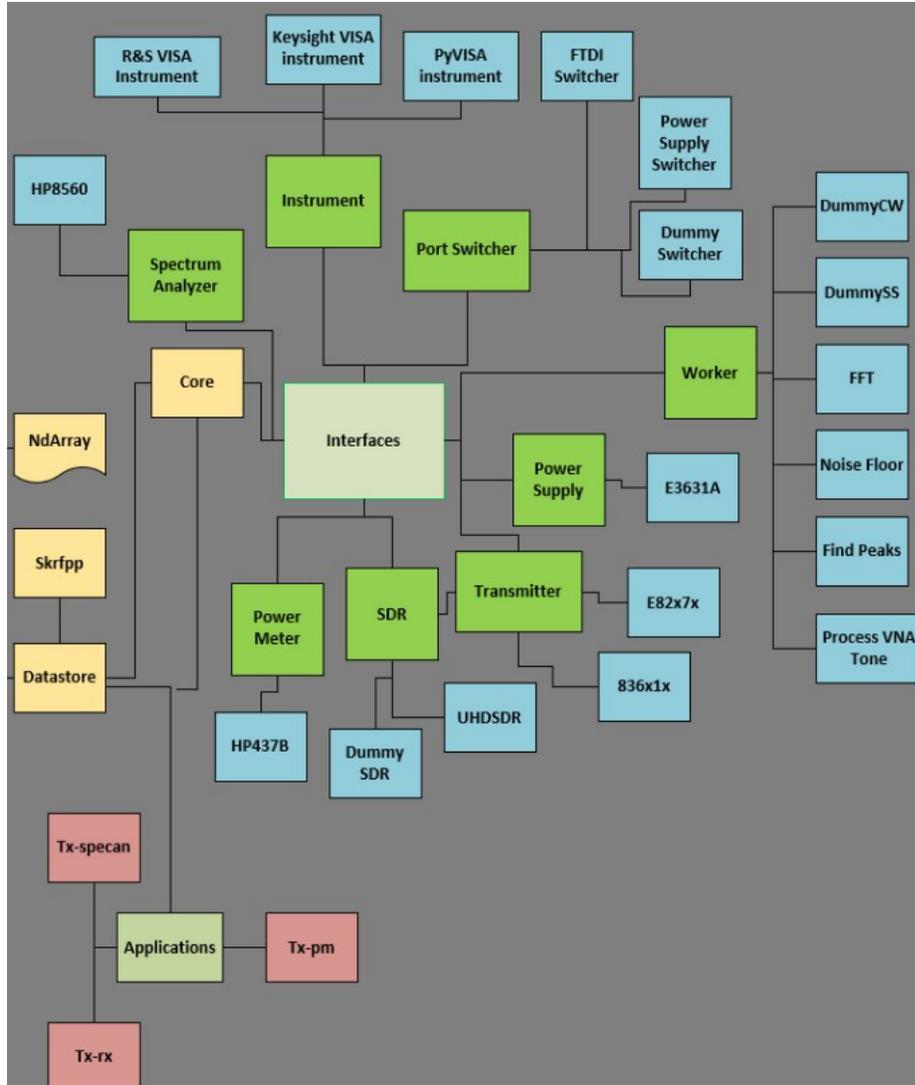


Figure A.1: Libraries and plugin-system block diagram

The integration of RF/microwave libraries, shift of functionality into a core library, and implementation of a plugin system together made it not just possible but easy to implement different kinds of sweep measurements. Early creations were power calibration sweeps using a power meter, drop-in replacement of a signal generator for the transmitter portion of SDR-based VNA measurements, and sweeps with a spectrum analyzer to analyze spurious content. Despite their similarities, each of these measurements required writing a new application; this inspired the search for a way to control these measurements dynamically.

Early efforts toward the implementation of dynamic measurement sweeps, while a step in the right direction, were almost more of a headache than they were worth. The code could generate a static

parameter sweep from a set of ranges and variable names or by being directly given a list of tuples. Procedural flow is dictated by a sequence of instructions in a JSON file rather than through handwritten procedural code. However, objects to be acted on had to be initialized manually, and operating on them required the use of a shim to call the required functions. This early implementation allowed for arbitrary parameter sweeps, for example, across frequency, gain, and port combinations. It also allowed for the reuse of arbitrary intermediate results across stages. While being able to specify some of the procedural flow via a JSON file helped encapsulate some of the sweeping logic, the requirement for manual initialization and registration, the lack of implicit synchronization at the library level, and the need for shim objects to execute tasks meant that in some ways utilizing this abstraction was more work than it was worth.

A.3 AERIS

The next logical step in the evolution of this architecture was to move beyond the earlier hybridized system, one that combined partially declarative JSON configuration with hardcoded initialization logic and shim-objects to execute tasks, toward a model that fully embraces dynamic, data-driven sweeping. This transition also marked a philosophical shift away from a purely radio-metrology focus, and when compounded with the realization that the name RMS conflicted with other existing software projects, prompted a final renaming to AERIS (AERIS is an Extensible Research and Instrumentation System).

As AERIS has evolved into something far more capable than its beginnings would have suggested, it is increasingly comparable in concept to established architectures like OpenTAP, LabVIEW, or TANGO – though much leaner and purpose-built for sweeping tasks and tight integration with high-performance general-purpose processing code.

This new architecture, still under development, is centered on a fully declarative asynchronous execution engine. Two JSON files define its runtime behavior. The first maps plugin instances by the plugin’s IID - which uniquely identifies the plugin object – and a name uniquely identifying each instance. The other file describes the execution flow: defining parameter sweeps, function calls, dependencies, resource usage, synchronization, and control flow. In this architecture, tasks are generated dynamically rather than statically precomputed, allowing for:

- Preemption of the task queue, for example, to retry failed measurements
- Looping and data-driven conditional logic * Interleaving tasks allows tuning for the following data point before processing to get a jump on settling times
- Calls to plugin methods, arbitrary free functions, or static methods via Qt’s signal-slot mechanism * Synchronization logic embedded into the scheduler and interfaces rather than in application code

AERIS was more of a slow, clumsy evolution than a framework design project. Features and complexity were added when and as needed. However, with each iteration, it increasingly serves as a general-purpose extensible automation platform. Ultimately, the goal of the most recent revision was to enable arbitrarily complex, nested sweeps without the user having to write or compile code, enabling full orchestration with the runtime behavior defined by a JSON-based task specification. Given this, comparisons to other orchestration frameworks are warranted, even if AERIS is not as mature. The following table compares the features of AERIS with several other software architectures.

Table A.1: Feature comparison between AERIS and other frameworks

Feature / Compatibility	AERIS	LabVIEW	GNU Radio	TANGO / EPICS	OpenTAP
Plugins	Yes	Yes	Yes	Yes	Yes
Declarative Scheduling	Yes (JSON)	Yes (Visual)	No	Partial	Yes
Nested Sweeps	Yes	Yes	No	Unclear	No
Async Tasks	Yes	Yes	Yes	Yes	No
Size/Portability	Small	Large	Medium	Large	Medium
GUI tooling	No	Yes	Yes	Partial	Yes

Compared to LabVIEW and TANGO, AERIS is exceptionally lightweight, measuring only a few MB in size, and runs efficiently on general-purpose processors. LabVIEW’s visual programming is conceptually similar to AERIS’s declarative model, but AERIS intends to integrate tightly with C++ code without the aid of IPC (interprocess communication). Although LabView’s development environment is extremely large, it can deploy relatively lightweight single-purpose applications. GNU Radio is built on an asynchronous stream-processing model, but it does not offer task-level orchestration (e.g. loops, conditionals, dependencies) in the way that AERIS or OpenTAP do. Although

GNU Radio could be embedded in an AERIS plugin, it would only address SDR/DSP-related components, not the broader orchestration problem. PyVISA is widely used and useful for hardware control, but does not warrant a direct comparison. AERIS optionally uses PyVISA (or other VISA) internally as a backend for SCPI-driven devices but layers a plugin interface and scheduling model on top of it. OpenTAP is the closest comparable framework. Both use a plugin system and support declarative task sequencing, but AERIS is more focused on dynamic sweep control, nested iteration, and runtime task generation. Because AERIS treats sweeping as a core construct, it may offer more flexibility for instrument-heavy research workflows. Furthermore, the dependency on C++/Qt instead of C#/.NET should lend itself to wider platform compatibility.

A.4 Chapter Summary

AERIS began as a fixed-purpose SDR-based VNA application, but has since evolved into a flexible general-purpose research automation framework. Supports asynchronous, instrument-aware execution, plugin-based extensibility, and declarative workflows. Although similar frameworks exist, AERIS strikes a different balance between generality and research-grade usability than existing options. Its evolution reflects a broadening of research instrumentation concerns shaped by necessity, refined by experience, and driven by extensibility.