A RADAR INTERROGATOR FOR WIRELESS PASSIVE TEMPERATURE SENSING

by

JEFFREY CHARLES LAMBERT

B.S. University of Central Florida, 2008

A thesis submitted in partial fulfillment of the requirements
for the degree of Master of Science in Electrical Engineering
in the Department of Electrical Engineering and Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, FL

Spring Term
2011

ABSTRACT

In this thesis I explore radio detection and ranging (RADAR) and software defined radio (SDR) in the context of wireless passive sensor interrogation. A RADAR topology is selected based upon preliminary measurements using ordinary laboratory instrumentation and then used for construction of a prototype *X*-band wireless measurement system using commercial, off-the-shelf (COTS) components. This research explores the feasibility of wireless passive sensor interrogation through practical application of SDR and RADAR techniques to the interrogation of a wireless passive resonator signal. This work serves as a foundation for further research on sensor interrogation through establishment of critical system parameters in the design of wireless measurement systems.

To María Isabel Aldea Guevara.

ACKNOWLEGEMENTS

I would like to acknowledge the support of my advisor, Dr. Xun Gong for providing me with the opportunity to do research in the UCF Antenna, RF and Microwave Integrated System (ARMI) lab and for providing me access to the necessary equipment and instrumentation.

I thank my committee members, Dr. Linwood Jones and Dr. Wasfy Mikhael for the support they have offered me in the process of completing this thesis.

Thanks go to Dr. William Crampton for supporting me throughout my Master’s work and for providing me with the opportunity to do research in a field I never could have imagined.

Lastly, I thank the members of the UCF ARMI Lab for their support, especially Justin Luther and Siamak Ebadi, and the many faculty and staff who have helped me during the thesis process.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

ADC	Analog-to-digital converter
AGC	Automatic gain control
AM	Amplitude modulation
API	Application programming interface
ARMI	Antenna, RF and Microwave Integrated
ATSC	Advanced Television Systems Committee
BSD	Berkeley Software Distribution
CIC	Cascaded integrator-comb
COTS	Commercial, off-the-shelf
CW	Continuous waveform
DAC	Digital-to-analog converter
DAQ	Digital acquisition
DC	Direct current
DDC	Digital down-converter
DUC	Digital up-converter
DSP	Digital signal processing
	Digital signal processor
FFT	Fast Fourier transform
FIFO	First-in, first-out

FM	Frequency modulation
FMCW	Frequency modulated continuous waveform
FPGA	Field-programmable gate array
FSCW	Frequency stepped continuous waveform
GNU	GNU is not Unix
GPL	General Public License
GPLL	Gated phase-locked loop
GRC	GNU radio companion
GUI	Graphical user interface
HFSS	High Frequency Structure Simulator
IC	Integrated circuit
IDT	Interdigital transducer
IEEE	Institute of Electrical and Electronics Engineers
IF	Intermediate frequency
I^{2}C	Inter-Integrated Circuit
ISM	Industrial, scientific and medical
LNA	Low-noise amplifier
LO	Local oscillator
LSB	Lower side-band
MEMS	Micro-electro-mechanical systems

MSPS	Mega-samples per second
NCO	Numerically controlled oscillator
NF	Noise figure
NTSC	National Television Systems Committee
PA	Power amplifier
PAL	Phase alternating line
PC	Personal computer
PCB	Printed circuit board
PGA	Programmable gain amplifier
PLL	Phase-locked loop
PPF	Peak power frequency
QF	Quality factor
QFN	Quad flat no-lead
RADAR	Radio detection and ranging
RBW	Resolution bandwidth
RF	Radio frequency
RFID	Radio-frequency identification
RTD	Resistive temperature detector
SA	Spectrum analyzer
SAW	Surface acoustic wave

SSB	Single side-band
SDR	Software defined radio
SMA	Sub-miniature version “A”
SNR	Signal-to-noise ratio
SPI	Serial peripheral interface
TCP	Transmission control protocol
USB	Universal serial bus
	Upper side-band
USRP	Universal software radio peripheral
VCO	Voltage controlled oscillator
VDC	Volts direct current
VI	Virtual instrument
VNA	Vector network analyzer
VSWR	Voltage standing wave ratio

CHAPTER 1: INTRODUCTION

Sensor systems have evolved tremendously as robust sensor technology matures to meet demands for more critical and extreme measurement scenarios. Sensors capable of measuring temperatures above 1000^{o}C place restrictions on materials which can be used for sensor fabrication. Harsh environments such as those within a turbine engine or nuclear reactor make it impossible for active sensors to be used because semiconductor materials do not operate at elevated temperatures. Techniques are needed for interrogation of passive sensors: to realize new techniques, hardware is designed to employ interrogation strategies and to realize a complete, functional sensor system.

In this thesis, it is my goal to explore interrogation strategies for a resonator-based temperature sensor with a focus on inexpensive, commercial, off-the shelf (COTS) components. A receiver is designed using commercially-available hardware and interfaced to a digitizer. A personal computer (PC) is then used for subsequent analysis of the captured sensor waveform. This receiver will serve as a foundation for future research and as a building block to establish a more solid knowledge base concerning wireless sensor interrogation and interrogator design.

I layout this thesis as follows: In Chapter 2 I overview existing sensor technologies, modern radio detection and ranging (RADAR) architectures and software defined radio (SDR). In Chapter 3 I propose system-level architecture for a wireless temperature sensor interrogator. In Chapter 4 I discuss fabrication, design, and simulation techniques. I discuss measurement results and performance of the interrogator in Chapter 5. In Chapter 6 I conclude this thesis with closing thoughts and remarks. Future research plans and considerations are given in Chapter 7.

CHAPTER 2: SENSORS, RADAR, AND SOFTWARE RADIO

In this chapter I will discuss: 1. Fundamental temperature sensor technologies, wireless and wired. 2. Fundamentals of RADAR and applications to wireless sensors. 3. SDR and applications to RADAR and radio frequency identification (RFID). It is my intent that within this chapter I shall inform the reader with required background information such that he or she can understand my reasoning behind the interrogator architecture that I propose in Chapter 3.

2.1 Temperature Sensors

Modern temperature sensors exist in several different forms with each sensor type using different materials properties to yield a temperature measurement. The most significant difference in operation of modern sensors is the technique used to interrogate each type of sensor: Resistive sensors are wired devices and require complex signal conditioning to yield accurate results. Surface acoustic wave (SAW) sensors are typically coupled directly to an antenna and interrogated wirelessly.

Temperature sensors can be classified into two fundamental types by examining the techniques used to interrogate the sensor; these are: 1. Wired sensors. 2. Wireless sensors. The method of interrogation for any given sensor is realized from the materials properties that translate into measured temperature and from this stems advantages and disadvantages. For example, a significant advantage of some wireless sensors is passive operation: Passive sensors eliminate the need for a battery. Thermocouples are passive sensors; because they are wired to the measurement device, a temperature reading can be made with an inexpensive multimeter.

2.1.1 Wired Sensors

The most common wired temperature sensors can be broken down into 4 groups: 1. Resistive temperature detectors (RTDs). 2. Thermocouples. 3. Thermistors. 4. Integrated circuit (IC) sensors. The first 3 types of sensors I list here typically exist in discrete form; advances in Semiconductor technology and micro-electro-mechanical systems (MEMS) technology has made it possible to produce sensors in IC form. The most basic and accurate temperature sensor is the p-n junction of a silicon diode [1]. This sensor is referred to as a thermodiode; it uses sensitivity of diode forward-voltage to temperature to yield a temperature measurement. This effect is very well known and has been studied extensively because of the difficulties it introduces in modern, solid-state electronics design. To measure the temperature of the p-n junction, a high quality voltmeter is used. The resulting voltage measurement is then translated to a temperature value through the use of a mapping function. The mapping function is generated through carefully made measurements made against a known reference standard.

Before I discuss each sensor type in detail, I provide a brief overview: Thermistors and RTDs are resistive devices. Thermocouples are voltage devices. IC sensors provide voltage, current, or digital serial data depending upon the device. RTD sensors are the most accurate; thermocouples can be manufactured inexpensively and the resulting temperature measured with a volt meter. For a more detailed overview of the temperature sensors that I discuss here, see [2].

2.1.1.1 Thermocouples

Thermocouple sensors are composed of a junction of two dissimilar metals. At this junction, thermoelectricity is generated causing a current flow. If the thermoelectric circuit is opened, a voltage results that is proportional to the junction temperature. This type of sensor is does not require power to operate. Thermocouples are available that can sense temperature up to 2300^{o}C however they suffer from limited operation lifetime, are damaged easily with exposure to corrosive chemicals, and are very expensive. Error in thermocouple sensors can be caused from: 1. Poor signal integrity (such as lack of shielding on long cables). 2. Low quality components in the measurement device, such as low resolution analog-to-digital converters (ADCs).

2.1.1.2 RTDs

RTDs rely on temperature-dependent resistivity of a metal element or film for operation. The most common RTD is the platinum RTD because of its ability to withstand very high temperatures and its excellent measurement stability. RTDs can be fabricated from a discrete element or can be fabricated as a film on a ceramic structure. Error in RTD sensors results from: 1. Mechanical stress and vibration of the sensor. 2. Poor signal integrity. 3. Error resulting from the measurement of sensor resistance (such as self-heating of the element). With concern for measurement error, RTD sensors are available in 3 configurations: 1. 2-wire. 2. 3-wire. 3. 4-wire. While only 2 wires are necessary for resistance measurement, the additional wires of 3- and 4-wire configurations are used to compensate for resistance and voltage offsets in the leads connecting the sensor to the measurement device.

2.1.1.3 Thermistors

The thermistor is the most sensitive of the wired temperature sensors however it lacks the wide temperature range of an RTD. This sensor type tends to exhibit a nonlinear response. Unlike the RTD, it tends to have a much higher resistance over the measurement range lessening the effects of error. These sensors are more susceptible to decalibration and drift at high temperatures than are RTDs.

2.1.1.4 IC Sensors

IC sensors are similar to thermistors with note that they are much more linear. They can provide analog output as either voltage or current or digital output via serial bus. Since IC sensors are powered devices, they can exhibit self-heating which contributes to measurement error and cannot withstand high temperatures. These types of sensors are mostly used for convenience offered in reading the measured temperature.

2.1.2 Wireless Sensors

Wireless sensors are significantly different from that of wired sensors based on the measurement techniques. Wired sensors of the types discussed in the previous section can be made wireless through the use of inexpensive microcontrollers and digital communications chipsets: smart systems such as these are not covered in this section and are not of concern for this thesis. Wireless sensors are those in which interrogation is performed in some way which allows for the sensor to be physically unattached to the interrogation circuits. Coupling between the wireless sensor and the interrogation unit can be: 1. Magnetic [3]. 2. Electromagnetic [4]. 3. Optical [5]. Passive wireless sensors use only coupled energy for responding to interrogator queries. I note that some active sensors can be made passive through the use of wireless energy harvesting, as in the case of 5; these “pseudo-passive” sensors are not of interest for this thesis because they rely on materials which will not operate at very high temperatures.

The most common types of wireless sensors are SAW sensors and LC resonator sensors. SAW sensors consist of a crystalline substrate upon which a surface wave is excited through the use of an interdigital transducer (IDT) structure (see [6] for more detail). The surface of the substrate affects the surface wave phase velocity. Design of SAW structures such that the phase velocity is sensitive to the measurand allows for sensors which can measure a variety of parameters such as temperature and pressure. LC resonator sensors operate in a variety of ways: One such type of LC resonator sensor uses a capacitor with a dielectric sensitive to the measurand [7]. Through magnetic coupling the impedance of an LC tank formed from the temperature sensitive capacitor can be measured and then translated into temperature.

2.1.2.1 SAW Sensors

SAW temperature sensors exist in many forms (see [8]) including: 1. Reflective Delay Lines. 2. Resonators. 3. Dispersive Delay Lines. The most significant differences lie between delay line sensors and resonator sensors. For purposes of this thesis, the resonator sensor is of most interest. Delay line sensors and resonator sensors operate on the principle of delaying the received interrogation signal by some amount before it is reflected back to a receiver or the environment. Delay line sensors use propagation delay across the substrate to achieve sensitivity to the measurand. In resonators, delay is caused by ringing as energy stored within resonator cavities is slowly released after excitation. SAW resonator sensors are effective because of high quality factor (QF) which means that energy within the resonator is released over a long period of time (longer than that of energy scattered to the receiver in a noisy environment).

2.1.2.2 LC Resonator Sensors

LC sensors are resonators built from lumped components (inductors and capacitors) or their equivalents. These types of sensors, like SAW sensors, can be used to measure a wide variety of parameters other than temperature.

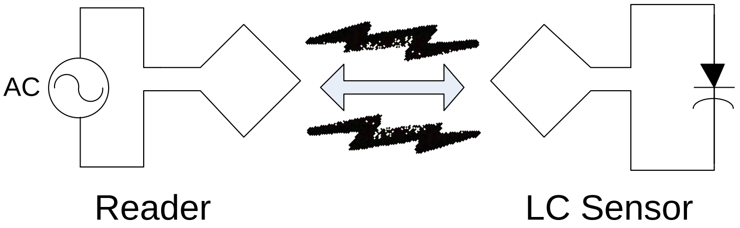


Figure 2.1: A simple LC sensor and reader. Inductive coupling is used to measure the impedance of the LC sensor. A varactor is controlled by the measurand resulting in a change of impedance over the sensors measurement range.

At the heart of an LC sensor is typically a varactor, a capacitor with a variable reactance. A varactor can be fabricated as micro-electro-mechanical system (MEMS) or can be built using larger, discrete components. In an LC resonator sensor, a resonant tank is created in a manner similar to that of an RLC circuit. A magnetic loop behaves as an inductor through which current flow is excited when an excitation flux is passed through it. Flux is generated by a loop in close proximity to that of the LC sensor and is coupled via mutual inductance. As a result of the magnetic coupling, these sensor types typically operate only over short distances from the interrogator.

2.2 RADAR

RADAR is an acronym which stands for Radio Detection and Ranging. It is a form of remote sensing that it is used for the observation of an object or objects of interest through the use of electromagnetic waves. RADAR can take many forms, can be simplified into a compact system or exist as a very large system with many complex subcomponents and subsystems, and can be used to monitor many different types of measurand (simultaneously or otherwise).

The most basic RADAR configurations consist of some form of waveform generator such as an oscillator, a transmitting antenna, a detector, and a receiving antenna [9]. Energy from the oscillator is radiated into the environment. Some of this energy is reflected back to the receiving antenna where it can be monitored by the receiver or detector. By comparing the received energy to that of the transmitted energy, we can learn details about the objects within the RADAR environment. The most common uses for RADAR are measurement of distance and measurement of velocity. More complex uses of RADAR include mapping of terrain, weather forecasting, and radio-frequency identification (RFID) tag interrogation. It should not be underestimated the complexity of even a simple RADAR; the components within a RADAR can take many forms depending upon application requirements and design.

In order to explain the most basic types of RADAR, I will classify RADAR topologies based on the most commonly used discriminant: the RADAR waveform. The RADAR waveform is as critical as the design of the hardware that composes the RADAR itself for (at least) the following reasons: 1. The RADAR waveform dictates the information that the RADAR returns to the system operator. 2. The hardware design of the RADAR changes significantly with the type of waveform specified. Based on these criterions, there are two fundamental types of RADAR (which I will list below). It is worth pointing out that classification of RADAR can be quite ambiguous - the fundamental nature of pulsed and CW RADAR can be combined given certain requirements and specifications, thus I add two additional, fundamental RADAR topologies to my list. The fundamental RADAR architectures that I will discuss here are: 1. Pulsed. 2. Continuous waveform (CW). 3. Pulse-Doppler. 4. Frequency modulated continuous waveform (FMCW).

2.2.1 Pulsed RADAR

As the name implies, Pulsed RADAR uses a waveform which is pulsed on and off to illuminate the area of observation. A transmitter operates only for a short duration at the beginning of an observation period: when the transmitter is switched off a receiver is switched on to detect environmental and other reflections from objects in the observation area.

The most significant advantage of Pulsed RADAR is that of high receiver sensitivity because of the limited duty cycle of the transmitter. When a single antenna is used in pulsed RADAR, isolation between the receiver and transmitter is necessary to prevent receiver front-end overload during transmitter operation: front-end de-sensitization from the RADAR’s own transmitter is eliminated or reduced (but can still be caused by external sources of RF power).

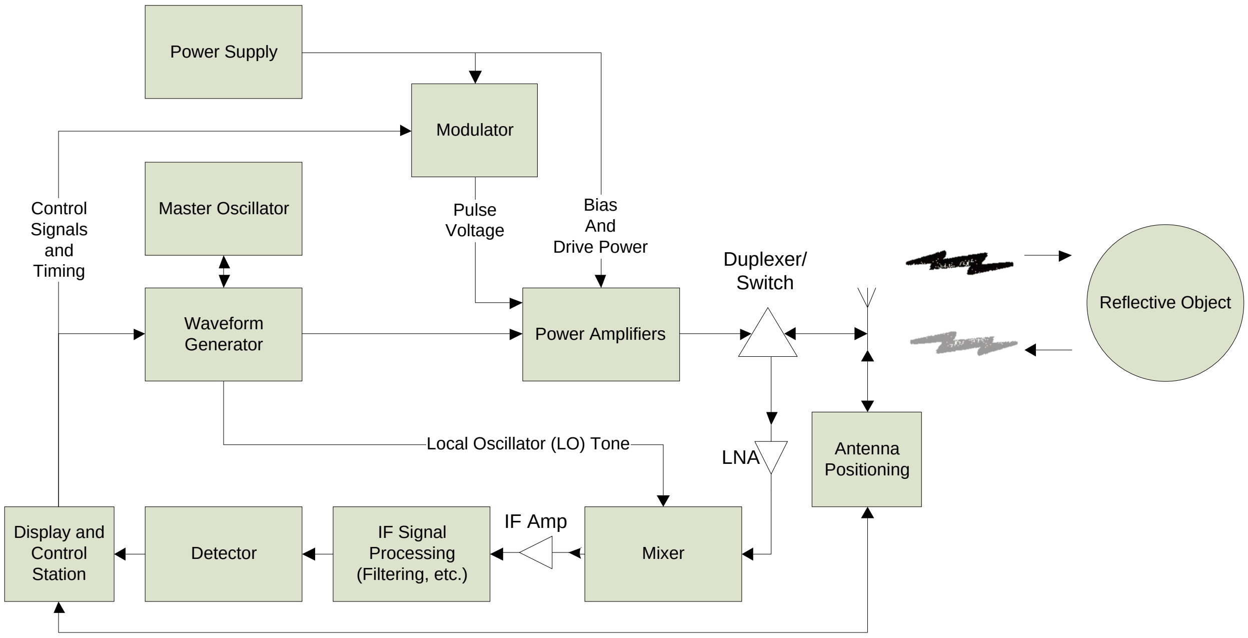


Figure 2.2: Architecture of Pulsed RADAR.

To illustrate a pulsed RADAR configuration, I have provided a block diagram for a typical pulsed RADAR architecture in Figure 2.2. At the heart of the RADAR is an oscillator which provides a drive tone for a waveform generator which feeds a power amplifier (PA) stage. A power supply provides necessary voltages for biasing the amplifier finals and providing drive current. A modulator is used to generate a control voltage which is fed into the PA: it is this signal, in combination with the waveform generator output that produces the pulse that is transmitted from the RADAR. A control station communicates with the waveform generator and modulator to keep them in sync and to produce dynamic changes in the RADAR's transmitted signal. The control station also acts to communicate with antenna positioning systems for setting the area of observation within the RADAR's complete field of view. When the RADAR transmits a pulse at an object in the field, some amount of electromagnetic energy will be reflected back. This energy is received by the antenna and amplified by a low-noise amplifier (LNA). A mixer is used to down-convert the frequency of the incoming energy to an intermediate frequency (IF). From there it is passed through subsequent signal processing such as filtering and automatic gain control (AGC) before it is processed by the detector. The output of the detector is sent to the display and control station where it is converted into information useful to the RADAR operator. To allow the transmitter and receiver to share the same antenna, some form of duplexer must be used. A duplexer can be an antenna switch or circulator, and may or may not include isolation filters to prevent excessive power from desensing or damaging the RADAR receiver.

2.2.2 Continuous Waveform RADAR

In a CW RADAR, the waveform that is generated is continuous with time and is typically transmitted with 100% duty cycle to illuminate the area of observation. CW RADAR is suited for measurement of speed because it provides a reference for which to observe Doppler shift. Doppler shift is a shift in frequency of an electromagnetic or acoustic wave relative to an observer when the source of the wave is moving [10]. Applications of CW RADAR include traffic speed detection, wind speed monitoring, and target illumination. CW RADAR offers the significant advantage that, because there is no need for timing circuitry, it can be manufactured inexpensively with a low number of components. Due to the self-mixing nature of CW RADAR, static objects in the environment are effectively invisible. This allows for CW RADAR to be used low to the ground or in enclosed environments where there may be large amounts of back scatter.

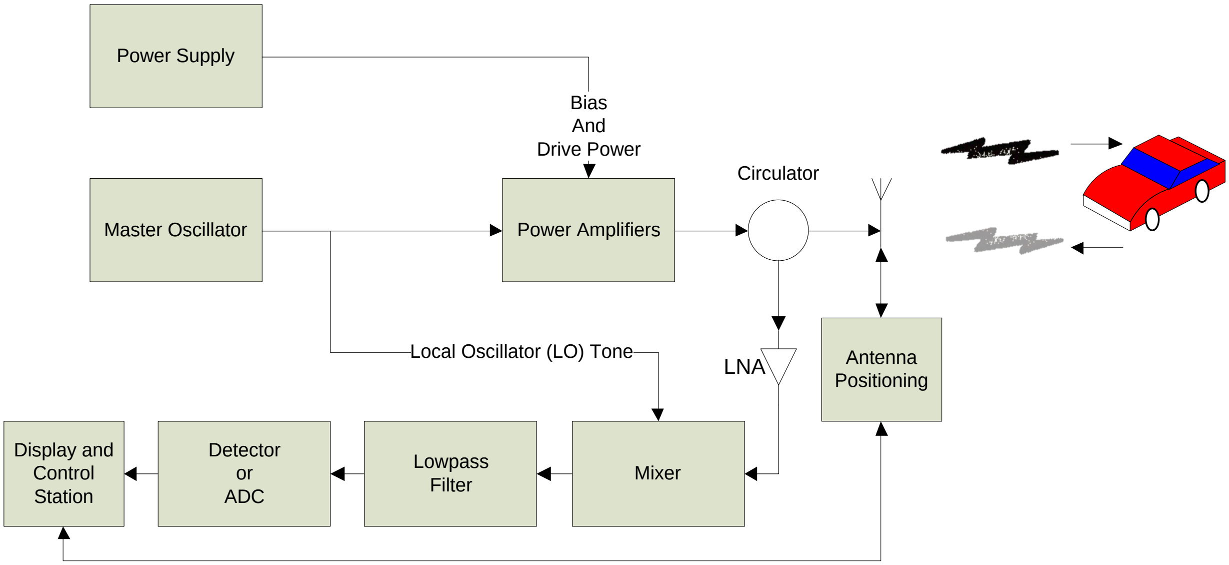


Figure 2.3: Architecture of CW RADAR.

Figure 2.3 shows a diagram of a typical CW RADAR configuration. A master oscillator feeds a PA. The PA is connected to a circulator which then drives an antenna. The antenna can be rotated through a rotor mechanism or, in the case of traffic RADAR, may simply be small enough to be used portably and hence is manually positioned. Signals received by the antenna are separated from the transmitted signal by the circulator and passed to an LNA. A mixer is used to down-convert the frequency of the received signal using a tone from the master oscillator. A low-pass filter removes spurious content and images that may result from down-conversion. This signal is then passed through a detector or an ADC where it results in a signal that can be used for display and measurement purposes. System configuration such as that of Figure 2.3 allows for phase coherence. Phase coherence improves Doppler resolution and signal-to-noise ratio (SNR) within the RADAR. Doppler estimation can be employed to further improve measurement accuracy from this type of RADAR.

2.2.3 Pulse-Doppler RADAR

Pulse-Doppler RADAR shares characteristics between pulsed RADAR and CW RADAR. In this architecture, we can measure the Doppler shift of a moving object by generating a pulse with known frequency components (it is likely that this is a narrowband pulse but it may also be broadband depending upon the system requirements). As in pulsed RADAR, a time-base is established by transmitting pulses at known time intervals; this time base can then be used to measure distance by monitoring the delay between when a pulse is transmitted and when the reflected signal is received again. To facilitate the measurement of Doppler shift, hardware architecture like that of the CW RADAR case is used (See Figure 2.3). If a narrowband pulse is transmitted from a Pulse-Doppler RADAR, the resulting Doppler shift is easy to measure if the response is down converted using the same tone as that used to generate the initial pulse. This type of RADAR shares many of the advantages of CW RADAR, however there are performance tradeoffs resulting from the hybridization of the measurement between speed and distance.

2.2.4 Frequency Modulated Continuous Waveform RADAR

FMCW RADAR is a hybrid RADAR like Pulse-Doppler RADAR. In this technique, modulation of the transmitted CW tone is used to improve SNR and to establish a time-reference. Consequently, FMCW RADAR can also be used to measure range in addition to velocity. Modulation of the transmitted tone dictates that the transmitter operates with a 100% duty cycle so isolation between the transmitter and receiver is critical for operation. Typical modulation of the transmitted CW tone uses the output of a saw tooth wave generator. In this case then, the range resolution is determined by the voltage excursion of the sweep over a relatively long period (thus giving this type of RADAR very good range resolution that is easily adjusted by the RADAR controls). The resulting slow modulation of the transmitted CW tone results in a large time-bandwidth product giving this type of RADAR a very low probability of interception. For more details refer to [11].

2.3 Software Defined Radio

SDR is a relatively new emergence in the fields of communication and remote sensing. In the traditional sense, radios are designed using modulators and demodulators built from discrete components or ICs. As embedded systems become more powerful and cheaper and the availability of high quality, low-cost ADCs and digital-to-analog converters (DACs) increases, software defined radio becomes more and more prevalent and offers previously unavailable flexibility. As explained in [12,13], software defined radio allows the underlying essence of the radio to be transformed through a software upgrade. As communications systems slowly transition from analog modulation schemes to complex digital modes, this also allows for interoperability of complex, proprietary radio systems. One such example of this is the SPEAKeasy project where a single radio is used to communicate with a number of existing systems while allowing for flexibility in designing new communication techniques without having to develop new hardware [14].

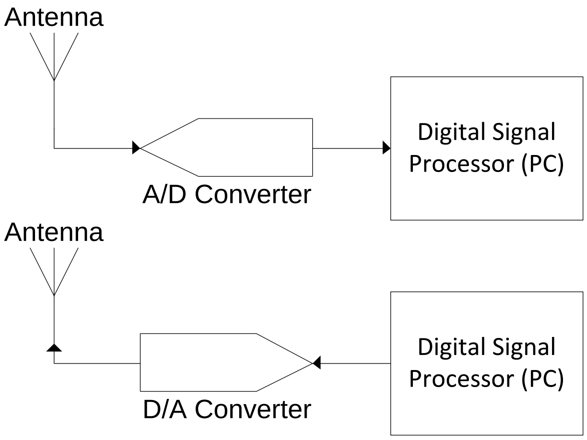


Figure 2.4: The ideal software defined radio case.

In the ideal case, the SDR architecture consists of an antenna interfaced to a digitizer. In a receiver the antenna is interfaced to an ADC and the digital output is then processed by a host which could be a computer, digital signal processor (DSP), field-programmable gate array (FPGA), or a combination of all of these. The resulting digital signal is demodulated digitally using DSP techniques. Similarly, a transmitter is constructed by interfacing a DAC to the antenna. The signal to be transmitted is generated digitally and modulated using software algorithms. This architecture requires ideal components to operate: 1. The ADCs and DACs should be free of nonlinearity, have infinite dynamic range, and have infinitely high sample rates. 2. The ADCs and DACs have impedances matching that of the antenna. 3. The signal processing has unlimited bandwidth and zero latency while processing the incoming and outgoing data streams. In the real world, ideal components don't exist resulting in much more complex systems necessary for solving issues of nonlinearity, signal processing and bandwidth limitations, signal integrity, antenna duplexing, and so on.

To provide more detail regarding the tools that I will utilize later in this thesis, I break this section into two subsections: 1. GNU Radio. 2. Universal Software Radio Peripheral (USRP). In each section, I go into detail about these tools and how to use them.

2.3.1 GNU Radio

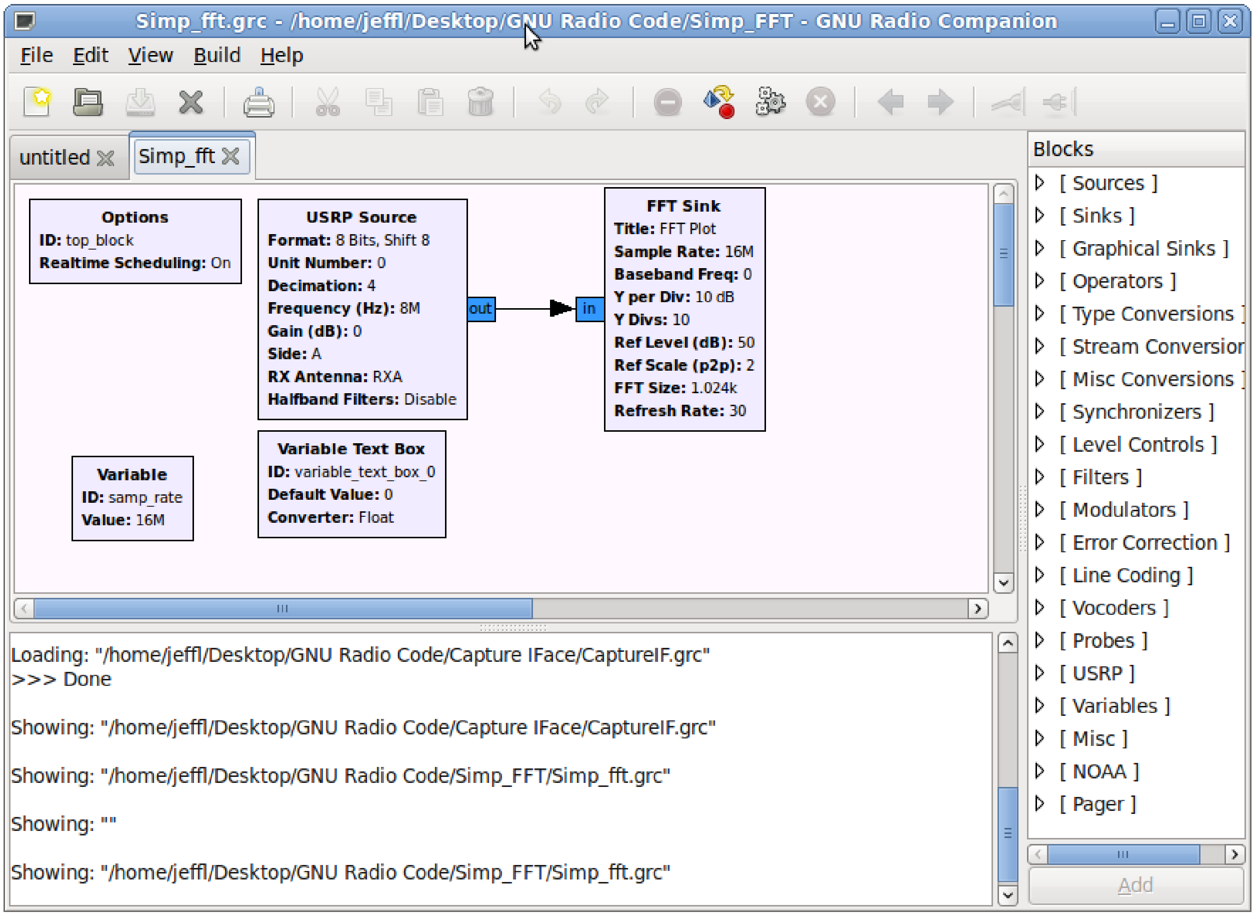
GNU Radio is a set of libraries, application programming interfaces (APIs) and software tools designed for the implementation of software radios. In SDR implementations, GNU Radio provides: 1. Signal processing. 2. Hardware support for radio interfaces such as the USRP. 3. Simplified interfaces to host resources such as sound cards and external signal processing libraries. GNU Radio when combined with a digital acquisition (DAQ) system allows for extreme flexibility. A wide-bandwidth RF front-end can be used in conjunction with GNU Radio to record entire spectrums of bandwidth and/or demodulate multiple signals from the same receiver. Modulation techniques and parameters can be tweaked with software changes instead of reworking hardware systems which can require large fabrication times and high cost.

GNU Radio is available for free as it is licensed under the GNU General Public License (GPL) version 3: it is not a commercial product. The 'GNU' in GNU Radio is a recursive acronym which reads 'GNU is not Unix'. Originally, it was a fork of Pspectra code developed under the SpectrumWare project at MIT. The code-base has been completely rewritten and today GNU Radio is an official GNU project. In only a few years' time, GNU Radio has been used for implementations of radio based systems including [15]: 1. IEEE 802.11b. 2. Bluetooth. 3. IEEE 802.15.4 (Zigbee). 4. Global System for Mobile Communications (GSM) base-stations. 5. Amplitude modulation (AM) and frequency modulation (FM) reception in wideband and narrowband formats. 6. Advanced Television Systems Committee/National Television Systems Committee (ATSC/NTSC) broadcast television reception. 7. Single- and upper/lower- sideband (SSB/USB/LSB) amateur radio transceivers.

GNU Radio consists of a set of modular signal processing blocks which are linked together in signal flow graphs. The signal processing blocks are individually written in C++. Signal flow graphs are written in Python code. Python is a multi-platform, object-oriented, interpreted language. Interpreted code, like that used in MATLAB or Octave, can be run without the need for compiling. In Python, everything is an object: objects are data structures with attributes and methods. Methods are like functions – they can return data, perform some action, or initialize the object to a default state. Attributes are properties or variables within the object. Multi-platform means that Python code will run on any operating system that Python source can be compiled upon including but not limited to: 1. Linux. 2. Windows. 3. MacOS X. 4. Berkeley Software Distribution (BSD) and other Unix systems.

2.3.1.1 GNU Radio Companion

GNU Radio Companion (GRC) is a graphical flow-chart tool designed by Josh Blum for creating and configuring signal flow graphs [16]. Similar to LabVIEW G-code and Simulink, signal flow graphs are assembled graphically with wires connecting blocks. Blocks representing the entire GNU Radio library can be selected from the menu on the right side of the screen. Basic graphical blocks are also provided to aid in designing graphical user interfaces (GUIs). Errors resulting from mismatched data-types, unconnected, or misconnected blocks are easily seen on screen. The signal flow graph can be quickly executed by clicking the ‘Run’ icon at the top of the window. From the graphically assembled block diagram, a signal flow graph is generated in Python code which can be executed as a standard Python script. Figure 2.5 illustrates the GNU Radio Companion interface.



Options
ID: top_block
Realtime Scheduling: On

USRP Source
Format: 8 Bits, Shift 8
Unit Number: 0
Decimation: 4
Frequency (Hz): 8M
Gain (dB): 0
Side: A
RX Antenna: RXA
Halfband Filters: Disable

FFT Sink
Title: FFT Plot
Sample Rate: 16M
Baseband Freq: 0
Y per Div: 10 dB
Y Divs: 10
Ref Level (dB): 50
Ref Scale (p2p): 2
FFT Size: 1.024k
Refresh Rate: 30

Variable
ID: samp_rate
Value: 16M

Variable Text Box
ID: variable_text_box_0
Default Value: 0
Converter: Float

```
Loading: "/home/jeffl/Desktop/GNU Radio Code/Capture IFace/CaptureIF.grc"
>>> Done

Showing: "/home/jeffl/Desktop/GNU Radio Code/Capture IFace/CaptureIF.grc"

Showing: "/home/jeffl/Desktop/GNU Radio Code/Simp_FFT/Simp_fft.grc"

Showing: ""

Showing: "/home/jeffl/Desktop/GNU Radio Code/Simp_FFT/Simp_fft.grc"
```

Figure 2.5: Screenshot of GNU Radio Companion software running on Ubuntu Linux.

In GNU Radio companion, multiple flow graphs can be opened simultaneously. A console at the bottom of the screen allows for review of errors generated by the flow graph and to check for buffer overflow notifications.

2.3.2 Universal Software Radio Peripheral

The Universal Software Radio Peripheral is, simply stated, a computer peripheral for interfacing analog radio frequency (RF) and microwave hardware to a computer. It was designed by team lead Matt Ettus who later formed Ettus Research LLC [17]. At the time of this writing, the USRP is available in several configurations: 1. Original USRP (version 1). 2. USRP version 2 (which is discontinued March, 2011). 3. USRP N200 series (N200/N210). 4. USRP E200 series (E200/E210).

The USRP version 1 contains two Analog Devices AD9862 mixed-signal front-end interfaces which contain two high-speed ADCs and two high-speed DACs each capable of operating at 64 mega-samples per second (MSPS) and 128 MSPS respectively [18,19]. For software radio purposes, the AD9862 ICs are interfaced to a PC using an Altera Cyclone EP1C12 FPGA (FPGA) and a Cypress EZ-USB FX2 microcontroller. The FPGA on the USRP mainboard functions as the heart of the device, controlling the flow of data and performing some basic signal processing operations. The microcontroller is interfaced to the FPGA providing the universal serial bus (USB) interface to the host PC. I will discuss the major components of the USRP in more detail within the next sections.

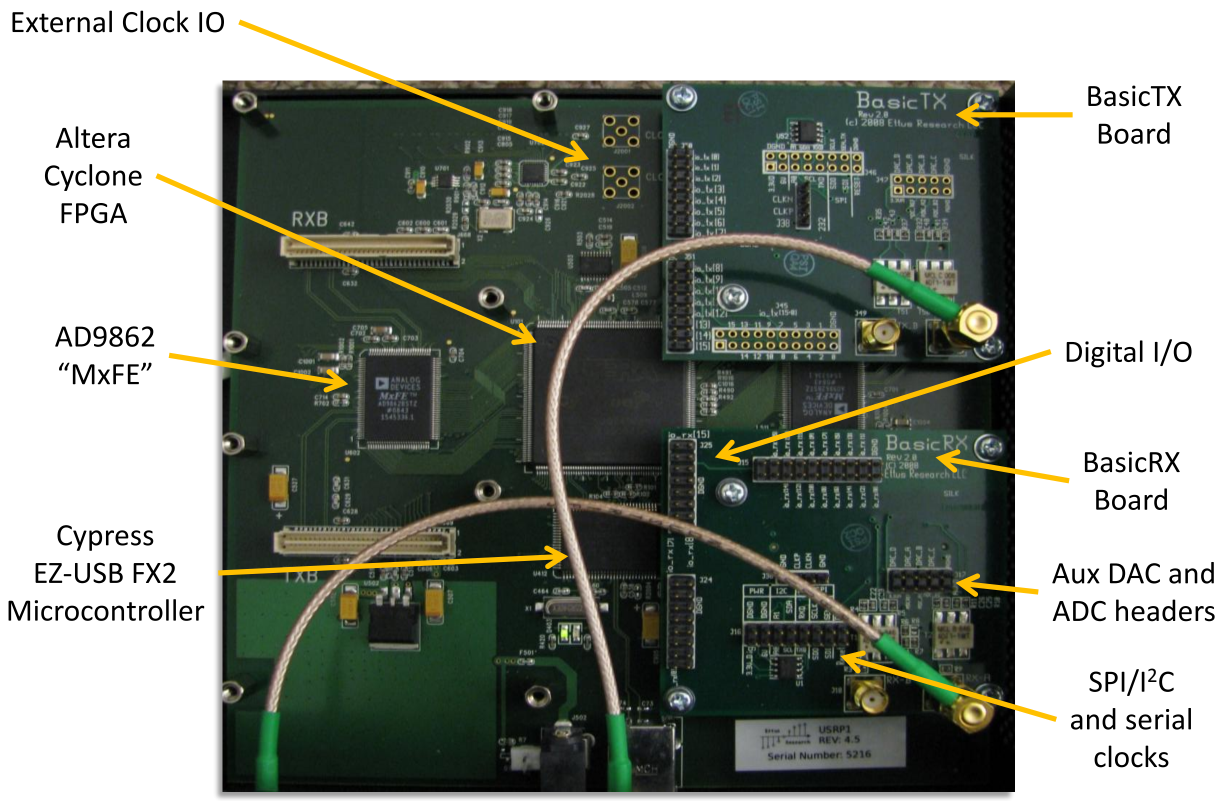


Figure 2.6: USRP mainboard with BasicTX and BasicRX daughterboards installed and SMA bulkhead cables attached.

A key to the success of the USRP is modular design as shown in Figure 2.6. The AD9862 ICs and associated analog and digital I/O are interfaced to daughterboards that attach to the USRP mainboard via interface connectors. 2 of the 4 interface connectors are for receiver boards, the other 2 for transmit boards. Ettus Research currently offers several daughterboards for use with the USRP [17]. For applications where daughterboards are not available, Ettus Research offers the BasicRX/BasicTX and LFRX/LFTX boards. The BasicRX and BasicTX boards contain impedance matching transformers to provide standard 50 Ohm RF interfaces to the AD9862 IC, as well as general purpose digital I/O lines, access to the USRP Serial Peripheral

Interface (SPI) and Inter-Integrated Circuit (I^2C) busses, and access to the onboard auxiliary DACs and ADCs.

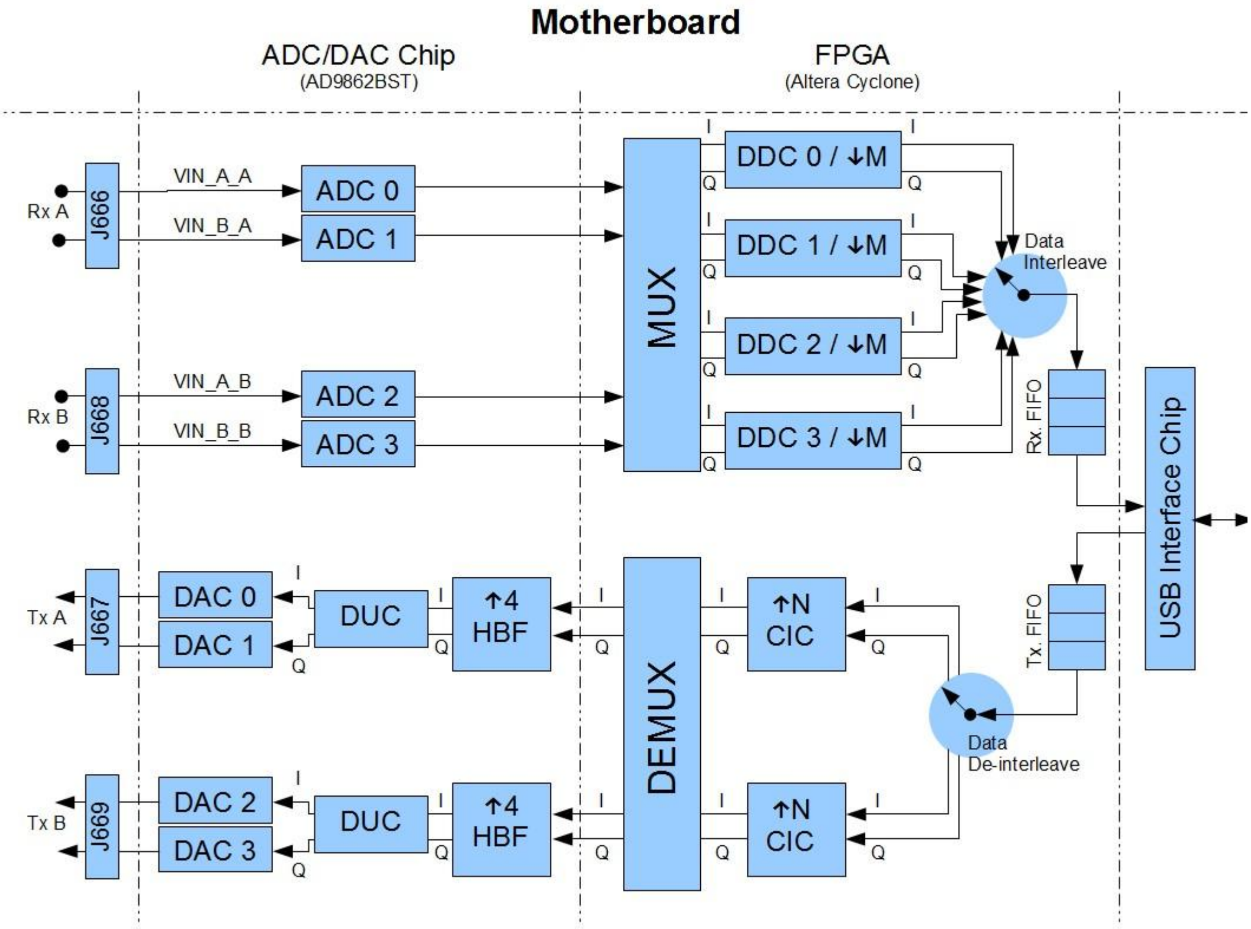


Figure 2.7: Receiver and transmitter data flow within the USRP. Up/down arrows indicate interpolation or decimation of signal in the digital domain by the specified amount.

Figure 2.7 illustrates a simplified flow of data within the USRP mainboard. For the receive section, two single-ended analog inputs form a pair which can be sampled together using quadrature sampling, or sampled individually as two separate inputs using real sampling. Two AD9862 ICs allow for sampling of two quadrature signals or 4 real signals. The received signal is passed to the FPGA in digital form (samples) where it is multiplexed and passed through digital down-converters (DDCs). The samples are interleaved and stored temporarily in a first-in, first-out (FIFO) buffer. From this buffer the samples are retrieved when they can be sent to the host PC for demodulation, filtering, and other processing depending upon the application. The transmit signal path within the USRP is similar except that it operates in reverse. Data from the host PC is stored in a FIFO buffer within the FPGA. It is de-interleaved and then interpolated using cascaded integrator-comb (CIC) filters. From there, it is sent to the AD9862 where it is passed through digital up-converters (DUCs) and then converted to an analog signal by the onboard high-speed DACs.

2.3.2.1 AD9862 Mixed-Signal Front-End

The Analog Devices AD9862 mixed-signal front-end is so called because it provides an interface between the digital and analog signal domain (and vice versa). Two high-speed ADCs and DACs are used for receive and transmit purposes in software radio applications; use of dual ADCs and DACs allows for quadrature sampling and playback to preserve phase information - this is useful in RADAR applications where phase coherence needs to be preserved into the digital domain. The high-speed ADCs operate from internal voltage references or can be configured for external references. The high-speed DACs feature programmable full-scale output current. In addition to high-speed interfaces, the AD9862 provides several auxiliary DACs and ADCs for interfacing of VCOs, programmable gain amplifiers (PGAs), signal level detectors and other devices using a voltage or current for output or control purposes. Receive and transmit signal paths contain PGAs with 0-20 dB gain to make optimal use of ADC and

DAC dynamic range. Receive transmit path contains an input buffer setting the differential input impedance to 200 Ohms which can be bypassed if desired.

Internally, the AD9862 is more than just mixed-signal front-end. It contains several registers which are configured through the use of the SPI bus. These registers configure many functions within the AD9862 such as: 1. SPI Bus interface configuration. 2. Receive side power-down. 3. Transmit side power-down. 4. Internal buffer bypassing. 5. Voltage reference configuration. 6. PGA gain. Additionally, it also performs the task of digital up-conversion. Digital up-conversion (and digital down-conversion) is analogous to that of the function performed by a mixer or transverter; it takes frequency content of a given bandwidth and shifts it up or down within the frequency domain.

2.3.2.2 Altera Cyclone EP1C12 FPGA

The FPGA onboard the USRP serves several functions with the overall purpose of interfacing the AD9862 to the EZ-USB Microcontroller. As shown in Figure 2.7, these roles include: 1. Multiplexing. 2. Data ordering. 3. Interpolation and decimation. 4. Buffering. These roles are performed on the FPGA to alleviate bandwidth requirements and to reduce load on the host system. USB 2.0 specification [20] allows for a maximum data rate of 480 megabits per second (mbps); the minimum decimation factor settable through USRP software is 8 (without using modified/truncated sample depths). This means that, at maximum, the USRP is capable of sustaining 32 megabytes per second (MBPS) to the host system (provided that the host system can sustain the load without dropping data). With 4 onboard ADCs, the USRP is capable of receiving more data than can be transferred into the host for processing. In the case where the IF or baseband bandwidth is very small, as in FM broadcast radio or narrowband communications systems, DDC and decimation filtering allows the digital signal from the AD9862 to be reduced in sample rate and thus requires less USB bandwidth when transferred to the host system. The reverse case is true for transmitting data from the USRP. USB 2.0 specification is half-duplex; data cannot be transmitted from the USRP to the host while the host is transferring data to the USRP.

2.3.2.3 Cypress EZ-USB FX2 Microcontroller

The microcontroller onboard the USRP serves mainly as the USB interface for the FPGA. This microcontroller is intended specifically to negotiate and maintain the high-speed (480 mbps) USB connection with the host PC (full-speed, 12 mbps USB connections are disabled). The FX2 is also responsible for configuration of USRP onboard components (via the SPI and I^2C busses), loading firmware into the FPGA, and monitoring of buffer overruns and underruns.

2.4 Existing Wireless Sensor Interrogators

Wireless sensor interrogation is not a new concept: RFID interrogation is widely employed in a number of applications. RFID interrogators can be readily purchased from commercial vendors which operate within different bands of the RF spectrum and which employ both active and passive sensors. RFID tagging systems typically operate with SAW sensors. Tags can be magnetically coupled or operate using electromagnetic back scattering. In this section, I will discuss: 1. Interrogation techniques. 2. Available interrogation systems including advantages and disadvantages of each. The interrogation systems and techniques outlined in this section serve as a basis for the interrogator design that will be discussed in Chapter 3.

2.4.1 Radio Request Methods and Interrogation Techniques

There are many interrogation techniques in existing literature. Interrogator design is most heavily impacted by radio request technique. Radio requests occur in the frequency domain or the time domain depending upon: 1. Sensor type. 2. Desired characteristics of the interrogator system. 3. Measurand of interest. Analysis of the interrogated signal can be performed using a wide variety of circuit elements and signal processing techniques such as the phase-locked loop (PLL) [21] and the Fourier transform [22].

Typically, time domain request methods operate in a manner similar to pulsed RADAR: the entire bandwidth of interest is interrogated rapidly in a short period of time. Frequency domain request methods operate more slowly, approximating CW, frequency stepped continuous wave (FSCW), or FMCW RADAR. The bandwidth of interest is measured in steps or sweeps with the excitation signal using only a percentage of the overall bandwidth at any given time. Time domain request techniques are suited for measurand which change rapidly with frequency domain techniques reserved for high resolution measurements and improved SNR.

While the interrogator architecture can vary significantly, the purpose of the interrogator is to reveal the measurand. In the figure below, a generic block diagram of an interrogator for use with resonator sensors is shown. The function of the interrogator is essentially the same regardless of architecture provided that resonator sensors are used.

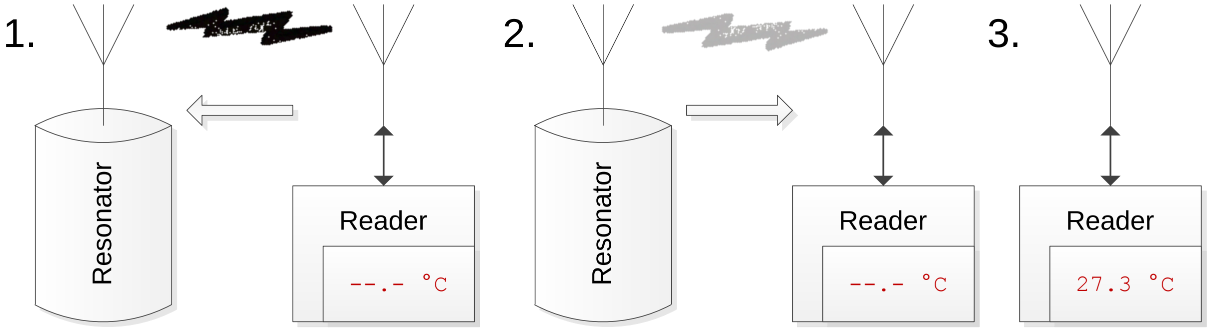


Figure 2.8: A basic diagram of the resonator sensor interrogator. Interrogation is a multi-step process: First the resonator is excited with a pulse from the reader (1). The pulse is turned off after a short time and the reader receives the response (2) which is then translated into a corresponding temperature value (3).

In Figure 2.8, a signal is transmitted to the resonator causing it to store energy. When the signal from the interrogator is switched off, the energy stored in the resonator will be slowly released over some period of time. The energy from the resonator is fed into an antenna causing a response signal to propagate back to the interrogator. This signal is received at the interrogator where it is amplified, filtered, down-converted, converted to frequency domain via a Fourier transform, and processed in other ways to reveal the resonant frequency. The resonant frequency is then converted into a numerical value corresponding to the measurand through the use of a calibrated mapping function.

2.4.1.1 The Fourier Transform and Passive Wireless SAW Sensors

With resonator sensors, the parameter of interest that corresponds to the measurand is the resonant frequency. If this frequency can be ascertained, the resulting frequency can be mapped to the parameter of interest (pressure, temperature, or otherwise). Calibration of the measurement system involves tuning the parameters of the translation function. The Fourier transform allows measurement of the resonant frequency directly. A requirement of the Fourier transform is digitization of the resonator response so that digital signal processing can be applied.

In [22], a time domain interrogator is described which uses a single-conversion superheterodyne receiver to down-convert the response from a SAW sensor. In the system described, an RF module is used to transmit an excitation signal to a SAW sensor and to receive and down-convert the resonator response to a low frequency for digitizing at a low sample rate. The IF bandwidth is sampled by an ADC and passed through a Fourier transform yielding the resonant frequency of the resonator. A controller sets the position of an antenna switch and the output frequency of two PLLs. The first PLL is used to set the frequency of down-conversion. The second PLL is used to generate the excitation waveform. An FPGA is used to interface the ADCs to an embedded computer system for digital signal processing.

The resulting digital signal is digitized and processed by a low-power, embedded computer. The FPGA module is interfaced to the computer and is responsible for controlling the RF components and buffering the data from the digitizers. When a SAW sensor is interrogated, the FPGA triggers the excitation signal, switches the RF module to receive the sensor response, and then sends the computer a signal when the sensor response has been stored. The computer retrieves the waveform, processes it with an FFT, detects the resonance frequency and then computes the physical parameter (in this case, temperature).

In this system, FFT resolution (and hence, resolution of the measurand) is dependent upon the number of measurement points and the sample rate of the measurement. Zero-filling techniques can be used to enhance FFT resolution without affecting the signal-to-noise ratio or the signal energy content [21].

2.4.1.2 Wireless Sensing Using Gated Phase-Locked Loops

Use of PLLs to recover the resonant frequency of a resonator sensor is possible. The gated PLL (GPLL) allows for locking the state of the PLL when the resonator response is not detected. This technique is based on RF burst signals used in the phase alternate loop (PAL) television system: in this system, the RF burst is used to synchronize a local oscillator (LO) which is used for coherent demodulation of color information within the TV signal.

The interrogator design for a GPLL system would likely be based on time domain request techniques. A burst signal is used to excite the resonator sensor. A single antenna is connected via a switch separating receiver and transmitter sections. An RF receiver amplifies and filters the response signal from the resonator sensor which is fed into the GPLL as a reference signal. When the reference signal is not present, a hold mode is enabled causing the GPLL to produce a constant, stable output. The GPLL produces a CW tone which can be measured using two methods: 1. Using a frequency counter. 2. Measurement of the LO control voltage directly with a high quality voltmeter.

Design of this interrogator is likely to be very difficult: PLLs by nature are inherently closed loop systems. For measurement of resonator sensors where the response frequency is heavily affected by the measurand and other random processes, the loop bandwidth must be very wide. According to [21], wideband loops are noisy because of high phase-noise: the fix is to use less bandwidth and search algorithms to lock onto the resonator response. To control the search function of a GPLL, an inexpensive microcontroller could be used. With this approach, use of complex digital signal processing components and techniques can be avoided with the caveat that design of this system would require more intimacy with analog systems and components.

2.4.2 Existing Interrogation Systems

In this section, I will discuss several available interrogation systems with emphasis on the system performance and methods of interrogation. In this section, interrogation systems using delay line sensors and LC resonator sensors will be discussed for comparison to those of resonator sensors.

2.4.2.1 A Passive SAW Based Sensor System for High-Temperature Measurement

Application of passive SAW sensors is particularly suited for high-temperature and other extreme environments where material degradation and failure could occur quickly. In [4], a high-temperature measurement system is designed that can operate between 200˚C and 1000˚C. Reflective delay lines are used to allow interrogation of multiple sensors simultaneously and for tagging purposes. Frequency of operation between 2.4 to 2.4835 GHz allows for worldwide use in the industrial, scientific and medical (ISM) band. This system operates using a FSCW RADAR architecture (frequency domain radio request technique). A PLL is used to generate a stepped frequency sweep which is used as the LO tone and as the excitation signal. The excitation signal is amplified and transmitted to the sensor wirelessly. Use of the transmitted signal as the LO tone for down-conversion means that the RF module is phase coherent. A DSP is used to process data from an ADC which is connected to the RF module. RS232 serial is used to communicate with the DSP. Various power levels are used with a combination of antenna gains and measurement temperatures to evaluate the maximum request distance of the sensor. In this paper, sensor accuracy of better than 0.5 °C is demonstrated for temperatures up to 200 °C. Typical readout distance is approximately 1-5 m depending upon temperature of the sensor.

2.4.2.2 Passive Wireless Sensing Using Inductive Coupling

In [3], a low-cost and reliable readout system for passive LC sensors is designed. It is explained that this system is effectively an impedance analyzer. In the readout system designed, an AC current is generated in a reader coil. This produces magnetic flux in the environment surrounding the coil. Through inductive coupling, a flux will be produced in any nearby coils such as that of the sensor: in this way, the sensor can be powered from the readout unit. In the reader, the resulting coupling to a sensor produces a voltage drop on its coil. [3] explains that this voltage drop is modeled as the reflected impedance of the sensor. As the frequency of the AC current in the reader is swept and approaches the resonant frequency of the sensor, the induced current in the sensor coil increases which in turn causes the reflected impedance to increase proportionally. Through monitoring of the coil impedance, the resonant frequency of the sensor can be determined. Reflected impedance can be measured as magnitude and phase quantities.

CHAPTER 3: INTERROGATOR ARCHITECTURE

In this chapter, I propose new interrogator architecture and describe in detail the logic and motivations for component selection and interfacing. I will start by building a description of the problem at hand. Following this I will explain my proposed solution including the hardware and software for its realization.

3.1 Problem Description

The goal for the research that accompanies this thesis is to develop a wireless passive sensor interrogator that is compatible with resonator sensors. Low-cost resonator sensors are being developed which can operate in harsh environments. Unlike SAW sensors, this novel sensor does not necessarily realize a high QF. Resonators with low QF have high loss; after excitation they resonate only for a short duration. Interrogation of the sensor is further complicated by operation in *X*-band where signal integrity is critical and COTS components are hard to find. Preliminary research used ordinary laboratory equipment to determine the feasibility of resonator interrogator design using RADAR like architecture: this will be discussed further in 5.1.

3.2 Proposed Solution

My proposed solution for the interrogation of a resonator sensor is based on a software defined RADAR approach. It involves three key components: 1. Design of an *X*-band RF front-end. 2. A DAQ for interfacing of the front-end to a PC. 3. Software to analyze data digitized from the front-end.

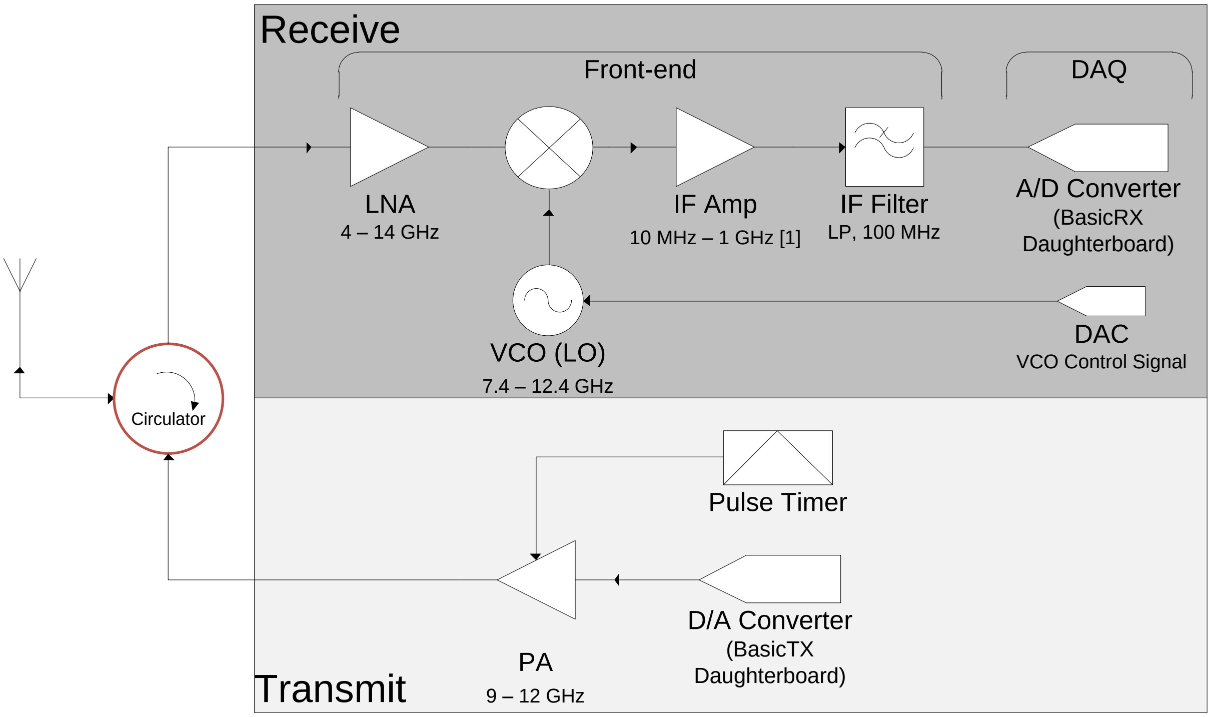


Figure 3.1: Block diagram of proposed temperature sensor interrogator.

In Figure 3.1 I describe in more detail the measurement setup (two of the components I describe in the previous paragraph). The measurement setup can be further broken down into two sub-components; these include: 1. An *X*-band superheterodyne receiver interfaced to a BasicRX daughterboard within the USRP. 2. An *X*-band transmitter interfaced to a BasicTX daughterboard within the USRP. The transmit and receive sections are connected to a circulator such that a single antenna can be used. It is important that it be mentioned here that for the purposes of this thesis, only the receiver is of interest here. The experimental setup used in this thesis describes all used components in detail in sections 5.2 and 5.3.1.

3.2.1 RF Front-End

The front-end of the interrogator consists of the components necessary for the amplification, filtering, tuning, and down-conversion of the signals of interest from the resonator sensor. Amplification is necessary to raise the power level of the received signal from the sensor such that the signal is not lost due to the loss and noise-floor of components in the front-end [23]. Components such as mixers and filters are typically passive devices and so contribute to the loss in the front-end. The most critical amplifier in the front-end is the first amplifier in the signal chain. In radios where sensitivity is critical, the LNA chosen should have the lowest noise figure possible. Frii's formula specifies that system noise figure (NF) is dominated by the performance of the first component within the signal chain.

To minimize the number of components needed, keep costs low, and to potentially simplify design, a single-conversion superheterodyne receiver design was selected for the interrogator. Tuning of the receiver topology used here is accomplished through the use of a VCO.

As illustrated in Figure 3.1, the RF front-end consists of 4 major components: 1. VCO. 2. Mixer. 3. LNA. 4. IF Amplifier. These components will be described in the following subsections.

3.2.1.1 Voltage Controlled Oscillator

The VCO selected for use with the interrogator needs to be of reasonably high quality for it will have an impact on the temperature resolution. Additionally, to eliminate the need for equalizers, amplifiers, or buffers, the VCO should have a reasonable power output level that is flat over the entire tuning range. Given these considerations, the requirements for selection of the front-end VCO include: 1. Low phase noise. 2. Power output. Satisfying these conditions, I selected RFMD's RFVC1800 [24]. This is a wideband VCO which covers a bandwidth of 7.4 to 12.4 GHz, more than sufficient for operation in *X*-band. Measurement results are provided in 5.2.2.

3.2.1.2 Mixer

The mixer selected for use with the interrogator prototype was mainly chosen for low conversion loss and a drive power level that would closely approximate that of the RFVC1800. Low conversion loss is necessary to minimize the system overall NF. The selected mixer is a Mini-Circuits ZX05-153+ [25] which is a broadband mixer that operates to 15 GHz on the LO/RF inputs. The IF output operates down to direct current (DC) which is compatible with the BasicRX daughterboard on the USRP and will work well when if used in a direct-conversion receiver.

3.2.1.3 Low-noise Amplifier

The low-noise amplifier within the receiver is the most critical component in the signal chain when a low NF is desired. This component is the first component in the receiver and is directly attached to the antenna when a wireless measurement system is used; to maximize the range over which wireless sensing will operate. Without the use of additional hardware, this amplifier should have the highest dynamic range possible. A TriQuint TGA2512 [26] was selected for its low noise figure, wideband operation, and quad flat no leads (QFN) packaging. Only a few wideband amplifiers are available which can operate over the full *X*-band, and in my survey all found but this one were not packaged. TriQuint's TGA2600 amplifier was initially chosen for this amplifier but was ultimately abandoned as the amplifier of choice for the interrogator prototype because of difficulties and time constraints that were involving in preparing it for use.

3.2.1.4 IF Amplifier

The IF amplifier in the proposed interrogator design serves the purpose of increasing overall gain of the circuit to overcome free space propagation losses and to account for conversion loss of the ZX05-153+ mixer. For this component, a wideband amplifier was selected to allow for maximum output bandwidth from the receiver into the digitizing device. A Mini-Circuits ZFL-1000VH2X [27] amplifier was selected. I note here that this amplifier was selected before the USRP was decided as the digitizing device so it was not known what the approximate IF bandwidth or center frequency would be.

3.2.2 Data Conversion

Data conversion is required for: 1. Digitizing the IF signal from the analog front-end. 2. Controlling amplifier gain in the TGA2512-2-SM. 3. Tuning the RFVC1800 VCO. Data conversion is performed using: 1. A USRP version 1. 2. Custom selected auxiliary DACs used for gain control and tuning of the receiver VCO.

3.2.2.1 Universal Software Radio Peripheral

The USRP [19] is used in conjunction with the RF front-end for digitizing the resonator response waveform. The USRP used for subsequent experiments in this thesis is fitted with: 1. A BasicRX daughterboard. 2. A BasicTX daughterboard. The USRP will be used on conjunction with the GNU Radio libraries because of its simplicity, COTS availability, and low cost. The use of SDR architecture, especially a flexible one such as the USRP, allows for gauging the feasibility of software and hardware modifications for the eventual realization of a stand-alone wireless resonator sensor interrogator.

3.2.2.2 Auxiliary DACs

Auxiliary DACs on the USRP are available in addition to the main RF inputs and outputs. These DACs are not used on the BasicRX and BasicTX boards and so are made available on jumpers for use with external hardware and other purposes. Aux DACs can be controlled through the GNU Radio software and so would result in the simplest method of controlling the VCO tuning voltage without requiring additional hardware, if possible.

CHAPTER 4: DESIGN, SIMULATION, AND FABRICATION

In this chapter, I will discuss the process of designing the interrogator proposed in Chapter 3. The design process naturally includes simulation and fabrication: these will be discussed here as well. Most relevant to this chapter is an attempt at optimizing a coax to microstrip transition for optimizing the performance of the receiver LNA. I note here the following: 1. This thesis makes use of readily available COTS components as much as possible, thus eliminating fabrication and involved simulation techniques. 2. The LNA component of the proposed receiver architecture was not ready for use at the time of final measurement; this chapter serves as a record of optimization techniques and design strategies which may prove useful for future research and eventual completion of the LNA fabrication process.

4.1 LNA

Some design is required for the TGA2512-2-SM amplifier: a substrate is needed that the QFN packaged amplifier can be mounted to such that SMA connectors can be attached to the amplifier input and output. To maintain low NF, careful attention to signal integrity is imperative. While resistive loss is not an issue with such a small board, proper impedance matching is important. This seems especially true with regards to the SMA connectors which present a coax to microstrip transition. I discovered during early testing of the TGA2512 amplifier that the coax to microstrip transition is critical in the overall system design: a poor match at the input to the amplifier will result in a poor overall system noise figure. The procedure that I used to optimize this transition is discussed in detail in section 4.1.1.

Initial testing of the TGA2512 amplifier yielded very low gain from the amplifier. It is possible to verify the amplifier is operating correctly through monitoring of the amplifier's drain and gate currents, so I suspect that the amplifier was operating normally. Knowing this, I removed the amplifier from the circuit and created a thru board. The first version of this board contains two SMA connectors at either side of a piece of Rogers RO4003C substrate and 50 Ohm microstrip connecting the SMA connectors together. This board was connected to a VNA to obtain S_{11} and S_{21} parameters. Measurement results are shown in Figure 4.1. Layouts of the thru printed circuit boards (PCBs) used for optimization of the SMA connectors are given in Appendix C.

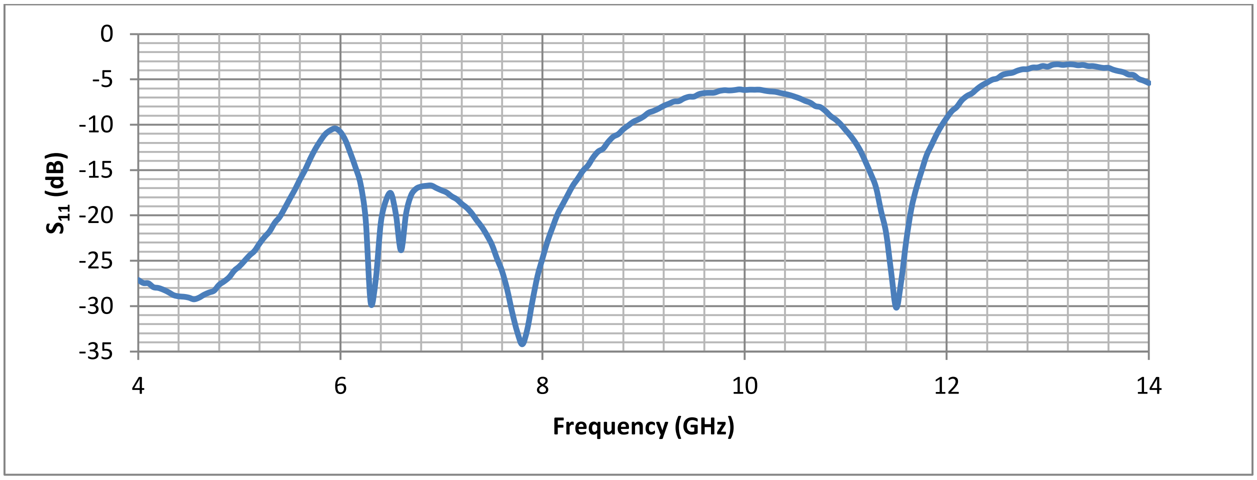


Figure 4.1: Thru board version 1 S_{11} parameters as measured.

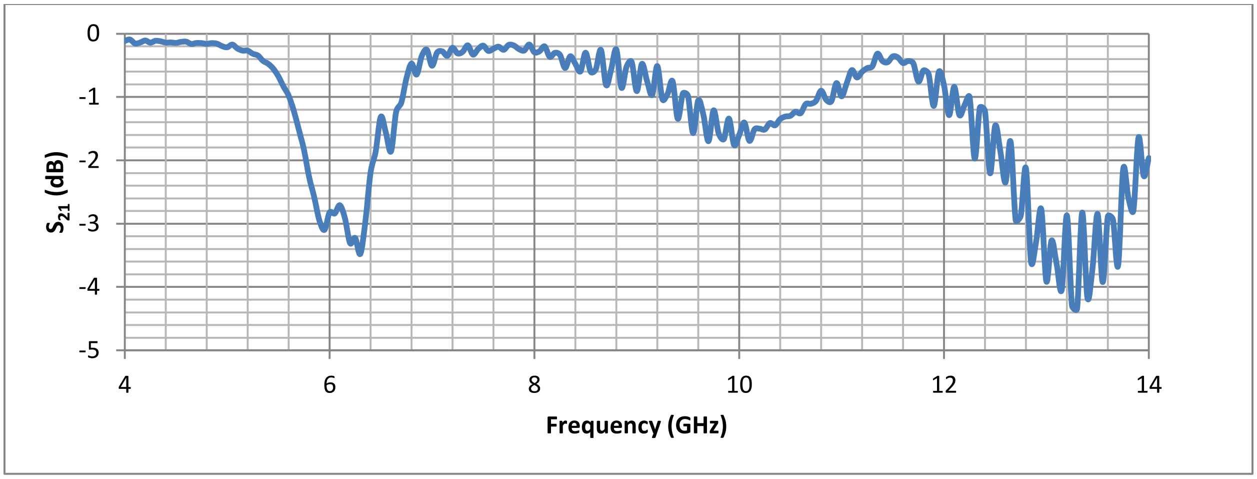


Figure 4.2: Thru board version 1 S_{21} parameters as measured.

From Figure 4.1 and Figure 4.2 it is apparent that at higher frequencies of operation, 50 Ohm microstrip does not match well with the SMA connectors. At 13 GHz, return loss is nearly 3 dB which corresponds to a voltage standing wave ratio (VSWR) of roughly 5.8. Insertion loss appears to be acceptable despite the bad match, at least between 7 to 12 GHz. In the next section, I will explain steps taken to rectify this in attempt to obtain reasonable amplifier performance.

4.1.1 Optimizing the Coax to Microstrip Transition

Optimizing the coax to microstrip transition is mainly intended to improve the match between a receiving antenna and the amplifier input. The optimization process that I followed is based upon two approaches: 1. Optimization of the SMA connector land pattern based upon recommendation of a data sheet. 2. Ansoft High Frequency Structure Simulator (HFSS) techniques.

4.1.1.1 Data sheet approach

Connector matching is partially dependent upon the land pattern and any transitions used into the microstrip that the connector is mated to. The datasheet for the connectors used on the thru boards and the TGA2512 amplifier board (or a very similar connector; part number 142-0701-881) suggests an optimal land pattern for mating this connector to 50 Ohm microstrip [28]. It is noted here that this land pattern is for FR4 substrate: the TGA2512 amplifier board and the thru boards used for optimization testing are fabricated using Rogers RO4003C substrate. To take into account the change in substrate, the "optimal" microstrip impedance was determined using the datasheet parameters and FR4 substrate values before the new layout was designed. A modified thru board was fabricated and measured and the results given in Figure 4.3 and Figure 4.4.

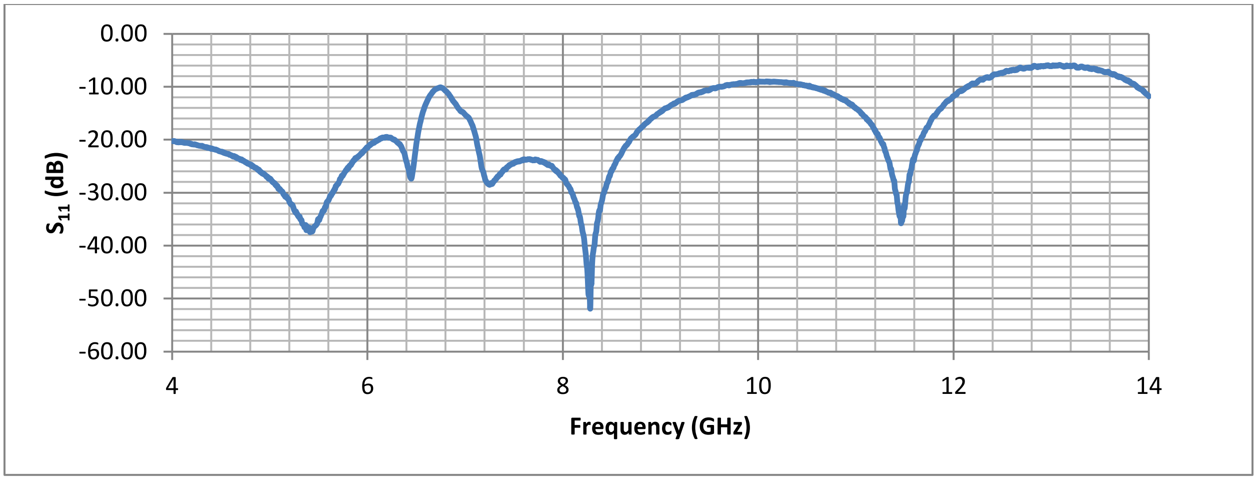


Figure 4.3: Modified thru board S_{11} parameters as measured.

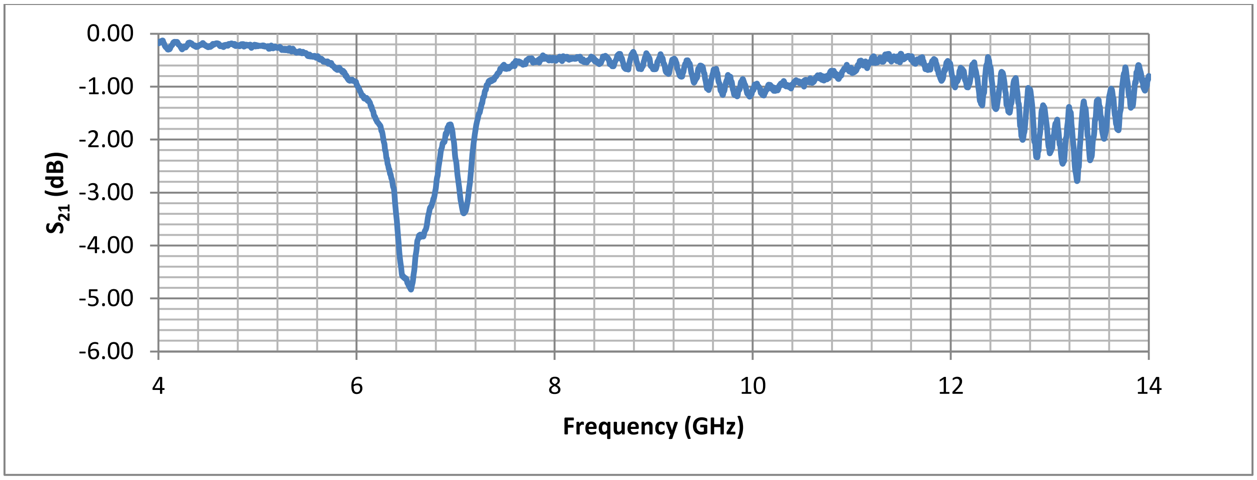


Figure 4.4: Modified thru board S_{21} parameters as measured.

4.1.1.2 Modeling the SMA to Microstrip Transition Using Ansoft HFSS

The second approach to optimizing the SMA to microstrip transition uses a simulation technique. An SMA connector model was drawn in Ansoft HFSS which represents the thru board and the SMA connector used in the TGA2512 amplifier board. I note here that the key to successful modeling of the connector within HFSS is the accurate modeling of the dimensions of the center pin. Center pin drawings are typically not available within the datasheet or on the manufacturer's website. To attain accurate dimensions of the center pin in the SMA connector, a cross-sectional cut was made to the connector using a rotary tool. The resulting center pin features can be seen in Figure 4.5. The SMA connector cross section drawing as modeled in HFSS is shown in Figure 4.6 and the thru board model is shown in Figure 4.7.

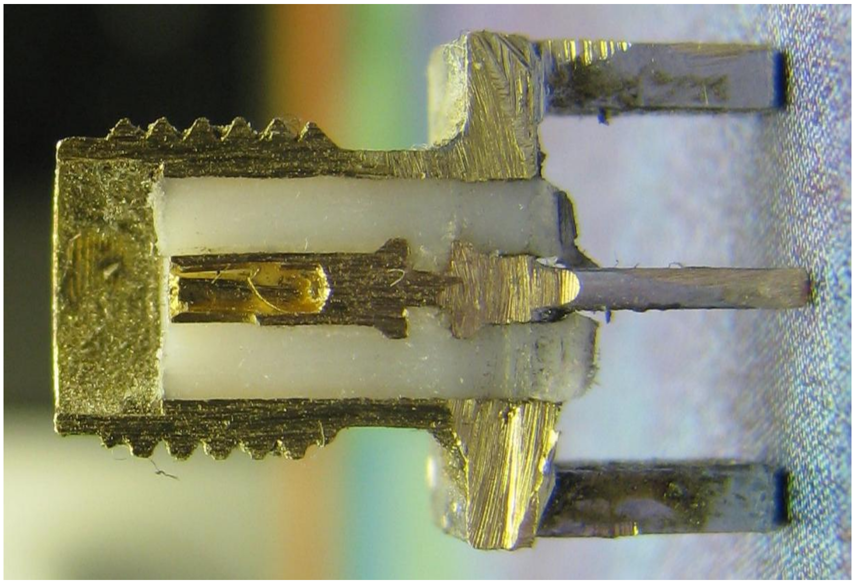


Figure 4.5: SMA connector cross-section showing non-uniform center pin width within connector dielectric.

Parametric sweeps were used to adjust various dimension parameters of the model including the microstrip width ground pad widths. Most dimensions of the connector had little or insignificant impact on the connector performance, even at high frequencies. The dimensions of the microstrip width and center pin have the most effect, as might be expected.

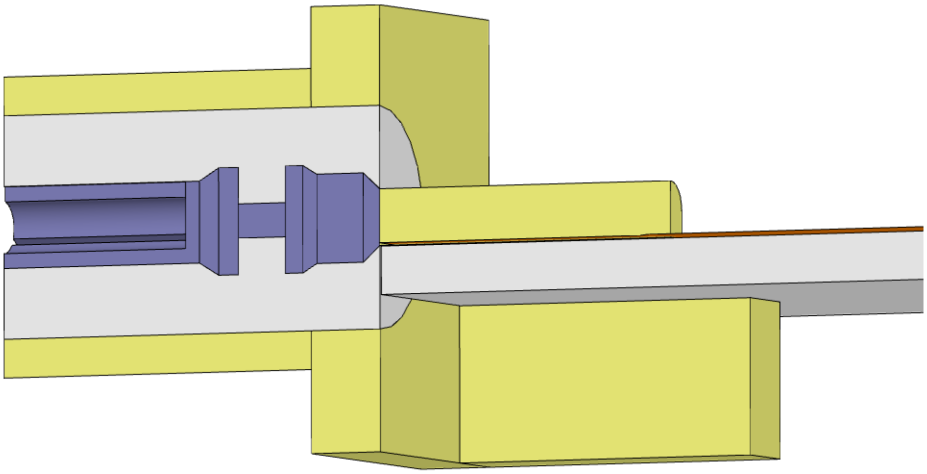


Figure 4.6: HFSS simulation of SMA connector.

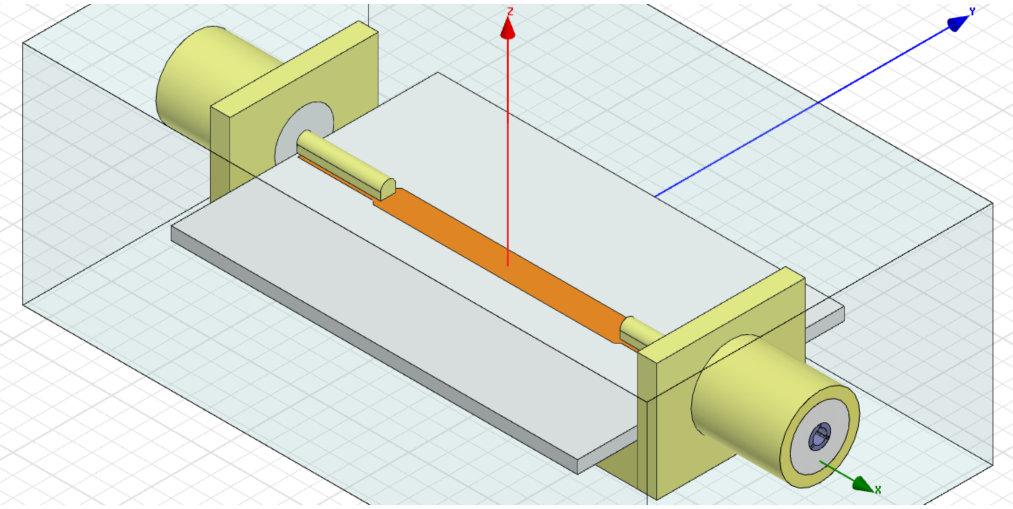


Figure 4.7: HFSS simulation of SMA thru board with top connector stubs removed.

As can be seen in Figure 4.7, the SMA stubs to the left and right of the center pin have been removed above the substrate. Comparison between the simulation results and the previously shown measured data suggests a more smooth response across the 4 to 14 GHz bandwidth. Simulations and measured data suggest that the top stubs of the SMA connector were causing a resonance around 6 GHz which appears to have been eliminated in the results below.

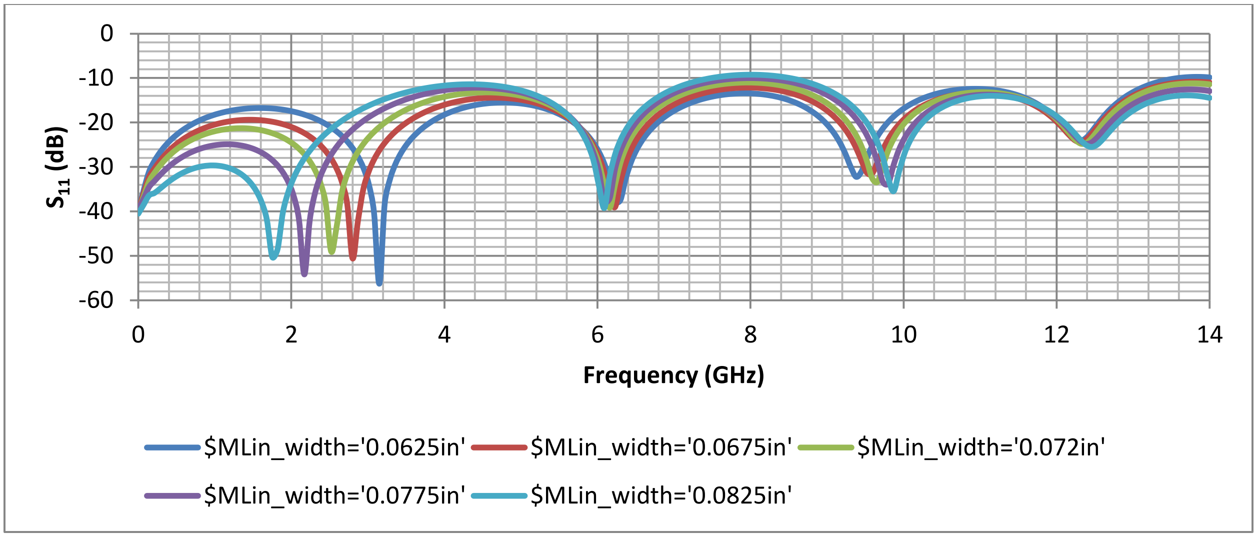


Figure 4.8: Final S_{11} simulation results from Ansoft HFSS.

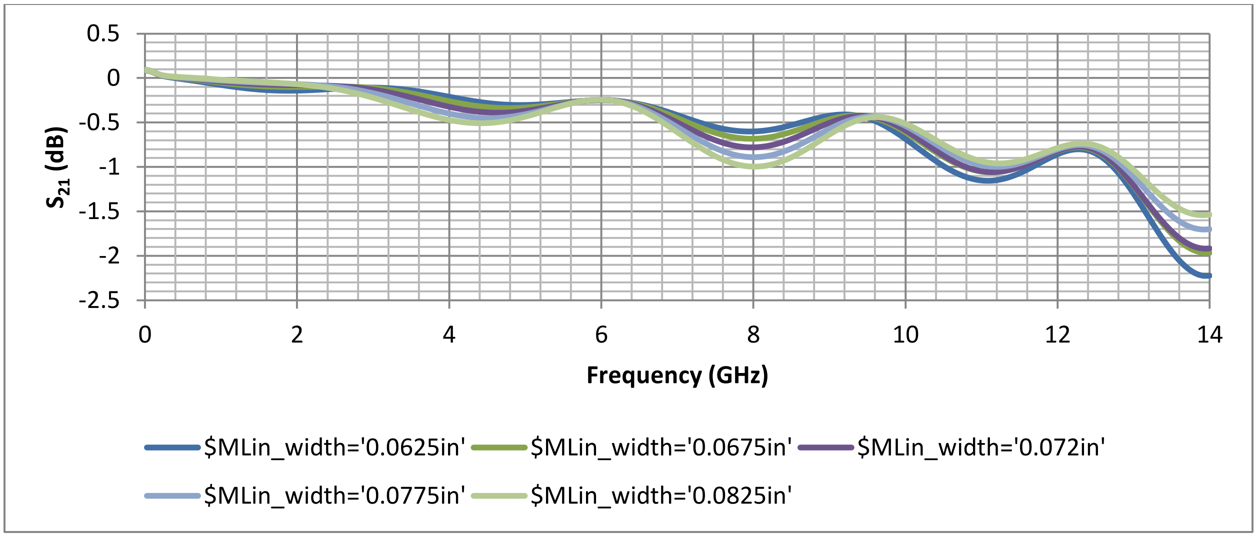


Figure 4.9: Final S_{21} simulation results from Ansoft HFSS.

CHAPTER 5: MEASUREMENTS AND PERFORMANCE

Measurement data collected during design, fabrication, and simulation of the resonator interrogator prototype is provided in this chapter (with the exception of the version 1 thru board measurement which is presented in Chapter 4 for clarity). I organize this chapter into sections ordered as follows: 1. Preliminary research. 2. Front-end performance and measurements. 3. Interrogator performance and results.

5.1 Preliminary Research

To learn the basic principles of resonator sensor interrogation, preliminary research was conducted using ordinary laboratory equipment that operates similar to that of conventional RADAR: 1. A vector network analyzer (VNA) operates similar to FMCW RADAR. 2. An oscilloscope and function generator capable of low duty cycle pulse modulation of CW carrier can be used to simulate pulsed RADAR. Both uses of laboratory equipment allow for experimentation with active RADAR configurations necessary for the excitation of the passive resonator sensor. To emulate the resonator sensor, a waveguide cavity resonator was used (and is discussed further in 5.3.1.2). This type of resonator has a very high QF compared to some resonator sensors and has fixed resonance modes.

Several trials were performed, each time varying the test configuration; the goal was to view the resonator response on either the VNA or the oscilloscope. The most reliable technique for interrogating the resonator is depicted by Figure 5.1 and was the first of the experimental trials. This method of interrogation does not use antennas so the resonator response will exhibit very little loss. We expect to easily view a response from the resonator on the display of the oscilloscope.

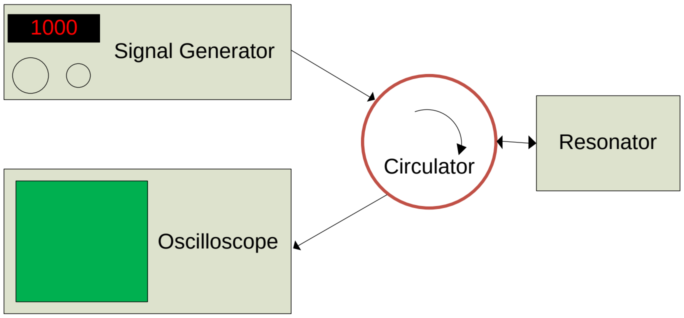


Figure 5.1: First preliminary experimental setup.

This measurement setup is reliable for several reasons: 1. A wireless link which will exhibit high levels of free-space attenuation is not used. 2. The oscilloscope has a very high sample rate (50 GS/s) and a wide analog bandwidth (16 GHz). Oscilloscopes in general do not have low NFs without the use of an external LNA, meaning attenuation should be kept to a minimum. The high sample rate of the oscilloscope means that the short duration of the resonator response waveform will be sampled to a sufficient number of points to yield an accurate Fourier transform (one with a high resolution). Figure 5.2 shows the modulated waveform of a cavity resonator. The yellow trace is the time domain waveform and the orange trace shows the corresponding frequency domain plot of the data gated between the yellow markers.

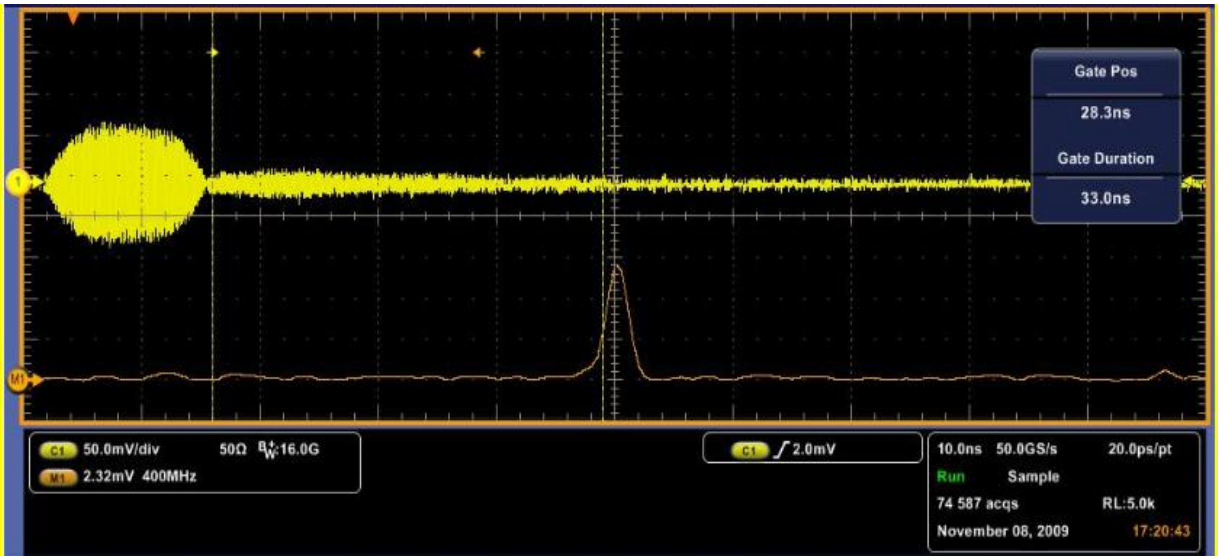


Figure 5.2: Resonator response waveform from the first preliminary experiment setup. Resonator ringing occurs between the yellow FFT gating markers, following the excitation pulse.

The third preliminary experimental setup (the second setup described here) extends on the measurement setup described in Figure 5.1: for this case the cavity resonator is interrogated wirelessly. A direct coaxial connection to the cavity resonator has been replaced with a wireless connection through the use of two patch antennas of similar characteristics.

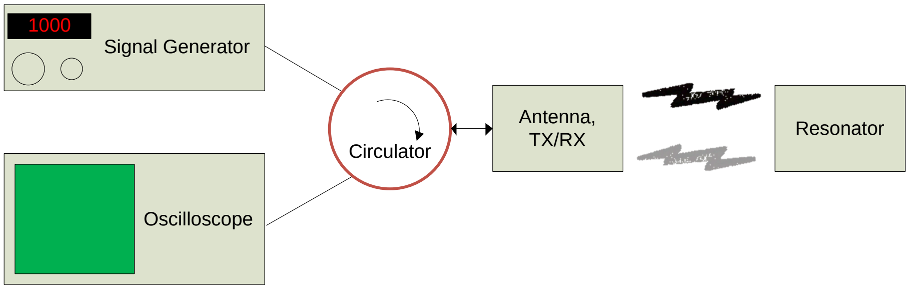


Figure 5.3: Third preliminary experimental setup.

This experimental setup was found to function as expected, but only when the interrogation antenna and the resonator antenna are very closely spaced. Ultimately this technique was able to measure the resonance frequency of the cavity resonator at a distance of approximately 1 – 2 cm.

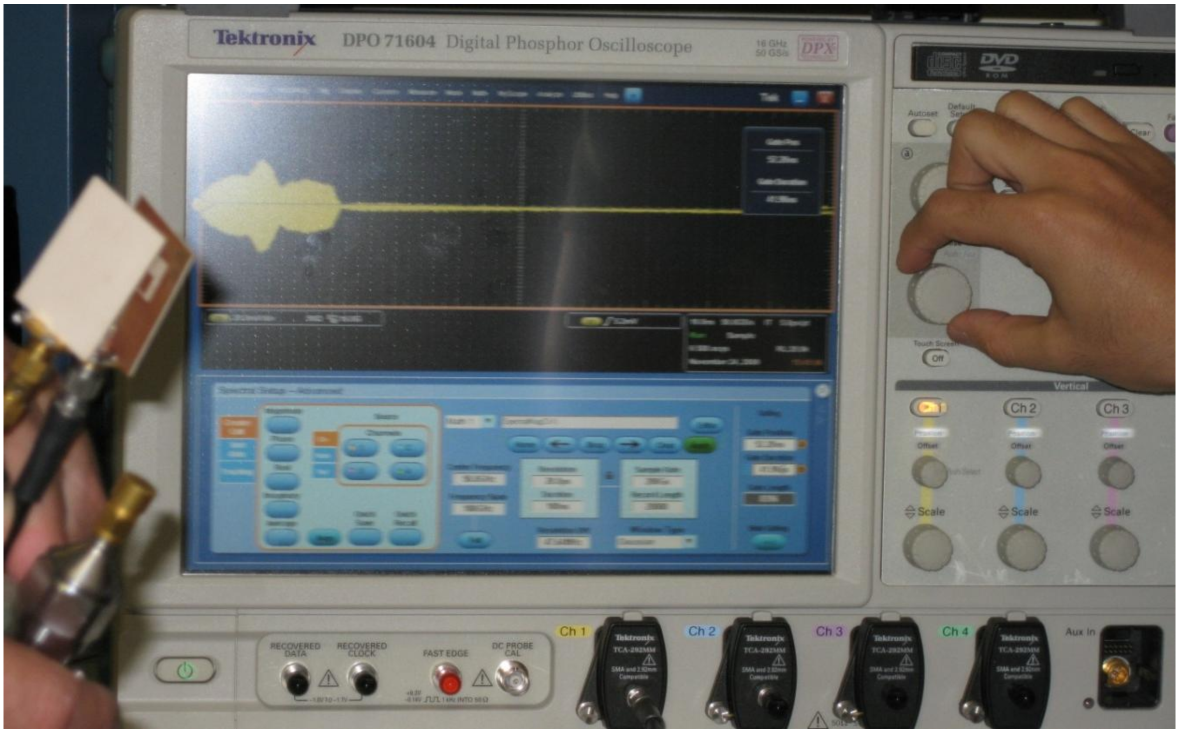


Figure 5.4: Resonator response after excitation using a pulse modulated CW tone at 10.64 GHz and 10 dBm power level. A resonator is excited through patch antennas and a circulator is used to view only the resonator response on the oscilloscope.

5.2 Front-End Performance and Measurements

Characterization of front-end performance is necessary to determine the overall system noise-figure. I provide here some measurements and notes regarding the various components used in the experimental setup which I will discuss in 5.3.

5.2.1 LNA

Measurements for the performance of the TGA2512-2-SM board as described elsewhere in this thesis are not available. I note here that this amplifier was ultimately not used in the final prototype receiver at the time of completion of this thesis.

5.2.2 VCO

Measurements for the RFVC1800 VCO are provided here. In Figure 5.5 the VCO output power is plotted against tuning voltage. Figure 5.6 illustrates VCO output frequency versus tuning voltage. For both cases, an Agilent E4448A spectrum analyzer is used to obtain the measurement data. Measured output power of this VCO has been corrected for cable losses in the measurement setup.

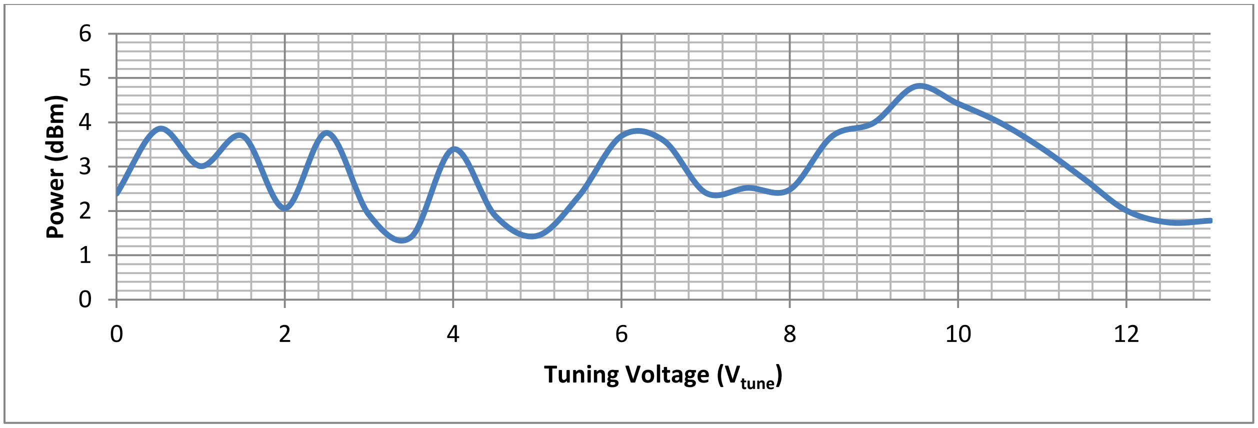


Figure 5.5: RFVC1800 output power versus tuning voltage (V_{tune}).

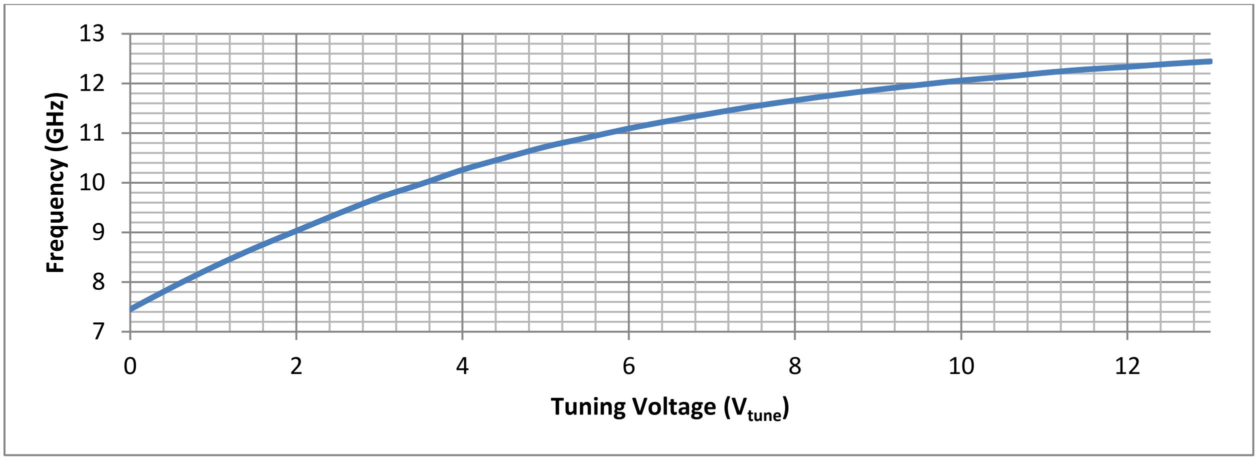


Figure 5.6: RFVC1800 output frequency versus tuning voltage (V_{tune}).

Figure 5.6 illustrates that the output frequency of the VCO is a nonlinear function of the tuning voltage. This can be corrected using software if the VCO tuning voltage is to be supplied by a DAC. It is worth mentioning that this curve may be temperature sensitive which can result in temperature related calibration issues. A 6th order curve fit was used in Excel to generate a Polynomial equation for translating V_{tune} to a corresponding output frequency. The coefficients of the equation with form $y=ax^6+bx^5+cx^4+dx^3+ex^2+fx+g$ are given in Table 5.1 below.

Table 5.1: Curve fit coefficients for the V_{tune} to VCO output frequency equation.

Coefficient:	**Value:**
a	2.6336891334466399E-7
b	-2.4018139626491598E-5
c	0.000601056244704523016
d	-0.00462097401214122978
e	-0.034222574628074702

f	0.87682987556036096
g	7.4552554259516901

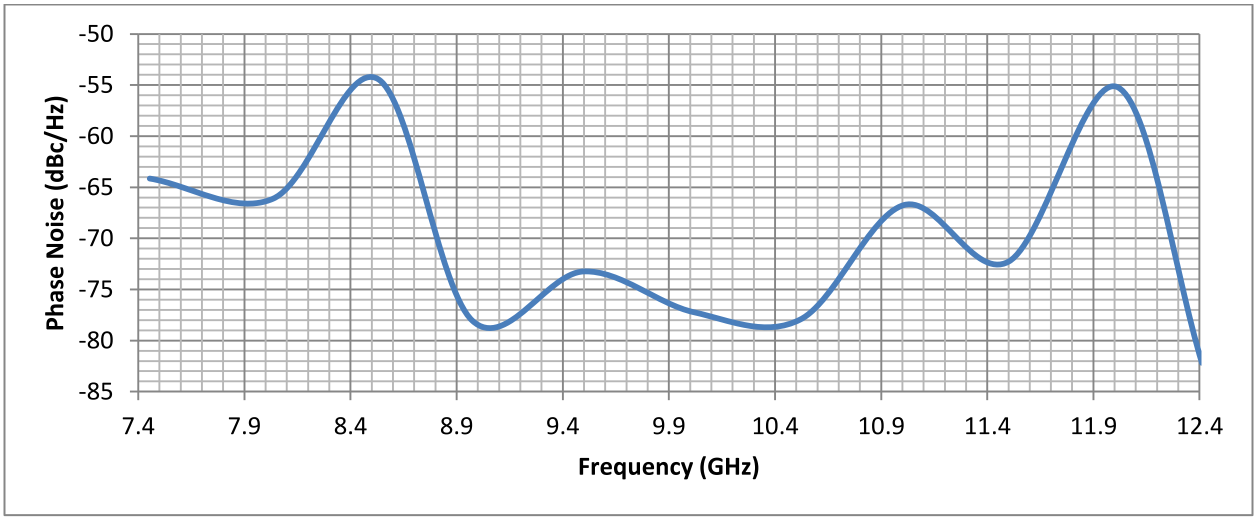


Figure 5.7: RFVC1800 phase noise at 10 kHz offset.

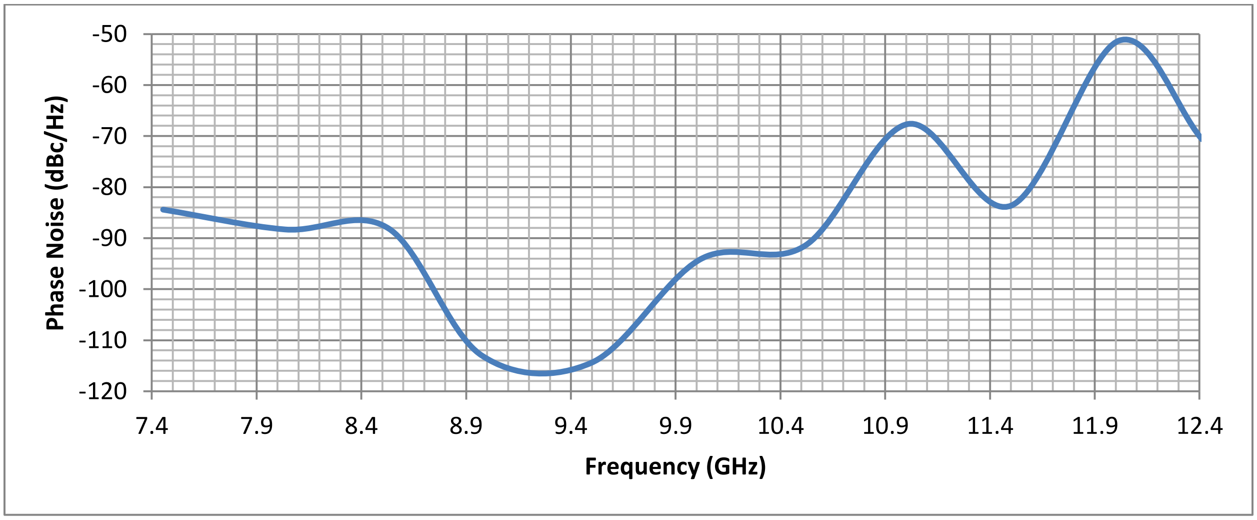


Figure 5.8: RFVC1800 phase noise at 100 kHz offset.

5.2.3 Mixer

Mixer conversion loss plays an important role in system noise figure. Measurements were made using: 1. Agilent E8257D signal generator. 2. Agilent E4448A spectrum analyzer. 3. SMA cables of approximately 3 ft. length. Loss of the measurement cables was determined by connecting each cable used between the signal generator and the spectrum analyzer and recording the difference in power between each instrument.

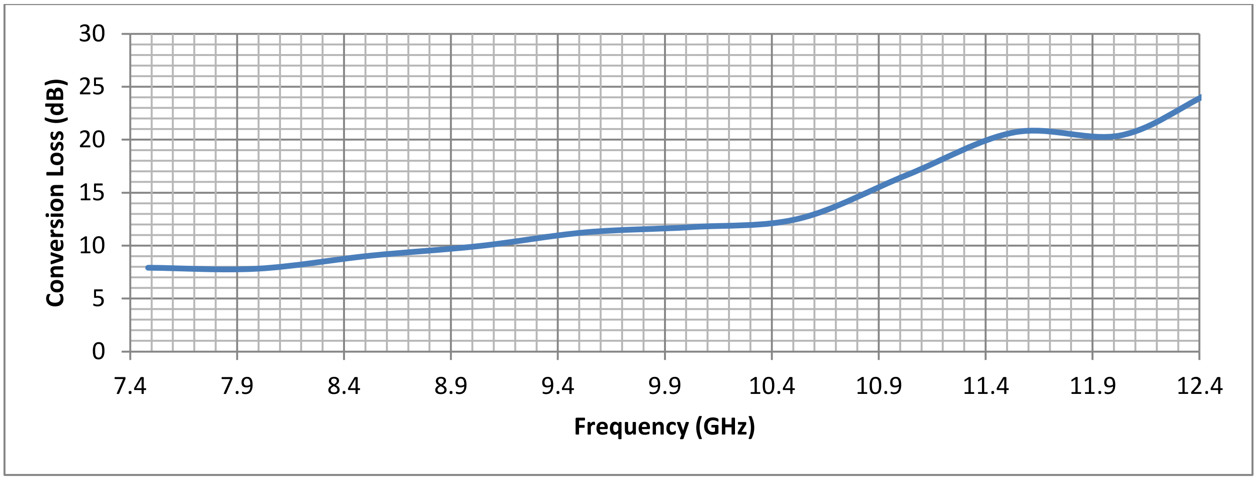


Figure 5.9: Mini-Circuits ZX05-153+ conversion loss as measured.

Conversion loss for this mixer is higher than what could be expected from the datasheet. This is due mainly to driving the LO port of the mixer with a slightly lower input power than the mixer is rated for. To improve conversion loss, increasing the LO drive power is the most effective option, aside from sourcing a more efficient mixer.

5.2.4 IF Amplifier

A Mini-Circuits ZFL-1000VH2X amplifier is used as the IF amplifier for the interrogator prototype. An Agilent N5230A VNA was used for the measurement. A 12 V, 7 AH gel-cell is used as the power supply for the amplifier. Measurement results were saved after the power had been connected to the amplifier for a period of at least 5 minutes to allow the battery voltage and amplifier operating temperature to stabilize. Battery voltage was measured to be 12.781 VDC when the measurement results were saved. To protect the VNA from excessive power levels, an HP 50 dB coaxial attenuator was connected between the output of the amplifier and port 2 of the VNA. A full 2-port calibration was made over the measurement bandwidth prior to connecting the amplifier to the VNA. The measurement results can be seen in Figure 5.10.

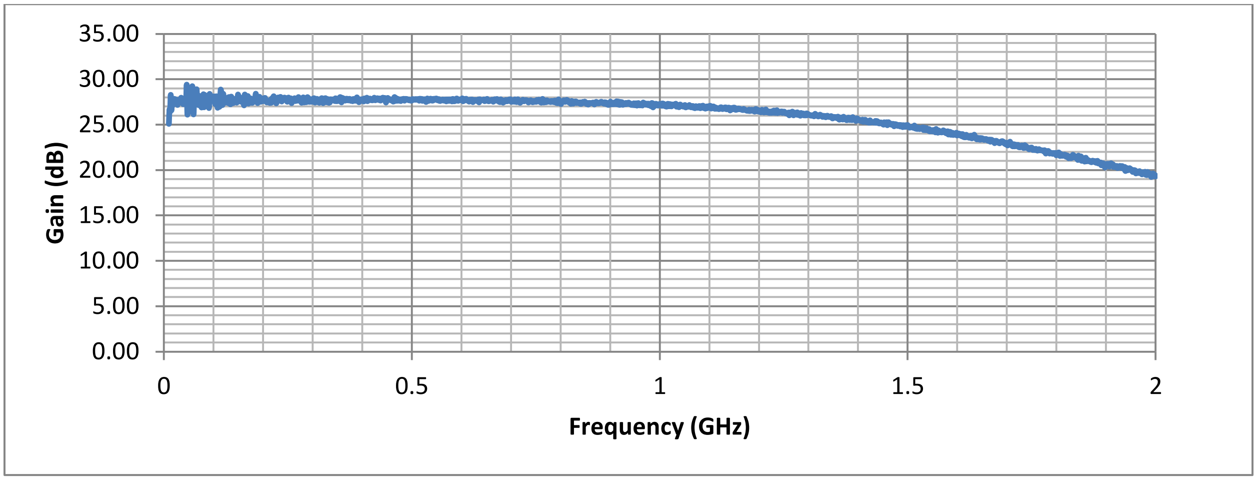


Figure 5.10: Mini-Circuits ZFL-1000VH2X amplifier gain as measured with a 12 V supply.

5.2.5 IF Filter

A Mini-Circuits BLP-100+ low-pass filter serves as the IF filter for the interrogator prototype. This is an inline, coaxial packaged filter with BNC connectors. Measurements of the filter response were made using an Agilent N5230A VNA. A full 2-port calibration was performed prior to measurement. Measurement results are shown in Figure 5.11 and Figure 5.12.

-

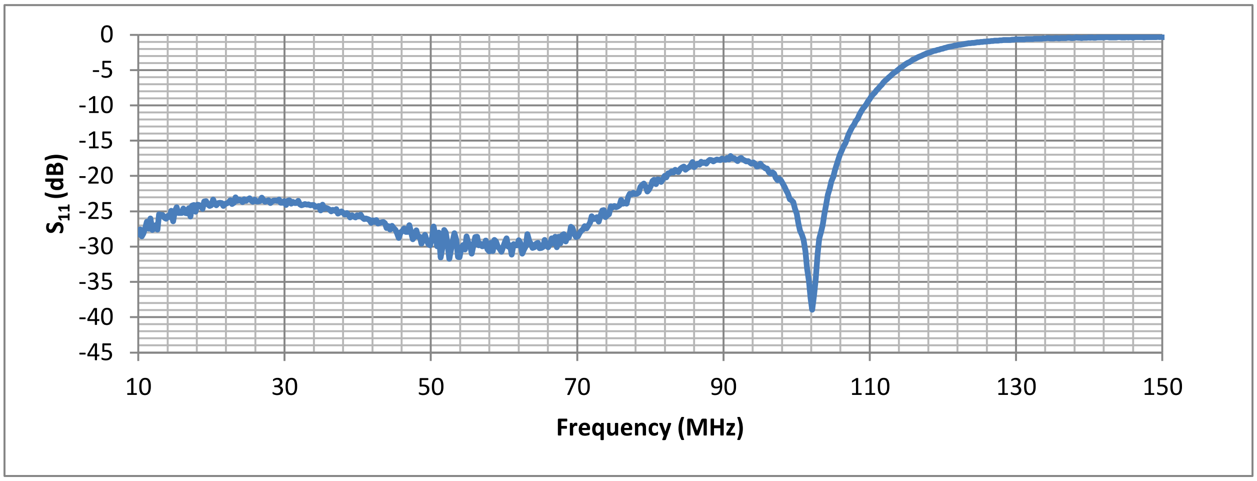


Figure 5.11: Mini-Circuits BLP-100+ low-pass filter S_{11} parameters as measured.

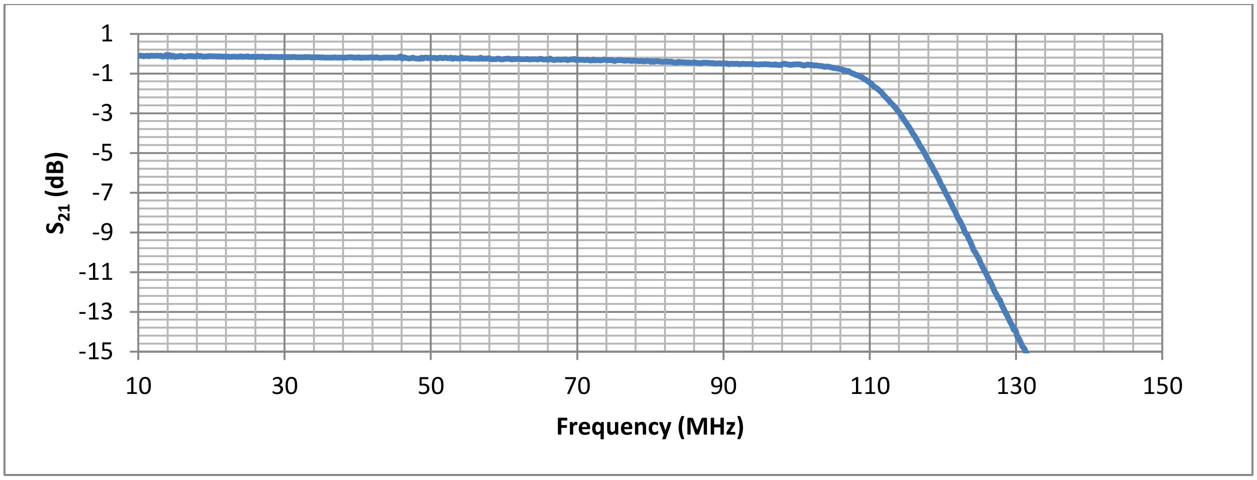


Figure 5.12: Mini-Circuits BLP-100+ low-pass filter S_{21} parameters as measured.

5.3 Interrogator Performance and Results

As there is no clearly defined, objective method for gauging interrogator performance, I characterize interrogator performance by collecting a series of waveforms and performing FFT for various gating positions. The most fundamental purpose of the interrogator is to measure the resonant frequency of the cavity resonator, thus, if this cannot be calculated from the data collected, it should be clear to the reader that the interrogator is not performing the intended function. I will: 1. Explain the experiment setup in detail. 2. Present the waveforms as collected from the USRP. 3. Post-process the collected waveforms. 4. Discuss the results from 2 and 3.

5.3.1 Experiment Setup

To test interrogator performance, the receiver front-end was connected to the USRP, as in Figure 5.13. An Agilent E8257D signal generator was configured to produce a pulsed output tone at 10.616 GHz and approximately 0 dBm output power. Pulse duration is typically 60 ns or greater (and is specified with the data collected) with a pulse period of 1 ms. A Ditom D3C8012 circulator is used as a duplexer for the cavity resonator, allowing the signal generator to excite the resonator and not the receiver. The RF signal generated by the cavity resonator is down converted with the Mini-Circuits ZX05-153+ coaxial mixer, which is fed with a LO tone from the RFMD RFVC1800. The IF output from the mixer is passed through a Mini-Circuits BLP-100+ low pass filter and then the IF amplifier before it is digitized by the USRP. A 12 VDC, 7 AH gel-cell is used to provide power for the RFVC1800 and the Mini-Circuits ZFL-1000VH2X IF amplifier: The IF amplifier is powered directly from the battery while an LM7805 regulator provides the +5 VDC for the VCO. An Agilent E3631A variable power supply is used to provide the tuning voltage for the VCO. The setup as used in the measurements to follow is connected as shown in Figure 5.13. A block diagram of the experiment setup is given in Figure 5.14.

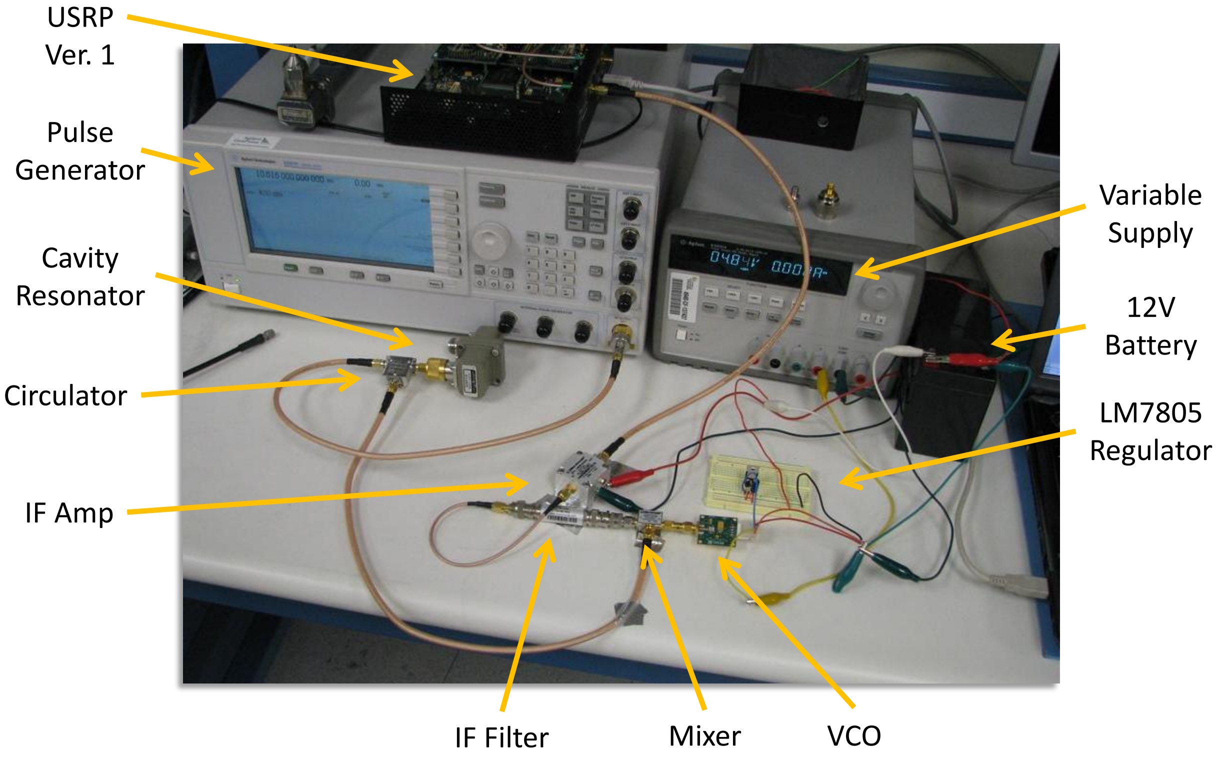


Figure 5.13: Assembled receiver as connected to the USRP and cavity resonator.

5.3.1.1 USRP Configuration and Operation

With regards to the USRP, a custom FPGA firmware was compiled and used. This firmware was compiled with the receiver decimation stages and numerically controlled oscillator (NCO) stages disabled. This results in a very high sample rate from the USRP into the host PC (higher than maximum possible sample rate into the host using 8 bit samples). The process of compiling and configuration of the GNU Radio source code to generate this custom firmware is outlined in Appendix B. With this custom firmware, a sample rate of approximately 21.33 MHz is achieved into the host PC using standard 16 bit samples. With 8 bit samples used a sample rate of 32 MHz can be achieved, however some loss of dynamic range is expected due to the method of lowering the quantization level.

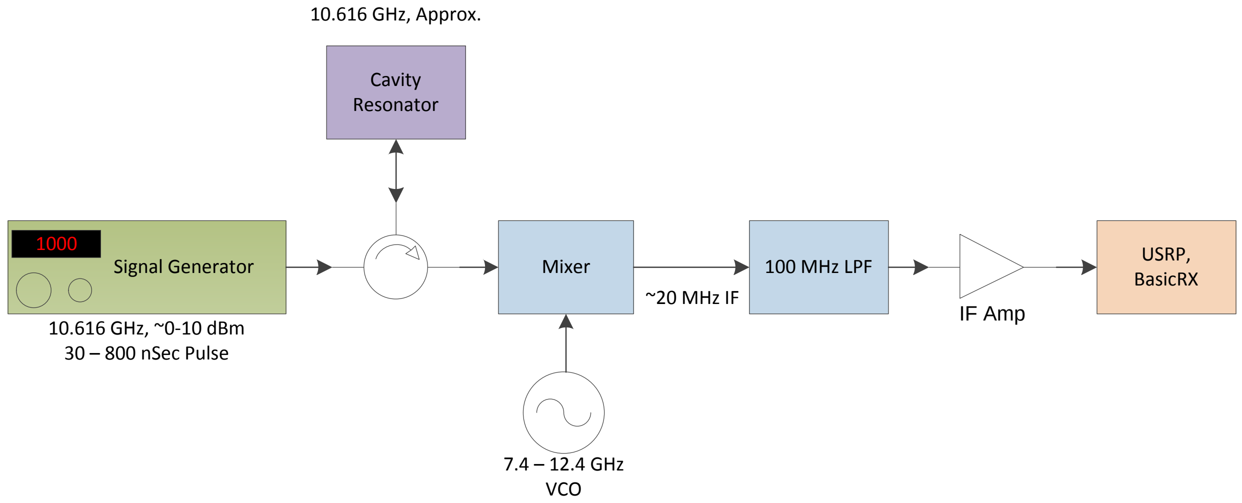
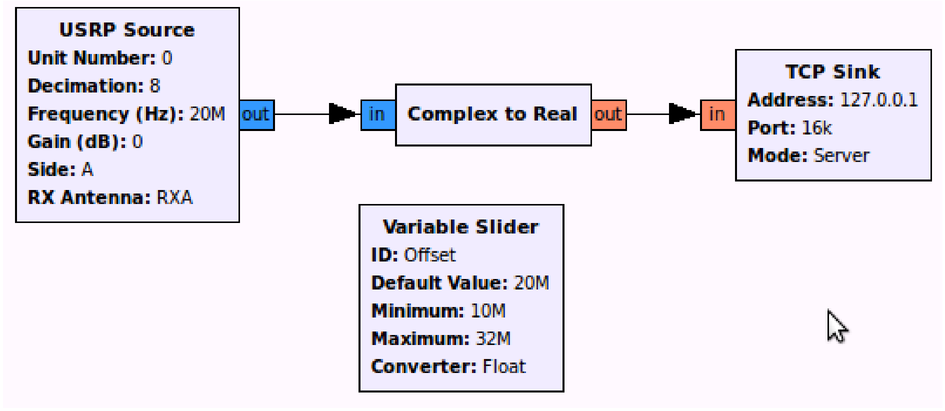


Figure 5.14: Experimental setup used to acquire modulated waveforms from the cavity resonator.

Data is collected from the USRP by interfacing the GNU Radio USRP sink to a LabVIEW virtual instrument (VI) through the use of a transmission control protocol (TCP) client and server. The GNU Radio signal flow graph was generated using GRC software and is shown in Figure 5.15.



USRP Source
Unit Number: 0
Decimation: 8
Frequency (Hz): 20M
Gain (dB): 0
Side: A
RX Antenna: RXA

Complex to Real

TCP Sink
Address: 127.0.0.1
Port: 16k
Mode: Server

Variable Slider
ID: Offset
Default Value: 20M
Minimum: 10M
Maximum: 32M
Converter: Float

Figure 5.15: GRC flow graph used to interface the USRP to a LabVIEW VI.

5.3.1.2 Cavity Resonator Measurements

To show more clearly to the reader of this thesis that the proposed interrogator design is effective in measuring the resonant frequency of a wireless passive sensor, I provide VNA measurements of the cavity resonator used in the experimental setup. The cavity resonator consists of two HP X281A WR90 waveguide adapters coupled together. This cavity resonator has a resonant mode at approximately 10.6145 GHz and a QF of approximately 3000 (QF has not been measured for this thesis).

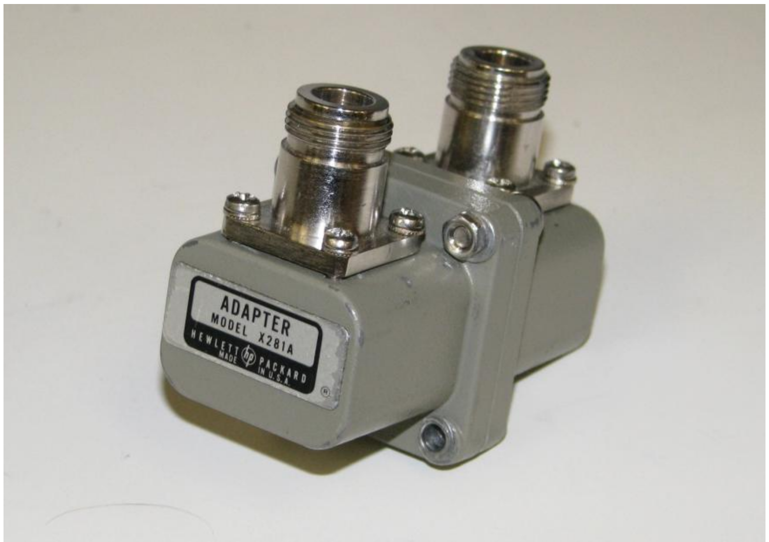


Figure 5.16: A cavity resonator using two HP X281A waveguide adapters.

A 1-port calibration was used prior to measurement of the cavity resonator over a bandwidth of 10.55 – 10.67 GHz. A time domain transform was used on the VNA for gating the S_{11} measurement such that only the resonator modulated waveform is given in Figure 5.17.

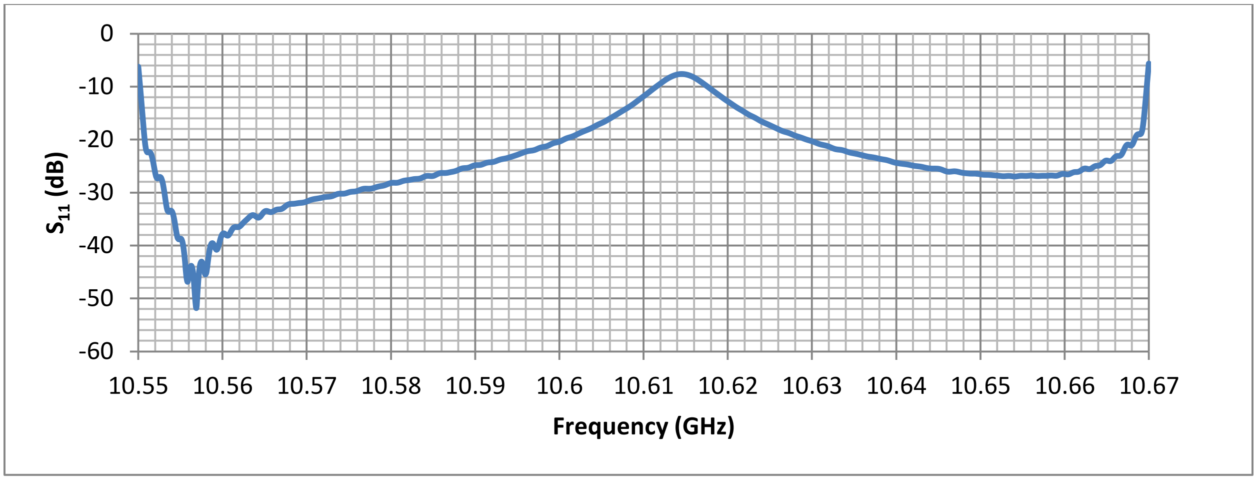


Figure 5.17: Time gated S_{11} magnitude of cavity resonator from Figure 5.16. Peak indicates a resonant mode at 10.614 GHz.

5.3.2 Results

I present the results of the experimental setup previously described in the following two sections. I will begin by presenting the waveforms as captured in the time domain. Following this, I will demonstrate post-processing of the data through an FFT algorithm in LabVIEW.

5.3.2.1 Receiver IF Output

To verify the IF output of the RF front-end, the circulator output containing the cavity resonator modulated waveform was connected to a Tektronix DPO 71604 oscilloscope with a 10 dB *X*-band waveguide coupler through the coupled port; the transmitted port of the coupler was connected to the mixer for down-conversion to IF. An example of a down-converted waveform as modulated from the cavity resonator is shown in Figure 5.18. The blue trace is the modulated cavity resonator waveform before down-conversion. The yellow trace is the down-converted signal at an IF frequency of approximately 90 MHz (for illustration purposes). The orange trace is the magnitude frequency spectrum of the IF output. Gating of the magnitude spectrum is such that the amplitude of the reflected resonator excitation pulse appears to have amplitude similar to that of the resonator ringing for purposes of illustration. The cavity excitation pulse waveform duration is approximately 30 ns as set on the signal generator.

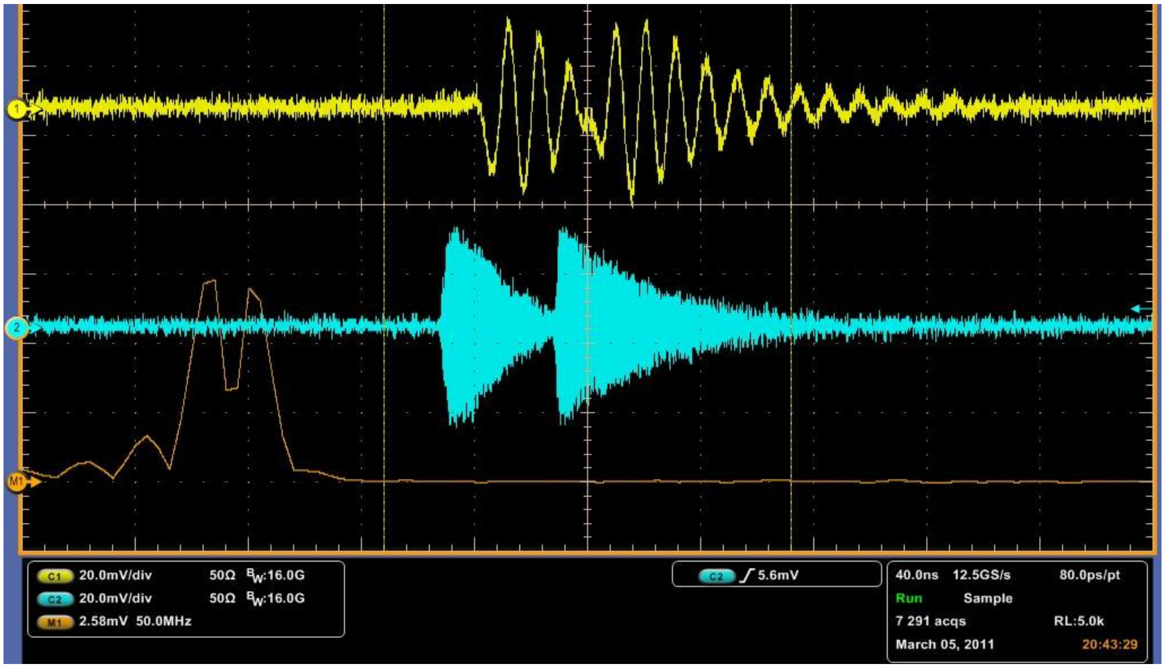


Figure 5.18: Interrogator prototype IF output in time domain and frequency domain.

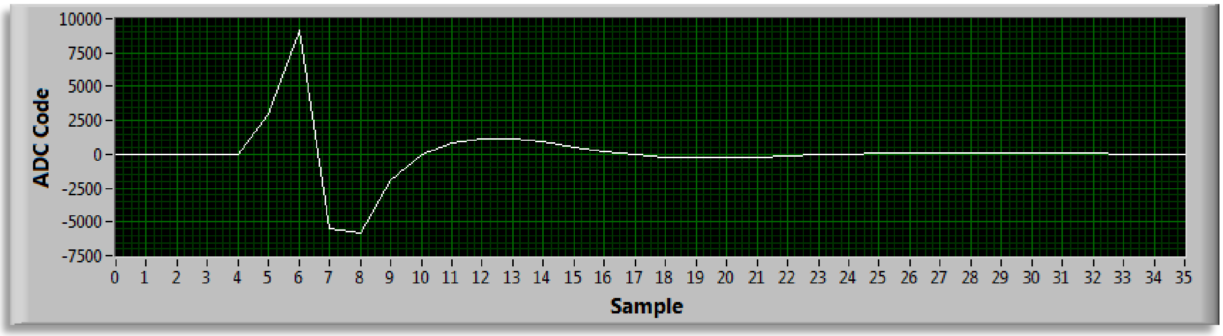
5.3.2.2 Collected Waveforms from the USRP

Two sets of waveforms were collected and will be presented here. The first set of waveforms was collected using an excitation pulse of short duration: 60 ns. The second set of waveforms uses a much longer excitation waveform of approximately 800 ns. Three waveforms were collected for each excitation pulse width to illustrate variations in the sampled waveform as a result of phase variance between the waveform and the sampling process. I note that data displayed in the waveform graphs in Table 5.2 uses an interpolating line format (as opposed to dots) when displaying waveform data.

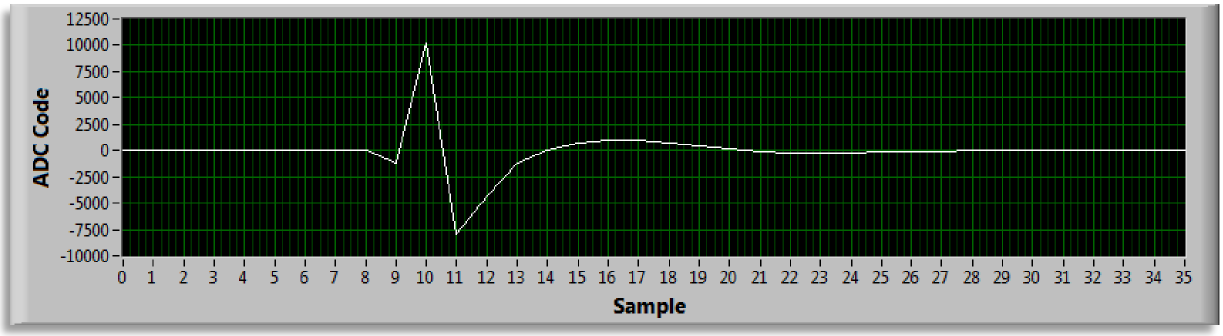
Table 5.2: Captured interrogation waveforms from the USRP.

Excitation Pulse 60ns:

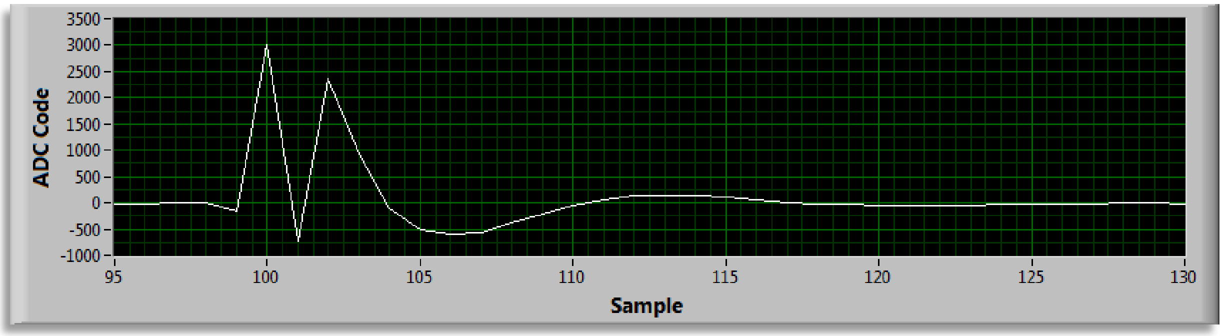
Waveform 1:



Waveform 2:

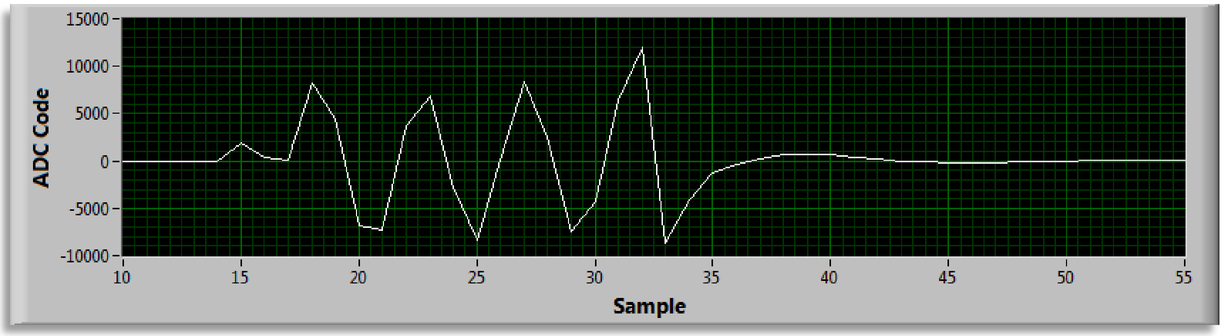


Waveform 3:

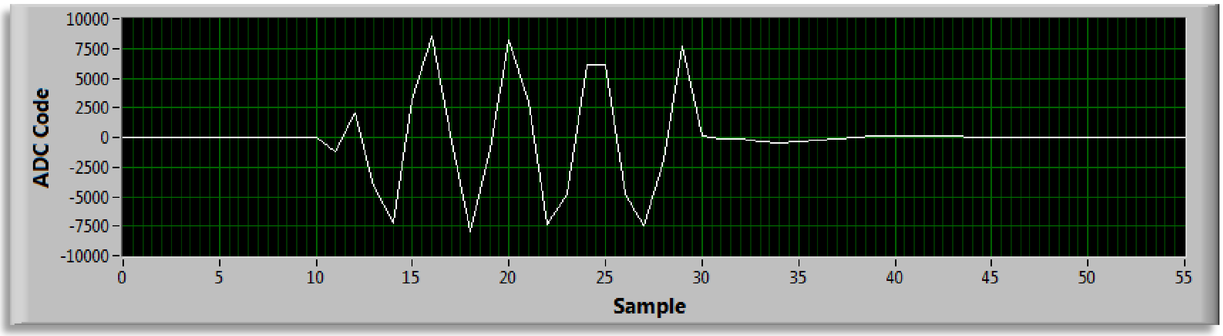


Excitation Pulse 800ns:

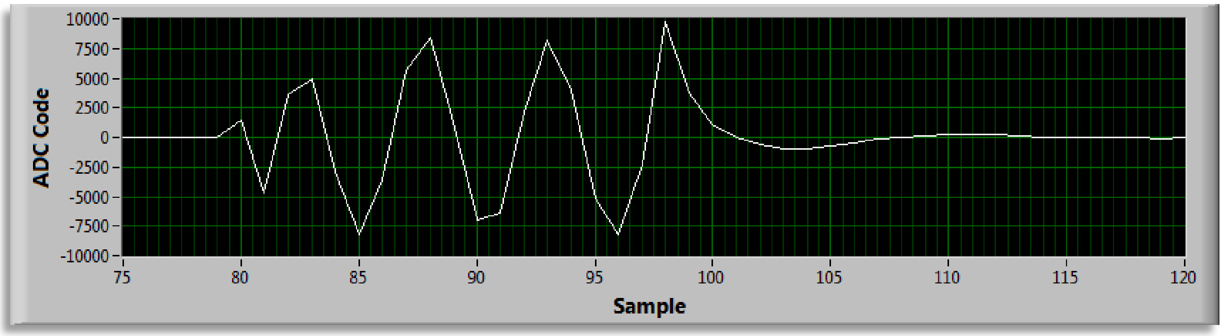
Waveform 1:



Waveform 2:



Waveform 3:



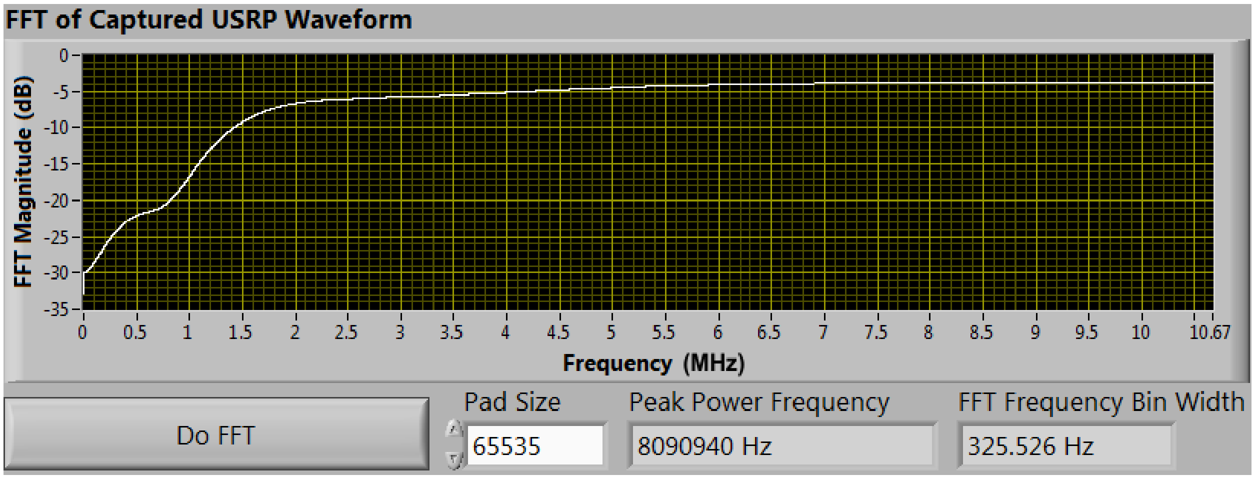
5.3.2.3 Post-Processing with the FFT

Post-processing is accomplished through the use of a Fourier transform and related code to gate the waveform components of interest. Gating is necessary such that only the resonator ringing is processed when performing the FFT: care needs to be taken such that the reflected excitation signal is not measured which can result in error when computing the peak power frequency (PPF) of the FFT. When proper gating is used on the response signal, the computed PPF of the FFT yields the resonator's resonant frequency.

A LabVIEW VI was written for the post-processing function. A hamming window was used in the FFT. Zero-padding was performed on the gated waveform data such that the buffer size of the waveform before FFT processing is approximately 65,535 samples in length. At a sample rate of 21.33 MHz with padding of the gated waveform, an FFT resolution of approximately 325 Hz is produced. Each of the waveforms shown in section 5.3.2.2 is post-processed twice for the purposes of this thesis: The first FFT shows the resulting frequency domain content when the entire resonator waveform is gated. In the second FFT, only the resonator ringing is gated, thus the frequency domain content will contain a PPF revealing the resonant frequency of the cavity resonator. I note here that the IF filter before the receiver front-end output does not function as an anti-aliasing filter. There is no anti-aliasing in the signal processing or before the USRP ADC.

Table 5.3: Waveform 1 FFT and PPF post-processing results, 60 ns excitation pulse.

FFT result with gating of excitation pulse and ringing together:

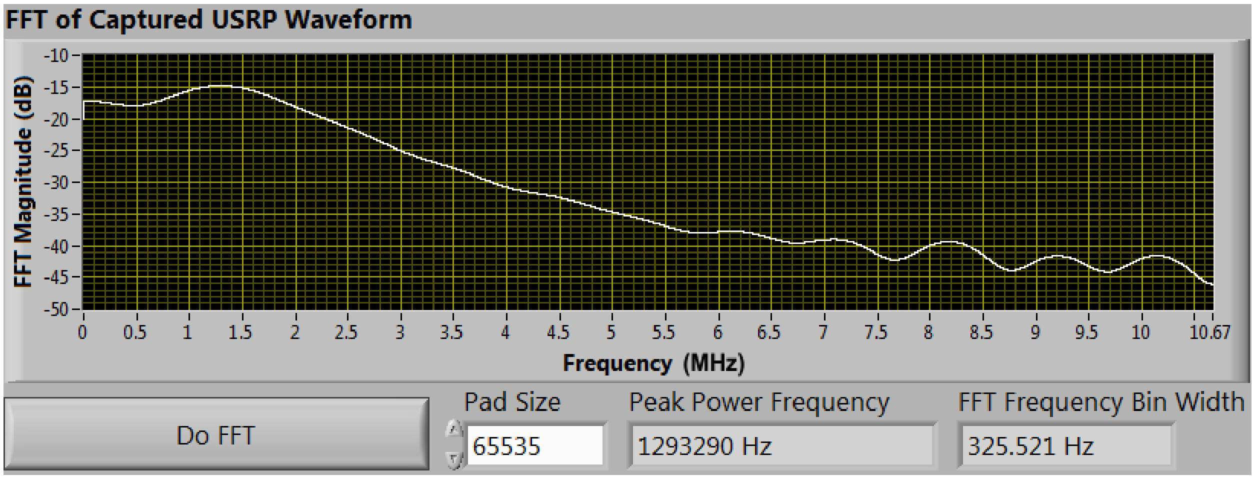


Pad Size: 65535

Peak Power Frequency: 8090940 Hz

FFT Frequency Bin Width: 325.526 Hz

FFT result gating only the resonator ringing:



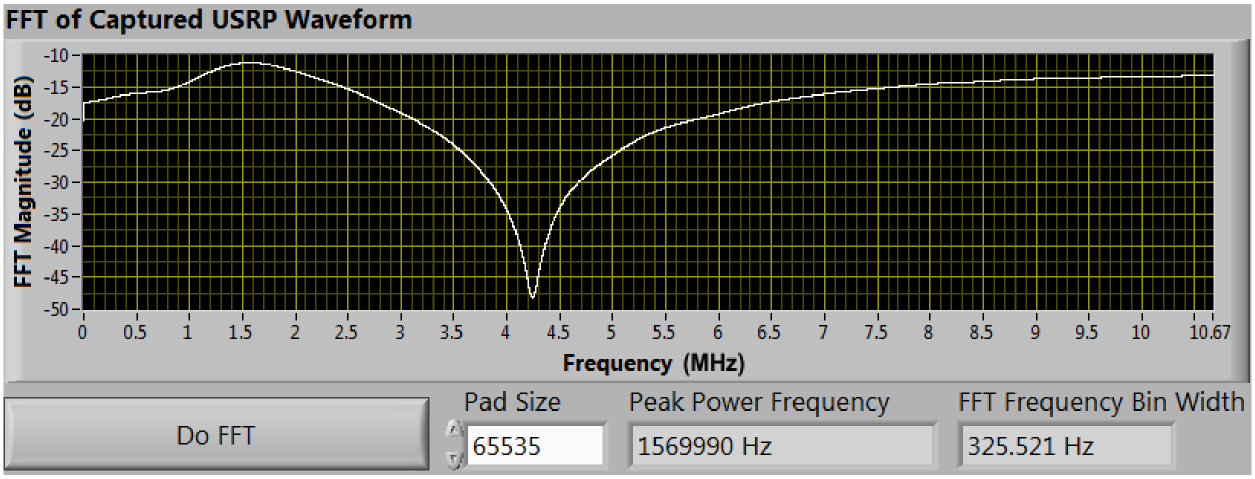
Pad Size: 65535

Peak Power Frequency: 1293290 Hz

FFT Frequency Bin Width: 325.521 Hz

Table 5.4: Waveform 2 FFT and PPF post-processing results, 60 ns excitation pulse.

FFT result with gating of excitation pulse and ringing together:

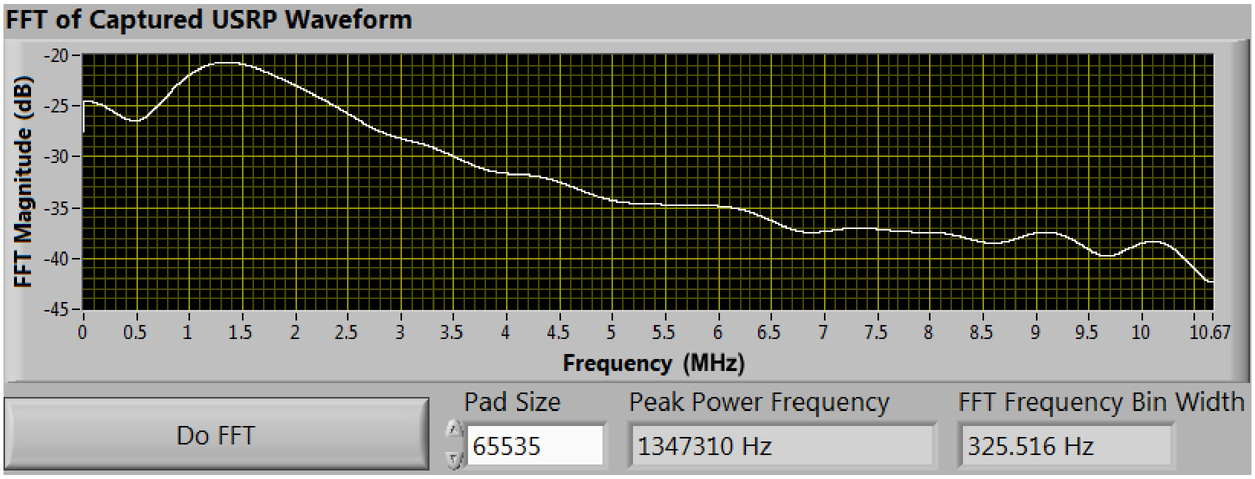


Pad Size: 65535

Peak Power Frequency: 1569990 Hz

FFT Frequency Bin Width: 325.521 Hz

FFT result gating only the resonator ringing:



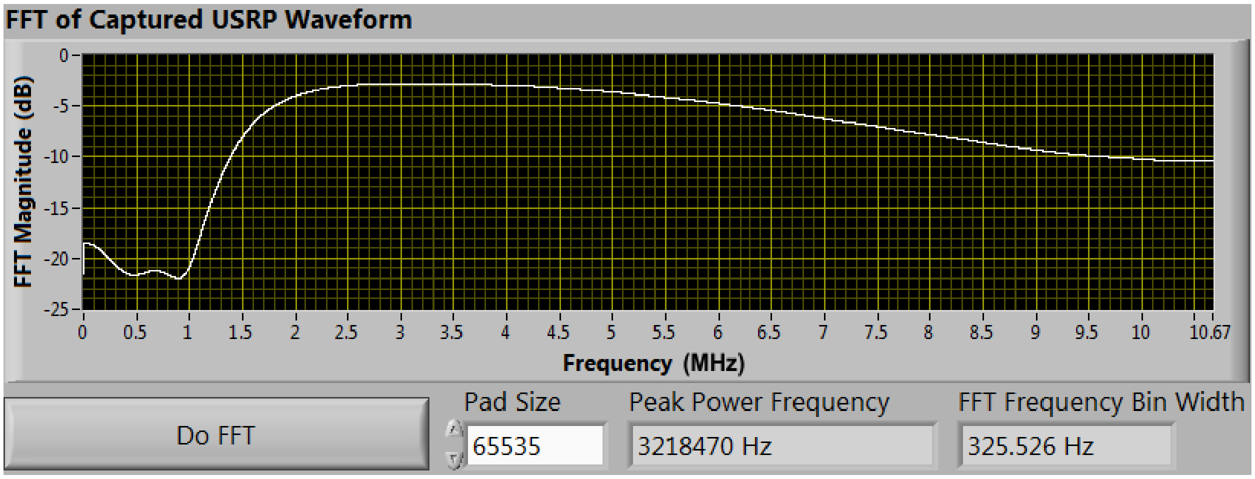
Pad Size: 65535

Peak Power Frequency: 1347310 Hz

FFT Frequency Bin Width: 325.516 Hz

Table 5.5: Waveform 3 FFT and PPF post-processing results, 60 ns excitation pulse.

FFT result with gating of excitation pulse and ringing together:

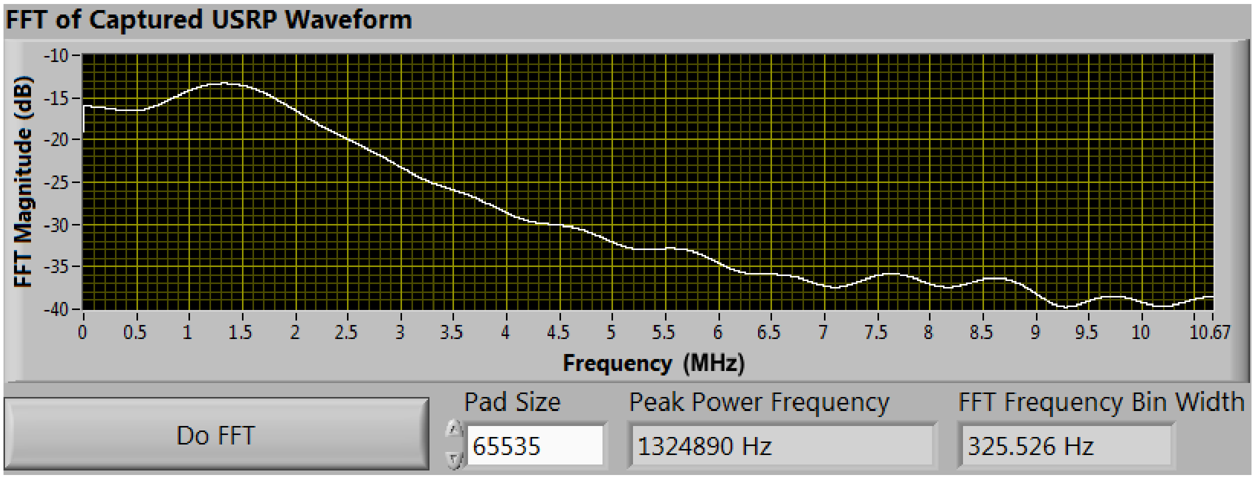

FFT of Captured USRP Waveform

Pad Size: 65535

Peak Power Frequency: 3218470 Hz

FFT Frequency Bin Width: 325.526 Hz

FFT result gating only the resonator ringing:


FFT of Captured USRP Waveform

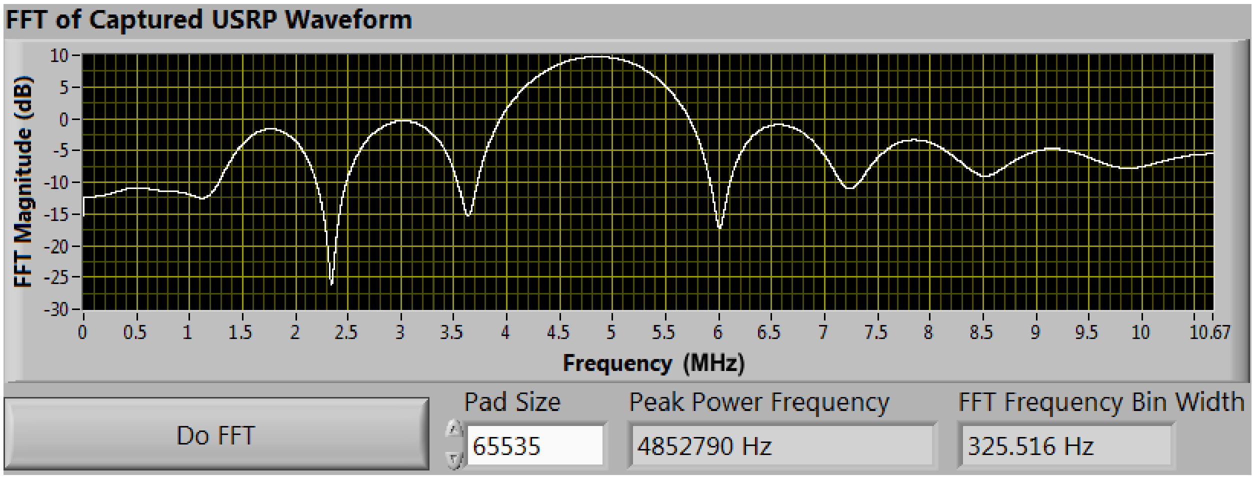
Pad Size: 65535

Peak Power Frequency: 1324890 Hz

FFT Frequency Bin Width: 325.526 Hz

Table 5.6: Waveform 1 FFT and PPF post-processing results, 800 ns excitation pulse.

FFT result with gating of excitation pulse and ringing together:

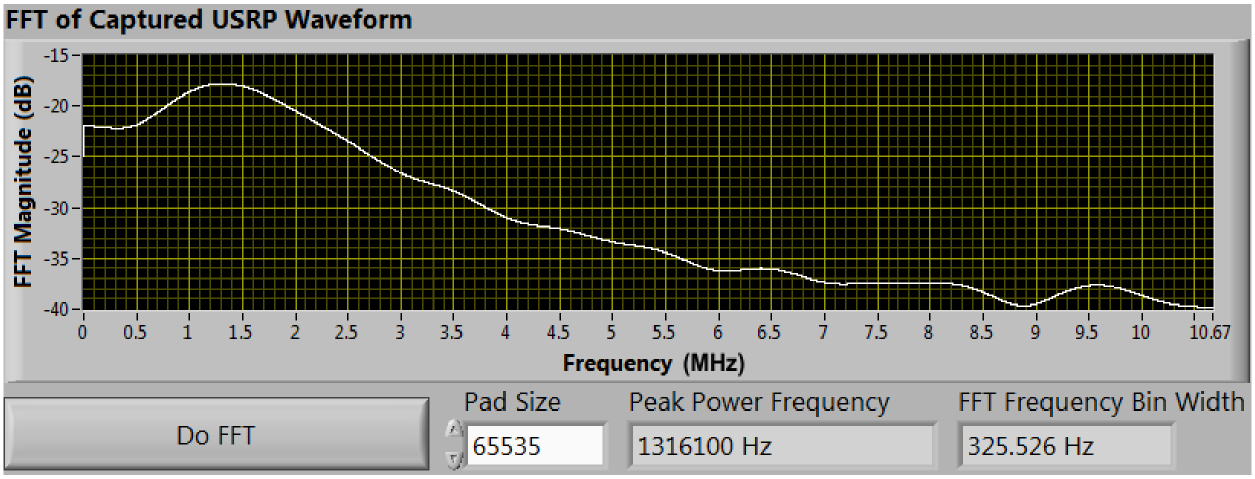


Pad Size: 65535

Peak Power Frequency: 4852790 Hz

FFT Frequency Bin Width: 325.516 Hz

FFT result gating only the resonator ringing:



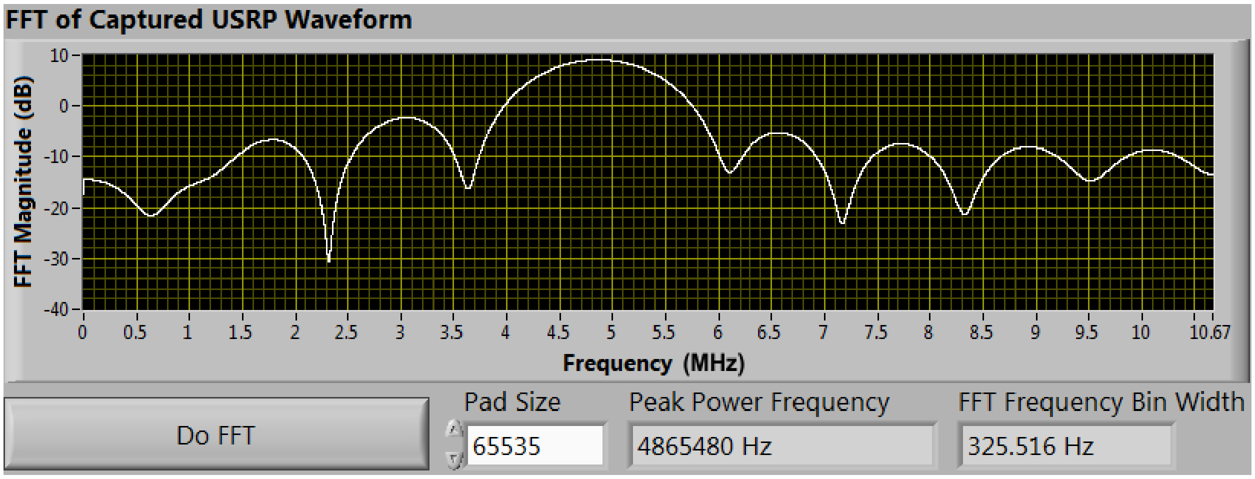
Pad Size: 65535

Peak Power Frequency: 1316100 Hz

FFT Frequency Bin Width: 325.526 Hz

Table 5.7: Waveform 2 FFT and PPF post-processing results, 800 ns excitation pulse.

FFT result with gating of excitation pulse and ringing together:

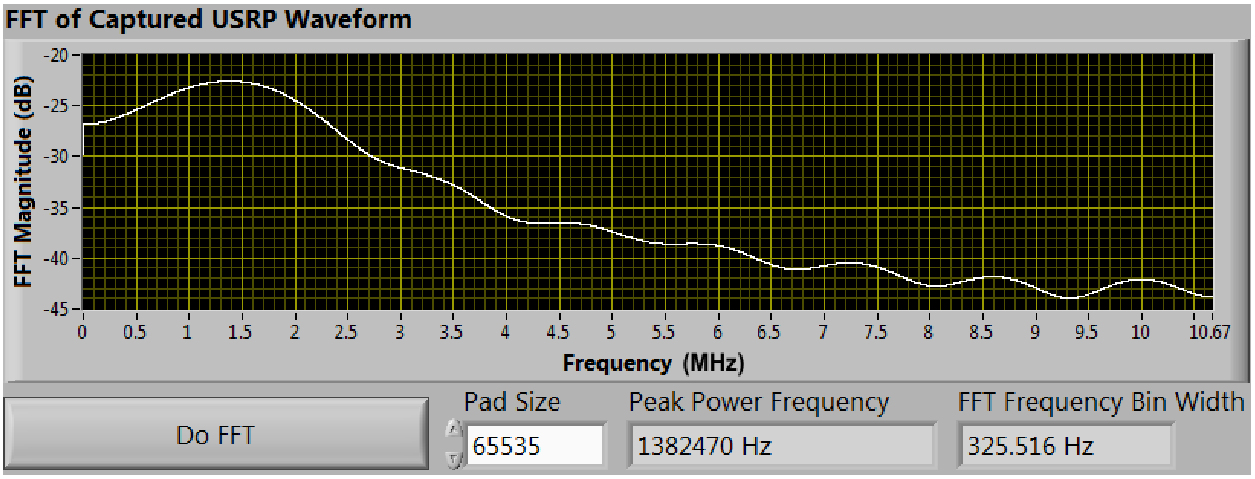


Pad Size: 65535

Peak Power Frequency: 4865480 Hz

FFT Frequency Bin Width: 325.516 Hz

FFT result gating only the resonator ringing:



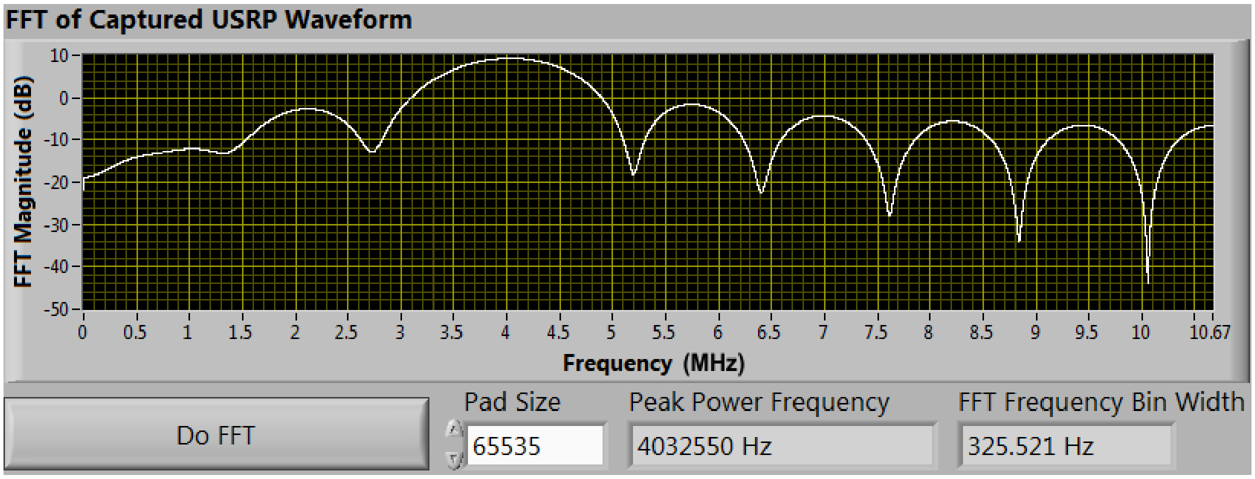
Pad Size: 65535

Peak Power Frequency: 1382470 Hz

FFT Frequency Bin Width: 325.516 Hz

Table 5.8: Waveform 3 FFT and PPF post-processing results, 800 ns excitation pulse.

FFT result with gating of excitation pulse and ringing together:

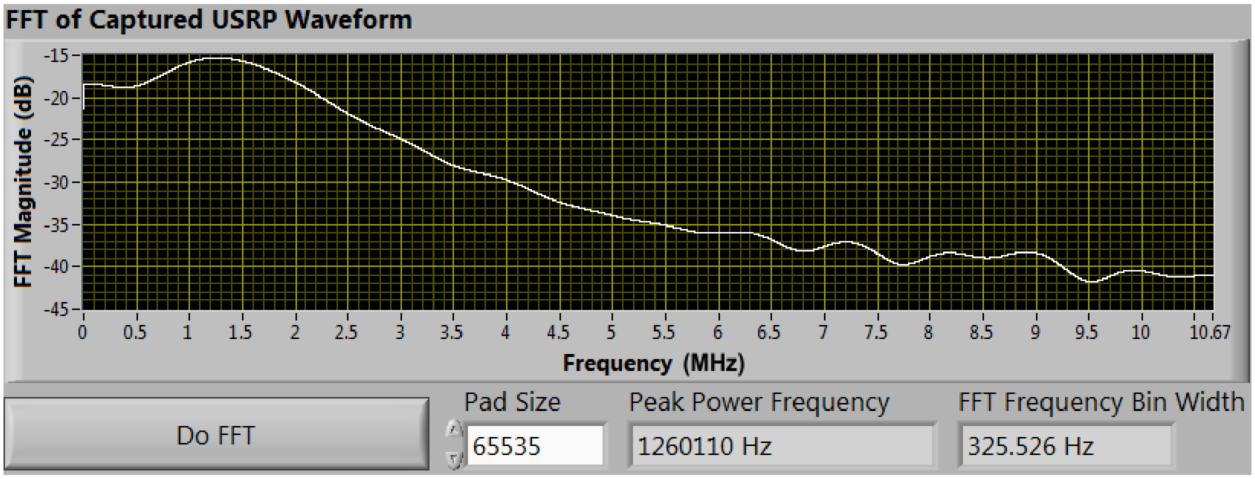


Pad Size: 65535

Peak Power Frequency: 4032550 Hz

FFT Frequency Bin Width: 325.521 Hz

FFT result gating only the resonator ringing:



Pad Size: 65535

Peak Power Frequency: 1260110 Hz

FFT Frequency Bin Width: 325.526 Hz

5.3.3 Discussion of Results

The results in section 5.3.2 illustrate the collected cavity resonator waveforms in both time domain and frequency domain. The PPF of the FFT output spectrum have been calculated.

In the following subsections I discuss: 1. Correlation of the IF measurement back to original RF spectrum. 2. Factors which limit the performance of the resonant frequency measurement.

5.3.3.1 Correlation of IF measurements to the resonant frequency

The resulting PPF values for the gated waveforms containing only the ringing of the cavity resonator after excitation can be translated to the resonant frequency of the cavity resonator. To calculate this value certain parameters need to be known, this includes: 1. Receiver VCO output tone frequency. 2. Sample rate of the ADC. 3. IF bandwidth center frequency (to calculate the Nyquist zone of the sampled data).

The first step in determining the RF frequency of the resonator response is to determine which Nyquist zone the PPF falls in. Once the Nyquist zone is known, we can find the IF frequency that the measured PPF values from 5.3.2.3 correspond to. From the IF frequency and the VCO output tone frequency the RF frequency prior to down conversion can be calculated. Because the VCO output tone frequency is greater than that of the modulated resonator waveform tones, we can use the equation:

$$LO - IF = RF$$

The collected waveforms in this thesis have an approximate IF frequency of 15 MHz, thus, the waveforms collected are captured in the 2nd Nyquist zone of the USRP given that I have achieved a sample rate of 21.33 MHz. Using the first resonator waveform captured with an excitation pulse of 800 ns, I will demonstrate the correlation of the measured PPFs to the RF frequency. The PPF of the modulated cavity resonator waveform fall within the 2nd Nyquist zone where the spectrum produced by the sampling process will be reversed: the translation equations depend upon whether the Nyquist zone of the tone is even or odd. I provide here the equations which will translate the measured PPF (PPF_m) to the actual PPF IF frequency (PPF_a) given the Nyquist zone(s) of the IF tones. Here, the sample rate is represented as f_s and the Nyquist zone is represented as n.

Table 5.9: Measured PPF to actual PPF transform equations.

Nyquist Zone (n):	**Equation:**
Even:	$\left(\frac{nf_s}{2}\right) - PPF_m = PPF_a$
Odd:	$\left(\frac{f_s}{2}\right)(n-1) + PPF_m = PPF_a$

From these equations I calculate the excitation signal to have an IF frequency of 17.48 MHz and the resonator ringing to have an IF frequency of 21.017 MHz. To calculate the RF frequency I add the VCO tone. Using the VCO measurement results in 5.2.2, a curve fit was performed to obtain the coefficients for a 6th order polynomial equation. From this equation I calculate the VCO output tone to be 10.634 GHz. This yields the excitation frequency to be 10.617 GHz (which matches the signal generator output frequency of 10.616 GHz) and the resonator's *X*-band resonant mode to be 10.613 GHz.

5.3.3.2 Factors limiting interrogator performance

Performance of the interrogator prototype is limited by components in the RF front-end as well as in the digital signal processing of the captured, resonator modulated signal. These factors include and are not limited to: 1. Phase noise and drift in the VCO of the RF front-end. 2. Phase noise of the USRP master clock. 3. Frequency resolution of the Fourier transform.

Phase noise produce in the VCO will produce phase noise in the IF output spectrum of the mixer, thus causing the modulated resonator signal to vary in frequency as the VCO does. Drift of the VCO due to temperature variation is possible (and was not noted during measurement). This is likely to be of concern where the VCO could be exposed to large temperature variations such as those which might be found under field conditions. For this reason, robust, precision measurements should be made using a VCO which has good thermal characteristics.

Phase noise of the USRP master clock can cause the actual sample rate of the AD9862 IC to vary slightly over time. This leads to the FFT producing results which may contain low levels of aliasing artifacts at the extremes of the FFT bandwidth.

The most notable limiting factor in the results presented is that of FFT resolution bandwidth (RBW). Due to the short time span of the modulated resonator signal, a high sample rate is necessary to ensure there enough samples are captured to produce an FFT with a low RBW, thus ensuring that calculation of the PPF can produce the most accurate result possible.

CHAPTER 6: CONCLUSIONS

In this thesis, I have developed and tested a functional *X*-band receiver built using COTS components and interfaced it with a digitizer for the acquisition of a resonator sensor waveform. The feasibility of SDR and RADAR techniques for resonator sensor interrogation has been explored. Waveforms have been captured from a cavity resonator and subsequently processed with an FFT for analysis of the resonant frequency. The PPF has been calculated and verified to match the resonant frequency of a resonant mode of a cavity resonator.

Collected data suggests that improvements to the USRP will be needed or a new digitizer selected to improve measurement accuracy and speed of the interrogation process. A reliable interrogation platform has nevertheless been developed which serves as the foundation for future research and improvements to wireless resonator sensor interrogation techniques and hardware.

CHAPTER 7: FURTHER RESEARCH

The work covered in this thesis serves as a first step and a proof-of-concept in the context of wireless passive sensor interrogation. As such, further research on this topic is essential to developing a more robust knowledge of sensor interrogator design. At present, I propose that continued research on this topic will continue in a number of sub-topics including: 1. Continued development and refining of software defined RADAR and related techniques. 2. Experimentation with interrogator receiver and transmitter hardware with focus on improving wireless measurement capabilities. 3. Automation of interrogator hardware and development of system algorithms for self-running, continuous measurements based on a series of user-configuration settings and desired system parameters. I will explain these three sub-topics in further detail in the following sections.

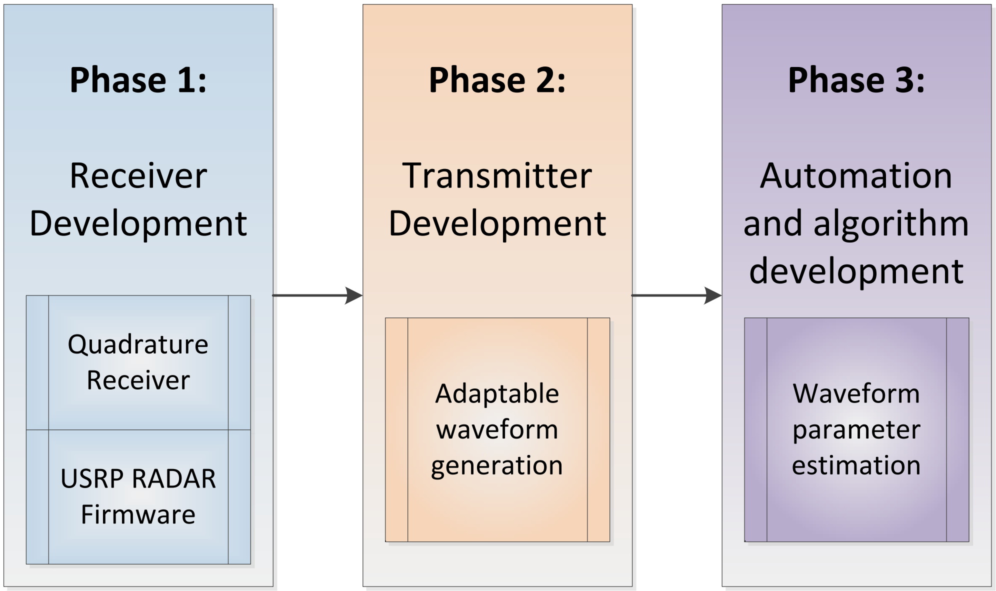


Figure 7.1: Continued research plan of attack.

7.1 Software-Defined RADAR Techniques

Development of software-defined RADAR techniques will enable more robust interrogation of resonator sensors. SDR differs from software defined RADAR techniques primarily in how the resulting signal is processed. SDR is typically concerned with signals that are continuous time. To alleviate bandwidth issues from the use of high-speed digital converters, decimation and interpolation techniques are used where narrow-bandwidth channels are used. In RADAR, continuous time signals are not the norm; a receiver may operate with less than a 100% duty cycle depending upon the RADAR configuration. For sensor interrogation, it is sufficient to process the response waveform itself. Much of the data from the interrogation channel is discarded. Use of triggering mechanisms can be used to process data on an FPGA or a high speed digital processor such that the full sample rate from the digitizers can be utilized, maximizing the receiver bandwidth in the digital domain while still lowering the overall bandwidth requirements within the system.

7.2 Interrogator Hardware

Continued research on interrogator hardware mainly is focused on the design of the receiver and transmitter sections: 1. Research on receiver design intends to improve the robustness of this hardware in the context of wireless sensing. 2. Design of a transmitter that can produce an optimal excitation waveform for excitation of the sensor within the receiver bandwidth is necessary for eventual automation of the interrogator.

The selected receiver design for this thesis is a single-conversion superheterodyne receiver. This receiver was selected for its low cost and simple design; it is not the most optimal receiver for sensor interrogation. To improve measurement accuracy and reliability, a phase coherent receiver is needed that is capable of producing quadrature output. Quadrature sampling can then be used in the USRP to preserve phase information into the digital domain. To design a phase coherent receiver will most likely require the use of a superheterodyne receiver for the down-conversion of the *X*-band signal to a lower IF center-frequency where the resulting quadrature demodulation can be performed with inexpensive and more simple to design low-frequency hardware. Preferably the center frequency of the IF for this receiver would be less than 70 MHz to simplify design requirements and provide for the largest component selection possible.

7.3 Interrogator Automation and Algorithm Development

Interrogator automation is the last phase of a long research program. Automation of the interrogator involves adapting the final receiver and transmitter design, and any other hardware necessary for interrogation of a resonator sensor, such that it can be controlled through a simple and common host interface. In the context of the USRP, receiver and transmitter hardware would be controlled via the SPI and I^2C serial busses. Algorithms would need to be developed to automatically search for and detect resonators within the interrogator sensing range. At this stage of interrogator design, the research focus would begin to shift from that of basic research to an applied approach. Existing research on wireless sensing of resonator sensors will be combined with a complex process of code development and testing until a fully automated, self-running resonator sensor measurement system is operational.

Additional research at this stage could include integration of heavily theory based components such as parameter estimation to further improve measurement accuracy of the wireless sensing platform.

APPENDIX A: INSTALLING AND RUNNING GNU RADIO

In this Appendix I provide usage notes and step-by-step instructions for installing GNU Radio to an Ubuntu Linux-based system. This section is intended to give the reader some background about the flexible nature of the GNU Radio libraries themselves, in addition to the flexibility that they offer in the context of SDR.

The nature of the GNU Radio code-base allows for installing GNU Radio under many different system configurations. GNU Radio can be compiled to run on PC hardware as well as on embedded-Linux systems. GNU Radio can be compiled under: 1. Linux. 2. BSD variants (FreeBSD, OpenBSD, etc.). 3. Windows. 4. MacOS X. GNU Radio under Windows can be compiled natively; compilation is aided by the support of Cygwin or MinGW libraries. By far the simplest method of installing GNU Radio is through the package manager of your preferred Linux distribution. It is my recommendation when installing GNU Radio on a Linux system that necessary prerequisites be installed through the package manager first and the GNU Radio sources are compiled from the Git repository to ensure that you are using the latest code. The Ubuntu 10.04 LTS repositories contain GNU Radio version 3.2.2, which is not the current stable version (3.3.0 at the time of this writing).

Each Linux distribution creates subtle differences in the steps that need to be performed to yield a functional installation of GNU Radio software. To aid the reader of this thesis, I will assume the user is operating an x86 PC and running the Ubuntu Linux distribution version 10.04 long-term support (LTS). Due to the complicated nature of installing any operating system, I will not cover the details of installing, setting up, or maintaining an existing Linux operating system or otherwise. It is assumed that the user has some basic knowledge of the Linux operating system before continuing.

The first step in installing the GNU Radio libraries is to install all necessary pre-requisites. The instructions here will install a generic list of prerequisites necessary for use of most tools and features within the GNU Radio libraries.

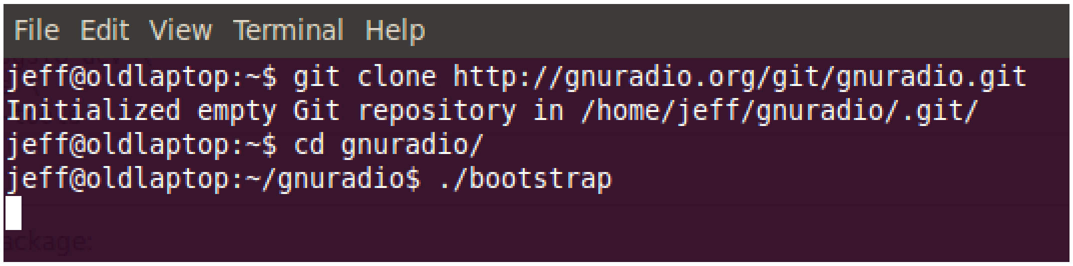
A.1 <u>GNU Radio Installation Procedures</u>

Step 1: Update your system with the command ‘`sudo apt-get update`’. This will ensure that you are installing the latest versions of available prerequisites onto your system. Whenever using sudo, you will need to have administrator privileges for the command to complete successfully.

Step 2: Install the necessary prerequisites with the following command (typed as one long string with no spaces):

‘
```
sudo apt-get -y install libfontconfig1-dev libxrender-dev libpulse-dev swig g++ automake autoconf libtool python-dev libfftw3-dev libcppunit-dev libboost-all-dev libusb-dev fort77 sdcc sdcc-libraries libsdl1.2-dev python-wxgtk2.8 git-core guile-1.8-dev libqt4-dev python-numpy ccache python-opengl libgsl0-dev python-cheetah python-lxml doxygen qt4-dev-tools libqwt5-qt4-dev libqwtplot3d-qt4-dev pyqt4-dev-tools
```
’.

Step 3: Download the latest GNU Radio source code from the Git repositories with the command ‘`git clone http://gnuradio.org/git/gnuradio.git`’.

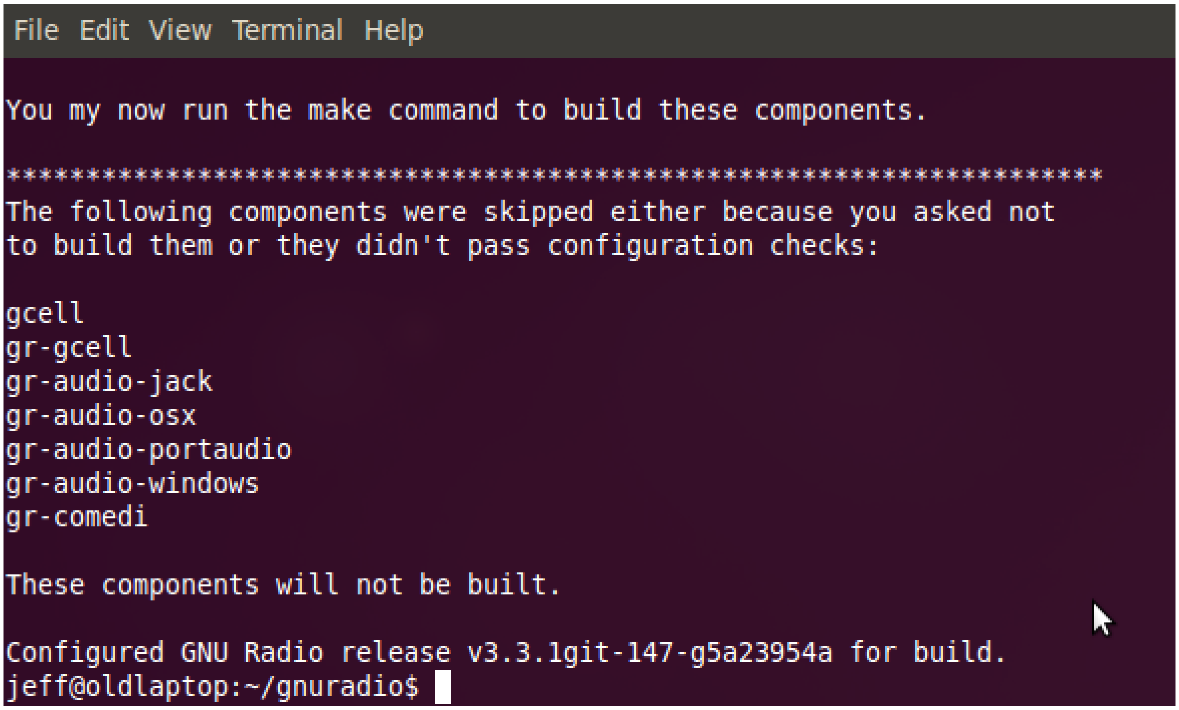


```
jeff@oldlaptop:~$ git clone http://gnuradio.org/git/gnuradio.git
Initialized empty Git repository in /home/jeff/gnuradio/.git/
jeff@oldlaptop:~$ cd gnuradio/
jeff@oldlaptop:~/gnuradio$ ./bootstrap
```

Figure A.1: Steps 3 and 4 of the GNU Radio install process.

Step 4: Now we'll bootstrap, configure, and compile the source code. This involves the following commands: 1. '`cd gnuradio`'. 2. '`./bootstrap`'. 3. '`./configure`'. 4. '`make`'.

Step 5: Check the binaries for errors and install them with: 1. '`make check`'. 2. '`sudo make install`'.



```
You my now run the make command to build these components.

**********************************************************************
The following components were skipped either because you asked not
to build them or they didn't pass configuration checks:

gcell
gr-gcell
gr-audio-jack
gr-audio-osx
gr-audio-portaudio
gr-audio-windows
gr-comedi

These components will not be built.

Configured GNU Radio release v3.3.1git-147-g5a23954a for build.
jeff@oldlaptop:~/gnuradio$
```

Figure A.2: Console output after the GNU Radio configuration process has completed.

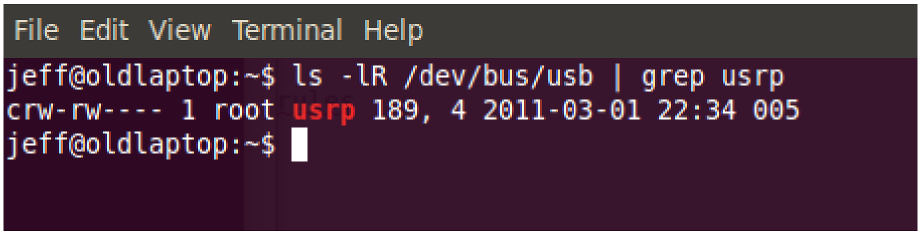
Step 6: At this point the binaries are now installed. To use the USRP, we’ll need to configure udev which is responsible for making the USRP visible to the operating system. Type the following commands: 1. ‘`sudo addgroup usrp`’. 2. ‘`sudo usermod -G usrp -a <YOUR_USERNAME>`’, where `<YOUR_USERNAME>` is your username. 3. ‘`echo 'ACTION=="add", BUS=="usb", SYSFS{idVendor}=="fffe", SYSFS{idProduct}=="0002", GROUP:="usrp", MODE:="0660"' > tmpfile`’. 4. ‘`sudo chown root.root tmpfile`’. 5. ‘`sudo mv tmpfile /etc/udev/rules.d/10-usrp.rules`’.

Step 7: This next step is a fix for what is considered by the GNU Radio project to be a broken libtool implementation. More information on libtool can be found here: http://www.gnu.org/software/libtool/. Type the following commands: 1. ‘`cp /etc/ld.so.conf /tmp/ld.so.conf`’. 2. ‘`echo /usr/local/lib >> /tmp/ld.so.conf`’. 3. ‘`sudo mv /tmp/ld.so.conf /etc/ld.so.conf`’. 4. ‘`sudo ldconfig`’.

Step 8: Restart your computer.

A.2 Verifying Operation of the USRP

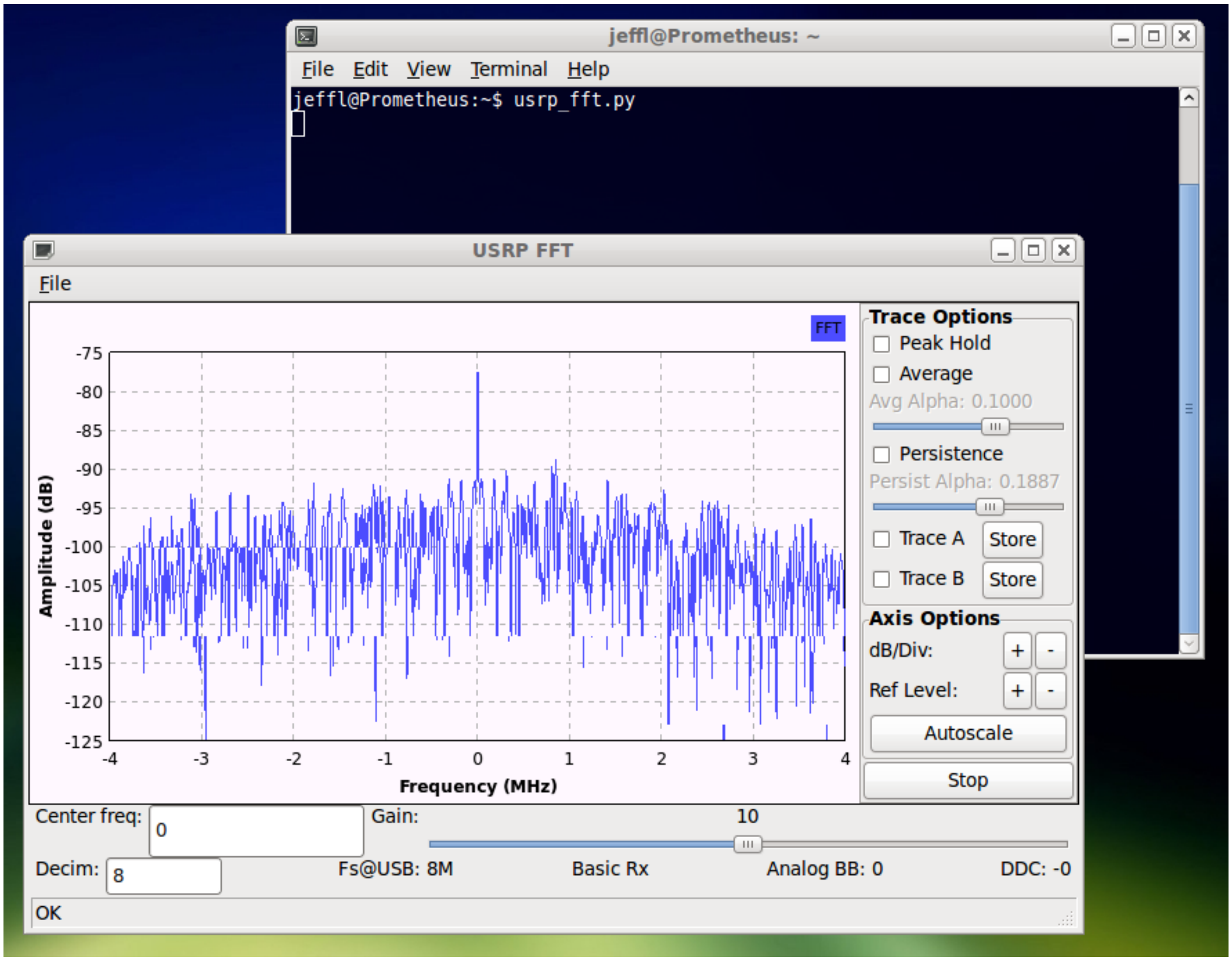
Step 1: Following installation of the GNU Radio libraries or to test a USRP device, the command ‘`ls –lR /dev/bus/usb | grep usrp`’ can be used. A line should appear below this command as shown in Figure A.3, below. If it does not, or it does not appear as it does below, you are likely to have issues using the USRP device. Repeating step 6 may resolve this issue.



```
jeff@oldlaptop:~$ ls -lR /dev/bus/usb | grep usrp
crw-rw---- 1 root usrp 189, 4 2011-03-01 22:34 005
jeff@oldlaptop:~$
```

Figure A.3: Verifying that the USRP has been detected by the operating system.

Step 2: Now that we have verified that the operating system can see the USRP hardware, we can test it using the usrp_fft.py script. To do this, type the command '`usrp_fft.py`' at a terminal window. A screen should appear as in Figure A.4 labeled 'USRP FFT'. This script simply processes data from the USRP through an FFT algorithm and then displays the resulting output. If the user wishes to test the maximum bandwidth possible for a given system configuration, the decimation factor can be set to 8 (this is the minimum value that can be used without special configuration). In the terminal window that the usrp_fft.py script is launched from, the letters "UoUo" may appear if buffer overflows and underruns are detected. It is normal that this may happen sporadically even on a relatively fast computer. If underruns occur consistently it is likely that the system is not fast enough to operate at the maximum USRP output bandwidth.



```
jeffl@Prometheus:~$ usrp_fft.py
```

□ Peak Hold
□ Average
Avg Alpha: 0.1000
□ Persistence
Persist Alpha: 0.1887
□ Trace A
□ Trace B

Center freq: 0
Gain: 10
Decim: 8
Fs@USB: 8M
Basic Rx
Analog BB: 0
DDC: -0
OK

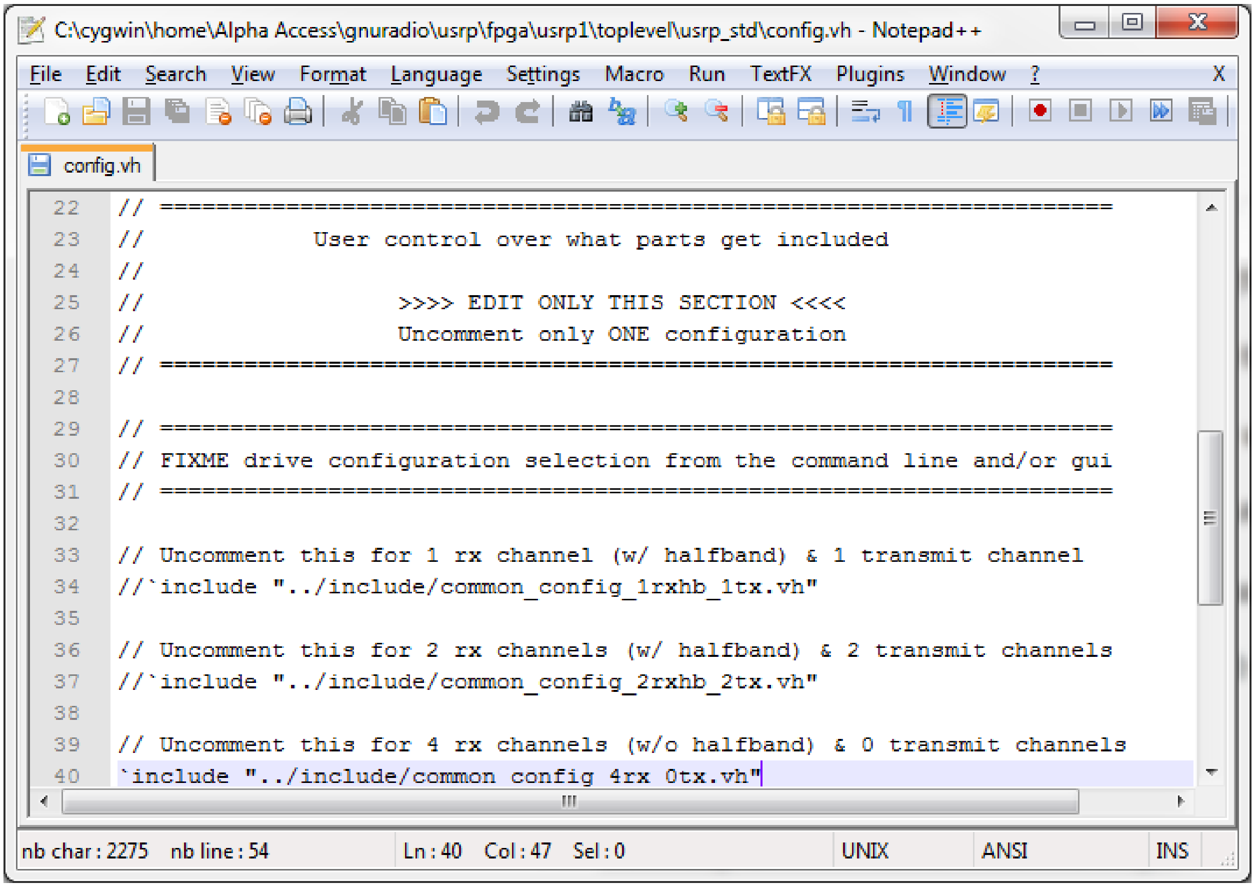
Figure A.4: Running the usrp_fft.py script to verify operation of a USRP device and check maximum possible host bandwidth.

APPENDIX B: MODIFICATIONS TO THE USRP FIRMWARE

Modifications made to the USRP FPGA firmware are specified here. Compiled firmware images for the USRP are included and installed with the GNU Radio libraries by default (they are located in /usr/share/usrp/rev4/ unless the installation path is changed). Firmware can be compiled using Altera Quartus II software. Quartus II Web edition is available for free and is capable of compiling the firmware used by the USRP FPGA. To obtain the latest FPGA firmware source code, you will need to download it to your system using the command '`git clone git://ettus.sourcerepo.com/ettus/fpga.git`'. On my own system, I moved the FPGA repository to the FPGA folder within the GNU Radio repository.

B.1 Disabling the NCO and Decimation Stages

For this firmware, I elected to modify the standard USRP FPGA firmware. A Quartus II project file is available under /fpga/usrp1/toplevel/usrp_std/. Under this same directory, config.vh was edited to change the firmware to compile with 4 receive channels and no transmit channels (See Figure B.1). Next, the file common_config_4rx_0tx.vh was edited under the directory /fpga/usrp1/toplevel/include/. In a similar manner to config.vh, the lines '`` `define RX_NCO_ON ``' and '`` `define RX_CIC_ON ``' are commented out by prepending them with double forward slashes ('//'). I recommend that these files are edited before loading the Quartus II project.



```
// ====================================================================
//              User control over what parts get included
//
//                    >>>> EDIT ONLY THIS SECTION <<<<
//                    Uncomment only ONE configuration
// ====================================================================

// ====================================================================
// FIXME drive configuration selection from the command line and/or gui
// ====================================================================

// Uncomment this for 1 rx channel (w/ halfband) & 1 transmit channel
//`include "../include/common_config_1rxhb_1tx.vh"

// Uncomment this for 2 rx channels (w/ halfband) & 2 transmit channels
//`include "../include/common_config_2rxhb_2tx.vh"

// Uncomment this for 4 rx channels (w/o halfband) & 0 transmit channels
`include "../include/common_config_4rx_0tx.vh"
```

Figure B.1: Editing the file ‘config.vh’ to compile a modified USRP FPGA firmware.

Before the code can be compiled, additional steps are needed. Changing the configuration files instructs the compiler to use specific blocks of code; some bugs exist that don’t cause an issue when the NCO and decimation stages are enabled. Required changes necessary for a successful compile are: 1. Changing the paths to various dependencies used within the code. 2. Correcting the receive chain signal flow. Correcting the dependency paths will vary from configuration to configuration and should be trivial for the user. Receive chain signal flow is determined by code within the file rx_chain.v. My modifications to this code are outlined in the table below; modified code is based on a patch submitted to the GNU Radio project (see http://gnuradio.org/redmine/issues/show/360).

Table B.1: Code modifications to rx_chain.v necessary to successfully modify a custom USRP FPGA firmware. Modified code is highlighted in yellow.

Rx_chain.v lines 51-89:

```
`ifdef RX_NCO_ON
    phase_acc #(FREQADDR,PHASEADDR,32) rx_phase_acc
     (.clk(clock),.reset(reset),.enable(enable),

.serial_addr(serial_addr),.serial_data(serial_data),.serial_strobe(serial_str
obe),
      .strobe(sample_strobe),.phase(phase) );

   cordic rx_cordic
     ( .clock(clock),.reset(reset),.enable(enable),
       .xi(i_in),.yi(q_in),.zi(phase[31:16]),
       .xo(bb_i),.yo(bb_q),.zo() );
`else
   assign bb_i = i_in;
   assign bb_q = q_in;
   //assign sample_strobe = 1; // PATCH
`endif // !`ifdef RX_NCO_ON

`ifdef RX_CIC_ON
   cic_decim cic_decim_i_0
     ( .clock(clock),.reset(reset),.enable(enable),

.rate(decim_rate),.strobe_in(sample_strobe),.strobe_out(decimator_strobe),
       .signal_in(bb_i),.signal_out(hb_in_i) );
       assign hb_strobe_in = decimator_strobe;
`else
   assign hb_in_i = bb_i;
   //assign decimator_strobe = sample_strobe; // PATCH
   assign hb_strobe_in = sample_strobe; // PATCH
`endif

`ifdef RX_HB_ON
   halfband_decim hbd_i_0
     ( .clock(clock),.reset(reset),.enable(enable),
       //.strobe_in(decimator_strobe),.strobe_out(hb_strobe), // PATCH
       .strobe_in(hb_strobe_in),.strobe_out(hb_strobe), // PATCH
       .data_in(hb_in_i),.data_out(i_out),.debugctrl(debugctrl) );
`else
   assign i_out = hb_in_i;
   //assign hb_strobe = decimator_strobe; // PATCH
   assign hb_strobe = hb_strobe_in;
`endif
```

Rx_chain.v lines 100-108:

```
`ifdef RX_HB_ON
   halfband_decim hbd_q_0
     ( .clock(clock),.reset(reset),.enable(enable),
       //.strobe_in(decimator_strobe),.strobe_out(), // PATCH
       .strobe_in(hb_strobe_in),.strobe_out(), // PATCH
       .data_in(hb_in_q),.data_out(q_out) );
`else
   assign q_out = hb_in_q;
`endif
```

Following modifications, the code can be compiled. The Python script usrp_fft.py can be used to test that the decimation and NCO are disabled. It is worth noting that the lowest decimation value settable will remain 4 when using this modified firmware because of checks that occur before setting the FPGA registers. The decimation and center frequency values will no longer have effect.

APPENDIX C: LAYOUTS AND SCHEMATICS

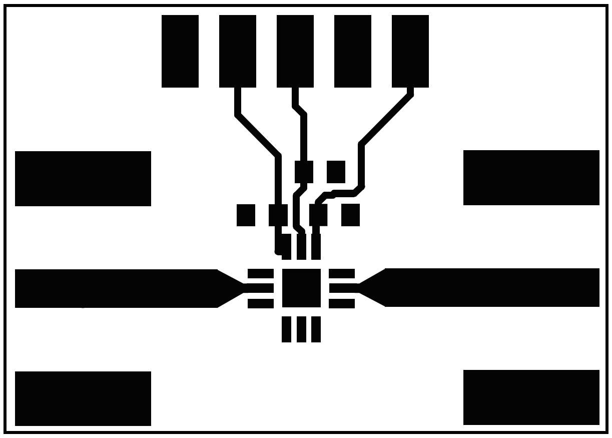


Figure C.1: Original PCB Layout for the TGA2512-2-SM amplifier, as measured (top layer). Bottom layer is ground plane.

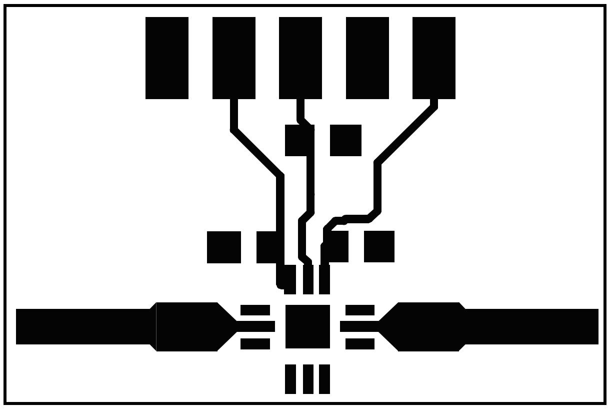


Figure C.2: Improved PCB Layout for the TGA2512-2-SM amplifier (top layer). Bottom layer is ground plane.

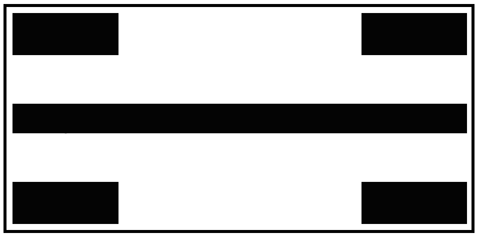


Figure C.3:Original “thru” board as measured for connector loss measurement and optimization (top layer).

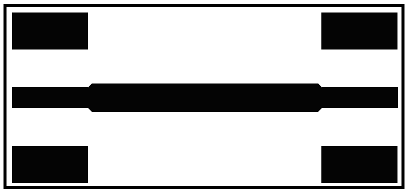


Figure C.4: Tapered “thru” board with ground pads for SMA connector (top layer).

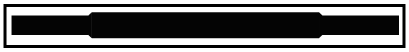


Figure C.5: Modified tapered “thru” board (top layer) with ground pads removed for connector modifications.

LIST OF REFERENCES

[1] Florin Udrea, Sumita Santra, and Julian W. Gardner, "CMOS Temperature Sensors - Concepts, State-of-the-Art and Prospects," *International Semiconductor Conference (CAS)*, vol. 2, pp. 31-40, October 2008.

[2] Agilent. (2010, March) Appl. Note 290: Practical Temperature Measurements. PDF.

[3] M. Nowak, E. Colinet, N. Delorme, F. Conseil, and G. Jacquemod, "A Wireless Sensing Platform for Battery-Free Sensors," *International Symposium on Circuits and Systems*, pp. 2122-2125, May 2008.

[4] R. Fachberger, G. Bruckner, R. Hauser, and L. Reindl, "Wireless SAW Based High-Temperature Measurement Systems," *International Frequency Control Symposium and Exposition*, pp. 358-367, June 2006.

[5] Michael Suster, Wen H. Ko, and Darrin J. Young, "Optically-Powered Wireless Transmitter for High-Temperature MEMS Sensing and Communication," *12th International Conference on Transducers, Solid-State Sensors, Actuators and Microsystems*, vol. 2, pp. 1703-1706, June 2003.

[6] E. Benes, M. Groschl, F. Seifert, and A. Pohl, "Comparison between BAW and SAW Sensor Principles," *IEEE International Frequency Control Symposium*, pp. 5-20, May 1997.

[7] Ya Wang, Yi Jia, Qiushui Chen, and Yanyun Wang, "A Passive Wireless Temperature Sensor for Harsh Environment Applications," *Sensors*, vol. 8, no. 12, pp. 7892-7995, December 2008.

[8] L. Reindl et al., "SAW Devices as Wireless Passive Sensors," *IEEE Ultrasonics Symposium*, vol. 1, pp. 363-367, November 1996.

[9] Merrill I. Skolnik, *Introduction to RADAR Systems*, 2nd ed. USA: McGraw-Hill, 1980.

[10] Avionics Department AIR-4.5. (1999, April) Electronic Warfare and RADAR Systems Engineering Handbook. PDF.

[11] A. G. Stove, "Linear FMCW radar techniques," *IEE Proceedings F Radar and Signal Processing*, vol. 139, no. 5, pp. 343-350, October 1992.

[12] John L. Shanton III, "A Software Defined Radio Transformation," *IEEE Military Communications Conference*, pp. 1-5, October 2009.

[13] Alan C. Tribble, "The Software Defined Radio: Fact and Fiction," *IEEE Radio and Wireless Symposium*, pp. 5-8, January 2008.

[14] Peter G. Cook and Wayne Bonser, "Architectural Overview of the SPEAKeasy System," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 4, pp. 650-661, April 1999.

[15] Danilo Valerio, "Open Source Software-Defined Radio: A Survery on GNUradio and its applications," Forschungszentrum Telekommunikation Wien, Vienna, Technical Report FTW-TR-2008-002, 2008.

[16] Josh Blum. [Online]. http://www.joshknows.com/grc/

[17] Ettus Research, LLC. (2010, December) [Online]. http://www.ettus.com/

[18] Analog Devices. (2010, December) AD9862 Datasheet. [Online]. http://www.analog.com/static/imported-files/data_sheets/AD9860_9862.pdf

[19] Ettus Research, "The USRP System," USRP Datasheet.

[20] USB Implementors Forum, Inc. (2000, April) Universal Serial Bus Specification (Revision 2.0). PDF.

[21] Alfred Pohl, Gerald Ostermayer, and Franz Seifert, "Wireless Sensing Using Oscillator Circuits Locked to Remote High-Q SAW Resonators," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 45, no. 5, pp. 1161-1168, September 1998.

[22] Matthias Hamsch, Rene Hoffmann, Werner Buff, Michael Binhack, and Stefan Klett, "An Interrogation Unit for Passive Wirelss SAW Sensors Based on Fourier Transform," *IEEE Transaction on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 51, no. 11, pp. 1449-1456, November 2004.

[23] Agilent. (2010, August) Appl. Note 57-1: Fundamentals of RF and Microwave Noise Figure Measurements. PDF.

[24] RFMD, "Wideband MMIC VCO With Buffer Amplifier, 8 GHz to 12 GHz," RFVC1800 Datasheet.

[25] Mini-Circuits, "Coaxial Frequency Mixer," ZX05-153+ Datasheet.

[26] TriQuint, "X Band Low Noise Amplifier," TGA2512 Datasheet.

[27] Mini-Circuits, "Coaxial Amplifier," ZFL-1000VH2X Datasheet.

[28] Emerson Network Power, "SMA 50 Ohm End Launch Jack Receptacle," SMA Connector Datasheet.