Linearization using Digital Predistortion

of a High-Speed, Pulsed, Radio Frequency Power

Amplifier for VHF Radar Depth-Sounder Systems

by

Kevin M. Player

B.Sc. Electrical Engineering, University of Kansas, 2007

Submitted to the graduate degree program in Electrical Engineering and Computer Science and the Graduate Faculty of the University of Kansas School of Engineering in partial fulfillment of the requirements for the degree of Master of Science.

Chairperson: --
Dr. Prasad Gogineni

Committee Members: --
Dr. Fernando Rodriguez-Morales

Dr. Sarah Seguin

Date of Defense: 29 March 2010

The Thesis Committee for Kevin Player certifies that this is the approved version
of the following thesis:

Linearization using Digital Predistortion

of a High-Speed, Pulsed, Radio Frequency Power

Amplifier for VHF Radar Depth-Sounder Systems

Committee:

Chairperson: ---
                    Dr. Prasad Gogineni

Committee Members: ---
                    Dr. Fernando Rodriguez-Morales

                    Dr. Sarah Seguin

Date approved: 29 March 2010

ACKNOWLEDGEMENTS

ABSTRACT

Depth-sounding radar systems provide the scientific data that are useful in modeling polar ice sheets and predicting sea-level rise. These radars are typically deployed on crewed aircraft; however, crewed missions over polar regions are difficult and dangerous. Thus, CReSIS is developing uninhabited aerial vehicles (UAVs) from which fine-resolution measurements can be made over vast areas. These fine-resolution measurements require highly linear power amplifiers (PAs) to create low range side-lobe levels. However, highly linear PAs are typically less efficient and require large and bulky heat sinks for heat dissipation, which increases the payload weight and decreases flight time. Furthermore, the linear FM chirp signal used for these radar systems creates Fresnel ripples and side-lobes will be generated when there are deviations from the ideal rectangular spectrum amplitude even with efficient windowing techniques, such as a Tukey window. Therefore, a 100 W, high-speed, pulsed, VHF power amplifier was developed and linearized using memoryless digital predistortion (DP) to obtain high linearity and high efficiency. The DP linearization decreased near-range side-lobe levels 11 dB from -46 dBc to -57 dBc, with a maximum reduction in the far-range side-lobe levels of 17 dB over the Tukey (transmit) and Blackmann[2] (receive) windowing alone. The high-speed switching circuit reduced current consumption to 117 mA (or 3.28 W at +28 V) for a 10-us pulse at 1-kHz PRF.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1: Introduction

1.1 Background

In recent years there have been debates in the political and scientific communities regarding climate change and sea level rise and their impact on the earth and its inhabitants. Current scientific data shows the ice sheets are melting and sea levels are rising. Regardless of whether these events are man-made or a natural part of the earth's cycle, there is a need to predict the sea level rise and the impacts of climate change on the earth. Since nearly 100 million people live within 1 meter of the current mean sea level [1,2] and 37% of the world's population lives in coastal regions, defined as being within 100 km of the coast [1], understanding possible contributions to sea level rise is of considerable importance. If the climate change trends continue, the social and economic consequences will be severe, especially to developing countries with little resources [1, 3].

1.2 Motivation

The Center for Remote Sensing of Ice Sheets (CReSIS) was founded by the National Science Foundation (NSF) in 2005 for the purpose of advancing scientific understanding of polar ice sheets as they respond to the global climate. Depth-sounding radars provide measurements of ice thickness, basal conditions below the ice, internal layers, liquid water layers or channels, and many other scientific parameters that are useful in modeling polar ice sheets and predicting sea-level rise. Currently, surface-based and aircraft-based platforms are the most common methods for transporting the radars for collecting data. Surface- based platforms are slow and cannot be safely used in crevassed areas. Therefore, CReSIS currently deploys a 150-MHz depth-sounding radar system on crewed aircraft such as the Orion P-3 and Twin Otter DHC-6. To obtain data over regions undergoing rapid changes that include fast-flowing glaciers, aircraft have to be flown at very low altitudes. This is difficult to accomplish over fast-flowing glaciers with steep. To resolve these difficulties, CReSIS is developing uninhabited aerial vehicles (UAV) from which fine-resolution measurements can be made over vast areas.

The Meridian UAV, shown in Figure 1, has been developed specifically to support low-altitude polar remote sensing missions. The current radar operates from 180 to 210 MHz using a distributed architecture with transmit/receive (T/R) modules located on the wings for signal conditioning prior to transmission or following reception. Eight wide-bandwidth antennas will be suspended beneath the wings, each with an attached T/R module providing signal amplification, filtering and switching. Currently, the CReSIS Aerial Vivaldi (CAV-A) Antenna will be used with the T/R module mounted in the area shown in Figure 2 [4, 5].
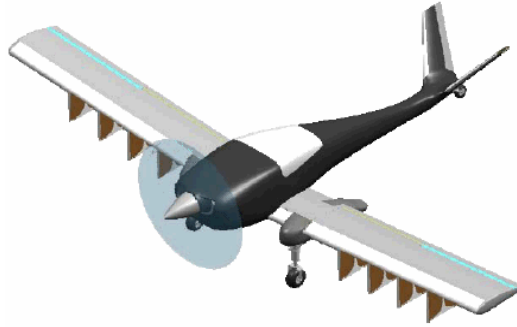
Figure 1: Meridian UAV with Vivaldi Antennas [4]
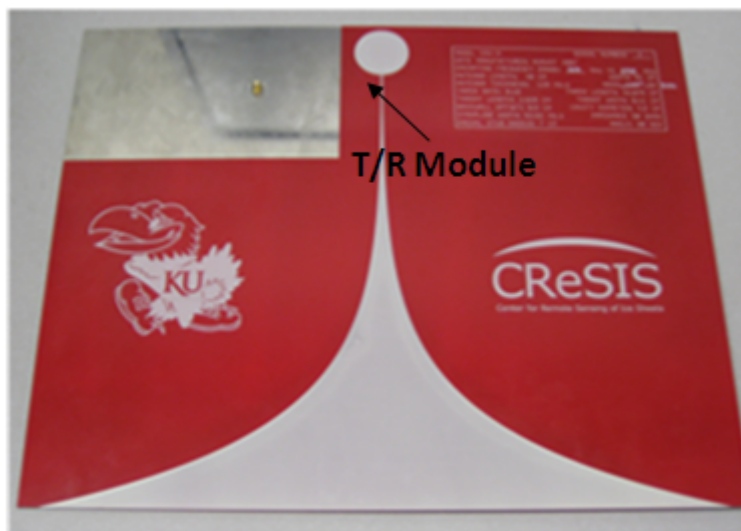
Figure 2: CReSIS Aerial Vivaldi (CAV-A) Antenna [5]

To accommodate both CReSIS depth-sounding radar systems, the operational bandwidth of the designed amplifier will extend from 140 to 210 MHz. As seen in Figure 2, the length and width dimension of the T/R module are governed by the available mounting area on the CAV-A antenna while the height (or thickness) dimension is restricted due to aerodynamic limitations of wing-mounted structures on the UAV.

Linearization of the power amplifier will allow it to operate at or near the 1-dB compression point; thereby, achieving higher power, higher efficiencies and better linearity. Higher power will increase the radar range improving depth-sounding capabilities, while higher efficiencies will reduce the size and weight of the heatsink needed to dissipate the heat generated by the PA. Even though the amplifier will be operated in polar regions where heat management would not be a major problem, a ground-based, radar system integration test is typically performed in non-polar regions. Improved linearity will reduce the side-lobes of the transmitted signal which will improve detection of off-nadir returns and improve internal layer resolution.

For RF power amplifiers, MOSFETs are used because they require simpler biasing and drivers, are more stable, switch faster, and do not experience thermal runaway. The main disadvantage of using FETs is they are less efficient than bipolar junction transistors [6]. The primary focus of this research deals with MOSFET amplifiers; therefore, all subsequent references to transistors and/or amplifiers will be assumed to be in reference to MOSFETs, unless otherwise stated.

Chapter 2: Overview of Power Amplifier Fundamentals

2.1 Relevant Figures of Merit for Power Amplifiers

2.1.1 Gain and Output Power

The gain of an amplifier is the ratio of the output power to input power, usually expressed in decibels (or dB). The most commonly used definition of power gain is the transducer gain, G_T, shown in Equation 1.

$$G_T = \frac{P_{load}}{P_{avail}}$$

(1)

P_{load} is the power delivered to the load by the amplifier, where the load may or may not be matched to the amplifier's output impedance. P_{avail} is the power available from the source assuming a matched condition at the amplifier input (i.e. $Z_i = Z^*_s$). Rewriting Equation 1 in terms of voltages and resistances, it can be shown that the maximum power will be delivered to the load by the amplifier if the load is matched to the conjugate of the amplifier's output impedance ($Z_L = Z^*_O$). However, as explained in [7], the conjugate match theorem only applies in a completely unrestricted case where currents and voltages at the generator terminals are unbounded by physical constraints, such as maximum transistor voltage ratings or the maximum available voltage from the DC supply. Therefore, by selecting a lower value of load resistance (or impedance) along the 'load-line' will accommodate the maximum permissible current and voltage swings at the output of the transistor, as illustrated in Figure 3. The resistance that provides maximum output power match is often referred to as 'R_{opt}' (or 'Z_{opt}' for reactive loads).
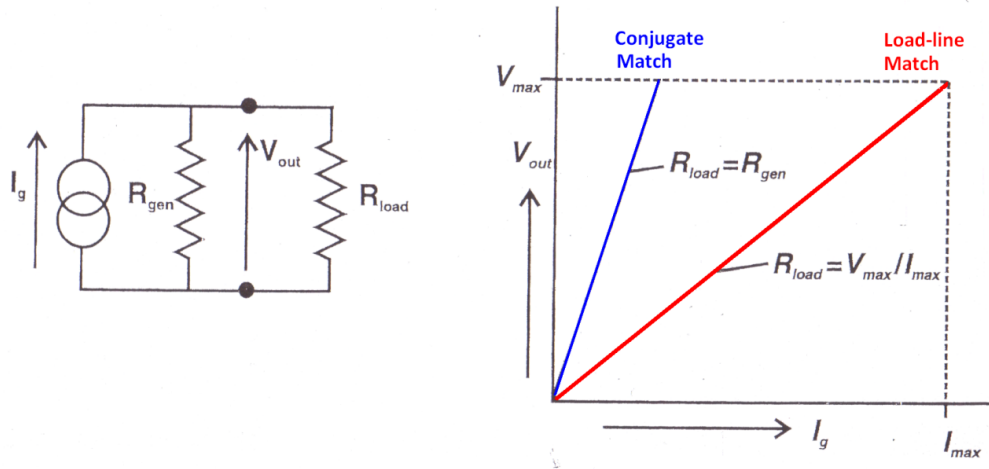
Figure 3: Conjugate Match and Load-Line Match [7]

Output power is specified and measured under two different conditions: continuous wave (CW) and pulsed operating conditions, depending on the applications. The amplifier designed in this thesis work is to be used in pulsed radars. Output power along with efficiency determines the amount of power that has to be removed by the heatsink. In pulsed conditions, the heatsink requirements are lower because the amplifier is operating with a certain duty cycle (D) and less power is dissipated from the amplifier, as shown in Equation 2 where PRI is the pulse repetition interval. When D approaches unity, Equation 2 describes the dissipated power in CW operation.

$$P_{diss,pulse} = D\left(\frac{1}{\eta}-1\right)\left(\frac{P_{out,pulse}}{PRI}\right) \tag{2}$$

2.1.2 Linearity

An amplifier is considered linear if the output power is linearly proportional to the input power, while the phase difference between the output and input should remain the same. One measure of linearity can be expressed at the output level at which the gain compresses and is often specified as the '1-dB compression point'. The 1-dB compression point is defined as the output power level at which the gain drops by 1-dB from the small signal value (or the linear response) as illustrated in the transfer characteristic of Figure 4. It should be emphasized that an amplifier operating at the 1-dB compression point is already heavily nonlinear. Generally, to achieve high linearity, amplifiers are backed-off and operated well below the 1-dB compression point; however, this reduces amplifier efficiency.
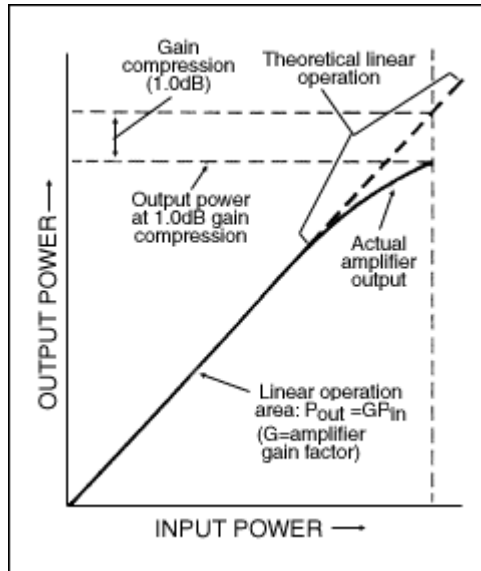
Figure 4: Linearity and Gain Compression

Non-linear amplifier response occurs in an amplifier when the output is driven to a point near compression. As the compression point is approached, the amplifier gain falls off, or compresses and the linearity decreases. The nonlinear amplifier output illustrated in Figure 4 can be expressed with the power series relationship of Equation 3.

$$V_{out}(t) = a_1 V_{in}(t) + a_2 V_{in}(t)^2 + a_3 V_{in}(t)^3 + ... \qquad (3)$$

The transfer characteristic now includes higher order terms, not only the linear term. Typically, for power amplifier (PA) transfer functions, the second-order coefficient is positive and the third-order coefficient is negative, which results in a compressive characteristic of the curve. The more the input signal grows, the larger the influence of the higher-order terms [8]. When observing the nonlinear response of an amplifier in the frequency domain, frequencies other than the fundamental operating frequency (i.e. harmonics) will be noticed as described in Equation 3.

2.1.3 Efficiency

Every amplifier requires DC power supplies that provide the ability of RF amplification. Figure 5 shows a diagram of the typical power flow in a generalized amplifier, where an input drive source is assumed to generate an RF signal at frequency f_o.
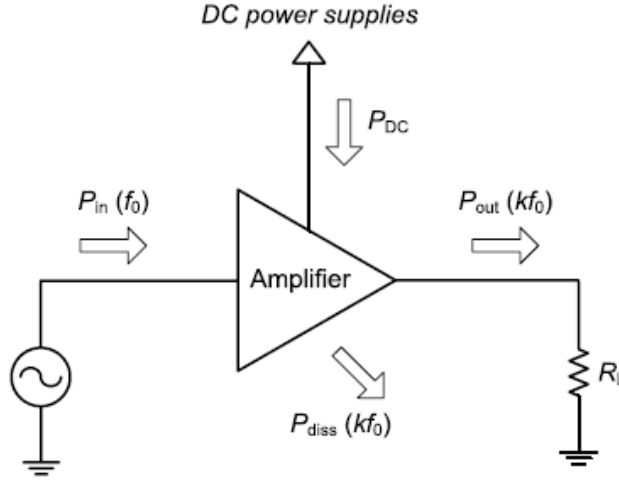
Figure 5: Power Flow in a Generalized Amplifier

According to the law of energy conservation, the total amount of power entering into an amplifier must be the same as the total power coming out of the amplifier. Due to the nonlinear operation of an amplifier, many harmonics are also generated as RF output power $P_{out}(kf_o)$.

Power amplifier (PA) efficiency is a measure of the effectiveness of converting DC power into RF power. In RF power amplifiers, there are two different measurements of efficiency, namely, drain efficiency and power-added efficiency (PAE). Drain efficiency is the ratio of output RF power to input DC power as shown in Equation 4.

$$\eta_{drain} = \frac{P_{out}(f_o)}{P_{DC}} = \frac{P_{out}(f_o)}{(V_{DC})(I_{DC})} \tag{4}$$

Drain efficiency is useful when input power level is of no primary significance, which occurs either when the gain is very high or the input drive source is assumed to generate sufficient power without extra constraints. Since the drain efficiency isolates the efficiency calculation from power loss in the input circuitry, the drain efficiency can be used as a comparison criterion for performance of different amplifier classes that are entirely determined by bias condition and output termination. Thus, drain efficiency is often used to evaluate the performance of switching-mode amplifiers, where the input drive is assumed to be sufficiently large to saturate the transistors.

The most common definition of efficiency in all types of amplifiers is power-added efficiency or PAE, in which the power produced from an amplifier is defined as the RF power "added" by an amplifier, i.e. the difference between the RF input and output at f_o as shown in Equation 5.

$$PAE = \frac{P_{out}(f_o) - P_{in}(f_o)}{P_{DC}} = \frac{P_{out}(f_o)}{P_{DC}}\left(1 - \frac{1}{G}\right) = \eta_{drain}\left(1 - \frac{1}{G}\right) \tag{5}$$

Since the RF input power is included in the "produced" power of the amplifier, this definition is not correct in a physical point of view. If the gain is below unity, the PAE could even be negative. However, the advantage of the PAE is that it combines the gain with the drain efficiency. When a PA is used within a system, the input signal is provided from the previous stage that has a common limitation on the output power level. Thus, the gain of the PA, in this case, is a critical factor to determine the efficiency of the overall system as well as of the amplifier itself. It can be seen from Equation 5 that the PAE will approach its maximum value, which is the drain efficiency, as the gain increases.

Furthermore, the output capacitance of the transistor has a large effect on the efficiency of an amplifier, since it must be charged to around twice the supply voltage and discharged again during each cycle of the operating frequency, and the power used in the charging process is dissipated in the transistor. At a single frequency, a part-but not all- of the capacitance can be tuned out, since its value varies with the output voltage swing. Equation 6 is the power loss due to the output capacitance, where C_{OSS} is the output capacitance, V_{CC} is the supply voltage, and f is frequency [6].

$$P_{loss} = \left(2 * C_{oss}\right) * \left(V_{CC}\right)^2 * f \tag{6}$$

It can be seen from Equation 6, that transistors with lower output capacitance will dissipate less power and have higher efficiencies. Also, operating transistors at lower supply voltages will significantly reduce the power loss since V_{CC} is squared.

2.1.4 Input and Output Matching

One of the most important aspects of power amplifier design is the impedance matching network for both the input and output of the amplifier. Assuming the amplifier circuit will be used in a 50-Ω system, the input and output impedance of the amplifier will need to be matched to 50 Ω. Any impedance mismatch in the source or load side leads to reduced device gain and large reflected power, which reduces the efficiency of the PA due to impedance mismatch. The reflected power also affects the reliability of the device and complicates the thermal management [9]. As explained in Section 2.1.1, the load impedance seen by the output of the transistor determines the output power and ultimately the efficiency of the amplifier. Therefore, an output matching network will be required to transform the 50-Ω system impedance to the desired load impedance, typically Zopt, according to Figure 6.
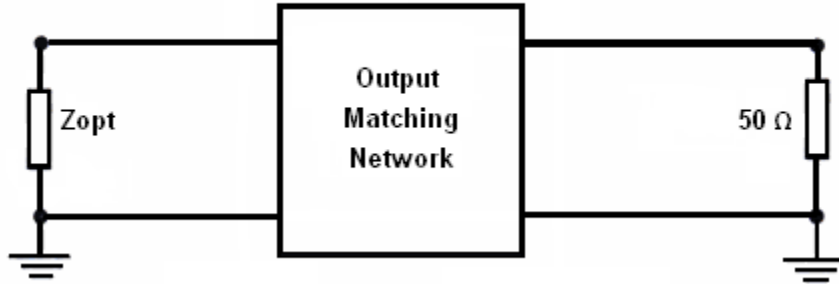
Figure 6: Output Matching Network Block Diagram

The input matching network is designed to transform the transistor's input impedance into the 50-Ω system impedance according to Figure 7. $R_{in} + jX_{in}$ represents the complex input impedance of the transistor.
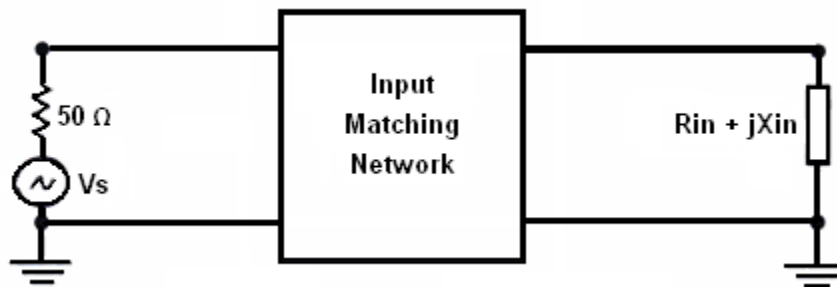
Figure 7: Input Matching Network Block Diagram

The impedance-matching network should provide matching over the complete operational frequency band. However, the challenge of designing matching networks, specifically output matching networks, is that the transistor's output impedance varies with frequency. Often times, over a decade bandwidth, the optimum impedance can drop by a factor of two. For example, if the low frequency load line is 6 Ω, the upper operating frequency could require an impedance of 3 Ω with some additional amount of inductive or capacitive reactance [13].

2.1.5 Operational Bandwidth and Frequency Response

The operational bandwidth of an amplifier represents the amount or "width" of frequencies for which the amplifier provides satisfactory performance according to several design constraints, such as gain, output power, gain flatness, efficiency, and/or VSWR. The most common measurement constraint is the half-power points (i.e. the frequency where the power drops by half its peak value) on the power vs. frequency curve. In this context, the operational bandwidth can be defined as the difference between the lower and upper half-power points, also known as the '3-dB bandwidth' as illustrated in Figure 8. Other tolerances in regards to

output power may be specified according to the 1-dB or 6-dB bandwidth, depending on the desired output power performance constraints.
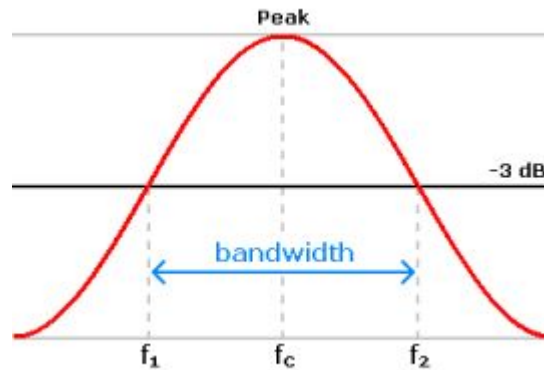
Figure 8: Definition of 3dB Bandwidth

However, the bandwidth is not the same as the band of frequencies that the amplifier is capable of amplifying according to the desired performance constraints. This range of frequencies is defined as the 'frequency response' of the amplifier and provides the lower and upper frequency limits of the amplifier. For example, according to Figure 8, the operational bandwidth would be a single value calculated as f_2 minus f_1, while the frequency response would be stated as two separate frequency limits f_1 to f_2. Thus, the amplifier operates from f_1 to f_2 with a bandwidth of f_2 minus f_1 according to the 3dB output power constraint.

2.2   *Classes of Power Amplifiers*

The two main categories of amplifiers are transconductance and switching. Transconductance amplifiers operate the transistor in the linear region (triode mode) and utilize the active resistance (i.e. transconductance) of the transistor to regulate power delivery. On the other hand, switching amplifiers operate the transistor in the saturation region (active mode) and utilize the transistor as a switch. Power amplifier circuits are further classified based upon the conduction angle, θ, of the input signal through the output of the amplifying device. That is, the angular portion of the input signal cycle during which the amplifying device conducts. The bias point (or quiescent point) of the transistor along with the amplitude of the input signal determine the conduction angle of the transistor, and thus the classification of the amplifier. The most common classifications of amplifiers are Class A, B, AB and C for transconductance amplifiers and Class D and E for switching amplifiers.

2.2.1   Class A

In a Class-A amplifier the conduction angle is 360°, thus the transistor conducts for 100% of the input signal as illustrated in Figure 9. Since the transistor is always conducting, even if there is

no input signal, power is drawn from the power supply which makes the Class-A topology inefficient. According to [7], the maximum theoretical efficiency of a Class-A amplifier is 50% assuming no losses. However, as seen in Figure 9, the Class-A amplifier is highly linear.
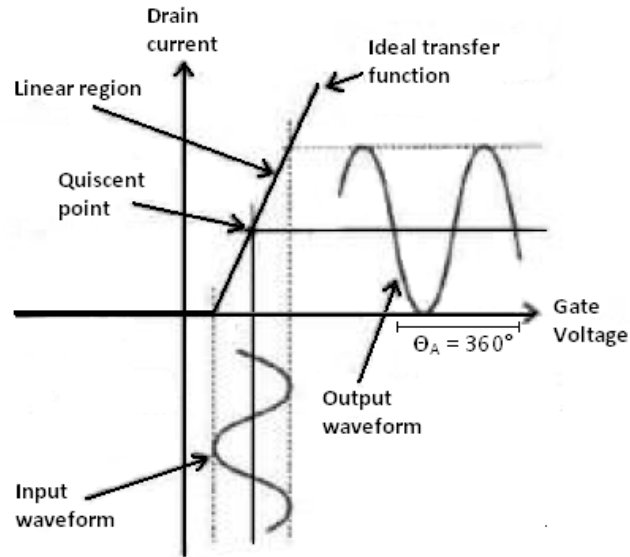
Figure 9: Class A Amplifier

2.2.2 Class B

In a Class-B amplifier the conduction angle is 180°, thus the transistor conducts for 50% of the input signal as illustrated in Figure 10. According to [7], the maximum theoretical efficiency of a Class B amplifier is 78.5% assuming no losses. However, as seen in Figure 10, the Class B amplifier is not very linear.
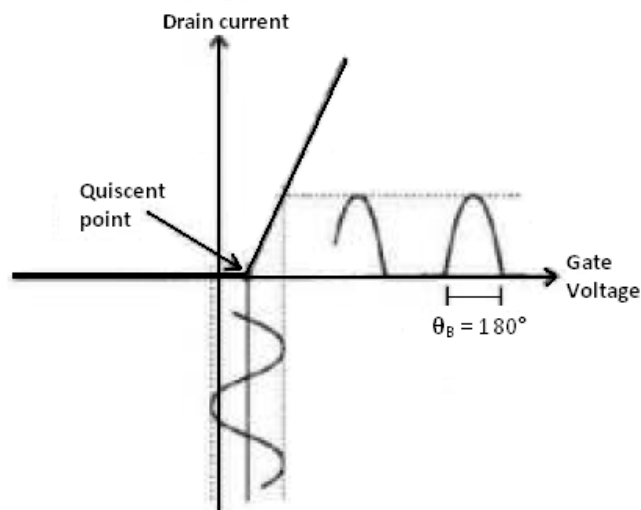
Figure 10: Class B Amplifier

2.2.3 Class AB

In a Class AB amplifier the transistor is biased somewhere between a Class A (360°) and a Class B (180°), thus the transistor conducts for 50 to 100% of the input signal as illustrated in Figure 11.
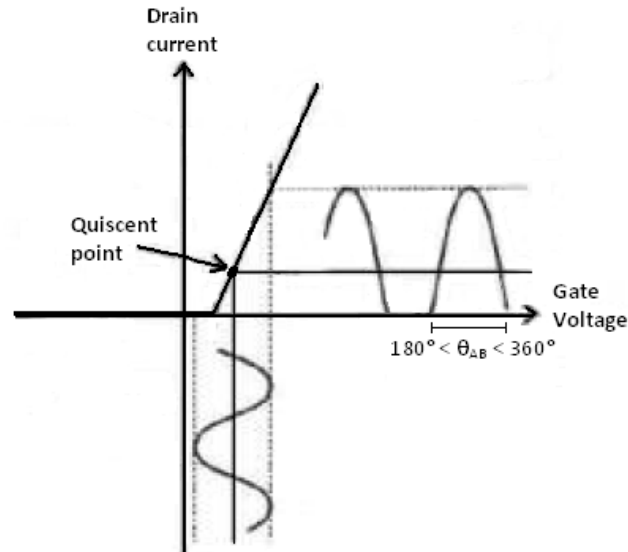
Figure 11: Class AB Amplifier

A 'weak-biased' class AB amplifier has a bias point closer to that of a class A, while a 'strong-biased' class AB amplifier is biased closer to that of a class B. The maximum theoretical efficiency of a Class AB amplifier is slightly less than 78.5%, if biased close to a Class-B amplifier with a conduction angle slightly less than 180°. However, the Class-AB amplifier is usually biased in the range between 180° and 360°; therefore, the efficiency range would fall somewhere between 50% (for Class A) and 78.5% (for Class B). Typically, the efficiency is around 45-60% for Class AB amplifiers. Overall, the linearity is moderate and depends on the bias point of the transistor.

2.2.4 Class AB Push-Pull

A Class-AB Push-Pull amplifier is biased similar to a Class-AB amplifier; however, a complementary pair of transistors is used to create a 'pushing' and 'pulling' effect by amplifying opposite halves of the input signal, which is then recombined at the output. Each transistor is biased to have a conduction angle slightly greater than 50%, thus the amplifier circuit conducts for 100% of the input signal as illustrated in Figure 12. The threshold voltage of the transistor must be overcome before conduction can take place; thus, as shown in Figure 12, there is a small mismatch at the 'joins' between the two halves of the output signal, which creates crossover distortion and degrades linearity. Thus, to remove the crossover distortion and improve the linearity, the bias point of each transistor is 'tuned' so the conduction voltage is overcome, but not by the input signal. Typically, the efficiency is similar to that of a single transistor Class AB amplifier (45-60%) but with an improvement in linearity.
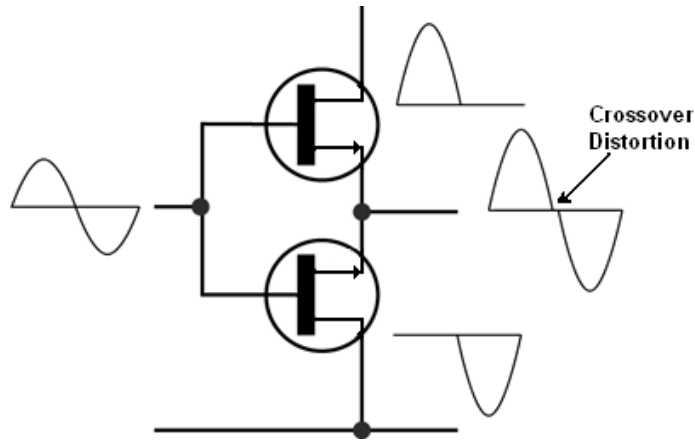
Figure 12: Class AB Push-Pull Amplifier

2.2.5 Class C

In a Class-C amplifier the conduction angle is less than 180°, thus the transistor conducts for less than 50% of the input signal as illustrated in Figure 13. According to [7] a Class-C amplifier can attain a theoretical efficiency as high as 100% when the conduction angle is 0°; however, the linearity suffers tremendously.
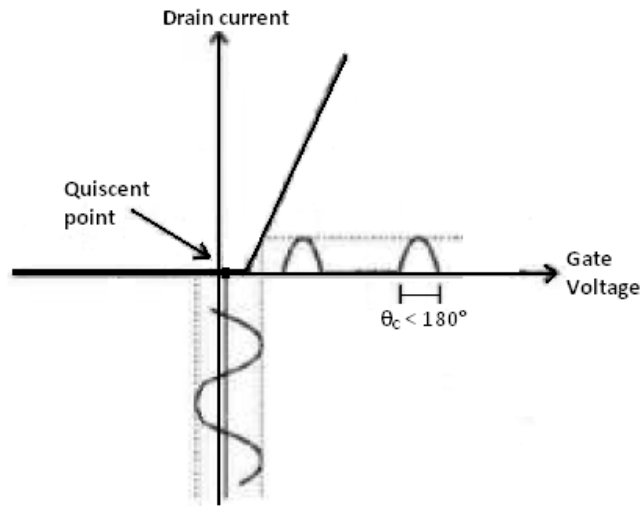
Figure 13: Class-C Amplifier

2.2.6 Class D and E

In a Class-D amplifier two transistors are used in a push-pull configuration so that each transistor can be alternatively switched on and off. The input signal is converted to a sequence of higher voltage output pulses by comparing a triangular waveform with the input signal waveform, thus creating a pulse-width modulated output waveform. The averaged-over-time power values of these pulses are directly proportional to the instantaneous amplitude of the input signal. Furthermore, the output pulses contain inaccurate spectral components (namely,

the pulse frequency and its harmonics) which must be removed by a low-pass passive filter. Figure 14 shows the basic block diagram of a Class D amplifier. Class D amplifiers are highly efficient, but not commonly used in RF amplifiers since the frequency of the output pulses is typically ten or more times the highest frequency in the input signal to be amplified. Also, Class D amplifiers are not very linear.
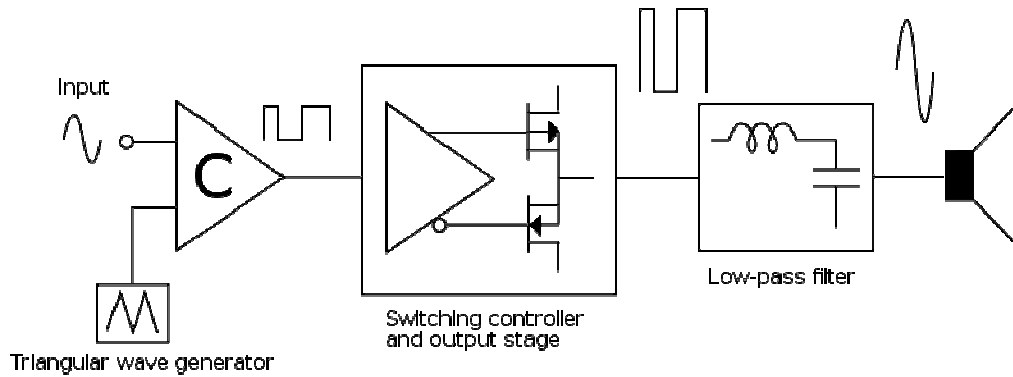
Figure 14: Class D Amplifier Block Diagram

In a Class E amplifier, shown in Figure 15, the transistor operates as an on/off switch and the load network shapes the voltage and current waveforms to prevent simultaneous high voltage and high current in the transistor minimizing power dissipation. According to [7] a Class E amplifier can attain a theoretical efficiency of 100% assuming no losses; however, typically efficiencies range from 85 to 90%. However, the resonant load network limits the operational bandwidth. Due to the switching nature, Class E amplifiers are also not very linear.
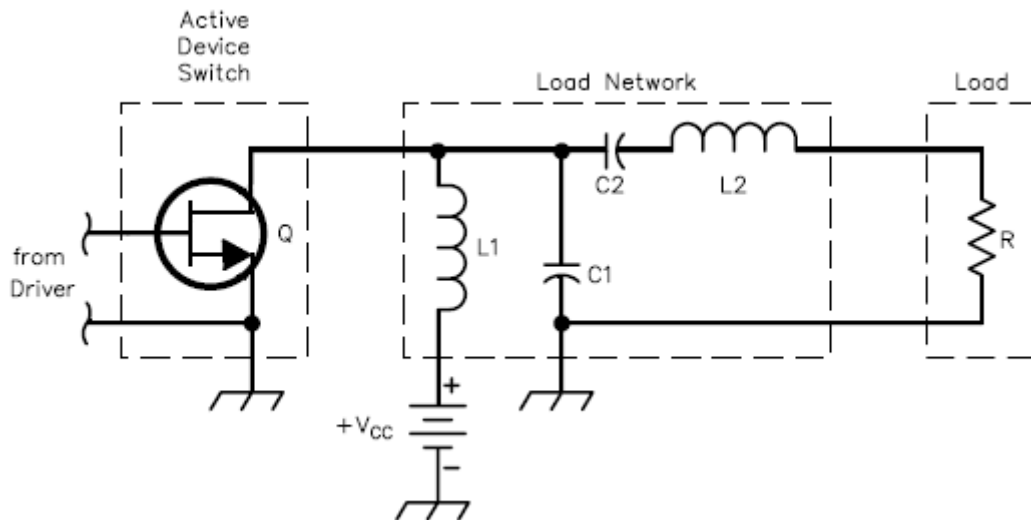
Figure 15: Class E Amplifier [10]

2.3 Technical Requirements for Power Amplifiers Used in Pulsed VHF Radar Systems for Ice Depth Sounding

2.3.1 Requirements and Desired Features

According to the radar range equation shown in Equation 7, the ratio of the transmitted power of a radar signal to the range is P_T/R^4. Assuming all other variables in the radar range equation remain constant, it can be seen that depth sounding radar systems will require large amounts of power to overcome the spherical spreading loss (R^4) and path loss (L) due to ice attenuation in order to satisfy the radar range equation and provide an adequate Signal-to-Noise Ratio (SNR) at the receiver.

$$SNR = \frac{P_R}{P_N} = \frac{P_T G^2 \lambda^2 \sigma}{\left(4\pi^3\right) R^4 k T_0 B F_n L} \tag{7}$$

Although it may appear a relatively simple solution to simply increase the transmitter power (i.e. the PA output power) as required in order to achieve the desired SNR, there are several limitations to this solution.

For depth sounding applications, the transmitter PA needs to be linear to reduce the side-lobe levels of the transmitted pulse. The CReSIS depth sounding radars use a pulse compressed, linear frequency modulated (FM) chirp signal with a pulse width between 1 and 10 μs, which, when transmitted with low time-bandwidth products creates mid-band ripples in the amplitude spectrum [11]. These mid-band ripples, known as Fresnel ripples, create range side-lobes which are typically reduced by applying an amplitude tapering function, typically a Tukey window. However the efficiency of the windowing is based on the concept of shaping a perfectly rectangular spectrum; thus, range side-lobes will be generated when there are deviations from the ideal rectangular spectrum amplitude and quadratic phase [12]. These range side-lobes can mask weak returns which could represent internal layers or surface clutter depending on the application.

As an illustration of this concept, a 140 to 160 MHz linear FM chirp with a Tukey weighting was generated using the AD9910 DDS of the 1U-DAQ digital system designed by CReSIS, then amplified using the 50 W Polyfet power amplifier module pairs (MADQ06 & MBDQ01), which are the current PAs used by CReSIS for many depth-sounding applications. After attenuating the signal by 50 dB in order to protect the receiver, the data were then recorded using the AD9640 analog-to-digital converter (ADC) of the 1U-DAQ and processed in MATLAB. As seen in Figure 16, the DDS generated pulse has about a 0.35 dB amplitude deviation, even with a Tukey window while the PA amplitude deviation is about 3 dB with a Tukey window as seen in Figure 17.
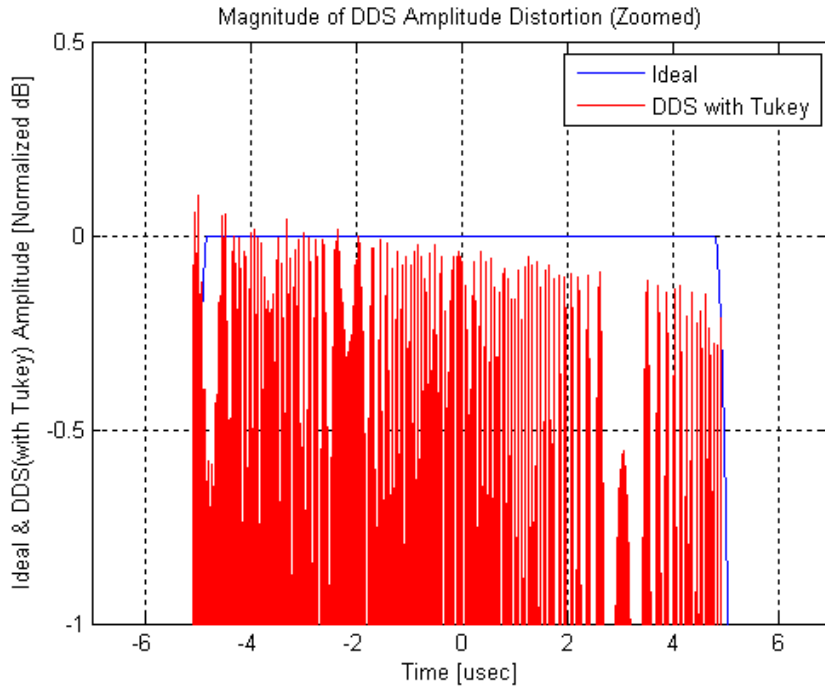
Figure 16: DDS Transmit Pulse Amplitude Distortion (with Tukey Window)
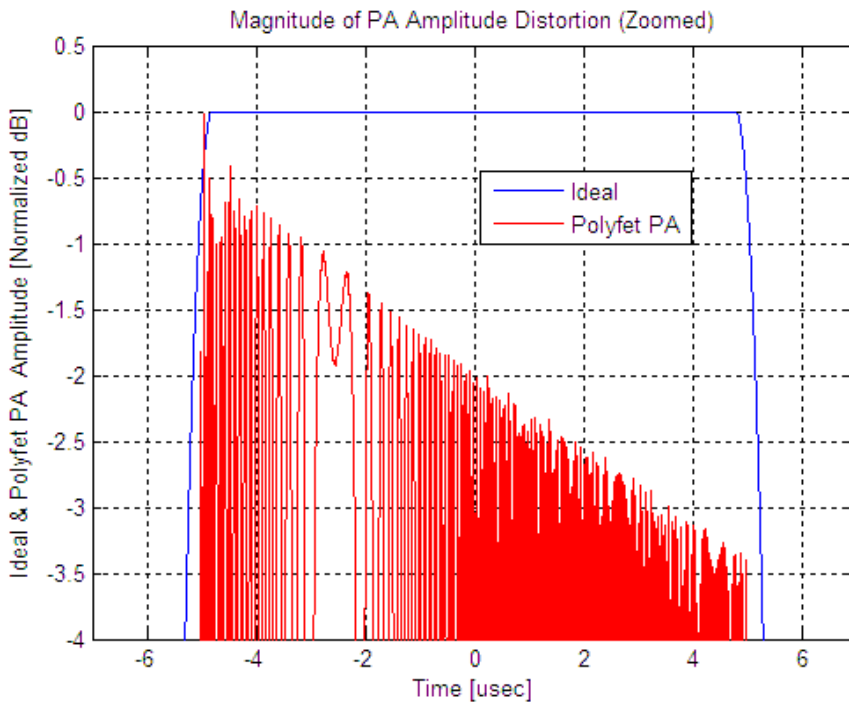
Figure 17: Polyfet 50W Transmit Pulse Amplitude Distortion (with Tukey Window)

Assuming the return signals will be attenuated versions of the PA transmitted pulse of Figure 17, a strong return (representing the bedrock) and a delayed weak return (representing an internal

layer) were simulated and filtered using a double-Blackmann (or Blackmann^2) window as shown in Figure 18. The delayed weak return has amplitude of -55 dB relative to the bedrock return.

Finally, a cross-correlation was performed for each return signal to simulate the pulse compression response as shown in Figure 19. From the simulated results, it can be seen that when using the Polyfet 50 W PA a weak return delayed by 0.5 µs with amplitude -55 dB relative to the bedrock return can be masked by the side-lobes when there is about a 3 dB variation in amplitude of the transmitted pulse even when windowing techniques are applied to the transmitted and received signals. Furthermore, the phase non-linearities of the PA also contribute to the increased side-lobe levels, which were not illustrated in this example. Therefore, linearization can be implemented to correct the amplitude & phase deviations in the transmit pulse resulting in a decrease in side-lobe levels.
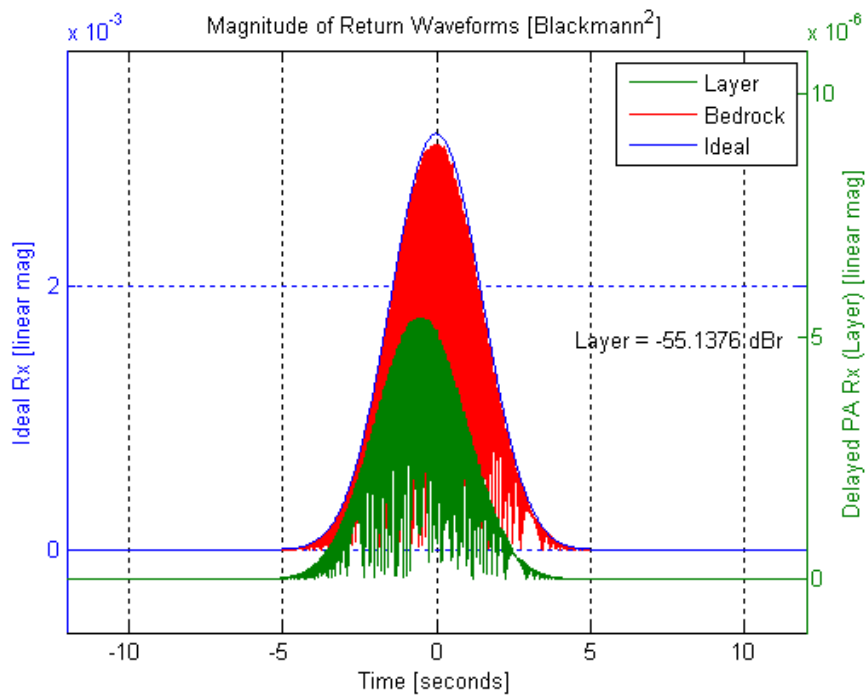
Figure 18: Polyfet 50W Simulated Returns (with Blackmann^2 Window)
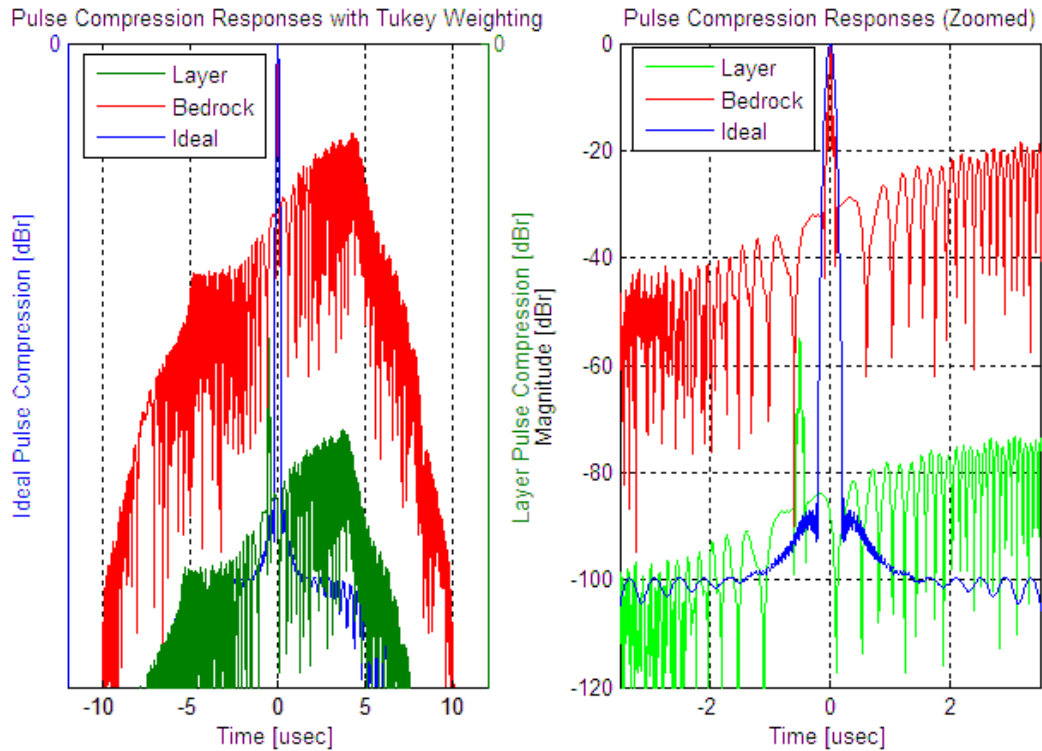
Figure 19: Polyfet 50W Simulated Pulse Compression Returns

The drawback of implementing a highly linear PA, as explained in Section 2.2, is that linear PAs dissipate large amounts of heat depending on the class of amplification. Typically, the heat is dissipated using large heat sinks; however, this conflicts with the weight and size limitation especially in the UAV application. Due to aerodynamic limitations on the antennas, the PA must also meet a certain size constraint. Moreover, since increasing weight equates to increasing fuel consumption, the weight of the T/R module is also limited. To overcome the size and weight limitations and at the same time maintain a high output power without sacrificing linearity, it was determined that the DC power to the PA could be pulsed. Given that the higher linearity amplifier classes consume power even when the PA is not transmitting, pulsing the DC power closely 'in sync' with the transmitted pulse decreases the dissipated power as explained in Section 2.1.1.

2.3.2 Topology and Class Selection

In order to maintain linearity at lower power levels, the pre-amplifier was biased as a Class A amplifier and the driver amplifier was biased as a 'weakly-biased' Class AB amplifier. In order to compromise between linearity (Class A) and efficiency (Class B), the Class AB push-pull amplifier configuration was selected for the main power amplifier stage of this design.

Chapter 3: Overview of Power Amplifier Linearization Techniques

The requirement of achieving high linearity and high efficiency in a power amplifier has motivated a great deal of research into linearization techniques. The basic idea is to operate the PA as close to saturation as possible to maximize the power efficiency, and then employ some linearization technique to suppress the distortion introduced in this near-saturated region. Understanding the nonlinear behavior of PAs is critical to applying an external linearization technique. The power series expression in Equation 8 can be used to describe the nonlinear effects in PAs.

$$V_{out}(t) = a_1 V_{in}(t) + a_2 V_{in}(t)^2 + a_3 V_{in}(t)^3 + ... \tag{8}$$

The linear small signal gain is represented by a_1, while a_2 and a_3 are the gain constants for the quadratic and cubic nonlinearities, respectively. The second-order coefficient is positive and the third-order coefficient is negative, which result in a compressive characteristic. If the coefficients are real valued, the system is considered nonlinear and memoryless. If the coefficients are complex valued, indicating a constant, frequency-independent phase shift, the system is considered nonlinear with memory. It should be noted that there is a difference between a system with 'memory' and 'memory effects'. Memory effects refer to bandwidth-dependant nonlinear effects often present in PAs such as envelope memory effects and frequency memory effects. Envelope memory effects are primarily a result of thermal hysteresis, and electrical properties inherent to PAs. Frequency memory effects are due to the variations in the frequency spacing of the transmitted signal. Memory effects were omitted from this research work and are a topic for future work.

When viewing the nonlinear response of the amplifier in the frequency domain, multiples (or harmonics) of the original frequency will be present. This is referred to as amplitude-to-amplitude (AM/AM) distortion, since the output amplitude will be distorted in relation to the input amplitude. As explained in Section 2.3.1, these nonlinearities degrade radar system performance by increasing side-lobe levels.

Through the implementation of linearization, these nonlinearities can be reduced, thus reducing side-lobe levels. This chapter provides a brief overview of some of the more common linearization techniques.

3.1   Power Back-off

Power back-off exploits the observation of small signal analysis, that is, that any amplifier appears linear for sufficiently small departures from its bias condition. Mathematically, this principle can be explained according to Equation 9 which is a polynomial similar to Equation 8, where I_{out} is the total output current, I_o is the bias current and V_{in} is the signal input voltage.

$$I_{out} = I_0 + a_1 V_{in} + a^2 V_{in}^2 + ... \tag{9}$$

It can be seen that by decreasing V_{in}, the linear term can be made to dominate all but the DC term I_o, which for RF power amplifiers can be easily blocked. Power back-off is the simplest linearization technique to implement; however, even though linearity is improved, the efficiency is decreased due to the reduced input signal level. Thus, power back-off does not satisfy the increased linearity and efficiency requirement.

3.2   Feed-back

Feed-back linearization is a well-known and established method of improving linearity in power amplifiers. A number of different configurations have been created, such as RF feedback, envelope feedback and Cartesian feedback. The simplest feed-back technique is RF feedback, as shown in Figure 20, which can be implemented as passive or active feed-back. Unfortunately, the electrical delays around the feed-back loop restrict the bandwidth of signals that can be linearized; thus, the fundamental limitation of any of the RF feed-back techniques is phase delay around the control loop resulting in narrow bandwidth linearization.
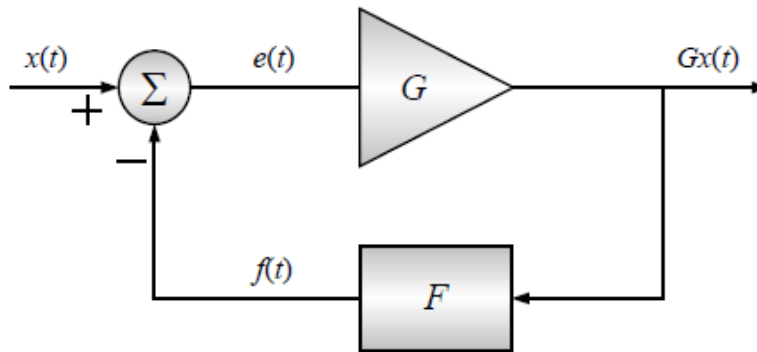
Figure 20: Feed-back Linearization

3.3   Feed-forward

Feed-forward linearization uses an additional amplifier to manipulate the output from the power amplifier as shown in Figure 21. With the PA output reduced to the same level as the input, the difference between the input and the reduced level output signal is only the distortion generated by the PA. Further, if this resulting distortion is then amplified with a different, highly linear amplifier and then subtracted from the original output, theoretically, only the linear amplified portion of the input signal should remain. Feed-forward linearization utilizes two circuits – an input signal cancellation circuit and a distortion- (or error-) cancellation circuit. The signal-cancellation circuit suppresses the input reference from the output of the main PA, leaving only its linear and nonlinear distortion components in an "error" signal. The error-cancellation circuit is used to suppress the distortion component of the PA output, leaving only the linear-amplified component of the output signal. In order to suppress the distortion component of the signal, the gain of the amplifier used in the error-cancellation circuit must be carefully chosen to match the sum of the effects of the sampling coupler, the fixed attenuator,

and the output coupler. Thus, the error signal is amplified to approximately the same level as the distortion component in the PA output signal [14]. Feed-forward generally provides good linearity, but it results in poor efficiency.
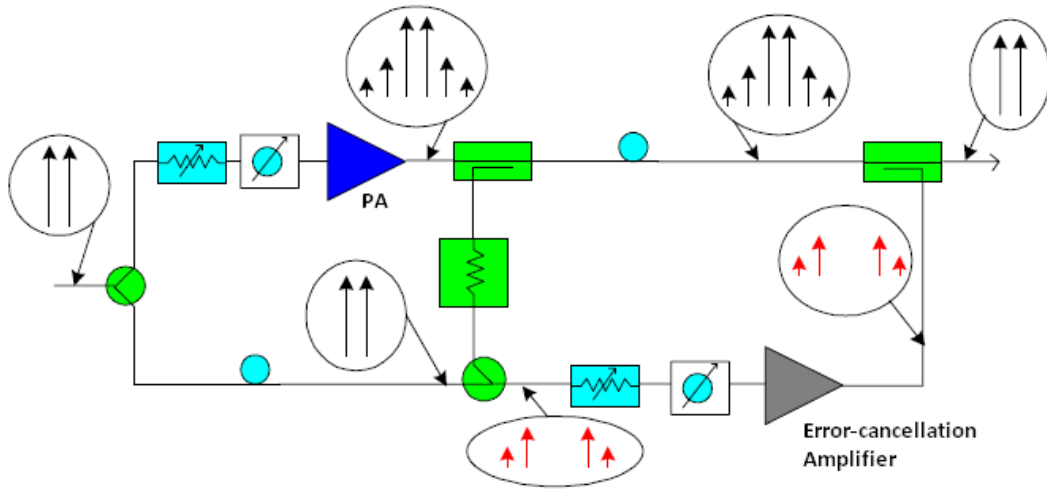
Figure 21: Feed-forward Linearization

3.4   Predistortion

Predistortion (PD) involves constructing a predistorter which has the inverse nonlinear characteristics of the PA. Therefore, when the predistorter's output signal is passed through the PA, the distortion components cancel and only the linear components remain as depicted in Figure 22. Since linearization is performed at the input of the PA, loss of efficiency is negligible. Predistortion techniques can be classified as either analog PD or digital PD. The PA response is assumed to be a memory-less model, that is only dependent upon the input signal magnitude. This simplification reduces the complexity of compensating for memory effects which are a result of the operating frequency band and instantaneous operating temperature.
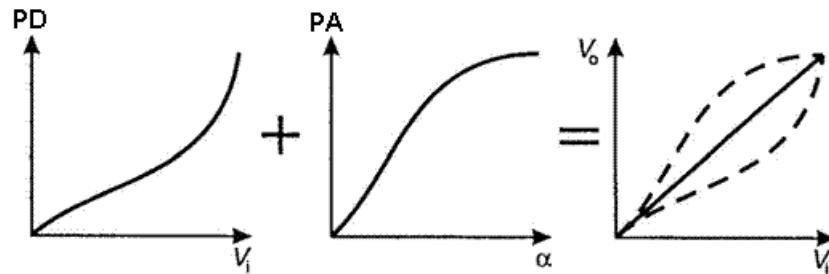
Figure 22: Predistortion Block Diagram

3.4.1 Analog Predistortion

An analog predistorter utilizes a nonlinear device (typically a diode or transistor) to predistort the input signal. The type of analog PD used is dependent on the nonlinearities generated by

the PA. Analog PDs can be constructed as a Square Law or Cubic Law devices or any combination of these two configurations for higher-order nonlinearities. An analog PD generally has two paths. One carries the fundamental components and the other is the distortion generator. The distortion generator's purpose is to eliminate the fundamental component; thereby, providing independent control of the distortion relative to the fundamental component. The two paths are time-aligned and then subsequently combined before being presented to the PA.

One of the simplest and least expensive analog PD techniques is the 'series diode' technique which consists of a nonlinear diode being placed in front of the PA as illustrated in Figure 23.
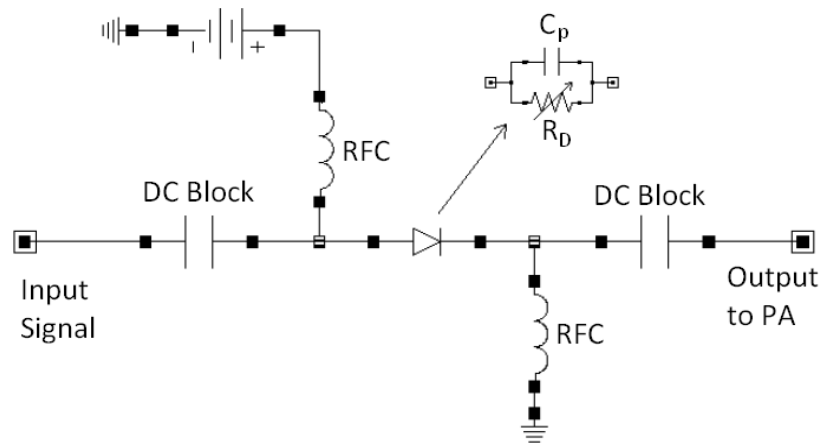
Figure 23: Series Diode Predistortion Linearizer

The diode functions as a nonlinear resistor (R_D) with a parasitic capacitance (C_p) in parallel. The resistance and capacitance is used to form a nonlinear RC phase shift network. The diode is forward biased to set R_D and C_p at an initial small-signal operating point. As the diode is driven past small signal conditions with RF power, the diode rectifies the RF power and the operating point changes with increasing input power. Effectively, R_D decreases with increasing input power since its operating point is moved up the I/V curve. R_D changes nonlinearly due to the diode's I/V characteristic, resulting in a nonlinear phase shift with increasing input power. The series diode technique provides about 6 dB improvement in linearity [15].

A more complex configuration utilizes transistors, diodes, and a parallel RC circuit to remove the 3rd, 5th and higher order nonlinearities. Figure 24 shows the block diagram of the 3rd-order intermodulation signal generator (IMG$_3$).
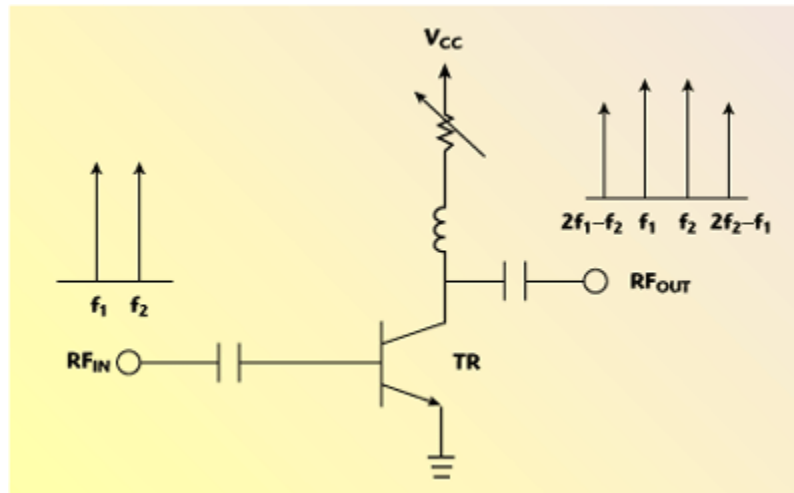
Figure 24: Intermodulation Signal Generator (IMG3) Block Diagram

Its output signal contains the fundamental and 3rd-order components. The bias voltage of the small signal transistor is controlled to generate a 3rd-order component as large as possible. Figure 25 shows the block diagram of the higher order intermodulation (IM) signals generator (IMG_H).
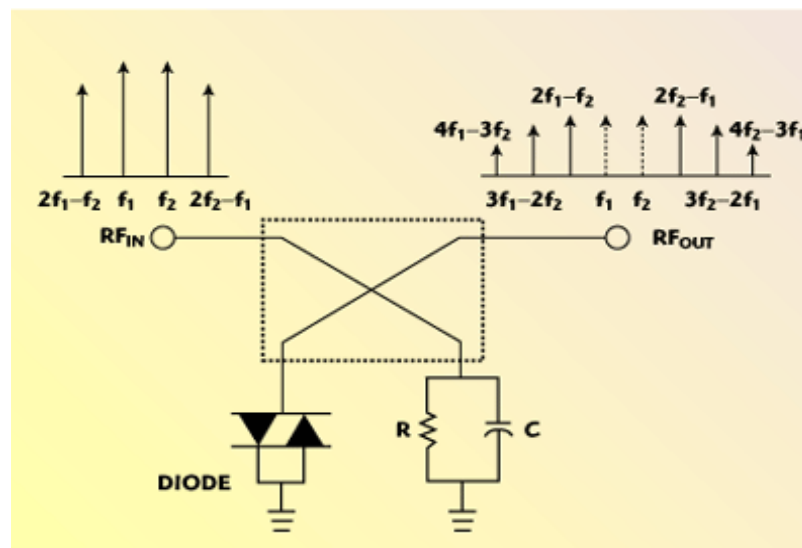
Figure 25: Intermodulation Signal Generator (IMGH) Block Diagram

IMG_H consists of a 3-dB hybrid coupler, two anti-parallel Schottky diodes for high order IM signal generation, and a resistor and capacitor to control amplitude and phase of the input signal to generate only the desired high order IM components. Figure 26 portrays the complete analog predistortion circuit using an automatic level controller (ALC) circuit, and IMG_3 and IMG_3 control block, an IMG_H and IMG_H control block and a high power amplifier (HPA). The IMG_3 and IMG_H control blocks consist of a variable attenuator and a variable phase shifter to control the magnitude and phase of the 3rd and higher order inverse IM signals to the HPA. The delay circuit

in the PA path compensates for the delay in the IM generators and IM control circuits. This complex analog predistortion circuit provides about 8 to 10 dB improvement in linearity [16].
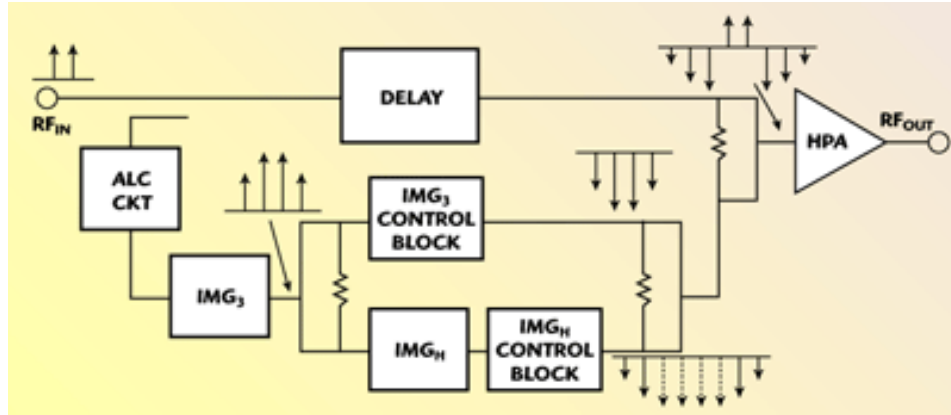
Figure 26: 3rd and Higher Order Analog Predistortion Block Diagram

3.4.2   Digital Predistortion

Digital Predistortion (DP) utilizes digital signal processing (DSP) techniques to predistort the input signal. The cascade of the DP response and the PA response produces the desired linear response. If the PA is operating in compression, the P_{out} versus P_{in} curve falls below the 'linear output' curve, and the actual output power of the PA is not sufficient for linear operation as illustrated in Figure 27. When the amplifier is operating in compression, the P_{out} versus P_{in} curve falls below the Linear Response curve, hence, the actual power of the PA is not sufficient for linear operation. The inclusion of digital predistortion prior to the power amplification has the effect of introducing expansion (i.e. the amplitude of the signal is increased so that the desired output power, on the Linear Output curve, is achieved. The expansion effect of the digital predistortion can be observed in Figure 27 where the input power, P_{in} (resulting in P_{out} before PD), is increased to P_{in-pd} so that the PA output power is raised to P_{out-pd} which coincides with the Linear Output curve. The output power cannot be expanded beyond the saturation level, thus, intersection of the Linear Output curve with the P_{sat} level determines the maximum input amplitude that will result in linear amplification [17].
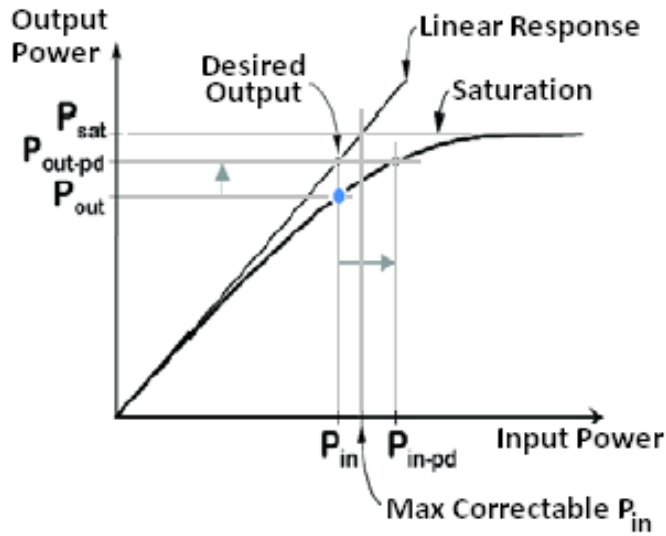
Figure 27: Power Amplifier P_{out} vs. P_{in} Curve and Digital Predistortion [17]

The phase response of a PA is also non-linear; therefore, DP is also used to linearize the typical non-linear phase response as depicted in Figure 28.
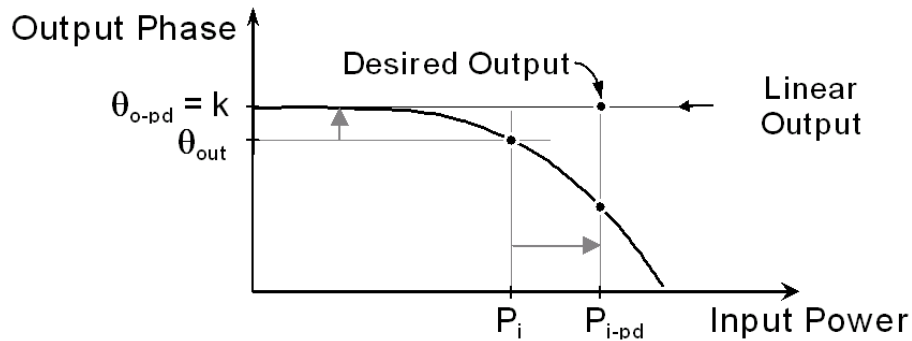
Figure 28: Linearization of Power Amplifier Phase Response [17]

The two most common DP techniques are the Vector (or Polar) mapping look-up table (LUT) approach and the Complex gain look-up table approach. The Vector mapping technique stores a compensation vector into the LUT for each input signal vector. This approach tends to require a large amount of data storage. The complex gain approach generates an inverse nonlinearity of the PA characteristic and stores the values in a LUT. This approach requires less LUT entries and can provide similar linearity improvement if the nonlinearity created by the PA is minimal at low levels of intermodulation. In either case, for adaptive predistortion the LUT is indexed by either magnitude or power. The resultant error signal generated by subtracting the PA output from the input signal is used to optimize the LUT entries. An adaptive delay is used to properly align the two signals [18].

3.5 Significance of Implementing Digital Predistortion (DP)

The challenge of implementing a linearization technique arises due to the linear FM chirp signal which increases the complexity of the linearization. Amplitude and phase corrections will be needed at various frequencies across the operational frequency range of the PA (i.e. 140 to 160 MHz or 180 to 210 MHz). The digital predistortion technique provides the most flexibility and the simplest method for implementing amplitude and phase compensation at various frequencies. The only drawback is that a digital system must be added to the PA to provide the linearization. Fortunately, the CReSIS radar systems currently utilize digital systems to generate the transmit waveforms; therefore, digital predistortion is the best choice for linearization.

Chapter 4: Implementation of Memoryless Digital Predistortion

As explained in Chapter 3, memoryless digital predistortion (DP) would only correct for the nonlinear amplitude variations since the coefficients are real valued. If the coefficients are complex valued, indicating a constant, frequency-independent phase shift, the system is considered nonlinear with memory. As illustrated in Section 2.3.1, the depth-sounding radar systems produce a nonlinear response with memory. The process for implementing nonlinear DP with memory is explained in Section 4.1.1; however, only memoryless DP was implemented in this thesis work leaving nonlinear DP with memory for future study.

4.1 Predistortion Algorithm Simulation and Synthesis

A system level block diagram of the digital predistortion system is shown in Figure 29. The Complex Gain Adjustment, Look-Up Table (LUT) and DAC are implemented within the DDS and the Adaptation Algorithm is implemented using MATLAB. The analog-to-digital converter (ADC) is the AD9640.
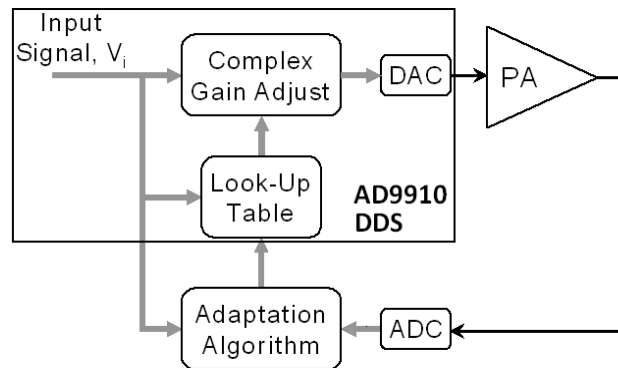
Figure 29: System Level Block Diagram of Digital Predistortion System

4.1.1 Direct Digital Synthesizer (AD9910 DDS)

The Input Signal (V_i) is generated using the Analog Devices AD9910 Direct Digital Synthesizer (DDS) which has a 1 GSPS, 14-bit DAC capable of generating an analog output up to 400 MHz.

The parameters of the reference signal (frequency, phase, and amplitude) are applied to the DDS at its frequency, phase offset, and amplitude control bits as shown in Figure 30.
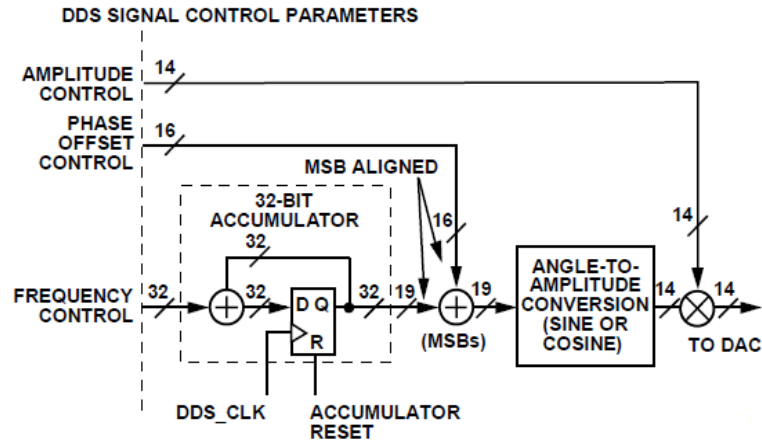
Figure 30: AD9910 DDS Block Diagram [19]

The output frequency is generated using a digital ramp generator (DRG) in digital ramp modulation mode where the output of the DRG is a 32-bit unsigned data bus that can be routed to any one of the DDS signal control parameters according to the two digital ramp destination bits in Control Function Register 2 (CRF2). To utilize the DRG to control frequency, the digital ramp destination bits of CRF2 should be set to '00' (according to Table 11 of the AD9910 datasheet) where the 32-bit output bus is MSB-aligned with the 32-bit frequency parameter.

For nonlinear predistortion with memory (i.e. amplitude and phase corrections), the amplitude and phase is generated using a polar modulation format in RAM modulation mode. The polar modulation mode partitions each 32-bit RAM sample into a magnitude (14 bits) and phase component (16 bits) with the first 2 bits being ignored. The 32-bit words output by the RAM during playback route to the DDS signal control parameters according to two RAM playback destination bits in Control Function Register 1 (CFR1). By setting the RAM playback destination bits in CFR1 to '11', the AD9910 is capable of generating polar signals, where the phase and amplitude can be independently generated from RAM (according to Table 12 of the AD9910 datasheet). The complex signal is represented by a 32-bit word in each RAM location which is segmented with bits 31 to 16 representing the phase, bits 15 to 2 representing the amplitude and bits 0 and 1 are ignored.

For memoryless predistortion (i.e. amplitude correction only), the amplitude corrections are generated using the amplitude modulation format in 'RAM modulation' mode. The amplitude mode partitions each 32-bit RAM sample into a 14-bit magnitude with the remaining 18 bits being ignored. By setting the RAM playback destination bits in CFR1 to '10', the AD9910 is capable of generating amplitude modulated signals (according to Table 12 of the AD9910 datasheet). Since memoryless predistortion was implemented in this thesis work, the RAM

playback destination bits in CFR1 were set to '10' as seen highlighted in blue in the IDL code under the section heading 'ROUTINE TO GENERATE DEFAULT DDS' in Appendix A.

The RAM playback mode is selected via the 3-bit RAM mode control word located in each of the RAM profile registers (bits 2 to 0). To select the direct switch mode, the RAM profile mode control word needs to be set to '000','101','110', or '111' (according to Table 13 of the AD9910 datasheet). The direct switch mode is used to route a single 32-bit word of data (which contains both amplitude and phase values in polar modulation mode or amplitude values only for amplitude modulation mode) from the RAM to the DDS.

The relative phase of the DDS signal can be digitally controlled by means of a 16-bit phase offset word (POW). As seen in Figure 30, the phase offset is applied prior to the angle-to-amplitude conversion block internal to the DDS core. The POW needed to develop an arbitrary Δθ is given in Equation 10, where Δθ is in radians.

$$POW = round\left(2^{16}\left(\frac{\Delta\theta}{2\pi} \right) \right)\quad [\Delta\theta\ in\ radians] \tag{10}$$

The relative amplitude of the DDS signal can be digitally scaled (relative to full scale) by means of a 14-bit amplitude scale factor (ASF). As seen in Figure 30, the amplitude scale value is applied at the output of the angle-to-amplitude conversion block internal to the DDS core. The ASF value needed for a particular amplitude scale factor is given in Equation 11, where the amplitude scale is expressed as a fraction of the full scale amplitude.

$$ASF = round\left(2^{14} * Amplitude\ Scale \right) \tag{11}$$

It is important to note that when the AD9910 is programmed to modulate any of the DDS signal control parameters, the maximum modulation sample rate is ¼ the system clock frequency. This means that the modulation signal exhibits images at multiples of ¼ the system clock frequency [19].

4.1.2 Adaptation Algorithm for Memoryless Digital Predistortion

The function of the adaptation algorithm is to calculate the complex gain adjustment required to linearize the amplifier response. The adaptation algorithm can be implemented using MATLAB which generates the 'amp.txt' file containing the amplitude values and the 'phase.txt' file containing the phase values. These values can then be programmed into the DDS as explained in Section 4.1.1. For memoryless DP, only the amplitude is adjusted; thus, the adaptation algorithm developed only generates the 'amp.txt' file. The DDS was programmed using Interactive Data Language (IDL) by altering the 'dds_gui.pro' program developed by John Ledford and Dr. Carl Leuschen. The new predistortion program is called 'dds_gui_pd.pro' which was used to generate the transmit waveform including the tapering function (i.e. Tukey) at the maximum amplitude weighting of 65,535, as well as, the predistorted waveform with Tukey weighting.

The measurement setup of Figure 31 was used to determine the amplitude weighting required to linearize the PA. The 1U-DAQ was used as the digital system which contains the AD9910 DDS for generating the transmit signal and the AD9640 analog-to-digital converter (ADC) for recording the receive signal. The DDS weighting was set to the maximum weighting of 65,535 with a Tukey tapering by selecting 'tukey' in the 'window' drop-down menu and inserting '65,535' in the 'weight' box of the DDS GUI. The output voltage of the PA was then displayed in the time domain and recorded (after the attenuator) using the AD9640 ADC and the 'daq_gui.pro' program developed by Dr. Carl Leuschen.   The attenuator transfer response was assumed to be linear for all measurements and calculations. The MATLAB program 'DDS_Predistortion.m', which contains the adaptation algorithm, was then used to generate the 'amp.txt' file. Then the DDS tapering in the 'window' drop-down menu was changed from 'tukey' to 'predistort' which reads the data from the 'amp.txt' file, applies a tukey weighting and directly stores the new amplitude values into their respective RAM locations as described in Section 4.2.1. The code for the 'dds_gui_pd.pro' IDL program is provided in Appendix A with the modifications from the 'dds_gui.pro' IDL program shown in red. The 'DDS_Predistortion.m' MATLAB program can also be found in Appendix A.
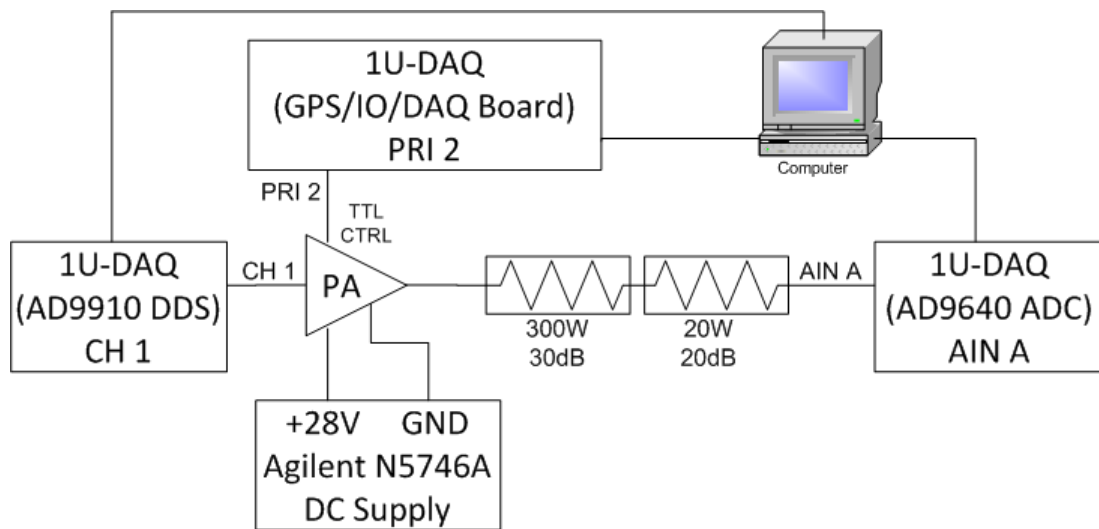
Figure 31: Predistortion Linearization Measurement Setup

Chapter 5: Design of a Power Amplifier for a Compact Transmitter Module for an Uninhabited Air Vehicle (UAV) VHF Depth Sounder Radar

5.1 Design Constraints

5.1.1 Input and Output Power

The maximum input power is +4.5 dBm (2.82 mW) due to the power output limitation of the Direct Digital Synthesizer (DDS). The output power should be between 47 to 50 dBm (50 to 100 W).

5.1.2 DC Power Supply

A single +28 V supply is available for DC power.

5.1.3 Efficiency

The power-added efficiency (PAE) should be greater than 30%.

5.1.4 Frequency Response and Bandwidth

The frequency response should be 140 to 210 MHz with a bandwidth of 70 MHz.

5.1.5 Volume and Weight

The size should be less than 3 ¾" x 3 ¾" x 1" (L x W x H) with a weight less than 16 ounces.

5.1.6 Amplifier Turn-On / Turn-Off Time

The amplifier should turn on within 500 nS of the rising edge of the RF control pulse and turn off within 500 nS of the falling edge of the RF control pulse.

5.2 Design Description and Simulations

5.2.1 Design Overview

A 3-stage power amplifier was designed as shown in the block diagram of Figure 32.
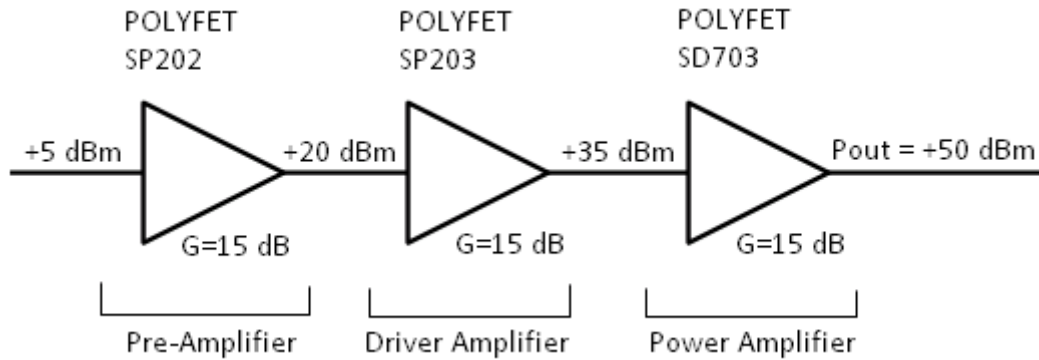
Figure 32: Power Amplifier Block Diagram

The pre-amplifier stage is a Class A amplifier designed using the Polyfet SP202 single-ended RF power transistor. The driver amplifier stage is a 'weak-biased' class AB amplifier designed using the Polyfet SP203 single-ended RF power transistor. The bias point is set just outside the bounds of a class A to reduce current consumption while maintaining moderate linearity. The power amplifier stage is a 'strong-biased', class AB, push-pull amplifier designed using the Polyfet SD703 push-pull RF power transistor. The specifications for the SP202, SP203 and SD703 transistors are summarized in Table 1 [20, 21, 22].

Table 1: Polyfet SP202, SP203 and SD703 Transistor Specifications

	Freq [MHz]	Gain [dB]	P1dB [dBm]	Bias Current [mA]	Zin [Ω]	Zout [Ω]
SP202	500	~15	+39	400	5.4 – j7.4	14.8 – j12
SP203	500	~15	+40.8	600	3.5 – j4.5	10.9 – j6.5
SD703	200	~15	+50.4	600	3.0 – j7.8	5.4 – j1.6

Each stage was designed and tested separately to be matched to 50 Ω on the input and output. Then, the pre-amplifier stage and driver stage were connected and tested together. Finally, the power amplifier stage was added and the complete 3-stage power amplifier was tested.

5.2.2 DC Bias Simulations

To determine the necessary bias current for the SP202, SP203 and SD703 transistors, a DC bias simulation was performed using Agilent's Advanced Design System (ADS) 2008 Version 2. The ADS simulation schematic has been provided in Appendix B. The non-linear Spice models were downloaded from the Polyfet website (www.polyfet.com) for the SP202, SP203 and SD703 transistor. It is important to note that according to the Polyfet website, "ADS2005 and ADS2006 MOSN model simulates incorrectly when Rd and/or Rs are less than or equal to 0.1 Ω. Consequently, the "workaround models" need to be used." The SP202 and SP203 transistors did not have a workaround model; therefore, the non-linear 'workaround' model was used only for the SD703 transistor. The schematic for each ADS transistor model is provided in Appendix B.

The DC bias simulation performed in ADS using the non-linear models did not provide the same results as those shown in the datasheet for any of the transistors. Therefore, a DC bias measurement on an actual SP203 transistor was performed in the lab using a drain-to-source voltage of +28 V. Table 2 provides the comparison between the datasheet, simulated and measured results for the SP203.

Table 2: SP203 DC Bias - Datasheet, Simulated and Measured

	Drain Current, Id (A)		
VGS (V)	Datasheet	Simulated	Measured
2	0	0	0
4	0.4	1.4	0.51
6	1.6	3.2	1.56
8	2.6	4.1	3.0

The data from Table 2 indicate that the datasheet values are more accurate and the SP203 non-linear ADS model is not accurate; therefore, the gate-to-source bias voltage (VGS) needs to be adjusted in ADS to force the model to provide the desired current. This VGS value will not be the actual VGS value used to bias the transistor in the implemented design; therefore, the VGS value used in ADS for the SP203 driver stage will be referred to as VGS_DA'. Since the simulated and datasheet DC bias values for the SP202 and SD703 were also conflicting, it was assumed that the datasheet values were accurate and the non-linear model was inaccurate and would also need to have adjusted VGS values, hereafter referred to as VGS_PRE' for the SP202 and VGS_PA' for the SD703.

For each amplification stage, the DC bias point (i.e. operating class) was determined according to the peak voltage (V_p) of the input waveform to that stage (or the peak output voltage of the previous stage, assuming no losses due to impedance mismatch). For example, the V_p of the input waveform to the pre-amplifier stage is based on the peak output voltage of the Direct Digital Synthesizer (DDS), assuming no reflected losses from impedance mismatch. The DDS has a maximum output power of about +4.5 dBm (or 2.82 mW) for a 50-Ω system. Using Equation 12, where Power is in Watts, the peak voltage is 1.0 V.

$$V_p = \sqrt{Power * 50\Omega} * \sqrt{2}$$ (12)

However, as seen in Table 1, the input impedance at the gate of the SP202 is much less than 50 Ω (5.4 Ω according to the datasheet), thus a matching transformer was used to transform the impedance. The impedance transformer will also transform the voltage and the input voltage will be stepped down according to the turns-ratio of the input matching transformer. A 4:1 impedance matching transformer was used on the input of each stage to transform the input impedance closer to 50 Ω; therefore, the maximum V_p of the input waveform at the gate of the transistor would be one-half the voltage of the 50-Ω system, or 500 mV. Table 3 provides a summary of the calculation of the operating conditions for each transistor.

Table 3: Operating Conditions - SP202, SP203 and SD703

	Max. P_{in} [dBm]	Max. P_{in} [W]	Max. V_p at Input [V]	V_p Swing at Gate [V]	Current Swing [A]
SP202	+4.5	0.00282	0.531	±0.266	0.48 to 0.51
SP203	+25	0.32	5.66	±2.82	0 to 1.75
SD703	+40	10	31.6	±15.8	0 to 25

According to the SP202 datasheet, a gate voltage (V_g) greater than 2 V is needed for conduction and a V_g of about 4 V is needed to bias the transistor at 400 mA drain current. Using the data from Table 3, it can be seen that the SP202 biased at 400 mA, is operating as a Class A amplifier since the input voltage swing will not cause the transistor to turn off (i.e. 4V – 0.5V = 3.5V > 2V). The SP203 datasheet also states a V_g greater than 2 V for conduction and a V_g of about 4.2 V is needed to bias the transistor at 600 mA drain current. The SP203 biased at 600 mA is operating as a 'weak-biased' Class AB amplifier since the input voltage swing causes the transistor to turn off for a small portion of the input waveform. (i.e. 4.2V – 2.82V = 1.38V < 2V). The SD703 needs to operate as a 'strong-biased' Class AB push-pull amplifier; therefore, the gate voltage should be such that the transistor is on the edge of conduction, which the datasheet states that a gate voltage of 2 V is needed for conduction. From Table 3, it can be seen that the SD703 biased at 600 mA is operating as a 'strong-biased' Class AB push-pull amplifier. It is important to note that according to the datasheet, the maximum DC drain current of the SD703 is 16 A; however, according to Table 3 the SD703 will swing up to 25 A. This does not exceed the maximum current rating of the transistor, since the 25 A is an AC current and the 16 A is for DC current. The 'Id & Gm vs Vg' curve on the datasheet shows a drain current of about 25 A for Vg > 20 V; thus, the transistor will operate within the tolerances specified.

Finally, the drain voltage bias point (V_d) is determined using the output voltage of the SD703 since it will have the largest voltage swing. From Equation 12, the V_p of the SD703 output waveform is 100 V for a 50-Ω system. Given that the datasheet states an output impedance of 5.4 –j1.6 Ω at 200 MHz, a 1:9 impedance transformer will be used to transform the output impedance of the SD703 to 50 Ω (or transform 50 Ω into the R_{opt} of 5.4 + j1.6 for maximum power). Thus, the V_p at the drain of the transistor, with an ideal turns ratio of 3, will be 33.3 V. Since the maximum available supply voltage is +28 V, the output voltage at the drain will swing from -5.3 V up to 61.3 V, thus there may be some clipping on the output voltage waveform (i.e. 28V – 33.3V = -5.3V < 0) depending on the actual turns ratio of the implemented 1:9 impedance transformer and output matching network. Also, the SD703 datasheet states that the maximum drain-to-source voltage is 70 V; hence, the transistor will operate within the tolerances specified.

5.2.3 Stability Simulations

The stability of an amplifier can be determined from the S-parameters, the matching networks, and the terminations. In a two-port network, oscillations are possible when either the input or

output port presents a negative resistance. This occurs when $|\Gamma_{IN}| > 1$ or $|\Gamma_{OUT}| > 1$, which for a unilateral device occurs when $|S_{11}| > 1$ or $|S_{22}| > 1$. Hence, if $|S_{11}| > 1$ the transistor presents a negative resistance at the input, and if $|S_{22}| > 1$ the transistor presents a negative resistance at the output. A convenient way of expressing the necessary and sufficient conditions for unconditional stability is given in Equation 13 and 14. [23]

$$K = \frac{1 - |S_{11}|^2 - |S_{22}|^2 + |S_{11}S_{22} - S_{12}S_{21}|^2}{2|S_{12}S_{21}|} > 1 \tag{13}$$

$$B_1 = 1 + |S_{11}|^2 - |S_{22}|^2 - |S_{11}S_{22} - S_{12}S_{21}|^2 > 0 \tag{14}$$

After selecting the operating bias current for each transistor, a stability simulation was performed using ADS for the SP202, SP203 and SD703 transistor from 10 MHz to 1 GHz at 1 MHz steps. The ADS measurement expressions 'stab_meas()' and 'stab_fact()' were used to test the necessary and sufficient conditions for unconditional stability. 'Stab_fact()' returns the Rollett stability factor (also known as 'K-factor') defined in Equation 13. 'Stab_meas()' returns the stability measure defined according to Equation 14. Each stage was simulated independently and initial stability simulations showed that the SP202 and SP203 stages were not unconditionally stable; therefore, drain-to-gate feedback was added to the SP202 and SP203 transistors to ensure unconditional stability. Drain-Gate feedback improves the stability of a MOSFET by using feedback to reduce the negative resistance of the transistor by limiting the low frequency gain to a level which will not support oscillation with any input termination [24].

Furthermore, the initial stability simulations also showed that the SD703 power stage was not unconditionally stable; therefore, shunt-gate resistance was added to the SD703 to ensure unconditionally stability. Shunt-gate resistance increases the loss of any series impedance terminating the gate terminal. At frequencies where the gate input impedance is negative, the shunt resistance prevents oscillation by effectively cancelling the negative resistance of the transistor.   A series capacitor is also included as a DC block. A final stability simulation was performed on the complete 3-stage design to ensure that the interstage matching did not affect the stability. The schematics for each ADS stability measurement are located in Appendix B. In summary, the simulation of the final amplifier design was unconditionally stable from 10 MHz to 1 GHz with an input power to the SP202 of 0 dBm.

5.2.4 Load-Pull Simulations

Load-pull consists of varying or "pulling" the load impedance seen by a transistor while measuring the performance of the transistor. The load-pull simulation is used to measure a transistor in actual operating conditions, and is important for large-signal, non-linear devices (such as transistors) where the operating point may change with power level or tuning. Thus, the output power and efficiency capabilities of a transistor can be simulated without the need of building and measuring and entire amplifier. In ADS design guide, the built-in example schematic 'HB1Tone_LoadPull' was modified to perform the load-pull measurements for the

SP202, SP203 and SD703 transistors. The 'HB1Tone_LoadPull' simulation generates contours that indicate load impedance values that when presented to the output of the transistor (along with specified source impedance and available source power) will cause a certain power to be delivered to the load. The actual contour lines for output power and PAE are generated for varying load impedances. The schematics for each ADS load-pull measurement are provided in Appendix B. It is important to note that the load-pull simulation for the SD703 was performed as a single-ended measurement even though the SD703 is a push-pull device. Assuming the same optimum load impedance (Zopt) will be presented to both halves of the SD703; Zopt for the push-pull transistor can be calculated by doubling the single-ended Zopt since both halves will be in series. Table 4 provides the single-ended load-pull measurements for the SP202, SP203 and SD703 transistors. Since the gain decreases as frequency increases, the value at 200MHz was used for Z_{opt}. P_{out} and PAE were calculated by taking an average of the measurement values from 140 to 220 MHz at 20-MHz intervals. It is significant to state that the PAE values in Table 4 do not consider any losses; therefore, the actual PAE will be lower than the values in Table 4. Table 5 provides the Z_{out} (where Z_{opt} is the complex conjugate of Z_{out}) for maximum power from the SP202, SP203 and SD703 datasheets at various frequencies. It can be seen that the simulated values for Z_{opt} in Table 4 agree with the datasheet values in Table 5 for maximum power.

Table 4: Load-Pull Simulation Results - SP202, SP203 and SD703 (single-ended)

	P_{in} [dBm]	Z_{opt} [Ω]	P_{out} [dBm]	P_{out} [W]	PAE [%]
SP202	+5	12.03 + j4.4	22.9	0.195	1.6
SP203	+22	12.58 + j11.6	41.5	14.1	52
SD703	+26	2.55 + j0.6	50.2	104.7	60

Table 5: Z_{out} vs Frequency - SP202, SP203 and SD703 (push-pull) Datasheet Values

		Z_{out} [Ω]			
	P_{out} [dBm]	100 MHz	200 MHz	500 MHz	600 MHz
SP202	+39	Not given	Not given	14.8 – j12	11.1 – j10
SP203	+40.8	Not given	Not given	10.9 – j6.5	7.2 – j5.0
SD703	+50.4	11.0 – j3.6	5.4 – j1.6	2.8 + j0.4	2.2 + j1.2

5.2.5 Input and Output Matching Networks

In power amplifier design, the output matching network is one of the most crucial design elements primarily because it is responsible for transforming the 50-Ω system impedance into the Z_{opt} of the transistor which primarily determines both output power and PAE. Also, since transistors are not unilateral devices, the input impedance (Z_{in}) will change based upon the output impedance (Z_{out}). For these reasons, the impedance matching networks for each transistor were designed according to the following procedure:

1) Z_{opt} measured using ADS Load-Pull simulation → Output MN designed to transform 50Ω system impedance into Z_{opt}.

2) Z_{in} measured using ADS with Output MN → Input MN designed to transform Z_{in} into 50Ω system impedance.

Using load-pull data from Table 4, Z_{opt} for the SD703 can be calculated as 5.1 + j1.2 Ω and it can be seen that a 1:9 transformer provides a wideband transformation of 50 Ω into 5.56 + jX Ω, where X is the transformation of the reactive component. To compensate for the parasitics of the transformer and further transform 5.56 Ω into Z_{opt} of 5.1 + j1.2 Ω, a lumped element matching network (described in Section 5.2.5.2) was designed to complete the transformation into Z_{opt}. The same technique of a wideband transformer with a lumped element matching network was utilized for the output transformer of the SP202 and SP203 with the exception that a 1:4 transformer was designed to transform 50 Ω into 12.5 + j11 Ω.

5.2.5.1 Impedance Transformers / Baluns

In the design of RF PAs, wide-band transformers play an important role in the quality of the amplifier as they are a fundamental component in determining the input and output impedances, gain flatness, linearity, power efficiency, and other performance characteristics. Selection of the magnetic materials, the conductors, and the method of construction is crucial as these choices are significant in determining the overall performance of the transformer. The theory and approach in [26] was used to design, simulate and construct the transformers used in the matching networks. Three different configurations of impedance transformers were utilized in the matching networks as outlined in Table 6.

Table 6: Impedance Matching Transformers / Baluns

	Input MN	Output MN
SP202	4:1 Equal Delay Unun	1:4 Equal Delay Unun
SP203	4:1 Equal Delay Unun	1:4 Equal Delay Unun
SD703	1:1 Balun w/ 4:1 Balbal	1:9 Balbal w/ 1:1 Balun

After constructing each transformer, 1-port S-parameters were taken using the HP8722D Vector Network Analyzer (VNA) with the transformer terminated into 50Ω. Then, two identical transformers were connected back-to-back and 2-port S-parameters were measured using the HP8722D VNA. A MATLAB script was used to retrieve the data via the GPIB port using a laptop computer and convert the .DAT files into .S1P and .S2P. The S-parameters (in the form of .S1P and .S2P files) were then introduced into ADS to create lumped element equivalent circuit models for each transformer based on the equivalent circuit model in [26]. The ADS schematics for determining the equivalent circuit models and the simulation results are provided in Appendix B. The equivalent circuit models for each transformer were then used in the power amplifier design

simulation instead of the ADS component 'XFERTL2', to provide a more accurate representation of the parasitic reactive elements of each transformer.

One of the prime factors limiting high frequency performance of transmission line transformers is the phase error caused by the arbitrary length of the transformers' interconnections. If these connections were made using a transmission line of the same length, velocity factor and impedance as the transformer line itself, then the phase error would be eliminated. This transformer configuration, as illustrated in Figure 33, is a modification of the basic 1:4 Ruthroff transformer and is referred to as an 'equal delay' (ED) transformer. No ferrite loading is required for the added interconnecting delay line. There is also no voltage drop on the outer conductor, thus no isolation is required from one end to the other [25].
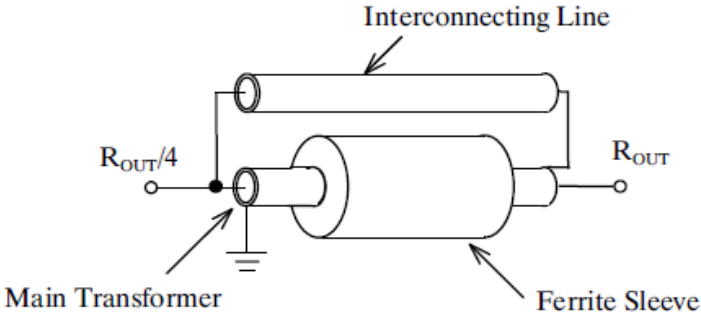
Figure 33: 1:4 Equal Delay Unun Transformer Diagram

Typically, the best response is achieved when the ED transformer is constructed using a straight, coaxial cable loaded with cylindrical cores or stacked toroids so that the opposite ends of the transformer can be separated by the length of the transmission lines. However, due to layout constraints of the T/R module, the ED transformer was also constructed using a bent coaxial cable loaded with ferrite so that it could be mounted vertically and conserve horizontal real estate on the board layout. The straight and bent ED transformer implementations are shown in Figure 34 and Figure 35, respectively. The geometric mean is used to determine the optimum characteristic impedance of the coaxial cable according to Equation 15 [25].

$$Zo = \sqrt{Z_{in} Z_{out}} \tag{15}$$

Thus, for both ED transformers the optimum Z_o of 25Ω was calculated using Equation 15 and 25-Ω coax (UT-047-25 from Micro-Coax) was used to construct both the straight and bent 1:4 ED transformer. A length of about 1 inch was used for the straight ED transformer and a length of 1.75 inches was used for the bent ED transformer. The UT-047-25 specifications are provided in Appendix E. Five, 43 material ferrite beads (FB-43-101 from Amidon) were used for the straight ED transformer, while one 43 material ferrite binocular core (BN-43-2402) was used for the bent ED transformer.

Furthermore, the 1:4 ED transformer was used in reverse to realize the 4:1 ED transformers. The simulated equivalent circuit model transformation results are given in Figure 36.

Figure 34: 1:4 Equal Delay Unun Transformer Implementation – Straight
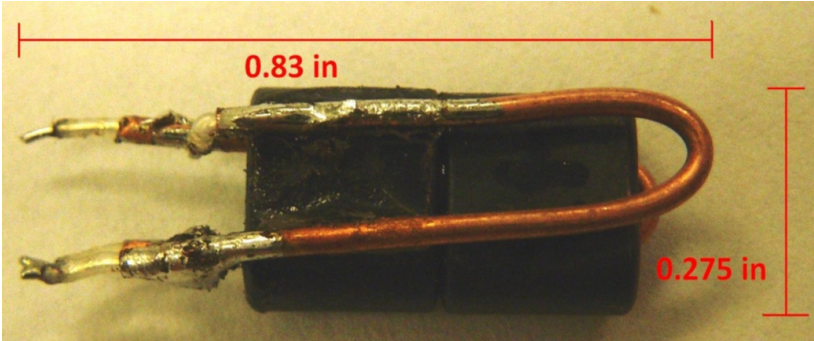
Figure 35: 1:4 Equal Delay Unun Transformer Implementation – Bent
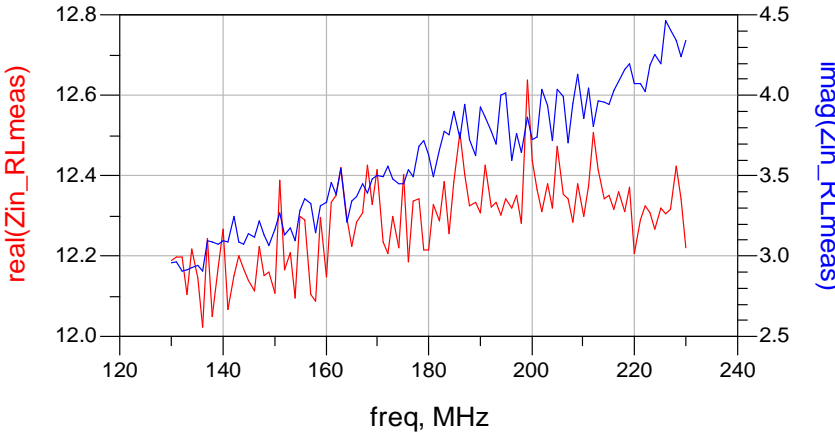
Figure 36: 1:4 ED Equivalent Circuit Impedance Transformation Results

The 4:1 Balbal (balanced-to-balanced) transformer was constructed according to the schematic, illustration and implementation shown in Figure 37, Figure 38 and Figure 39, respectively [26]. The optimum Z_o of 25 Ω was calculated using Equation 15 and two, 1.5 inch long, 25-Ω coax (UT-047-25 from Micro-Coax) cables were used to construct the 4:1 Balbal. Two, 43 material binocular cores (BN-43-2302 from Amidon) were used for the ferrite material. Typically, this 4:1 transformer is implemented in a horizontally planar fashion, as depicted in Figure 38, but to save horizontal real estate on the board layout the transformer was implemented vertically instead. The simulated equivalent circuit model transformation results are given in Figure 40.
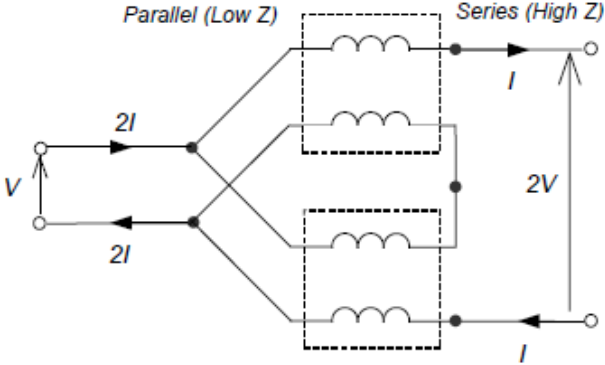
Figure 37: 4:1 Balbal Transformer Schematic
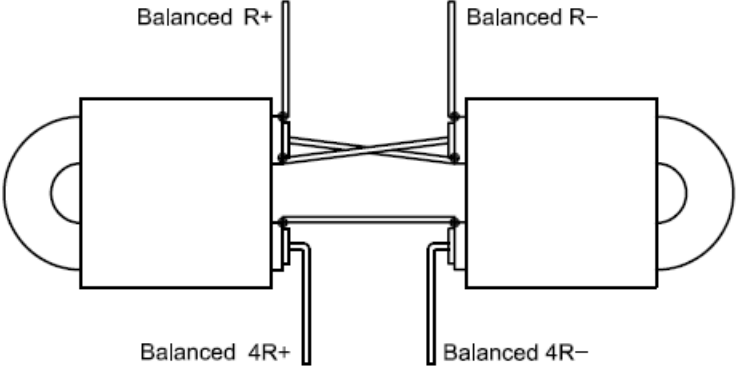
Figure 38: 4:1 Balbal Transformer Diagram
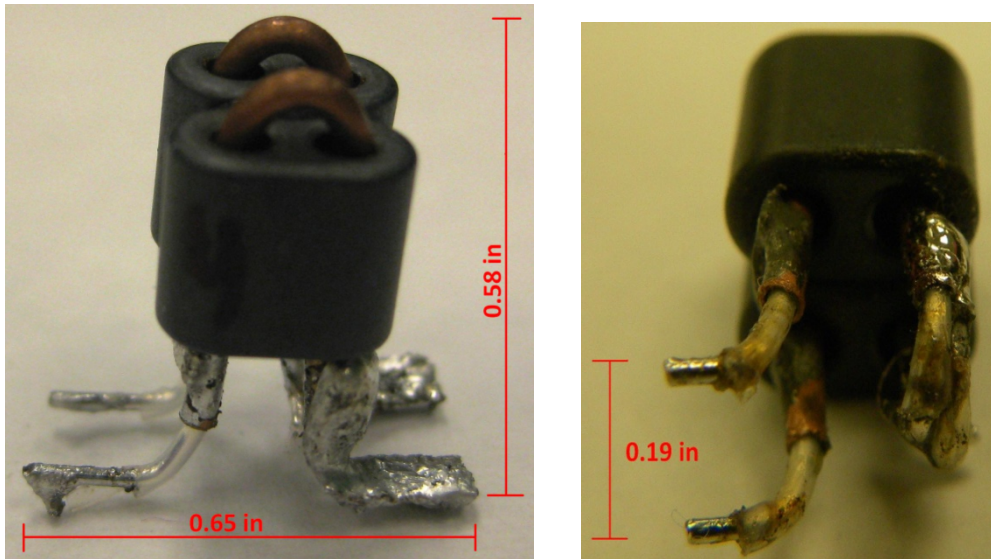
Figure 39: 4:1 Balbal Transformer Implementation
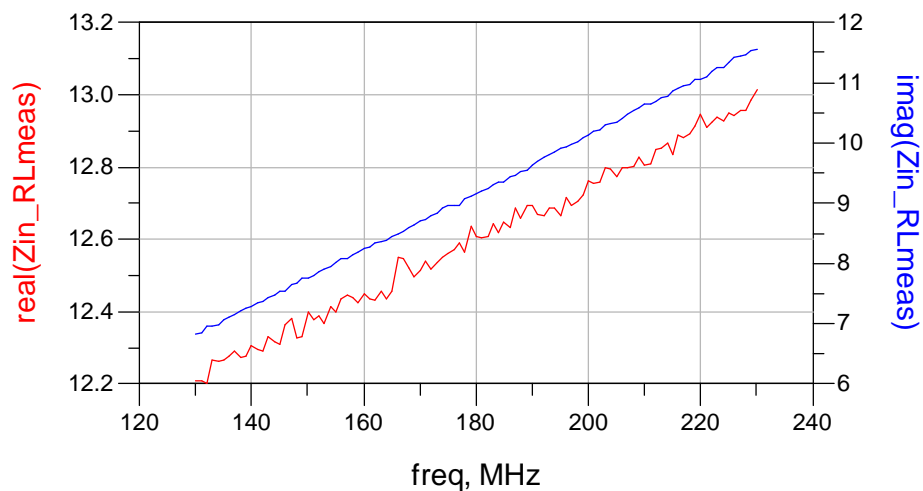
Figure 40: 4:1 Balbal Equivalent Circuit Impedance Transformation Results

The 1:9 Balbal transformer was constructed according to the schematic, illustration and implementation shown in Figure 41, Figure 42 and Figure 43, respectively [26]. The optimum Z_o of 16.7 Ω was calculated using Equation 15 and 15-Ω coax (UT-085C-15 from Micro-Coax) was used to construct the 1:9 Balbal transformer. The UT-085C-15 specifications are provided in Appendix E. It is important to note that smaller diameter coax could have been used in this design since the average power handling capability at maximum output power would only be about 10 W with a 10% duty cycle. However, the larger diameter coax was used for two reasons. One, to ensure a CW power handling capability of 100 W at 210 MHz and two, to reduce the insertion loss of the cable since insertion loss decreases as the diameter increases. The ferrite material used

were two, 43 material binocular cores (BN-43-6802 from Amidon). Typically, this 1:9 transformer is implemented in a horizontally planar fashion, as depicted in Figure 42. However, to conserve board layout space the transformer 'halves' were stacked together and laid out horizontally, as shown in Figure 43, which reduced the area occupied by the transformer by one-half and still met the height restriction of 1 inch even with the 62-mil board and 250-mil heat-sink mounting plate. Since this transformer is responsible for the most crucial transformation of the design, the longer cores and cables were used to increase the performance of the transformer [26] which extended the transformer height beyond the 1 inch design limit; thus, this transformer could not be mounted vertically. As a final note, 61 or 64 material could also be used for this design if 43 material is not available as they have similar characteristics beyond 100 MHz. The simulated equivalent circuit model transformation results are shown in Figure 44.
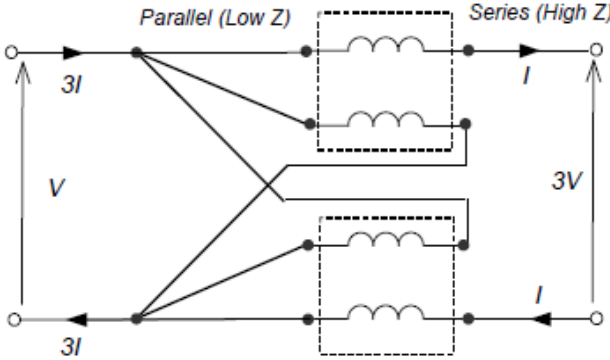
Figure 41: 1:9 Balbal Transformer Schematic
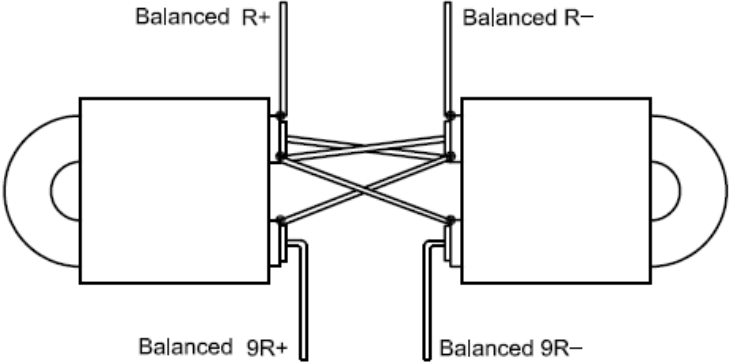
Figure 42: 1:9 Balbal Transformer Diagram
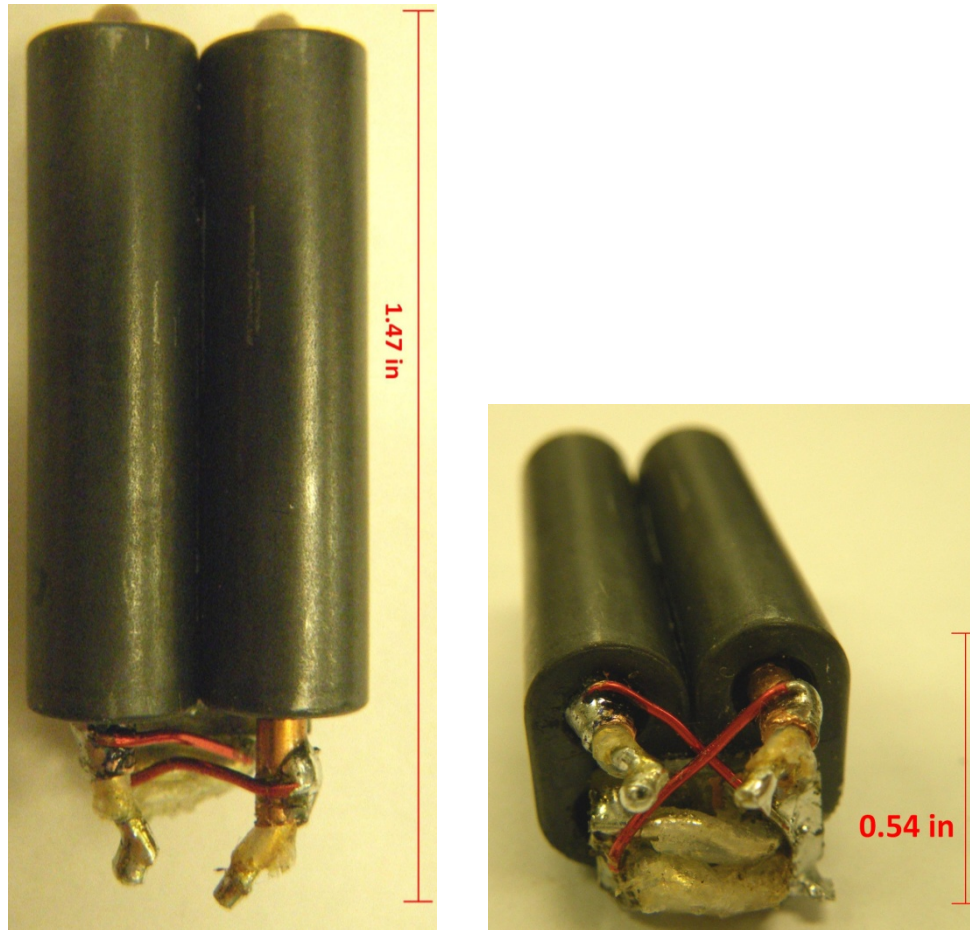
Figure 43: 1:9 Balbal Transformer Implementation
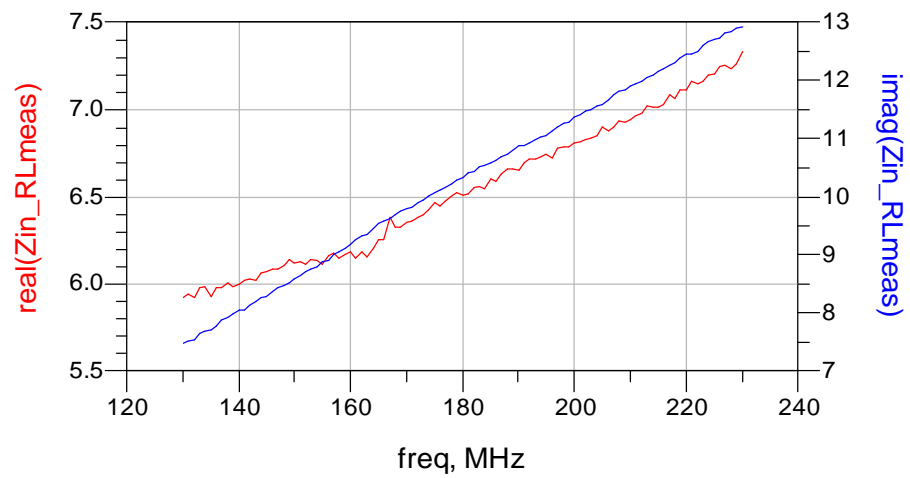
Figure 44: 1:9 Balbal Equivalent Circuit Impedance Transformation Results

5.2.5.2 Lumped Element Matching Network

Once the values of the parasitic reactive elements of the transformer model have been determined, the inductive reactance of the transmission lines (i.e. the coax) in each transformer will usually dominate and can be used along with additional external capacitance to form a lumped element matching network to complete the matching to the desired Z_{opt} from the load-pull simulations [26]. Table 7 provides a summary of the Z_{opt} data from Table 4 and data from Figure 36, Figure 40 and Figure 44. The data from Table 7 shows that the 1:4 ED Unun transformer is sufficient in generating Z_{opt} for both the SP202 and SP203 transistors output matching networks; thus, additional external matching is not necessary. However, it can be seen that the measured results of the 1:9 Balbal do not provide a sufficient match and additional external capacitance will be needed to reduce the inductive reactance and complete the transformation.

Table 7: Z_{opt} vs. Measured Transformed Impedance

	Z_{opt} [Ω]	Measured Z @ 200 MHz [Ω]
SP202	12.03 + j4.4	~12.5 + j3.5 (1:4 ED Unun)
SP203	12.58 + j11.6	~12.9 + j9.5 (1:4 ED Unun)
SD703	5.1 + j1.2	~6.8 + j11.4 (1:9 Balbal)

Using ADS, the lumped element matching network of Figure 45 was added to the 1:9 Balbal transformer to create the Output MN. The ADS equivalent circuit model named '1to9_2COAX_Balun_UT85_EQ_MODEL' in Figure 45 also includes the 1:1 Balun which was constructed from 50Ω RG405/U coaxial cable (Belden 1671A). The simulation results for the SD703 output MN are given in Figure 46, which shows a transformation of 50 Ω to 5.14 + j2 Ω at 200 MHz which is sufficient for matching to the Z_{opt} from Table 7.
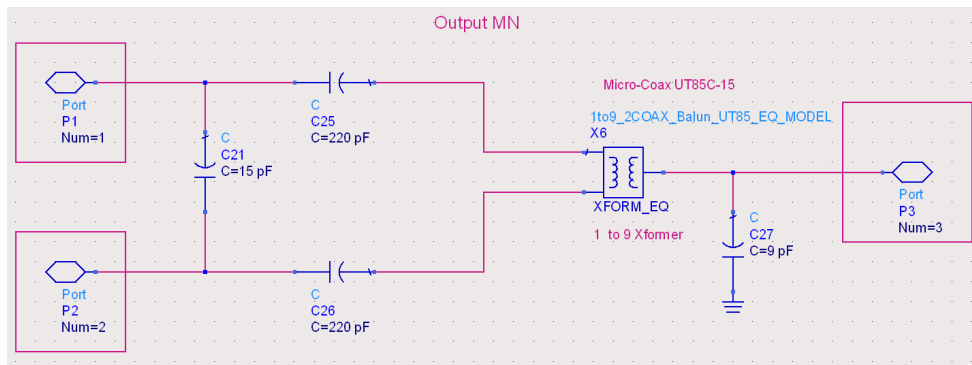
Figure 45: Output Matching Network Model ('MN_OUTPUT_SD703_MODEL')
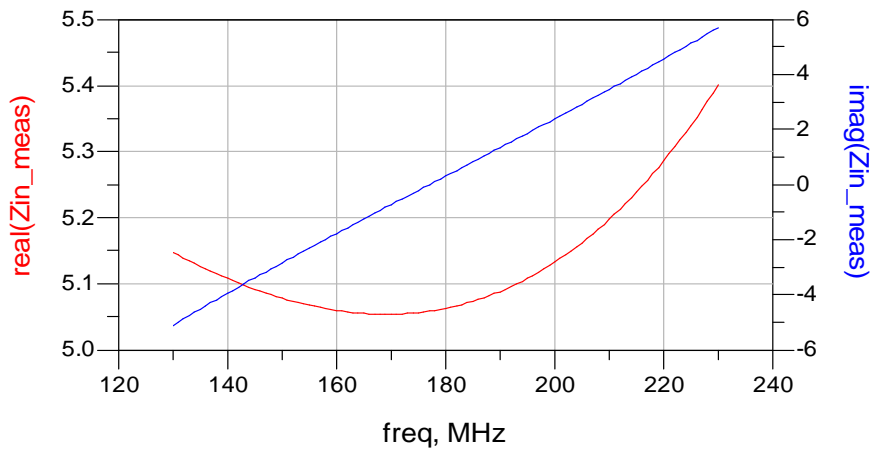
Figure 46: SD703 Output Matching Network Simulation Results

5.2.5.3 Input Matching Networks

The input MN was designed for each stage by first, attaching the previously designed output MN, then inserting the appropriate matching transformer on the input (according to Table 6) and measuring the input impedance (Z_{in}) with the circuit under biased operating conditions. If needed, a lumped element matching network was designed to complete the matching of Z_{in} to 50 Ω. From the initial simulation results it was determined that only the SD703 input transformer required an additional lumped element matching network as shown in Figure 49. The 4:1 ED input transformers for the SP202 and SP203 sufficiently provided the transformation with a VSWR less than 2 from 140 to 210 MHz as shown in Figure 47 and Figure 48. Figure 50 provides the simulated results for the SD703 input MN given in Figure 49.
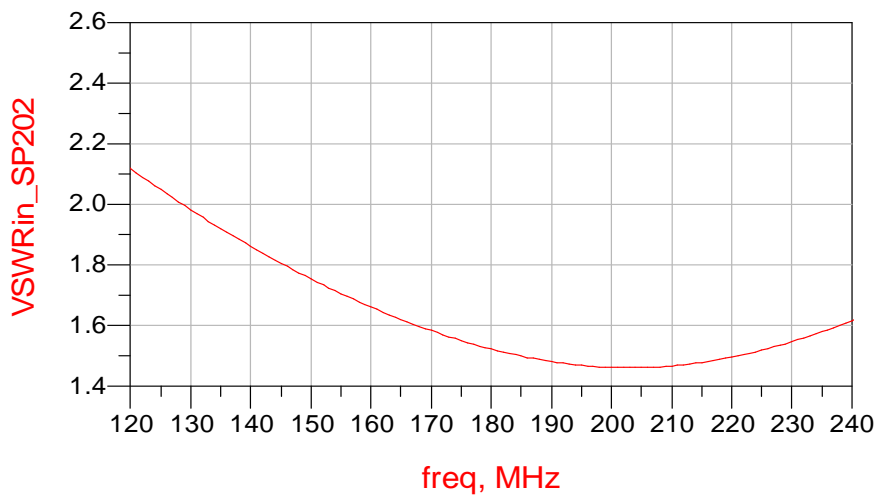
Figure 47: SP202 Input Matching Network Simulation Results
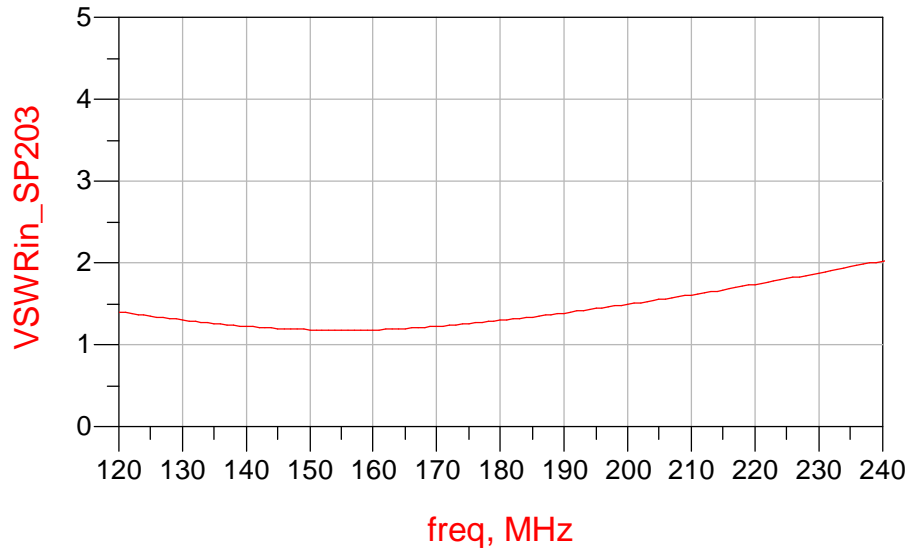
Figure 48: SP203 Input Matching Network Simulation Results
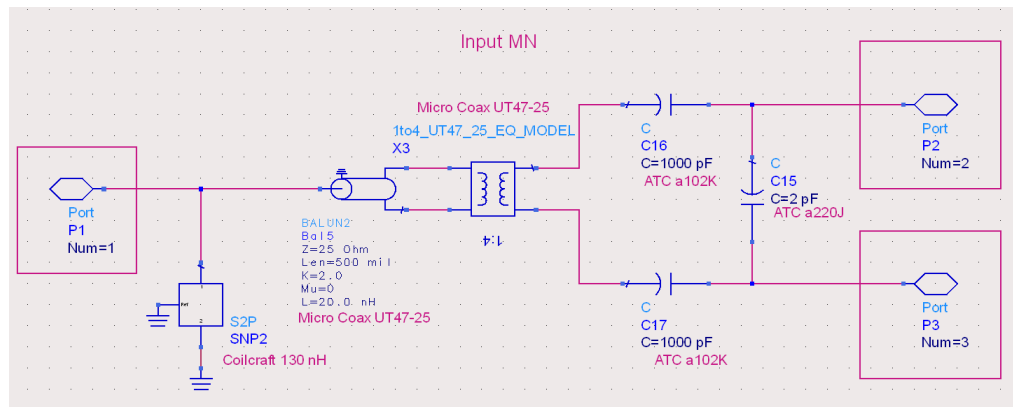
Figure 49: SD703 Input Matching Network ('MN_INPUT_SD703_MODEL')
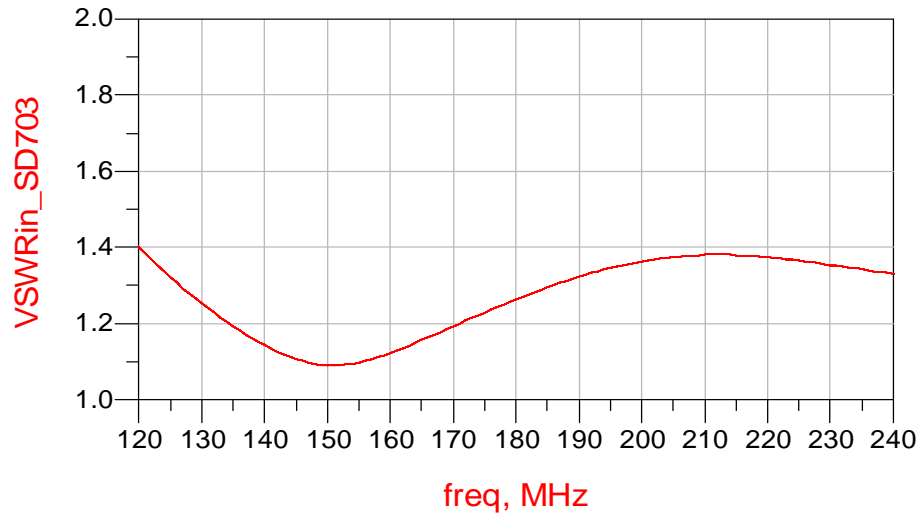
Figure 50: SD703 Input Matching Network Simulation Results

5.2.6 Complete Power Amplifier Design Simulation

After designing the initial input and output matching networks, each stage was simulated independently to determine the effects of the bias circuit with the input MN on the output impedance, since the previous design procedure did not take into account these effects for the output impedance. The final design schematics for each stage can be found in Appendix B. Figure 51, Figure 52 and Figure 53 are the simulated results for the output VSWR for the SP202, SP203 and SD703, respectively.
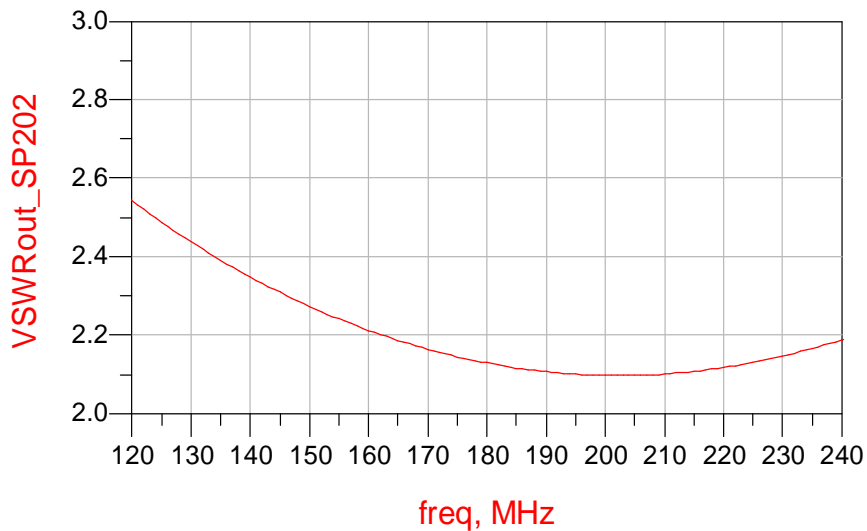
Figure 51: SP202 Output VSWR Simulation Results
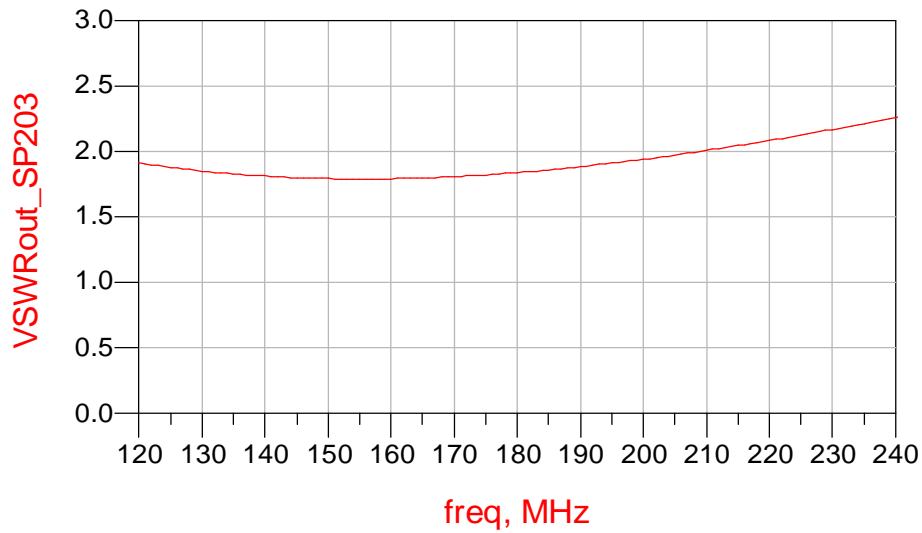
Figure 52: SP203 Output VSWR Simulation Results
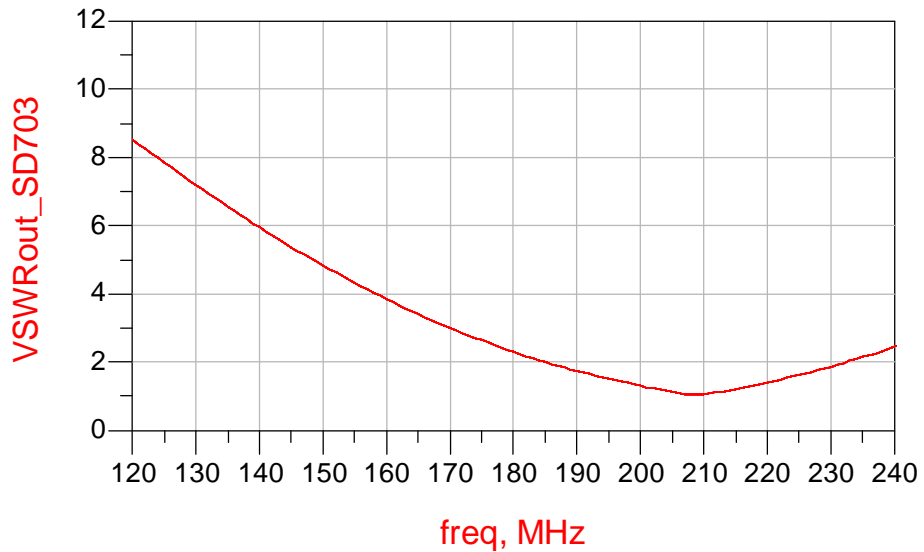
Figure 53: SD703 Output VSWR Simulation Results

Finally, the three stages were connected together with 2-dB attenuators placed between each stage to reduce any reflections between stages due to mismatch, as shown in Figure 54. The schematics for the amplifier blocks 'SP202_DA_Final_Model', 'SP203_DA_Final_Model', and 'SD703WA_PA_Final_Model' can be found in Appendix B. The S-parameter results are provided in Figure 55 and Figure 56.
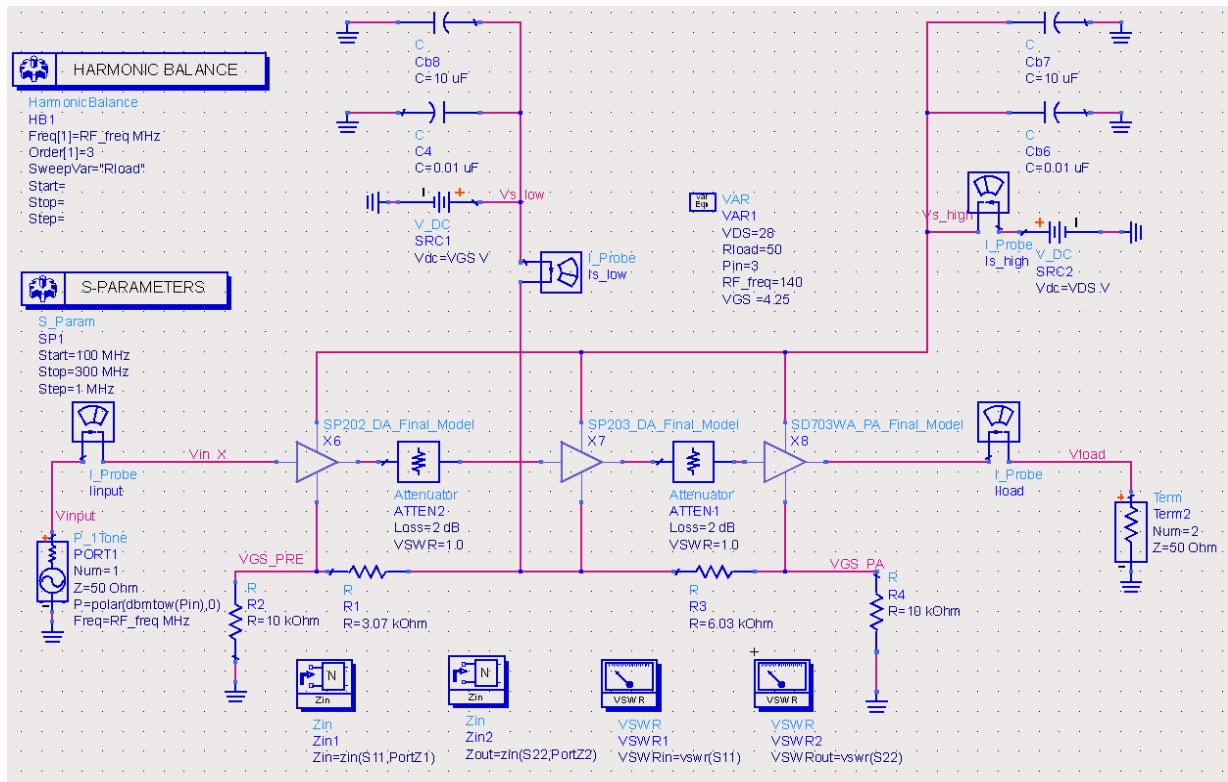
Figure 54: Final Design Simulation Schematic
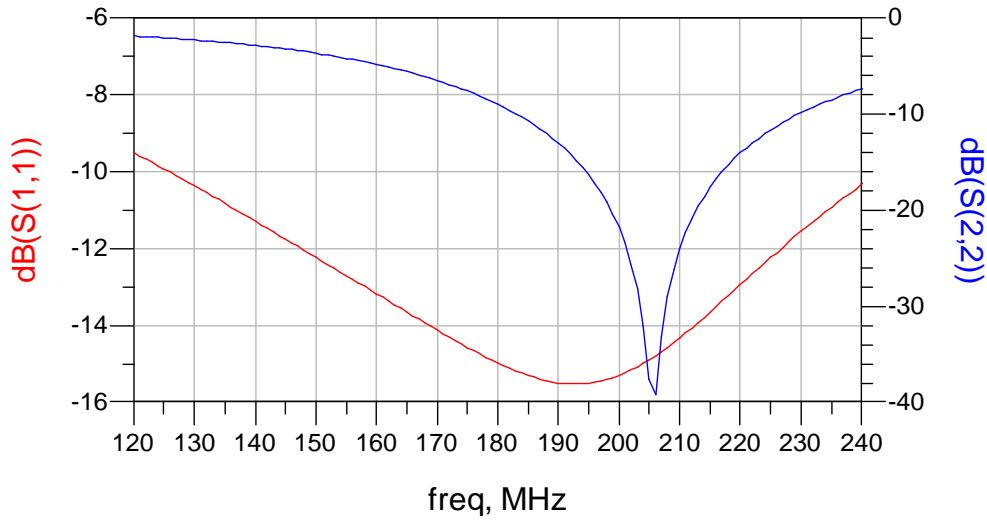
Figure 55: S11 and S22 of Final Design Simulation
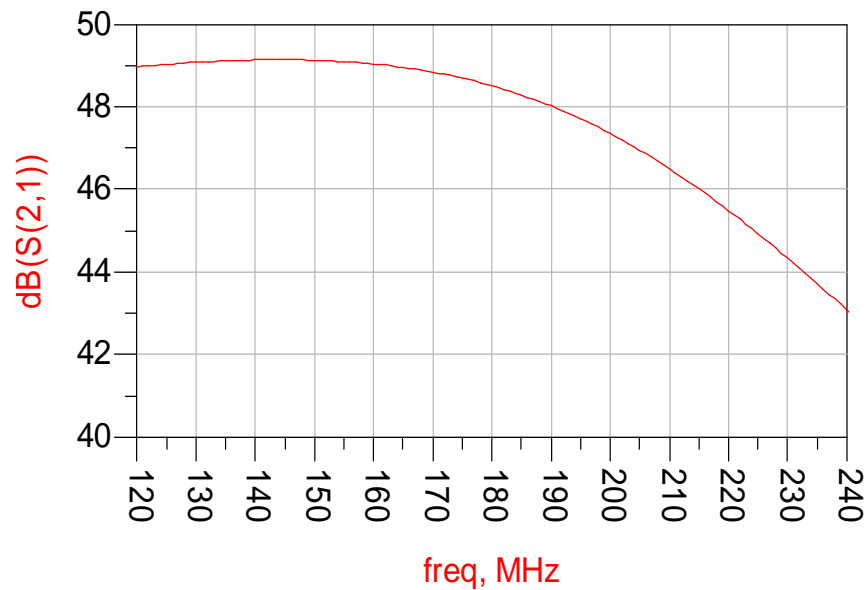
Figure 56: S21 of Final Design Simulation

As seen in Figure 55, the return loss (specifically S22) was 'tuned' to the higher end of the frequency band in order to compensate for the high-frequency gain reduction which occurs in most RF PAs, as is apparent in Figure 56. The gain has about a 3-dB variation from 140 to 210 MHz in the simulation, which will cause an increase in side-lobes as discussed in section 2.3.1. However, the gain can be 'tuned' and 'flattened' in the hardware in the final implementation by adjusting the external capacitance in the SD703 output MN. Also, the output power can be flattened by adjusting the input power at individual frequencies through the digital predistortion linearization as discussed in Chapter 6.

5.2.7   MOSFET High Speed, High Current Switching Circuit Design

5.2.7.1 Theory of Operation

One of the novel features of this PA design is the ability to switch (or pulse) the PA synchronously with the transmitted radar pulse; thereby, only allowing the transistors to draw current approximately when the pulse is present. Since the PAE is calculated based on a CW mode of operation when the amplifier is conducting (i.e. transmitting), this technique will increase the overall efficiency of the amplifier since according to Table 1 there is at least 1.6 A of bias current being drawn even when the PA is not transmitting. Using a +28 V supply, this equates into about 45 W of power being dissipated even when the PA is not transmitting. Typically, the gate bias voltage (V_{GS}) is switched to 'remove' the bias current; however, this requires a large current to be driven into the gate in order to charge the large input gate capacitance (C_{GS}) of the FET. Figure 57 illustrates the MOSFET equivalent circuit with only those parasitic components

that have the greatest effect on switching. L_D and L_S are the drain and source inductances, respectively, and typically are around a few tens of nH [27].
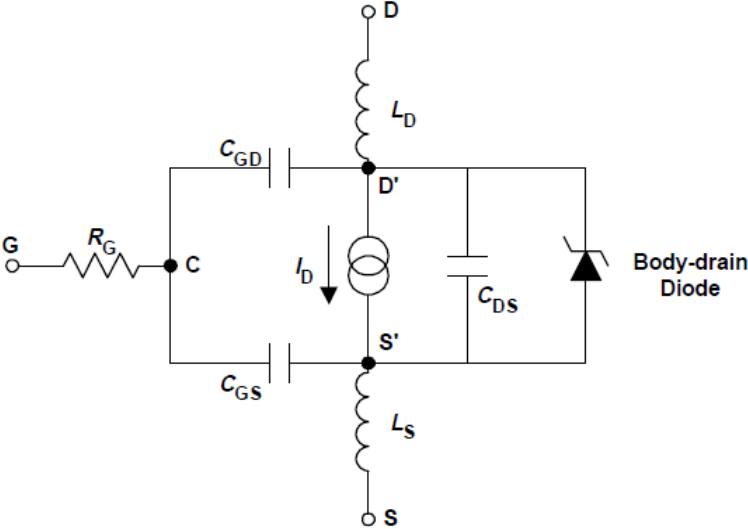
Figure 57: MOSFET Equivalent Circuit (Switching Components Only) [27]

Once C_{GS} is fully charged, the drive current starts to charge the Miller capacitance (C_{GD}) which requires a longer charge time than that for C_{GS}. These charge times result in a turn-on and turn-off delay of the transistor. The turn-on delay, $\tau_{d(on)}$, is the time taken to charge the input capacitance C_{iss} (i.e. $C_{GS} + C_{GD}$) before drain current conduction can reach its maximum value. Similarly, turn-off delay, $\tau_{d(off)}$ is the time taken to discharge the capacitance after the FET is switched off. The turn-on charging time waveforms are illustrated in Figure 58, where $\tau_{d(on)} = t_3$ [27].
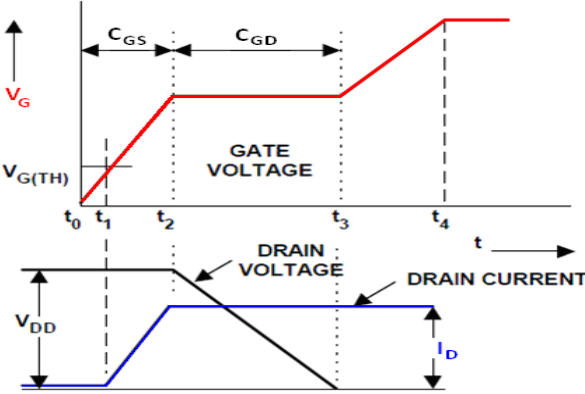
Figure 58: Gate and Drain Charging Waveforms [27]

If V_{GS} is constantly applied and V_{DS} is switched, the amplifier will be 'pre-biased' since C_{GS} will be 'pre-charged' and the conduction channel will already be open ready to complete the charging of C_{GD} once the drain voltage is applied. This will improve efficiency by

reducing the turn-on delay of the transistor since the charging time will begin between t_1 and t_2 depending on the bias point, rather than at t_0.

A high speed switching circuit was designed to achieve the desired turn-on and turn-off times of less than 500 ns. Figure 59 illustrates the operation of the switching circuit.
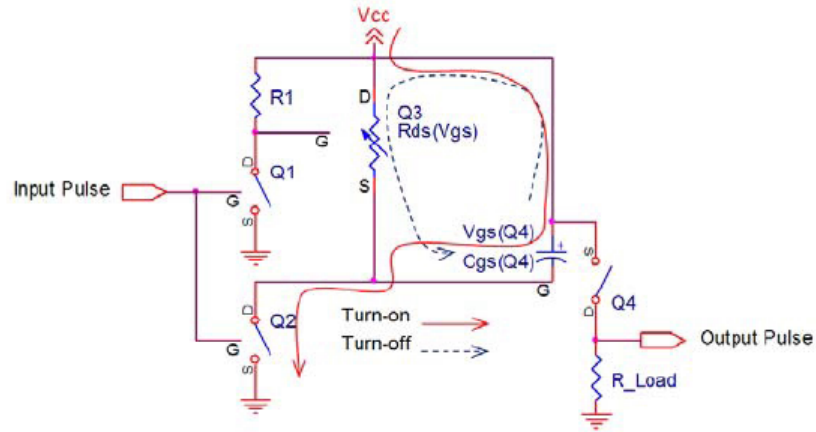
Figure 59: Operation of High Speed Switching Circuit [28]

Q1 through Q3 is a TTL level drive circuit for fast switching of Q4. R1 controls the operating drive current of the circuit with smaller values of R1 providing higher current which result in faster switching speeds, but also increased power dissipation. When the switching circuit is turned on, Cgs(Q4) is charged along the solid current line according to the charging time constant in Equation 16.

$$\tau\_ch\arg e = Rds(on, Q2) * Cgs(Q4) \tag{16}$$

Rds(on,Q2) is the "on" state drain-source resistance of Q2 and Cgs(Q4) is the gate-source capacitance of Q4. When the switching circuit is turned off, Cgs(Q4) is discharged along the dotted line according to Equation 17 with Q3 acting as a voltage controlled variable resistor.

$$\tau\_disch\arg e = Rds(Q3) * Cgs(Q4) \tag{17}$$

Rds(Q3) is the voltage controlled variable resistance of Q3. A BJT, rather than a FET, is used for Q3 to reduce the fall time due to the lower input threshold voltage of Q3 which maintains a lower Rds(Q3) while Cgs(Q4) is discharged [28]. Also, since Q4 has a maximum V_{GS} of ±20 V, the switching circuit requires a zener diode, as shown in Figure 60, to protect Q4 from the +28 V supply voltage. The breakdown voltage of the zener diode is selected according to Equation 18 [28].

$$\left|V_{gs(th)}\ of\ Q4\right| < \left|V_{BR,Zener} - V_{CC}\right| < \left|Maximum\ V_{gs}\ of\ Q4\right| \tag{18}$$
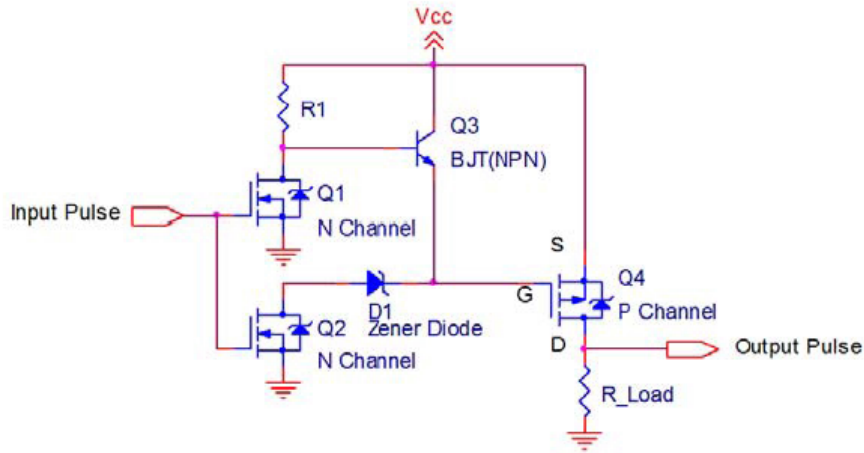
Figure 60: High Speed Switching Circuit General Schematic [28]

It can be seen from Equation 16 and 17 that the FETs and BJT selected should have low Rds(on) and low Cgs for faster switching. Table 8 indicates the components selected for simulating and implementing the switching circuit shown in Figure 60.

Table 8: Switching Circuit Component List

	Component
R1	250 Ω, 2W
Q1,Q2	IRLML2803
Q3	MMBT2222
Q4	SPB08P06P
D1	BZX4C15L

Linear Technology's LTspice IV was used to simulate the switching circuit as shown in Figure 61. The most effective value for R1 was determined by performing a series of simulations using a 5-Ω load. R1 was chosen to be 250 Ω according to the simulation results provided in Table 9. A parallel combination of three, 750 Ω, 1 W, SMD resistors were used to implement the 250 Ω resistor so that the power could be divided among them.

Table 9: Switching Circuit Optimization of R1 - Simulation Results

R1 [Ω]	Rise Time [ns]	Fall Time [ns]	I_R1 [mA]	P_R1 [W]
5k	230	1625	5.6	0.016
1k	230	460	28	0.078
500	230	305	56	1.57
250	235	224	112	3.14
100	245	170	280	7.84
50	265	138	560	15.68

Since the gate biasing voltages will also be created using the +28 V supply, the switching circuit will be switching only the +28 V supply that is attached to the drain of each transistor; thus, the RF choke inductors were also included in the simulation. However, the switching speed will be influenced by the frequency dependent inductance of the RF chokes and the pulse contains high frequency content; therefore, a more accurate RF model was created for each RF choke using ADS [29]. Initial simulation results indicated ringing at the bottom of the falling edge; therefore, a high voltage, high current Schottky diode (MSBR340) was placed in parallel with each RF choke inductor to dampen the oscillations. Also, a 4.7 Ω resistor was placed on the gate of Q4 to suppress gate ringing [30]. The RF choke equivalent circuit simulation and results can be found in Appendix C. The design implementation of the RF chokes is discussed further in Section 5.2.7.1. Also included in the simulation is the circuitry for a switching indicator LED to provide a visual method of determining if the switching circuit is operational. The switching circuit simulation results of Figure 62 and Figure 63 show a rise time of 80 ns and fall time of 100 ns, respectively.
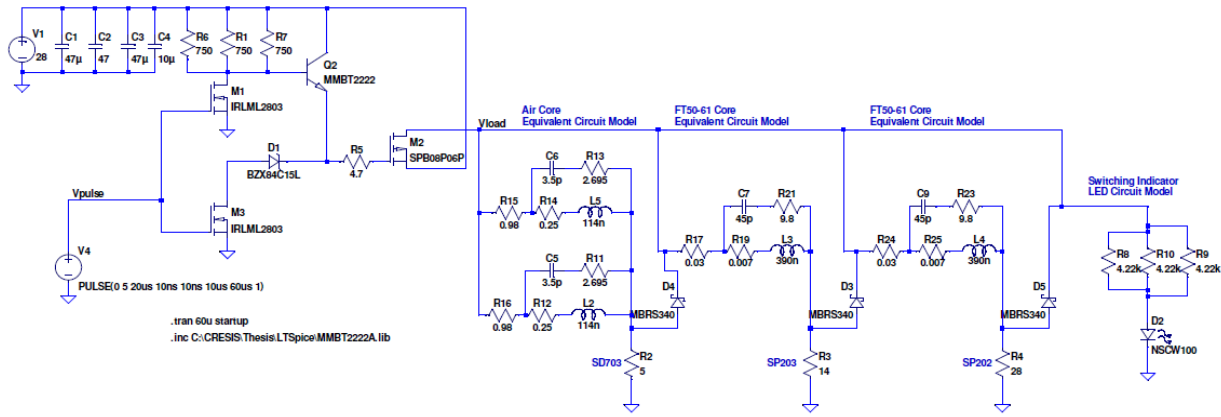
Figure 61: High Speed Switching Circuit LTspice Simulation
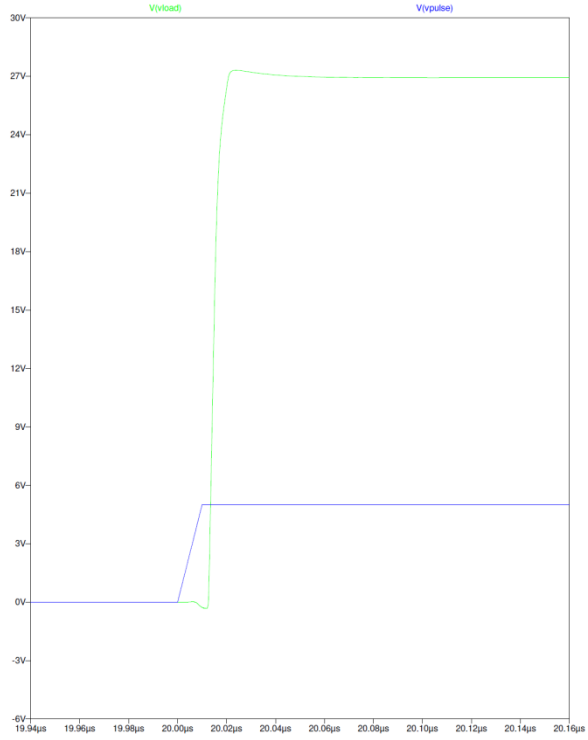
Figure 62: Rise Time of Switching Circuit - Simulation Results
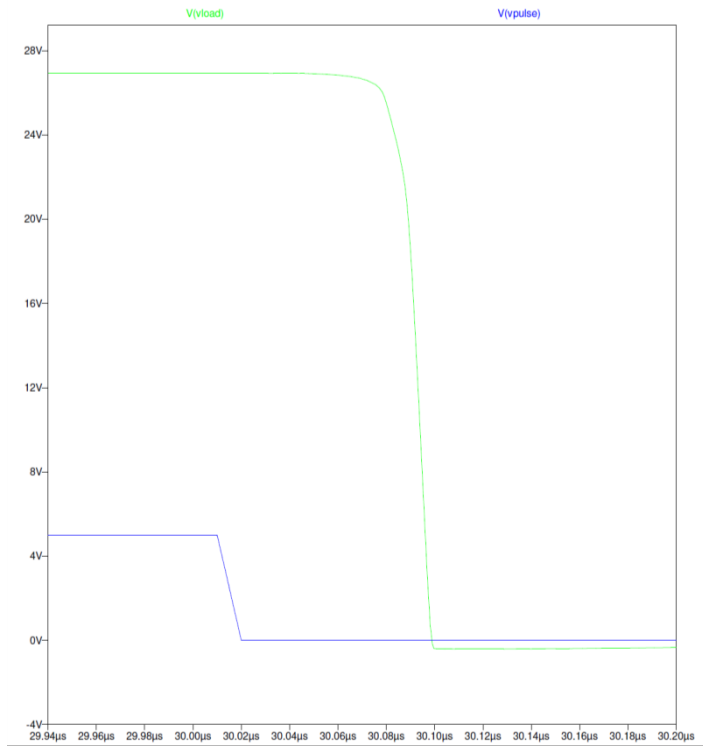
Figure 63: Fall Time of Switching Circuit - Simulation Results

5.2.7.2 RF Choke Inductor Design and Simulation

Typically, an air core inductor rather than a ferromagnetic core inductor is used as an RF choke for PAs due to having lower equivalent series resistance (ESR). However, air core inductors occupy a larger amount of space due to the large wire diameter for current handling along with the number of turns needed to realize high values of inductance. A ferromagnetic core inductor can realize higher inductances with less number of turns and occupy less space. Therefore, since the pre-amplifier and driver amplifier stages draw significantly less current than the power amplifier stage, a toroidal RF choke was implemented using 19 turns of 20 AWG wire on an Amidon FT50-61 core, as shown in Figure 64; thus, the power losses due to the higher ESR of the toroidal inductor will be less significant.
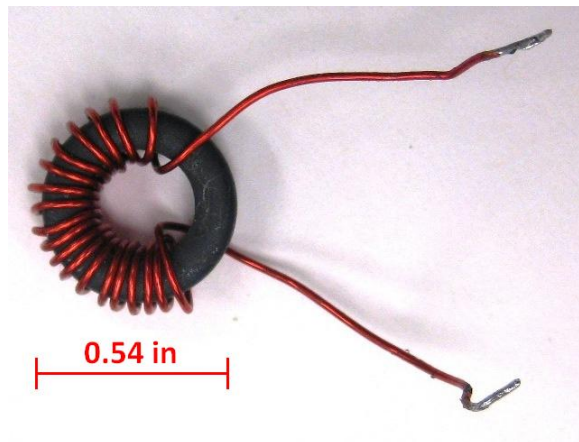
Figure 64: FT50-61 RF Choke Inductor

Furthermore, as seen in the ADS simulations in Appendix C, the self-resonant frequency (SRF) of the toroidal core is about 40 MHz, which means the RF choking reactance is actually capacitive from 140 to 210 MHz, ranging from -450 to -350 Ω. This 'capacitive' choking reactance should also be beneficial for the drain pulsing circuit, since the effective capacitance is less than 2.5 pF rather than an equivalent inductance of about 265 nH required to realize the same impedance using 'inductive' choking reactance.

Since the power amplifier stage experiences higher currents, the power loss due to the ESR is significant; therefore, an air core inductor was constructed using 15 turns of 20 AWG wire on a 141 mil form (i.e. inner diameter = 141 mil) as shown in Figure 65.   As seen in the ADS simulation results in Appendix C, the air core inductor provides an inductance reactance of 235 to 1100 Ω from 140 to 210 MHz.

0.55 in
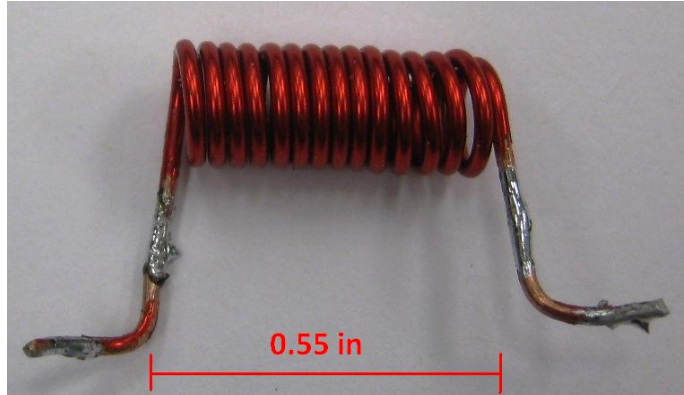
Figure 65: Air Core RF Choke Inductor

5.2.7.3 Results and Conclusions

The drain pulsing circuit was initially tested without the transistors in the circuit. An equivalent resistance of 25 Ω and capacitances from the datasheet for each transistor based on the equivalent circuit model of Figure 57 were used to initially test the drain pulsing circuit. Initial testing showed a rise-time of 110 ns and fall-time of 150 ns.

However, when the drain pulsing circuit was tested with the transistors in the circuit, the SD703 transistor was continuously being destroyed. It was determined that the SD703 transistor was being damaged due to the 'dv/dt capability' (or 'dv/dt induced turn-on') of a MOSFET transistor. This occurs when a positive, fast-changing voltage (dv) appears across the drain-to-source junction within a very short time interval (dt). The applied dv/dt results in an instantaneous current flow through the charge of the MOSFET parasitic drain-to-gate capacitance (C_{GD}) as depicted in Figure 57 and an induced voltage is generated at the gate of the MOSFET [31]. It was determined that the +28 V drain voltage was being induced onto the gate which exceeded the maximum V_{GS} of +20 V for the SD703. This conclusion was verified by applying a drain voltage of +15 V which would be less than the V_{GS} maximum of +20 V, then measuring the induced gate voltage transient at the gate of each transistor. As seen in Figure 66, the V_{GS} of the SD703 (blue trace) reaches a maximum voltage of +15 V.
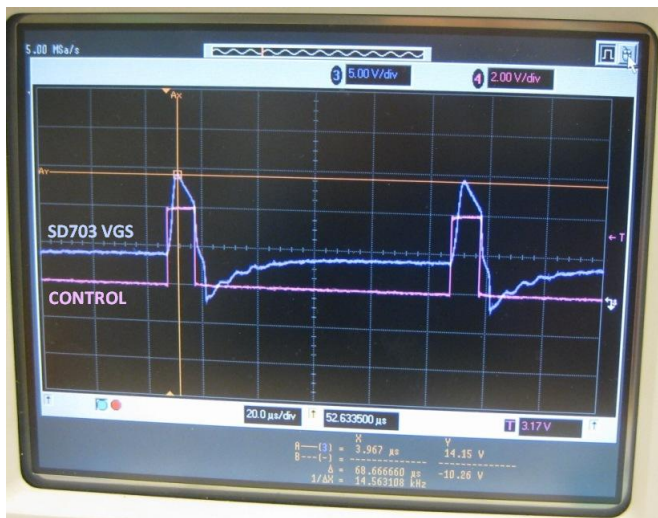
Figure 66: SD703 Induced V$_{GS}$ of Drain Pulsing Circuit

The induced gate voltage measurements of the SP202 and SP203 were similar; therefore, only the SP203 measurement is provided in Figure 67 to verify that neither the SP202 nor the SP203 experienced an induced gate voltage. This was because a RC drain-to-gate feedback stability network was used with a resistance of 221 Ω and capacitance of 10 nF for the SP202 and SP203 as depicted in Figure 68, which resulted in a larger charging time constant than the SD703. Ultimately, it was determined that drain pulsing would not be implemented due to the damaging effects of the dv/dt induced voltage; therefore, modifications were made to the design as described in Section 5.2.7.4.
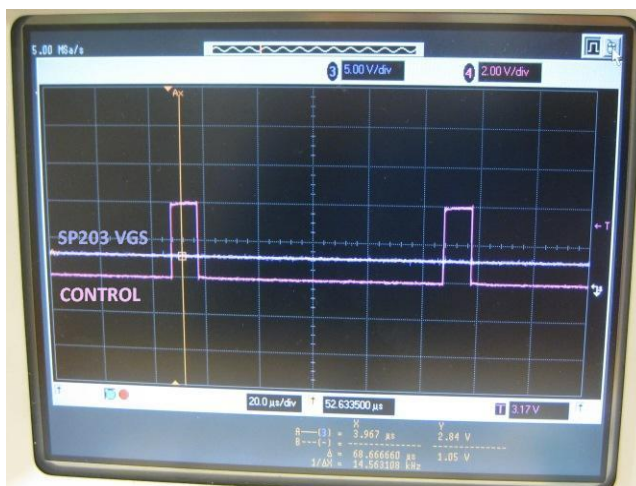
Figure 67: SP203 VGS of Drain Pulsing Circuit
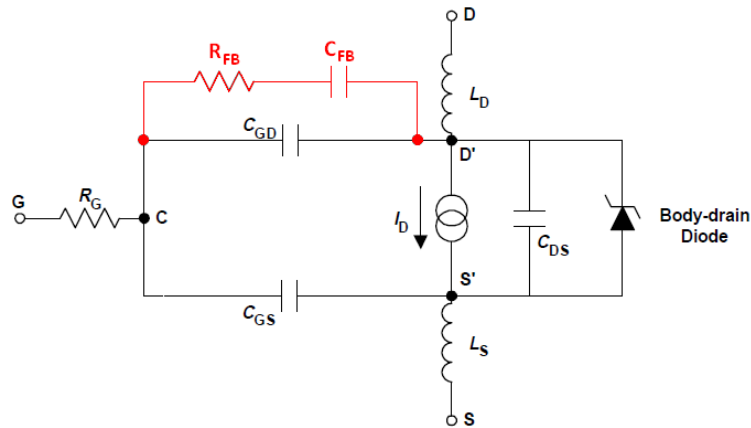
Figure 68: MOSFET Equivalent Circuit with RC Drain-to-Gate Feedback

Furthermore, the toroidal RF choke inductors caused ringing on the drain voltage; therefore, the RF choke inductors for the SP202 and SP203 were also changed.

5.2.7.4 Switching Circuit Modifications

In order to correct for the limitations of pulsing the drain current as described in Section 5.2.7.3, a modified version of the drain pulsing circuit was designed to switch the gate bias voltages as shown in Figure 69. However, once again due to the RC drain-to-gate, a large current was needed in order to switch the gate bias under 500 ns; thus, the voltage divider network used to generate the gate bias voltages for the SP202 and SP203 transistors would require as low as possible resistor values to maintain a higher current for quickly charging the RC drain-to-gate feedback network. Nevertheless, the higher current will generate larger power dissipation in the low valued biasing resistors; therefore, a trade-off between power dissipation and switching time was considered and the resulting value of 100Ω was selected as shown in Figure 69. The 100 Ω resistance should be implemented with at least three, parallel, 1 W, 300 Ω resistors for operation at 10-kHz PRF; however, in the prototype depicted in Figure 73 only one, ½ Watt resistor was used and the prototype was only tested at a PRF of 1-kHz. Also, a small valued resistor of 20 Ω was placed in parallel with the tuning potentiometer in order to protect the gate of the transistors from exceeding the maximum V_{GS} of +20 V. Furthermore, the bypass capacitors on the gate of each transistor were removed along with the 1 kΩ resistors.
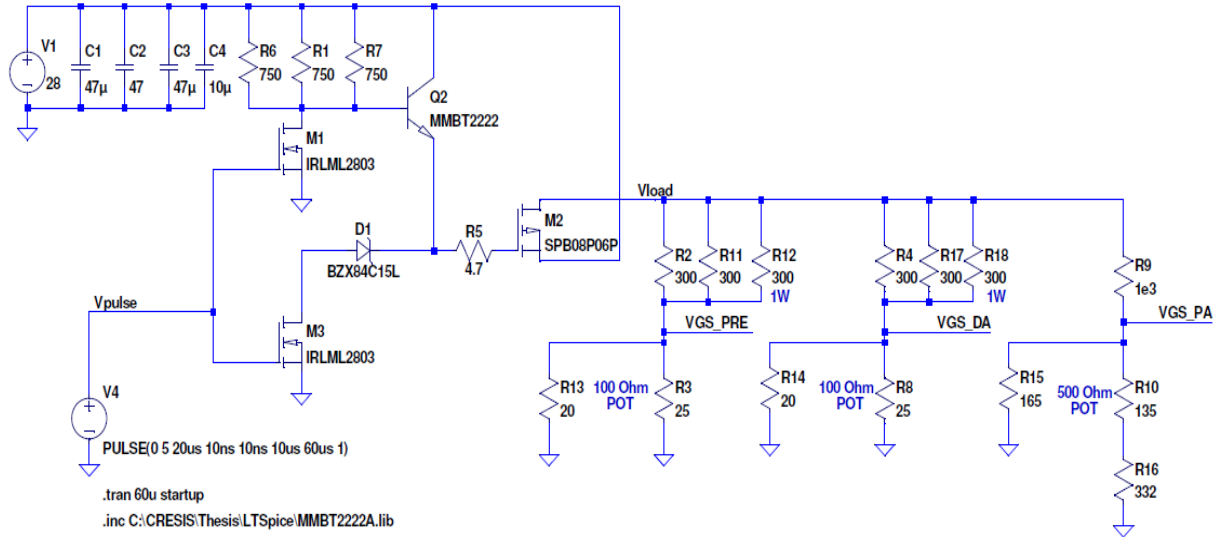
Figure 69: Gate Pulsing Circuit with MOSFET Biasing

5.2.8 Printed Circuit Board Design and Packaging

A 2-layer PCB layout of the original design (without the switching circuit modifications), shown in Figure 70, was generated using CadSoft's Eagle 4.16r2 PCB design software. The red area is the top layer, the blue area is the bottom layer and the pink lines are the cut layer. Figure 70 also provides an indication of how the PCB will fit into the T/R module enclosure along with mounting locations of external SMA connectors, DC bias pins and LEDs. The two yellow lines running across the board indicate the need for jumper wires from the '+28V' feed pin on the right side of the board to the '+28V_FEED' pad on the upper left side of the board. Another jumper wire is needed to connect the '+28V_CTRL' feed pin on the right side of the board to the '+28V_CTRL' pad on the upper left side of the board. Creating a multi-layer board would easily remove the need for jumper wires; however, in the prototype design it was deemed unnecessary and not cost effective. Furthermore, the RF choke inductors described in Section 5.2.7.1 are not shown in Figure 70; however, the mounting locations are indicated as 'L1_A' & 'L1_B' for L1, 'L4_A' & 'L4_B' for L4, 'L7_A' & 'L7_B' for L7 and 'L8_A' & 'L8_B' for L8. The PCB schematic used to generate the original prototype layout can be found in Appendix D. Also, the PCB schematic for the modified prototype is also located in Appendix D.
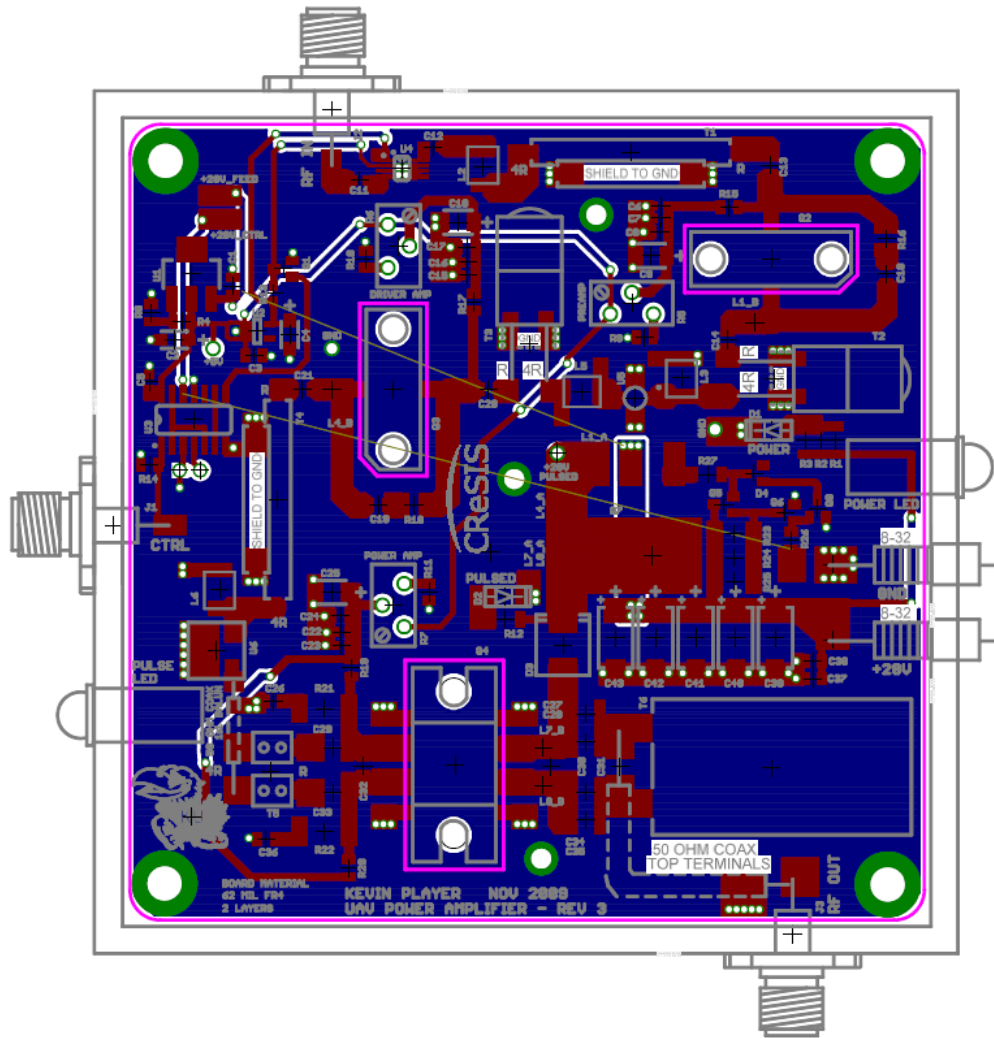
Figure 70: Power Amplifier Board Layout with Packaging (without modifications)

5.2.9   Prototype Fabrication

CadSoft's Eagle 4.16r2 was also used to generate the gerber files needed for manufacturing the circuit board. Table 10 provides the file extensions that were submitted to Sierra Proto-Express for manufacturing. It is important to note that the '.cut' file includes the internal cuts so the transistors can be mounted to the mounting plate. However, the '.drd' file also includes the drill holes for mounting the transistor. By assumption, the manufacturer may ignore the internal cuts and only drill the holes; thus, it is a good idea to inform the manufacturer that the internal cuts need to be removed even though there are drill holes inside the boundaries of the cuts. It may seem counter-productive to include drill holes inside a cutout region; however, the drill holes are included on the '.drd' file because the same gerber files are used in order to fabricate the mounting plate, shown in Figure 71 and Figure 72. The mounting plate was fabricated in the CReSIS machine shop on the Haas CNC mill with the assistance of Dennis Sundermeyer. The mounting plate was made out of aluminum for lighter weight. The height of the mounting pads

for the SP202 and SP203 transistors are at the same level as the corner mounting hole pads (i.e. 0.250"), while the SD703 height is 60-70 mils shorter since the SD703 transistor mounting base is about twice the thickness of the SP202 and SP203 transistors. The switching circuit modifications could be made using the original prototype fabricated board; thus, the prototype in the T/R module enclosure with switching circuit modifications is shown in Figure 73 and the schematic for the modified prototype is provided in Appendix D. The total weight of the PA with enclosure and mounting plate as shown in Figure 73 was 12.9 ounces which is less than the desired 16 ounces. A significant amount of the weight is from the aluminum mounting plate which can be removed if surface mount potentiometers are used. The primary purpose of the mounting plate is to elevate the board so the through-hole potentiometer leads do not touch the bottom of the enclosure. If surface mount (or other packaging style) potentiometers are used, they need to be rated at 1/2 watt power or greater. Also, since there are traces on the bottom layer of the 2-layer board, there needs to be some kind of insulation layer to protect the trace from touching the bottom of the enclosure.

Table 10: Gerber Files Used for Manufacturing

File Extension	Description
.top	Top Layer
.tsm	Top Solder Mask
.tsk	Top Silk Screen
.bot	Bottom Layer
.bsm	Bottom Solder Mask
.cut	Cut Layer
.drd	Drill File
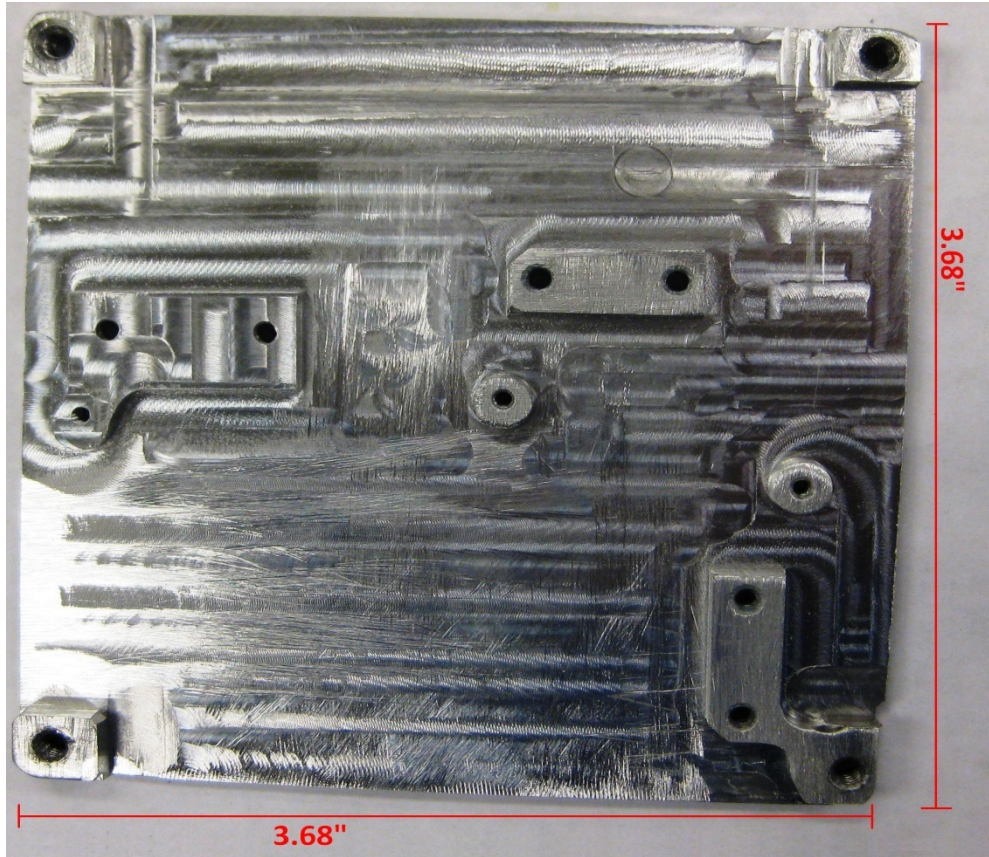
Figure 71: Manufactured Mounting Plate – Top View
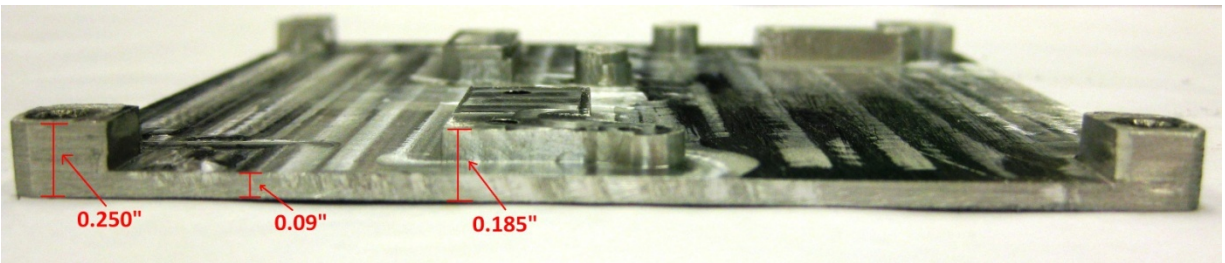
Figure 72: Manufactured Mounting Plate - Side View
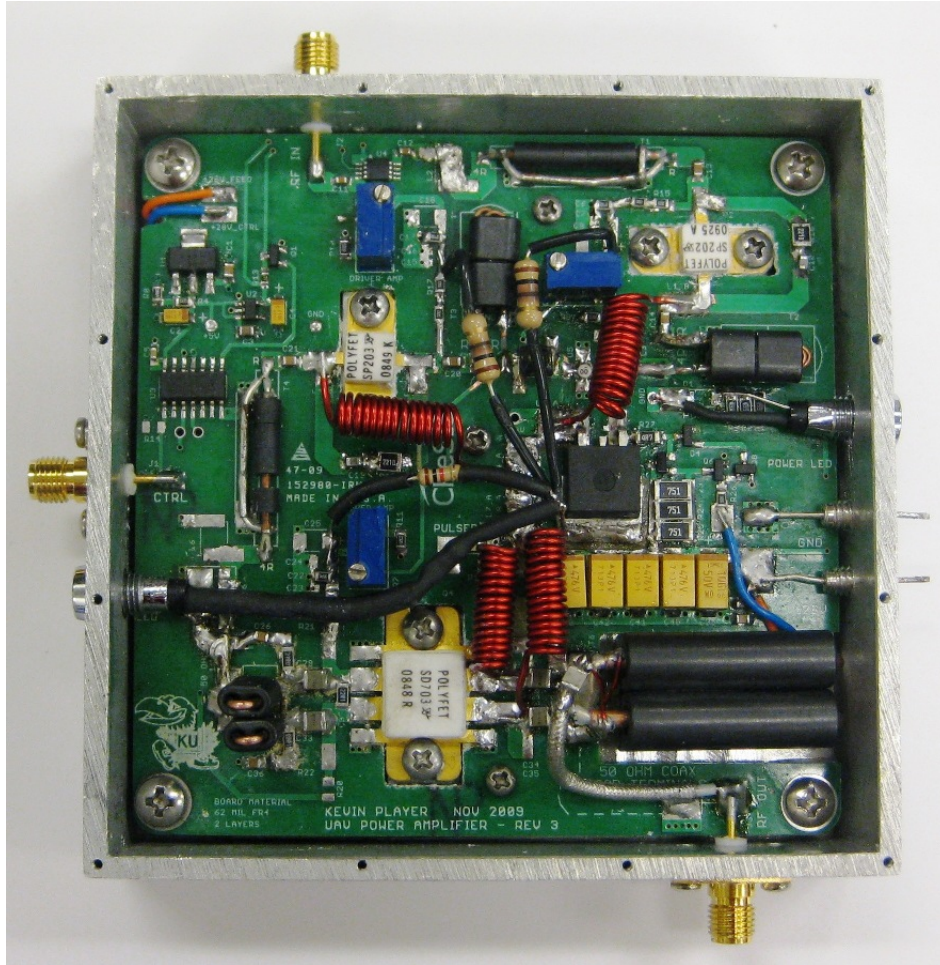
Figure 73: Final Prototype with Modifications in T/R Module Enclosure

5.3   UAV VHF Power Amplifier Characterization

5.3.1   Gain and Output Power

The maximum output power was measured in continuous-wave (CW) mode; thus, a constant +5 V was applied to the 'TTL control line' of the UAV PA. The maximum input power of +4.5 dBm (2.82 mW) was used to determine the maximum output power of the UAV PA. The large signal output power of the UAV PA was measured from 140 to 160 MHz and 180 to 210 MHz at 5-MHz intervals using the HP437B power detector according to the block diagram of Figure 74. The 50-dB of attenuation varied about 0.5 dB on the network analyzer which results in a power variance of about 12 W at 50 dBm. To obtain a more accurate measure of the attenuation, the 50-dB of attenuation was measured independently and the average of five measurements was used at each frequency interval. The large signal gain was calculated using the maximum output power measurements (i.e. at Pin = +4.5 dBm) at each frequency interval. The large signal gain and output power results are shown in Figure 75. The plots for the PA output power transfer

function at each 5-MHz interval from 140 to 160 MHz and 180 to 210 MHz are provided in Appendix F.
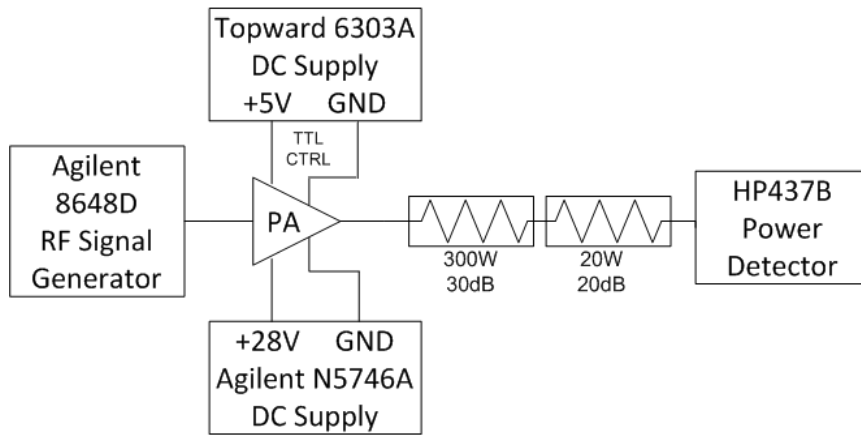
Figure 74: Large Signal Output Power and Gain Measurement Block Diagram
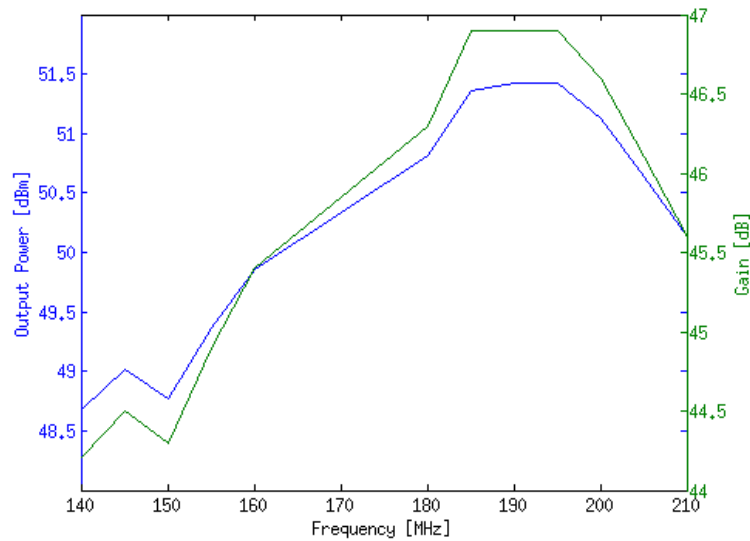
Figure 75: Large Signal Output Power and Gain Measurement Results

5.3.2 Linearity

The linearity of the UAV PA was determined for each frequency using the 1-dB compression point, as well as, the 2nd and 3rd harmonic power levels. The linearity measurement setup is similar to the output power setup of Figure 74, except the HP8565E spectrum analyzer (SA) was used to measure the output power level of the fundamental and harmonic components. Even though the power detector provides a more accurate power measurement of the fundamental, the SA was used in order to measure the harmonics and express their power level in relation to the fundamental output power level (i.e. dBc) using the same power measurement instrument.

Using the measurement setup of Figure 76, the UAV PA transfer characteristic was measured from 140 to 160 MHz and 180 to 210 MHz at 5-MHz intervals to determine the 1-dB compression point at each frequency. The measured results were then linearly interpolated between each 5-MHz interval to generate an estimated 1-dB compression point at each frequency, as shown in Figure 77. Since the maximum output power of the DDS is +4.5 dBm, the PA was designed to reach the 1-dB compression point at or around +4.5 dBm input power. Typically, for a transmitter PA , the 1-dB compression point is stated as output power. However, to illustrate the accuracy of the PA design relative to the maximum output power of the DDS, the 1-dB compression point is given as input power as shown in Figure 77. The transfer characteristic for each of the measured frequencies is provided in Appendix F. It is important to note that the drain current (I_d) was also measured at the same time as the transfer characteristic measurements in order to be used for the efficiency calculations of Section 5.3.3.
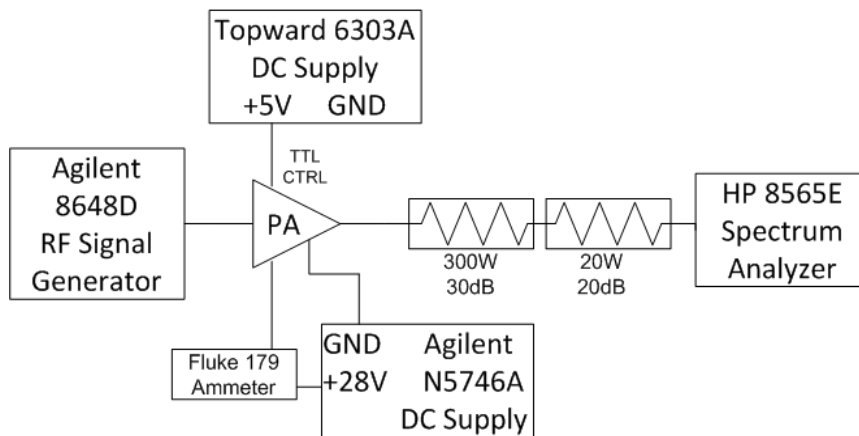
Figure 76: Linearity and Efficiency Measurement Block Diagram
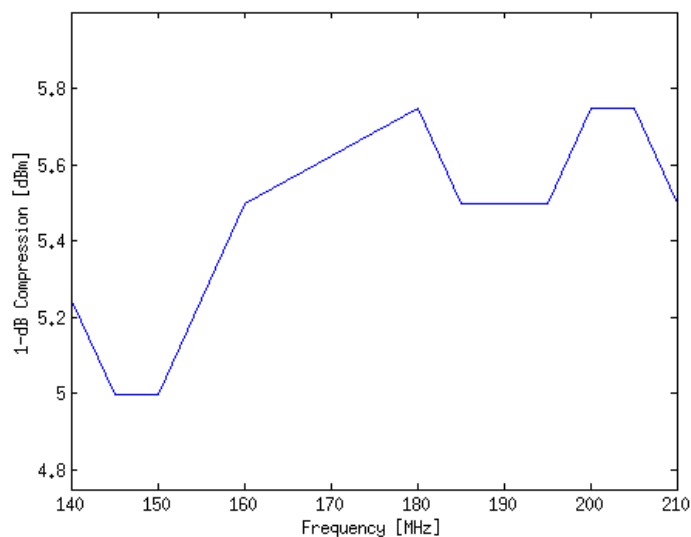
Figure 77: 1-dB Compression Point Measured Results with Interpolation

As seen in Figure 77, the PA operates very closely to the 1-dB compression point when the +4.5 dBm maximum input power is used. Therefore, since the purpose of the predistortion is to linearize the PA at or around the 1-dB compression point, the 2nd and 3rd harmonic output power levels were measured using the +4.5 dBm maximum input power level from 140 to 210 MHz at 5-MHz intervals using the setup in Figure 76. The measured results were then linearly interpolated between each 5-MHz interval to generate an estimated 2nd and 3rd harmonic output power level at each frequency, as shown in Figure 78.
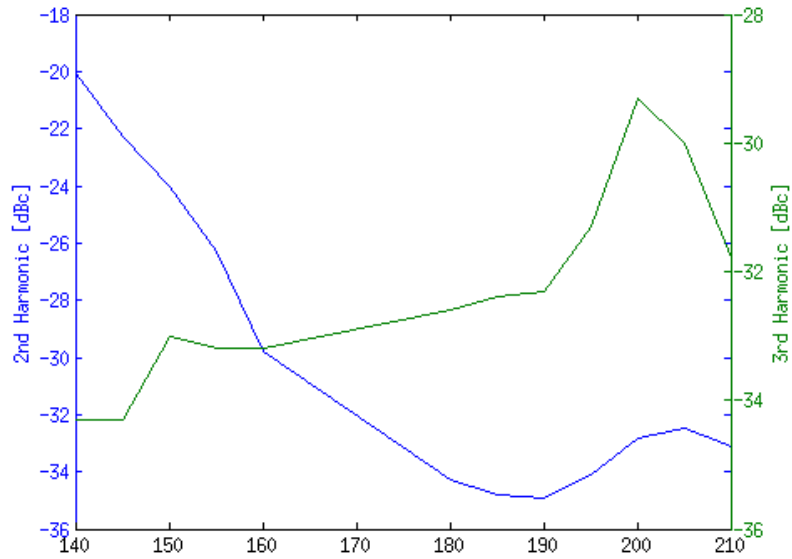
Figure 78: Measured 2nd and 3rd Harmonic Output Power Levels

5.3.3 Efficiency

The PAE was calculated using Equation 5 where $P_{in}(f_o)$, $P_{out}(f_o)$ and I_d were taken from the transfer characteristic data measured in Section 5.3.2. The P_{DC} was calculated by multiplying the +28 V supply voltage times the measured drain current, I_d. The graphs of the measured PAE from 140 to 160 MHz and 180 to 210 MHz at 5-MHz intervals are provided in the Appendix. Once again, the efficiency measurements shown in Figure 79 were calculated at +4.5 dBm rather than at the 1-dB compression point since the PA operates very closely to the 1-dB compression point when the +4.5 dBm maximum input power is used.
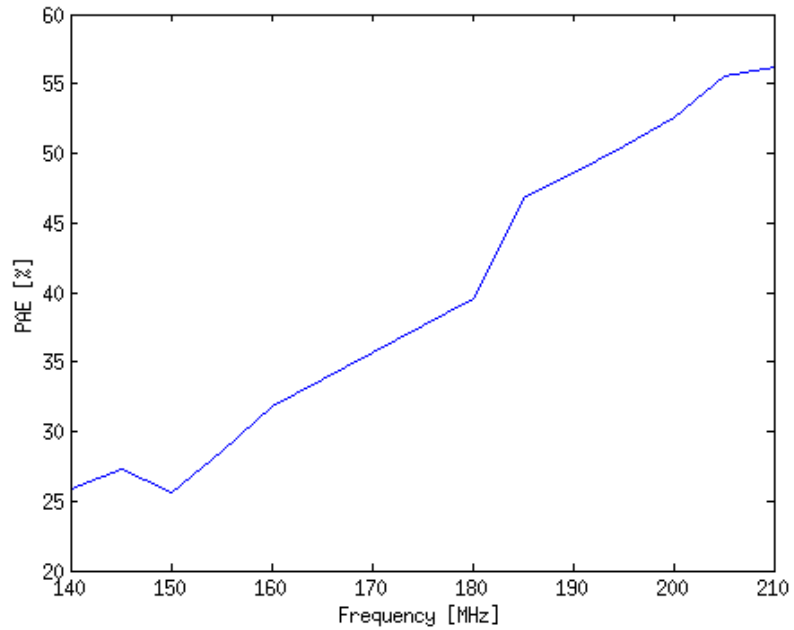
Figure 79: Measured PAE at +4.5 dBm Maximum Input Power

5.3.4 Input and Output Impedance

The input and output impedance were measured using a 1-port network analyzer measurement. An .s1p file was created for each measurement and the results were plotted in ADS. The return loss and VSWR for the input and output are shown in Figure 80 and Figure 81, respectively.
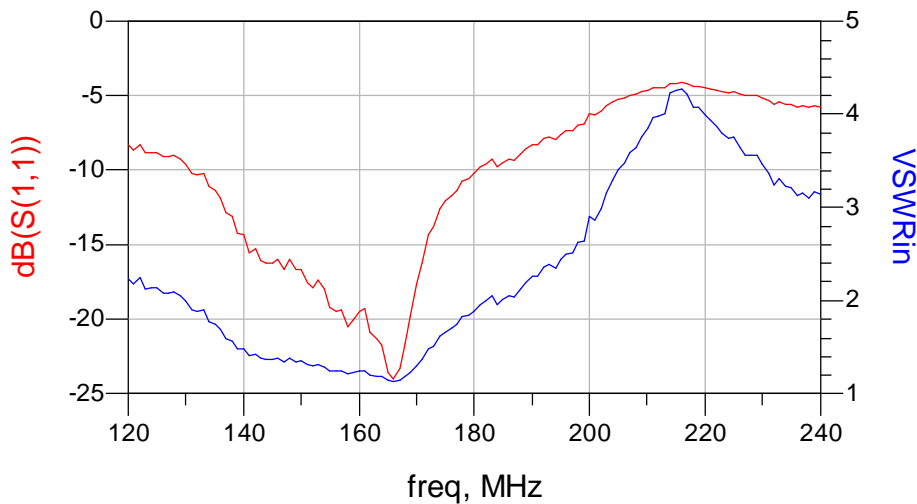
Figure 80: Large Signal S11 and Input VSWR Measurement
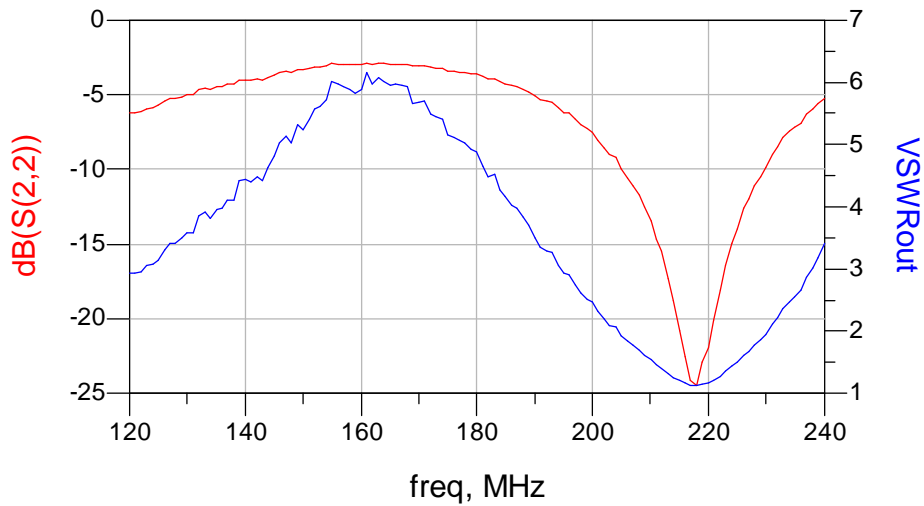
Figure 81: Large Signal S22 and Output VSWR Measurement

5.3.5 Operational Bandwidth and Frequency Response

The measurement results of Figure 75 indicate the PA has a 3-dB bandwidth of 70-MHz with a frequency response from 140 to 210 MHz.

5.3.6 High-Speed Pulsing Circuit

The measurement setup of Figure 82 was used to determine the rise and fall time of the switching circuit. The rise time was determined by decreasing the 'TTL' lines in the DDS gui under the 'Receive' tab until the transmit chirp waveform began to display an RC charging distortion. The rise time was measured to be 391 ns using a 10-μs, 1-kHz pulse. Since the rise time was less than 1 μs, it was assumed the 1-μs pulse would generate similar results.

The fall time was determined by decreasing the 'TTL' lines in the DDS gui under the 'Receive' tab which would generate the TTL control signal to turn off the PA about half-way through the pulse as seen in Figure 84. The fall time was measured to be 79.1 ns using a 10-μs, 1-kHz pulse. Once again since the fall time was less than 1 μs, it was assumed the 1-μs pulse would generate similar results.
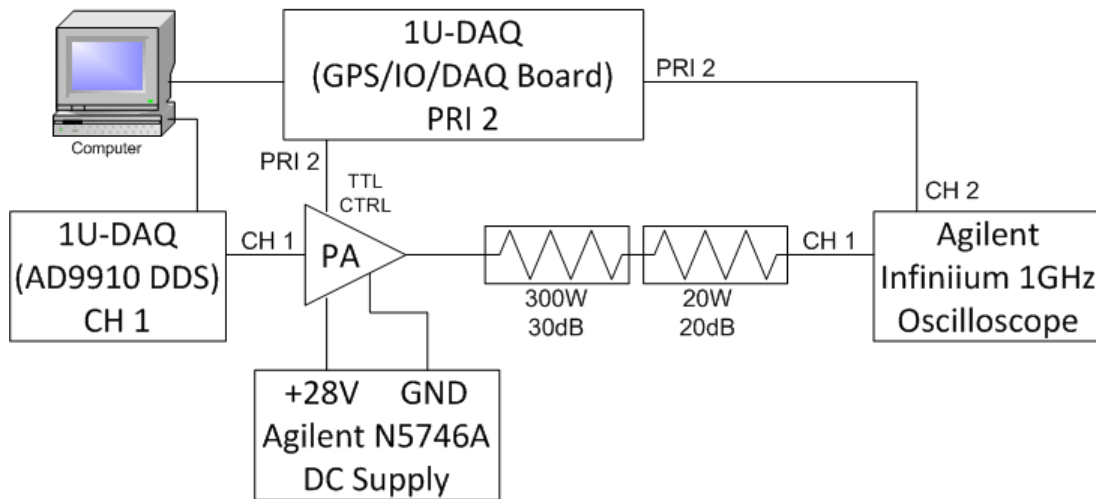
Figure 82: High Speed Pulsing Circuit Measurement Block Diagram
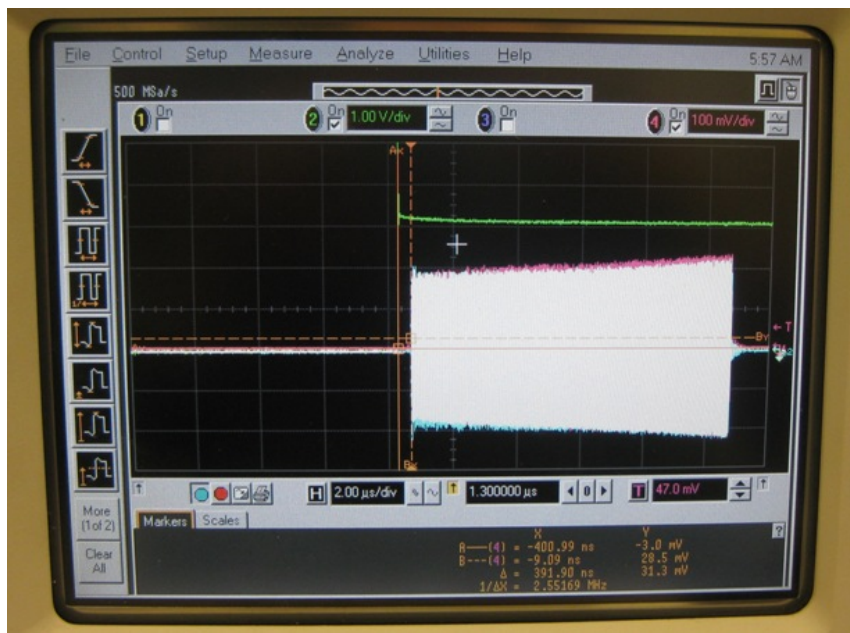
Figure 83: Rise Time Measurement of 10us Pulse
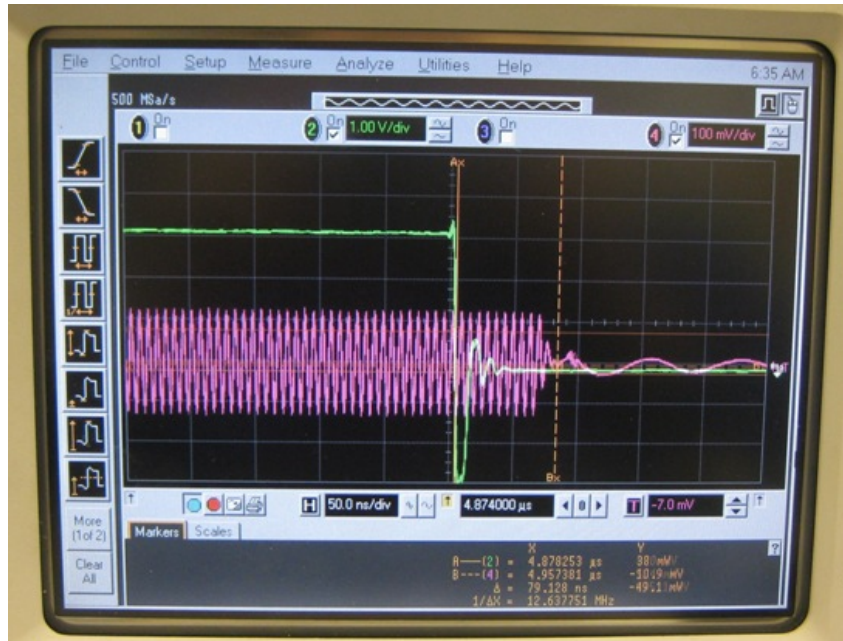
Figure 84: Fall Time Measurement of 10us Pulse

As a final test, the PA was continuously operated in the lab from 180 to 210 MHz at maximum output power (i.e. DDS weight = 65,535) with a 10-μs pulse at 1-kHz PRF for 12 hours with no instabilities or over-heating problems of any of the components. The average current was 117 mA. Although a 10-kHz PRF would increase the current consumption and stress the PA even more, the current through the biasing resistors would be too large for the ½ W resistors. Therefore, it was assumed that the PA would operate at 10-kHz PRF with the modified PA design implemented with the three, paralleled 300 Ω, 1 W biasing resistors.

Chapter 6: Implementation of Digital Predistortion for the UAV VHF Power Amplifier

6.1   *Experimental Setup*
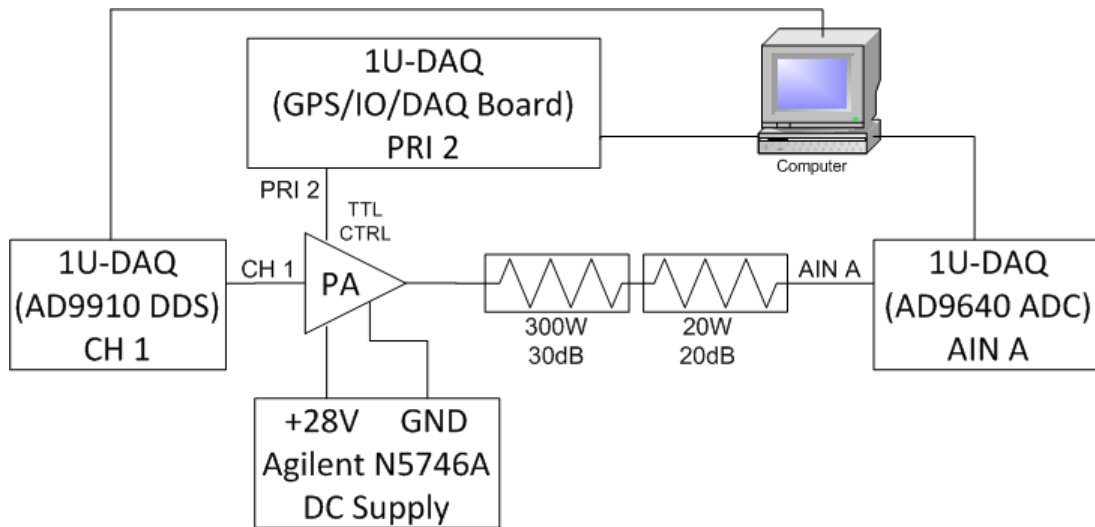
Figure 85: Amplitude Digital Predistortion Measurement Block Diagram

After initializing the 1U-DAQ digital system, the following procedure was followed to implement the digital predistortion for the PA using a 10-µs pulse at 1-kHz PRF:

1.   Run 'dds_gui_pd.pro' to start the DDS gui and generate the waveform

      a.   Select 'tukey' under the 'window' drop-down menu

      b.   Insert operating frequency range with 'start' and 'stop' frequency boxes

      c.   Insert '65535' in the 'weight' box

      d.   Select 'on' under the 'DDS Output' drop-down menu

2.   Run 'daq_gui.pro' to start the DAQ gui to record the data

      a.   Record data for 10 seconds

3.   On the DDs gui, turn off the DDS output by selecting 'off' under the 'DDS Output' drop-down menu

4.   Open MATLAB program 'DDS_Predistortion.m' to determine predistortion algorithm

      a.   Adjust weighting for the appropriate operating frequency range

      b.   Run 'DDS_Predistortion.m' to generate 'amp.txt' file

5.   Generate the predistorted waveform

      a.   On the DDS gui, Select 'predistort' under the 'window' drop-down menu

      b.   Select 'update' and wait until 'DDS Output' box is not greyed out

      c.   Select 'on' under the 'DDS Output' drop-down menu

6.   On the DAQ gui, record data for 10 seconds

6.2   *Experimental Results and Discussion*

Using the procedure in Section 6.1 for 140 to 160 MHz, it can be seen from Figure 86, Figure 87 and Figure 88 that the predistortion algorithm provided the necessary correction to linearize the PA output signal and provide a relatively flat response across the FM chirp signal.
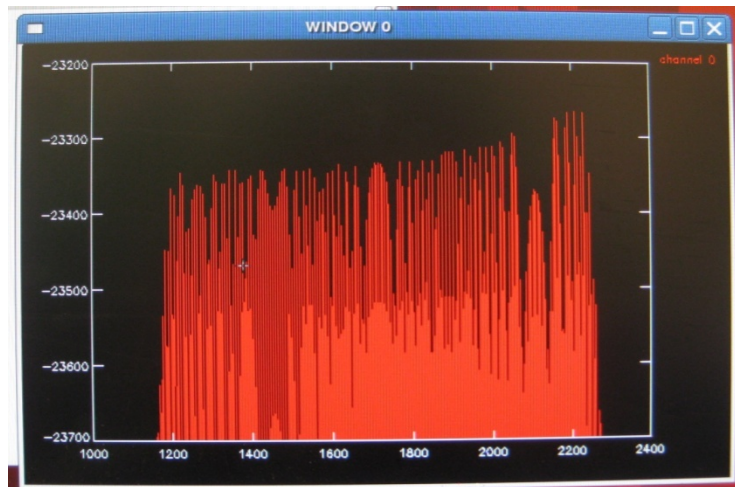
Figure 86: DAQ GUI Screenshot of 140 to 160 MHz Pulse with Tukey Weighting (Zoomed)
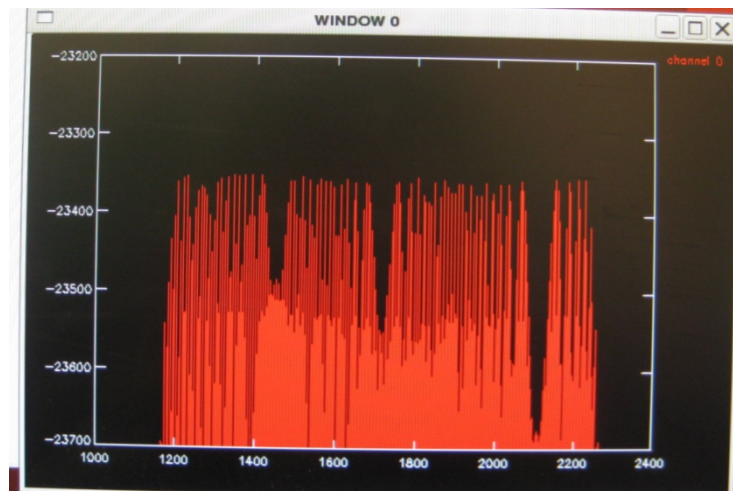
Figure 87: DAQ GUI Screenshot of 140 to 160 MHz Pulse with Amplitude Predistortion (Zoomed)
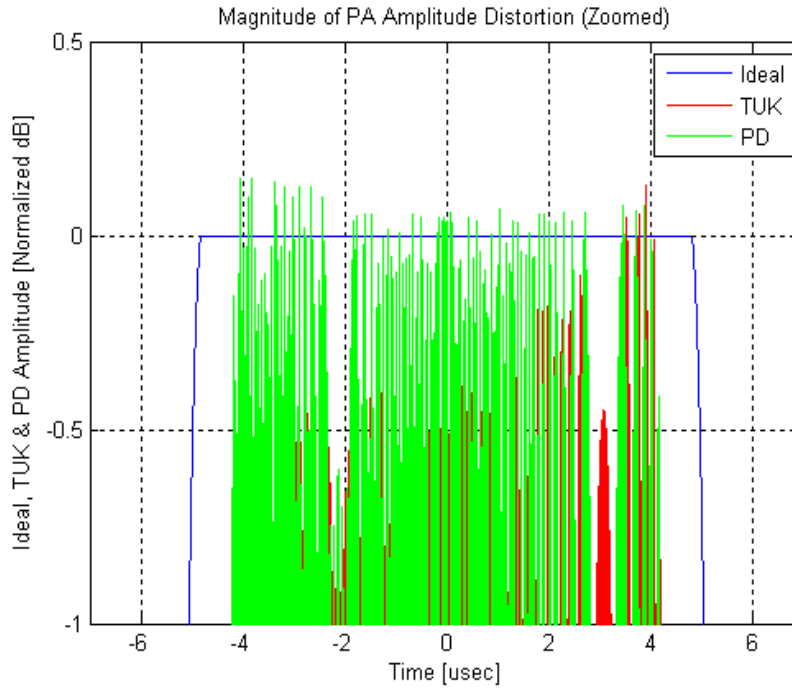
Figure 88: 140 to 160 MHz MATLAB Amplitude Comparison (Zoomed)

The relative improvement in side-lobe levels can be seen by comparing Figure 89, Figure 90, and Figure 91. Figure 89 shows the pulse compression response of the DDS generated waveform without predistortion using only a tukey window on the transmit signal (i.e. the signal shown in Figure 86) and a Blackmann[2] window on the receive signal. Figure 90 shows the pulse compression response of the DDS generated waveform with predistortion which also includes a tukey window on the transmit (i.e. the signal shown in Figure 87) and a Blacmann[2] window on the receive signal. Figure 91 provides a comparison of the range side-lobe levels with the non-predistorted signal on the left side and the predistorted signal on the right side. The level of improvement varied depending on the time range; therefore, Table 11 provides a summary of the side-lobe level of improvement based on the time range.

Table 11: Pulse Compression Side-Lobe Level Improvement

Time Range [μs]	Side-lobe Level Improvement [dB]
-10 to -5	~17
-5 to 0	~ 10
0 to 5	~1
5 to 10	~6
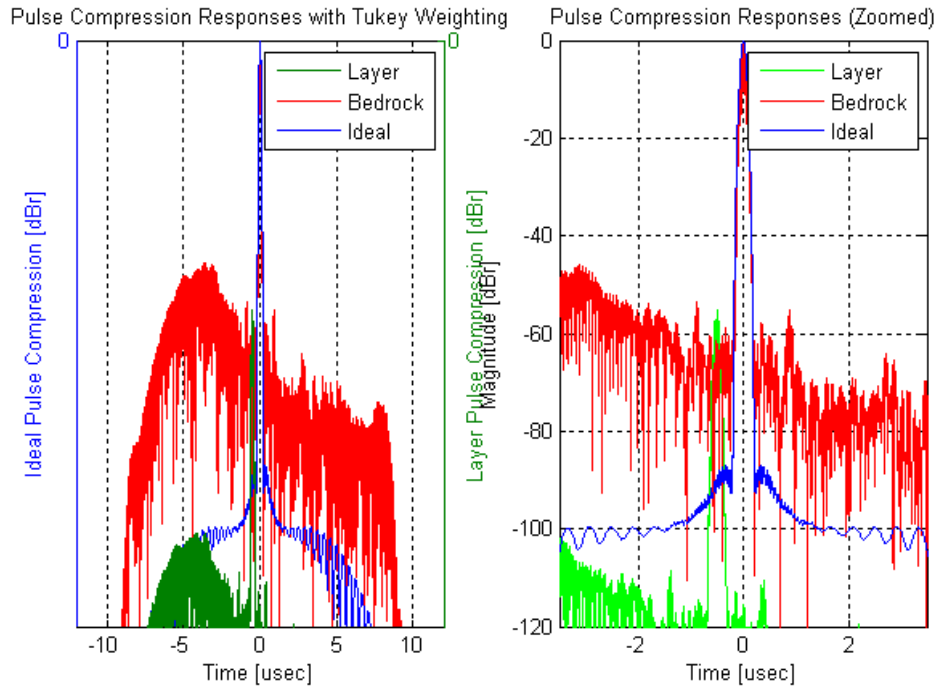
Figure 89: 140 to 160 MHz Pulse Compressed Response with Tukey Weighting
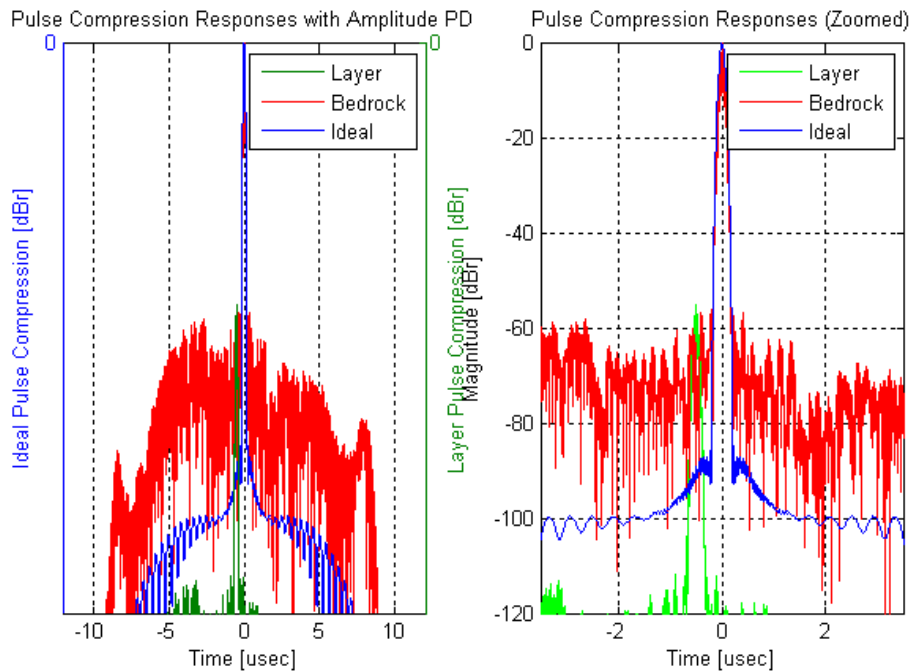
Figure 90: 140 to 160 MHz Pulse Compressed Response with Amplitude Predistortion
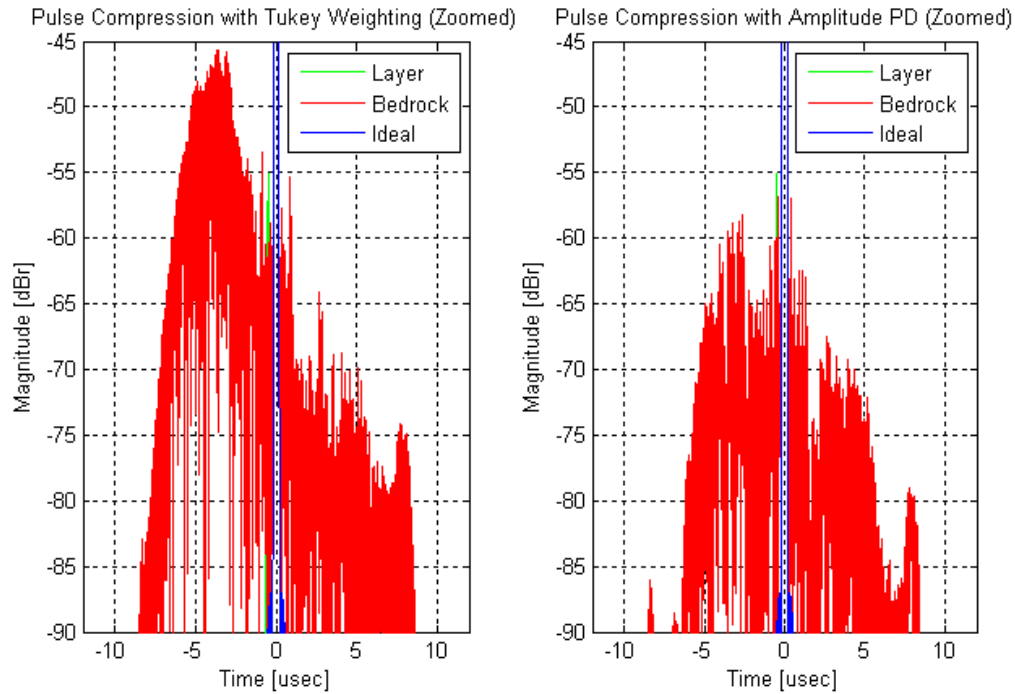
Figure 91: 140 to 160 MHz Pulse Compressed Response Side Lobe Levels

Chapter 7: Conclusion

7.1   Summary

A high-speed, pulsed, VHF power amplifier was developed and linearized using memoryless digital predistortion (DP) to obtain high linearity and high efficiency. The frequency response was 140 to 210 MHz with 70-MHz bandwidth and PAE ranging from 25 to 56%. The switching circuit reduced current consumption to 117 mA (or 3.28 W at +28 V) for a 10-μs, 1-kHz PRF pulse at maximum output power with a rise time of 391 ns and a fall time of 79.1 ns. The digital predistortion (DP) linearization decreased far range side-lobe levels below -57 dBc, with a maximum reduction of 17 dB over the Tukey (transmit) and Blackmann^2 (receive) windowing alone. Altogether, all the design constraints and objectives were either met or exceeded.

7.2   Significant Contributions

It is often humorously stated that power amplifier design is an art more than a science due to the complex combination of components, erratic behavior and difficult design constraints. The transistors themselves are difficult to model and often behave differently depending on numerous external conditions, such as temperature. Also, the constant battle of design trade-offs, specifically the famous struggle of obtaining high linearity with high efficiency. This research work has conquered many of the major difficulties in power amplifier design, as well as, radar transmitter design. The following list emphasizes some of the more significant contributions of this research work.

1. A 3-stage PA was designed using discrete transistors with each requiring an input and output matching network that were each designed using custom, hand-made, miniature impedance transformers.

2. The switching circuit reduced the current consumption to a mere 117 mA (3.28 W at +28 V) for a 10-µs pulse at 1-kHz PRF even at maximum power output of over 100 W. This allowed the PA to operate without the need of any heatsink, decreasing the weight and size of the PA which is crucial for UAV applications.

3. The final prototype design fit into an enclosure measuring 3 ¾" x 3 ¾" x 1" and weighed only 12.9 ounces including the enclosure and mounting plate.

4. The memoryless digital predistortion improved the linearity on average by 7 dB implemented with amplitude predistortion only. This improvement was obtained without adding any external circuitry or components to the radar system since a digital system is already utilized in CReSIS radar systems.

7.3   Future Work

Although the results of this research produced a prototype design which met the design requirements, further contributions can be made to improve the design before final deployment. The following areas of improvement or study are topics for future research work:

A. The prototype was fabricated using a 2-layer board and modifications to the switching circuit were made using this 2-layer board; therefore, a new multi-layer board layout can be designed for easier implementation and a more compact package. The modified schematic provided in Appendix D should be used for the new board layout.

B. The modified switching circuit uses a PMOS transistor for providing the high current switching. Using an NMOS transistor for the high current switching circuit instead of a PMOS transistor could decrease the switching time. NMOS devices traditionally have lower R_{ds_on} than PMOS devices; however, they require higher gate voltages for biasing which introduces more complicated circuitry. Since the PMOS switching circuit satisfied the design requirements, there was no need for further investigation of alternative switching techniques. For all these reasons, the NMOS switching circuit can be considered for future research.

C. As explained in Chapter 4, only memoryless digital predistortion (DP) was implemented in this research work. Thus, DP with memory (i.e. amplitude and phase correction) can be considered for future work to better improve the linearity of the PA. Furthermore, as explained in Chapter 3, memory effects can also be corrected by implementing adaptive DP.

D. Due to the extreme sensitivity of the depth-sounding radar systems used by CReSIS and the proximity of the PA module to the antenna, the UAV PA should be tested for electromagnetic compatibility (EMC).

APPENDIX A

'dds_gui_pd.pro' IDL Program Code

```
; DDS GUI --------------------------------------------------------------- ;

; --------------------------------------------------------------- ;
; ROUTINE TO GENERATE GUI STRUCTURE ;
; --------------------------------------------------------------- ;
FUNCTION gen_gui_struct
  nwaveforms = 2
  nchannels = 1
  nlines = 4
  max_weight = 65535
  RETURN, { $
  transmit:   {wid:         0, $
               tab_name:    'transmit', $
               sel_name:    'waveform', $
               sel_number:  nwaveforms, $
               sel_index:   0, $
               procedure:   'gui=config_ddsreg(gui)', $
               extra:       0, $
               properties: { $
                            enable:     {wid:0, name:'DDS Output       ',
units:['off','on'],               value:INTARR(nwaveforms),
procedure:'gui=start_stop(gui)'}, $
                            trig:       {wid:0, name:'Trigger          ',
units:['internal','external'],  value:INTARR(nwaveforms),
procedure:''}, $
                            fclk:       {wid:0, name:'DDS frequency   ',
units:' MHz',                 value:FLTARR(nwaveforms)+1000,
procedure:'gui=const_clk(gui)'}, $
                            sync:       {wid:0, name:'MultiChip Sync  ',
units:' ',                    value:INTARR(nwaveforms),
procedure:''}, $
                            prf:        {wid:0, name:'PRF             ',
units:' Hz',                  value:FLTARR(nwaveforms)+1000,
procedure:'gui=const_prf(gui)'}, $
                            fstart:     {wid:0, name:'start frequency ',
units:' MHz',                 value:FLTARR(nwaveforms)+140,
procedure:''}, $
                            fstop:      {wid:0, name:'stop frequency  ',
units:' MHz',                 value:FLTARR(nwaveforms)+220,
procedure:''}, $
                            length:     {wid:0, name:'pulse length    ',
units:' us',                  value:FLTARR(nwaveforms)+10,
procedure:''}, $
                            taper:      {wid:0, name:'window          ',
units:['none','rect','hann','tukey','inv','predistort','user'],
value:INTARR(nwaveforms)+1,         procedure:''}, $
                            user:       {wid:0, name:'user window (rs)',
units:'',                     value:STRARR(nwaveforms),
procedure:''}, $
                            weight:     {wid:0, name:'weight          ',
units:'',                     value:INTARR(nwaveforms)+max_weight,
procedure:''}}}, $
```

```
 $
 digital:    {wid:        0, $
             tab_name:    'ttl lines', $
             sel_name:    'line', $
             sel_number:  nlines, $
             sel_index:   0, $
             procedure:   'gui=config_ttlreg(gui)', $
             extra:       0, $
             properties: { $
                          start:      {wid:0, name:'start          ',
units:' samples',           value:INTARR(nlines)+1,
procedure:''}, $
                          length:     {wid:0, name:'length         ',
units:' samples',           value:INTARR(nlines)+1500,
procedure:''}}}, $
  $
  receive:   {wid:        0, $
             tab_name:    'receive', $
             sel_name:    'channel', $
             sel_number:  nchannels, $
             sel_index:   0, $
             procedure:   'gui=config_daqreg(gui)', $
             extra:       0, $
             properties: { $
                          start:      {wid:0, name:'start index    ',
units:' samples',           value:INTARR(nchannels)+281,
procedure:''}, $
                          length:     {wid:0, name:'record length  ',
units:' samples',           value:INTARR(nchannels)+5000,
procedure:''}, $
                          presum:     {wid:0, name:'presums        ',
units:' ',                  value:INTARR(nchannels)+1,
procedure:''}, $
                          bits:       {wid:0, name:'output range   ',
units:['0','1','2','3'],     value:INTARR(nchannels),
procedure:''}}}, $
  $
  gui_name:   'HARDWARE GUI', $
  tab_number: 3, $
  tab_index:  0, $
  procedure:  'stop_on_exit, gui', $
  extra:      0}
END
; ------------------------------------------------------------------------ ;
;FUNCTION serial_flush
;  x = SIZE(serial_read(1))
;  WHILE x[0] DO x = SIZE(serial_real(1))
;END
;
;PRO dds_write, x
;  num = SIZE(x,/N_ELEMENTS)
;  serial_write, x
;  y = serial_read(1)
;  n = SIZE(y)
;  IF n[0] EQ 0 THEN PRINT, 'Serial error - no response'
;  IF y NE x[num-1] THEN PRINT, 'Serial error - bad return'
;END
```

```
FUNCTION config_daqreg, gui
  FOR i0 = 0,gui.transmit.sel_number-1 DO BEGIN
    gui.transmit.properties.enable.value[i0] = 0
  ENDFOR
  WIDGET_CONTROL, gui.transmit.properties.enable.wid, SET_COMBOBOX_SELECT=0
  serial_write, ['77'XB,'34'XB,'00'XB]
  start = gui.receive.properties.start.value[gui.receive.sel_index]
  length = gui.receive.properties.length.value[gui.receive.sel_index]
  presum = gui.receive.properties.presum.value[gui.receive.sel_index]
  bits = gui.receive.properties.bits.value[gui.receive.sel_index]
  print, bits
  daq_change, length, start, presum, bits

  ;gui = config_ddsreg(gui)
  RETURN, gui
END
; ---------------------------------------------------------------------- ;
; ROUTINE TO TURN DDS ON AND OFF ;
; ---------------------------------------------------------------------- ;
FUNCTION start_stop, gui
  FOR i0 = 0,gui.transmit.sel_number-1 DO BEGIN
    gui.transmit.properties.enable.value[i0] =
gui.transmit.properties.enable.value[gui.transmit.sel_index]
  ENDFOR
  IF gui.transmit.properties.enable.value[gui.transmit.sel_index] EQ 0 THEN
BEGIN
    PRINT, 'DDS Output off'
    serial_write, ['77'XB,'34'XB,'00'XB] ;Wavegen Config Reg - [7:3]-NA, [2]-
1int/0ext timing, [1]-1int/0ext clk, [0]-1en/0disabled
  ENDIF ELSE BEGIN
    daq_reset;, gui.receive.properties.start.value[gui.receive.sel_index],
gui.receive.properties.length.value[gui.receive.sel_index]
    PRINT, 'DDS Output on'
    data = '05'XB +
BYTE(24*gui.transmit.properties.trig.value[gui.transmit.sel_index])
    print, data
    serial_write, ['77'XB,'34'XB,data] ;enabled using internal timing and
external clock.
  ENDELSE
  RETURN, gui
END
; ---------------------------------------------------------------------- ;
PRO stop_on_exit, gui
  FOR i0 = 0,gui.transmit.sel_number-1 DO BEGIN
    gui.transmit.properties.enable.value[i0] = 0
  ENDFOR
  gui    = start_stop(gui) ;force all channels to stop before updating
  serial_close
END
; ---------------------------------------------------------------------- ;
; ROUTINE TO CHANGE ALL WAVEFORMS CLOCKS (THERE IS NO REASON WHY THEY SHOULD
BE DIFFERENT ;
; ---------------------------------------------------------------------- ;
FUNCTION const_clk, gui
  PRINT, 'Changing DDS clock for all waveforms'
```

```
   FOR i0 = 0,gui.transmit.sel_number-1 DO BEGIN
      gui.transmit.properties.fclk.value[i0] =
gui.transmit.properties.fclk.value[gui.transmit.sel_index]
   ENDFOR
   RETURN, gui
END
; -------------------------------------------------------------------- ;
; ROUTINE TO CHANGE ALL WAVEFORMS PRFs (THERE IS NO REASON WHY THEY SHOULD BE
DIFFERENT ;
; -------------------------------------------------------------------- ;
FUNCTION const_prf, gui
   PRINT, 'Changing PRF for all waveforms'
   FOR i0 = 0,gui.transmit.sel_number-1 DO BEGIN
      gui.transmit.properties.prf.value[i0] =
gui.transmit.properties.prf.value[gui.transmit.sel_index]
   ENDFOR
   RETURN, gui
END
; -------------------------------------------------------------------- ;
; ROUTINE TO GENERATE DEFAULT DDS REGISTERS ;
; -------------------------------------------------------------------- ;
FUNCTION get_ddsreg
RETURN,  {cfr1       :['00'XB,  $ ;00000000 Control Function Register 1
                       '40'XB,  $ ;01000000 7-RAM enable, 6:5-RAM destination
(00=frequency,01=phase,10=amplitude,11=polar),  4:0-Open
                       '00'XB,  $ ;00000000 7-Manual OSK, 6-Inv. Sinc, 5-
Open, 4:1-Int. Profile Control, 0-Sine output
                       '60'XB,  $ ;01100000 7-Load LRR@IO, 6:5-Autoclear
ramp/phase, 4:3-Clear ramp/phase, 2-Load ARR@IO, 1-OSK Enable, 0-Auto OSK
                       '02'XB], $ ;00000010 7:4-PD digital/DAC/REFCLK
in/AuxDAC, 3-Ext. PD control, 2-Open, 1-SDIO in only, 0-LSB first
           cfr2       :['01'XB,  $ ;00000001 Control Function Register 2
                       '00'XB,  $ ;00000000 7:2-Open, 1-Ramp over pin enable,
0-Amp profile scale enable
                       '4E'XB,  $ ;01001110 7-Int. IO active, 6-SYNC_CLK
enable, 5:4-Ramp destination, 3-Ramp enable, 2:1-Ramp no-dwell high/low, 0-
Read eff FTW
                       '00'XB,  $ ;00000000 7:6-IO rate control, 5:4-Open, 3-
PDCLK enable, 2-PDCLK invert, 1-TxEnable invert, 0-Open
                       '20'XB], $ ;00100000 7-Matched latency enable, 6-Data
hold last, 5-SYNC Validation disable, 4-P-port enable, 3:0-FM gain
           cfr3       :['02'XB,  $ ;00000010 Control Function Register 3
                       '1D'XB,  $ ;00000000 7:6-Open, 5:4-DRV0, 3-Open, 2:0-
VCO Select
                       '3F'XB,  $ ;00000000 7:6-Open, 5:3-Icp, 2:0-Open
                       'C1'XB,  $ ;00000000 7:6-REFCLK input divider
bypass/reset, 5:1-Open, 0-PLL enable
                       '20'XB], $ ;00000000 7:1-PLL N, 0-Open
           auxdac     :['03'XB,  $ ;00000011 Auxiliary DAC Control
                       '00'XB,  $ ;00000000 7:0-Open
                       '00'XB,  $ ;00000000 7:0-Open
                       '00'XB,  $ ;00000000 7:0-Open
                       'FF'XB], $ ;01111111 7:0-Full scale current
           io_update  :['04'XB,  $ ;00000100 I/O Update Rate
                       'FF'XB,  $ ;11111111
                       'FF'XB,  $ ;11111111
                       'FF'XB,  $ ;11111111
```

```
                        'FF'XB], $ ;11111111
        ftw       :['07'XB, $ ;00000111 Frequency Tuning Word
                   '04'XB, $ ;00000000
                   '00'XB, $ ;00000000
                   '00'XB, $ ;00000000
                   '00'XB], $ ;00000000
        pow       :['08'XB, $ ;00001000 Phase Offset Word
                   '80'XB, $ ;00000000
                   '00'XB], $ ;00000000
        asf       :['09'XB, $ ;00001001 Amplitude Scale Factor
                   '00'XB, $ ;00000000 7:0-Amplitude ramp rate [15:8]
                   '00'XB, $ ;00000000 7:0-Amplitude ramp rate [7:0]
                   '00'XB, $ ;00000000 7:0-Amplitude scale factor [13:6]
                   '00'XB], $ ;00000000 7:2-Amplitude scale factor [5:0],
1:0-Amplitude step size
        sync      :['0A'XB, $ ;00001010 Multichip Sync
                   '00'XB, $ ;00000000 7:4-Sync validation delay, 3-Rec
enable, 2:1-Generator enable/polarity, 0-Open
                   '00'XB, $ ;00000000 7:2-State preset, 1:0-Open
                   '00'XB, $ ;00000000 7:3-Output generator delay, 2:0-
Open
                   '00'XB], $ ;00000000 7:3-Input receiver delay, 2:0-
Open
        ramp_limit :['0B'XB, $ ;00001011 Digital Ramp Limit
                   '40'XB, $ ;00000000
                   '00'XB, $ ;00000000
                   '00'XB, $ ;00000000
                   '00'XB, $ ;00000000
                   '20'XB, $ ;00000000
                   '00'XB, $ ;00000000
                   '00'XB, $ ;00000000
                   '00'XB], $ ;00000000
        ramp_step :['0C'XB, $ ;00001100 Digital Ramp Step Size
                   '00'XB, $ ;00000000
                   '00'XB, $ ;00000000
                   '00'XB, $ ;00000000
                   '01'XB, $ ;00000000
                   '00'XB, $ ;00000000
                   '00'XB, $ ;00000000
                   '00'XB, $ ;00000000
                   '01'XB], $ ;00000000
        ramp_rate :['0D'XB, $ ;00001101 Digital Ramp Rate
                   '00'XB, $ ;00000000
                   '01'XB, $ ;00000000
                   '00'XB, $ ;00000000
                   '01'XB], $ ;00000000
        profile0  :['0E'XB, $ ;00001110 RAM Profile 0
                   '00'XB, $ ;00000000 Open
                   '00'XB, $ ;00000000 Step rate [15:8]
                   '04'XB, $ ;00000000 Step rate [7:0]
                   '20'XB, $ ;00000000 End address[9:2]
                   '80'XB, $ ;00000000 7:6-End address[1:0], 5:0-Open
                   '00'XB, $ ;00000000 Start address[9:2]
                   '00'XB, $ ;00000000 7:6-Start address[1:0], 5:0-Open
                   '21'XB]}   ;00000000 7:6-Open, 5-No-dwell high, 4-
Open, 3-Zero xing, 2:0-RAM mode
END
```

```
; ------------------------------------------------------------------------- ;
; FUNCTION FOR READING IN EXTERNAL WEIGHTS FOR PREDISTORTION ;
; ------------------------------------------------------------------------- ;
FUNCTION load_file_EXT, ram_size
  PRINT, 'PREDISTORTION ON'
  PRINT, 'RAM SIZE = ',ram_size
  OPENR, lun, 'amp.txt', /GET_LUN   ; amp.txt contains DDS weighting values
calculated in matlab
  weights = MAKE_ARRAY(ram_size,VALUE=0)  ; need to enforce that last value
of array = 0
  READF, lun, weights
  FREE_LUN, lun
  RETURN, weights
END


; ------------------------------------------------------------------------- ;
; ROUTINE TO CONFIGURE THE DDS REGISTERS AND RAM TO CURRENT GUI SETTINGS ;
;-------------------------------------------------------------------------- ;
FUNCTION config_ddsreg, gui
  WIDGET_CONTROL, gui.transmit.properties.enable.wid, SENSITIVE=0 ;don't
allow dds enable select while configuring
  FOR i0 = 0,gui.transmit.sel_number-1 DO BEGIN
    gui.transmit.properties.enable.value[i0] = 0
  ENDFOR

  IF gui.transmit.properties.enable.wid NE 0 THEN BEGIN
    WIDGET_CONTROL, gui.transmit.properties.enable.wid, SET_COMBOBOX_SELECT=0
  ENDIF

  gui    = start_stop(gui) ;force all channels to stop before updating
  gui    = const_clk(gui)  ;make sure the DDS clock is updated for all
channels
  gui    = const_prf(gui)  ;make sure the PRF is updated for all channels
  ;get all current parameters from the gui
  index      = gui.transmit.sel_index
  trig       = gui.transmit.properties.trig.value[gui.transmit.sel_index]
  clk        = gui.transmit.properties.fclk.value[index]*1e6
  prf        = gui.transmit.properties.prf.value[index]
  fclk       = gui.transmit.properties.fclk.value[index]*1e6
  presum     = gui.receive.properties.presum.value[gui.receive.sel_index]
  pri_arr    = val2bytarr((clk/16)/prf - 1.0)
  ttlram     = BYTARR(32)
  print, gui.digital.sel_number-1
  FOR i0 = 0,gui.digital.sel_number-1 DO BEGIN
    print, i0
    ttlram[i0*4+0] = BYTE(gui.digital.properties.start.value[i0] MOD 256)
    ttlram[i0*4+1] = BYTE(gui.digital.properties.start.value[i0]/256)
    ttlram[i0*4+2] = BYTE(gui.digital.properties.length.value[i0] MOD 256)
    ttlram[i0*4+3] = BYTE(gui.digital.properties.length.value[i0]/256)
  ENDFOR
;  ;set ttl lines
;  ttlram = ['0'XB,'00'XB,'00'XB,'01'XB, $  ;
;           '0'XB,'00'XB,'00'XB,'01'XB, $  ;
;           '0'XB,'00'XB,'00'XB,'01'XB, $  ;
;           '0'XB,'00'XB,'00'XB,'01'XB, $  ;
;           '0'XB,'00'XB,'00'XB,'01'XB, $  ;
```

```
;           '0'XB,'00'XB,'00'XB,'01'XB, $   ;
;           '0'XB,'00'XB,'00'XB,'01'XB, $   ;
;           '0'XB,'00'XB,'00'XB,'01'XB]     ;
  init_dds, pri_arr, presum, ttlram
  FOR ich = 0,1 DO BEGIN
    ddsreg = get_ddsreg()    ;get default DDS regesters
    ddsreg.sync[4] = BYTE(gui.transmit.properties.sync.value[ich])
    fstart     = gui.transmit.properties.fstart.value[ich]*1e6
    fstop      = gui.transmit.properties.fstop.value[ich]*1e6
    length     = gui.transmit.properties.length.value[ich]/1e6
    fstart_arr = val2bytarr(LONG(2d0^32*fstart/fclk))
    fstop_arr  = val2bytarr(LONG(2d0^32*fstop/fclk))
    frate      = (256*ddsreg.ramp_rate[1] + ddsreg.ramp_rate[2])/(0.25*fclk)
    fstep      = (fstop-fstart)*frate/length
    fstep_arr  = val2bytarr(LONG(2d0^32*fstep/fclk))
    ram_step   = MIN(WHERE(length*.25*fclk/((MAKE_ARRAY(65535,/INDEX)+1) LT
750))+1
    ram_step_arr = val2bytarr(ram_step)
    ram_size     = FIX(FLOOR(length*fclk/(4.0*ram_step)))
    end_arr      = val2bytarr(ram_size*64)
    ddsreg.ramp_limit[1:8] = [fstop_arr,fstart_arr]
    ddsreg.ramp_step[1:8]  = [fstep_arr,fstep_arr]
    ddsreg.profile0[2:3]   = ram_step_arr[2:3]
    ddsreg.profile0[4:5]   = end_arr[2:3]
    ram_size = FIX(ddsreg.profile0[4])*4+ddsreg.profile0[5]/64+1
    taper = gui.transmit.properties.taper.value[ich]
    up = ((fstop-fstart)/fclk)*(1.0-MAKE_ARRAY(ram_size-
1,/FLOAT,/INDEX)/(ram_size-2.0))+(fstart/fclk)
    rs = ram_size
    weight = gui.transmit.properties.weight.value[ich]
    IF ich = 0 THEN BEGIN
      PRINT ,'ram size = ', rs
      PRINT ,'weight = ', weight
    ENDIF
    CASE taper OF
      0:    win = [MAKE_ARRAY(ram_size,VALUE=weight)]              ;NONE
      1:    win = [MAKE_ARRAY(ram_size-1,VALUE=weight),0]          ;RECT
      2:    win = weight*[HANNING(ram_size-1),0]                   ;HANNING
      3:    win = weight*[tukey(ram_size-1,.2),0]                  ;TUKEY
      4:    win = weight*[(up/SIN(2.0*!DPI*up)),0]                 ;INV
      5:    win = load_file_EXT()                                  ;PREDISTORT
      6:    success = EXECUTE('win='+gui.transmit.properties.user.value[ich])
      ;USER
      ELSE: win = inv*[MAKE_ARRAY(ram_size-1,/INTEGER,VALUE=30000),0]
    ENDCASE
    amp = BYTARR(4*ram_size)
    FOR i0 = 0,ram_size-1 DO BEGIN
      amp[i0*4+0] = win[i0] / 256
      amp[i0*4+1] = win[i0] MOD 256
      amp[i0*4+2] = 0
      amp[i0*4+3] = 0
    ENDFOR
    ram = ['16'XB,amp]
    config_channel, ddsreg, ram, ich
  ENDFOR

  serial_write, ['77'XB,'40'XB,'00'XB]
```

```
  WIDGET_CONTROL, gui.transmit.properties.enable.wid, SENSITIVE=1
  RETURN, gui
END
; -------------------------------------------------------------------------- ;
PRO init_dds, pri_arr, presum, ttlram
  ;Set GPS
  serial_write, ['77'XB,'A0'XB,'01'XB]
  WAIT, 3.0
  serial_write, ['77'XB,'A0'XB,'00'XB]
  WAIT, 0.1
  serial_write, ['77'XB,'A1'XB,'00'XB]
  serial_write, ['73'XB,'A2'XB,'00'XB,'20'XB,ttlram]
  WAIT, 0.1
  ;Set PRI
  serial_write, ['77'XB,'35'XB,pri_arr[1]]
  serial_write, ['77'XB,'36'XB,pri_arr[2]]
  serial_write, ['77'XB,'37'XB,pri_arr[3]]
  WAIT, 0.1
  ;Set EPRI (# of PRFs per cycle)
  serial_write, ['77'XB,'38'XB,'00'XB]
  serial_write, ['77'XB,'39'XB,BYTE(presum-1)]
  WAIT, 0.1
  ;Set delay between PRF and DDS trig
  serial_write, ['77'XB,'3A'XB,'02'XB]
  serial_write, ['77'XB,'3B'XB,'2B'XB]
  WAIT, 0.1
  ;Set the per-waveform values
  ram = [ $
          '00'XB,'03'XB $ ;idle 00
$
         ,'00'XB,'02'XB $ ;0x08 01
         ,'00'XB,'02'XB $
         ,'00'XB,'02'XB $
         ,'00'XB,'02'XB $
         ,'FF'XB,'02'XB $
         ,'00'XB,'02'XB $
         ,'00'XB,'02'XB $
         ,'00'XB,'02'XB $
$
         ,'FF'XB,'02'XB $ ;0x00 09
         ,'00'XB,'02'XB $
         ,'00'XB,'02'XB $
         ,'00'XB,'02'XB $
         ,'00'XB,'02'XB $
         ,'00'XB,'02'XB $
         ,'00'XB,'02'XB $
$
         ,'00'XB,'02'XB $ ;0x00 17
         ,'00'XB,'02'XB $
         ,'00'XB,'02'XB $
         ,'00'XB,'02'XB $
         ,'00'XB,'02'XB $
         ,'00'XB,'02'XB $
         ,'00'XB,'02'XB $
$
```

```
        ,'00'XB,'01'XB $ ;0x08 25
        ,'00'XB,'01'XB $
        ,'00'XB,'01'XB $
        ,'00'XB,'01'XB $
        ,'FF'XB,'01'XB $
        ,'00'XB,'01'XB $
        ,'00'XB,'01'XB $
        ,'00'XB,'01'XB $
$
        ,'00'XB,'01'XB $ ;0x00 33
        ,'00'XB,'01'XB $
        ,'00'XB,'01'XB $
        ,'00'XB,'01'XB $
        ,'00'XB,'01'XB $
        ,'00'XB,'01'XB $
        ,'00'XB,'01'XB $
$
        ,'00'XB,'01'XB $ ;0x00 41
        ,'00'XB,'01'XB $
        ,'00'XB,'01'XB $
        ,'00'XB,'01'XB $
        ,'00'XB,'01'XB $
        ,'00'XB,'01'XB $
        ,'00'XB,'01'XB $
$
        ,'00'XB,'03'XB $ ;idle 49
        ,'00'XB,'03'XB $ ;idle 50
]

  serial_write, ['77'XB,'4B'XB,'00'XB]
  serial_write, ['73'XB,'4C'XB,'00'XB,'66'XB,ram]
  WAIT, 0.1
  ;send waveform config values
  byte0 = BYTE(ISHFT(presum,-4))
  byte1 = BYTE(ISHFT(presum,4) MOD 256)
  serial_write, ['77'XB,'50'XB,byte0] ; WF1 1 PRESUM
  serial_write, ['77'XB,'51'XB,byte1]
  serial_write, ['77'XB,'52'XB,'40'XB] ; NO zero/pi addr=0
  serial_write, ['77'XB,'53'XB,'00'XB] ; addr = 0
  WAIT, 0.1
  serial_write, ['77'XB,'54'XB,'00'XB]
  serial_write, ['77'XB,'55'XB,'00'XB]
  serial_write, ['77'XB,'56'XB,'40'XB]
  serial_write, ['77'XB,'57'XB,'32'XB] ; addr = 50
  WAIT, 0.1
  ;Reset DDS
  serial_write, ['77'XB,'40'XB,'0D'XB] ;dds reset & bit-bang mode
  serial_write, ['77'XB,'40'XB,'01'XB] ;dds enabled & bit-bang mode
  WAIT, 0.1
END
PRO config_channel, ddsreg, ram, channel
  CS_N = BYTE('FF'XB - ISHFT('01'XB,channel))
  ;Set DDS registers
  names = TAG_NAMES(ddsreg)
  FOR i0 = 0,SIZE(TAG_NAMES(ddsreg),/N_ELEMENTS)-1 DO BEGIN
```

```
    number = BYTE(SIZE(ddsreg.(i0),/N_ELEMENTS))
    serial_write,['77'XB,'41'XB,CS_N]
    serial_write,['73'XB,'43'XB,'00'XB,number,ddsreg.(i0)]
    serial_write,['77'XB,'41'XB,'FF'XB]
    serial_write,['77'XB,'3E'XB,'00'XB] ;update DDS
    WAIT, 0.1
  ENDFOR
  ;Set DDS ram
  ram_size = SIZE(ram,/N_ELEMENTS)
  num_high = BYTE((ram_size)/256)
  num_low  = BYTE((ram_size) MOD 256)
  serial_write,['77'XB,'41'XB,CS_N]
  serial_write,['73'XB,'43'XB,num_high,num_low,ram]
  serial_write,['77'XB,'41'XB,'FF'XB]
  serial_write,['77'XB,'3E'XB,'00'XB] ;update DDS
  WAIT, 0.1
  ;Enable DDS ram
  ddsreg.cfr1[1] = ddsreg.cfr1[1] OR  '80'XB ;set ram enable bit
  number = BYTE(SIZE(ddsreg.cfr1,/N_ELEMENTS))
  serial_write,['77'XB,'41'XB,CS_N]
  serial_write,['73'XB,'43'XB,'00'XB,number,ddsreg.cfr1]
  serial_write,['77'XB,'41'XB,'FF'XB]
  serial_write,['77'XB,'3E'XB,'00'XB] ;update DDS
  WAIT, 0.1
END
; ------------------------------------------------------------------------- ;
; ROUTINE TO SEND THE SERIAL CONFIGURATION TO THE DDS BOARD ;
; ------------------------------------------------------------------------- ;
PRO send_serial, ddsreg0, ram0, ddsreg1, ram1, pri_arr
;  ;set ttl lines
;  ram = ['0'XB,'00'XB,'00'XB,'01'XB, $  ;
;         '0'XB,'00'XB,'00'XB,'01'XB, $  ;
;         '0'XB,'00'XB,'00'XB,'01'XB, $  ;
;         '0'XB,'00'XB,'00'XB,'01'XB, $  ;
;         '0'XB,'00'XB,'00'XB,'01'XB, $  ;
;         '0'XB,'00'XB,'00'XB,'01'XB, $  ;
;         '0'XB,'00'XB,'00'XB,'01'XB, $  ;
;         '0'XB,'00'XB,'00'XB,'01'XB]    ;
;
;  serial_write, ['77'XB,'A1'XB,'00'XB]
;  serial_write, ['73'XB,'A2'XB,'00'XB,'20'XB,ram]
;  WAIT, 0.1
END
; ------------------------------------------------------------------------- ;
; MAIN PROGRAM - LOAD GENERIC GUI CODE, SET GLOBALS, AND WRAP TO GENERIC GUI
------------------------------------------------------------------------- ;
PRO dds_gui_pd
  daq_setup
  serial_open
  RESTORE, 'gui_base.sav'
  generic_gui
END
; ------------------------------------------------------------------------- ;
```

'DDS_Predistortion.m' MATLAB Program Code

```matlab
% Written by: Kevin Player
% December 2009
% Generates weighting amplitudes for DDS & writes to 'amp.txt' file

clc; close all; clear all;
Max_wt    = 65535;      % Maximum weighting in AWG
 ram_size  = 630;        % Size of RAM used to create DDS waveform (from IDL window)
%********************** DETERMINE AMPLITUDE VALUES ******************
%  Amplitude weights for testing DDS
% rect_test      = round((Max_wt./2).*ones(1,ram_size).*tukeywin(ram_size,0.2).');
% ramp_up        = round(linspace(0,Max_wt,ram_size));
% ramp_down   = round(linspace(Max_wt,0,ram_size));
% up_down        = round([linspace(0,Max_wt,ram_size/2) linspace(Max_wt,0,ram_size/2)-1]);
% Algorithm for 140 to 160MHz ****************************************************
test1              = linspace(1,0.9,(0.8*ram_size));
test2              = linspace(0.9,0.72,(0.2*ram_size));
test_lin          = Max_wt.*[test1 test2];
% Frequency for 140 to 160MHz
test_freq         = 140:((160-140)/(ram_size-1)):160;
% Algorithm for 180 to 210MHz ****************************************************
% test1            = linspace(1,0.68,(0.5*ram_size));
% test2            = linspace(0.68,0.48,(0.5*ram_size));
% test_lin         = Max_wt.*[test1 test2];
%Compensation for 180-210MHz
% for i = 254:296
%     test_lin(i) = 0.98*test_lin(i);     % Correct for amplitude spike from 192 to 194MHz
% end
% Frequency for 180 to 210MHz
% test_freq       = 180:((210-180)/(ram_size-1)):210;
% *********************************************************************************
% Plot weights
figure;plot(test_freq,test_lin)
% Apply tukey window to calculate predistortion amplitude weighting
test_lin          = test_lin.*tukeywin(ram_size,0.2).';
% **NOTE: the DDS is loaded in reverse(i.e. last MATLAB value=1st DDS value)
for i = 1:length(test_lin)
    if test_lin(i) > Max_wt
        test_lin(i) = Max_wt-200;        % ensure all values are less than 65,535 (max weight)
    end
end

test_lin_flip     = fliplr(test_lin);       % reverse order of array
%  Write reversed amplitude weighting values to text file *********************************
format short;
dlmwrite('weighting.txt',test_lin_flip,'newline','pc');     % write weighting to .txt file
figure;
```

```
subplot(1,2,1);plot(test_lin)
title('Calculated Weighting Amplitude')
xlabel('Ram Location');ylabel('Actual Amplitude Value')
subplot(1,2,2);plot(test_lin_flip)
title('Calculated DDS Amplitude')
xlabel('Ram Location');ylabel('DDS Amplitude Value ')
```

APPENDIX B

ADS DC Bias Simulation – SP202 and SP203

FET Curve Tracer

Vdrain

Drain

V_DC
SRC1
Vdc=VDS

I_Probe
IDS

Polyfet Rf Devices

Gate

GATE   DRAIN

V_DC
SRC2
Vdc=VGS

SOURCE

SP202
X2

Source

PARAMETER SWEEP   DC

ParamSweep
Sweep1
SweepVar="VGS"
SimInstanceName[1]="DC1"
SimInstanceName[2]=
SimInstanceName[3]=
SimInstanceName[4]=
SimInstanceName[5]=
SimInstanceName[6]=
Start=2.5
Stop=4.0
Step=0.05

DC
DC1
SweepVar="VDS"
Start=0
Stop=40
Step=0.1

Set drain and gate voltage
sweep limits as needed.

VAR
VAR1
VDS =28 V
VGS =3.5 V

DisplayTemplate
disptemp1
"FET_curve_tracer"

ADS DC Bias Simulation – SD703

FET Curve Tracer

Drain

V_DC
SRC1
Vdc=VDS

I_Probe
IDS

Polyfet Rf Devices

Gate

GATE   DRAIN

V_DC
SRC2
Vdc=VGS

SOURCE

SD703
X2

Source

PARAMETER SWEEP   DC

ParamSweep
Sweep1
SweepVar="VGS"
SimInstanceName[1]="DC1"
SimInstanceName[2]=
SimInstanceName[3]=
SimInstanceName[4]=
SimInstanceName[5]=
SimInstanceName[6]=
Start=2.0
Stop=3.0
Step=0.05

DC
DC1
SweepVar="VDS"
Start=0
Stop=40
Step=0.1

Set drain and gate voltage
sweep limits as needed.

VAR
VAR1
VDS =28 V
VGS =4.0 V

DisplayTemplate
disptemp1
"FET_curve_tracer"

Polyfet SP202 Non-Linear ADS Model
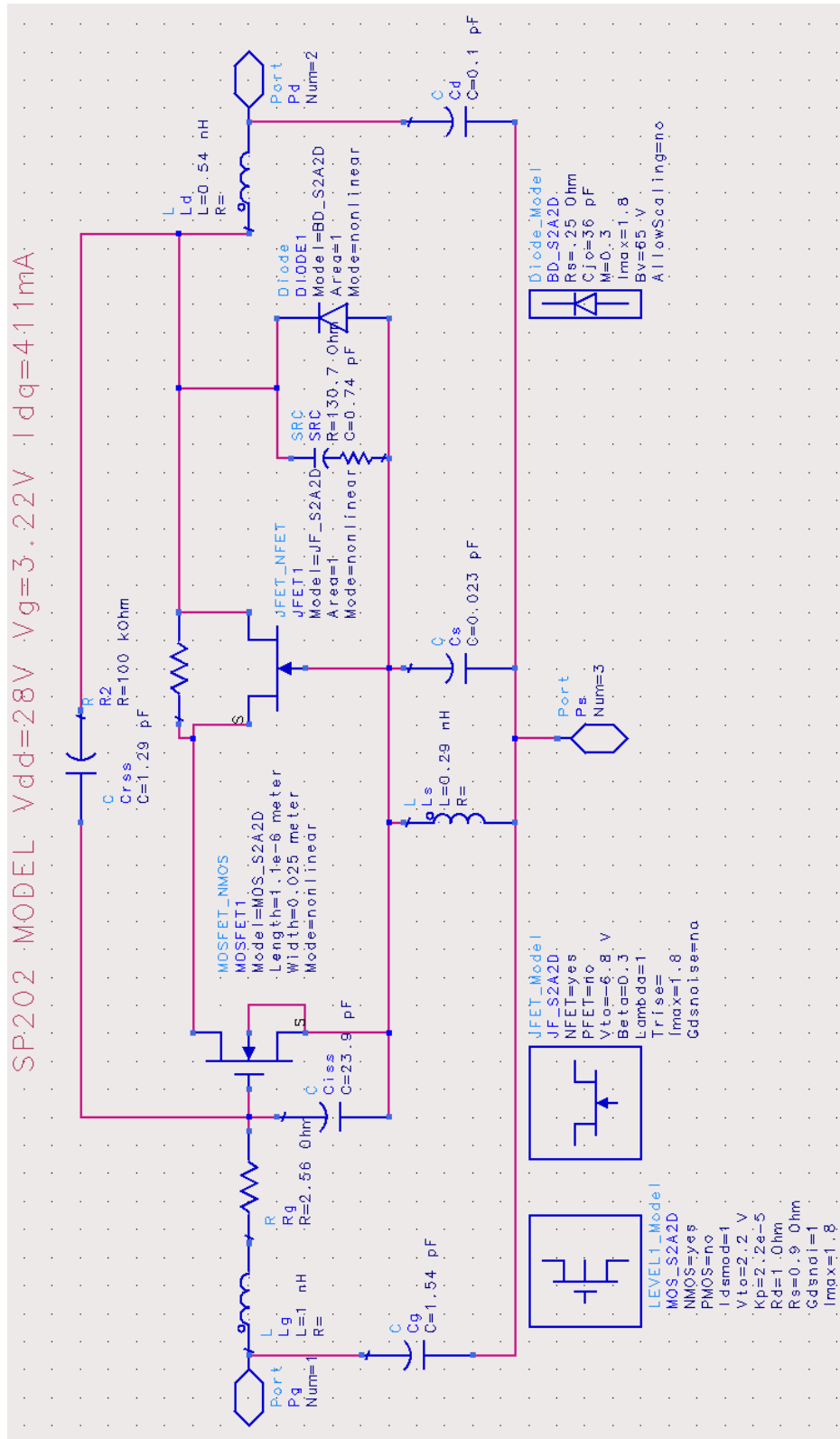
Polyfet SP203 Non-Linear ADS Model

SP203 MODEL Vdd=28V Vg=3.18V Idq=608mA

Port
Pd
Num=2

L
Ld
L=0.79 nH
R=

Diode
DIODE1
Model=BD_S2A3D
Area=1
Periph=
Scale=
Region=
Temp=
Trise=
Mode=nonlinear

C
Cd
C=0.11 pF

Diode_Model
BD_S2A3D
Rs=.25 Ohm
Cjo=55 pF
M=0.3
Bv=65 V
AllowScaling=no

SRC
SRC
R=158.8 Ohm
DC=4.3 pF

JFET_NFET
JFET1
Model=JF_S2A3D
Area=1
Region=
Temp=
Trise=
Mode=nonlinear

C
Cs
C=0.058 pF

R
R2
R=100 kOhm

C
Crss
C=1.87 pF

MOSFET_NMOS
MOSFET1
Model=MOS_S2A3D
Length=1.1e-6 meter
Width=0.0375 meter
Temp=
Trise=
Mode=nonlinear

Port
Ps
Num=3

L
Ls
L=0.14 nH
R=

C
Ciss
C=37.5 pF

JFET_Model
JF_S2A3D
NFET=yes
PFET=no
Vto=-6.8 V
Beta=0.3
Lambda=1
Trise=
Imelt=
Gdsnoise=no

R
Rg
L=0.65 nH R=1.69 Ohm

C
Cg
C=1.95 pF

L
Lg
L=0.65 nH
R=

Port
Pg
Num=1

LEVEL1_Model
MOS_S2A3D
NMOS=yes
PMOS=no
Idsmod=1
Vto=2.2 V
Kp=2.2e-5
Rd=0.85 Ohm
Rs=0.55 Ohm
Gdsnoi=1

Polyfet SD703WA Non-Linear ADS Model (Workaround Model)

SD703(WA.)(Half) MODEL. Vdd=28V. Vg=2.61V. Idq=616mA.
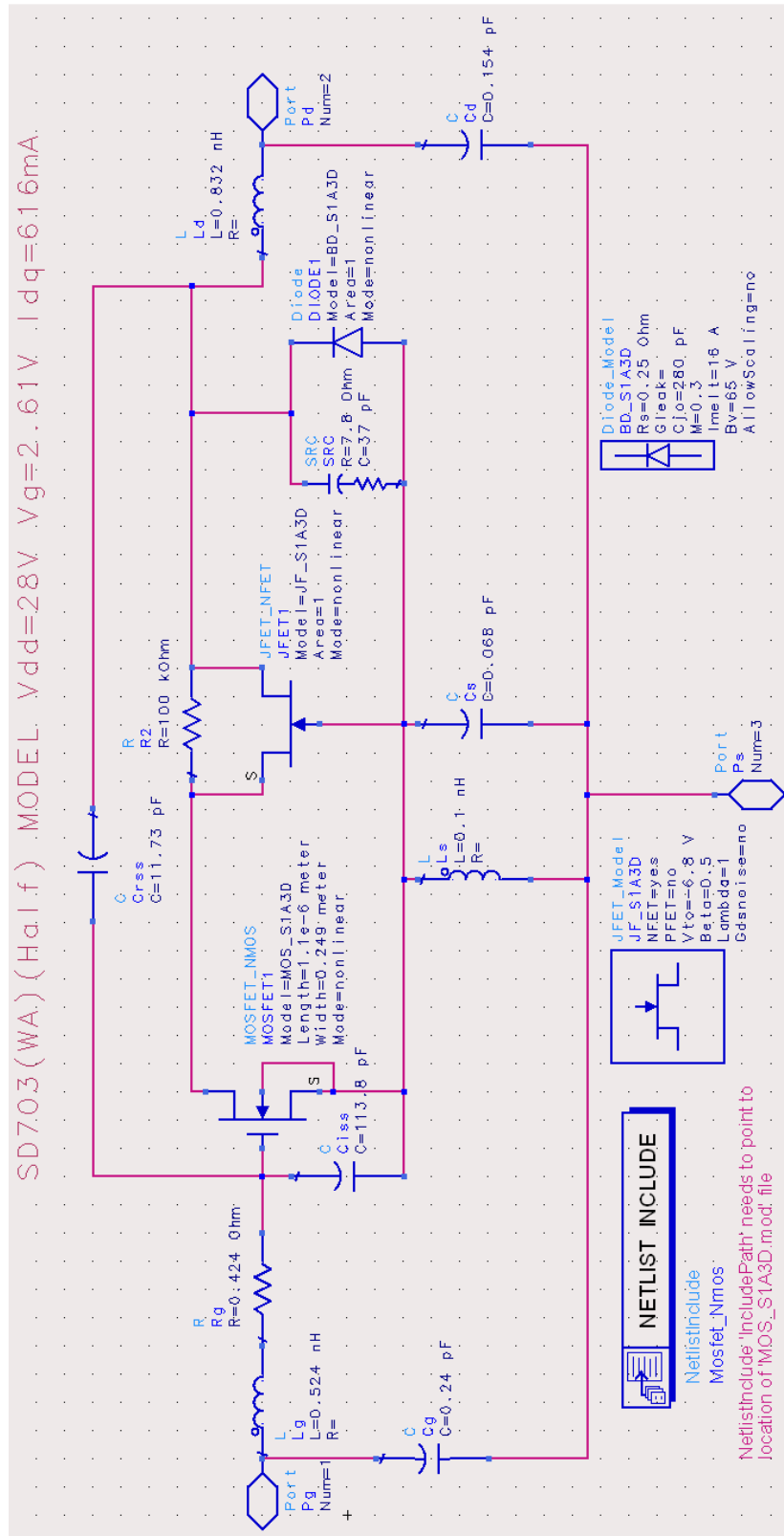
ADS Stability Simulation – SP202
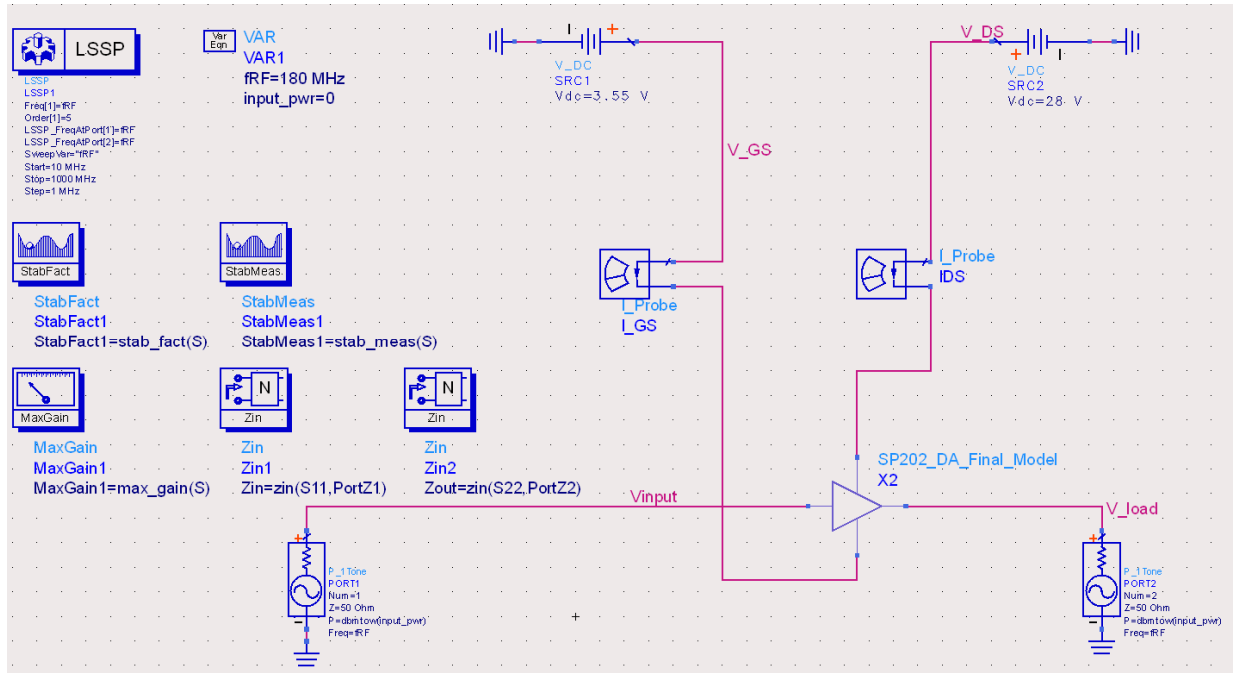
'SP202_DA_FINAL_MODEL' Amplifier Model
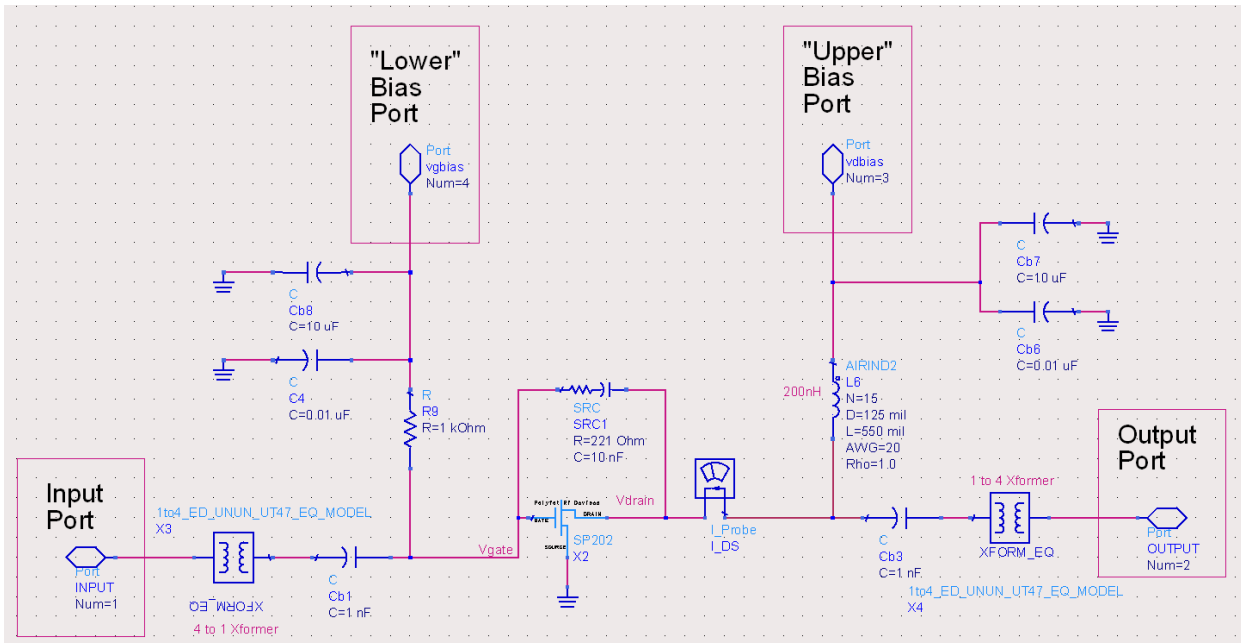
ADS Stability Simulation – SP203
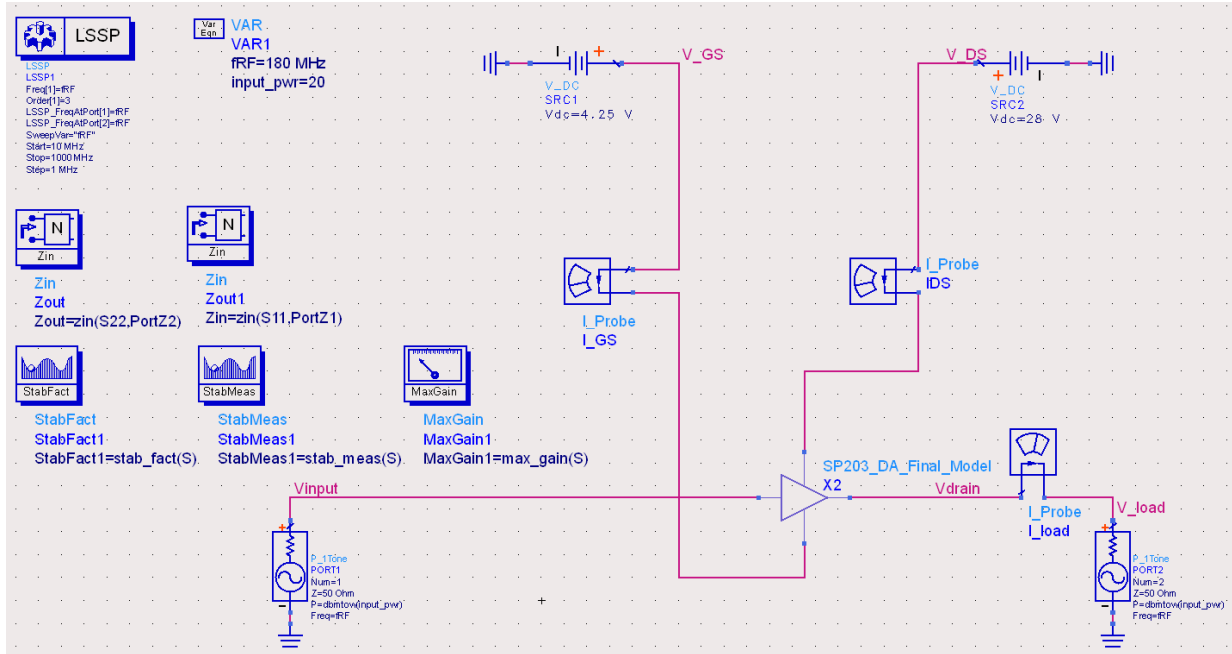
'SP203_DA_FINAL_MODEL' Amplifier Model
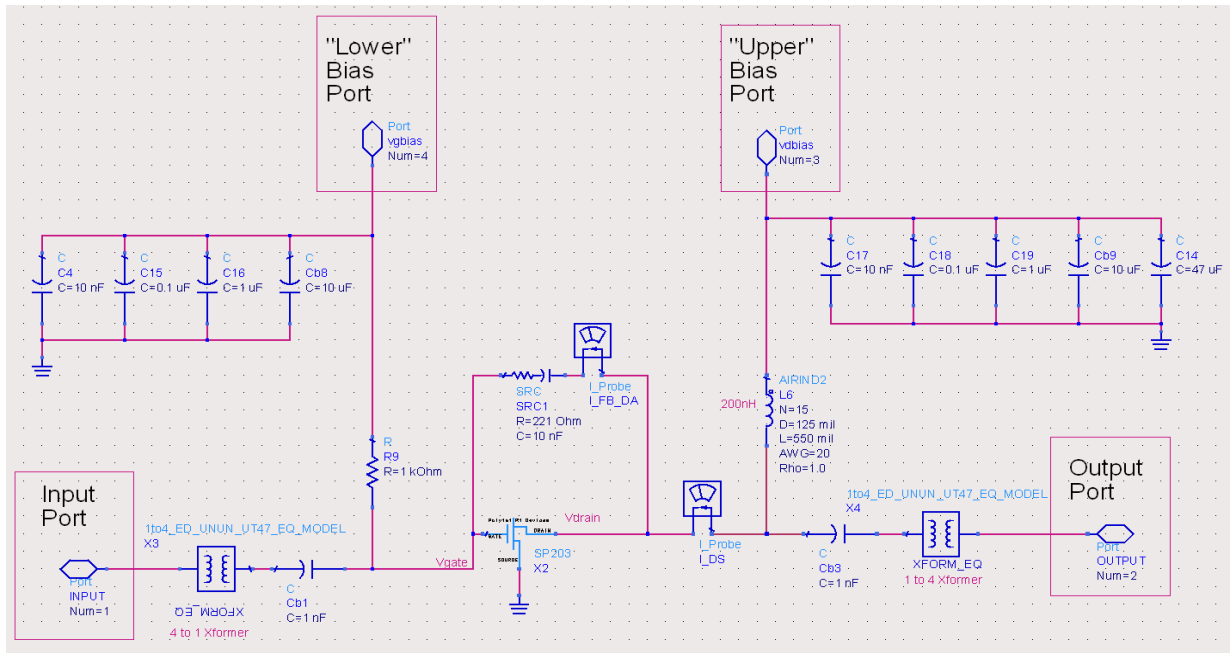
ADS Stability Simulation – SD703
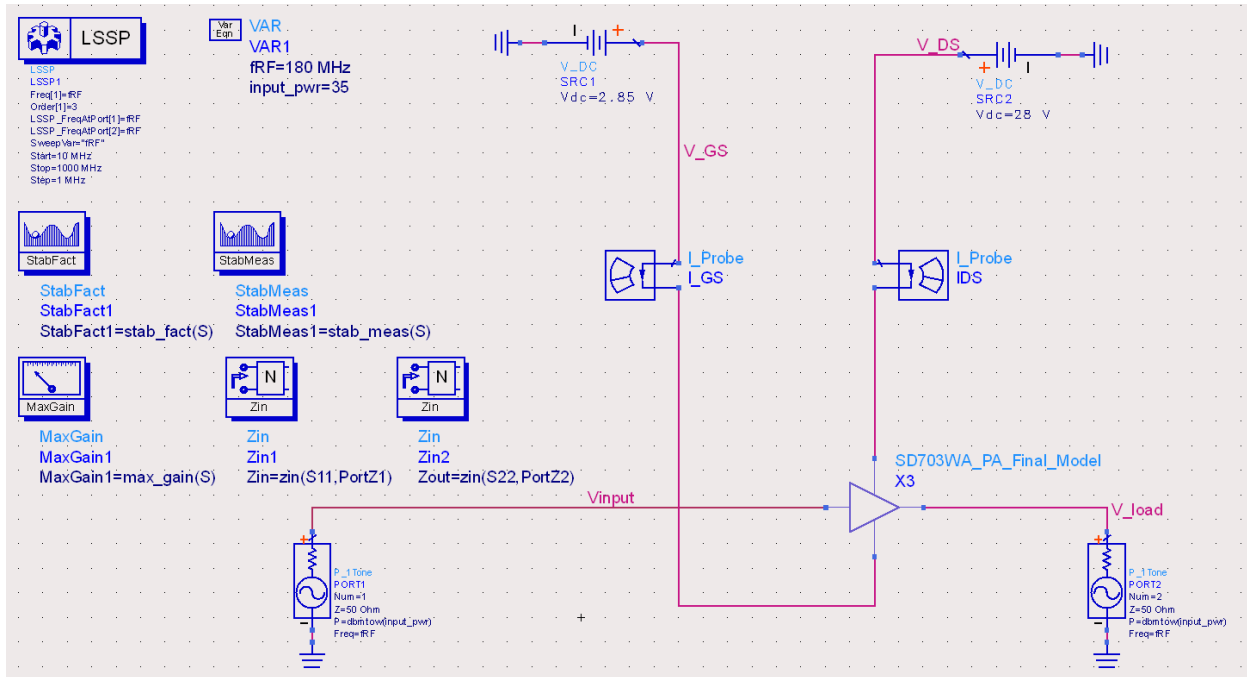
'SD703WA_PA_FINAL_MODEL' Amplifier Model
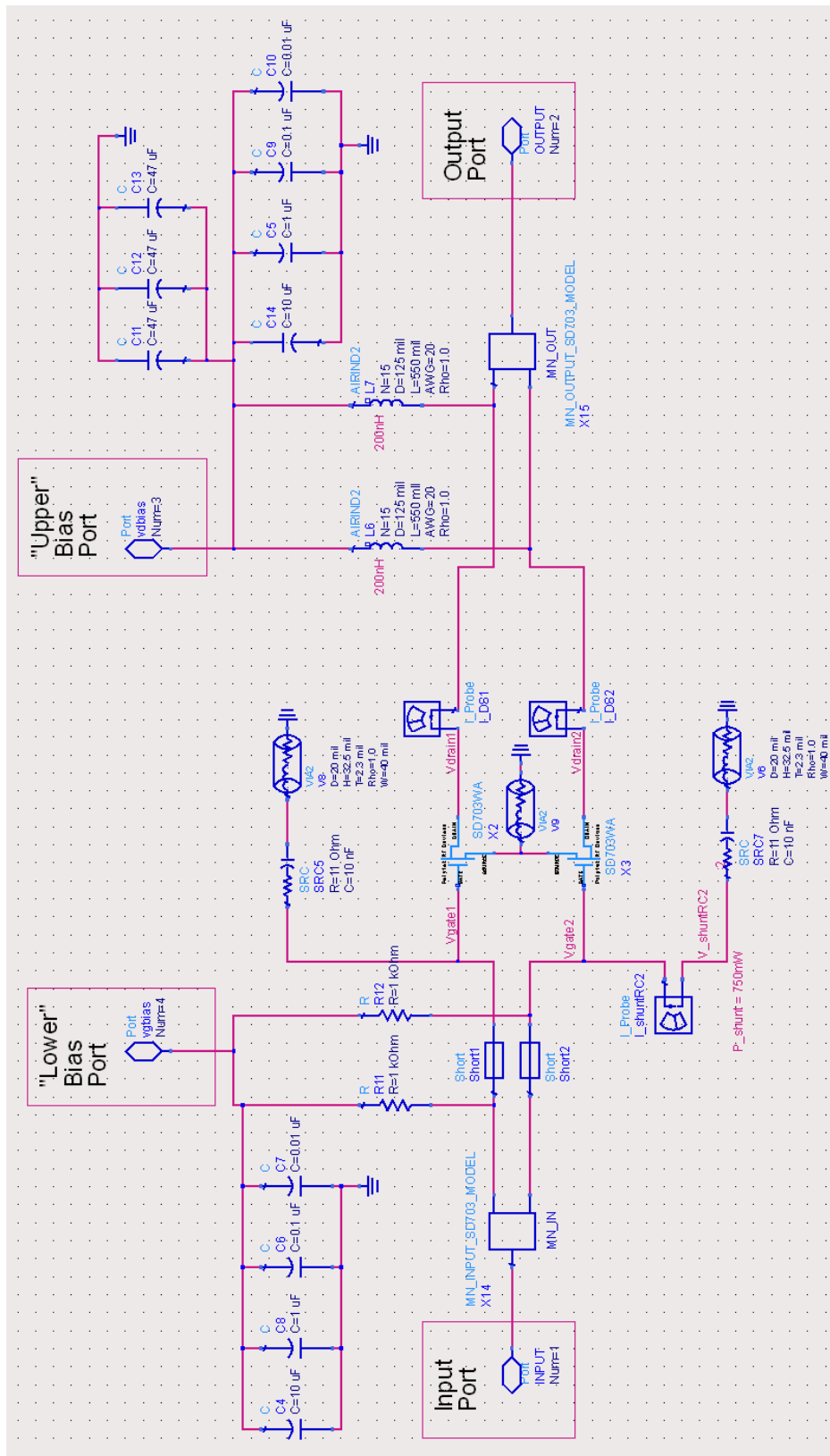
ADS Stability Simulation – Final Design (SP202 + SP203 + SD703)

ADS Load-Pull Simulation – SP202

ADS Load-Pull Simulation – SP203

ADS Load-Pull Simulation – SD703

C
Cb8
C=10 uF

C
Cb7
C=10 uF

C
C4
C=0.01 uF

I_Probe
Is_high

C
Cb6
C=0.01 uF

V_DC
SRC4
Vdc=Vlow V

Vs_high

V_DC
SRC1
Vdc=Vhigh

Specify desired Fundamental Load Tuner coverage:
s11_rho is the radius of the circle of reflection coefficients
    generated. However, the radius of the circle will be
    reduced if it would otherwise go outside the Smith Chart.
s11_center is the center of the circle of generated reflection coefficients.
pts is the total number of reflection coefficients generated.
Z0 is the system reference impedance.

s11_rho is the radius
and s11_center is the
center of the circle.
(But this is just a
static drawing.)

SRC
SRC5
R=11 Ohm
C=10 nF

Via2
V8
D=20 mil
H=32.5 mil
T=2.3 mil
Rho=1.0
W=40 mil

R
Rg
R=1 kOhm

VAR
SweepEquations
s11_rho =.15
s11_center =-0.81 -j*0.0
pts=225
Z0=50

AIRIND2
L6
N=15
D=125 mil
L=550 mil
AWG=20
Rho=1.0

200nH

C
C2
C=10 nF

I_Probe
Iload   vload

Polyfet RF Devices

P_1Tone
PORT1
Num=1
Z=Z_s
P=dbmtow(Pavs)
Freq=RFfreq

C
Cb1
C=10 nF

SD703WA
X1

C
C1
C=10 nF

S1P_Eqn
S1
S[1,1]=LoadTuner
Z[1]=Z0

One Tone Load Pull Simulation:
output power and PAE found at
each fundamental load impedance

Set Load and Source
impedances at
harmonic frequencies

Refer to the data display file
"ReflectionCoefUtility" for
help in setting s11_rho and
s11_center. Also, refer to
the example design file: examples/RF_Board/
LoadPull_prj/HB1Tone_LoadPull_eqns for
details about how this simulation is run.

Set these values:

VAR
STIMULUS
Pavs=36 _dBm
RFfreq=180 MHz
Vhigh=28
Vlow=2.85

PARAMETER SWEEP

ParamSweep
Sweep1

VAR
VAR2
Z_l_2 =10*Z0 + j*0
Z_l_3 =10*Z0 + j*0
Z_l_4 =10* Z0 + j*0
Z_l_5 =10* Z0 + j*0
Z_s_fund =10 + j*0
Z_s_2 =10* Z0 + j*0
Z_s_3 =10* Z0 + j*0
Z_s_4 =10* Z0 + j*0
Z_s_5 =10* Z0 + j*0

VAR
ImpedanceEquations

HARMONIC BALANCE

HarmonicBalance
HB1
Freq[1]=RFfreq
Order[1]=5

ADS Simulation – UT-047-25 1:4 Equal Delay Unun Transformer
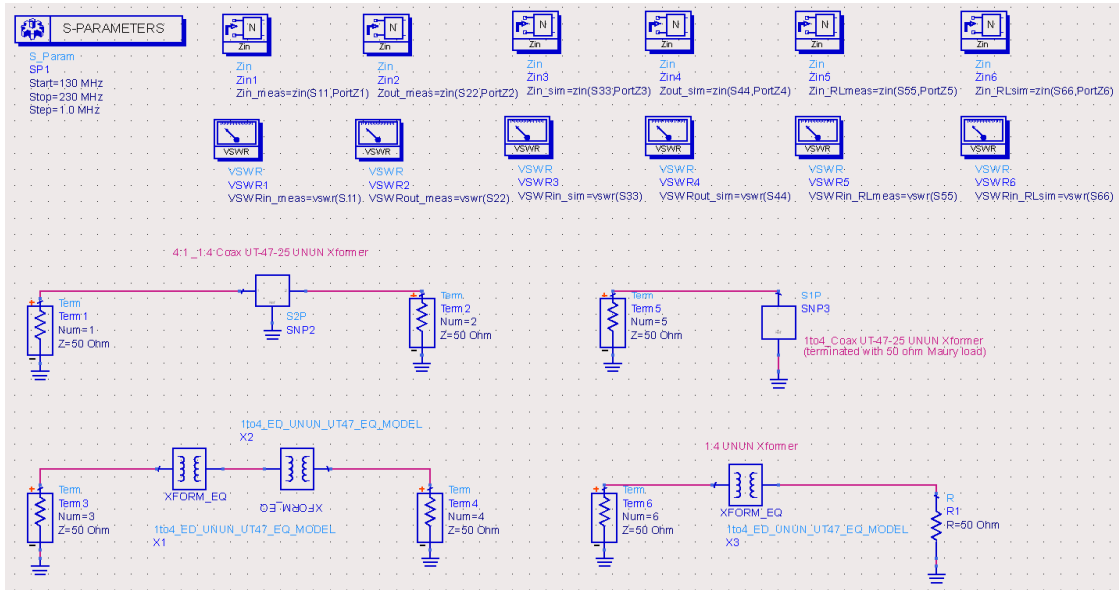
ADS Simulation Results – UT-047-25 1:4 Equal Delay Unun Transformer

Eqn Pd_meas=1-(mag(S(1,1))*mag(S(1,1))+mag(S(2,1))*mag(S(2,1)))   Eqn Pd_meas_dB=10*log(Pd_meas)

Eqn Pd_sim=1-(mag(S(3,3))*mag(S(3,3))+mag(S(4,3))*mag(S(4,3)))   Eqn Pd_sim_dB=10*log(Pd_sim)
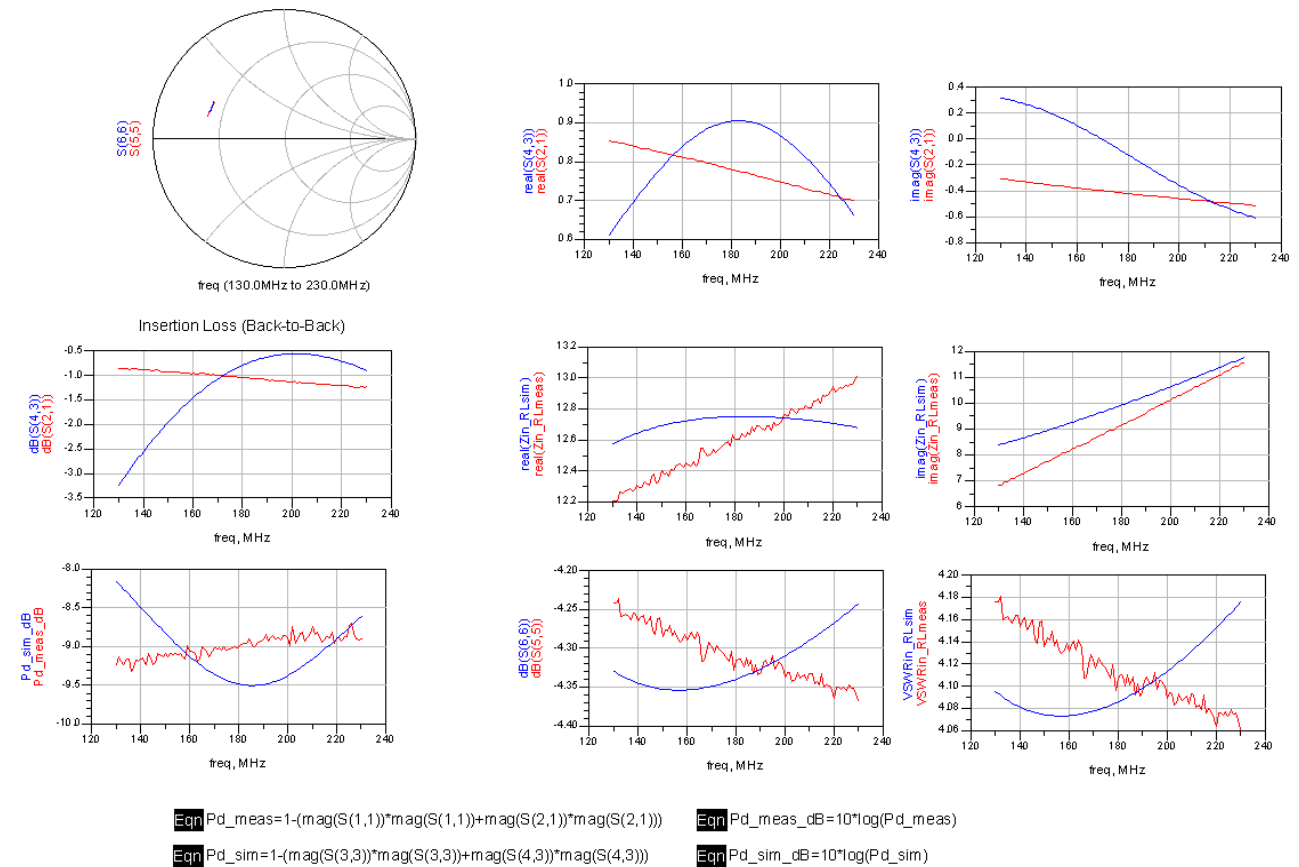
ADS Simulation – UT-047-25 4:1 Balbal Transformer
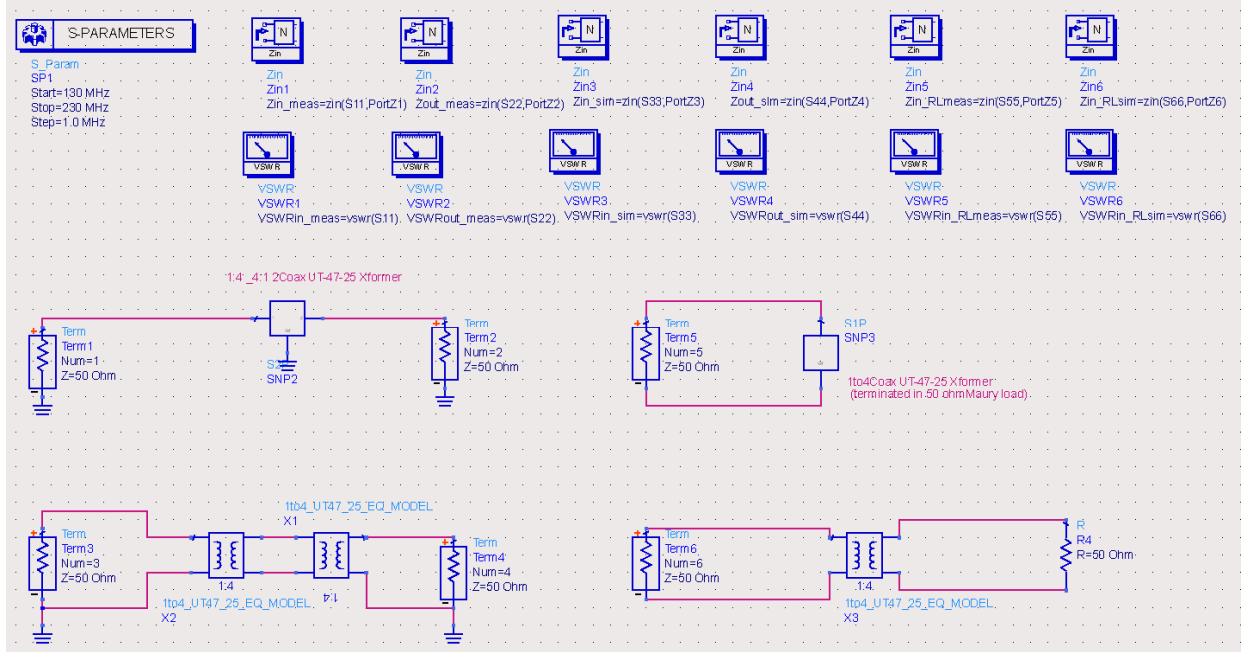
ADS Simulation Results – UT-047-25 4:1 Balbal Transformer
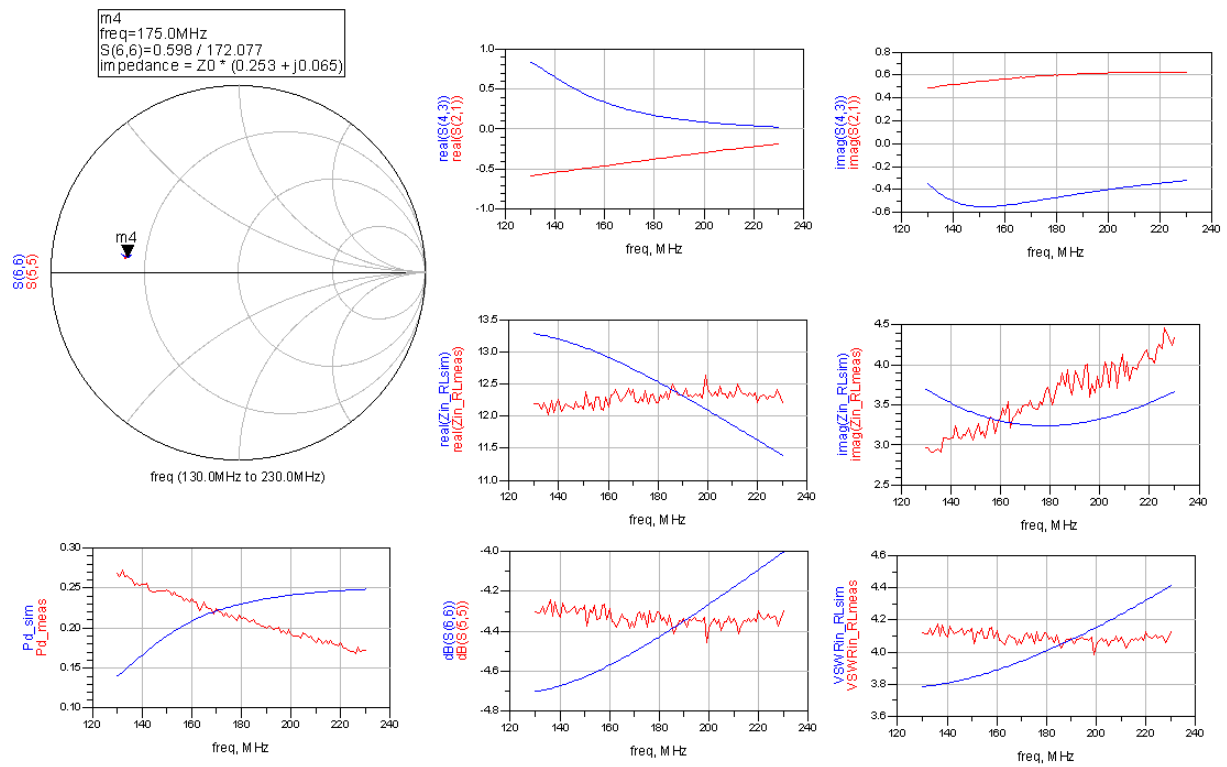
Egn Pd_meas=1-(mag(S(1,1))*mag(S(1,1))+mag(S(2,1))*mag(S(2,1)))

Egn Pd_sim=1-(mag(S(3,3))*mag(S(3,3))+mag(S(4,3))*mag(S(4,3)))

ADS Simulation – UT-085C-15 1:9 Balbal Transformer
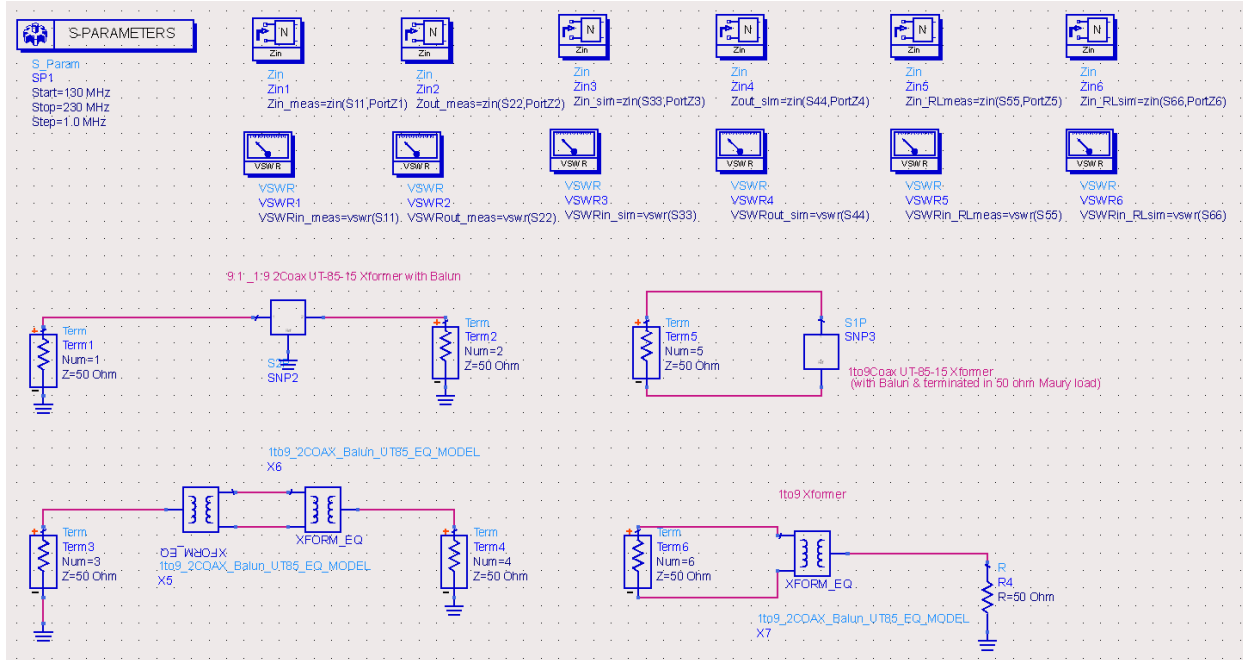
ADS Simulation Results – UT-085C-15 1:9 Balbal Transformer
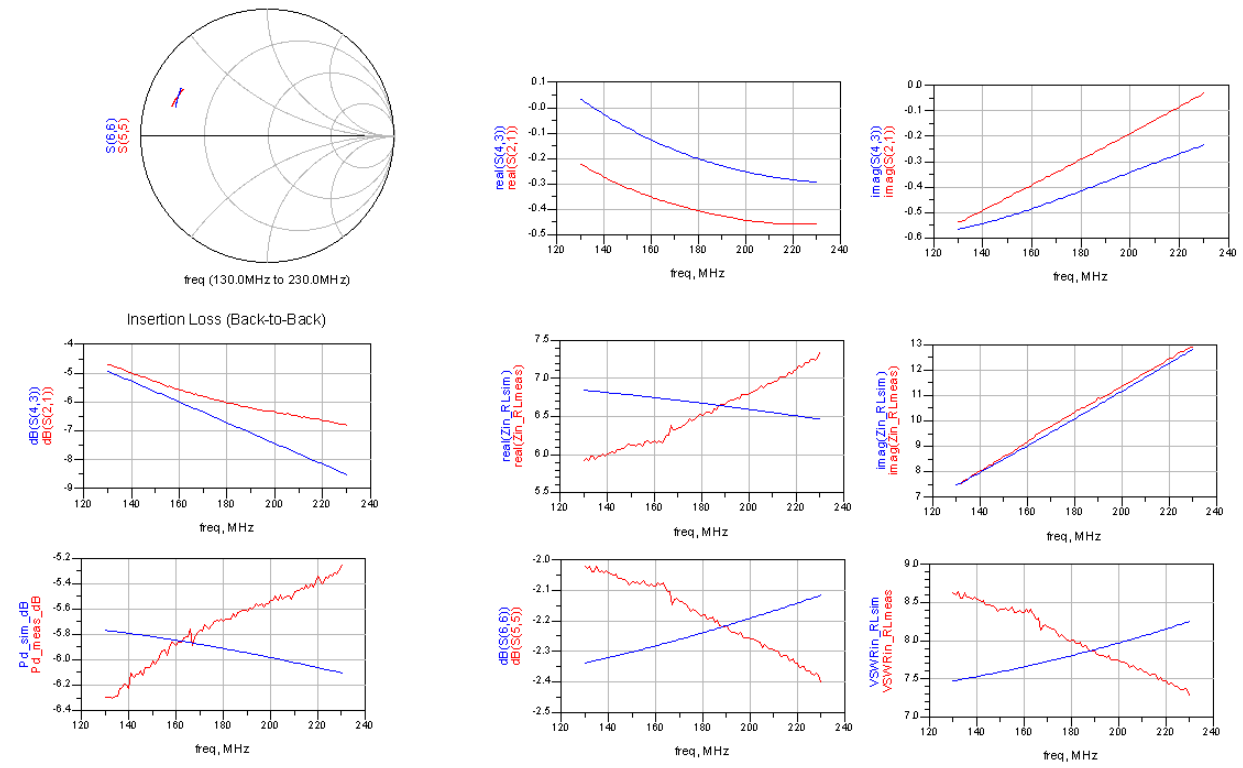
Eqn Pd_meas=1-(mag(S(1,1))*mag(S(1,1))+mag(S(2,1))*mag(S(2,1)))   Eqn Pd_meas_dB=10*log(Pd_meas)

Eqn Pd_sim=1-(mag(S(3,3))*mag(S(3,3))+mag(S(4,3))*mag(S(4,3)))   Eqn Pd_sim_dB=10*log(Pd_sim)

'1to4_EQ_UNUN_UT47_EQ_MODEL' Transformer Model
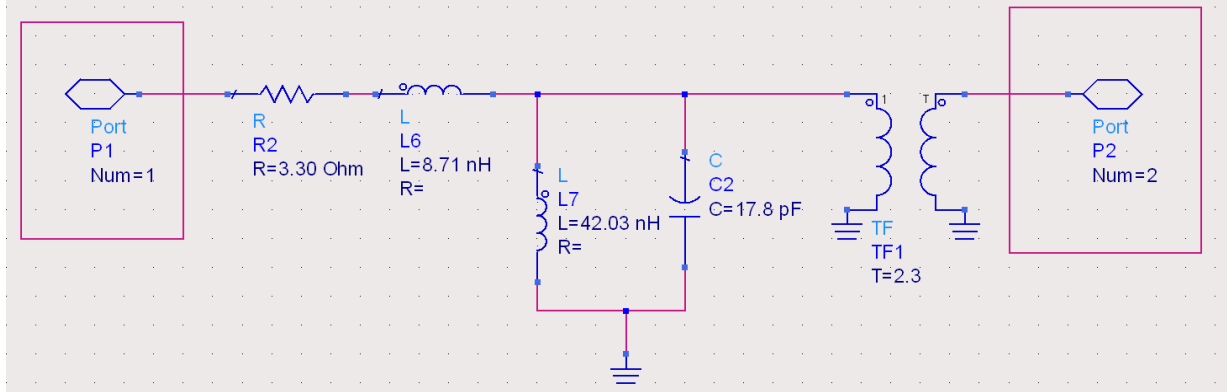
'1to4_UT47_25_EQ_MODEL' Transformer Model
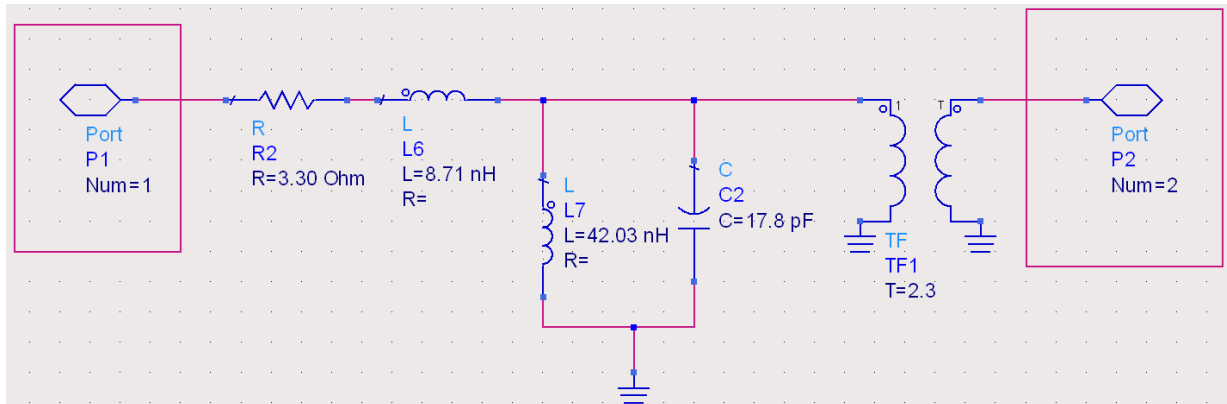
'1to9_2COAX_Balun_UT85_EQ_MODEL' Transformer Model
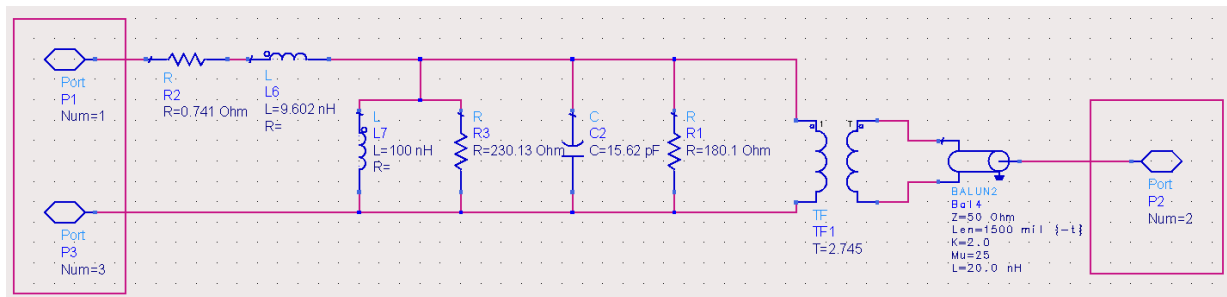
APPENDIX C

ADS Simulation - RF Choke Inductor Equivalent Circuit Model

S-PARAMETERS

S_Param
SP1
Start=0.1 kHz
Stop=500 MHz
Step=1 MHz

Zin
Zin2
Zair=zin(S11,PortZ1)

Zin
Zin1
Zcore=zin(S22,PortZ2)

Zin
Zin3
ZairMeas=zin(S33,PortZ3)

Zin
Zin4
ZcoreMeas=zin(S55,PortZ5)

Zin
Zin5
ZairEQ=zin(S77,PortZ7)

Zin
Zin6
ZcoreEQ=zin(S88,PortZ8)

MeasEqn
Meas1
Ind_air_nH= (imag(Zair)/(2*pi*freq))*1e9
Ind_core_nH= (imag(Zcore)/(2*pi*freq))*1e9
Q_air = imag(Zair)/real(Zair)
Q_core = imag(Zcore)/real(Zcore)

MeasEqn
Meas2
Ind_airMeas_nH= (imag(ZairMeas)/(2*pi*freq))*1e9
Ind_coreMeas_nH= (imag(ZcoreMeas)/(2*pi*freq))*1e9
Ind_coreMeasCap= -(imag(ZcoreMeas)/(2*pi*freq))
Q_airMeas = imag(ZairMeas)/real(ZairMeas)
Q_coreMeas = imag(ZcoreMeas)/real(ZcoreMeas)

Term
Term1
Num=1
Z=50 Ohm

AIRIND2
L6
N=15
D=141 mil
L=580 mil
AWG=20
Rho=1.0

Term
Term2
Num=2
Z=50 Ohm

CIND
L8
N=19
AL=56 nH

15 turn Air Core
141 mil ID

19 turn FT50-61 Core
non-Enamel Wire

Term
Term3
Num=3
Z=50 Ohm

S2P
SNP1

Term
Term4
Num=4
Z=50 Ohm

Term
Term5
Num=5
Z=50 Ohm

S2P
SNP2

Term
Term6
Num=6
Z=50 Ohm

VAR
Meas1
Cair=3.5 {-t}
Rvar1=0.25 {-t}
Rvar2=50000 {-t}
R1=2.695 {-t}
R2=0.98 {-t}
Lair=114 {-t}

C
C1
C=Cair pF

R
R1
R=R1 Ohm

R
R4
R=R2 Ohm

R
R2
R=Rvar1 Ohm

L
L9
L=Lair nH
R=

Term
Term7
Num=7
Z=5 Ohm

VAR
Meas1
Ccore=45 {t}
Rvar1c=0.007 {t}
Rvar2c=50000 {-t}
R1c=9.8 {t}
R2c=0.03 {t}
Lcore=390 {t}

C
C2
C=Ccore pF

R
R7
R=R1c Ohm

R
R5
R=R2c Ohm

R
R8
R=Rvar1c Ohm

L
L10
L=Lcore nH
R=

Term
Term8
Num=8
Z=5 Ohm

ADS Simulation Results – Air Core RF Choke Inductor Equivalent Circuit Model
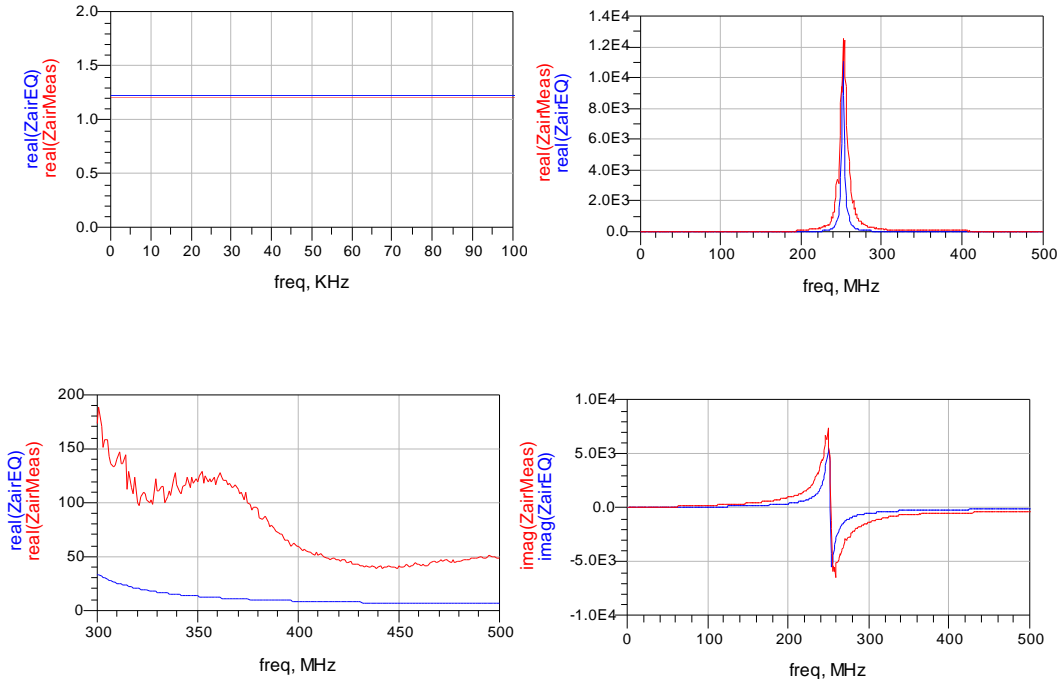
ADS Simulation Results – FT50-61 Core RF Choke Inductor Equivalent Circuit Model
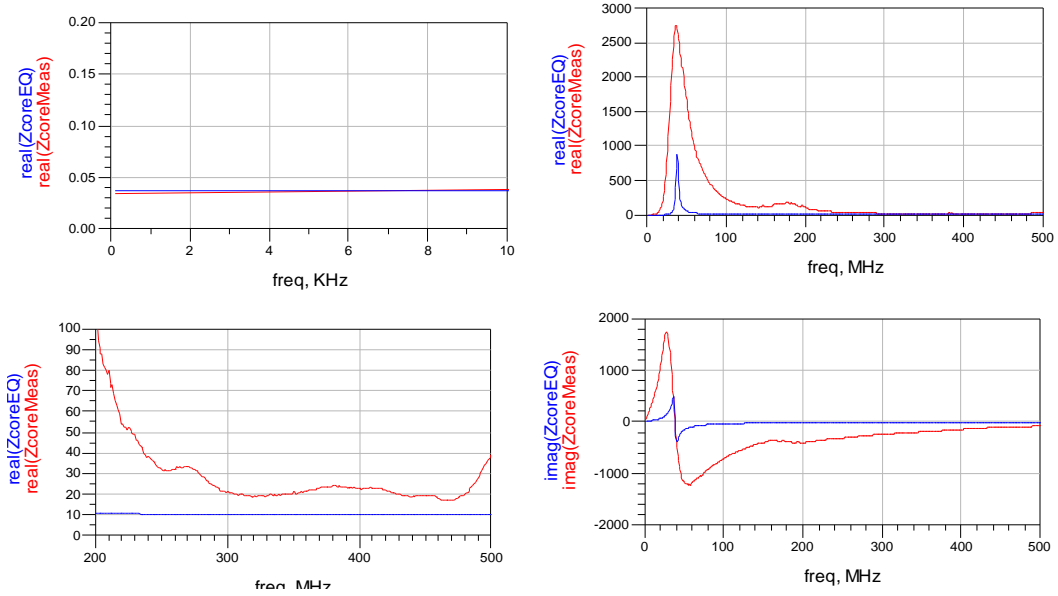
APPENDIX D

Eagle PCB Original Prototype Layout Schematic – Page 1

Eagle PCB Original Prototype Layout Schematic – Page 2

Eagle PCB Original Prototype Layout Schematic – Page 3

Eagle PCB Original Prototype Layout Schematic – Page 4

Eagle PCB Modified Prototype Schematic – Page 1

Eagle PCB Modified Prototype Schematic – Page 2

Eagle PCB Modified Prototype Schematic – Page 3

UAV 2009 - POWER AMPLIFIER WITH MOSFET DRIVER

POWER AMPLIFIER STAGE

PA_IN

2 dB (OPT)
U6
RFP-2502S0-4AA
A
B   B_BALIN_IN
GND
GND
GND

UT-044-12
L = 500 mil   1:1 TLT BALUN
GND

C18
10pF
ATC 800B

LOW_Z_1   T5
LOW_Z_2
1:9 TLT_BALBAL_XFMR
HIGH_Z_1
HIGH_Z_2

C14
10nF
0805
GND

R21
22
1206

C16
220pF
ATC 800B

R23
22
1206

C20
220pF
ATC 800B

C22
10nF
0805
GND

VGS_PA

R22
22
1206

G1

G2
Q4
POLYFET_SD703
GND

+28V_FEED

L5
AIR_CORE_HAND_MADE
N=15
I.D. = 141mil
L=580mil
AWG=20
L5_A   ~225nH   L5_B

C15
220pF
ATC 800B

C17
3.9pF
ATC 800B

C21
220pF
ATC 800B

+28V_FEED

L6
AIR_CORE_HAND_MADE
N=15
I.D. = 141mil
L=580mil
AWG=20
L6_B   ~225nH   L6_A

BAL_1   T6
BAL_2
1:9_TLT_BALUN

50 OHM COAX
1:1 TLT BALUN

C19
20pF
ATTACH ACROSS
50 OHM BALUN

SMA_CHASSIS_0.125_IN
J3
GND

CReSIS

Eagle PCB Modified Prototype Schematic – Page 4

DISCRETE COMPONENT MOSFET DRIVER CIRCUIT

UAV 2009 - POWER AMPLIFIER WITH MOSFET DRIVER
HIGH SPEED / HIGH VOLTAGE MOSFET DRIVER CIRCUITS

TURN ON TIME = 391 nS
TURN OFF TIME = 79.1 nS

CReSIS - University of Kansas
2335 Irving Hill Rd.
Lawrence, KS 66045
(785) 864-4390
http://www.cresis.ku.edu/

Project: UAV 2009 - Power Amplifier with MOSFET Driver
Engineer: Kevin Player
Date: 3/13/2010 10:06:23 AM   Sheet: 4/4
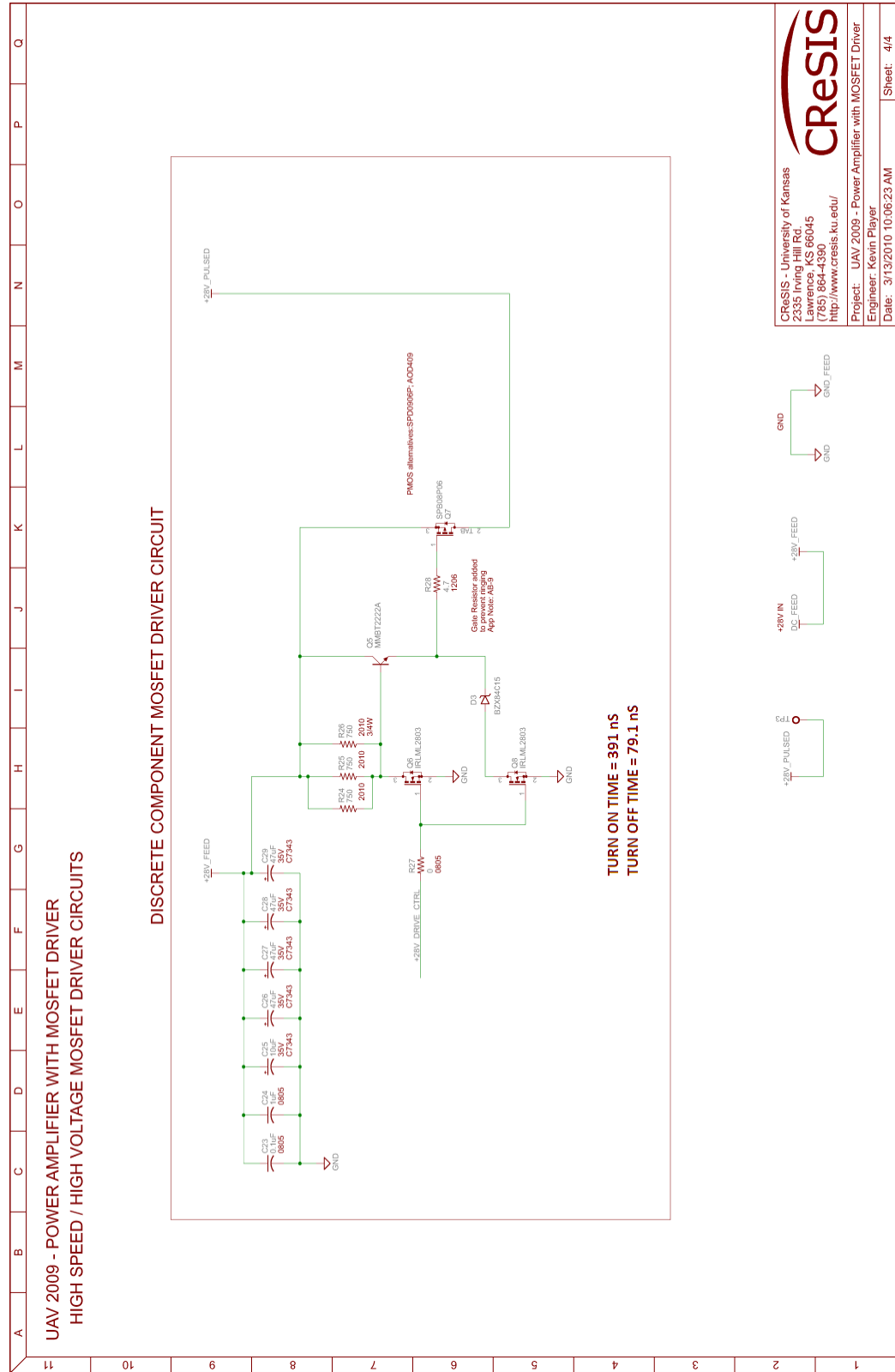
APPENDIX E

UT-047-25 Micro-Coax Datasheet

MICRO-COAX®
Leading the way in transmission line solutions.

UT-047-25/CA25047

206 Jones Blvd. Pottstown, PA 19464 USA
Phone: 610-495-0110 : 800-223-2629
www.micro-coax.com

Semi-Rigid Coaxial Cable

―――――――――――――― MECHANICAL CHARACTERISTICS ――――――――――――――

Outer Conductor Diameter, inch (mm)	0.047+/-0.003 (1.194+/-0.0762)
Dielectric Diameter, inch (mm)	0.028 (0.711)
Center Conductor Diameter, inch (mm)	0.0159+/-0.0005 (0.404+/-0.0127)
Maximum Length, feet (meters)	25 (7.62)
Minimum Inside Bend Radius, inch (mm)	0.125 (3.175)
Weight, pounds/100 ft. (kg/100 meters)	0.54 (0.81)

―――――――――――――― ELECTRICAL CHARACTERISTICS ――――――――――――――

Impedance, ohms	25+/-3.0
Frequency Range GHz	DC-120
Velocity of Propagation %	70
Capacitance, pF/ft. (pF/meter)	58.9 (193.2)

Typical Insertion Loss, dB/ft. (dB/meter) and Average Power Handling, Watts CW at 20 degrees Celsius and Sea level

Frequency	Insertion Loss	Power
0.5 GHz	0.43 (1.40)	47.3
1.0 GHz	0.61 (1.99)	33.4
5.0 GHz	1.38 (4.51)	14.7
10.0 GHz	1.97 (6.47)	10.3
20.0 GHz	2.84 (9.30)	7.2

Corona Extinction Voltage, VRMS @ 60 Hz	850
Voltage Withstand, VRMS @ 60 Hz	1500

―――――――――――――― ENVIRONMENTAL CHARACTERISTICS ――――――――――――――

Outer Conductor Integrity Temperature, Deg Celsius	175
Maximum Operating Temperature, Deg Celsius	150

―――――――――――――― MATERIALS ――――――――――――――

Outer Conductor	Copper
Dielectric	PTFE
Center Conductor	SPCW

UT-085C-15 Micro-Coax Datasheet

MICRO-COAX®
Leading the way in transmission line solutions.

UT-085C-15
( )

206 Jones Blvd. Pottstown, PA 19464 USA
Phone: 610-495-0110 : 800-223-2629
www.micro-coax.com

Semi-Rigid Coaxial Cable

MECHANICAL CHARACTERISTICS

Outer Conductor Diameter, inch (mm)	0.0865+/-0.001 (2.197+/-0.0254)
Dielectric Diameter, inch (mm)	0.066 (1.676)
Center Conductor Diameter, inch (mm)	0.046+/-0.001 (1.168+/-0.0254)
Maximum Length, feet (meters)	20 (6.1)
Minimum Inside Bend Radius, inch (mm)	0.125 (3.175)
Weight, pounds/100 ft. (kg/100 meters)	1.76 (2.62)

ELECTRICAL CHARACTERISTICS

Impedance, ohms	15+/-1.0
Frequency Range GHz	DC-47
Velocity of Propagation %	70
Capacitance, pF/ft. (pF/meter)	96.7 (317.4)

Typical Insertion Loss, dB/ft. (dB/meter) and Average Power Handling, Watts CW at 20 degrees Celsius and Sea level

Frequency	Insertion Loss	Power
0.5 GHz	0.25 (0.83)	103.5
1.0 GHz	0.36 (1.17)	72.8
5.0 GHz	0.82 (2.70)	31.8
10.0 GHz	1.19 (3.90)	22.1
20.0 GHz	1.73 (5.68)	15.3

Corona Extinction Voltage, VRMS @ 60 Hz	850
Voltage Withstand, VRMS @ 60 Hz	2000

ENVIRONMENTAL CHARACTERISTICS

Outer Conductor Integrity Temperature, Deg Celsius	150
Maximum Operating Temperature, Deg Celsius	125

MATERIALS

Outer Conductor	Copper
Dielectric	PTFE
Center Conductor	SPC

APPENDIX F

UAV Pin vs Pout at 140 MHz

Pin vs Pout [140MHz]

$y = 0.556x + 46.918$

Pout [dBm]

Pin [dBm]

UAV Pin vs Pout at 145 MHz

Pin vs Pout [145MHz]

$y = 0.597x + 47.091$

Pout [dBm]

Pin [dBm]

UAV Pin vs Pout at 150 MHz

Pin vs Pout [150MHz]

$y = 0.572x + 47.121$

Pout [dBm] vs Pin [dBm]

UAV Pin vs Pout at 155 MHz

Pin vs Pout [155MHz]

$y = 0.562x + 47.536$

Pout [dBm] vs Pin [dBm]

UAV Pin vs Pout at 160 MHz

Pin vs Pout [160MHz]

$y = 0.518x + 48.144$

Pout [dBm] vs Pin [dBm]

UAV Pin vs Pout at 180 MHz

Pin vs Pout [180MHz]

$y = 0.523x + 48.984$

Pout [dBm] vs Pin [dBm]

UAV Pin vs Pout at 185 MHz

Pin vs Pout [185MHz]

$y = 0.539x + 49.822$

Pout [dBm]

Pin [dBm]

UAV Pin vs Pout at 190 MHz

Pin vs Pout [190MHz]

$y = 0.454x + 50.127$

Pout [dBm]

Pin [dBm]

UAV Pin vs Pout at 195 MHz

Pin vs Pout [195 MHz]

$y = 0.419x + 50.292$

Pout [dBm] (y-axis, 49 to 52.5)
Pin [dBm] (x-axis, -3 to 5)

UAV Pin vs Pout at 200 MHz

Pin vs Pout [200MHz]

$y = 0.368x + 50.174$

Pout [dBm] (y-axis, 49 to 52)
Pin [dBm] (x-axis, -3 to 5)

UAV Pin vs Pout at 205 MHz

Pin vs Pout [205MHz]

$y = 0.357x + 49.796$

Pout[dBm] vs Pin [dBm]

UAV Pin vs Pout at 210 MHz

Pin vs Pout [210MHz]

$y = 0.415x + 49.225$

Pout [dBm] vs Pin [dBm]

APPENDIX G

MATLAB Code: Waveform Simulations using Measured UAV PA DAQ Data

```matlab
% Written by: Anthony Hoch
% Modified by: Kevin Player - Nov 2009
% script to simulate variations in transmitted waveform
% pulse compression of the waveform A(t) * exp( 2*pi*j*f*t + Phi(t) )
%
clear all; close all; clc;
% fTUK is the output file of the PA with Tukey weighting (i.e. no PD)
% fPD is the output file of the PA with Tukey & PD weighting
fTUK      = 'C:\Documents and
Settings\playerk\Desktop\Backup\Cresis\Thesis\Measurement_Data\UAV_PA\01_02_10\140_160
_TUK_01_02_10.dat';
fPD       = 'C:\Documents and
Settings\playerk\Desktop\Backup\Cresis\Thesis\Measurement_Data\UAV_PA\01_02_10\140_160
_PD_01_02_10.dat';
recs_2_load = 1;            % number of records to analyze
coh_avgs    = 1;            % number of coherent averages
wf_gen_freq = 1e9/8;        % sampling frequency: 111MHz < 2*Fmax=210MHz
Fs          = wf_gen_freq;  % shorter naming convention
start_cut   = 970;          % samples to remove from beginning
cut_length  = 7530;         % samples to remove from end
record      = 1;            % record to analyze
% Read DAQ data from file
[DAQ_TUK_hdr, DAQ_TUK_data] =
Read1UDAQ(fTUK,rec_start,recs_2_load,coh_avgs,wf_gen_freq,start_cut,cut_length);
[DAQ_PD_hdr, DAQ_PD_data] =
Read1UDAQ(fPD,rec_start,recs_2_load,coh_avgs,wf_gen_freq,start_cut,cut_length);
% Measurement Setup: DDS(@ weight = 65,535)-->PA-->~50dB Atten-->DAQ
f0          = 140;      % Minimum operating frequency [MHz]
f1          = 160;      % Maximum operating frequency [MHz]
Min_wt      = 10e3;     % Minimum weighting in AWG
Max_wt      = 65535;    % Maximum weighting in AWG
wt          = 60e3;     % Current operating weight in AWG
samples     = length(DAQ_TUK_data);     % number of samples
nt          = samples;                  % shorter naming convention
% Create TUK_data and PD_data array (non-equalized)
TUK_data    = DAQ_TUK_data(:,record);   % TUK with DC offset
PD_data     = DAQ_PD_data(:,record);    % PD with DC offset
% Remove DC offset from TUK_data and PD_data array
PA_TUK      = TUK_data - mean(TUK_data); % TUK with DC offset removed
for i=1:length(PA_TUK)
    if PA_TUK(i) == max(PA_TUK)      % Find spike in data
        PA_TUK(i) = mean(PA_TUK);    % Set spike equal to mean
    end
end
PA_PD       = PD_data - mean(PD_data);   % Vin with DC offset removed
PA_TUK      = PA_TUK/max(PA_TUK);        % Vout [Normalized]
PA_PD       = PA_PD/max(PA_PD);          % Vin [Normalized]
% DEFINE STANDARD VARIABLES, (used in all processes)    ==================
dt          = 1/Fs;         % Sampling Period
tau         = 10e-6;        % Pulse Length (3uS needs 0.25 tukey)
time        = [(0:1:nt-1)*dt].';
% DEFINE AMPLITUDE RESPONSE, A(t) (apply to 'rx' and 'ref') =============
type_A                          = 'tukey';
switch type_A
    case 'square'
        A = ones(nt,1);
    case 'tukey'
```

```matlab
        A = tukeywin(nt,0.20);
end
% DEFINE PHASE RESPONSE, P(t)  (apply to 'rx' and 'ref')    ==============
type_P                          = 'linear frequency';
switch type_P
    case 'linear frequency'    % CReSIS standard
        f0      = 140e6;
        f1      = 160e6;
        alpha   = pi*(f1-f0)/tau;
        f       = f0;
        P       = alpha*(time.^2);
    case 'linear phase'
        f       = 150e6;
        P       = [1:1:nt].'.*1e-6*[2*pi/nt];
    case 'flat'
        f       = 150e6;
        P       = ones(nt,1);
end
% PRODUCE IDEAL CHIRP WAVEFORM  ==========================================
waveform        = A.*exp(2*pi*j*f.*time + j.*P);    % Transmit Waveform with tapering
= A
% DEFINE FREQUENCY FILTER, F(t) (apply to one part of match)    ==========
% F               = ones(nt,1);          % Rectangular window
% F               = hamming(nt);         % Hamming window
% F               = hanning(nt);         % Hanning window
% F               = blackman(nt);        % Blackman window
% F               = chebwin(nt);         % Dolph-Chebyshev window
F                = blackman(nt).^2;      % Double Blackman window
% PROCESS RANGE COMRESSION  ==============================================
% IDEAL_TX   = IDEAL TRANSMIT CHIRP
% DDS_TX     = DDS TRANSMIT CHIRP (MEASURED AT OUTPUT OF DDS)
% IDEAL_RX   = IDEAL RESPONSE CHIRP
% PA_TX      = PA TRANSMIT CHIRP (MEASURED AT OUTPUT OF PA + 50dB ATTENUATION)
% TARGET1    = SIMULATED DELAYED WEAK RETURN OF LAYER (USING PA_Tx)
ideal_Tx     = [zeros(nt/2,1); waveform; zeros(nt/2,1)];   % Ideal Tx chirp
ideal_Tx_dB  = 20.*log10(abs(ideal_Tx));                   % Convert to dB
TUK_Tx       = [zeros(nt/2,1); A.*PA_TUK; zeros(nt/2,1)];  % Tx chirp with TUK applied
in DDS
TUK_Tx       = TUK_Tx/max(TUK_Tx);                         % Normalize TUK amplitude
TUK_Tx_dB    = 20.*log10(abs(TUK_Tx)+1e-9);                % Convert to dB
PD_Tx        = [zeros(nt/2,1); A.*PA_PD; zeros(nt/2,1)];   % PA Tx chirp with RECT
applied in DDS
PD_Tx        = PD_Tx/max(PD_Tx);                           % Normalize PA amplitude
PD_Tx_dB     = 20.*log10(abs(PD_Tx)+1e-9);                 % Convert to dB
% Simulation of bedrock return
atten        = 3.162e-3;    % 50dB Attenuation of Ideal Response
ideal_Rx     = [zeros(nt/2,1); atten.*F.*waveform; zeros(nt/2,1)];  % Filtered Ideal
Rx chirp
ideal_Rx_dB  = 20.*log10(abs(ideal_Rx)+1e-9);                       % Convert to dB
TUK_Rx       = [zeros(nt/2,1); F.*atten.*PA_TUK; zeros(nt/2,1)];    % Filtered TUK Rx
chirp (after 50dB atten)
TUK_Rx_dB    = 20.*log10(abs(TUK_Rx)+1e-9);                         % Convert to dB
PD_Rx        = [zeros(nt/2,1); F.*atten.*PA_PD; zeros(nt/2,1)];     % Filtered PD Rx
chirp (after 50dB atten)
PD_Rx_dB     = 20.*log10(abs(PD_Rx)+1e-9);                          % Convert to dB
% Simulation of ice layer return (delayed to the left of ideal_Rx)
% Simulation of surface clutter (delayed to the left or right of ideal_Rx)
delay       = round(nt/24);      % Delay of weak return
att_tar     = 1.75e-3*atten;     % dB attenuation level of weak return
layerTUK    = [zeros(nt/2-delay,1); att_tar.*F.*PA_TUK; zeros(nt/2+delay,1)];    %
Delayed weak return (internal layer)
layerPD     = [zeros(nt/2-delay,1); att_tar.*F.*PA_PD; zeros(nt/2+delay,1)];     %
Delayed weak return (internal layer)
```

```matlab
tarTUK_dB  = 20.*log10(abs(layerTUK)+1e-9); % Convert to dB
tarPD_dB   = 20.*log10(abs(layerPD)+1e-9); % Convert to dB
% Convolution of Ideal Tx and Rx Chirp - Range Compressed
RC_ideal      = fftshift(ifft(fft(ideal_Rx).*conj(fft(ideal_Tx))));
normalization = max(abs(RC_ideal));
RC_ideal_Tx_dB = 20.*log10(abs(RC_ideal)./normalization);
% Convolution of TUK_Tx and TUK_Rx Chirp - Range Compressed
RC_TUK        = fftshift(ifft(fft(TUK_Rx).*conj(fft(TUK_Tx))));
RC_TUK_dB = 20.*log10(abs(RC_TUK)./max(abs(RC_TUK)));   % Normalized dB
% Convolution of PD_Tx and PD_Rx Chirp - Range Compressed
RC_PD = fftshift(ifft(fft(PD_Rx).*conj(fft(PD_Tx))));
RC_PD_dB = 20.*log10(abs(RC_PD)./max(abs(RC_PD)));      % Normalized dB
% Convolution of TUK_Tx and delayed weak return (layerTUK) - Range Compressed
RC_tarTUK = fftshift(ifft(fft(layerTUK).*conj(fft(TUK_Tx))));
RC_tarTUK_dB = 20.*log10(abs(RC_tarTUK)./max(abs(RC_TUK)));   % Normalized dB
% Convolution of PD_Tx and delayed weak return (layerPD) - Range Compressed
RC_tarPD = fftshift(ifft(fft(layerPD).*conj(fft(PD_Tx))));
RC_tarPD_dB = 20.*log10(abs(RC_tarPD)./max(abs(RC_PD)));    % Normalized dB
% Calculate Phase Results
ideal_phase   = hilbert(unwrap(angle(ideal_Tx))); % Phase values of ideal [rad]
TUK_phase     = hilbert(unwrap(angle(TUK_Tx)));   % Phase values of DDS [rad]
PD_phase      = hilbert(unwrap(angle(PD_Tx)));    % Phase values of Vout [rad]
% PLOT RESULTS   ========================================================
time_plot = 1e6*[(-nt:1:nt-1)*dt].';
tp = time_plot;
freq_plot = [1:1:nt].'.*Fs./nt;
% CREATE SUBPLOTS ========================================================
% Plot Transmit Waveforms
figure; subplot(3,2,1);
plot(tp,abs(ideal_Tx));hold on
plot(tp,abs(PD_Tx),'-g')
plot(tp,abs(TUK_Tx),'-r');hold off
axis([tp(1) tp(end) -0.1 max(abs(ideal_Tx))+0.1]);grid
title('Magnitude of Transmit Waveforms');
xlabel('Time [seconds]');
ylabel('Ideal, Tukey & PD Tx [Normalized]')
% Plot Return Waveforms
subplot(3,2,3);
plot(tp,abs(TUK_Rx),'-r');hold on
[AX,H3,H4]  = plotyy(tp,abs(ideal_Rx),tp,abs(layerTUK),'plot');hold off
set(get(AX(1),'Ylabel'),'String','Ideal Response [linear mag]')
set(get(AX(2),'Ylabel'),'String','Target Response [linear mag]')
axis(AX(1),[tp(1) tp(end) -atten/5 max(abs(TUK_Rx))+atten/5])
axis(AX(2),[tp(1) tp(end) -att_tar/5 max(abs(layerTUK))+att_tar]);grid
tarTUK_dBr    = max(tarTUK_dB) - max(TUK_Rx_dB);
text(4.5,max(abs(ideal_Rx)+atten/3),strcat('Target = ',num2str(max(tarTUK_dBr)),'
dBr'))
title('Magnitude of Return Waveforms [Blackmann^2]');
xlabel('Time [usec]');
% Plot Magnitude of Amplitude Distortion (Zoomed) in dB
subplot(3,2,5);
plot(tp,ideal_Tx_dB);hold on
plot(tp,TUK_Tx_dB,'-r');hold off
axis([tp(1) tp(end) -1 max(ideal_Tx_dB)+0.5]);grid
title('Amplitude Distortion (Zoomed)');
xlabel('Time [usec]');
ylabel('Ideal & PA Amplitude [Normalized dB]')
% Plot Pulse Compression Response
subplot(2,2,2);
plot(tp,RC_TUK_dB,'-r');hold on
[AX,H5,H6] = plotyy(tp,RC_ideal_Tx_dB,tp,RC_tarTUK_dB,'plot');hold off
set(get(AX(1),'Ylabel'),'String','Ideal Pulse Compression [dBr]')
set(get(AX(2),'Ylabel'),'String','Target 1 Pulse Compression[dBr]')
```

```
axis(AX(1),[tp(1) tp(end) -110 max(RC_ideal_Tx_dB)+5]);grid
axis(AX(2),[tp(1) tp(end) -110 max(RC_TUK_dB)+5])
title('Pulse Compression Responses');
xlabel('Time [usec]');
% Plot Pulse Compression Response (Zoomed)
subplot(2,2,4);
plot(tp,RC_ideal_Tx_dB);hold on
plot(tp,RC_tarTUK_dB,'-g')
plot(tp,RC_TUK_dB,'-r');hold off
crop = 8.5;
axis([tp(1)+crop tp(end)-crop -110 0]); grid;
title('Pulse Compression Responses (Zoomed)');
xlabel('Time [usec]');
ylabel('Magnitude [dBr]');
% PLOT WAVEFORMS IN SEPARATE FIGURES ====================================
% Plot Amplitude of Transmit Waveforms
figure; plot(tp,abs(ideal_Tx));hold on
plot(tp,abs(TUK_Tx),'-r')
plot(tp,abs(PD_Tx),'-g');hold off
axis([tp(1) tp(end) -0.1 max(abs(ideal_Tx))+0.1]);grid
title('Magnitude of Transmit Waveforms');
xlabel('Time [seconds]');
ylabel('Ideal, Tukey & PD Tx [Normalized]')
% Plot Phase of Transmit Waveforms
figure; plot(tp,ideal_phase);hold on
plot(tp,TUK_phase,'-r')
plot(tp,PD_phase,'-g');hold off
title('Phase of Transmit Waveform [Tukey]');
xlabel('Time [seconds]');
ylabel('Phase [radians]');
legend('Ideal','TUK','PD','Location','Northwest')
% Plot Magnitude of PA Amplitude Distortion in dB
figure; plot(tp,ideal_Tx_dB);hold on
plot(tp,TUK_Tx_dB,'-r')
plot(tp,PD_Tx_dB,'-g');hold off
axis([tp(1)+5 tp(end)-5 -1.0 max(ideal_Tx_dB)+0.5]);grid
title('Magnitude of PA Amplitude Distortion (Zoomed)');
xlabel('Time [usec]');
ylabel('Ideal, TUK & PD Amplitude [Normalized dB]')
legend('Ideal','TUK','PD')
% Plot TUK Return Waveforms
figure; plot(tp,abs(TUK_Rx),'-r');hold on
[AX,H7,H8]  = plotyy(tp,abs(ideal_Rx),tp,abs(layerTUK),'plot');hold off
set(get(AX(1),'Ylabel'),'String','Ideal Rx [linear mag]')
set(get(AX(2),'Ylabel'),'String','Delayed TUK Rx (Layer) [linear mag]')
axis(AX(1),[tp(1) tp(end) -atten/5 max(abs(ideal_Rx))+atten/5])
axis(AX(2),[tp(1) tp(end) -att_tar/5 max(abs(layerTUK))+att_tar]);grid
tarTUK_dBr   = max(tarTUK_dB) - max(TUK_Rx_dB);
layer_str  = ['Layer = ' num2str(max(tarTUK_dBr)) ' dBr'];
text(4.5,atten/2,layer_str)
title('Magnitude of TUK Return Waveforms [Blackmann^2]');
xlabel('Time [seconds]');
legend('Layer','Bedrock','Ideal')
% Plot PD Return Waveforms
figure; plot(tp,abs(PD_Rx),'-r');hold on
[AX,H9,H10]  = plotyy(tp,abs(ideal_Rx),tp,abs(layerPD),'plot');hold off
set(get(AX(1),'Ylabel'),'String','Ideal Rx [linear mag]')
set(get(AX(2),'Ylabel'),'String','Delayed PD_Rx (Layer) [linear mag]')
axis(AX(1),[tp(1) tp(end) -atten/5 max(abs(ideal_Rx))+atten/5])
axis(AX(2),[tp(1) tp(end) -att_tar/5 max(abs(layerPD))+att_tar]);grid
tarPD_dBr   = max(tarPD_dB) - max(PD_Rx_dB);
layer_str  = ['Layer = ' num2str(max(tarPD_dBr)) ' dBr'];
text(4.5,atten/2,layer_str)
```

```matlab
title('Magnitude of PD Return Waveforms [Blackmann^2]');
xlabel('Time [seconds]');
legend('Layer','Bedrock','Ideal')
% Plot Pulse Compression Response: Ideal, TUK, tarTUK
figure; subplot(1,2,1);plot(tp,RC_TUK_dB,'-r');hold on
[AX,H11,H12] = plotyy(tp,RC_ideal_Tx_dB,tp,RC_tarTUK_dB,'plot');hold off
set(get(AX(1),'Ylabel'),'String','Ideal Pulse Compression [dBr]')
set(get(AX(2),'Ylabel'),'String','Layer Pulse Compression [dBr]')
axis(AX(1),[tp(1) tp(end) -120 0]);grid
axis(AX(2),[tp(1) tp(end) -120 0])
title('Pulse Compression Responses with Tukey Weighting');
xlabel('Time [usec]');
legend('Layer','Bedrock','Ideal')
% Plot Pulse Compression Response: Ideal, TUK & tarTUK (Zoomed)
subplot(1,2,2);plot(tp,RC_tarTUK_dB,'-g');hold on
plot(tp,RC_TUK_dB,'-r')
plot(tp,RC_ideal_Tx_dB);hold off
crop    = 8.5;
axis([tp(1)+crop tp(end)-crop -120 0]); grid;
title('Pulse Compression Responses (Zoomed)');
xlabel('Time [usec]');
ylabel('Magnitude [dBr]');
legend('Layer','Bedrock','Ideal')
% Plot Pulse Compression Response: Ideal, PD, tarPD
figure; subplot(1,2,1);plot(tp,RC_PD_dB,'-r');hold on
[AX,H11,H12] = plotyy(tp,RC_ideal_Tx_dB,tp,RC_tarPD_dB,'plot');hold off
set(get(AX(1),'Ylabel'),'String','Ideal Pulse Compression [dBr]')
set(get(AX(2),'Ylabel'),'String','Layer Pulse Compression [dBr]')
axis(AX(1),[tp(1) tp(end) -120 0]);grid
axis(AX(2),[tp(1) tp(end) -120 0])
title('Pulse Compression Responses with Amplitude PD');
xlabel('Time [usec]');
legend('Layer','Bedrock','Ideal')
% Plot Pulse Compression Response: Ideal, PD & tarPD (Zoomed)
subplot(1,2,2);plot(tp,RC_tarPD_dB,'-g');hold on
plot(tp,RC_PD_dB,'-r')
plot(tp,RC_ideal_Tx_dB);hold off
crop    = 8.5;
axis([tp(1)+crop tp(end)-crop -120 0]); grid;
title('Pulse Compression Responses (Zoomed)');
xlabel('Time [usec]');
ylabel('Magnitude [dBr]');
legend('Layer','Bedrock','Ideal')
% Plot Pulse Compression Response: Ideal, TUK & tarTUK (Zoomed to
% Side-lobe)
figure;
subplot(1,2,1);plot(tp,RC_tarTUK_dB,'-g');hold on
plot(tp,RC_TUK_dB,'-r')
plot(tp,RC_ideal_Tx_dB);hold off
axis([tp(1) tp(end) -90 -45]); grid;
title('Pulse Compression with Tukey Weighting (Zoomed)');
xlabel('Time [usec]');
ylabel('Magnitude [dBr]');
legend('Layer','Bedrock','Ideal')
% Plot Pulse Compression Response: Ideal, PD & tarPD (Zoomed to Side-lobe)
subplot(1,2,2);plot(tp,RC_tarPD_dB,'-g');hold on
plot(tp,RC_PD_dB,'-r')
plot(tp,RC_ideal_Tx_dB);hold off
axis([tp(1) tp(end) -90 -45]); grid;
title('Pulse Compression with Amplitude PD (Zoomed)');
xlabel('Time [usec]');
ylabel('Magnitude [dBr]');
legend('Layer','Bedrock','Ideal')
```

REFERENCES

[1]   B. Douglas, M. Kearney and S. Leatherman. "Sea Level Rise: History and Consequences."
        International Geophysics Series, Vol. 75, Academic Press: San Diego, CA. 2001.

[2]   B. Douglas and W.R. Peltier. "The Puzzle of Global Sea-Level Rise." *Physics Today*.
        pp. 35-40, March 2002.

[3]   John Paden. "Bistatic/Monostatic Synthetic Aperture Radar for Ice Sheet
        Measurements." Master's Thesis. University of Kansas. April 18, 2003.

[4]   W. Blake, J. Ledford, and et al. "A VHF Radar for Deployment on a UAV for Basal
        Imaging of Polar Ice." CReSIS, University of Kansas, USA.

[5]   Ben Panzer. "Development of an Electrically Small Vivaldi Antenna: The CReSIS Aerial
        Vivaldi (CAV-A). Master's Thesis. University of Kansas, USA. December 2007.

[6]   N. Dye and H. Granberg. Radio Frequency Transistors, Principles and Applications.
        2nd edition. Woburn, MA: Butterworth-Heinemann. 2001.

[7]   Steve Cripps. RF Power Amplifiers for Wireless Communications.
        Norwood, MA: Artech House. 2006.

[8]   Robert Karlsson. "Digital Predistortion of Semi-linear Power Amplifier."
        Swedish Defence Research Agency. Linköpings, Sweden. November 2004.

[9]   Manoj Mhala, Piyush Desai and Girish Kumar. "Broadband Impedance Matching
        Network for RF Power Amplifiers." IIT Bombay, India.

[10]   Nathan Sokal. "Class-E Power Amplifiers." QEX. Jan/Feb 2001.

[11]   M. Kowatsch, and et. "Effect of Fresnel Ripples on Sidelobe Suppression in Low Time-
        Bandwidth Product Linear FM Pulse Compression." *IEEE PROC*., Volume 129,
        Pt. F, No. 1, February 1982.

[12]   Thanassis Misaridis and Jorgen Arendt Jensen. "Use of Modulated Excitation Signals in
        Medical Ultrasound. Part II: Design and Performance for Medical Imaging
        Applications. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency
        Control,* Volume 52, No. 2, February 2005.

[13]   C.G. Gentzler and S.K. Leong. "Broadband VHF/UHF Amplifier Design Using
        Coaxial Transformers." *High Frequency Electronics*. May 2003.

[14]   Shawn Stapleton. "Adaptive Feedforward Linearization for RF Power Amplifiers-Part 1."
        Agilent EEsof EDA. Agilent Technologies. November 2000.

[15]   Christopher B. Haskins. "Diode Predistortion Linearization for Power Amplifier RFICs in
        Digital Radios." Master's Thesis. Virginia Polytechnic Institute and State
        University, USA. April 2000.

[16]   Young Kim, Ik-Soo Chang and Yong-Chae Jeong. "An Analog Predistortion Linearizer
        Design." *Microwave Journal*. February 2005.

[17]   "Guide to Digital Predistortion." Agilent Technologies. September 2004.

[18]   "Linearization DesignGuide." Agilent Technologies. 2003.

[19]   "AD9910 Datasheet." Revision B. Analog Devices. December 2008.

[20]   "SP202 Datasheet." Revision 1. Polyfet RF Devices. October 2007.

[21]   "SP203 Datasheet." Revision 1. Polyfet RF Devices. October 2007.

[22]   "SD703 Datasheet." Revision 1. Polyfet RF Devices. October 2007.

[23]   Guillermo Gonzales. Microwave Transistor Amplifiers. 2nd edition.
        Upper Saddle River, NJ: Prentice Hall. 1997.

[24]   S.K. Leong. "Stabilizing Mosfet Amplifiers." 1997 MTT Presentation.
        Polyfet RF Devices. 1997.

[25]   Jerry Sevick. Transmission Line Transformers. 4th edition.
        Raleigh, NC: Scitech Publishing. 2006.

[26]   Chris Trask. "Designing Wide-band Transformers for HF and VHF Power Amplifiers."
        QEX. Mar/Apr 2005.

[27]   Vrej Barkhordarian. "Power MOSFET Basics." International Rectifier.

[28]   H. Yi and S. Hong. "Design of L-Band High Speed Pulsed Power Amplifier Using LDMOS
        FET." *Progress in Electromagnetics Research M*, vol. 2, pp. 153-165, 2008.

[29]   "Spice Model – 1008HQ." Document 158. Coilcraft. March 2009.

[30]   "Suppressing MOSFET Gate Ringing in Converters : Selection of a Gate Resistor."
        Application Bulletin AB-9, Rev. 1.0.2. Fairchild Semiconductor. 1998.

[31] Steve Mappus. "DV/DT Immunity Improved in Synchronous Buck Regulators."
Power Electronics Technology. July 2005.