DESIGN GUIDE FOR CMOS PROCESS ON-CHIP 3D INDUCTOR USING THRU-WAFER VIAS

By

Gary VanAckern

A thesis

submitted in partial fulfillment

of the requirements for the degree of

Master of Science in Electrical Engineering

Boise State University

May 2011

BOISE STATE UNIVERSITY GRADUATE COLLEGE

DEFENSE COMMITTEE AND FINAL READING APPROVALS

of the thesis submitted by

Gary VanAckern

Thesis Title: Design Guide For CMOS Process On-Chip 3D Inductor Using Thru-wafer Vias

Date of Final Oral Examination: 09 March 2011

The following individuals read and discussed the thesis submitted by student Gary VanAckern, and they evaluated his presentation and response to questions during the final oral examination. They found that the student passed the final oral examination.

R. Jacob Baker, Ph.D.	Chair, Supervisory Committee
Amy Moll, Ph.D.	Member, Supervisory Committee
Vishal Saxena, Ph.D.	Member, Supervisory Committee

The final reading approval of the thesis was granted by R. Jacob Baker, Ph.D., Chair of the Supervisory Committee. The thesis was approved for the Graduate College by John R. Pelton, Ph.D., Dean of the Graduate College.

ACKNOWLEDGEMENTS

It has been a great honor and privilege to complete my Masters of Science Degree in Electrical Engineering (MSEE) at Boise State University (BSU).

First of all, I would like to thank my advisor, Dr. Jake Baker, who has provided me with countless hours of quality instruction, guidance, and motivation to complete this thesis. I thank Dr. Saxena for his encouragement and continued support while attending BSU.

I would like to extend a special thanks to Dr. Amy Moll for accepting me to work on the Darpa research project at BSU. This support provided the core funding for the 3D Thru-Wafer Interconnect (TWI) research and provided the opportunity for my research. Additional thanks goes to Research Triangle Inc (RTI) for fabricating the devices in their silicon process.

I also would like to thank my close friend Claude R. Swarthout whose friendship spurred me to attend graduate school. At the same time, I would be very remiss had I not expressed my sincere gratitude to my extremely supportive wife and daughter for the many hours spent away from home while completing this program. Finally, I would like to thank my parents for everything that parent's do that they never get credit for.

ABSTRACT

Three-dimensional (3D) inductors using high aspect ratio (10:1) thru-wafer via (TWV) technology in a complementary metal oxide semiconductor (CMOS) process have been designed, fabricated, and measured. The inductors were designed using 500μm tall vias with the number of turns ranging from 1 to 20 in both a wide and narrow trace width to space ratios. Radio frequency characterization was studied with emphasis upon de-embedding techniques and resulting effects. The open, short, thru de-embedding (OSTD) technique was used to measure all devices. The highest quality factor (Q) measured was 11.25 at 798MHz for a 1-turn device with a self-resonant frequency (f_{sr}) of 4.4GHz. The largest inductance (L) measured was 45nH on a 20-turn wide trace device with a maximum Q of 4.25 at 732MHz. A 40% reduction in area is achieved by exploiting the TWV technology when compared to planar devices. This technology shows promising results with further development and optimization.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVATIONS

AC……..……...…….... Alternating Current.
BSU……..……Boise State University.
BW……..……..……Bandwidth.
CMOS……..……..……Complimentary Metal-Oxide Semi-Conductor.
CVD…….……...…….... Chemical Vapor Deposition.
CVO…….……Voltage-Controlled Oscillator.
DB……...……..... Decibels.
DQTM……...……... Dielectric Quasi-TEM Mode.
DRI……..……...……... Deep Reactive-Ion.
DUT……..……..……...……... Device Under Test.
EMF……...……... Electro-Magnetic Force.
FEA……...……... Finite Element Analysis.
GPIB……..……...……... General Purpose Interface Bus.
GSG……..……...……... Ground-Signal-Ground.
HFSS…….……...……..... High-Frequency Simulation Solver.
IEEE-488……..….... Institute of Electrical and Electronics Engineers.
ISS……..……...……... Impedance-Sub-Strate.
LNA……Low-Noise Amplifier.
LRM……..……..……...……... Load-Reflection-Match.
LRRM……..……...…….... Load-Reflection-Match-Match.
MCNC……...……... Microelectronics Center of North Carolina.
MMIC……...……... Micro-Machined Integrated Circuit.
MOCVD……...……... Metal Organic Chemical Vapor Deposition.
NT……...……... Narrow-Trace.
OPD……..……..……... Open-De-embedding.
OSD……..……..……...……... Open-Short De-embedding.
OSTD……...……..……...……... Open-Short-Thru De-embedding.

PGS……..……...…….... Patterned Ground Shield.

RF……High Frequency.

RLC……..……...…….... Resistive-Inductive-Capacitive.

SEM_1……...…….... Skin-Effect Mode.

SEM_2……..……...…….... Scanning Electron Microscope.

SOLT……..……..……...…….... Short-Open-Load-Thru.

SWM……...…….... Slow-Wave Mode.

TiN……..……...…….... Titanium Nitride.

TRL……..……..…...…….... Thru-Reflection-Load.

TWV……...……Thru-Wafer Via.

U.S. ……..……...…….... United States.

WT……..……...…….... Wide-Trace.

3D……...…….... Three Dimensional.

LIST OF SYMBOLS

AMD……...…….... Arithmetic Mean Distance.

A ……...…….... Area.

A_{sw} ……...…….... Side-Wall Area.

GHz……..……...…….... Giga-Hertz.

GMD……...…….... Geometric Mean Distance.

G_{si}…….……...…….... Silicon Conductance.

HF……...…….... High Frequency.

Hz…….……...…….... Hertz.

C ……..…..……...…….... Capacitance.

C_{shld}……...…….... Capacitance with Patterned Ground Shield.

C_{ox}……...…….... Oxide Capacitance.

Cp…….……...…… Parallel (Feed-Thru) Capacitance.

$C_{si..}$…….……...…….... Silicon Self-Capacitance.

C_{sw}……...…….... Side-Wall Capacitance.

C_{up}……...…….... Underpass Capacitance.

D_{avg} ……..……...…….... Average Distance.

D_{in}……...…….... Planar Inductor Inside Distance.

D_{out}……..……..……...…….... Planar Inductor Outside Distance.

d……..…...…….... Distance.

f……..…...…….. Frequency.

f_o ……...……..... Characteristic Frequency at Q_{max}.

f_{sr} ……..Self-Resonant Frequency.

Imag()…….……...……...Imaginary Part.

L ……..……...…….... Inductance.

L_i ……..……...…….... Self-Inductance.

l……..…….....……..... Inductor Length.

l_{fix} ……..……...…….. Fixture Length.

l_{dut} ……..……...…….. DUT Length.

M_i……..……...…….. Metal Level i.

$M_{i,j}$……..……...…….. Mutual Inductance Between i and j.

N ……...…….. Number of Turns.

n ……...…….. Number of Turns over the Underpass.

nH……..…..……...…….. Nano-Henries.

pH……..……...…….. Pico-Henries.

Q ……...…….. Quality Factor.

Q_{max}……...……..... Maximum Quality Factor.

R ……..Resistance.

Real()……..…...…….. Real Part.

R_s ……..……...…….. Series Resistance.

R_{si}……...…….. Silicon Resistance.

SF_4……..……...…….. Sulfur Tetraflouride.

S-Parameters……..……...…….. Scattering Parameters.

S_{ij} ……..……...…….. Scattering Parameters Between Port i and j.

T ……..……...…….. Frequency Correction Factor.

t……..……...……...…….. Metal Thickness.

t_{eff}……..……...…….. Effective Metal Thickness.

$t_{ox,m1\text{-}m2}$……..…… Oxide Thickness Between Metal Layers 1 and 2.

w ……..……...…….. Metal Trace Width.

Y_{ij} ……..……...…….. Admittance in Terms of Port i and j.

Y_p ……..……...…….. Parallel Admittance.

Z_{ij} ……..……...…….. Impedance Between Ports i and j.

Z_l ……..……...…….. Impedance of the Load.

Z_o ……..……...…….. Characteristic Impedance(50Ω).

Γ……..……...…….. Reflection Coefficient.

δ……..……...…….. Skin Depth.

η_{dut}……..……...…….. Proportion of DUT to Fixture.

Ω ……..…….....……... Ohms.

ε ….……..……... Permittivity.

ε_o ……...……... Permittivity in Free Space.

ε_{ox} ……...……... Oxide Permittivity.

ε_r …….……...……... Relative Permittivity.

Φ_m…..……...……... Total Magnetic Flux.

σ……...……... Conductivity.

π…….…...……... PI Constant.

μ ……...……... Permeability.

μ_o ……...……... Permeability in Free Space.

μ_r ……...……... Relative Permeability.

ω ……..... Frequency in Terms of PI.

ω_o……..……... Characteristic Frequency in Terms of PI.

$|Z|$ …….…... Magnitude of Impedance.

CHAPTER 1 - INTRODUCTION

1.1 Background

The interest and proliferation of radio-frequency (RF) circuits in recent years has provided broad opportunity for development of front-end RF modules such as the voltage-controlled oscillator (VCO), low-noise assembly (LNA), transformer, filter, and regulator to support a multitude of new wireless applications [1, 2]. These RF modules have had their foundation built upon discrete passive circuit components like the high frequency (HF) inductor. In the last decade, integration of monolithic inductors built in silicon-based complementary metal oxide semiconductors (CMOS) has been realized rather than relying on their predecessor off-chip components [3]. CMOS has shown itself to be the most preferred technology due to the aggressive scaling in MOS devices and its improved performance above 1GHz [4, 5]. Multi-mode wireless technology also looks to utilize high quality CMOS inductors [50].

As devices scale, designers are challenged with producing smaller and more efficient modules while maintaining or improving circuit performance, predictability, and robustness [2]. These three design requirements directly transcend to the passive components that make up the modules and thus have fueled the quest for a much improved integrated inductor.

1.2 Motivation

While the inductive coil has been around for nearly 200 years, its wide-spread use in modern silicon-based CMOS circuits has been limited by its relatively large size (when compared to other circuit elements) and its inherent performance and integration limitations.

In order to achieve a reasonable inductance value (~10nH), the device needs to be designed and manufactured with an extremely large footprint, on the order of 250um^2. This factor alone is why most inductors are forced to be implemented off-chip. This becomes apparent in CMOS technology since increasing the inductor size both increases the manufacturing cost and produces undesired parasitic effects. Device integration is limited mostly by manufacturing process maturity while parasitic effects reduce the fundamental performance factors of an inductor. This includes a poor quality factor (Q), a reduction in self-resonant frequency (f_{sr}), and a low inductance value (L). Due to these issues, several works in recent years have focused on identifying new processing techniques and understanding how the underlying parasitics are limiting the performance of CMOS-based inductors [3]. By having this understanding in hand, circuit designers will be able to optimize and further experiment with new design solutions to achieve a better integrated inductor. This paper provides a starting point for an alternative inductor design: the 3D inductor using through-wafer via interconnects (TWV).

1.3 Thesis Organization

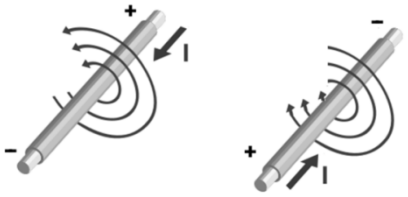
This thesis aims to provide insight into the design of a 3D integrated inductor using TWV technology versus the conventional CMOS monolithic inductor. Chapter 2 covers a review of inductor physics based upon the monolithic planer spiral topology in a CMOS process. This discussion will cover the inductive phenomena, mutual- and self-inductance, and discuss previous work regarding the electromagnetic fields present within the monolithic inductor. Chapter 3 will provide a more in-depth discussion on the specific monolithic architectures. Parasitics will be identified and their respective measurement parameters will be discussed. The chapter will round out by assembling the measurement parameters into an equivalent circuit physical model. Chapter 4 will introduce the parameter calculation methods for the equivalent circuit physical model presented in Chapter 3. Chapter 5 will present the underlining 3D inductor architecture and discuss its manufacturing process in general terms. Chapter 6 will briefly review the necessary HF measurement setup, de-embedding techniques and their impact on the accuracy of measured results. Chapter 7 will discuss the measured performance of the 3D device compared to published data on an equivalent monolithic device. Chapter 8 will show the development of an equivalent physical model for the 3D inductor. Chapter 9 will draw conclusions on device performance and discuss possible future work for continued research of the 3D inductor using TWVs.

CHAPTER 2 - INDUCTOR PHYSICS

This chapter will briefly review the inductive phenomena as it relates to conducting wires and coil inductors. A brief discussion of the following concepts will also be covered: mutual- vs. self-inductance, and the skin effect.

2.1 The Inductive Phenomena

Whether considering a straight wire, a simple coil of wound wire (solenoid), or a CMOS monolithic planar spiral inductor, when an alternating current (AC) source is applied at one terminal of a two terminal device, with the other terminal grounded, an electric current propagates back and forth through the conducting material. The current flow gives rise to a magnetic field intensity (H) and is measured in units of A/m. The alternating nature of the H with change in current direction is shown in Figure 2.1a-b.



a) **b)**

Figure 2.1 Magnetic Flux of a Current Carrying Wire Segment

The magnetic field intensity is related to the magnetic flux density (B) as seen below in Equation (2.0), where μ is absolute magnetic permeabililty (μ). Magnetic flux

density carries the units of tesla = $W_b/m = H\text{-}A/m^2$. In the case of free space, absolute permeability is simply $\mu_o = 4\pi \cdot 10^{-7} = 1.257 \cdot 10^{-6}$ (H/m).

$$B = \mu H \qquad (2.0)$$

In the absence of freespace, Equation 2.1 illustrates the change in absolute magnetic permeability. In this case of non-free space, the relative permeability (μ_r) of the introduced material scales the permeability of free space (μ_o). This equation indicates that for a relative permeability $\mu_r = N > 1$, the magnetic flux density is N-times greater in the material than it would have been in free space [51].

$$\mu = \mu_r \mu_o \ (H/m) \qquad (2.1)$$

The magnetic field flux is analogous to electric current flow whereas magnetic flux density is to voltage [52]. Unlike electric fields, magnetic fields can occur outside of material where there is an absence of free flowing electrons.

The magnetic flux (Φ_m), as shown in Equation 2.3, represents the total magnetic flux being equal to the integral of the magnetic flux density over an area of a surface S that intersects the field lines. In the special case of a planar surface, this can be simplified to (2.4) [45], where A is the cross-sectional area of intersecting surface and θ is the angle between the surface and the magnetic field lines that extend normal to the flow of current.

$$\Phi_m = \int_S B \cdot dS \qquad (2.3)$$

$$\Phi_m = BA\cos\theta \qquad (2.4)$$

Flux linkage (λ) represents the total magnetic flux passing through a surface S of a single loop of current carrying wire as seen below in Equations (2.4) and Figure 2.2, where N is the number of loops.

$$\lambda = \Phi_T \cdot \mathrm{N} \qquad (2.4)$$

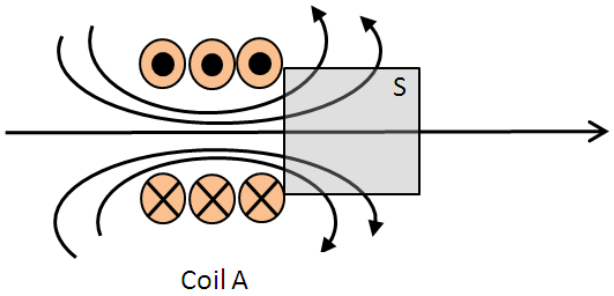


Figure 2.2 Total Magnetic Flux Through A Surface

For the case of two 3 loop tightly wound current carrying wires with the same intersecting surface S, the magnetic flux generated from each loop is passed through both loops. As such, the total magnetic flux linkage is increased by the square of the number of loops N times the magnetic flux of one loop of wire. This is shown below in Equation (2.5) and Figure 2.3 [53].

$$\lambda = \Phi_1 \cdot \mathrm{N}^2 \qquad (2.5)$$

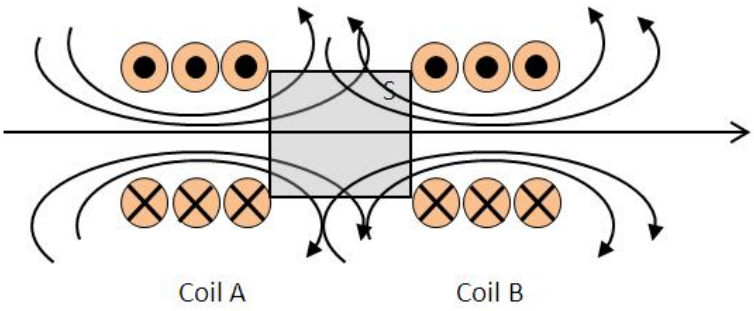


Figure 2.3 Mutual-Inductance of Two Coils

Inductance is primarily a function of geometric shape. Analagous to capacitors storing electric charge, an inductor stores magnetic energy within the core of its windings where the flux density is greatest. The quantity of inductance can be determined by the ratio of flux linkages to the current that creates the magnetic flux as shown below in Equation (2.6) for the cylindrical N_{Loop} solenoid shown in Figure 2.2 [53].

$$L = \frac{\lambda}{I} = \frac{\phi_T \cdot N}{I} = \frac{\mu \cdot N^2 \cdot \pi \cdot a^2}{h} \qquad (2.6)$$

2.2 Mutual- and Self-Inductance

In the same manner as presented in the last section, two types of inductance make up the total inductance: mutual- and self-inductance. Mutual-inductance is a result of the proximity effect occurring between two closely spaced circuits, circuit elements, or wires [21]. This can occur with two elements that are either in series or parallel and depends on the amount of flux linkages interacting between the two elements. Illustrated below in Figure 2.4 are two 3 loop coils of wire with interacting flux linkages. Coil A is being

driven by current I_A and as such is creating the flux density from coil A while coil B is not being driven, but rather receiving. The consequence of coil A on coil B is shown in Equation (2.7). The total flux Φ_{AB} from coil A to a single loop of coil B is found by integrating the flux density over the shared surface S. The flux linkages are multiplied by the number of turns N_B in coil B. The mutual-inductance in Coil B can then be found as shown in Equation (2.8).

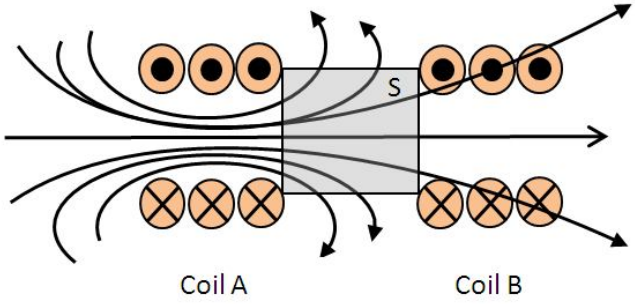


Figure 2.4 Mutual-Inductance of Two Coils

$$\lambda_{AB} = N_B \cdot \Phi_{AB} \qquad (2.7)$$

$$M_{A,B} = \frac{\lambda_{AB}}{I_A} \qquad (2.8)$$

A general Equation (2.9) has been included below for the mutual-inductance of two current carrying 2 loop coils in which both loops are being driven by a current (I_A and I_B, respectively). In this case where I_A and I_B are equal and flowing in the same direction, the equation simplifies. The total inductance of either loop A or loop B is the summation of all mutual inductances with a net result that can be either positive or negative depending on the direction of current flow.

$$L_M = \frac{\lambda_{M_{AB}} \cdot \mathrm{N}^2}{\mathrm{I_{AB}}} = \frac{\lambda_{M_{AB}} \cdot 2^2}{\mathrm{I_{AB}}} = \frac{M_{Aa}}{\mathrm{I_A}} + \frac{M_{Ab}}{\mathrm{I_A}} + \frac{M_{Bb}}{\mathrm{I_B}} + \frac{M_{Ba}}{\mathrm{I_B}} = \frac{2 \cdot M_{AB}}{\mathrm{I_{AB}}} \qquad (2.9)$$

Self-inductance is a special case of mutual-inductance and is referred to in literature as simply inductance [7]. In this case, rather than two circuit elements, two or more individual wire segments have influence on one another within the same element. Thus, the inductance present is understood to be self-inductance and occurs within its own turns rather than from an outside source. This is shown below in Equation (2.10).

$$L_{AB} = \frac{\lambda_{\mathrm{AB}}}{\mathrm{I}} = \frac{\Phi_{S_{A,B}}}{\mathrm{I}} = L_A + L_B \qquad (2.10)$$

The total inductance of a circuit or circuit element is the summation of all mutual- and self-inductances with a net result that can be either positive or negative as shown below in Equation (2.11) [7].

$$L_{Total} = \frac{\Phi_{S_{A,B}}}{\mathrm{I}} + \frac{\Phi_{M_{A,B}}}{\mathrm{I}} = L_A + L_B \mp 2 \cdot M_{AB} \qquad (2.11)$$

Extending the principles of mutual- and self-inductance to the case of a 1.25 turn monolithic planar spiral inductor as seen below in Figure 2.5, each line segment contributes a self-inductance component. Each line segment pair that has currents traveling in the same direction contribute a positive mutual-inductance term, while line pairs that have opposite direction currents contribute a negative mutual-inductance term. The total inductance is again the summation of all self- and mutual-inductances as shown in Equation (2.12).

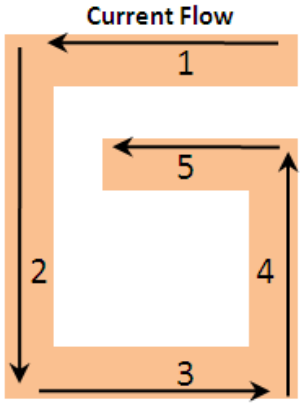


Figure 2.5 Mutual- and Self-Inductance of a Planar Spiral Inductor

$$L_{Total} = \frac{\Phi_{S_{A,B}}}{\mathrm{I}} + \frac{\Phi_{M_{A,B}}}{\mathrm{I}} = L_1 + L_2 + L_3 + L_4 + L_5 - 2\left(M_{1,3} + M_{2,4} + M_{3,5}\right) + 2M_{1,5} \qquad (2.12)$$

In general, the summation process is extended to as many segments as are contained in each of the individual elements. The larger the circuit, the more complex this becomes computationally.

2.3 Skin Effect

The phenomenon of the skin effect is well documented and is the tendency of current to flow on the surface or *skin* of a conductive material [21, 53]. In the case of a single cylindrical conductor, current flow is on the conductor's outer surface. For any given material, the depth of current flow, skin depth (δ), is determined by the relationship of current density (J) as a function of depth (z) within the conductor. The current density, as shown in Figure 2.6, is a decaying exponential function within a semi-infinite thick slab of material that has been excited by an incident electric field in the x-direction. As seen in this figure, the materials resistivity (ρ) plays a roll by affecting the exponential

decay. The intersection of the point at which the magnitude of current density has been reduced to $e^{-1} = .37$ is defined as the skin depth. Thus, materials with lower ρ will cause the exponential to decay faster and, as a result, force the current to flow closer to the conductor's surface as seen below in Equation (2.13).

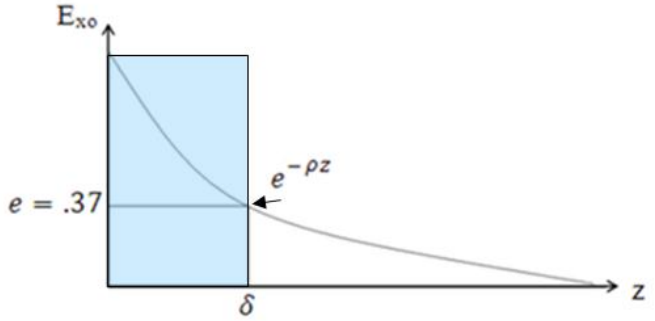


Figure 2.6 Electric Field vs. Skin Depth

$$\delta = \sqrt{\frac{\rho}{\pi \mu_r \mu_o f}} \ (\mu m) \qquad (2.13)$$

In semiconductor processing, trace resistance can be cast into terms of sheet resistance (R_{sheet}) or ohms per square where length (L) and width (w) determine one square as seen in Equation (2.14) below.

$$R = \frac{\rho}{t}\frac{L}{w} = R_{sheet}\left(\frac{\Omega}{\text{square}}\right)\frac{L}{w}(\text{square}) \qquad (2.14)$$

In the same way that metal thickness affects sheet resistance, the skin effect gives rise to a resistance (R_{skin}) that affects the trace resistance in a similar manner by changing the effective thickenss of the conductor in which current can flow. In practical terms, a semi-infinite slab, while previously assumed, is not feasible. As such, effective thickness

must be determined as seen in Equation (2.15) by calculating δ (Equation 2.13) and inserting the specific process parameter t for conductor thickness. In practice, the conductor current will not be limited or attenuated by the skin resistance if the metal thickness implemented is several skin depths thick. Inserting the effective thickness (t_{eff}) into Equation (2.16) will produce a result of the true resistance of the conductor due to skin effect in ohms/square. One additional factor that affects R_{skin} is the presense of current crowding at corners, which has the affect of decreasing the effective cross-sectional area (t_{eff}) of the conductor.

$$t_{eff} = \delta\left(1 - e^{-\frac{t}{\delta}}\right) (\mu m) \qquad (2.15)$$

$$R_{skin} = \sqrt{\frac{\rho l}{W t_{eff}}} \left(\frac{\Omega}{\blacksquare}\right) \qquad (2.16)$$

Skin effect is an important aspect of inductor design because one of the techniques for improving inductor performance is to reduce conductor resistances by increasing trace widths and/or thicknesses [8]. In doing so, the traces are made less susceptible to undesired resistive parasitics from not being able to utilize the full conductor cross-sectional area [4].

For common conductive materials like aluminum and copper, the skin depth at 1GHz is 2.59μm and 2.09μm, respectively. For quick estimates of skin depth, [9] provides a useful web-based tool. The plot seen in Figures 2.7 – 2.9 shows the impact of

frequency on skin depth, effective thickness, and resistance for various metals as previously discussed [4].

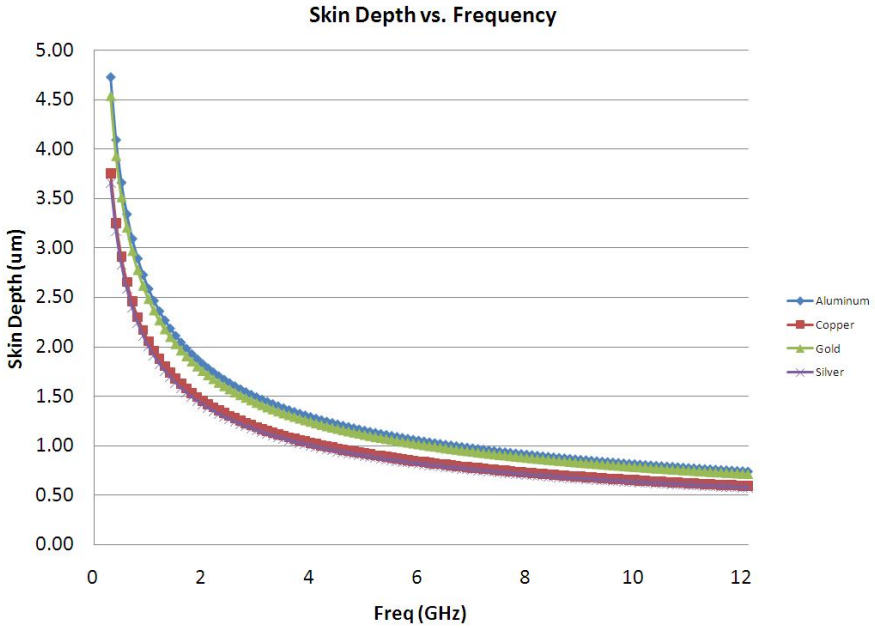


Figure 2.7 Skin Depth (δ) vs. Frequency

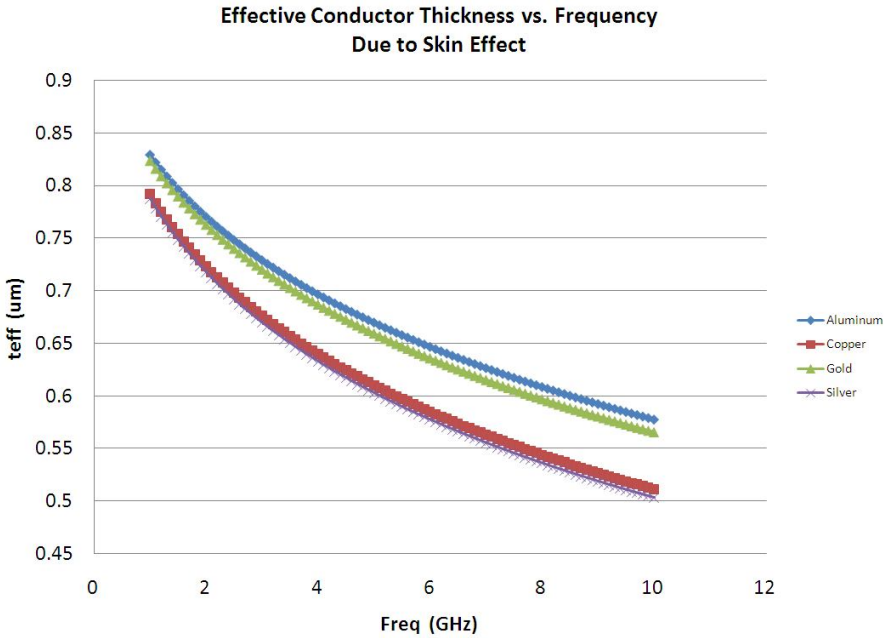


Figure 2.8 Effective Conductor Thickness (t_{eff}) vs. Frequency

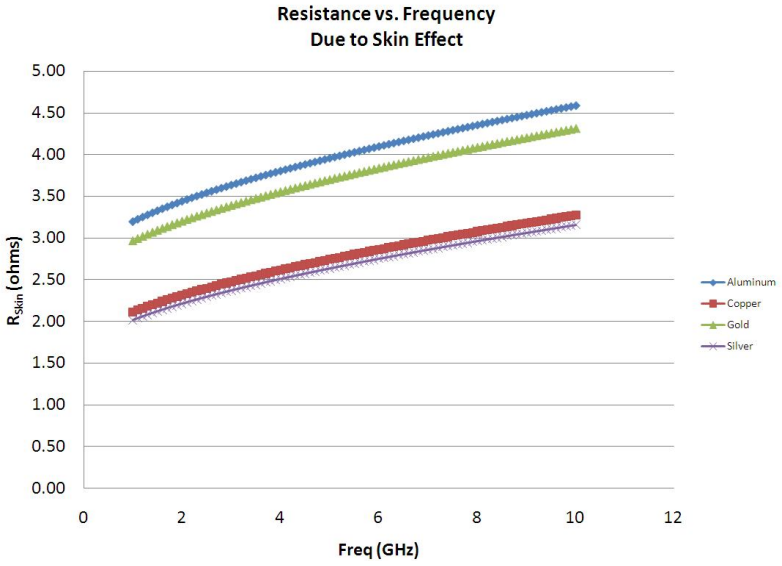


Figure 2.9 Resistance (R_{skin}) vs. Frequency

Figure 2.7 shows the relationship to the decrease in skin depth and effective conductor thickness with increasing frequency using Equations 2.13 and 2.15, respectively. Finally, Figure 2.9 graphically represents how the increase in resistance relates to frequency due to the δ and t_{eff} [4, 13]. In general, we can see that all metals exhibit a similar response; however, aluminum is affected slightly less than other materials. For this reason, many CMOS processes use aluminum for metallization.

CHAPTER 3 - MONOLITHIC CMOS PLANER INDUCTOR

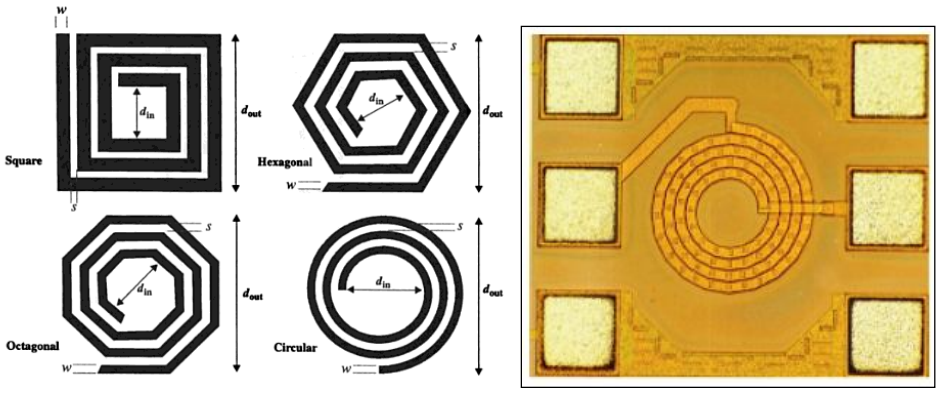
This chapter will begin by discussing various monolithic CMOS planar inductor architectures. The electromagnetic fields present within a planar inductor will be presented as they apply to their respective inductor equivalent physical circuit model elements.

3.1 Inductor Architectures

The monolithic CMOS planar inductor architecture has become widely used due to its relatively simple integration with existing CMOS capabilities and processing steps. The use of damascene processing and inter-layer vias has enabled integration of the inductor in the upper-most layers of standard multi-metal processes. This section will discuss the various planar inductor architectures being integrated in CMOS circuits. In a later chapter, the 3D inductor architecture of focus will be presented.

Figure 3.1a illustrates 4 common layout shapes found in modern inductive CMOS devices. The simplest geometry, or most commonly implemented, is the square spiral. The selection of this shape is a general result from limitations in CAD layout tools, which use simple polygons (also known as Manhattan design rules), rather than complex shapes. Subsequently, the shape selection results generally from expediency in layout rather than inductor function [23] and has the result of not necessarily producing the most efficient

designs. Research shows that circular shapes result in greater efficiency due to the reduction in current crowding within the device corners [21]. Figure 3.1b shows a circular planar inductor micrograph [11].



a) **b)**

Figure 3.1 a) Planar Spiral Inductor Shapes [23] b) Circular Inductor [17]

Figure 3.2 shows, from left to right, the layout and cross-section in 2 directions for a typical 2-turn square inductor fabricated in a 2-metal process. Cross-section A-A' illustrates an inductor fabricated in the upper-most layer of metal (M2). Upper layers of metal are typically used to implement the inductor traces because they offer lower resistance since increased metal thicknesses are allowed [13]. Another advantage to device performance is a reduction in substrate coupling with the increased distance between the metal traces and the substrate. Cross-section B-B' illustrates the M1 underpass required to pass port 2 outside of the inductor traces for subsequent connection or termination. The underpass is usually made of the lowest possible metal in order to

decrease the inter-layer capacitance between the traces and underpass. Using a lower level of metal contributes to performance degradation and is reflected by a slight lowering of the self-resonant frequency (f_{sr}).

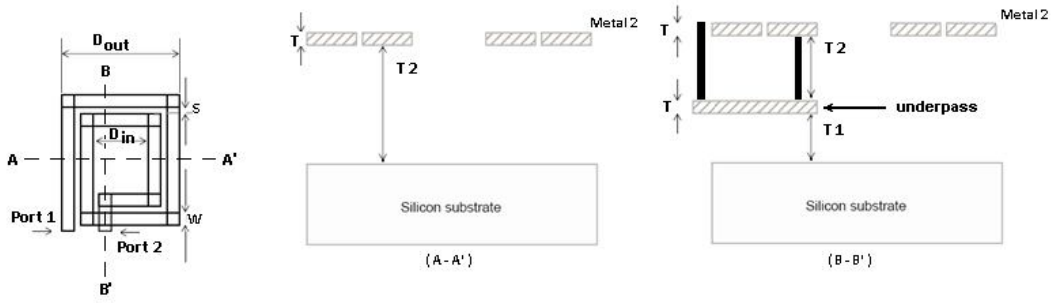


Figure 3.2 Layout and Cross-Section of 2 Metal Square Spiral Inductor [21]

Note the use of the nomenclature D_{in} and D_{out} in Figure 3.2. D_{out} represents the outer-most dimension of the square inductor while D_{in} is the inner most. Another parameter, D_{avg}, will be referred to as the arithmetic mean of the previous two parameters. This convention is widely used when comparing the inductance values against an equivalent uniform current sheet. As shown in Equation 3.1, the ratio of the outer to inner diameters is referred to in literature as the fill factor *(ρ)*. This is intuitive by inspection since ρ approaches 1 when $D_{out} \approx D_{in}$ and goes to 0 when it is becomes hollow [23].

Figure 3.3 illustrates a typical 3-metal architecture. This architecture is common when a large number of turns are needed to obtain the desired inductance value. This

technique is preferable as it reduces the series resistance by tying stacked layers of metal together. In this case, M2 and M3 are shorted mirror images as shown in cross-section A-A'. In this architecture, M1 is used for the underpass as seen in cross-section B-B'. The tradeoff is seen as an increase in interlayer capacitance [2, 3, 4, 44**]**

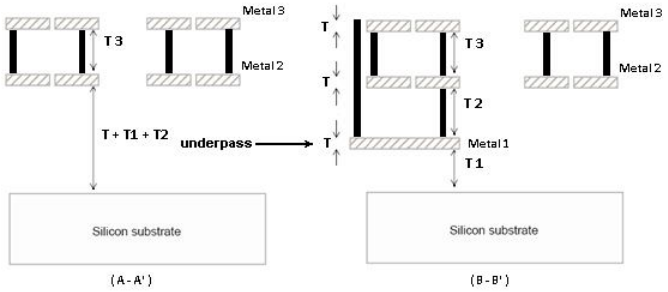


Figure 3.3 Cross-Section of Stacked and Shorted Spiral Inductor [21]

Figure 3.4 illustrates a final planar architecture. This architecture can be used to shield the electric field from termination in the substrate and is referred to in literature as a square planar spiral inductor with a patterned ground shield (PGS). Illustrated from left to right are the inductor layout followed by a pattern structure where a lower level of metal, most likely silicided poly silicon, is added just above the substrate. The PGS structure utilizes alternating N and P doped trenches placed orthogonally to the traces to oppose current flow by *shielding* the electric field from the substrate [1, 21]. This creates a high resistance return path to ground that acts to inhibit current flow in the substrate. The PGS as drawn shows a black "X", which is a metal 1 strap to ground. The ground strap provides the shortest path to ground if any current does flow. The third image is the

combination of the first two pictures using the PGS. The cross sections A-A' and B-B' show a three metal (M2, M3) shorted inductor with M1 being utilized to strap the poly-silicon to ground in facilitating the underpass [1, 44].

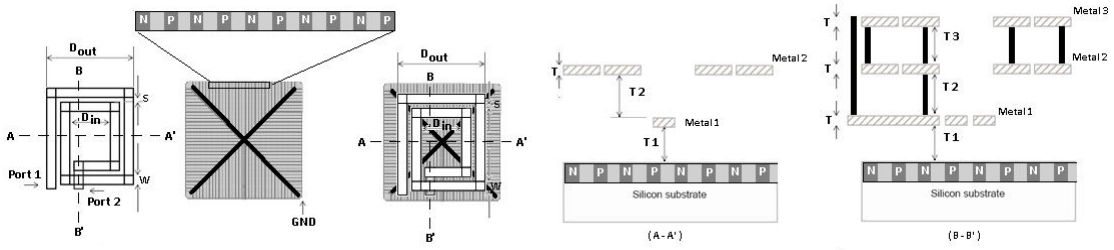


Figure 3.4 Layout and Cross-Section of Spiral Inductor with PGS [1, 21]

In addition to the three common architectures presented, there have been many other methods researched and reported to address one or more parasitic issue. The interested reader can review [14] for a proton beam isolation method while [15] can be reviewed for deep trench isolation techniques.

3.2 Modeling the Planar Inductor

Modeling inductor behavior has been a widely covered subject by many authors [12, 16, 17, 18, 22, 25, 44]; however, two primary modeling methods are used today: numerical methods, and the equivalent physical circuit model method. The numerical method is based upon finite element analysis (FEA) [4, 7, 16, 21]. This approach is based upon iterative convergence of the arithmetic mean distance and geometric mean distance of each wire segment with every other wire segment in the circuit. While accurate, this

method takes expensive software and a large amount of computing resources to complete even for small devices [17].

The planar equivalent circuit model derives its origin from the ability to completely understand all parasitic phenomena and related parameters present such that assignment of physical or simulated circuit components (resistors, capacitors, etc) are possible. This method allows for a reasonably accurate and faster model development at a reduced cost. Limitations of physical modeling can arise from undetermined high-frequency phenomena like eddy-currents [4, 16, 21]. However, developing an understanding of all parasitic factors allows for a more intuitive approach to modifying the inductor to achieve the desired design results. The physical model will be covered next.

The Physical Model

While topology choices for CMOS spiral inductors abound, the planar square spiral provides the best illustration of the electromagnetic fields exhibited and the utility of the physical model. As illustrated in Figure 3.5, one magnetic and three electric fields are produced when an AC voltage is applied [20]. The reader is referred to Figure 3.6 for the equivalent planar inductor circuit model.

The first electric field (E_1) is a result of the voltage difference between the terminal connections of the spiral and is simply due to ohmic losses in the traces [20].

This is directly dependent upon material resistivity (ρ) and is modeled as the series resistance R_S.

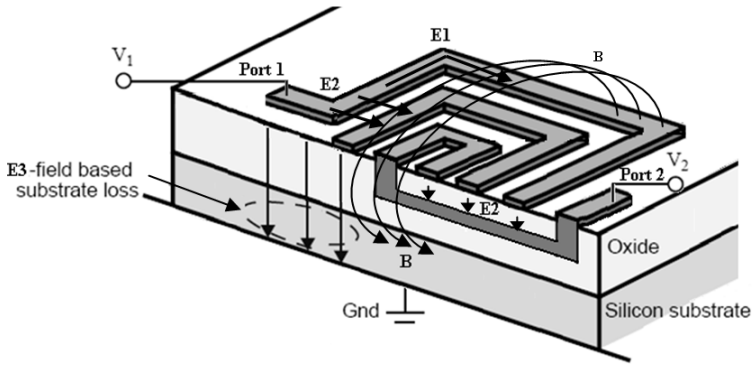


Figure 3.5 Electric and Magnetic Fields in a CMOS Planner Inductor [21]

The second electric field (E_2) is a consequence of the voltage difference between any two turns in the spiral and any individual turn and the underpass [20]. This is a consequence of the second port being connected using a lower level of metal, which induces an inter-winding parasitic capacitance due to the presence of the interlayer dielectric. The modeling parameter for E_2 is C_P [20].

The third electric field (E_3) is present due to the voltage difference between the silicon substrate and the metal of the spirals. Field E_3 induces capacitive coupling to the substrate and is oftentimes the most predominant parasitic since it extends into the substrate [20]. This is modeled as the parameter C_{OX}. The effect of this field is made worse because most CMOS circuits use low-resistivity substrates in the range of <10Ω/cm. This allows for current to flow in the substrate easily. Due to this current

flow, it is necessary to include modeling parameters for the intrinsic substrate capacitance and resistance. These parameters are identified as C_{SUB} and R_{SUB}, respectively.

The final field is the magnetic field (B) produced by the AC current that flows through the traces of the spiral. While the magnetic field is what induces the desired inductive behavior, this also creates a complementary parasitic behavior in the metal traces due to eddy-currents as discussed in Chapter 2 [4, 20, 21, 24, 44].

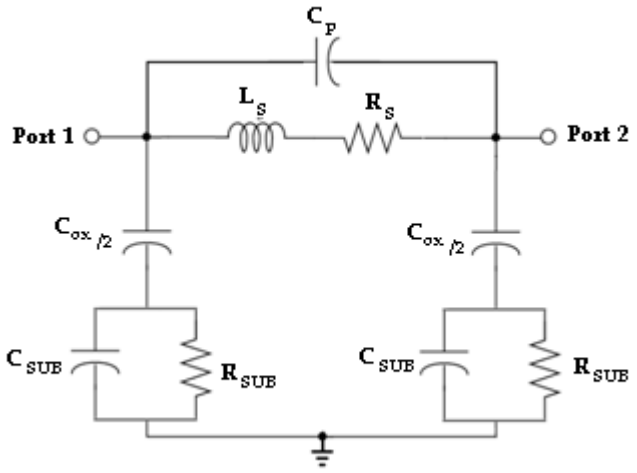


Figure 3.6 Basic Planar Inductor Equivalent Circuit Physical Model [4, 20, 21, 24]

Further inspection of the physical model in Figure 3.6 shows the presence of two sub-models as seen in Figure 3.7. The first sub-model represents the inductor in free space and would be present for any inductor. The second is the model of any conducting metal placed on top of a silicon substrate.

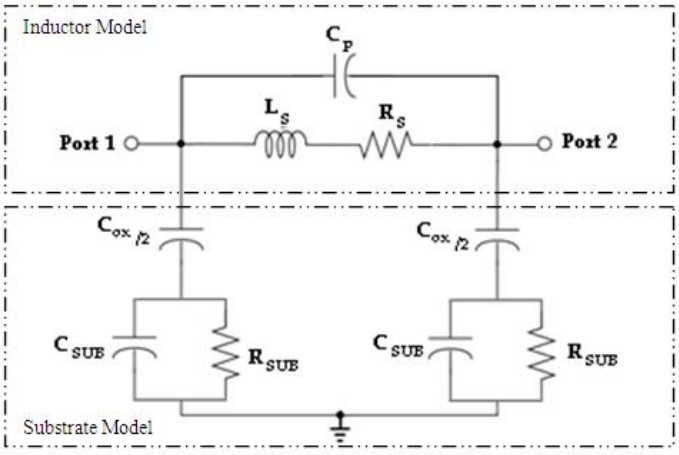


Figure 3.7 Inductor Sub-Models [4, 20, 21, 24]

As previously shown in Figure 3.4, the planar inductor architecture was modified to include a patterned ground shield that acts to shield the field E3 prior to it penetrating into the substrate [1, 2, 20, 21]. Figure 3.8 shows graphically side-by-side the comparison between the two methods while Figure 3.9 illustrates the addition of the ground shield in the equivalent physical circuit model. The corresponding model element changes involve removing $C_{OX/2}$ and replacing C_{Si} || R_{Si} with series elements C_{shld} and R_{shld}. This modification is commonly implemented to achieve higher device performance.

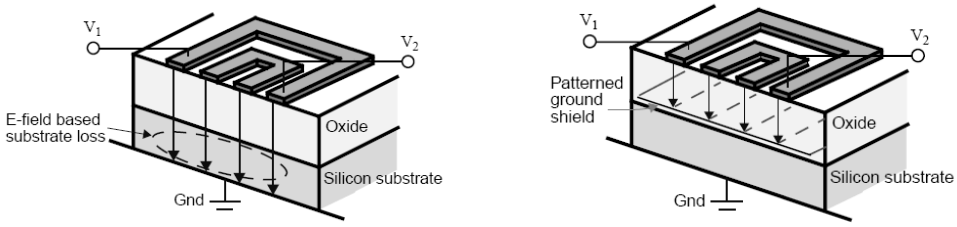


Figure 3.8 Substrate Coupling with and without PGS [21]

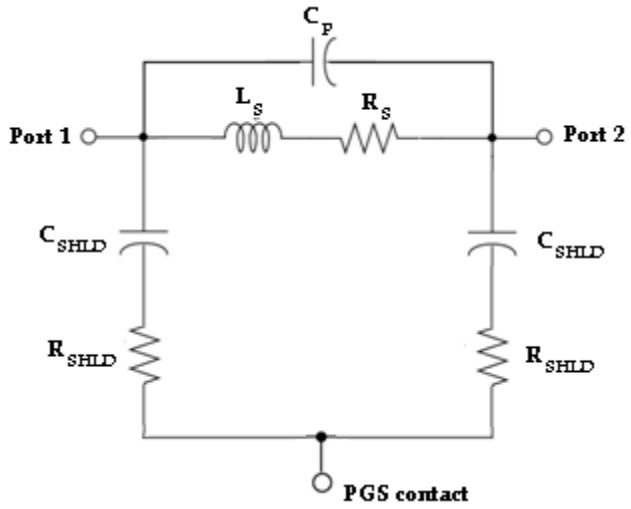


Figure 3.9 Basic Planar Inductor Equivalent Circuit Physical Model with PGS [21]

Reflecting on Figures 3.5 and 3.6, it becomes more intuitive of how to optimize device performance. The first parameter, R_S, can be reduced by utilizing low resistivity materials like copper or aluminum for metallization. Another method would be to tie two or more exact image metal layers together as illustrated in Figures 3.3 and 3.4; however, an increase in C_{ox} will be traded for the reduction in resistance as a result of the M1-M2 capacitance adding to the M1 to substrate MOS capacitance. The oxide capacitance can be reduced by increasing the inter-layer dielectric between the M1-M2 layers, by increasing the height that the inductor is placed above the substrate, or by adding a PGS [1, 15, 16, 18, 19, 21].

Next, the line trace-width to space-ratio of the traces should be as wide as possible until the sidewall capacitance C_{sw} increases with a tradeoff of decreased self-

resonant frequency. A few suggestions to reduce the feed-thru capacitance would be to increase the space between the turns and the underpass, eliminate the underpass completely, or keep the device footprint as small as possible. Elimination of the underpass in the planar architecture is not feasible without using the center tap as a bond pad, which would introduce additional parasitics. Feed-thru capacitance degrades the self-resonant frequency, thus any improvements to this parameter will help to minimize existing degradation.

A reduction in C_{OX} can be achieved by increasing the oxide thickness resulting in the inductor setting higher above the substrate. One final technique would be to increase the substrate resistivity. This, however, can make it difficult to integrate with other CMOS devices where speed and cost are factors.

CHAPTER 4 – PARAMETER CALCULATION METHODS

This chapter will briefly review the calculation methods for the modeling parameters identified in Chapter 3. Coverage of inductance will cover the most widely accepted method, which is the Greenhouse Method [11]. This chapter will include coverage for both the calculation and plot extraction methods needed to determine the key performance metrics related to inductor evaluation. Additionally, coverage of the quality factor (Q), and self-resonant frequency (f_{SR}) will be discussed.

4.1 Inductance (L) Calculation

As discussed in Chapter 2, inductance is the measure of a coil's ability to store magnetic energy within its windings and is based primarily on the magnetic flux density created from the current density. Many varying analytical formulas for calculating inductance exist in literature with some being more accurate than others. Within these formulas, it is often unclear what restrictions or boundary conditions apply to them [23]. With so many formulas available, only the Greenhouse Method will be discussed. The interested reader can review [5, 7] for a historical progression of the art of inductance calculation methods.

Greenhouse Method

In the case of non-standard geometric shapes, most designers utilize numerical techniques, curve fitting, and empirical formulas as reported in [7]. A couple of the numerical techniques utilized here are cross-sectional area integration, Taylor series expansion, and geometric mean distance. With the proliferation of silicon-based CMOS devices, Greenhouse furthered the art by developing an algorithm-based approach motivated from Grover's methods by calculating inductance as the summation of all mutual- and self-inductances of individual line segments [7]. In the case of orthogonal line segments, only a weak mutual coupling exists and thus is not considered. Basic calculations for mutual- and self-inductance were shown in Chapter 2.

Extending the Greenhouse algorithm to the case of Figure 4.1, the mutual- and self-inductance can be calculated by breaking the inductor into individual line segments. The total inductance (L_{Total}) of the inductor with five line segments is then calculated by extending Equation (2.5), where L_i represents the self-inductance of each line segment "i", and $M_{i,j}$ is the mutual-inductance between the two line segments "i" and "j".

Equation (4.1) is obtained by taking the self-inductance of each line segment, then adding the mutual-inductance for each parallel line segment pair that has current flow in the same direction, and subtracting the mutual-inductance for each parallel line segment pair that has current flowing in the opposite direction.

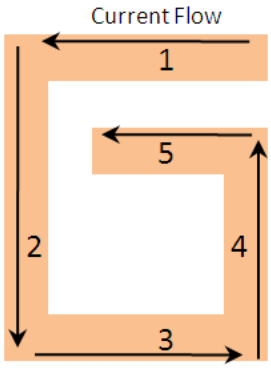


Figure 4.1 Greenhouse Method [7, 11]

$$L_{Total} = L_1 + L_2 + L_3 + L_4 + L_5 - 2\left(M_{1,3} + M_{2,4} + M_{3,5}\right) + 2M_{1,5} \qquad (4.1)$$

The above approach relies on Equations (4.2)-(4.6) reported by Greenhouse in [11] for calculating either the self- or mutual-inductance of each line segment. This relies on geometric factors:

- Conductor Width (w) in microns
- Conductor Thickness (t) in microns
- Distance Between Conductor Filaments (d)
- Relative Permeability (μ)
- Geometric Mean Distance (GMD)
- Arithmetic Mean Distance (AMD)

- Mutual Inductance Parameter (Q)

- Frequency Correction Factor (T)

$$L_i = 0.0002 \cdot l_i \left(\ln\left(2 \cdot \frac{l_i}{GMD}\right) - 1.25 + \frac{AMD}{l_i} + \mu \cdot \left(\frac{T}{4}\right) \right) \quad (4.2)$$

$$M_{i,j} = 0.0002 \cdot l_i \cdot Q_i \quad (4.3)$$

$$\ln(GMD_i) = \ln(d) - \frac{1}{12 \cdot \left(\frac{d}{w}\right)^2} - \frac{1}{60 \cdot \left(\frac{d}{w}\right)^4} - \frac{1}{168 \cdot \left(\frac{d}{w}\right)^6} - \frac{1}{360 \cdot \left(\frac{d}{w}\right)^8} \ldots. \quad (4.4)$$

$$Q_i = ln\left(\frac{l_i}{GMD} + \left[1 + \left(\frac{l_i}{GMD}\right)^2\right]^{0.5} \right) - \left[1 + \left(\frac{l_i}{GMD}\right)^2\right]^{0.5} + \frac{GMD}{l_i} \quad (4.5)$$

$$AMD = w + t \quad (4.6)$$

Revised Greenhouse Method

The above Greenhouse algorithm applies to the ideal case of a rectangular inductor in free space. In [24], Krafcsik and Dawson later revised the Greenhouse algorithm by accounting for the non-free space ground plane exhibited in CMOS devices as shown in Figure 4.2. This research brought to light the presence of a reflected image in the substrate below the ground plane interface at a distance equal to the distance the inductor sits above the ground plane. Unfortunately, this image acts to reduce overall inductance by contributing a negative mutual-inductance (negative current), as presented in Equation (4.7).

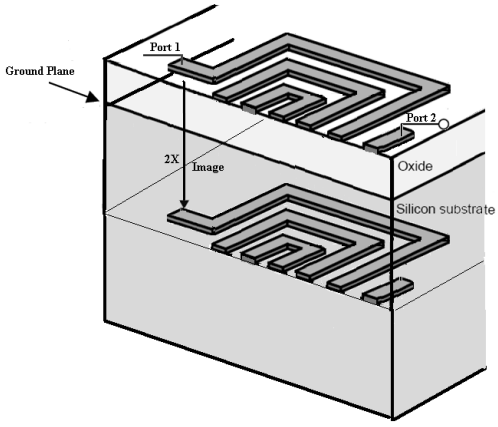


Figure 4.2 Reflected Image [24, 5]

$$L_{Total,CMOS} = \left[L_1 + L_2 + L_3 + L_4 + L_5 - 2\left(M_{1,3} + M_{2,4} + M_{3,5}\right) + 2M_{1,5} - M1,1' + M2,2' + M3,3' + M4,4' + M5,5' + M1',5'\right. \quad (4.7)$$

When more accurate results are required, the use of full magnetic wave field solvers can be used. This can be the case when implementing non-standard geometric shapes or when highly complex line segment structures are being used. However, the major drawback of this method is the length of simulation time, often taking days, and the cost of related systems and software.

4.2 Series Resistance (R_S) Calculation

At low frequencies, below ~500MHz, series resistance can be approximated by measuring the simple DC resistance or using the derived Re[Z_{11}] value from the 1-Port S_{11} measurement as shown in [49]. As operational frequencies have increased above 500MHz, it was observed that the accuracy of the model started to deviate from

expectations as it didn't remain constant. Several authors have investigated this and have identified the presence of a frequency dependent component. This dependence is understood to be a result of contributions from both the presence of skin depth and eddy-currents. As such, Yue and Wong. and Yue, et al. reported in [16, 25] that a good closed form approximation is shown in (4.9), where the following parameters are defined:

- l is the spiral length
- ρ is the resistivity at DC
- δ is the skin depth of the metal
- t is the metal trace thickness
- w is the metal trace width

$$R_S = \frac{\rho l}{W t_{eff}} = \frac{\rho l}{W\delta(1-e^{\frac{-t}{\delta}})} \qquad (4.8)$$

Equation (4.8) show the series resistance is frequency dependent upon the skin effect previously shown in Equation (2.6), the length, width, and thickness of the inductor wire trace. The series resistance can be determined from measured 1-port values by converting the S-parameter S_{11} to Z_{11} and taking the real part of the complex impedance.

4.3 Feed-Thru Capacitance (C_P) Calculation

The feed-thru capacitance value C_p takes into account the contribution from two parallel plate capacitances. The first is the capacitance created between the sidewall area and inter-winding distance (C_{sw}) (4.9), while the second is the capacitance due to the underpass (m1) and the inductors M2 traces (C_{up}) (4.10) [16]. The combined equation is shown in (4.11). Due to the small contribution from the sidewall capacitance, a good approximation for C_p is C_{up} as shown in (4.12).

- d is the horizontal distance between traces (i.e., Space length)
- 1 is the length of the inductor
- t is the metal thickness
- n is the number of turns over the underpass
- w is the line width
- ε_o is the permittivity in a vacuum
- ε_r is the relative permittivity

$$C_{sw} = \frac{(\varepsilon_o * \varepsilon_r) \cdot l \cdot t}{d} = \frac{\varepsilon_{ox} \cdot A_{sw}}{d} \qquad (4.9)$$

$$C_P = n \cdot w^2 \cdot \frac{\varepsilon_{ox}}{t_{ox,M1-M2}} \qquad (4.10)$$

$$C_P = C_{sw} + C_{up} \qquad (4.11)$$

$$C_P \approx C_{up} \qquad (4.12)$$

4.4 Oxide Capacitance (C_{OX}) Calculation

The oxide capacitance parameter represents the parallel plate capacitance created by the inductors metal traces above the silicon substrate with the SiO_2 dielectric layer sandwiched in between. The oxide capacitance is a straight forward calculation from parameters as shown in (4.13).

$$C_{ox} = \frac{(\varepsilon_o * \varepsilon_r) \cdot A}{t_{ox}} = \frac{\varepsilon_{ox} \cdot A}{t_{ox}} \qquad (4.13)$$

4.5 Silicon Resistance (R_{si}) and Capacitance (C_{si})

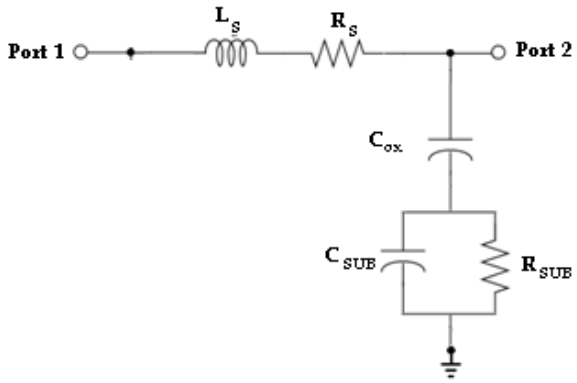
The semiconductor substrate layer resistance and capacitance are represented by the parasitic parameters R_{si} (4.14) and C_{si} (4.15), respectively. In these two equations, the additional parameters C_{sub} and G_{sub} are present. The additional parameters represent the capacitance and conductance per unit area, respectively, and can be obtained from measured data. Substrate resistance R_{si} is predominantly determined by the majority carrier concentration as determined by doping concentrations and the area the inductor occupies [12]. The substrate capacitance is the self-capacitance and is attributed to the high frequency effects occurring in the substrate [18, 25]. Additionally, R_{si} and C_{si} can be approximated as being proportional to area.

$$C_{si} = \frac{1}{2} \cdot l \cdot w \cdot C_{SUB} \quad (4.14)$$

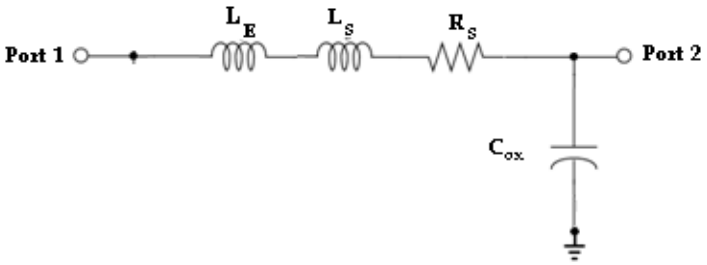
$$R_{si} = \frac{2}{l \cdot w \cdot G_{SUB}} \quad (4.15)$$

In [26], Hasegawa et al. illustrates the presence of three distinct modes of operation that affect the silicon layer resistivity with a microstrip line situated above on an SiO_2 layer. The three modes of operation are Dielectric Quasi-TEM Mode (DQTM), Skin-Effect Mode (SEM), and Slow-Wave Mode (SWM). These modes are a result of the electric and magnetic fields generated from the AC signal. The properties are thus a function of electric permittivity and magnetic permeability.

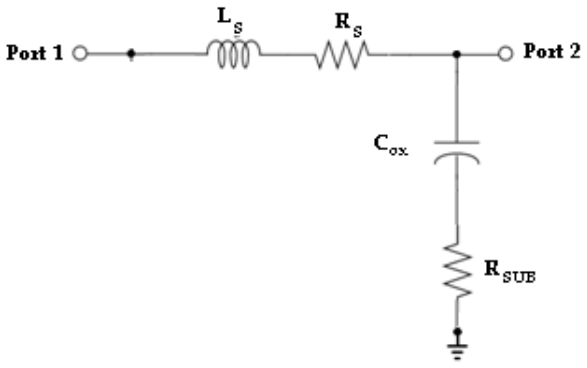
In the mode DQTM, both the frequency and substrate resistivity are typically high. In this mode, almost all of the energy is transmitted through the silicon layer as a displacement current. This is illustrated in the circuit model as shown in Figure 4.3a. In SEM, the substrate acts as a lossy conductor as illustrated in Figure 4.3b. This occurs when the quantity of the frequency and the conductivity is large. The final mode, SWM, occurs when a moderately doped substrate (<1 Ohm-cm) and frequency is being used. This mode is illustrated in Figure 4.3c. This third mode was typical of CMOS circuits until operating frequencies were raised above 500MHz. Other literature refers to this as the Quasi-Static TEM [26]. Between the three modes lays a distinct transition region as illustrated in Figure 4.4, regenerated from [26] where dual modes might be present. DQTM is the most practical mode as frequencies reach into the GHz range.



a)



b)



c)

Figure 4.3 Modes of Operation Circuit Diagrams [26]

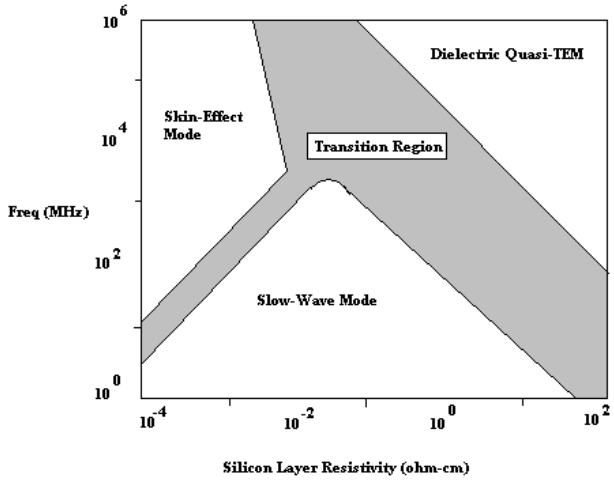


Figure 4.4 Modes of Operation [26]

4.6 Quality Factor (Q) Calculation

The quality factor (Q) is another critical parametric that abounds in formulas that have been adjusted depending on geometry, etc. However, the true meaning of Q is simply the efficiency of an inductor to store energy in spite of parasitic effects. The fundamental definition is thus based upon the energy Equation (4.15) and is a dimensionless parameter [1, 13, 25, 43].

$$Q = 2\pi \frac{|\text{Peak Magnetic Energy} - \text{Peak Electric Energy}|}{Energy\ Loss\ in\ One\ Oscillation\ Cycle} \qquad (4.15)$$

A quality factor estimate for an ideal inductor can be achieved by plotting Equation (4.16). A good approximation to this can be obtained from the complex

impedance Z_{11} from a 1-port S-parameter measurement where $\omega \cdot L_S$ represents the reactive part and R_s the real part [1, 13, 25, 43].

$$Q = \frac{\omega \cdot L_S}{R_S} \approx \frac{|X_L|}{R_S} = \frac{\text{Imag}(Z_{11})}{\text{Real}(Z_{11})} \qquad (4.16)$$

Equation (4.16) above is an approximation and does not fully account for all of the parasitic affects present when designing inductors over Si-SiO_2. As such, two additional terms need to be added to Equation (4.17) [13, 25, 43]. The first is the substrate loss factor, which accounts for the energy that is dissipated in the substrate. The second is the self-resonance factor, which accounts for both Q peaking and other reductions as discussed in the next section. As such, Equation (4.17) presents the silicon based quality factor in terms of variables discussed previously in this chapter.

$$Q = \frac{\omega \cdot L_S}{R_S} \cdot \underbrace{\frac{R_{Si}}{R_{si} + \left[\left(\frac{\omega L_S}{R_S}\right)^2 + l\right] \cdot R_S}}_{\text{Substrate Loss Factor}} \cdot \underbrace{\left(1 - \frac{R_{si}{}^2 \cdot \left(\frac{C_{ox} \cdot C_{si}}{C_{ox} \cdot +} + C_s\right)}{L_S} - \omega^2 L_S \left(\frac{C_{ox} \cdot C_{si}}{C_{ox} \cdot +} + C_s\right)\right)}_{\text{Substrate Self-Resonace Factor}} \qquad (4.17)$$

One additional method commonly used in HF designs is shown below in Equation (4.18). This results from plotting of power transfer function |Hjω| in decibels (db). Here the center frequency ω_o is found at the power transfer function peak and $\Delta\omega$ (also called

the band width BW) is found at the -3db power point where the differences between the upper and lower values of ω are extrapolated.

$$Q = 2\pi \frac{\omega_o}{\Delta\omega} \qquad (4.18)$$

4.7 Self-Resonant Frequency (f_{SR}) Extraction

The first self-resonant frequency (f_{sr}) is a critical inductor parametric. This parameter represents the first frequency at which the impedance of the inductor and the capacitor are equal in value and are thus caused to resonate. This is referred to in literature as self-resonance. Figure 4.5 shows typical Q and |Z| versus frequency inductor plots that has been stacked. Careful observation shows that from DC up to the characteristic frequency (f_o), the plot of the quality factor is inductance whereas from f_o to f_{sr} it is capacitive. In the bottom plot, it also becomes evident that f_{sr} can be determined by the peak of |Z|.

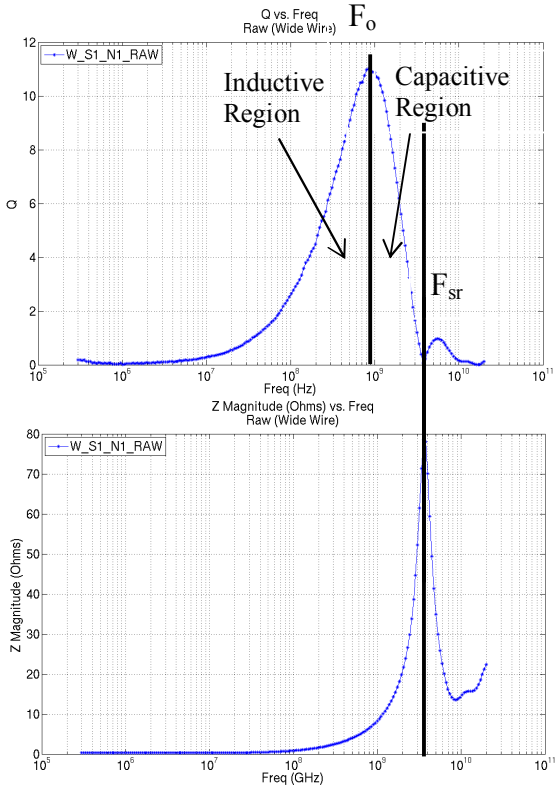


Figure 4.5 Q and |Z|

Knowing the self-resonance frequency, the capacitance at the self-resonant frequency can also be determined by Equation (4.17 and 4.18) [1].

$$f_{sr} = \frac{1}{2\pi\sqrt{L_{sr}.C_{sr}}} \qquad (4.17)$$

$$C_{sr} = \frac{1}{(2\pi f_{sr})^2 \cdot L_{sr}} \qquad (4.18)$$

CHAPTER 5 - 3D INDUCTOR

This chapter will introduce the underlying 3D inductor architecture and briefly cover the unique manufacturing process. For a detailed understanding of the necessary design cycles needed to obtain and optimize a device for manufacturing, [28-34, 36] can be reviewed for further understanding. Additionally, new design parameters will be introduced.

5.1 Architecture

Device designers have sought to improve inductor performance by pursuing 3D architectures that utilize the area above the wafer surface. In most cases, the devices have one side abutted to the silicon surface while others have attempted to use micro-machining techniques to suspend the inductor in some fashion. Unfortunately, most of these variations have tradeoffs in either manufacturability or mechanical instability that make them less preferable for implementation within a CMOS process.

The 3D inductor presented here, using through-wafer interconnect vias, offers a unique approach by using both the top and bottom wafer surfaces for patterning the metal traces. This technology relies on the ability to implement high aspect ratio TWVs for connecting the top and bottom surfaces of the wafer. Thus, a 1-turn inductor has a single trace on each the top and bottom wafer surface connected by two vias. This approach is

similar to planar technology that ties multiple stacked metal layers together with inter-layer vias. In this case, however, rather than a 1um long via through inter-layer dielectric (ILD), the via spans the thickness of the wafer (~500um). The overall inductor length is only limited by the via processing technology and final desired wafer thickness.

Figure 5.1 shows a 3D physical model of a 3-turn inductor using Ansoft's HFSS 3D full-wave electromagnetic field simulation layout tool. The material layers have been thickened here to better emphasize the individual layers present. The grey translucent box represents the silicon wafer. Orange is metal1 (M1) copper while blue is the TiN seed layer and pink is a diffusion barrier layer of parylene. As illustrated here, the inductor was laid out for 1-port ground-signal-ground (GSG) RF testing. Port 1 is shown at the left-center contact while port 2 is attached to the guard ring on the lower right contact. Notice that the guard ring is present on both the top and bottom surfaces of the wafer and is connected by TWVs. The guard ring provides isolation from extrinsic noise while taking measurements.

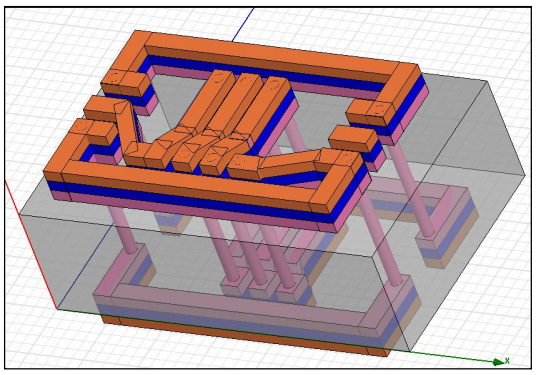


Figure 5.1 HFSS 3D Inductor Architecture

The 3D TWV design topology offers some advantages when compared to traditional planar spiral inductors. The largest of which is found in exploiting the silicon substrate. Figure 5.2 illustrates the area savings that can be realized for a 3-turn equal line-space 3D inductor. As shown, it's possible to obtain a 40% similar area device with a tradeoff in a full thickness wafer being used and thus creating a 36 times longer inductor. When a larger 3D inductor is desired, the area impact results in a 40% area increase and a 30% length increase per turn. The same 1-turn increase on a conventional planar device results in both an area and length increase of 40%.

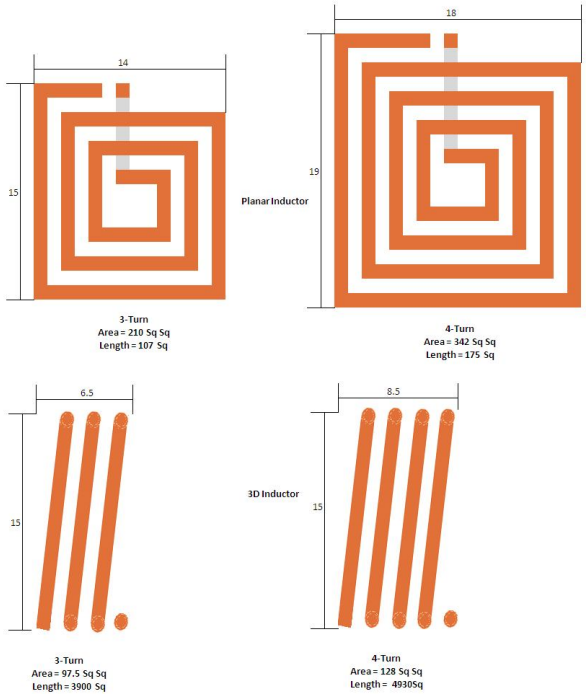


Figure 5.2 Planar vs. 3D Inductor Area

A unique advantage of the 3D inductor occurs as a result of using the TWV technology. The inductor length can be tuned in production without adjusting the layout that would otherwise have resulted in additional costs associated with spinning new reticles. This becomes apparent with realization that the backside grind process step allows for customized control of the inductor length. Additionally, the underpass is no longer needed and will eliminate the underpass coupling.

5.2 3D Inductor Fabrication

As with fabricating any new device, integration is limited by the ability of additional processing steps and techniques to be compatible with the existing CMOS process flow. The dominant CMOS processing steps rely heavily upon wet and dry etching techniques. Recent advances in both wet and dry etching techniques have made the realization of the 3D inductor using TWVs possible. While both wet and dry etching technologies have been able to create vias for decades, the advancements in plasma-based dry etching techniques have only recently allowed formation of the high aspect ratio TWV in a production environment possible. New chemistries now offer the ability for a highly selective process over a wide range of materials [27]. The 3D TWV inductor is now well suited for CMOS integration.

While most CMOS processes limit the amount of backside processing, this topology exploits the ability to use both wafer surfaces. As a result, processing steps

need to account for this with additional masks and must take measures that will afford appropriate reverse side surface protection. This will become more apparent in the device measurement chapter as it will be necessary to account for the traces on the reverse side of the wafer.

The Process Flow

The following process flow was developed in collaboration between Boise State University (BSU) and MCNC Research Development Institute under U.S. Government research contract N66001-01-C-8034 to develop TWV technology. The manufacturing process flow that follows is based upon this work. As such, the flow will only outline the basic concept of via formation and subsequent transformation into the 3D inductor. Not all details related to fabricating the via will be covered. The inductors were fabricated in the MCNC .3um process.

The Original 3D Inductor Mask Set

A 3D inductor mask set was created (Figure 5.3) that used polygon lines to pattern the inductor wire traces. It was discovered while processing the first couple of wafers that Dupont's WB5030 dry film photoresist did not have the optical resolution to adequately resolve the line width spacing without creating shorts between adjacent traces [36]. As such, a design rule was needed to set the minimum line-to-line spacing to 30um. The original mask set used 16um spacing. The reticle masks and poor results produced by this reticle set are illustrated in Figure 5.3.

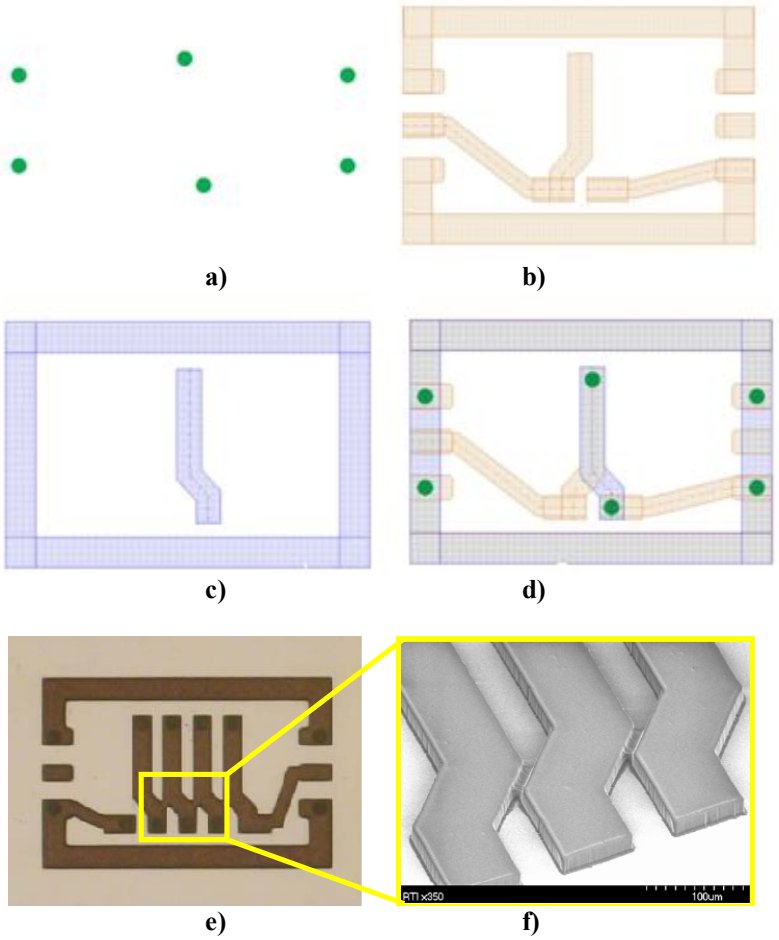


Figure 5.3 3D Inductor Masks [28]

The New 3D Inductor Mask Set

A second set of masks were created to ensure a minimum 30um line-to-line spacing design rule was implemented. Additionally, the new mask set focused on using straight-lined traces that were off-angle. Two types of inductors were developed for

measurement. The first type will be referred to as the wide trace (WT) width vs. narrow space and the other will be a narrow trace (NT) width vs. a wide space as shown in Figure 5.4d-e. The drawn line widths are WT = 70um and NT = 25um while the spaces are 30um and 100um, respectively

Figure 5.4a-c identify the masking levels where a) shows the via mask, b) shows the top metal mask with ground-signal-ground (GSG) probe pads and guard ring, and c) shows the bottom metal mask with guard ring. Figure 5.4d-e shows all masks overlaid to make up the wide and narrow trace inductors.

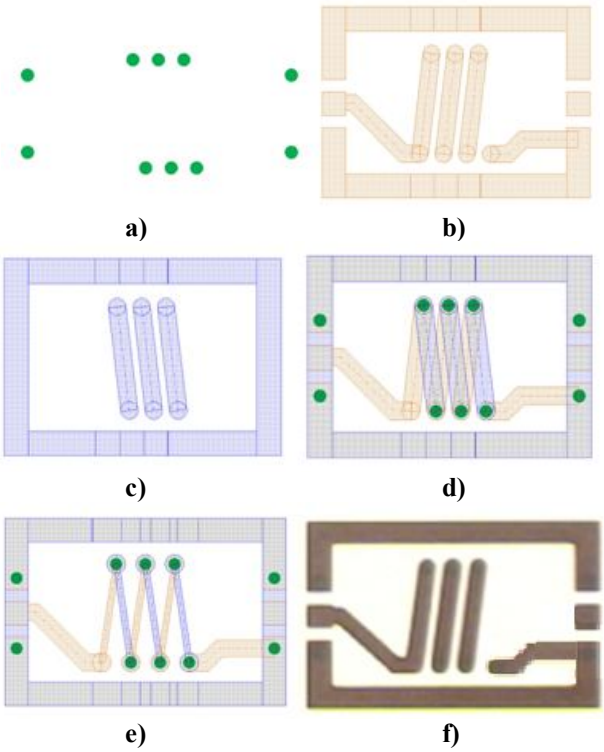


Figure 5.4 3D Inductor Masks [28]

TWV Etching

The via formation is the enabling technology for realization of the 3D TWV inductor. In most CMOS processes, the TWV would be started in the front end of the line; however, this could be tailored as dictated by the process.

Figure 5.5a-g shows the processing steps required to form the TWV. The process begins with a bare 500um wafer as shown in a). In b), a thermal oxide is grown to provide an etch stop on the backside of the wafer. A thick layer of positive develop photo resist is also deposited on the wafer backside to provide protection while in the Oxford Instruments Plasmalab Model 100 using a licensed deep reactive ion (DRI) Bosch etch process [29]. A thick layer of the same resist is then spun on the top side of the wafer c) to receive the TWV photo mask pattern. The TWV pattern is then transferred onto the wafer using a contact aligner followed by exposure and bake. The wafer is then placed in a wet developer to activate the photo resist followed by a strip and cleaning with a deionized rinse. In d), the wafer is put in the Bosch DRI tool and the via is etched using a chemistry of SF_6. The via is formed by subjecting the pattern to a predetermined number of iterative anisotropic etch and passivation steps (~500) [29]. This allows for the silicon to be removed in the vertical direction while the sidewalls stay protected. Once the oxide etch stop has been reached on the backside of the wafer, the top and bottom side resists are stripped. The TWV is etched with an aspect ratio of 10:1. The final step in the via hole formation is to remove the oxide on the backside of the wafer.

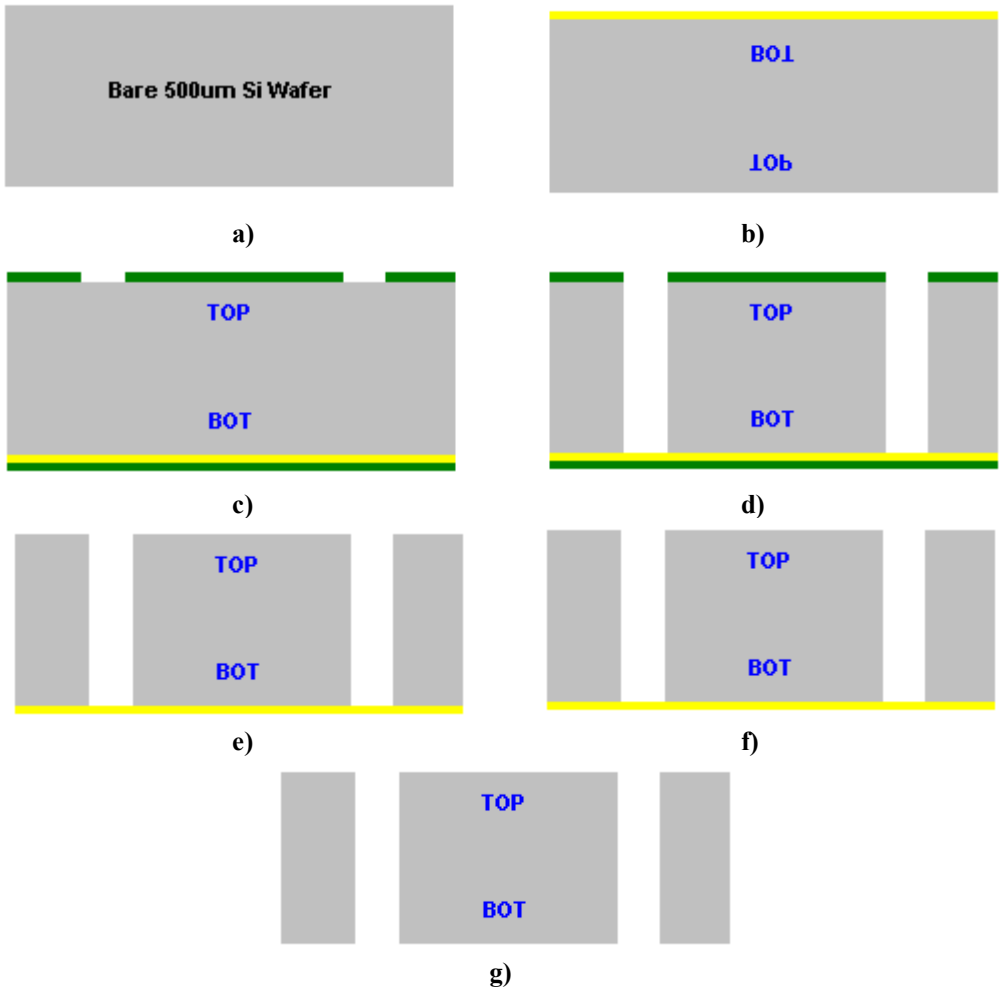
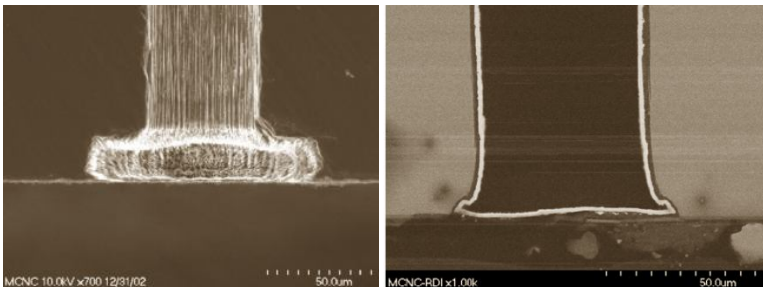


Figure 5.5 Via Formation [30]

It was observed during the final etch steps prior to punching through that degradation occurred in the via profile at the bottom of the via. The via profile was suspiciously wider. It was discovered that by using the thermal oxide as an etch stop on the backside of the wafer, excess charge buildup occurred. This resulted in a breakdown of the sidewall passivation and thus the silicon was etched in the horizontal direction. It

was also noted that not all vias etched at the same rate. As such, vias that etched faster thus continued the etching through the oxide and continued to etch into the thermal chuck. Figure 5.4a shows a scanning electron microscope (SEM) micrograph depicting the results of the oxide charge buildup on the via profile [30].

Two methods were used to mitigate this issue and provide a more robust process. The first was to reduce the RF power source. Figure 5.6b shows the corresponding improvement in the via profile. The second method was to use a sacrificial carrier wafer affixed to the bottom side of the target wafer. Affixing the carrier wafer to the target wafer occurred by using photo resist as an adhesive bonding layer.



a) **b)**

Figure 5.6 Via Profile with Oxide Charge Buildup [30]

Conformal Insulator Barrier Deposition

Copper was selected to be the metal of choice to fill the vias and pattern the inductor traces due to its lower resistivity than aluminum. With the use of copper, it becomes necessary to provide a diffusion barrier layer that prohibits the copper from

migrating into the silicon. If not inhibited, deterioration of interconnects or deep-level traps in the silicon that could lead to increased standby leakage [31].

Parylene-C was chosen as the diffusion barrier material for several reasons. The first of which is its ability to create a pinhole free conformal diffusion layer. Second, it has the ability to penetrate narrow cavities and bores of small diameter tubes. Third, it has the ability to withstand high temperature processing steps that usually occur later in a standard CMOS process flow. Finally, it has a low-k dielectric constant of ~3.15 [32]. Application of parylene as a thin film is achieved in a specialty tool under vacuum by a dedicated chemical vapor deposition (CVD) process that involves no liquid phase [33].

Parylene Deposition

Figure 5.3a-j shows the processing steps required to deposit the conformal Parylene-C liner [34]. We start with the post via etched wafer in a). In b), the Dupont WB5030 dry film photo resist is laminated to the reverse side of the wafer to protect the chuck from having Parylene deposited on it. The dry film photo resist was selected for its ability to span the open TWV hole. Using liquid photoresist would have plugged the TWV. The wafer is then placed in a CVD tool where 1um of Parylene is applied to the wafer top side as shown in c). Pane d-e) shows the removal of the backside resist and the application of resist on the wafer top side. With the resist now protecting the wafer top side, another 1um of Parylene is applied to the bottom side of the wafer as shown in f).

Dry film photoresist is again applied to the backside of the wafer in g) to allow for patterning the via mask. A punch through etch clears the bottom of the via hole as shown in h) and i). Finally, all photoresist is removed from the top and bottom of the wafer. The silicon is now protected from copper migration in subsequent steps.

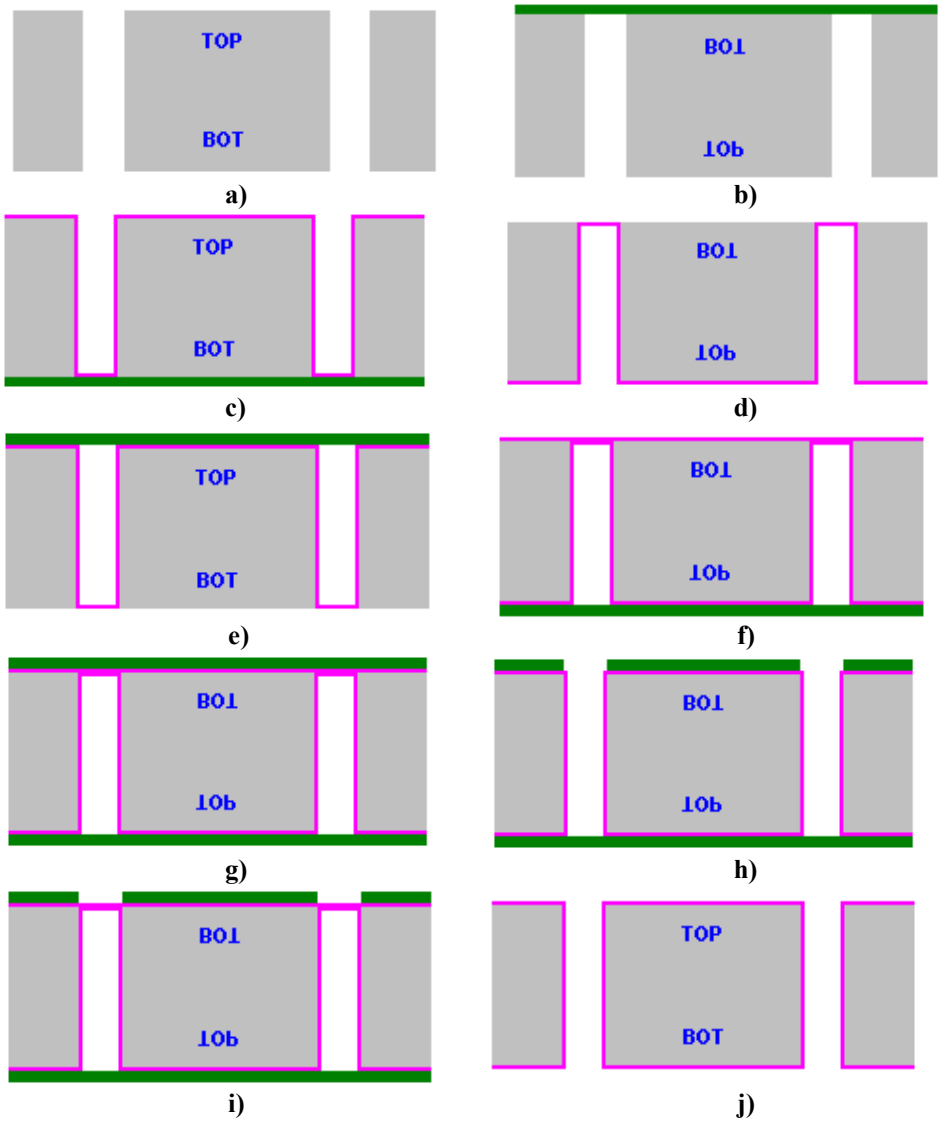
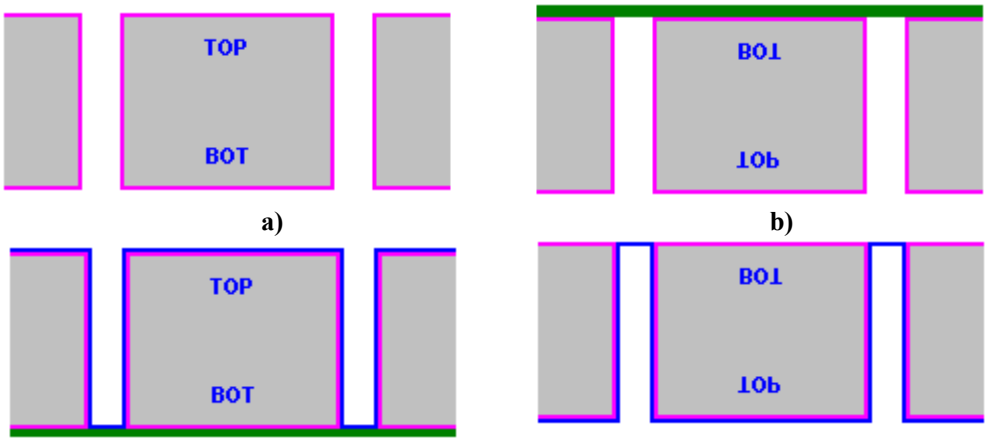


Figure 5.7 Parylene CVD Process Flow [30]

Conformal Seed Deposition

Copper is a low resistivity metal that has recently been carefully introduced into modern CMOS processes. As with most metals, it requires a precursory adhesion layer to avoid lift off and provide a reliable bond. In this case, titanium nitride (TiN) was chosen to serve this purpose. TiN is applied with a metal organic chemical vapor deposition (MOCVD) process. The MOCVD process allows for the deposition of a continuous uniform thin film on both planar and complex geometries [35]. In this application, it works equally well with both horizontal and vertical surface adhesion. TiN is deposited 1000Å thick using the identical process steps as the Parylene flow with the exception that the deposition occurs in an MOCVD tool. Figure 5.8a-j shows the steps.



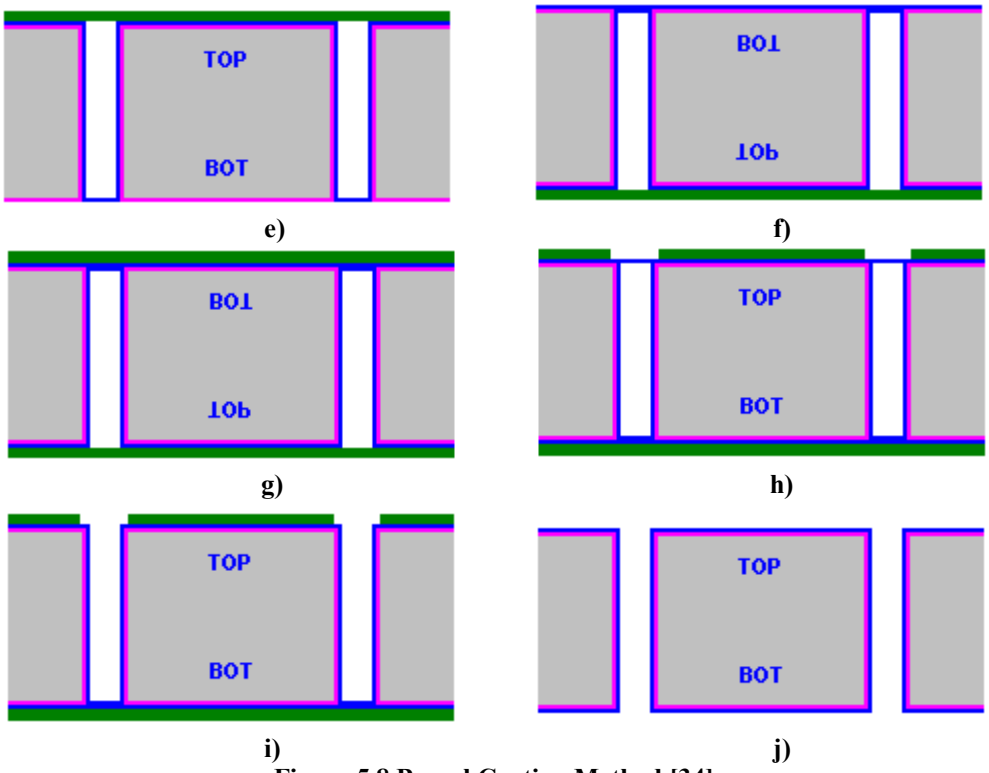
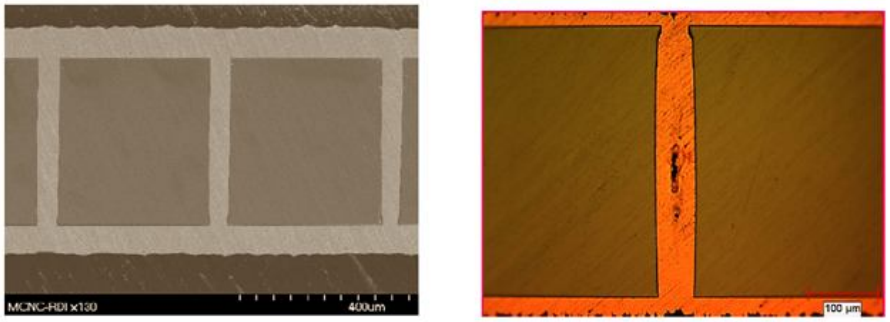


Figure 5.8 Barrel Coating Method [34]

Copper Electroplating

The via is now ready to be plated with copper. Various methods, such as physical vapor deposition (PVD), chemical vapor deposition (CVD), and sputtering are being reported in literature and commonly utilized in the IC industry. These methods work reasonably well for planar device formation using the damascene method; however, they fail to provide both defect and void free films when used to fill high aspect ratio vias. The best known method reported to date to fill the vias has been the use of reverse pulse plating that implements bottom up via filling [48]. Figure 5.9a shows a cross-section

SEM micrograph of a 50um via with an aspect ratio of 10:1. Observation shows nearly void free filling of the vias. In practice, the higher the aspect ratio, the likelihood of the vias remaining void free is decreased as shown in Figure 5.9b [30].



a) **b)**

Figure 5.9 Copper Filled TWV [30]

Due to the voiding issues and necessity of extremely high aspect vias, a secondary method, barrel coating, was developed by MCNC to provide electrical connection between the top and bottom traces within the via. This method leaves the via core open as shown in SEM micrograph Figure 5.10 [30].

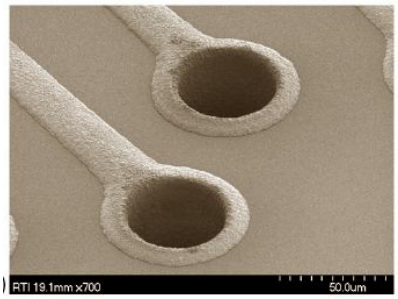
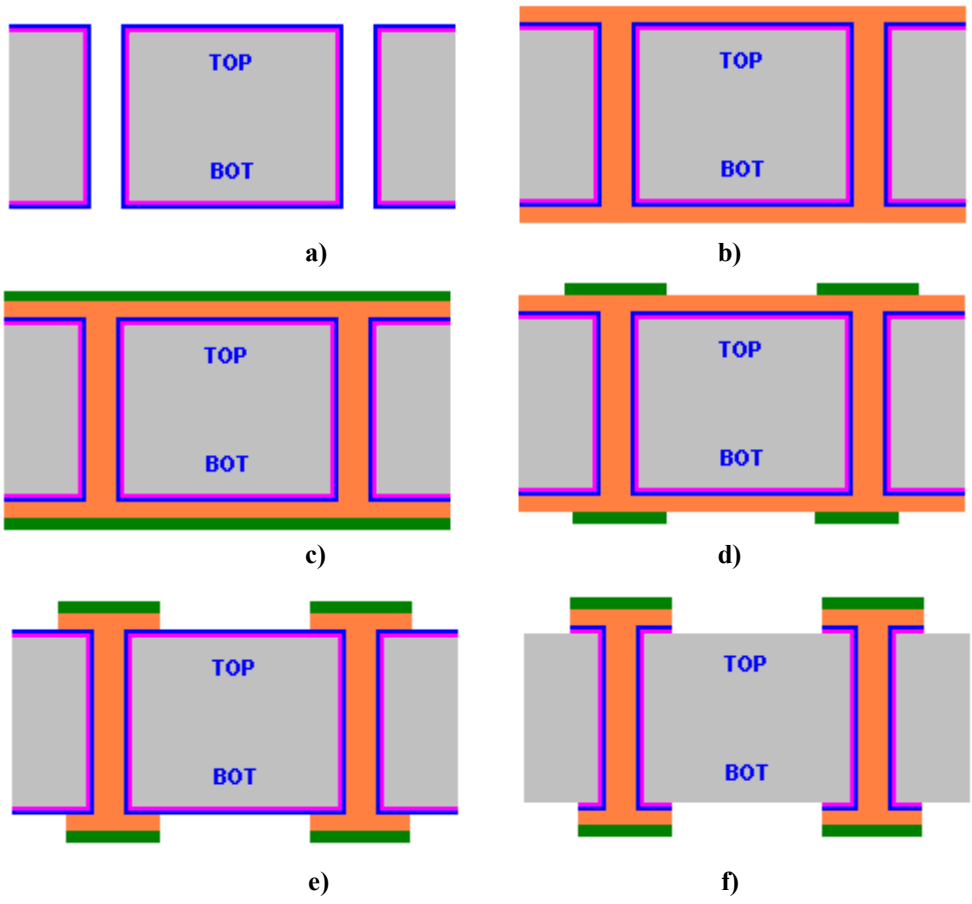
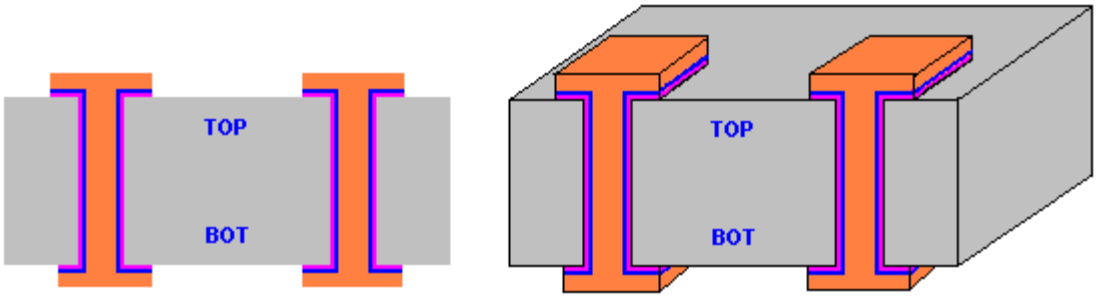


Figure 5.10 SEM of Barrel Coated TWV [30]

The final process steps are shown below in Figure 5.11a-g. The electroplating of the current carrying copper occurs in b) which results in 2-4um thick traces. This is followed by applying photoresist and patterning the inductor traces is shown in c-e. The TiN and Parylene-C layers can either be etched back to expose the bare silicon f) and g) or left for continued protection. The full 3D cross-section through the via hole is shown in h). The inductors are now ready for electrical testing.





g) **h)**

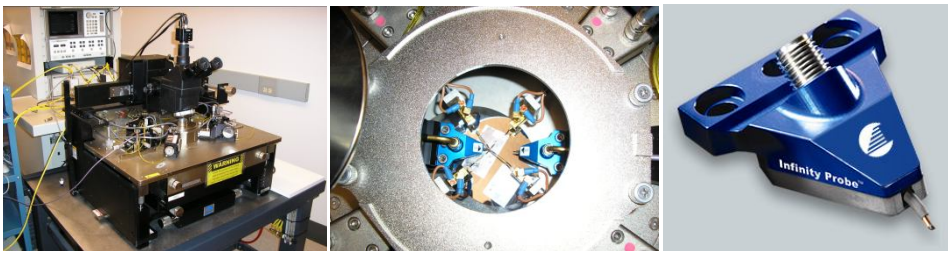
Figure 5.11 Barrel Coating Process Flow [30]

CHAPTER 6 – MEASUREMENT TECHNIQUE

This chapter will briefly review the necessary measurement equipment setup, the calibration technique employed, the de-embedding procedure implemented, and their impact on the accuracy of scattering parameters (S-parameters) measurements. In addition, some of the pitfalls with 3D device characterization will be discussed. Appendix A contains a review of the Smith Chart, (S-parameters), and parameter extraction from the same. Appendix B can be referred to for a quick review of Smith Charts and related parameters.

6.1 Equipment Setup

The equipment used to measure the 3D inductors was an HP8510C vector network analyzer (VNA) in conjunction with a manual Cascade™ Microtech Summit microwave probe station and standalone PC. The VNA is interfaced to the Summit probe station by multiple 3.5mm waveguide cables and are connected to 150um pitch ground signal ground (GSG) Infinity Probes™. Figure 6.1a shows the VNA and probe station, b) the view looking into the top-hat with both RF and DC probes, and c) a close-up of a RF GSG probe [37]. The PC is used to communicate and control the VNA thru an IEEE-488 GPIB cable. The software used to calibrate the system and measure the inductors is Cascade™ Mircotech's Wincal XE™.



a) b) c)

Figure 6.1 a) Cascade™ Microtech Probe Station. b) Top-hat View. c) GSG Probe [37].

6.2 Equipment Calibration

In order to accurately perform device characterization using the Cascade™ Microtech probe station, it is necessary to remove all RLC parasitics present between the VNA output and the Infinity Probe™ tips. In order to achieve this, all RF cables must be completely removed from the VNA output and the probe tip input. The connector ends, the outside cable jackets, and the adapters must be cleaned with isopropyl alcohol to remove any debris or hand oils that might skew the measurements. The user needs to wear disposable nitrile gloves from this point forward while performing the cleaning, during cable reinstallation, and while working with any part of the probe station. Each serial numbered RF probe comes with a calibration offset table that is unique to it. These coefficient values need to be entered into the Wincal XE software.

Cascade's Windows-based calibration software WinCal XE™ allows for automated (motorized chuck) or semi-automated (manual chuck) calibration using one of

the following four methods: Short-Open-Load-Thru (SOLT); Line-Reflect-Match (LRM); Line-Reflect-Reflect-Match (LRRM); or Transmission-Reflect-Line (TRL) methods. The interested reader can review [38, 39] for advantages and drawbacks of each method.

The SOLT calibration method provides a reasonably accurate (within 10%) measurement below 20GHz that removes parasitics from the VNA output up to the probe tips. Each calibration method above requires a corresponding impedance standard substrate (ISS). The SOLT ISS is made up of a thin piece of ceramic substrate that contains 40 gold calibration site groups as shown in Figure 6.2 [40]. Each site contains a patterned short, thru, and load structure as shown in the zoomed in window. The loads are trimmed to exactly 50 ohms by the manufacturer; however, not all sites on the ISS will provide a good calibration. As such, the manufacturer provides a paper copy of the ISS that identifies load sites that will provide exact measurements. The sites not circled will be less than 50Ω and can be used for training. The ISS is vacuum mounted on one of the auxiliary chucks. Special care must be taken to keep from destroying the probes since the auxiliary chuck sits higher than the primary wafer chuck.

The only drawback of using the SOLT method is found in measurement repeatability. Special attention must be taken when manually placing the probe tips on the ISS calibration structures and on the DUTs. The user is required to skate the probe so

it is exactly centered on the probe pads as shown in Figure 6.3a. If deviation in probe skate occurs either by over-skating as shown in 6.3b, or orthogonal mis-alignment, then the calibration can be skewed up to another 5% [38]. Thus, skate is important to measurement repeatability. Figure 6.3c shows a close up micrograph of the bottom side of the probes.

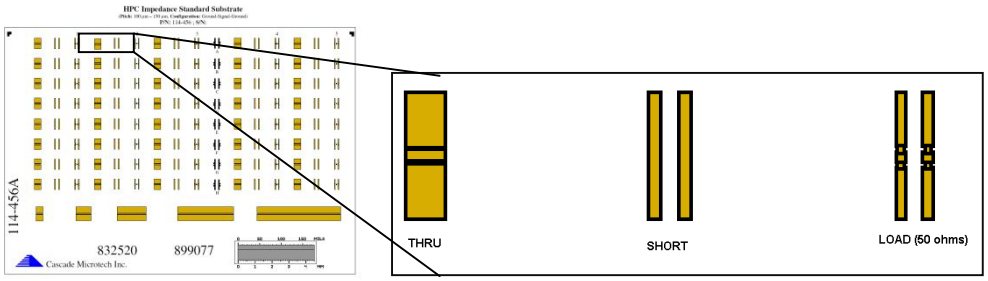


Figure 6.2 Impedance Standard Substrate (ISS) [40]

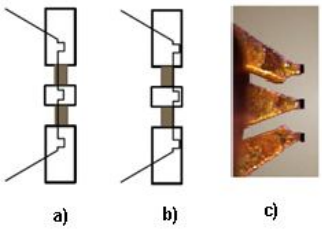


Figure 6.3 Infinity Probe™ Skating [38]

The probes must be aligned both orthogonally and planarized prior to the ISS calibration. Proper orthogonal alignment is achieved by adjusting the chuck rotation so that the probe tip is aligned with a far left and far right thru device. This requires traversing the ISS several times for proper adjustment. Planarization is achieved by

touching the probe down on the transmission lines at the bottom of the ISS and adjusting the probe holder roll so that three divots with equal size and indentation are shown.

The actual software directed ISS calibration only takes approximately 60 seconds to complete. The open measurement is taken with the probes in a raised position >700um from DUT so no coupling occurs between the probes, chuck, or DUT. The user is then directed to place both port-1 and port-2 probes on the ISS. At the end of the calibration, the measured offsets are stored into the VNA in one of the calibration memory locations. A measurement is taken at the end of the calibration sequence that indicates the accuracy of the calibration. Figure 6.4 shows the typical S11 measurement error vs. calibration frequency between the LRRM and SOLT calibration methods [38].

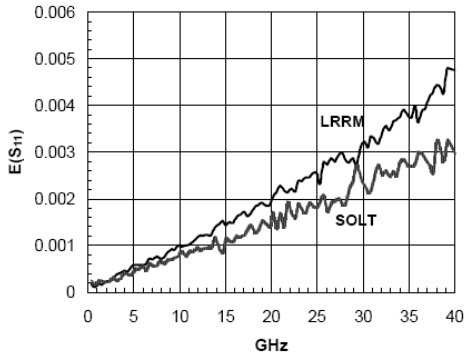


Figure 6.4 Calibration Error by Method [38]

6.3 Parasitic De-Embedding

The ISS provides an ideal calibration standard which is not always reflective of the system on which the devices are being fabricated. In the 3D TWV inductor case, the

CMOS substrate is silicon. MOS capacitances are created between the probe pads and the lossy semiconducting substrate due to the presence of the dielectric. As such, it is more accurate to design on-wafer calibration structures that will facilitate on-wafer de-embedding, and then use the open-short-thru de-embedding (OSTD) method in [39] to account for the substrate parasitics.

By implementing the OSTD method, the reference plane is established at the DUT edge and accounts for all parasitics up to the DUT rather than simply up to the probe tips. Figure 6.5 illustrates the de-embedding structures designed for this experiment in both layout and on-wafer micrograph. The open and short structures are the same for all devices; however, each N-turn inductor has a unique thru structure that requires measurement.

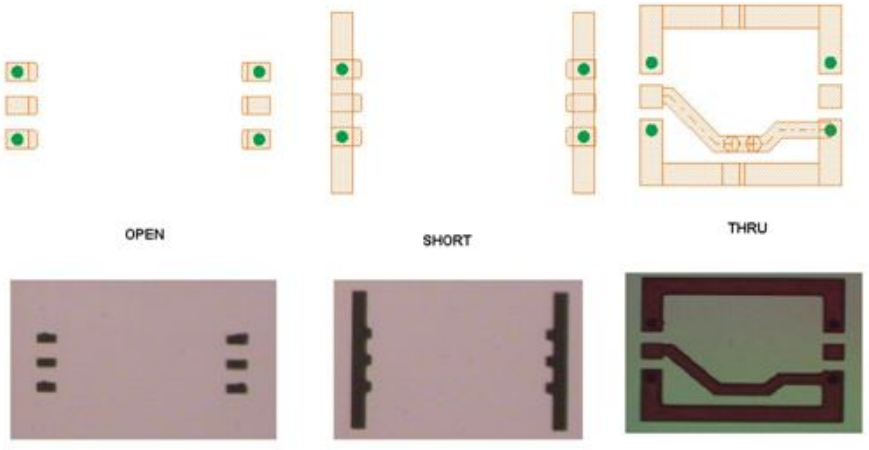


Figure 6.5 On-Wafer De-Embedding Structures

As reported in [39], the open, short, and thru on-wafer measured values will be used to calculate the correct de-embedding factors at each measurement frequency. The HP 8510C VNA has the capability to measure the reflection coefficient (Γ) in 1-port, 2-port, multi-port mode. Gamma can be plotted directly on a Smith Chart in terms of scattering or S-parameters. When taking S-parameter measurements, it is important to understand the subscripting. Here S_{ij} is defined as the incident wave at port "j" and reflected at port "i". There are $N^{\wedge 2}$ port terms present, thus in a 2-port measurement four S-parameters are available; S_{11}, S_{12}, S_{22}, S_{21}.

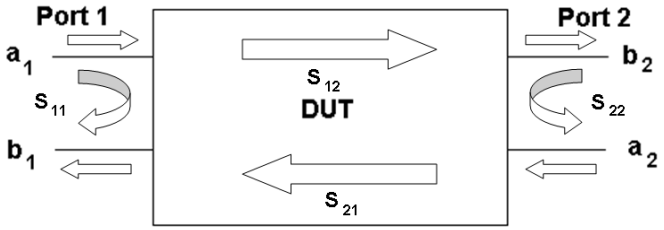


Figure 6.6 2-Port S-Parameter Measurement Block Diagram

Measurements taken on the 3D TWV inductor in Chapters 6 and 7 were taken in 1-port mode. As such, the only pertinent S-parameter is $\Gamma = S_{11}$, which is defined as the ratio of voltages b_1/a_1, as shown in Figure 6.7, where a is incident and b is reflected.

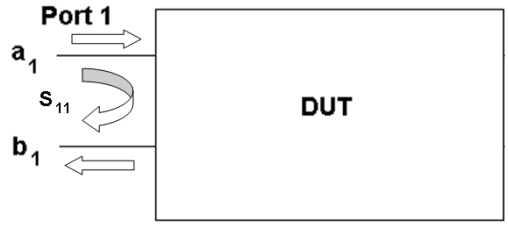


Figure 6.7 1-Port S-Parameter Measurement Block Diagram

Gamma has a one-to-one relationship with both Z and Y as shown in equation (6.1). Here the characteristic impedance of the VNA is $Z_0 = 1/Y_0 \approx 50$ohms. Solving (6.1) for impedance Z_{11}, we obtain Equation (6.2). Z_{11} represents complex impedance and can be transformed into admittance with (6.3) [39].

$$\Gamma = S_{11} = \frac{Z_{11}-Z_0}{Z_{11}+Z_0} = \frac{Y_0-Y_{11}}{Y_0+Y_{11}} \quad (6.1)$$

$$Z_{11} = Z_0 \frac{1+S_{11}}{1-S_{11}} \quad (6.2)$$

$$Y_{11} = 1/Z_{11} \quad (6.3)$$

Open De-Embedding

The open de-embedding structure is used to obtain the parallel admittance Y_P that occurs due to the pad. Figure 6.14a shows the open de-embedding (OPD) block diagram [39]. The admittance is obtained from converting the open impedance measurement $Z_{11,Open}$ to Y_P at each frequency as shown in (6.4). If the user was only concerned about de-embedding the pad then the Y_P value from the open structure can be directly

subtracted from the measured device admittance values Y_{11} at each measured frequency as shown in (6.4) and (6.5) [39]. This requires conversion of the DUT S_{11} to Z_{11} and then to Y_{11} as shown in Equations (6.1) to (6.3) for 1-port measurements. Figure 6.8 shows the raw Q measurement of a 1-turn 3D inductor vs. OPD de-embedding. This method provided a 1.5% improvement in Q_{max} (f_o = 843.6MHz) and 460MHz improvement in F_{sr}.

$$Y_p = 1/(Z_{11,Open}) \quad (6.4)$$

$$Y_{11,OPD} = Y_{11} - Y_p \quad (6.5)$$

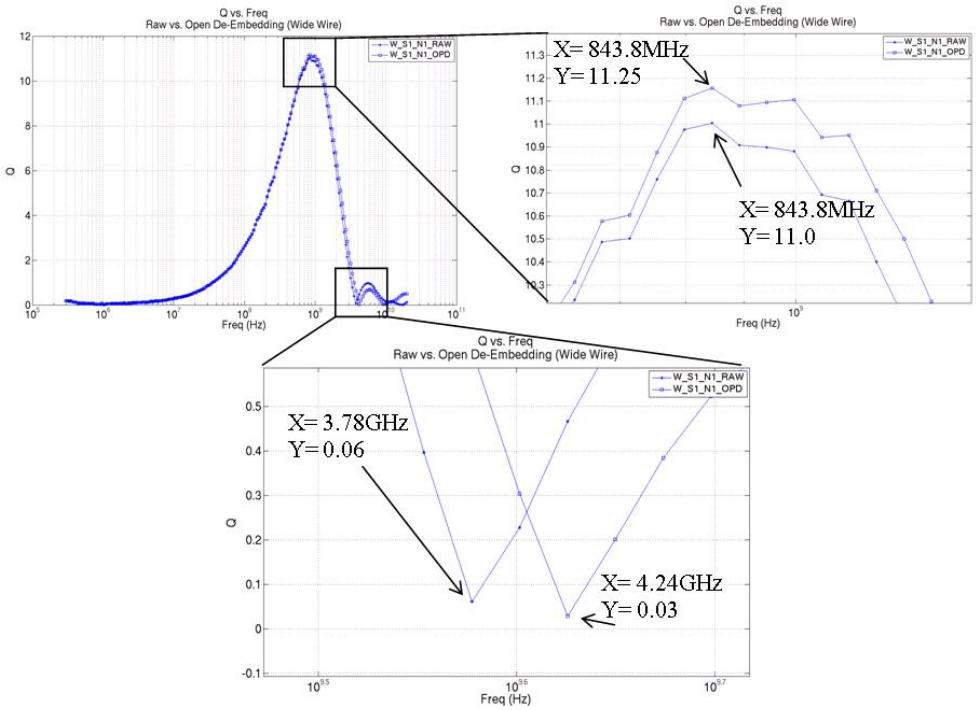


Figure 6.8 OPD De-Embedding vs. Q and F_{sr}

Open, Short De-Embedding

The short de-embedding structure is used to remove the input contact impedance (Z_i) that occurs when switching from calibrating on the gold ISS pads and then measuring a device with copper metallization. Figure 6.14b shows the block diagram for open-short de-embedding (OSD) [39]. As the name implies, the open measurement (6.6) is combined with the short measurement (6.5) to obtain Equation (6.7), which is an equivalent Y_p. As with the open, the parallel admittance can be directly subtracted from the measured device Y_{11} values at each frequency to obtain $Y_{11,OSD}$ in (6.8). If the user only desired de-embedding for the silicon substrate and the contact resistance, then the OSD method would be adequate. Figure 6.9 shows that the contact resistance is very small when compared to the overall DUT resistance and no improvement in |Z| at Q_{max} is observed between the two methods.

$$Z_{11,Short} = Z_{input} \qquad (6.6)$$

$$Y_p = 1/(Z_{11,Open} - Z_{11,Short}) \qquad (6.7)$$

$$Y_{11,OSD} = 1/(Y_{11} - Y_p) \qquad (6.8)$$

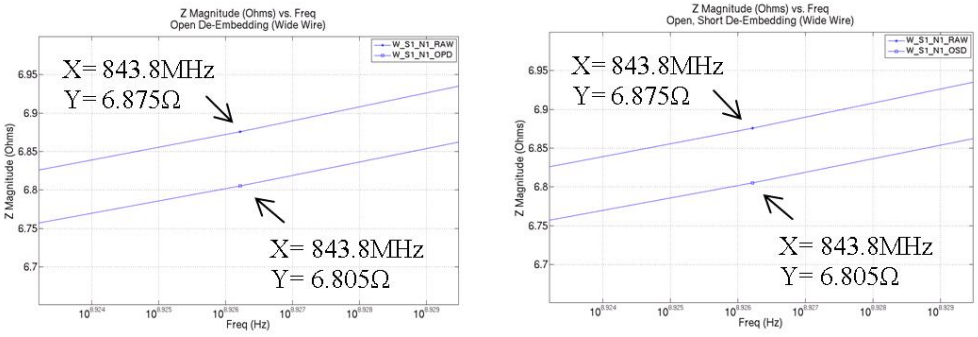


Figure 6.9 |Z| vs. OPD and OSD De-Embedding

Open, Short, Thru De-Embedding

The thru de-embedding structure is used to remove the parasitic element of the transmission line that extends from the pads to the DUT. This again requires the combined value obtained in (6.8) and lead impedance correction factor Z_l. The lead impedance uses a scaling factor (η_{dut}) as determined by the ratio of the DUT to the gap as estimated by Equation (6.9). Here l_{fix} is the effective length between the signal pads and l_{dut} is the effective length of the DUT. Referring to Figure 6.5, the DUT length in the thru structure is only the distance from via-to-via and not the full 3D inductor length. The $Y'_{11,thru}$ value in (6.10) is obtained by first de-embedding the $Y_{11,thru}$ values with the OSD values obtained in (6.8). Z_l can then be calculated as shown in (6.10) [39]. The $Z'_{11,OSTD}$ impedance is obtained by subtracting the Z_l value in (6.10). The block diagram is shown in 6.14c.

$$\eta_{dut} = (l_{fix} - l_{dut})/l_{fix} \qquad (6.9)$$

$$Z_l = \eta_{dut}/(2 * Y'_{11,thru}) \qquad (6.10)$$

$$Z_{11,OSTD} = Z_{11} - Z_l \qquad (6.11)$$

Figure 6.8-6.10 shows a very small improvement in Q_{max} (~.03%) with a corresponding reduction in f_o of 45MHz. A 26% reduction in |Z| occurs (Figure 6.11) while a 300pH reduction in L was observed (Figure 6.12).

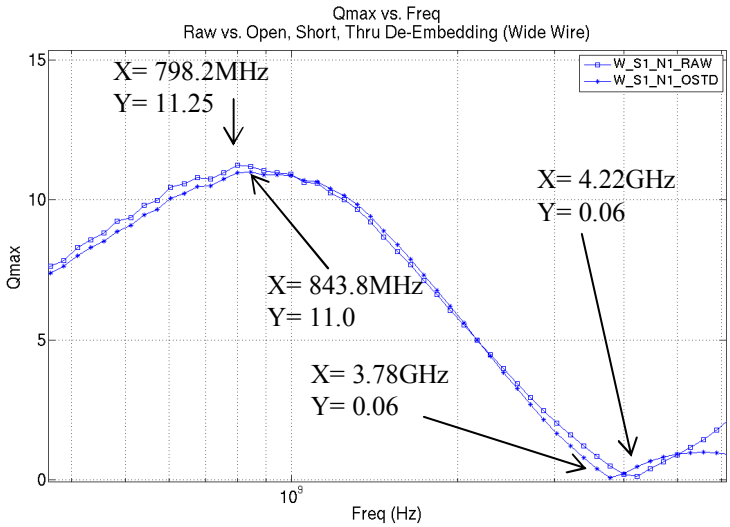


Figure 6.10 OSTD De-Embedding for Q_{max} vs. Frequency

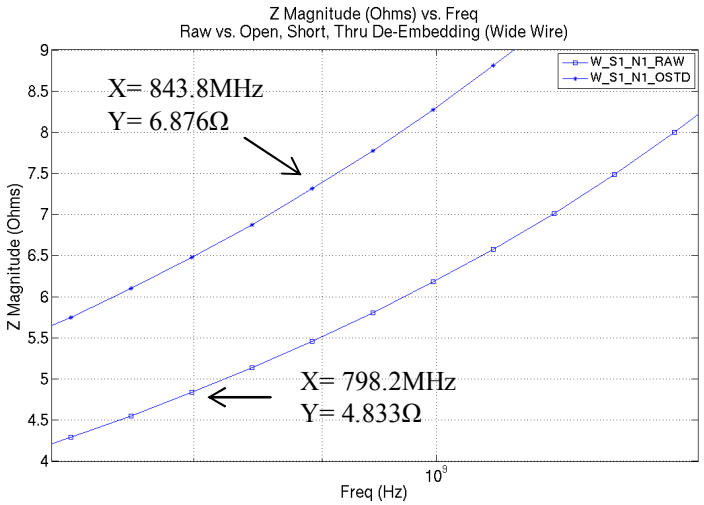


Figure 6.11 OSTD De-Embedding for |Z| vs. Frequency

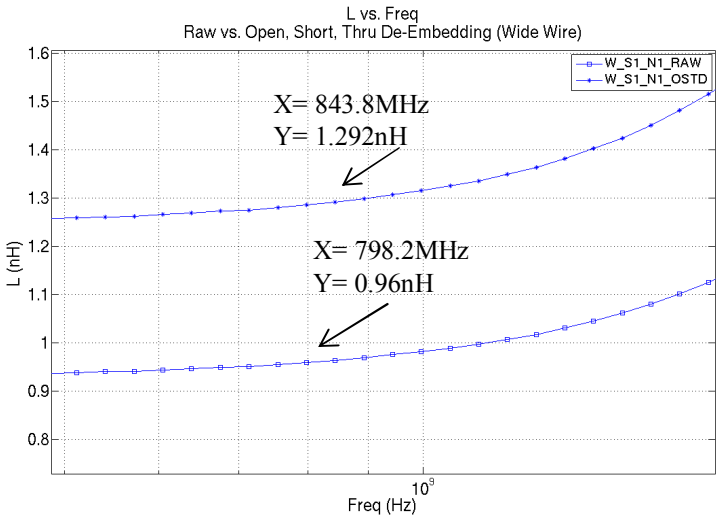


Figure 6.12 OSTD De-Embedding for L vs. Frequency

With implementing OSTD de-embedding, the presence of an odd harmonic with a much higher $Q_{max,2} = 281$at $f_o = 7.78$GHz is seen. Operating a device above the first self-resonant frequency is frowned upon as it unstable.

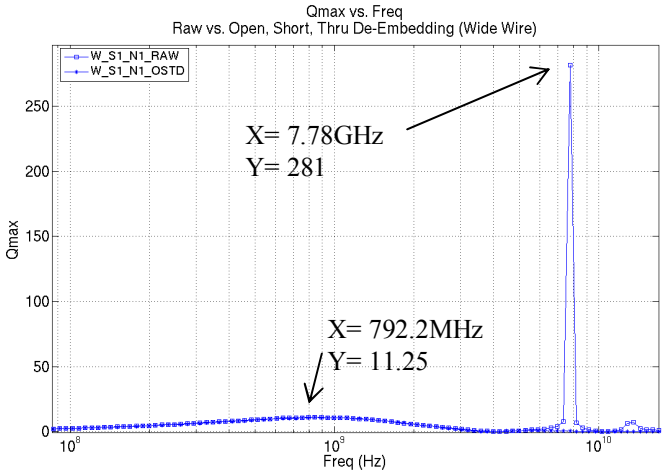


Figure 6.13 OSTD De-Embedding for Q_{max} vs. Frequency

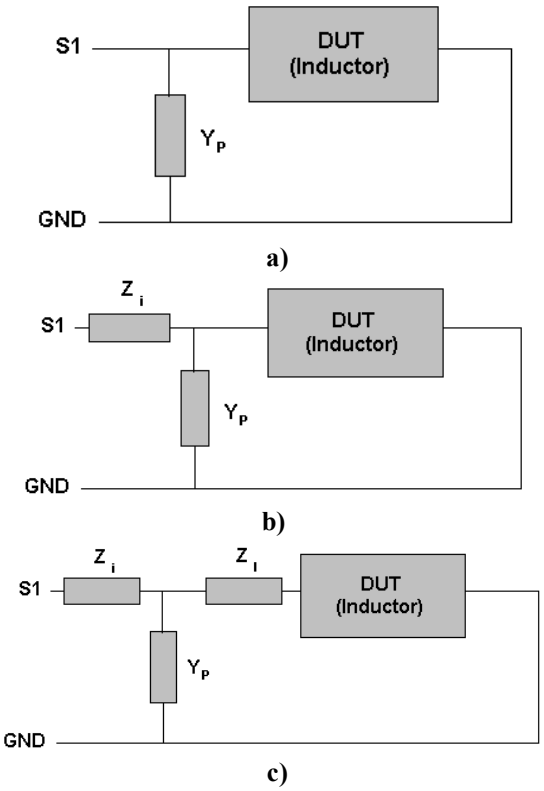


Figure 6.14 De-Embedding Block Diagrams a) Open b) Open-Short, c) Open-Short-Thru [39]

6.4 A 3D Measurement Pitfall

Prior to measuring the 3D TWV inductor presented in this thesis, it became apparent that accurate measurements would not occur if the unpassivated bottom inductor traces came into contact with the grounded chuck. It was thought to use a thin-film barrier on the chuck, but this would have just created a parasitic capacitor between the bottom traces and the chuck. Figure 6.15a shows the situation where the bottom traces would touch the chuck and short to ground.

In order to work around this, an auxiliary chuck was designed and manufactured locally that would adequately raise the 3D wafer ¼" above the chuck. The auxiliary chuck extends the vacuum from the primary chuck as shown in Figure 6.15b. Figure 6.16 shows a micrograph of the manufactured auxiliary chuck backlit with a black light.

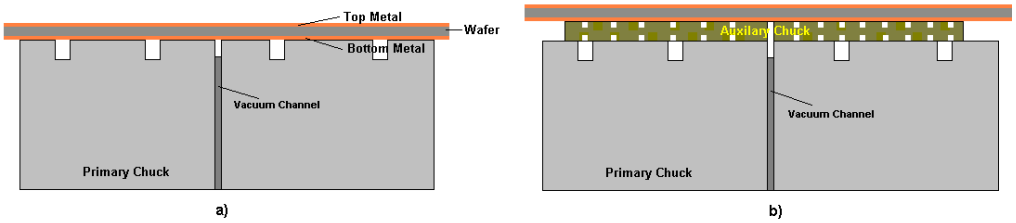


Figure 6.15 Auxiliary Chuck

Figure 6.16 Auxiliary Chuck Micrograph

CHAPTER 7 – 3D TWV INDUCTOR MEASURED PERFORMANCE

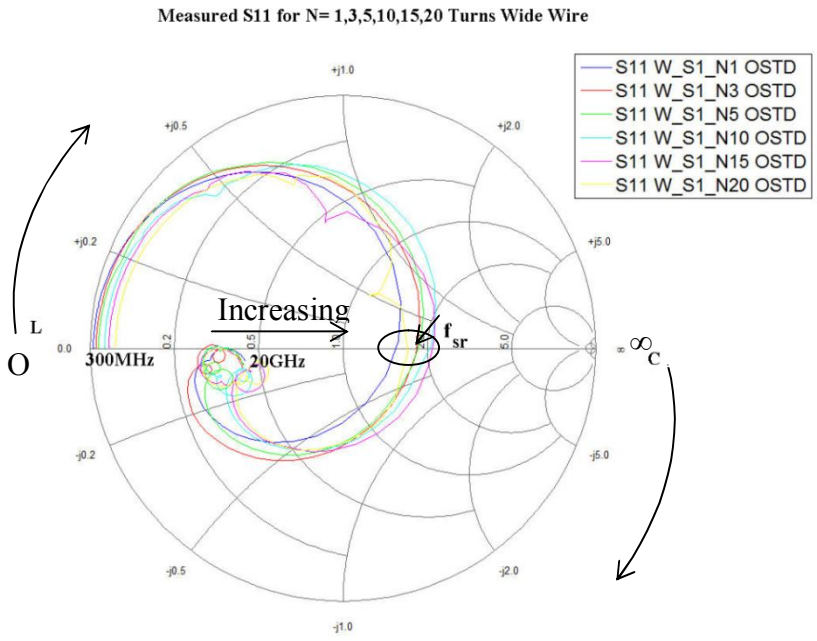
This chapter will discuss the measured performance of the 1-turn 3D TWV inductor in terms of S11 values. The primary parameters Q, f_{sr}, L, and Z will be discussed with comparisons between the wide trace (WT) and narrow trace (NT) width-to-space ratio using the OSTD de-embedding method. Plots in this chapter will only show the 1, 3, 5, and 10 turn inductors for both the WT and NT widths from device site group 1 since they measured the overall highest Q. The 15 and 20 turn WT inductors have also been plotted, but were not available for the NT width devices. Site group 1 provided the best measurements. Appendix B contains plots with all measured devices for reference. In addition to measured performance parameters, specific design parameters that were not available in the first iteration of the device will be covered in general terms to provide the interested reader with some insight into additional parameters that can be adjusted for enhanced device performance.

7.1 Smith Chart Measured S11

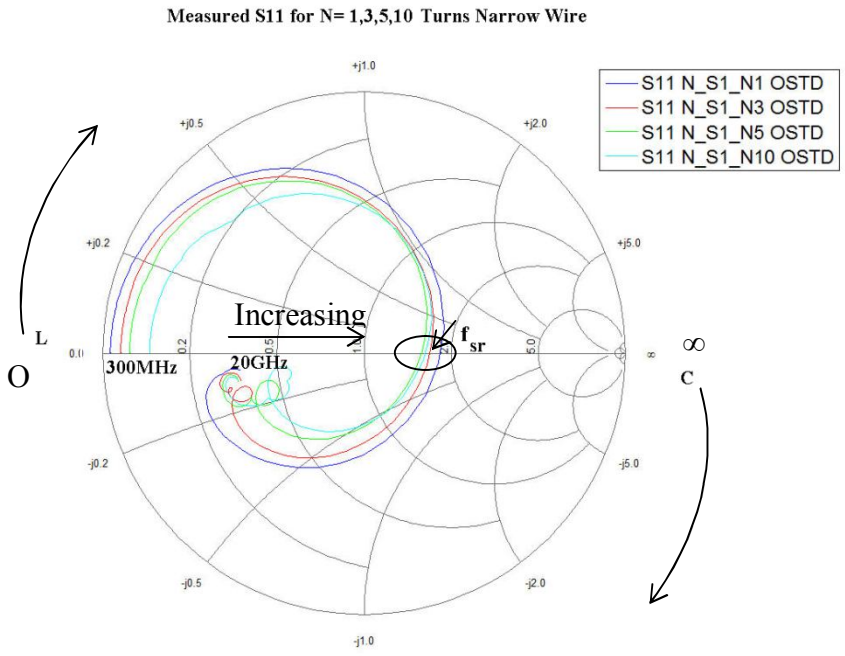
Figure 7.1a-b shows the Smith chart S11 measured values for the WT and NT devices vs. N_{turns}. For a review of Smith charts see APPENDIX B. All S_{11} curves originate in the inductive region close to zero ohms at 300MHz as seen in [47]. Observation shows the DC resistance to be increasing with each additional N_{turns}. NT

devices show an increased series resistance due to the narrow trace width. The self-resonant frequency is found at the transition between the upper and lower-half planes.

Figure 7.2 shows the measured S11 values for the 1-turn WT and NT devices. Comparing these two, it is clear that they are well behaved and vary slightly in resistance and inductance. However, careful observation of Figure 7.1a-b shows that the inductors are not so well behaved with increasing turns. This is evident by the dips or lack of smooth curvature. The higher turn inductors show there to be a parasitic series resistance, inductance, and capacitance term(s) that are forcing multiple resonant frequencies affecting the quality factor as will be shown in the next section.



a)



b)

Figure 7.1 a) Measured S11 for WT b) Measured S11 NT Devices for N=1, 3, 5, 10

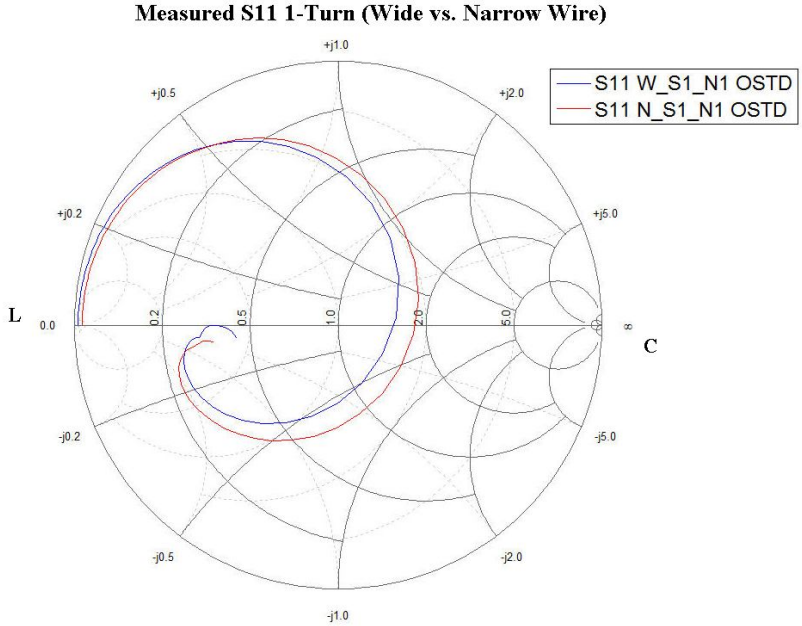
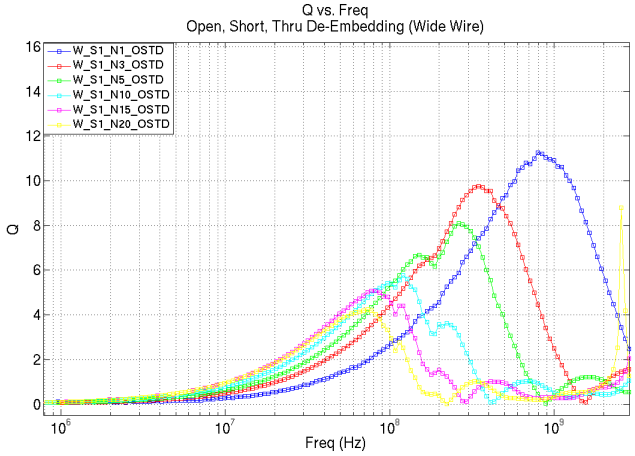


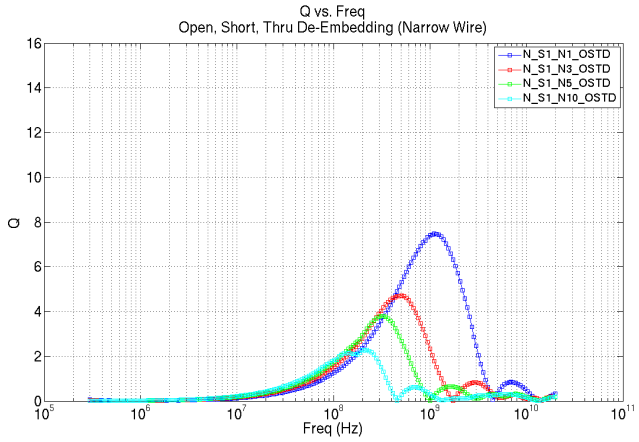
Figure 7.2 Measured S_{11} for WT and NT Devices for N=1

7.2 Measured Quality Factor (Q)

Figure 7.3a-b shows comparison plots of the WT and NT width 3D inductors measured Q performance verse frequency for each N-turn device. General observation indicates the WT inductors provide a higher quality factor (Q_{max}), yet the NT inductors are well behaved and provide a higher characteristic frequency at maximum Q (f_o). The plots also show the presence of a single (WT N=5) and double dip (WT N=15, 20) in Q, which shows dependency on number of turns present. This will be discussed later in this chapter.



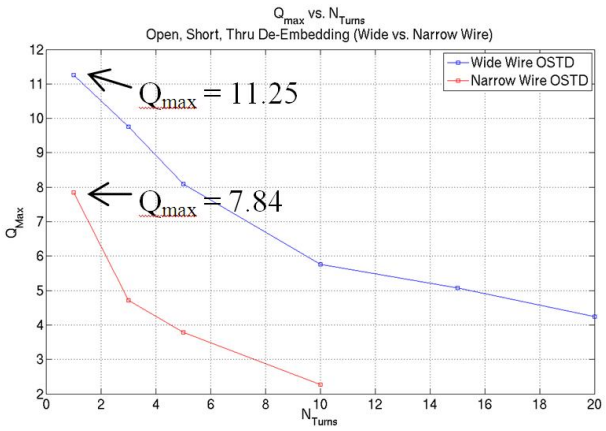
a)



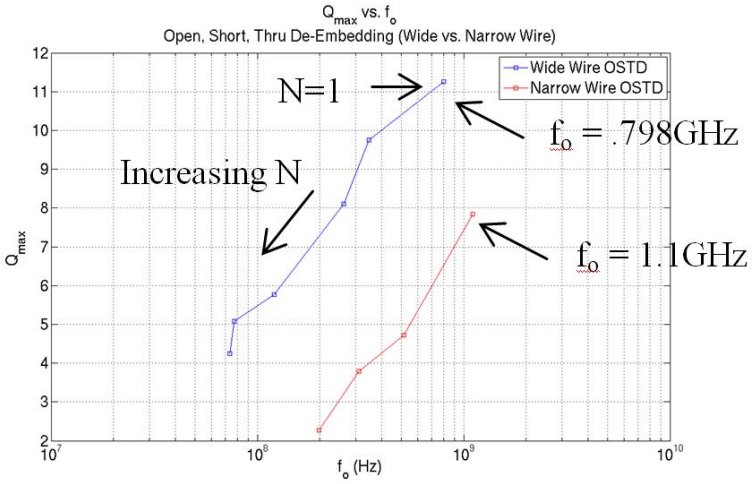
b)

Figure 7.3 a) Q vs. Frequency for WT b) Q vs. Frequency NT Devices (1, 3, 5, 10, {15, 20})

The highest quality factor observed below the self-resonant frequency (f_{sr}) on the measured devices occurred on the 1-turn inductors and measured Q_{max} values of 11.25 (WT) and 7.84 (NT), as shown in Figure 7.4a. Figure 7.3b shows a plot of the measured Q_{max} vs. f_o. The highest operating frequency occurred on the NT 1-turn device and measured 1.1GHz while the same 1-turn wide device measured 798MHz. In both of these cases, Q_{max} and f_o decrease with an increasing number of turns.



a)



b)

Figure 7.4 a) Q_{max} vs. N_{Turns} b) Q_{max} vs. f_o

7.3 Measured Self-Resonant Frequency (f_{SR})

As discussed in Chapter 4, f_{sr} can be obtained from the quality factor plot or the impedance plot for |Z|. Figure 7.5 shows a plot of the 3D inductors f_{sr} vs. N-turns for both the WT and NT devices groups. The highest f_{sr} observed on these devices occurred

on the 1-turn NT width inductor and measured an f_{sr} = 4.58GHz. The WT devices degrade slightly faster in f_{sr} with increasing number of turns than the NT devices.

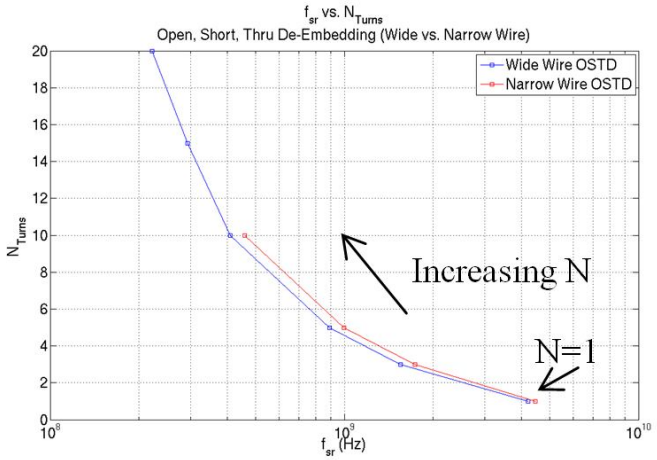
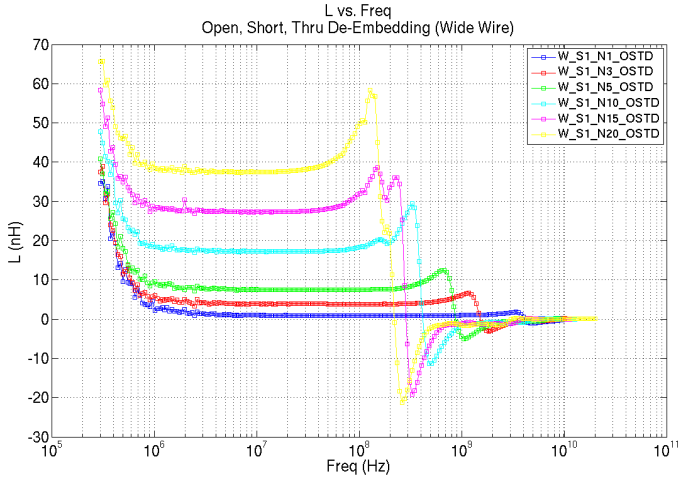


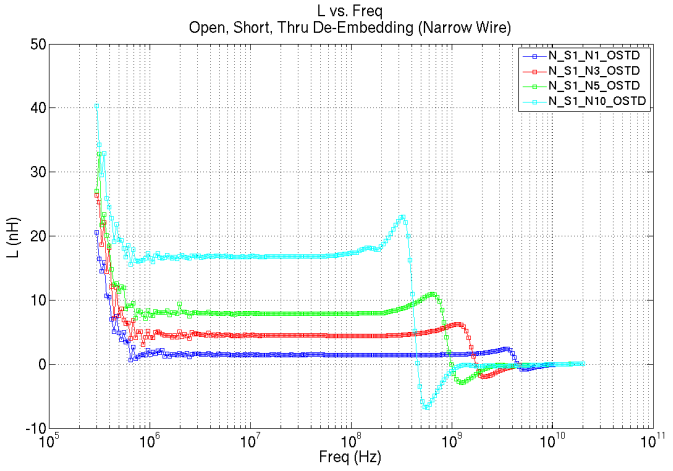
Figure 7.5 $f_{sr\ vs.}$ N_{Turns}

7.4 Measured Inductance (L) vs. Frequency

Figure 7.6a-b shows a comparison between the measured inductance of the WT and NT width devices vs. frequency. The basic 1-turn inductor measures inductance values of 1nH (WT) and 1.5nH (NT). As expected, inductance increases with the number of turns due to the increase in flux linkages.



a)

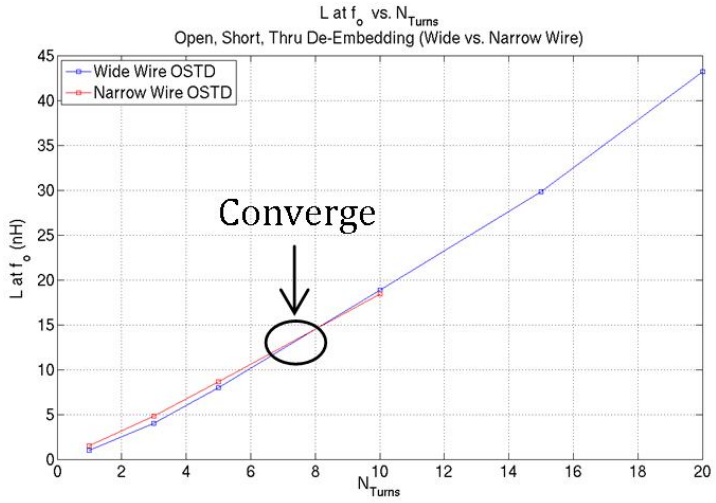


b)

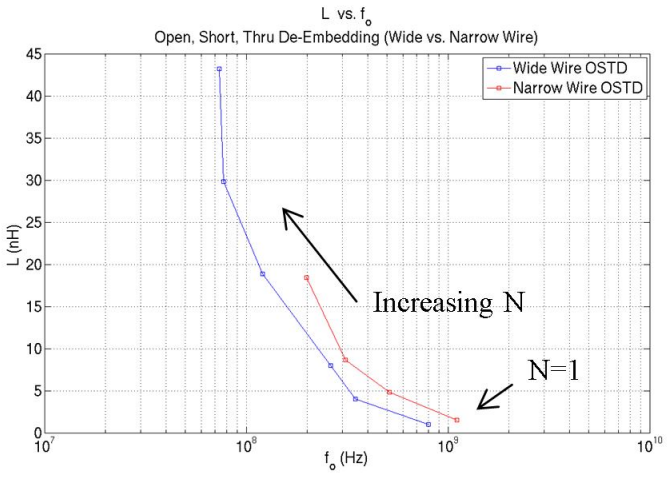
Figure 7.6 a) L vs. Frequency WT b) L vs. Frequency NT

When L at f_o is plotted as a function of number of turns, convergence is observed between the two trace width designs at approximately 8 turns as shown in Figure 7.7a. The NT devices measure higher L at f_o below 8 turns, while the WT devices measure higher L at f_o above 8 turns. The growth with number of turns is showing a growth rate

slightly higher than linear. Figure 7.7b shows that while inductance is increasing with N, f_o is also dropping exponentially.



a)

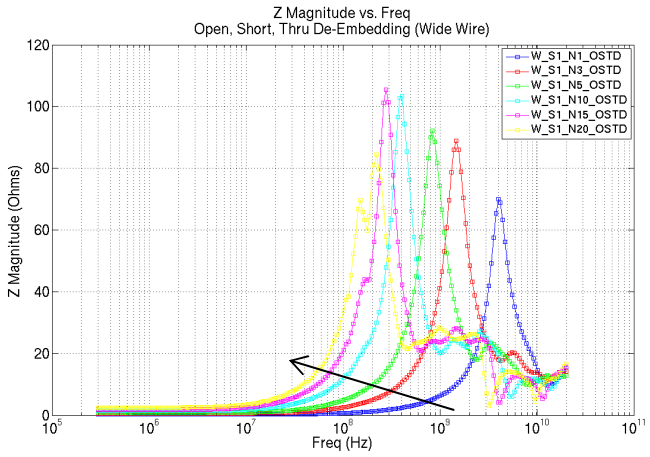


b)

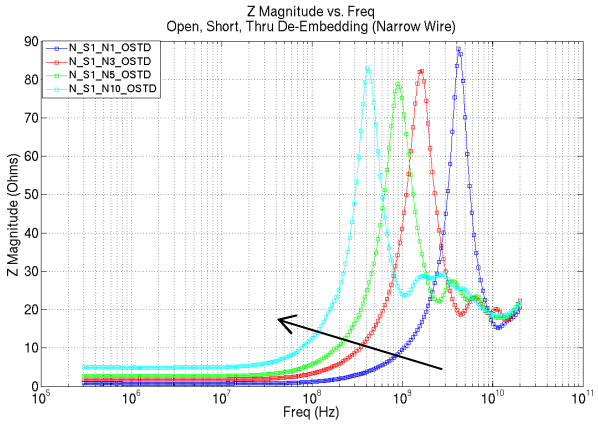
Figure 7.7 a) L at f_o vs. N-Turns b) L vs. f_o

7.5 Measured |Z| vs. Frequency

Figure 7.8a shows a comparison between the measured impedance |Z| of the WT and NT device groups vs. frequency. The WT devices measured lower in resistance than the NT with the 1-turn WT inductor measuring 54% less resistance. This value drops with increasing turns to ~43%. The impedance is measured at along the arrows, which represents the frequency of Q_{max} or f_o.



a)

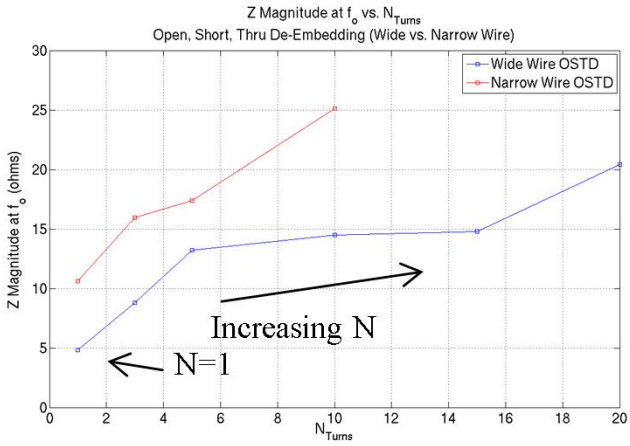


b)

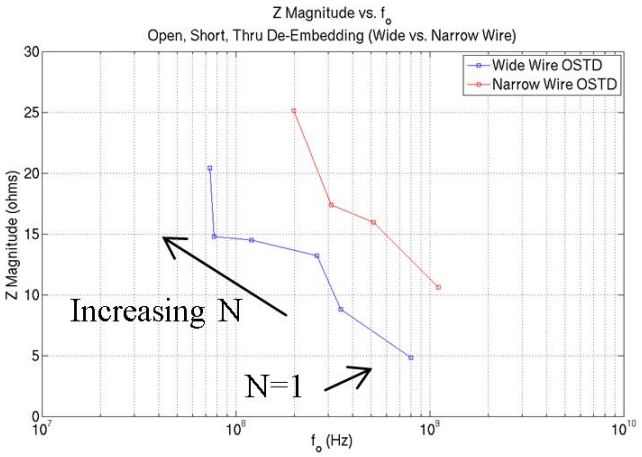
Figure 7.8 a) |Z| vs. Frequency WT b) |Z| vs. Frequency NT

Figure 7.9a illustrates the impedance growth with increasing N values. The WT devices measure a |Z| of 4.8Ω while the 20-turn device measures 20Ω. The resistance is increasing non-linearly with an overall average increase of 2Ω per-turn. Figure 6.b shows the NT device |Z| to be similarly behaved as the WT with the NT 1-turn |Z| measuring 10.62Ω.

Figure 7.9b reaffirms the measurements of f_{sr} above as this replicates the highest impedance observed being at the self-resonant frequency. In both of the plots, it is clear that the wider trace devices offer a lower resistance as there is shown to be a 5.7Ω offset when comparing the 1-turn devices WT to NT devices.



a)



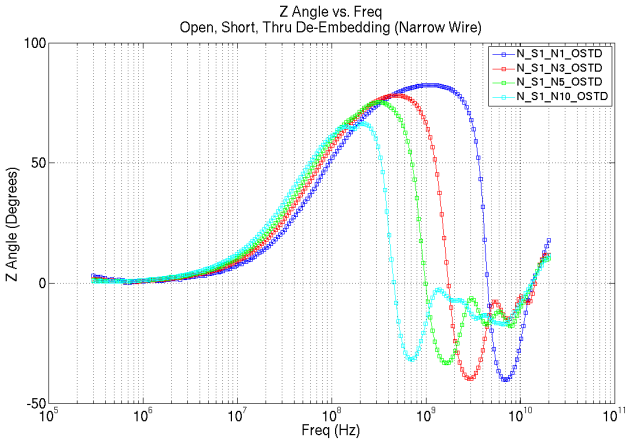
b)

Figure 7.9 a) |Z| vs. N b) |Z| vs. f_o

7.6 Measured Z (Phase Angle) vs. Frequency

Figure 7.10a shows a comparison between the measured Z phase angle of the WT and NT device groups vs. frequency. The phase angle is also measured at f_o. The WT devices measured angles higher in the range of 84º to 76º for N=1 and N=20, respectively. The NT devices measured angles in the range of 82º to 66º for N=1 and N=10, respectively.

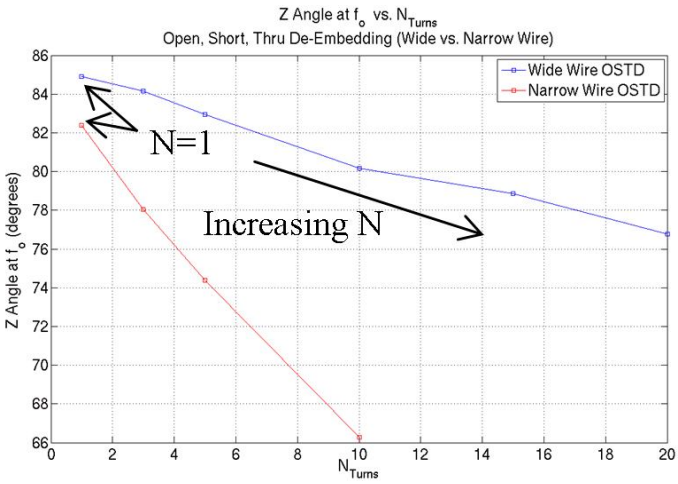
a)



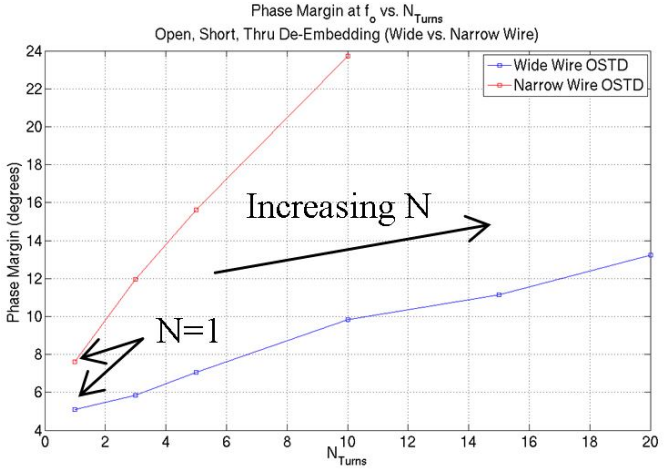
b)

Figure 7.10 a) Z Angle vs. Frequency WT b) Z Angle vs. and Frequency NT

Figure 7.11a shows a comparison between the measured Z phase angle of the WT and NT device groups versus N-turns while b) shows the phase margin for both versus N-turns. The phase angle provides more phase margin with increasing N on both device types.



a)



b)

Figure 7.11 a) Z Angle vs. N_{turns} b) Phase Margin at f_o vs. N_{turns}

7.7 Line Trace Width to Line Space Ratio

The measured values presented thus far have shown the differences in line trace width versus line spacing. While the available silicon did not contain sufficient width-to-

space ratios to optimize the 3D TWV inductor, the following general results can be gleaned.

For a WT line-to-space ratio of .7 and NT of .24, the following has been observed. An increase in inductance below N=8 turns occurs due to more flux linkage lines outside of the narrow trace. However, above N=8, the WT devices increases more rapidly. Thus, inductance can be increased by bringing the lines as close together as possible while trading inter-winding capacitance and lower of the f_{sr}. Also, the same increase in inter-winding capacitance reduces the phase margin.

7.8 Via Height

The via height was designed to be the thickness of the wafer (~ 500um). This via resistance needs to be kept as small as possible to improve the quality factor. The barrel coated vias measure {6m Ω for Solid insert value}Ω. This resistance can be reduced by back grinding the wafer to shorten the length and lower this resistance component. This was not attempted on the first iteration of the device.

7.9 Via Size and Via-to-Via Pitch

The via size used in this design was 50um and provided an aspect ratio of 10:1. The via-to-via pitch is a critical parameter in minimizing the capacitance present in the substrate. This first device iteration did not modulate the via size due to an unrealistic aspect ratio and the equipment available. There was a small inter-winding via pitch

difference (25um) between the WT (100um) and NT (125um) devices. This perhaps may be the reason that the NT devices exhibited a well behaved quality factor compared to the WT devices, which saw capacitive influenced dips. The via pitch in the surface trace direction remained the same between device types.

7.10 Surface Trace Lengths

As just mentioned, the top and bottom surface trace lengths remained the same. This parameter was not adjusted in the first cycle of device development. It is plausible that there might be some advantage in increasing the trace length with a reduction in via height.

7.11 Inductor Radius

The inductor radius can be changed by either adjusting the surface trace length or the via height as previously discussed. This dimension was again not modulated in this experiment.

CHAPTER 8: SUMMARY AND CONCLUSIONS

With the proliferation of radio-frequency (RF) circuits, designers are faced with producing smaller and more efficient modules while maintaining or improving circuit performance, predictability, and robustness. The use of inductors in CMOS circuits is highly desired, yet they have been used sparingly due to their relatively large footprint, low inductance values, and low quality factor. In order to address these issues, a 3D inductor manufactured using TWV technology was developed.

With the purpose of providing a 3D TWV inductor design guide, the architecture and physical model of a planar device were first reviewed to provide a foundation of the understood planer parasitic effects caused by the inductors electric and magnetic fields. The 3D inductor architecture addresses some of these issues.

The 3D TWV inductor architecture provides a 40% smaller device footprint with a maximum 1-turn wide trace device de-embedded Q of 11.25 and f_{sr} = 4.4GHz. While the 1-turn WT device measures ~1nH and increases non-linearly to ~45nH up to 20-turns, convergence between WT and NT devices occurs at N=8 turns. The NT device provides higher inductance below N=8 and the WT device above 8. The 1-turn WT device series resistance measured 4.8Ω and increases to 20Ω for N=20. However, each additional turn added drops off to 1.02Ω per turn above N=15.

As with the planar inductor, the 3D TWV inductor suffers similarly from capacitive coupling to the substrate. As such, future work on this architecture should be focused on optimization of the via height (wafer thickness), the via pitch, the inductor radius, and the line-to-width space ratio. The architecture would also benefit from devising a scheme to either remove or replace the silicon substrate within the core of the inductor.

The 3D inductor architecture presented in this thesis successfully confirms the plausibility of using the thru-wafer via in realization of a smaller inductor by utilizing both wafer surfaces within a CMOS process. While further development and optimization is required, the framework has now been established to do this.

BIBLIOGRAPHY

[1] Chang, C.A.; Sung-Pi Tseng; Jun Yi Chuang; Shiue-Shr Jiang; Yeh, J.A.;, "Characterization of spiral inductors with patterned floating structures," Microwave Theory and Techniques, IEEE Transactions on , vol.52, no.5, pp. 1375- 1381, May 2004

[2] Chia-Hsin Wu; Chih-Chun Tang; Chi-Kun Chiu; Shen-Iuan Liu;, "Analysis and application of miniature 3D inductor," *Circuits and Systems, 2002. ISCAS 2002. IEEE International Symposium on* , vol.2, no., pp. II-811- II-814 vol.2, 2002

[3] Jinghong Chen; Jun Zou; Chang Liu; Schutt-Aine, J.E.; Kang, S.-M.K.;, "Design and modeling of a micromachined high-Q tunable capacitor with large tuning range and a vertical planar spiral inductor," *Electron Devices, IEEE Transactions on* , vol.50, no.3, pp. 730- 739, March 2003

[4] Xiaomeng Shi; Jian-Guo Ma; Kiat Seng Yeo; Manh Anh Do; Erping Li;, "Equivalent circuit model of on-wafer CMOS interconnects for RFICs," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on* , vol.13, no.9, pp. 1060- 1071, Sept. 2005

[5] Langford-Smith, F. Radio Designer's Handbook. Newnes, 1997.

[6] Crawford, A.M.; Gardner, D.; Wang, S.X.;, "High-frequency microinductors with amorphous magnetic ground planes," *Magnetics, IEEE Transactions on* , vol.38, no.5, pp. 3168- 3170, Sep 2002

[7] F. W. Grover, Inductance Calculations, Van Nostrand, Princeton, N.J., 1946; reprinted bv Dover Publications. New York, N.Y., 1962.

[8] Yun-Seok Choi; Jun-Bo Yoon;, "Experimental analysis of the effect of metal thickness on the quality factor in integrated spiral inductors for RF ICs," *Electron Device Letters, IEEE* , vol.25, no.2, pp. 76- 79, Feb. 2004

[9] "Skin-Depth." http://www.microwaves101.com/encyclopedia/calsdepth.cfm. Microwaves101.com, n.p. n.d. Web. 3 Feb. 2011.

[10] Han-Su Kim; Kyuchul Chong; Ya-Hong Xie; Devincentis, M.; Itoh, T.; Becker, A.J.; Jenkins, K.A.;, "A porous Si based novel isolation technology for mixed-signal integrated circuits," VLSI Technology, 2002. Digest of Technical Papers. 2002 Symposium on , vol., no., pp. 160- 161, 2002

[11] Greenhouse, H.;, "Design of Planar Rectangular Microelectronic Inductors," *Parts, Hybrids, and Packaging, IEEE Transactions on* , vol.10, no.2, pp. 101-109, Jun 1974

[12] Burghartz, J.N.; Edelstein, D.C.; Soyuer, M.; Ainspan, H.A.; Jenkins, K.A.;, "RF circuit design aspects of spiral inductors on silicon," *Solid-State Circuits, IEEE Journal of* , vol.33, no.12, pp.2028-2034, Dec 1998

[13] Ermolov, V.; Lindstrom, T.; Nieminen, H.; Olsson, M.; Read, M.; Ryhanen, T.; Silanto, S.; Uhrberg, S.;, "Microreplicated RF toroidal inductor," *Microwave Theory and Techniques, IEEE Transactions on* , vol.52, no.1, pp. 29- 37, Jan. 2004

[14] Chungpin Liao; Tzuen-Hsi Huang; Chwan-Ying Lee; Denny Tang; Shan-Ming Lan; Tsun-Neng Yang; Li-Fu Lin;, "Method of creating local semi-insulating regions on silicon wafers for device isolation and realization of high-Q inductors," *Electron Device Letters, IEEE* , vol.19, no.12, pp.461-462, Dec 1998

[15] Cheon Soo Kim; Piljae Park; Joung-Woo Park; Nam Hwang; Hyu Kyu Yu;, "Deep trench guard technology to suppress coupling between inductors in silicon RF ICs," *Microwave Symposium Digest, 2001 IEEE MTT-S International* , vol.3, no., pp.1873-1876 vol.3, 2001

[16] Yue, C.P.; Wong, S.S.;, "Physical modeling of spiral inductors on silicon," *Electron Devices, IEEE Transactions on* , vol.47, no.3, pp.560-568, Mar 2000

[17] Radulescu, M., "Application Note USM 1-5." US Microwaves. n.d. http://www.usmicrowaves.com/appnotes/usm_an_105_spiral_chip_inductors_design_modeling_overview.htm>, Web. 4 Mar. 2008.

[18] Yue, C.P.; Wong, S.S.;, "On-chip spiral inductors with patterned ground shields for Si-based RF ICs," *Solid-State Circuits, IEEE Journal of* , vol.33, no.5, pp.743-752, May 1998

[19] Kamogawa, K.; Nishikawa, K.; Toyoda, I.; Tokumitsu, T.; Tanaka, M.;, "A novel high-Q and wide-frequency-range inductor using Si 3-D MMIC technology," *Microwave and Guided Wave Letters, IEEE* , vol.9, no.1, pp.16-18, Jan 1999

[20] Aguilera, J., et al. "A guide for on-chip inductor design in a conventional CMOS process for RF applications," *Applied Microwave & Wireless, 2001.* pp.56-65, Oct. 2001

[21] N. Talwalkar, "Integrated CMOS Transmit-Receive Switch Using On-Chip Spiral Inductors," Ph.D Dissertation, Stanford University, Stanford, December 2003

[22] Burghartz, J.N.;, "Progress in RF inductors on silicon-understanding substrate losses ," *Electron Devices Meeting, 1998. IEDM '98 Technical Digest., International*, vol., no., pp.523-526, 6-9 Dec 1998

[23] Lee, T. H., The Design of CMOS radio-Frequency Integratd Circuits. 2nd Edition. Cambridge, United Kingdom: Cambridge University Press, 2004.

[24] Krafcsik, D.M.; Dawson, D.E.;, "A Closed-Form Expression for Representing the Distributed Nature of the Spiral Inductor," *Microwave and Millimeter-Wave Monolithic Circuits* , vol.86, no.1, pp. 87- 92, Jun 1986

[25] Yue, C.P.; Ryu, C.; Lau, J.; Lee, T.H.; Wong, S.S.;, "A physical model for planar spiral inductors on silicon," *Electron Devices Meeting, 1996. IEDM '96., International* , vol., no., pp.155-158, 8-11 Dec. 1996

[26] Hasegawa, H.; Furukawa, M.; Yanai, H.;, "Properties of Microstrip Line on Si-SiO_2 System," *Microwave Theory and Techniques, IEEE Transactions on* , vol.19, no.11, pp. 869- 881, Nov 1971

[27] Abdolahad, M., et al., "Fluorine-free high-resolution selective plasma etching of silicon-oxide layers on silicon substrates 2010." Journal Phys. D: Appl. Phys. 43

[28] Moll, Amy, Ph.D. "Advanced Processing Techniques for Fabrication of 3D Microstructures for Future Electronic Devices: Characterization, Reliability and Applications of 3-Dimensional Microstructures." *Quarterly Status Report No. 4* 3/22/2004 – 6/21/2004: Pages 18-22. Report.

[29] Johnson, V.N.; Jozwiak, J.; Moll, A.;, "Through wafer interconnects on active pMOS devices," *Microelectronics and Electron Devices, 2004 IEEE Workshop on*, vol., no., pp. 82- 84, 2004

[30] Moll, Amy, Ph.D. “Advanced Processing Techniques for Fabrication of 3D Microstructures.” *Quarterly Status Report No. 9*, 9/22/2003 – 12/21/2003: Pages 2-7. Report.

[31] Cacouris, Ted. “Preventing cross-contamination caused by copper diffusion and other sources.” Micro-Magazine.com., n.p. n.d. 20 Oct 2008.

[32] "Parylene Properties and Characteristics." http://www.vp-scientific.com/parylene_properties.htm. V&P Scientific, Inc., n.p. n.d. Web. 11 Feb. 2011.

[33] "Parylene-Coating." http://www.x-magnet.net/Machinery/parylene-coating.html. Yuxiang Magnetic Materials, n.p. n.d. Web. 11 Feb. 2011.

[34] Moll, Amy, Ph.D. “Advanced Processing Techniques for Fabrication of 3D Microstructures.” *Quarterly Status Report No. 12*, 6/22/2004 – 9/21/2004: Pages 3-4. Report.

[35] TOMPA, G. A., et al,; “MOCVD Process Model for Deposition of Complex Oxide Ferroelectric Thin Films”, Structured Materials Industries Inc. May 2001. June 2010.

[36] Moll, Amy, Ph.D. “Advanced Processing Techniques for Fabrication of 3D Microstructures.” *Quarterly Status Report No. 11*, 3/22/2004 – 6/21/2004: Pages 3-5. Report.

[37] "Infinity Probe." Cascademicrotech.com. Cascade Microtech, n.p. n.d. Web. 10 Feb. 2011.

[38] Cascade; "A Guide to Better Vector Network Analyzer Calibrations for Probe Tip Measurements", Cascade Microtech Application Note, TECHBRIEF 4-0694, 1994

[39] Kolding, T.E.;, "On-wafer calibration techniques for giga-hertz CMOS measurements ," *Microelectronic Test Structures, 1999. ICMTS 1999. Proceedings of the 1999 International Conference on* , vol., no., pp.105-110, 1999

[40] "ISS 114-456A," http://www.cmicro.com/file/iss-map-114-456. Cascade Microtech, n.p. 29 June 2009 Web. 10 Feb. 2011.

[41] Piernas, B.; Nishikawa, K.; Kamogawa, K.; Nakagawa, T.; Araki, K.;, "Improved three-dimensional GaAs inductors," *Microwave Symposium Digest, 2001 IEEE MTT-S International* , vol.1, no., pp.189-192 vol.1, 2001

[42] Ludwig, R., Bretchko, P. RF Circuit Design, Theory and Applications. Upper Saddle River, NJ: Prentice Hall, 2000.

[43] Pinel, S.; Cros, F.; Nuttinck, S.; Yoon, S.-W.; Allen, M.G.; Laskar, J.;, "Very high-Q inductors using RF-MEMS technology for System-On-Package wireless communication integrated module," *Microwave Symposium Digest, 2003 IEEE MTT-S International* , vol.3, no., pp. 1497- 1500 vol.3, 8-13 June 2003

[44] Kodali, S.; Allstot, D.J.;, "A symmetric miniature 3D inductor," *Circuits and Systems, 2003. ISCAS '03. Proceedings of the 2003 International Symposium on* , vol.1, no., pp. I-89- I-92 vol.1, 25-28 May 2003

[45] "Inductance." The Physics Hypertextbook. 10 June 2009. <http://physics.info/inductance/>.

[46] "Smith Chart Tutorial Part1" RF, RFIC & Microwave Theory, Design. 10 June 2009. <www.odyseus.nildram.co.uk/RFMicrowave>.

[47] Sischka, F., "3.3.3.1. BASICS OF S-PARAMETERS, part 1." n.d. <http://www.google.com/url?q=http://www.ece.unh.edu/courses/ece711/refrense_material/s_parameters/1SparBasics_1.pdf>.

[48] Gun Ho Chang et al., "Via/hold Filling by Pulse-Reverse Copper Electroplating for 3D SiP." 2006, Materials Science Forum, 510-511, 942.

[49] "IC-CAP Modeling of Sprial Inductors." Agilent Technologies. 15 Jan 2011. <http://edocs.soco.agilent.com/display/iccapmhb/Inductor>.

[50] Prashant, M.; Kai Liu; Seung Wook Yoon; , "Integrated passive devices (IPD) integration with eWLB (embedded wafer level BGA) for high performance RF applications," *Electronics Packaging Technology Conference (EPTC), 2010 12th* , vol., no., pp.677-680, 8-10 Dec. 2010

[51] "Magnetic properties of materials." University of Surrey, Guildford. Clarke, R., 2 Aug 2008. <http://info.ee.surrey.ac.uk/Workshop/advice/coils/mu/index.html#mu>.

[52] "Magnetic units of measurement." All About Circuits, Vol I-DC. n.a. n.d. <http://www.allaboutcircuits.com/vol_1/chpt_14/3.html>

[53] Wentworth, S.M., Fundamentals of Electromangetics with Engineering Applications. Hoboken, NJ: John Wiley and Sons, Inc., 2005.

APPENDIX A

Smith Chart Tutorial

Smith Chart Tutorial

The following information has been drawn from [46] and [47], which is reiterated here with permission for the benefit of the reader to provide a brief background of Smith charts and their utility.

The Smith Chart finds its origin in the voltage standing wave ratio (VSWR) when a load Z_l is excited by an incident AC voltage and the reflection is measured. Thus, VSWR is the ratio of the value of the reflected AC voltage over the incident AC voltage as shown in Equation (A.1) and Figure A.1 where V1 is the incident wave and V2 is the reflected wave. The reflected value contains phase information and is referred to as the reflection coefficient (Γ). Thus Γ is a complex value as shown in (A.2).

$$\text{VSWR} = \frac{V_{max}}{V_{min}} = \frac{|V_1|+|V_2|}{|V_1|-|V_2|} \qquad (A.1)$$

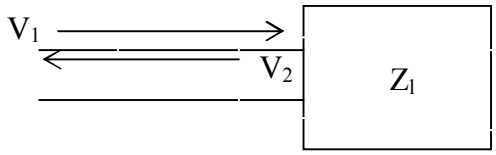


Figure A.1 Incident and Reflected Wave from a Load

$$\Gamma = \frac{V_2}{V_1} \cdot e^{-j2\beta l} \quad (A.2)$$

A rectilinear impedance graph is shown in Figure A.2. Here it is clear that the x-axis is real and spans from 0 impedance to ∞, while the y-axis is imaginary and spans ∓j.

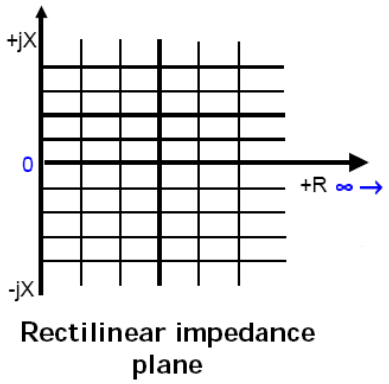


Figure A.2 Rectilinear Impedance Plane (Complex)

The Smith Chart is a transformation of the rectilinear impedance graph into the polar plane by stretching the two complex half-planes to connection as shown in Figure A.3

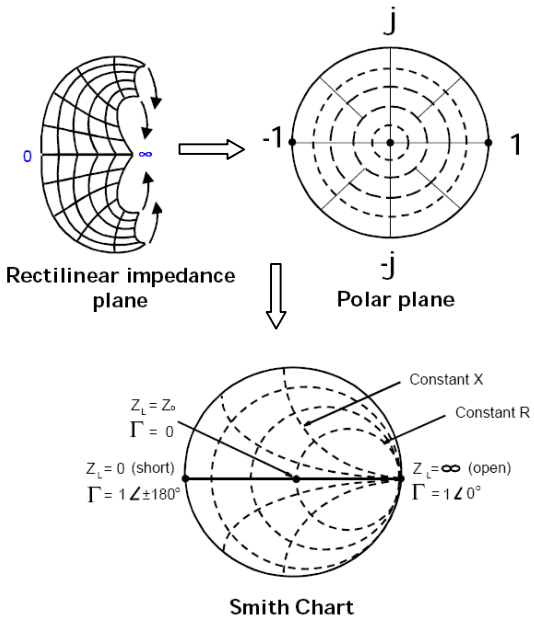


Figure A.3 Smith Chart from Rectilinear Plane and Polar Plane

From basic circuits, inductors carry a phase relationship that is +j or $\mp 180°$ while capacitors are –j or 0°. Thus, the top half of the Smith Chart represents inductive values and the bottom half capacitive values. Remembering that inductance at DC is a short or 0Ω and a capacitor is ∞Ω forces an inductor to start on the left side of the horizontal plane and a capacitor to start on the right side of the same. With increasing frequency, both move in a clockwise fashion.

Test equipment must be terminated in order to make measurements. As such, they are typically terminated at 50 or 75Ω. This value is referred to as the characteristic impedance Z_o. Measured complex impedance is considered normalized as shown in Equation A.3.

$$\frac{Z_{in}}{Z_o} = \frac{1+\Gamma}{1-\Gamma} \qquad (A.3)$$

Depending on how many N ports are being utilized will determine the number of scattering parameters (S-parameters) that result as shown in Equation (A.4)

$$N^2 \qquad (A.4)$$

S-parameters are put typically in terms of 1 or 2 ports and are annotated as S_{ij} where i and j represent the port. Figure A.4 shows the relationship between the 2-port parameters while Figure A.5 shows the block diagram. The following are the 2-port coefficient definitions:

- $|S_{11}|^2$ power reflected from port1
- $|S_{12}|^2$ power transmitted from port1 to port2
- $|S_{21}|^2$ power transmitted from port2 to port1
- $|S_{22}|^2$ power reflected from port2

$$\underline{S}_{11} = \frac{\underline{b}_1}{\underline{a}_1} = \frac{V_{\text{reflected at port1}}}{V_{\text{towards port1}}}\Bigg|_{\underline{a}_2 = 0} \qquad \underline{S}_{12} = \frac{\underline{b}_1}{\underline{a}_2} = \frac{V_{\text{out of port1}}}{V_{\text{towards port2}}}\Bigg|_{\underline{a}_1 = 0}$$

$$\underline{S}_{21} = \frac{\underline{b}_2}{\underline{a}_1} = \frac{V_{\text{out of port2}}}{V_{\text{towards port1}}}\Bigg|_{\underline{a}_2 = 0} \qquad \underline{S}_{22} = \frac{\underline{b}_2}{\underline{a}_2} = \frac{V_{\text{reflected at port2}}}{V_{\text{towards port2}}}\Bigg|_{\underline{a}_1 = 0}$$

Figure A.4 2-Port Scattering Parameter Coefficients [43]

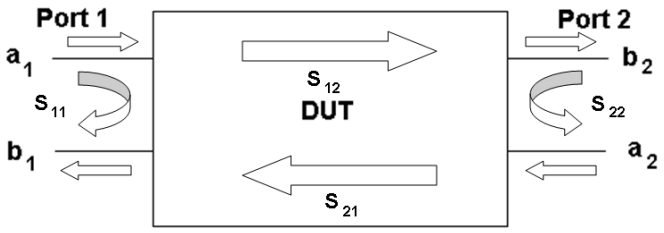


Figure A.5 2-Port Measurement

In the case of 1-port measurements, port 2 is not present and the coefficients are reduced to the block diagram shown in Figure A.6. As such, $S_{11} = Z_{11}$

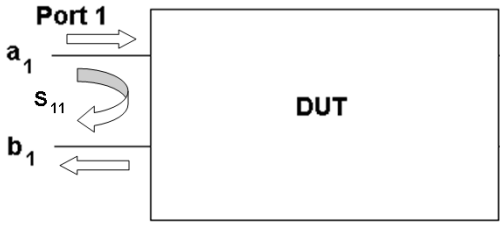


Figure A.6 1-Port Measurement

APPENDIX B

Measured and Calculated Data - All Devices

Measured Data – All Devices

Wide Trace Site Group 1

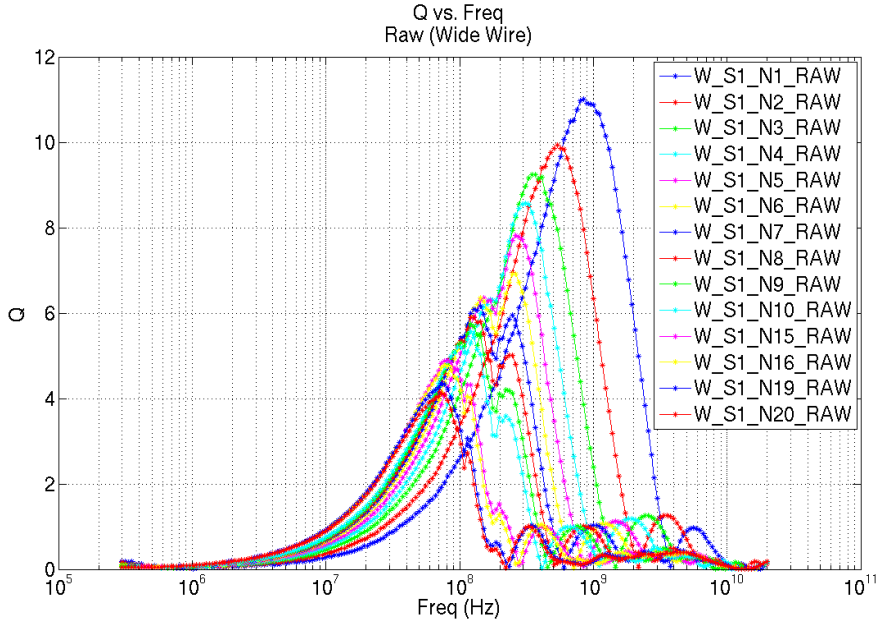


Figure B.1 Q vs. Frequency Wide Trace Site Group 1

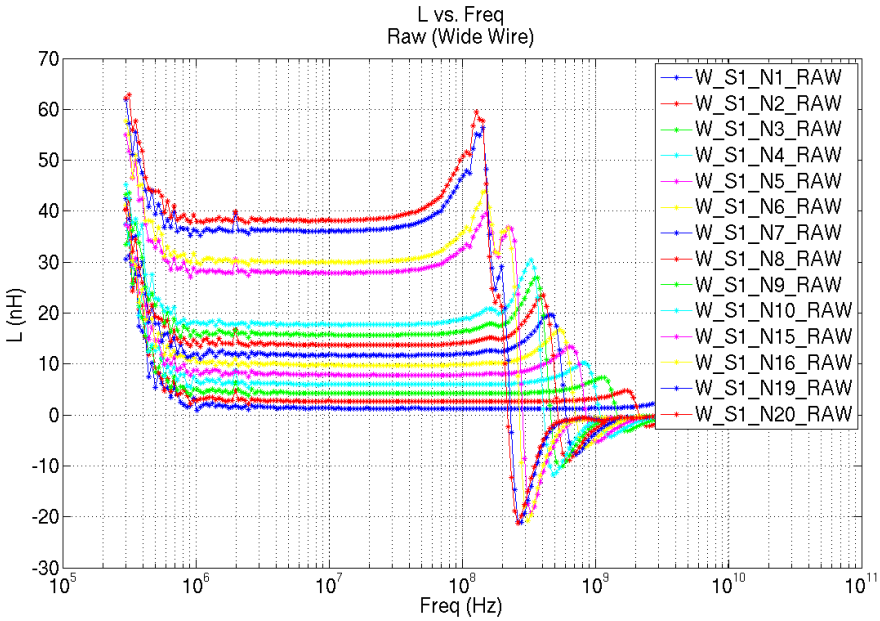


Figure B.2 L vs. Frequency Wide Trace Site Group 1

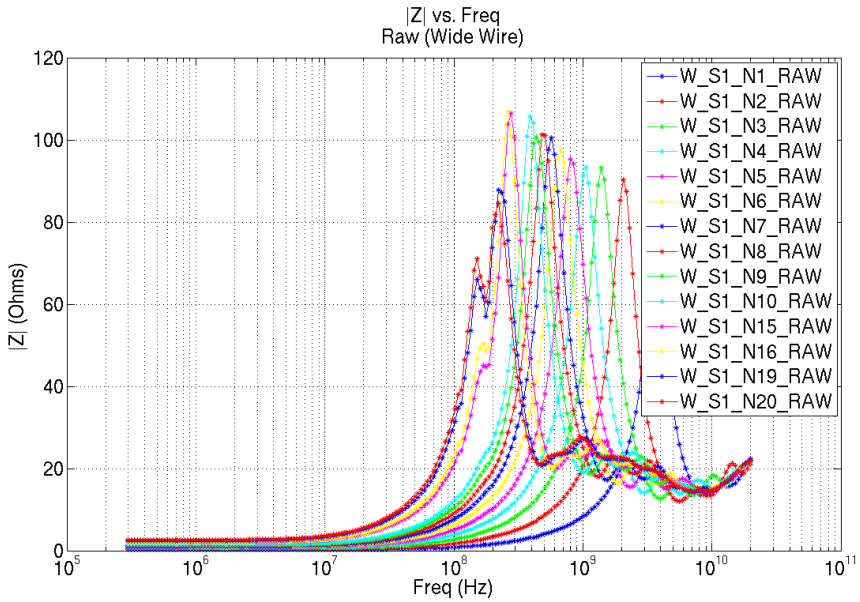


Figure B.3 |Z| vs. Frequency Wide Trace Site Group 1

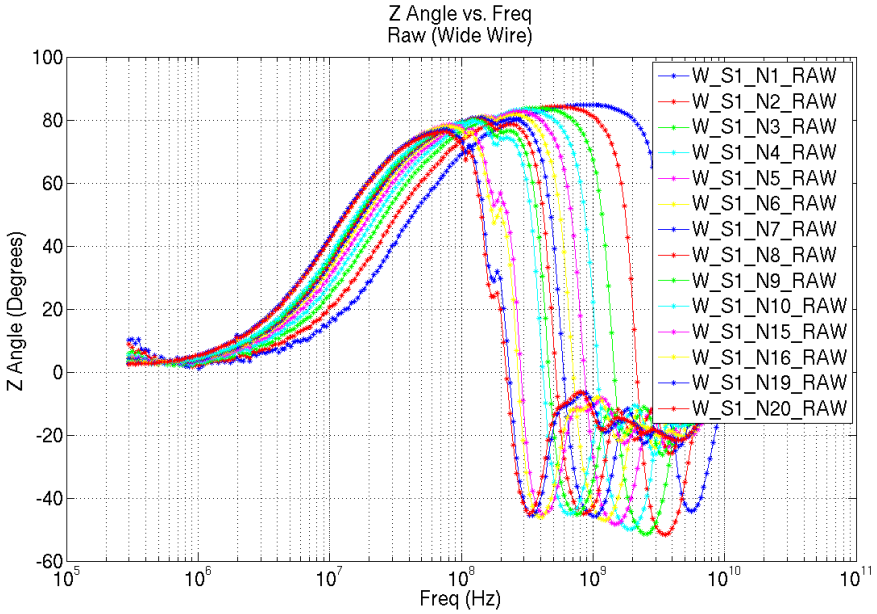


Figure B.4 Z Phase Angle vs. Frequency Wide Trace Site Group 1

Wide Trace Site Group 2

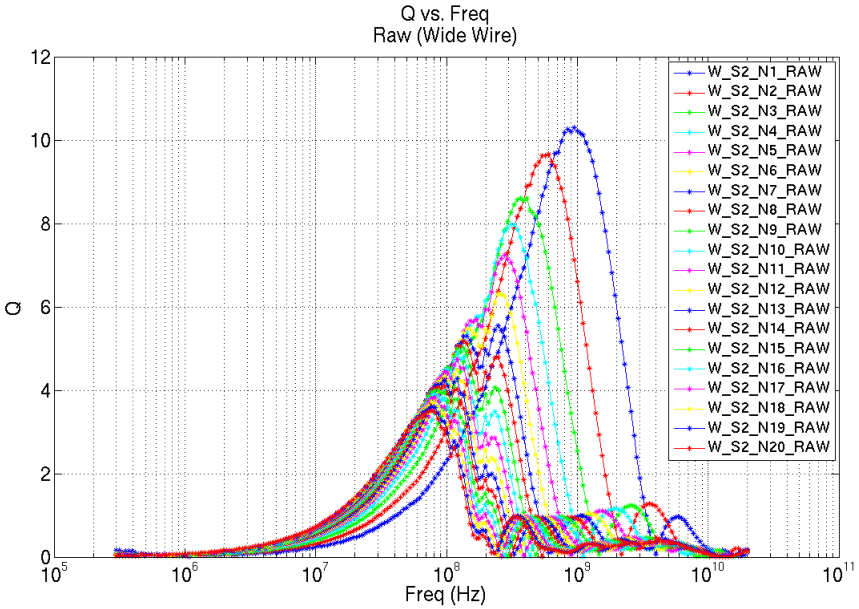


Figure B.5 Q vs. Frequency Wide Trace Site Group 2

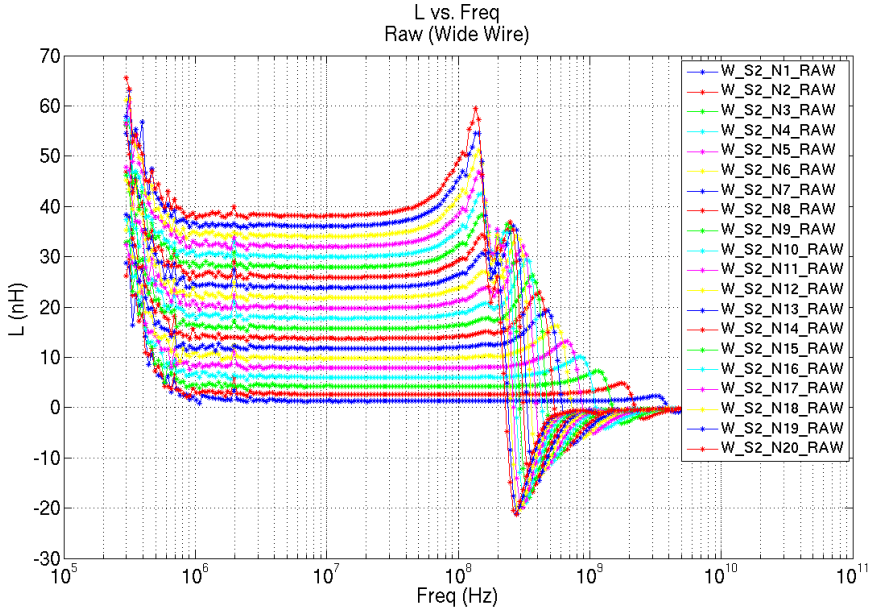


Figure B.6 L vs. Frequency Wide Trace Site Group 2

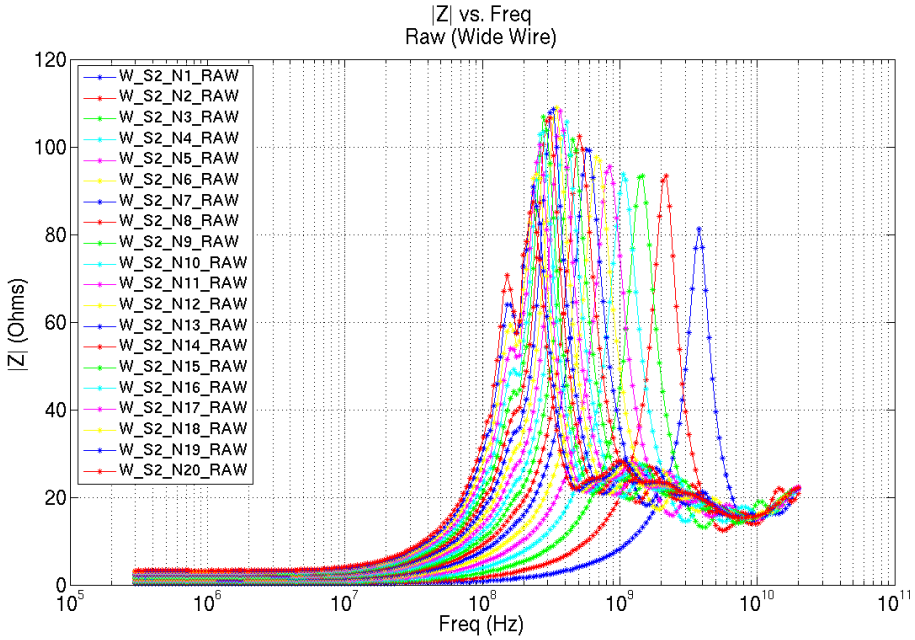


Figure B.7 |Z| vs. Frequency Wide Trace Site Group 2

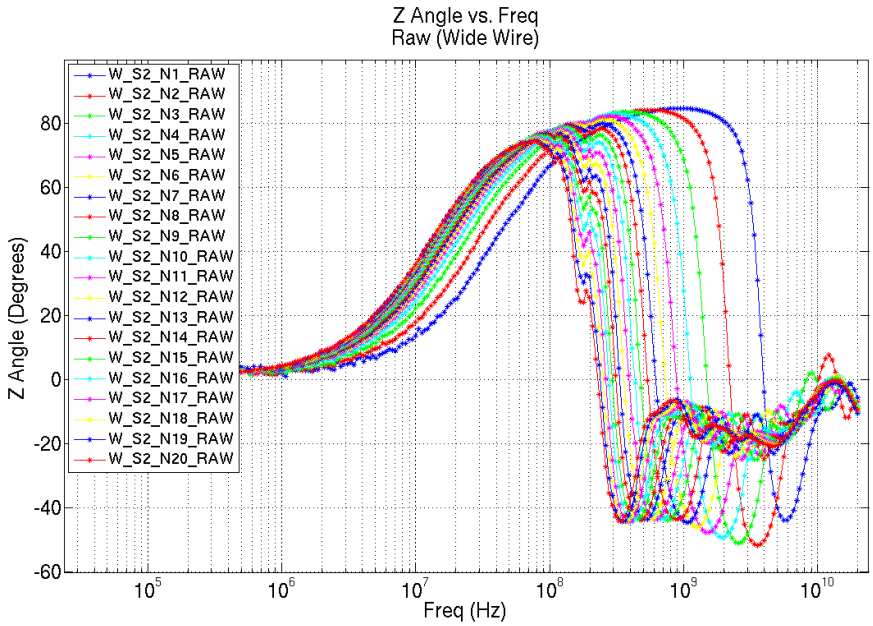


Figure B.8 Z Phase Angle vs. Frequency Wide Trace Site Group 2

Narrow Trace Site Group 3

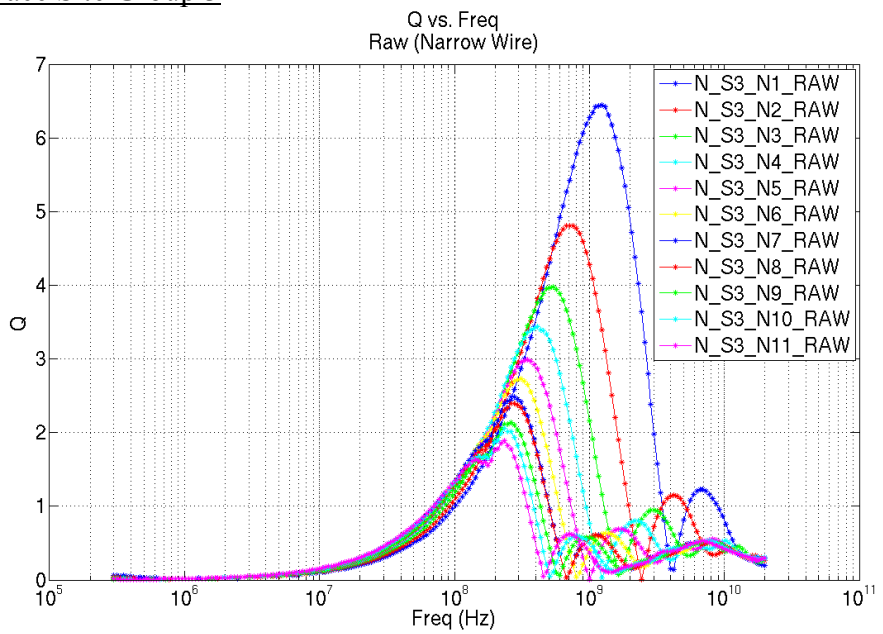


Figure B.9 Q vs. Frequency NarrowTrace Site Group 3

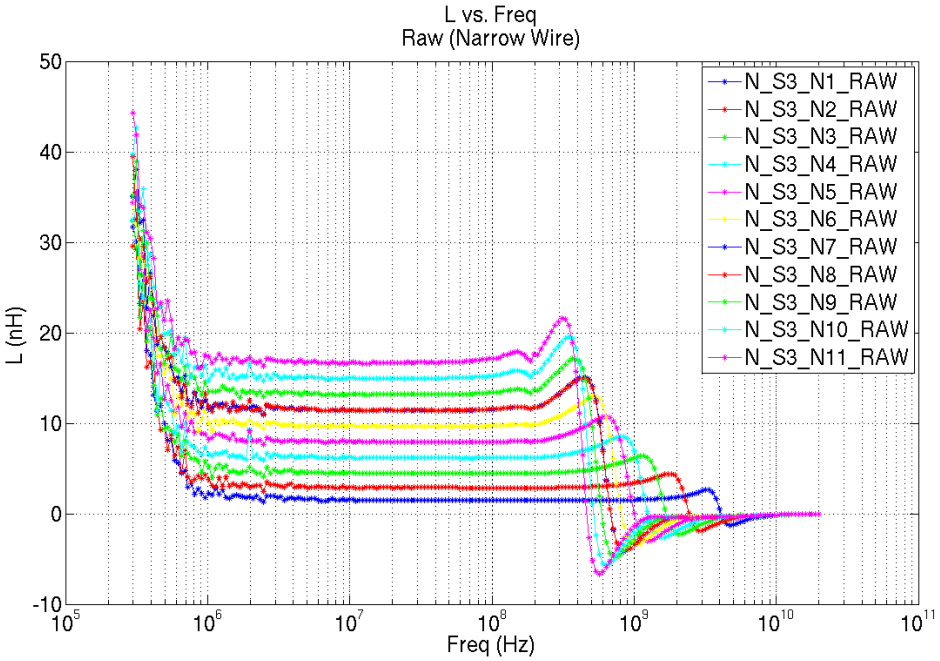


Figure B.10 L vs. Frequency NarrowTrace Site Group 3

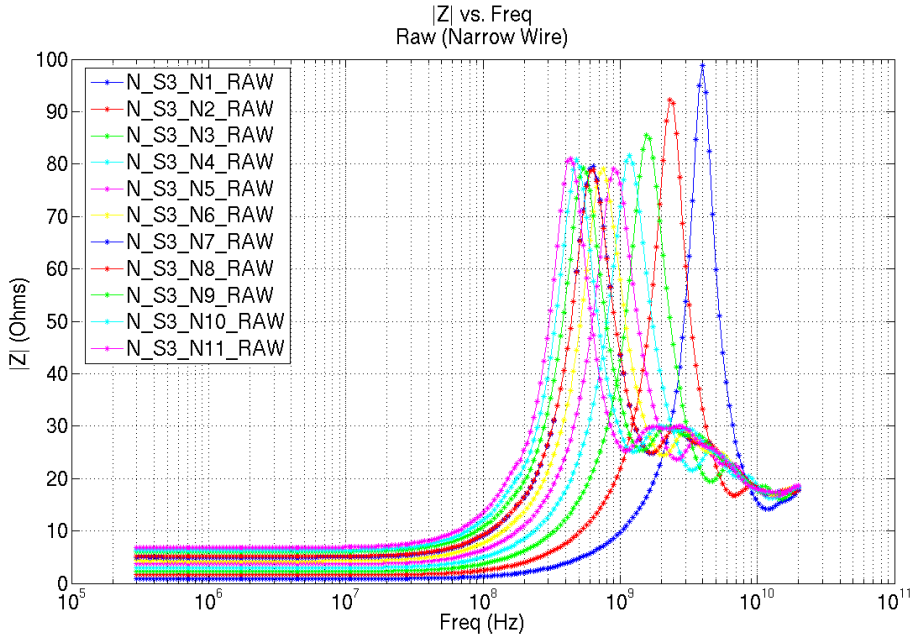


Figure B.11 |Z| vs. Frequency NarrowTrace Site Group 3

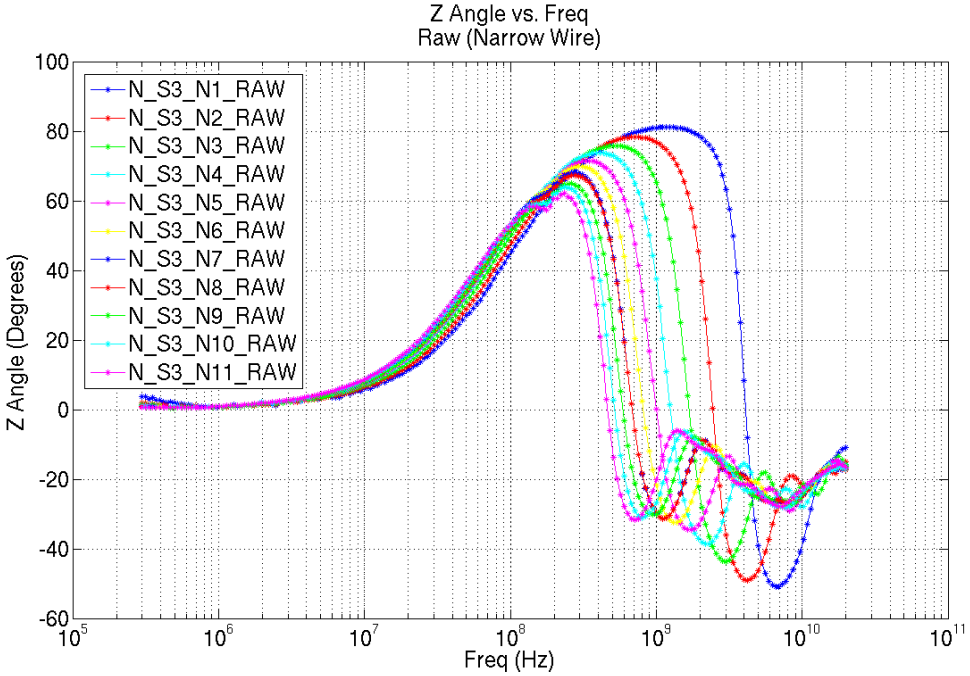


Figure B.12 Z Phase Angle vs. Frequency NarrowTrace Site Group 3

Narrow Wire Site Group 4

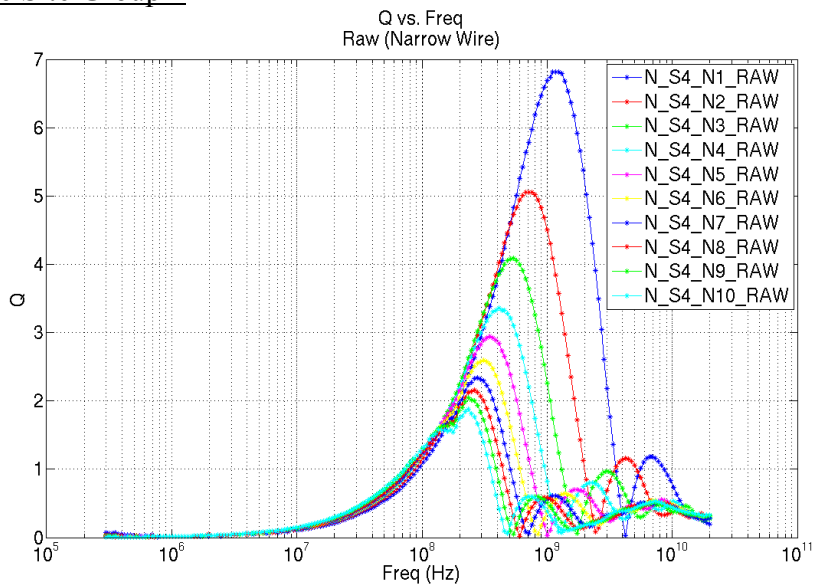


Figure B.13 Q vs. Frequency NarrowTrace Site Group 4

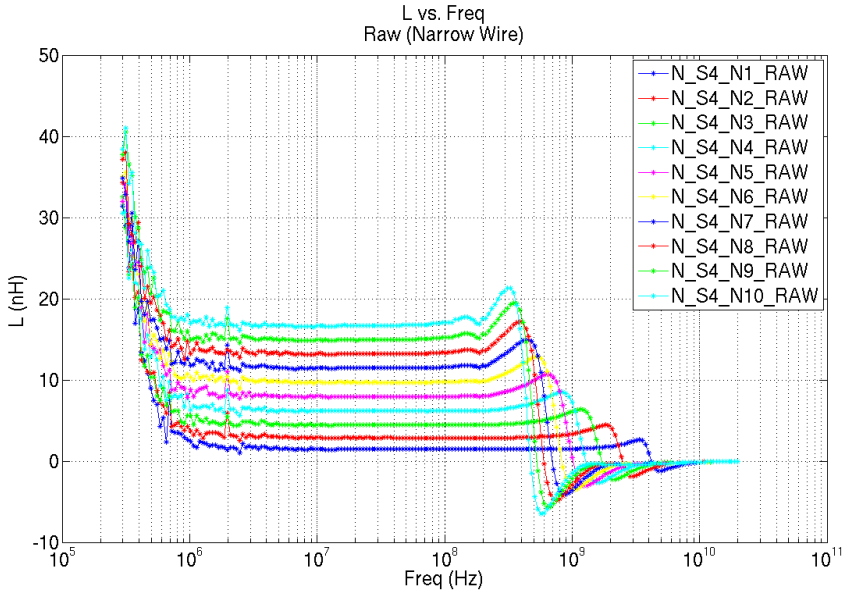


Figure B.14 L vs. Frequency NarrowTrace Site Group 4

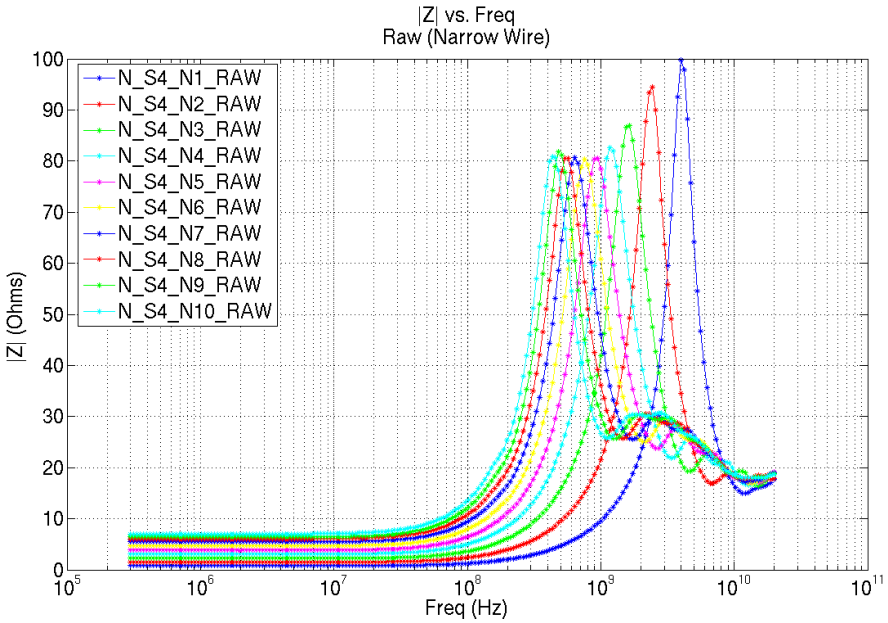


Figure B.15 |Z| vs. Frequency NarrowTrace Site Group 4

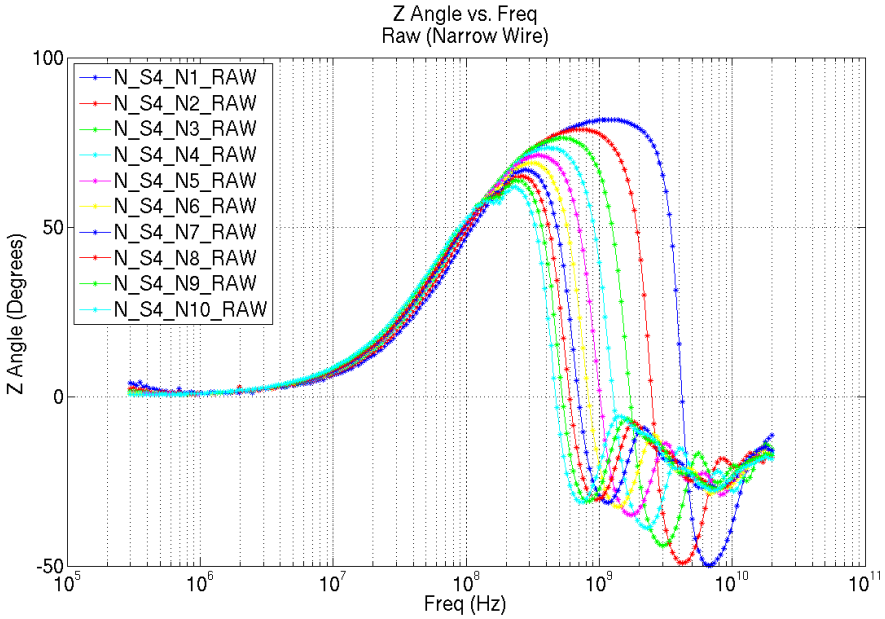


Figure B.16 Z Phase Angle vs. Frequency NarrowTrace Site Group 4

Layout Extracted Data

Wide Wire Trace Groups Length Data

Wide Wire Lengths					
Wide Wire	Wire Length	Via Length (2 via)	Total length	X - Width	Y-Height
1	849.67	1000.00	1.85E-03	1.70E-04	5.00E-04
2	1699.34	2000.00	3.70E-03	2.70E-04	5.00E-04
3	2549.01	3000.00	5.55E-03	3.70E-04	5.00E-04
4	3398.68	4000.00	7.40E-03	4.70E-04	5.00E-04
5	4248.35	5000.00	4248.36	400.00	500.00
6	5098.02	6000.00	5098.03	500.00	500.00
7	5947.69	7000.00	12947.69	600.00	500.00
8	6797.36	8000.00	14797.36	700.00	500.00
9	7647.03	9000.00	16647.03	800.00	500.00
10	8496.70	10000.00	18496.70	900.00	500.00
11	9346.37	11000.00	20346.37	1000.00	500.00
12	10196.04	12000.00	22196.04	1100.00	500.00
13	11045.71	13000.00	24045.71	1200.00	500.00
14	11895.38	14000.00	25895.38	1300.00	500.00
15	12745.05	15000.00	27745.05	1400.00	500.00
16	13594.72	16000.00	29594.72	1500.00	500.00
17	14444.39	17000.00	31444.39	1600.00	500.00
18	15294.06	18000.00	33294.06	1700.00	500.00
19	16143.73	19000.00	35143.73	1800.00	500.00
20	16993.40	20000.00	36993.40	1900.00	500.00

Table B.1 Wide Trace Wire Groups Length vs. N_{Turns}

Narrow Trace Wire Groups Length Data

Narrow Wire Lengths					
Thin Wire	Wire Length	Via Length (2 via)	Total length	X - Width	Y-Height
1	849.67	1000.00	1849.67	205.00	500.00
2	1699.34	2000.00	3699.34	330.00	500.00
3	2549.01	3000.00	5549.01	455.00	500.00
4	3398.68	4000.00	7398.68	580.00	500.00
5	4248.35	5000.00	9248.35	705.00	500.00
6	5098.02	6000.00	11098.02	830.00	500.00
7	5947.69	7000.00	12947.69	955.00	500.00
8	6797.36	8000.00	14797.36	1080.00	500.00
9	7647.03	9000.00	16647.03	1205.00	500.00
10	8496.70	10000.00	18496.70	1330.00	500.00

Table B.2 Narrow Trace Wire Groups Length vs. N_{Turns}

Wide Trace Wire Groups Nu DUT

Wide Trace (70um, 50um Via)				
N Turns	Lfix	Lleads	Ldut	ηDut
1	942	792	150	0.84
2	1092	792	300	0.73
3	1242	792	450	0.64
4	1392	792	600	0.57
5	1542	792	750	0.51
6	1692	792	900	0.47
7	1842	792	1050	0.43
8	1992	792	1200	0.40
9	2142	792	1350	0.37
10	2292	792	1500	0.35
11	2442	792	1650	0.32
12	2592	792	1800	0.31
13	2742	792	1950	0.29
14	2892	792	2100	0.27
15	3042	792	2250	0.26
16	3192	792	2400	0.25
17	3342	792	2550	0.24
18	3492	792	2700	0.23
19	3642	792	2850	0.22
20	3792	792	3000	0.21

Table B.3 Wide Wire Groups η_{dut} vs. N_{Turns}

Narrow Trace Wire Groups Nu DUT

Thin Trace (25um, 50um Via)				
N Turns	Lfix	Lleads	Ldut	ηDut
1	1032	857	175	0.83
2	1207	857	350	0.71
3	1382	857	525	0.62
4	1557	857	700	0.55
5	1732	857	875	0.49
6	1907	857	1050	0.45
7	2082	857	1225	0.41
8	2257	857	1400	0.38
9	2432	857	1575	0.35
10	2607	857	1750	0.33
11	2782	857	1925	0.31
12	2957	857	2100	0.29
13	3132	857	2275	0.27
14	3307	857	2450	0.26
15	3482	857	2625	0.25
16	3657	857	2800	0.23
17	3832	857	2975	0.22
18	4007	857	3150	0.21
19	4182	857	3325	0.20
20	4357	857	3500	0.20

Table B.4 Narrow Wire Groups η_{dut} vs. N_{Turns}

APPENDIX C

Software Written to Support Thesis

Perl Program 1-Port.pl

```
#!/usr/bin/perl
# This perl script will take as input a 1-port S-Parameter file and output the deembeded
parameters
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
#1-Port.pl -singlefile=Y -deembed=ALL -inputfile=dirbatch

# Process Parameters
#$mu_copper = 1.2566290 × 10-6 H/m;
#Parylene: 1 um
#Ti: 1000 A
#Cu: 2 - 4 um
# Via Size = 50um
# Trace width Wide = 70um
# Trace width Narrow = 25um
# Wafer thickness or via length = 500um
# total length per N = 1854um
$l_per_turn=.001854;

# Constants
use constant PI=> 4 * atan2(1,1);
use Math::Complex;
use Math::Trig;

$pi=3.141592653;
$sigma_copper =59600000;
$Zo=50;

@inductor_length = (0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20);
# Calcs with Effective length.
# For 1-port measurement
@new_dut_N =
(0.0,0.92,0.87,0.82,0.79,0.76,0.74,0.72,0.70,0.69,0.68,0.67,0.66,0.65,0.64,0.63,0.63,0.62
,0.62,0.61,0.61);
@new_dut_W =
(0.0,0.93,0.87,0.83,0.80,0.77,0.75,0.73,0.71,0.70,0.68,0.67,0.66,0.66,0.65,0.64,0.63,0.63
,0.62,0.62,0.61);

# For 2-port measurement
#@new_dut_N =
(0.0,0.83,0.71,0.62,0.55,0.49,0.45,0.41,0.38,0.35,0.33,0.31,0.29,0.27,0.26,0.25,0.23,0.22
,0.21,0.20,0.20);
#@new_dut_W =
(0.0,0.84,0.73,0.64,0.57,0.51,0.47,0.43,0.40,0.37,0.35,0.32,0.31,0.29,0.27,0.26,0.25,0.24
,0.23,0.22,0.21);

# Narrow Wire and Wide Space Factors
$width_narrow_wire = .00000025;
$width_wide_space = .00000100;
$via2via_narrow_space = .00000125;

# Wide Wire and Narrow Space Factors
$width_wide_wire = .00000070;
$width_narrow_space = .00000030;
$via2via_wide_space = .00000100;

$type = "";
# Keep me from divide by zero
$R       = 0.00000000000000000001;
```

```
$XL      = 0.00000000000000000001;
$ZL      = 0.00000000000000000001;
$Leff    = 0.00000000000000000001;
$Z11_mag = 0.00000000000000000001;

# De-embedding Array
@Z_O=(); # Open
@Y_O=(); # Open
@Y_S=(); # Short
@Z_S=(); # Short
#@Z_T=(); # Thru
#@Y_T=(); # Thru

my $counter = 0;
my $fileout_S = "cal_Single"; # Calculated output file
#my $fileout_D = "$file_D"; # Calculated output file with deemb
my $fileout_SD = "dem_Single"; # Calculated output file with deemb
my $fileout_SDA = "dem_Single_all"; # Calculated output file with deemb
my $file_THRU="";
my $N = "";
my $N_D = "";

# Input Arg defs
$inputfile="";
$deembed="N";
$singlefile="Y";
$clean="";
$debug;
$help;

#Process the input arguments
foreach (@ARGV)
{

        #Command Line Arguments
        /^\-inputfile=([\w\/.-]+)$/ and do {$inputfile = $1 ;  next};
        /^\-deembed=([\w\/,-]+)$/ and do {$deembed = $1 ;  next};
        /^\-singlefile=([\w\/,-]+)$/ and do {$singlefile = $1 ;  next};
        /^\-clean=([\w\/,-]+)$/ and do {$clean = $1 ;  next};
        /^\-debug=([\w\/\,]+)$/ and do {$debug = $1;  next};
        /^\-help$/ and die "Usage:\n -inputfile=[S1N1,S1N2,S1N3, or dirbatch] -
deembed=[OPD,OSD,OSTD, ALL, or leave blank for raw] -singlefile=[Y/N] \n -";
        die "Bogus argument $_ (not supported)\n";
}

#split the file names
chomp($inputfile);
@file_list = split(",", $inputfile);

if ($debug =~ "inputfile"){print "INPUTFILE:@file_list\n";};

# If want to batch process all .s1p files
if ($inputfile =~ "dirbatch")
{
  #chomp($inputfile);
  #@file_list = split(",", $inputfile);

  # Must delete the preexisting CAL files first
  `\\rm cal*`;
  @SearchFiles = `ls *.S1P`;
  @SearchThruFiles = `ls Thru_De_Embed_*.S1P`;
```

```
  @file_list = ();

        # Move the Thru cal files to the Thru Array
        foreach $f (@SearchFiles)
        {
                if ($f =~ m/^Thru_De_Embed_|Open_De_Embed|Short_De_Embed/)
                {
                        next;
                }
                else
                {
                        push(@file_list, $f);
                        $TEMP = substr($f, 0, 1);

                        if (substr($f, 1, 1) =~ "F"|substr($f, 1, 1) =~ "S")
                        {
                                push(@file_listN, $f);
                        }
                        elsif (substr($f, 2, 1) =~ "F"|substr($f, 2, 1) =~ "S")
                        {
                                $f = substr($f, 1);
                                push(@file_listN, $f);
                        };
                };
        }
}
elsif ("singlefilebatch")
{
};


# Create the Files as needed by De-embedding
if ($singlefile =~ "Y" && $deembed =~ "ALL" )
{
        open (OUT_SDA, ">$fileout_SDA")
                || die ("unable to open $fileout_SDA $!");
};
if ($singlefile =~ "Y" && ($deembed =~ "OPD" || $deembed =~ "OSD" || $deembed =~ "OSTD"
|| $deembed =~ "ALL"))
{
        open (OUT_S, ">$fileout_S")
                || die ("unable to open $fileout_S $!");
        open (OUT_SD, ">$fileout_SD")
                || die ("unable to open $fileout_SD $!");
}
elsif ($singlefile =~ "Y" && $deembed =~ "N")
{
        open (OUT_S, ">$fileout_S")
                || die ("unable to open $fileout_S $!");
};

# print the files output header
print "\n           PLEASE SEE FILES:\n\n";

foreach $file (@file_list)
{
        if ($file =~ m/Open|Short|Thru/gi)
        {
                next;
        };
        $TEMP=substr($file,2, 1);
```

```
        if (substr($file, 1, 1) =~ "F"|substr($f, 1, 1) =~ "S")
        {
                $fileN = substr($file, 1);
        }
        elsif (substr($file, 2, 1) =~ "F"|substr($file, 2, 1) =~ "S")
        {
                $fileN = substr($file, 2);
        };

        my $fileIn = "$file"; # Input file
    my $fileout = "cal_$fileN"; # Calculated output file
    #my $Ind_Sec_Num = $fileIn;
    my $Ind_Sec_Num = $fileN;
        #print "IND:$Ind_Sec_Num\n";

        my $Ind_Sec_Num_D = "";
        $Ind_Sec_Num =~ s/cal_//gi;
    $Ind_Sec_Num =~ s/.S1P//gi;
    $Ind_Sec_Num_D = $Ind_Sec_Num;
        #print "IND:$Ind_Sec_Num_D\n";
        $Ind_Sec_Num =~ s/_/\\_/gi;
        $Ind_Sec_Num_D =~ s/_/\\_/gi;
        chomp($Ind_Sec_Num);
    chomp($Ind_Sec_Num_D);
    #print "$Ind_Sec_Num\n";

    if ($singlefile =~ "N" && ($deembed =~ "OPD" || $deembed =~ "OSD" ||$deembed =~
"OSTD" || $deembed =~ "ALL"))
        {
                open (OUT_D, ">$fileout")
                        || die ("unable to open $fileout_D $!");
        };


        if ( $deembed =~ "OPD" )
        {
                        $Ind_Sec_Num_D = "$Ind_Sec_Num_D"."\\_OPD";
        }
        elsif ( $deembed =~ "OSD" )
        {
                        $Ind_Sec_Num_D = "$Ind_Sec_Num_D"."\\_OSD";
        }
        elsif ( $deembed =~ "OSTD")
        {
                        $Ind_Sec_Num_D = "$Ind_Sec_Num_D"."\\_OSTD";
        }
        elsif ( $deembed =~ "ALL" )
        {
                $Ind_Sec_Num_OPD = "$Ind_Sec_Num"."\\_OPD";
                $Ind_Sec_Num_OSD = "$Ind_Sec_Num"."\\_OSD";
                $Ind_Sec_Num_OSTD = "$Ind_Sec_Num"."\\_OSTD";
        };

        open (INA, "$fileIn")
          || die ("unable to open $fileIn $!");

        open (OUT, ">$fileout")
                || die ("unable to open $fileout $!");
```

```
        # find out how many turns N the inductor is
        if ($Ind_Sec_Num_D =~ m/F\\_/gi)
        {
                $narrow_or_wide = "W";
                #$Ind_Sec_Num_D =~ s/F_//gi;
        }
        elsif ($Ind_Sec_Num =~ m/S\\_/gi)
        {
                $narrow_or_wide = "N";
                #$Ind_Sec_Num_D =~ s/S_//gi;
        };
        #$Ind_Sec_Num_D =~ s/_//gi;

        $N = $Ind_Sec_Num;
        $N =~ s/S1//gi;
        $N =~ s/S2//gi;
        $N =~ s/S3//gi;
        $N =~ s/S4//gi;
        $N =~ s/S//gi;
        $N =~ s/F//gi;
        $N =~ s/N//gi;
        $N_D = $N;
        $N =~ s/\\_//gi;
        #print "N:$N\n";

        ###############################################################
        # Open the Open deembed file, then create Z, Y, R, L from it.
        # Process for OPD De-embedding
        # to get Yp due to pad parasitics
        ###############################################################
        if ($deembed =~ "OPD" || $deembed =~ "OSD" || $deembed =~ "OSTD" || $deembed =~
"ALL")
        {
                my $file_OPEN = "Open_De_Embed.S1P";
                open (IN_D_OPEN, "$file_OPEN")
                        || die ("unable to open $file_OPEN $!");


                        @datafile_O = <IN_D_OPEN>;
                        $counter = 0;
                        foreach $line (@datafile_O)
                        {
                                if ( ($line=~ m/\!|#/g) )
                                {
                                        next;
                                }
                                else
                                {
                                        @dataline = split(/\s+/,$line);
                                        if ($dataline[0] =~ "")
                                        {
                                                next;
                                        }
                                        else
                                        {
                                                $freq_O = $dataline[0];
                                                $S11_real_O = $dataline[1];
                                                $S11_img_O = $dataline[2];
                                                $R_O[$counter] = -
$Zo*(($S11_real_O**2)+($S11_img_O**2)-1)/(($S11_real_O**2)+($S11_img_O**2)-
(2*($S11_real_O))+1);
```

```
                                        $X_O[$counter] = ($S11_img_O * 100) / (
($S11_real_O**2) + ($S11_img_O**2) - (2*$S11_real_O) + 1);
                                        $Z_O[$counter] = ($R_O[$counter] +
($X_O[$counter]*i));
                                        $Y_O[$counter] = 1/$Z_O[$counter];
                                        $Z_NEW[$counter] = ( ( 1 + ($S11_real_O +
$S11_img_O*i) ) * 50) / ( 1 - ($S11_real_O + $S11_img_O*i));
                                        #print "R:$R_O[$counter]
X:$X_O[$counter]\n";
                                        #print "ZO:$Z_O[$counter]
ZNEW:$Z_NEW[$counter]\n";
                                        $counter = $counter + 1;
                                    };
                            };
                    };
                    $counter = $counter - 1; # remove the last increment to get the
correct size.
        };

        ############################################################
        # Open the Short deembed file, then create Z, Y, R, L from it.
        # Process for OSD De-embedding, requires that Open De-embedding has already
processed
        ############################################################
        if ( $deembed =~ "OSD" || $deembed =~ "OSTD" || $deembed =~ "ALL")
        {
                my $file_SHORT = "Short_De_Embed.S1P";
                open (IN_D_SHORT, "$file_SHORT")
                          || die ("unable to open $file_SHORT $!");

                @datafile_S = <IN_D_SHORT>;

                $counter = 0;
                foreach $line (@datafile_S)
                {
                          if ( ($line=~ m/\!|#/g) )
                        {
                                        next;
                        }
                        else
                        {
                                @dataline = split(/\s+/,$line);
                                if ($dataline[0] =~ "")
                                {
                                        next;
                                }
                                else
                                {
                                        $freq_S = $dataline[0];
                                        $S11_real_S = $dataline[1];
                                        $S11_img_S = $dataline[2];
                                        $R_S[$counter] = -
$Zo*(($S11_real_S**2)+($S11_img_S**2)-1)/(($S11_real_S**2)+($S11_img_S**2)-
(2*($S11_real_S))+1);
                                        $X_S[$counter] = ($S11_img_S * 100) /
((($S11_real_S**2) + ($S11_img_S**2)-(2*$S11_real_S)+1));
                                        $Z_S[$counter] = ($R_S[$counter] +
($X_S[$counter]*i));
                                        $Y_S[$counter] = 1/$Z_S[$counter];
                                        #print "R:$R_O[$counter]
X:$X_O[$counter]\n";
```

```
                                                #print "ZO:$Z_O[$counter]
ZNEW:$Z_NEW[$counter]\n";
                                                $counter = $counter + 1;
                                        };
                                };
                        };
                        $counter = $counter - 1;
        };

        ############################################################
        # Open the Thru deembed file, then create Z, Y, R, L from it.
        # Process for OSD De-embedding, requires that Open De-embedding has already
processed
        ############################################################
        @Z_T=(); # Thru
        @Y_T=(); # Thru
        if ( $deembed =~ "OSTD" || $deembed =~ "ALL")
        {
                my $file_THRU = "Thru_De_Embed_"."$fileN";
                open (IN_D_THRU, "$file_THRU")
                          || die ("unable to open $file_THRU $!");

                        @datafile_T = <IN_D_THRU>;

                        $counter = 0;
                        foreach $line (@datafile_T)
                        {
                                  if ( ($line=~ m/\!|#/g) )
                                {
                                                next;
                                }
                                else
                                {
                                        @dataline = split(/\s+/,$line);
                                        if ($dataline[0] =~ "")
                                        {
                                                next;
                                        }
                                        else
                                        {
                                                $freq_T = $dataline[0];
                                                $S11_real_T = $dataline[1];
                                                $S11_img_T = $dataline[2];
                                                $R_T[$counter] = -
$Zo*(($S11_real_T**2)+($S11_img_T**2)-1)/(($S11_real_T**2)+($S11_img_T**2)-
(2*($S11_real_T))+1);
                                                $X_T[$counter] = ($S11_img_T * 100) /
((($S11_real_T**2) + ($S11_img_T**2)-(2*$S11_real_T)+1));
                                                $Z_T[$counter] = ($R_T[$counter] +
($X_T[$counter]*i));
                                                $Y_T[$counter] = 1/$Z_T[$counter];
                                                $counter = $counter + 1;
                                        };
                                };
                        };
                        $counter = $counter - 1;
        };

        ########################################################
        # S11 to Y11 conversion = Y11 = (1+S11)/(1-S11)
        # ALL the values have been normalized at this point.
```

```
	#######################################################

	# if De-embedding is selected, go calculate the Deembed Thru values based off of
N.
	@datafile = <INA>;

	# Reset the counter so we start at the begining of the file.
	$counter = 0;
	foreach $line (@datafile)
	{
		if ($debug =~ "Line"){print "LINE:$line\n";};
		{
			@temp_array = split(/=/,$line);
			shift(@temp_array);
		};

		if ( ($line=~ m/\!/g) )
		{
			# Skip the line if comment and has no value
		next;
		}
		elsif ( ($line=~ m/#/g) )
		{
			$line =~ s/# //gi; # Remove the # indicator
			@header = split(/\s+/,$line);
			$Zo=$header[4];
		}
		else
		{
			# Get S11 and R from the file
			@dataline = split(/\s+/,$line);
			if ($dataline[0] =~ "")
			{
				next;
			}
			else
			{
				if ($debug =~ "data"){print "Freq: $freq\n";};
				if ($debug =~ "data"){print "S11R: $S11_real\n";};
				if ($debug =~ "data"){print "S11I: $S11_img\n";};

				# Calculations based on S11 values
				$freq = $dataline[0];
				$S11_real = $dataline[1];
				$S11_img = $dataline[2];

				$R = -$Zo*(($S11_real**2)+($S11_img**2)-
1)/(($S11_real**2)+($S11_img**2)-(2*($S11_real))+1);
				$X = ($S11_img * 100) / ((($S11_real**2) + ($S11_img**2)-
(2*$S11_real)+1));

				$Z = ($R + ($X*i));
				$Z_mag = sqrt(($R**2)+($X**2));

				$Y = 1 / $Z;
				$Y_real = Re($Z);
				$Y_img = Im($Z);

				$theta = atan2($X,$R);
```

```
                                $L = $X/(2 * PI * $freq);  # the i in top cancels the i in
the bottom.
                                $Q = abs($X*i/$R);

                                # Proving it to myself
                                $Reff =  ($Z_mag * cos($theta)); # Reff = R
                                $L1 =  ($Z_mag * sin($theta));
                                $Q2 = -($Y_img/$Y_real);

                                if ( $deembed =~ "OPD" || $deembed =~ "OSD" || $deembed =~
"OSTD" || $deembed =~ "ALL")
                                {
                                        # Grab Z from the Open File
                                        $Z_OP = $Z_O[$counter];

                                        # deembeded Z OPD
                                        $Z_OPD = 1 / ( $Y - $Y_O[$counter]);

                                        # Calculated off of deembeded Z OPD
                                        $R_OPD = Re($Z_OPD);
                                        $X_OPD = Im($Z_OPD);
                                        $Z_mag_OPD = $Z_mag_D =
sqrt(($R_OPD**2)+($X_OPD**2));  # Double assignment allows for passing of either Single
or ALL Deembed value.
                                        $theta_OPD = $theta_D = atan2($X_OPD,$R_OPD);
                                        $L_OPD = $L_D = $X_OPD/(2 * PI * $freq);
                                        $Q_OPD = $Q_D = abs($X_OPD*i/$R_OPD);
                                };
                                if ( $deembed =~ "OSD" || $deembed =~ "OSTD" || $deembed =~
"ALL")
                                {
                                        # Grab Z from the Short file
                                        $Z_OS = $Z_S[$counter];

                                        # Calculate Yp = 1/(Zopen - Zshort) Z_OP is taken
from above and is indexed
                                        $Y_S_D = 1/($Z_OP - $Z_OS);

                                        # Deembeded Z OSD
                                        $Z_OSD = 1 / ( $Y - $Y_S_D);

                                        # Calculated off of deembeded Z OSD
                                        $R_OSD = Re($Z_OSD);
                                        $X_OSD = Im($Z_OSD);
                                        $Z_mag_OSD = $Z_mag_D =
sqrt(($R_OSD**2)+($X_OSD**2));
                                        $theta_OSD = $theta_D = atan2($X_OSD,$R_OSD);
                                        $L_OSD = $L_D = $X_OSD/(2 * PI * $freq);
                                        $Q_OSD = $Q_D = abs($X_OSD*i/$R_OSD);
                                };

                                if ($deembed =~ "OSTD" || $deembed =~ "ALL")
                                {
                                        # OSD Deembed the Thru Y11 to create Y'11,Thru
                                        $Y11_T = $Y_T[$counter] - $Y_S_D;

                                        # If narrow or wide
                                        if ($narrow_or_wide =~ "W")
                                        {
                                                $new_dut = $new_dut_W[$N];
                                        }
```

```
                                        elsif ($narrow_or_wide =~ "N")
                                        {
                                                $new_dut = $new_dut_S[$N];
                                        };

                                        $Z_l = $new_dut / (2 * $Y11_T);

                                        # Deembed Z OSTD
                                        $Z_OSTD = $Z_OSD - $Z_l;

                                        # Calculated off of deembeded Z OSTD
                                        $R_OSTD = Re($Z_OSTD);
                                        $X_OSTD = Im($Z_OSTD);
                                        $Z_mag_OSTD = $Z_mag_D =
sqrt(($R_OSTD**2)+($X_OSTD**2));
                                        $theta_OSTD = $theta_D = atan2($X_OSTD,$R_OSTD);
                                        $L_OSTD = $L_D = $X_OSTD/(2 * PI * $freq);
                                        $Q_OSTD = $Q_D = abs($X_OSTD*i/$R_OSTD);
                                };

                                # default is to print to a single file
                                if ($singlefile =~ "Y")
                                   {
                                        # print into a single file with no De-embedding RAW
                                        print OUT_S
"$N,$Ind_Sec_Num,$freq,$L,$Z_mag,$theta,$Q,\n";

                                        # print into a single file with DESIGNATED De-
embedding TECHNIQUE
                                        if ($deembed =~ "OPD" || $deembed =~ "OSD" ||
$deembed =~ "OSTD")
                                        {
                                                print OUT_SD
"$N,$Ind_Sec_Num,$freq,$L,$Z_mag,$theta,$Q,$Ind_Sec_Num_D,$L_D,$Z_mag_D,$theta_D,$Q_D,\n"
;
                                        }
                                        # print into a single file with ALL De-embedding
TECHNIQUES
                                        elsif ($deembed =~ "ALL")
                                        {
                                                print OUT_SDA
"$N,$Ind_Sec_Num,$freq,$L,$Z_mag,$theta,$Q,$Ind_Sec_Num_OPD,$L_OPD,$Z_mag_OPD,$theta_OPD,
$Q_OPD,$Ind_Sec_Num_OSD,$L_OSD,$Z_mag_OSD,$theta_OSD,$Q_OSD,$Ind_Sec_Num_OSTD,$L_OSTD,$Z_
mag_OSTD,$theta_OSTD,$Q_OSTD,\n";
                                        };
                                };
                                $counter = $counter + 1;
                                # Always print the calculated file with no De-embedding RAW
                                print OUT "$N,$Ind_Sec_Num,$freq,$L,$Z_mag,$theta,$Q,\n";
                        };
                };
        };
        if ($inputfile =~ "dirbatch")
        {
                print "            Orig Calculated:  $fileout";
        }
        else
        {
                print "            Orig Calculated:  $fileout\n";
        };
        close(INA);
```

```
        close(OUT);
        close(OUTD);
};

# Finish printing output filenames if there is Single file or Deembedding
if ($singlefile =~ "Y")
{
        print "          Single File RAW Calcs:  $fileout_S\n";
};
if ($deembed =~ "OPD" || $deembed =~ "OSD" ||$deembed =~ "OSTD" )
{
        print "  Single File Raw Calculated + Deemb:  $fileout_SD       Deembed Method:
$deembed\n";
};
if ($deembed =~ "ALL")
{
        print "  Single File Raw Calculated + Deemb:  $fileout_SDA   Deembed Method:
$deembed\n";
};
print "\n\n";
close(OUT_S);
close(OUT_SD);
close(OUT_SDA);

function [OK, msg] = ReadS1P(mode,titleheading)
```

Matlab Program ReadS1p.m

```
function [OK, msg] = ReadS1P(mode,titleheading)

%addpath('U:\Thesis_011\April')
%addpath('\u\gvanackern\Thesis_08\April\')
set(findobj('type','axes'),'FontSize',20)

% This function assumes that the file was saved with a single layout
OK = 0;
msg = '';

% set the window to display long eng format
format long eng

% Read the first line to find out how many variables are involved
if isequal('raw',mode) | isequal('other',mode)
    filename = 'cal_Single'
    % read the first line and determine how many elements there are
    firstLine = textread(filename, '%s', 1, 'delimiter','\n\r');
    % How many commas since each variable is delimited by a comma
    numVars = sum(firstLine{1} == char(44));
    % Now read in the whole file delimited on commas
    data = textread(filename, '%s', 'whitespace', '\b\n\r ', 'delimiter',',');
    % import the data and assign to respective variable.
   [N,inductor,freq,L,Z_mag,theta,Q]=textread(filename, '%f %s %f %f %f %f %f',
'whitespace', '\b\n\r ', 'delimiter',',');
elseif isequal('deembed',mode) |isequal('deembed RAW',mode) | isequal('deembed
OPD',mode)| isequal('deembed OSD',mode)| isequal('deembed OSTD',mode)|isequal('deembed
RAWOSTD',mode)
    filename = 'dem_Single'
    firstLine = textread(filename, '%s', 1, 'delimiter','\n\r');
    % How many tabs since each variable is delimited by a tab
    numVars = sum(firstLine{1} == char(44));
    % Now read in the whole file delimited on commas
    data = textread(filename, '%s', 'whitespace', '\b\n\r ', 'delimiter',',');
    % import the data and assign to respective variable.

[N,inductor,freq_D,L,Z_mag,theta,Q,inductor_D,L_D,Z_mag_D,theta_D,Q_D]=textread(filename,
'%f %s %f %f %f %f %s %f %f %f %f', 'whitespace', '\b\n\r ', 'delimiter',',');
elseif isequal('deembed ALL',mode)
    filename = 'dem_Single_all'
    firstLine = textread(filename, '%s', 1, 'delimiter','\n\r');
    % How many tabs since each variable is delimited by a tab
    numVars = sum(firstLine{1} == char(44));
    % Now read in the whole file delimited on commas
    data = textread(filename, '%s', 'whitespace', '\b\n\r ', 'delimiter',',');
    % import the data and assign to respective variable.

[N,inductor_DA,freq_D,L_D,Z_mag_D,theta_D,Q_D,inductor_DO,L_DO,Z_mag_DO,theta_DO,Q_DO,ind
uctor_DOS,L_DOS,Z_mag_DOS,theta_DOS,Q_DOS,inductor_DOST,L_DOST,Z_mag_DOST,theta_DOST,Q_DO
ST]=textread(filename, '%f %s %f %f %f %f %f %s %f %f %f %f %s %f %f %f %f %s %f %f %f %f
', 'whitespace', '\b\n\r ', 'delimiter',',');
elseif isequal('other',mode)
    % Skip to the other plots
end


% Have we got the number of variables correct
numRows = length(data)/numVars;
```

```
if floor(numRows) ~= numRows
        msg = 'Unable to reconcile shape of ReadS1P text file';
        return
end
data = reshape(data, [numVars numRows])';

%Find number of inductors
numinductors=floor(numRows/200);
l=cell(1,numinductors);
for m = 1:numinductors
    if isequal('raw',mode);
        inductor = strrep(inductor,'F','W');
        inductor = strrep(inductor,'S\_S','N\_S');
    elseif isequal('deembed',mode) | isequal('deembed OPD',mode)| isequal('deembed
OSD',mode)| isequal('deembed OSTD',mode)|isequal('deembed RAWOSTD',mode);
        inductor = strrep(inductor,'F','W');
        inductor = strrep(inductor,'S\_S','N\_S');
        inductor_D = strrep(inductor_D,'F','W')
        inductor_D = strrep(inductor_D,'S\_S','N\_S');
    elseif isequal('deembed ALL',mode);
        inductor = strrep(inductor_DA,'F','W');
        inductor = strrep(inductor_DA,'S\_S','N\_S');
        inductor_D = strrep(inductor_DO,'F','W');
        inductor_D = strrep(inductor_DO,'S\_S','N\_S');
        inductor_DOS = strrep(inductor_DOS,'F','W');
        inductor_DOS = strrep(inductor_DOS,'S\_S','N\_S');
        inductor_DOST = strrep(inductor_DOST,'F','W');
        inductor_DOST = strrep(inductor_DOST,'S\_S','N\_S');
    end
end

% Sepearate the Inductors for plotting
for m = 1:numinductors
    switch (m)
        case 1
            if isequal('deembed OSTD',mode)
                col_D = '-sb';
            else
                col = '-*b';
                col_D = '-sb';
                col_DO = '-xb';
                col_DOS = '-hr';
                col_DOST = '-dg';
            end
        case 2
            if isequal('deembed OSTD',mode)
                col_D = '-sr';
            else
                col = '-*r';
                col_D = '-sr';
                col_DO = '-xb';
                col_DOS = '-hr';
                col_DOST = '-dg';
            end
        case 3
            if isequal('deembed OSTD',mode)
                col_D = '-sg';
            else
                col = '-*g';
                col_D = '-sg';
                col_DO = '-xb';
```

```
        col_DOS = '-hr';
        col_DOST = '-dg';
    end
case 4
    if isequal('deembed OSTD',mode)
        col_D = '-sc';
    else
        col = '-*c';
        col_D = '-sc';
    end
case 5
    if isequal('deembed OSTD',mode)
        col_D = '-sm';
    else
        col = '-*m';
        col_D = '-sm';
    end
case 6
    if isequal('deembed OSTD',mode)
        col_D = '-sy';
    else
        col = '-*y';
        col_D = '-sy';
    end
case 7
    if isequal('deembed OSTD',mode)
        col_D = '-sb';
    else
        col = '-*b';
        col_D = '-sb';
    end

case 8
    if isequal('deembed OSTD',mode)
        col_D = '-sr';
    else
        col = '-*r';
        col_D = '-sr';
    end
case 9
    if isequal('deembed OSTD',mode)
        col_D = '-sg';
    else
        col = '-*g';
        col_D = '-sg';
    end
case 10
    if isequal('deembed OSTD',mode)
        col_D = '-sc';
    else
        col = '-*c';
        col_D = '-sc';
    end
case 11
    if isequal('deembed OSTD',mode)
        col_D = '-sm';
    else
        col = '-*m';
        col_D = '-sm';
    end
case 12
```

```
    if isequal('deembed OSTD',mode)
        col_D = '-sy';
    else
        col = '-*y';
        col_D = '-sy';
    end
case 13
    if isequal('deembed OSTD',mode)
        col_D = '-sb';
    else
        col = '-*b';
        col_D = '-sb';
    end
case 14
    if isequal('deembed OSTD',mode)
        col_D = '-sr';
    else
        col = '-*r';
        col_D = '-sr';
    end
case 15
    if isequal('deembed OSTD',mode)
        col_D = '-sg';
    else
        col = '-*g';
        col_D = '-sg';
    end
case 16
    if isequal('deembed OSTD',mode)
        col_D = '-sc';
    else
        col = '-*c';
        col_D = '-sc';
    end
case 17
    if isequal('deembed OSTD',mode)
        col_D = '-sm';
    else
        col = '-*m';
        col_D = '-sm';
    end
case 18
    if isequal('deembed OSTD',mode)
        col_D = '-sy';
    else
        col = '-*y';
        col_D = '-sy';
    end
case 19
    if isequal('deembed OSTD',mode)
        col_D = '-sb';
    else
        col = '-*b';
        col_D = '-sb';
    end
case 20
    if isequal('deembed OSTD',mode)
        col_D = '-sr';
    else
        col = '-*r';
        col_D = '-sr';
```

```
            end
        case 21
            if isequal('deembed OSTD',mode)
                col_D = '-sg';
            else
                col = '-*g';
                col_D = '-sg';
            end
        case 22
            if isequal('deembed OSTD',mode)
                col_D = '-sc';
            else
                col = '-*c';
                col_D = '-sc';
            end
        case 23
            if isequal('deembed OSTD',mode)
                col_D = '-sy';
            else
                col = '-*y';
                col_D = '-sy';
            end
        case 23
            if isequal('deembed OSTD',mode)
                col_D = '-sb';
            else
                col = '-*b';
                col_D = '-sb';
            end
        otherwise
                col = '-*k';
                col_D = '-k';
    end

% Calculates the needed offsets for each of the 200 freq for each
% inductor
y=((((m-1)*200)+m):(((m)*200)+m));

if isequal('raw',mode)
    l(1,m)=inductor((((m)*200)+1),1);
elseif isequal('deembed',mode) | isequal('deembed OPD',mode)| isequal('deembed
OSD',mode)| isequal('deembed OSTD',mode)|isequal('deembed RAWOSTD',mode)
    l(1,m)=inductor((((m)*200)+1),1);
    l_D(1,m)=inductor_D((((m)*200)+1),1);
elseif isequal('deembed ALL',mode)
    l_DA(1,m)=inductor_DA((((m)*200)+1),1);
    l_DO(1,m)=inductor_DO((((m)*200)+1),1);
    l_DOS(1,m)=inductor_DOS((((m)*200)+1),1);
    l_DOST(1,m)=inductor_DOST((((m)*200)+1),1);
end


if isequal('raw',mode) | isequal('deembed OPD',mode) | isequal('deembed OSD',mode) |
isequal('deembed OSTD',mode) | isequal('deembed RAWOSTD',mode) | isequal('deembed
ALL',mode)
    figure(1)
        set(findobj('type','axes'),'FontSize',20)
        set(gcf, 'color', 'white');
        if isequal('raw',mode)
            semilogx(freq(y),Q(y),col);
```

```
        elseif isequal('deembed',mode)| isequal('deembed OPD',mode)| isequal('deembed
OSD',mode)|isequal('deembed RAWOSTD',mode)
            semilogx(freq_D(y),Q(y),col,freq_D(y),Q_D(y),col_D); % both
            %semilogx(freq_D(y),Q(y),col);semilogx(Fmax,Qmax,'.r'); % RAW
            %semilogx(freq_D(y),Q_D(y),col_D); %semilogx(Fmax_D,Qmax_D,'.r'); % OSTD
        elseif isequal('deembed ALL',mode)

semilogx(freq_D(y),Q_D(y),col_D,freq_D(y),Q_DO(y),col_DO,freq_D(y),Q_DOS(y),col_DOS,freq_
D(y),Q_DOST(y),col_DOST);
        elseif isequal('deembed OSTD',mode)
            semilogx(freq_D(y),Q_D(y),col_D);
        end
        xlabel('Freq (Hz)');
        ylabel('Q');
        if isequal('narrow',titleheading)
            if isequal('deembed RAW',mode)
                title({'Q vs. Freq';'Raw (Narrow Wire)'})
            elseif isequal('deembed OPD',mode)
                title({'Q vs. Freq';'Raw vs. Open De-embedding (Narrow Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'Q vs. Freq';'Raw vs. Open, Short De-embedding (Narrow Wire)'})
            elseif isequal('deembed RAWOSTD',mode)
                title({'Q vs. Freq';'Raw vs. Open, Short, Thru De-embedding (Narrow
Wire)'})
            elseif isequal('deembed OSTD',mode)
                title({'Q vs. Freq';'Open, Short, Thru De-embedding (Narrow Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'Q vs. Freq';'Raw vs. ALL De-embedding (Narrow Wire)'})
            end
        elseif isequal('wide',titleheading)
            if isequal('raw',mode)
                title({'Q vs. Freq';'Raw (Wide Wire)'})
            elseif isequal('deembed OPD',mode)
                title({'Q vs. Freq';'Raw vs. Open De-embedding (Wide Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'Q vs. Freq';'Raw vs. Open, Short De-embedding (Wide Wire)'})
            elseif isequal('deembed RAWOSTD',mode)
                title({'Q vs. Freq';'Raw vs. Open, Short, Thru De-embedding (Wide
Wire)'})
            elseif isequal('deembed OSTD',mode)
                title({'Q vs. Freq';'Open, Short, Thru De-embedding (Wide Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'Q vs. Freq';'Raw vs. ALL De-embedding (Wide Wire)'})
            end
        elseif isequal('narrow',titleheading)
            if isequal('raw',mode)
                title({'Q vs. Freq';'Raw (Narrow Wire)'})
            elseif isequal('deembed OPD',mode)
                title({'Q vs. Freq';'Open De-embedding (Narrow Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'Q vs. Freq';'Open, Short De-embedding (Narrow Wire)'})
            elseif isequal('deembed RAWOSTD',mode)
                title({'Q vs. Freq';'Raw vs. Open, Short, Thru De-embedding (Narrow
Wire)'})
            elseif isequal('deembed OSTD',mode)
                title({'Q vs. Freq';'Open, Short, Thru De-embedding (Narrow Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'Q vs. Freq';'Raw vs. ALL De-embedding (Narrow Wire)'})
            end
        elseif isequal('wide',titleheading)
            if isequal('raw',mode)
```

```
                title({'Q vs. Freq';'Raw (Wide Wire)'})
            elseif isequal('deembed OPD',mode)
                title({'Q vs. Freq';'Open De-embedding (Wide Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'Q vs. Freq';'Open, Short De-embedding (Wide Wire)'})
            elseif isequal('deembed RAWOSTD',mode)
                title({'Q vs. Freq';'Raw vs. Open, Short, Thru De-embedding (Wide
Wire)'})
            elseif isequal('deembed OSTD',mode)
                title({'Q vs. Freq';'Open, Short, Thru De-embedding (Wide Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'Q vs. Freq';'Raw vs. ALL De-embedding (Wide Wire)'})
            end
        elseif isequal('narrowwide',titleheading)
            if isequal('raw',mode)
                title({'Q vs. Freq';'Raw (Narrow vs. Wide Wire)'})
            elseif isequal('deembed OPD',mode)
                title({'Q vs. Freq';'Open De-embedding (Narrow vs. Wide Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'Q vs. Freq';'Open, Short De-embedding (Narrow vs. Wide Wire)'})
            elseif isequal('deembed RAWOSTD',mode)
                title({'Q vs. Freq';'Raw vs. Open, Short, Thru De-embedding (Narrow vs.
Wide Wire)'})
            elseif isequal('deembed OSTD',mode)
                title({'Q vs. Freq';'Open, Short, Thru De-embedding (Narrow vs. Wide
Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'Q vs. Freq';'Raw vs. ALL De-embedding (Narrow vs. Wide Wire)'})
            end
        elseif isequal('narrow',titleheading)
            if isequal('raw',mode)
                title({'Q vs. Freq';'Raw (Narrow Wire)'})
            elseif isequal('deembed OPD',mode)
                title({'Q vs. Freq';'Open De-embedding (Narrow Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'Q vs. Freq';'Open, Short De-embedding (Narrow Wire)'})
            elseif isequal('deembed RAWOSTD',mode)
                title({'Q vs. Freq';'Raw vs. Open, Short, Thru De-embedding (Narrow
Wire)'})
            elseif isequal('deembed OSTD',mode)
                title({'Q vs. Freq';'Open, Short, Thru De-embedding (Narrow Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'Q vs. Freq';'Raw vs. ALL De-embedding (Narrow Wide Wire)'})
            end
        end
        %xlim([41433053.31022 22598605075.35646])

        hold on

    figure(2)
        set(gcf, 'color', 'white');
        set(findobj('type','axes'),'FontSize',20)
        if isequal('raw',mode)
            semilogx(freq(y),L(y)*1E9,col);
        elseif isequal('deembed',mode)| isequal('deembed OPD',mode)| isequal('deembed
OSD',mode)|isequal('deembed RAWOSTD',mode)
           semilogx(freq_D(y),L(y)*1E9,col,freq_D(y),L_D(y)*1E9,col_D);
        elseif isequal('deembed ALL',mode)

semilogx(freq_D(y),L_D(y)*1E9,col_D,freq_D(y),L_DO(y)*1E9,col_DO,freq_D(y),L_DOS(y)*1E9,c
ol_DOS,freq_D(y),L_DOST(y)*1E9,col_DOST);
```

```
        elseif isequal('deembed OSTD',mode)
            semilogx(freq_D(y),L_D(y)*1E9,col_D);
        end
        xlabel('Freq (Hz)');
        ylabel('L (nH)');
        if isequal('narrow',titleheading)
            if isequal('raw',mode)
                title({'L vs. Freq';'Raw (Narrow Wire)'})
            elseif isequal('deembed OPD',mode)
                title({'L vs. Freq';'Open De-embedding (Narrow Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'L vs. Freq';'Open, Short De-embedding (Narrow Wire)'})
            elseif isequal('deembed RAWOSTD',mode)
                title({'L vs. Freq';'Raw vs. Open, Short, Thru De-embedding (Narrow
Wire)'})
            elseif isequal('deembed OSTD',mode)
                title({'L vs. Freq';'Open, Short, Thru De-embedding (Narrow Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'L vs. Freq';'Raw vs. ALL De-embedding (Narrow Wire)'})
            end
        elseif isequal('wide',titleheading)
            if isequal('raw',mode)
                title({'L vs. Freq';'Raw (Wide Wire)'})
            elseif isequal('deembed OPD',mode)
                title({'L vs. Freq';'Open De-embedding (Wide Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'L vs. Freq';'Open, Short De-embedding (Wide Wire)'})
            elseif isequal('deembed RAWOSTD',mode)
                title({'L vs. Freq';'Raw vs. Open, Short, Thru De-embedding (Wide
Wire)'})
            elseif isequal('deembed OSTD',mode)
                title({'L vs. Freq';'Open, Short, Thru De-embedding (Wide Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'L vs. Freq';'Raw vs. ALL De-embedding (Wide Wire)';})
            end
        elseif isequal('narrowwide',titleheading)
            if isequal('raw',mode)
                title({'L vs. Freq';'Raw (Narrow vs. Wide Wire)'})
            elseif isequal('deembed OPD',mode)
                title({'L vs. Freq';'Open De-embedding (Narrow vs. Wide Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'L vs. Freq';'Open, Short De-embedding (Narrow vs. Wide Wire)'})
            elseif isequal('deembed RAWOSTD',mode)
                title({'L vs. Freq';'Raw vs. Open, Short, Thru De-embedding ((Narrow vs.
Wide Wire)'})
            elseif isequal('deembed OSTD',mode)
                title({'L vs. Freq';'Open, Short, Thru De-embedding ((Narrow vs. Wide
Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'L vs. Freq';'Raw vs. ALL De-embedding (Narrow vs. Wide Wire)'})
            end
        elseif isequal('narrow',titleheading)
            if isequal('raw',mode)
                title({'L vs. Freq';'Raw (Narrow Wire)'})
            elseif isequal('deembed OPD',mode)
                title({'L vs. Freq';'Open De-embedding (Narrow Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'L vs. Freq';'Open, Short De-embedding (Narrow Wide Wire)'})
            elseif isequal('deembed RAWOSTD',mode)
                title({'L vs. Freq';'Raw vs. Open, Short, Thru De-embedding ((Narrow vs.
Wide Wire)'})
```

```
            elseif isequal('deembed OSTD',mode)
                title({'L vs. Freq';'Open, Short, Thru De-embedding ((Narrow vs. Wide
Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'L vs. Freq';'Raw vs. ALL De-embedding (Narrow Wide Wire)'})
            end
        end
         %xlim([41433053.31022 22598605075.35646])
        hold on

    figure(3)
    set(gcf, 'color', 'white');
        set(findobj('type','axes'),'FontSize',20)
        if isequal('raw',mode)
            semilogx(freq(y),Z_mag(y),col);
        elseif isequal('deembed',mode)| isequal('deembed OPD',mode)| isequal('deembed
OSD',mode)|isequal('deembed RAWOSTD',mode)
            semilogx(freq_D(y),Z_mag(y),col,freq_D(y),Z_mag_D(y),col_D);
        elseif isequal('deembed ALL',mode)

semilogx(freq_D(y),Z_mag_D(y),col_D,freq_D(y),Z_mag_DO(y),col_DO,freq_D(y),Z_mag_DOS(y),c
ol_DOS,freq_D(y),Z_mag_DOST(y),col_DOST);
        elseif isequal('deembed OSTD',mode)
            semilogx(freq_D(y),Z_mag_D(y),col_D);
        end
        %loglog(freq(y),Z_mag(y),col)
        xlabel('Freq (Hz)');
        ylabel('|Z| (Ohms)');
        if isequal('narrow',titleheading)
            if isequal('raw',mode)
                title({'|Z| vs. Freq';'Raw (Narrow Wire)'})
            elseif isequal('deembed OPD',mode)
                title({'|Z| vs. Freq';'Open De-embedding (Narrow Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'|Z| vs. Freq';'Open, Short De-embedding (Narrow Wire)'})
            elseif isequal('deembed RAWOSTD',mode)
                title({'|Z| vs. Freq';'Raw vs. Open, Short, Thru De-embedding (Narrow
Wire)'})
            elseif isequal('deembed OSTD',mode)
                title({'|Z| vs. Freq';'Open, Short, Thru De-embedding (Narrow Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'|Z| vs. Freq';'Raw vs. ALL De-embedding (Narrow Wire)'})
            end
        elseif isequal('wide',titleheading)
            if isequal('raw',mode)
                title({'|Z| vs. Freq';'Raw (Wide Wire)'})
            elseif isequal('deembed OPD',mode)
                title({'|Z| vs. Freq';'Open De-embedding (Wide Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'|Z| vs. Freq';'Open, Short De-embedding (Wide Wire)'})
            elseif isequal('deembed RAWOSTD',mode)
                title({'|Z| vs. Freq';'Raw vs. Open, Short, Thru De-embedding (Wide
Wire)'})
            elseif isequal('deembed OSTD',mode)
                title({'|Z| vs. Freq';'Open, Short, Thru De-embedding (Wide Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'|Z| vs. Freq';'Raw vs. ALL De-embedding (Wide Wire)'})
            end
        elseif isequal('narrowwide',titleheading)
            if isequal('raw',mode)
                title({'|Z|) vs. Freq';'Raw (Narrow vs. Wide Wire)'})
```

```
            elseif isequal('deembed OPD',mode)
                title({'|Z| vs. Freq';'Open De-embedding (Narrow vs. Wide Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'|Z| vs. Freq';'Open, Short De-embedding (Narrow vs. Wide Wire)'})
            elseif isequal('deembed RAWOSTD',mode)
                title({'|Z| vs. Freq';'Raw vs. Open, Short, Thru De-embedding (Narrow vs.
Wide Wire)'})
            elseif isequal('deembed OSTD',mode)
                title({'|Z| vs. Freq';'Open, Short, Thru De-embedding (Narrow vs. Wide
Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'|Z| vs. Freq';'Raw vs. ALL De-embedding (Narrow vs. Wide Wire)'})
            end
        elseif isequal('narrow',titleheading)
            if isequal('raw',mode)
                title({'|Z| vs. Freq';'Raw (Narrow Wire)'})
            elseif isequal('deembed OPD',mode)
                title({'|Z| vs. Freq';'Open De-embedding (Narrow Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'|Z| vs. Freq';'Open, Short De-embedding (Narrow Wide Wire)'})
            elseif isequal('deembed RAWOSTD',mode)
                title({'|Z| vs. Freq';'Raw vs. Open, Short, Thru De-embedding (Narrow vs.
Wide Wire)'})
            elseif isequal('deembed OSTD',mode)
                title({'|Z| vs. Freq';'Open, Short, Thru De-embedding (Narrow vs. Wide
Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'|Z| vs. Freq';'Raw vs. ALL De-embedding (Narrow Wide Wire)';})
            end
        end
        % xlim([41433053.31022 22598605075.35646])
        hold on

    figure(4);
        set(gcf, 'color', 'white');
        set(findobj('type','axes'),'FontSize',20)
        if isequal('raw',mode)
            semilogx(freq(y),theta(y)*(180/pi),col);
        elseif isequal('deembed',mode)| isequal('deembed OPD',mode)| isequal('deembed
OSD',mode)|isequal('deembed RAWOSTD',mode)

semilogx(freq_D(y),theta(y)*(180/pi),col,freq_D(y),theta_D(y)*(180/pi),col_D);
        elseif isequal('deembed ALL',mode)

semilogx(freq_D(y),theta_D(y)*(180/pi),col_D,freq_D(y),theta_DO(y)*(180/pi),col_DO,freq_D
(y),theta_DOS(y)*(180/pi),col_DOS,freq_D(y),theta_DOST(y)*(180/pi),col_DOST);
        elseif isequal('deembed OSTD',mode);
            semilogx(freq_D(y),theta_D(y)*(180/pi),col_D);
        end
        xlabel('Freq (Hz)');
        ylabel('Z Angle (Degrees)');
        %set(gca,'YTick',-180:90:180)
        %set(gca,'YTickLabel',{'180','90','45','0','45','90','180'})
        if isequal('narrow',titleheading)
            if isequal('raw',mode)
                title({'Z Angle vs. Freq';'Raw (Narrow Wire)'})
            elseif isequal('deembed OPD',mode)
                title({'Z Angle vs. Freq';'Open De-embedding (Narrow Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'Z Angle vs. Freq';'Open, Short De-embedding (Narrow Wire)'})
            elseif isequal('deembed RAWOSTD',mode)
```

```
                title({'Z Angle vs. Freq';'Raw vs. Open, Short, Thru De-embedding (Narrow
Wire)'})
            elseif isequal('deembed OSTD',mode)
                title({'Z Angle vs. Freq';'Open, Short, Thru De-embedding (Narrow
Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'Z Angle vs. Freq';'Raw vs. ALL De-embedding (Narrow Wire)'})
            end
        elseif isequal('wide',titleheading)
            if isequal('raw',mode)
                title({'Z Angle vs. Freq';'Raw (Wide Wire)'})
            elseif isequal('deembed OPD',mode)
                title({'Z Angle vs. Freq';'Open De-embedding (Wide Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'Z Angle vs. Freq';'Open, Short De-embedding (Wide Wire)'})
            elseif isequal('deembed_m_a_x RAWOSTD',mode)
                title({'Z Angle vs. Freq';'Raw vs. Open, Short, Thru De-embedding (Wide
Wire)'})
            elseif isequal('deembed OSTD',mode)
                title({'Z Angle vs. Freq';'Open, Short, Thru De-embedding (Wide Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'Z Angle vs. Freq';'Raw vs. ALL De-embedding (Wide Wire)'})
            end
        elseif isequal('narrowwide',titleheading)
            if isequal('raw',mode)
                title({'Z Angle vs. Freq';'Raw (Narrow vs. Wide Wire)'})
            elseif isequal('deembed OPD',mode)
                title({'Z Angle vs. Freq';'Open De-embedding (Narrow vs. Wide Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'Z Angle vs. Freq';'Open, Short De-embedding (Narrow vs. Wide
Wire)'})
            elseif isequal('deembed RAWOSTD',mode)
                title({'Z Angle vs. Freq';'Raw vs. Open, Short, Thru De-embedding (Narrow
vs. Wide Wire)'})
            elseif isequal('deembed OSTD',mode)
                title({'Z Angle vs. Freq';'Open, Short, Thru De-embedding (Narrow vs.
Wide Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'Z Angle vs. Freq';'Raw vs. ALL De-embedding (Narrow vs. Wide
Wire)'})
            end
        elseif isequal('narrow',titleheading)
            if isequal('raw',mode)
                title({'Z Angle vs. Freq';'Raw (Narrow Wire)'})
            elseif isequal('deembed OPD',mode)
                title({'Z Angle vs. Freq';'Open De-embedding (Narrow Wire)'})
            elseif isequal('deembed OSD',mode)
                title({'Z Angle vs. Freq';'Open, Short De-embedding (Narrow Wide Wire)'})
            elseif isequal('deembed RAWOSTD',mode)
                title({'Z Angle vs. Freq';'Raw vs. Open, Short, Thru De-embedding (Narrow
vs. Wide Wire)'})
            elseif isequal('deembed OSTD',mode)
                title({'Z Angle vs. Freq';'Open, Short, Thru De-embedding (Narrow vs.
Wide Wire)'})
            elseif isequal('deembed ALL',mode)
                title({'Z Angle vs. Freq';'Raw vs. ALL De-embedding (Narrow Wide Wire)'})
            end
        end
        %xlim([41433053.31022 22598605075.35646])
        hold on
```

```
elseif isequal('other',mode)
%setup max Q array

Freq_W = [7.98e8,3.47e8,2.63e8,1.21e8,7.75e7,7.33e7]
Freq_N = [1.11e9,5.12e8,3.11e8,1.99e8]

N_W = [1,3,5,10,15,20]
N_N = [1,3,5,10]

Qmax_W = [11.25,9.75,8.1,5.764,5.082,4.25]
Qmax_N = [7.841,4.718,3.786,2.277]

L_W = [0.96,4.017,7.951,18.83,29.78,43.19]
L_N = [1.504,4.848,8.608,18.39]

Zmag_W = [4.833,8.803,13.23,14.5,14.77,20.43]
Zmag_N = [10.62,15.98,17.37,25.13]

Zang_W = [84.92,84.14,82.96,80.16,78.87,76.76]
Zang_N = [82.39,78.03,74.37,66.29]

Fsrmax_W = [4.224e9,1.554e9,8.919e8,4.099e8,2.937e8,2.225e8]
Fsrmax_N = [4.465e9,1.737e9,9.967e8,4.58e8]

figure(5)
        set(gcf, 'color', 'white');
       plot(N_W,Qmax_W,'-sb',N_N,Qmax_N,'-sr');
       title({'Q_m_a_x vs. N_T_u_r_n_s';'Open, Short, Thru De-embedding (Wide vs. Narrow
Wire)'})
        xlabel('N_T_u_r_n_s');
        ylabel('Q_M_a_x');

figure(6)
        set(gcf, 'color', 'white');
       semilogx(Freq_W,Qmax_W,'-sb',Freq_N,Qmax_N,'-sr');
       title({'Q_m_a_x vs. f_o';'Open, Short, Thru De-embedding (Wide vs. Narrow Wire)'})
       xlabel('f_o (Hz)');
       ylabel('Q_m_a_x');

figure(7)
        set(gcf, 'color', 'white');
       % calculate C at fo
       %CW1 = 2*pi*Freq_W
       %for i=1:length(CW1)
       %      CW11 = CW1(i)*CW1(i)
       %      C_W_fo(i) = 1/(CW1(i)*L_W(i))
       %      C_W_fo(i) = C_W_fo(i)*1E12
       %end

       %CN2 = 2*pi*Freq_N
       %for i=1:length(CN2)
       %       CN11 = CN2(i)*CN2(i)
       %       C_N_fo(i) = 1/(CN2(i)*L_N(i))
       %       C_N_fo(i) = C_N_fo(i)*1E12
       %end
       %semilogx(Freq_W,L_W,'-sb',Freq_N,L_N,'-sr',Freq_W,C_W_fo,'-sb',Freq_N,C_N_fo,'-
sr');

       semilogx(Freq_W,L_W,'-sb',Freq_N,L_N,'-sr');
       title({'L  vs. f_o';'Open, Short, Thru De-embedding (Wide vs. Narrow Wire)'})
```

```
       xlabel('f_o (Hz)');
       ylabel('L (nH)');

figure(8)
        set(gcf, 'color', 'white');
       plot(N_W,Zmag_W,'-sb',N_N,Zmag_N,'-sr');
       title({'|Z| at f_o vs. N_T_u_r_n_s';'Open, Short, Thru De-embedding (Wide vs.
Narrow Wire)'})
       xlabel('N_T_u_r_n_s');
       ylabel('|Z| at f_o (ohms)');

figure(9)
        set(gcf, 'color', 'white');
       semilogx(Freq_W,Zmag_W,'-sb',Freq_N,Zmag_N,'-sr');
       title({'|Z| vs. f_o';'Open, Short, Thru De-embedding (Wide vs. Narrow Wire)'})
       xlabel('f_o (Hz)');
       ylabel('|Z| (ohms)');

figure(10)
        set(gcf, 'color', 'white');
       plot(N_W,Zang_W,'-sb',N_N,Zang_N,'-sr');
       title({'Z Angle at f_o  vs. N_T_u_r_n_s';'Open, Short, Thru De-embedding (Wide vs.
Narrow Wire)'})
       xlabel('N_T_u_r_n_s');
       ylabel('Z Angle at f_o (degrees)');

figure(11)
        set(gcf, 'color', 'white');
       Z_phase_margin_W = abs(Zang_W-90)
       Z_phase_margin_N = abs(Zang_N-90)
       plot(N_W,Z_phase_margin_W,'-sb',N_N,Z_phase_margin_N,'-sr');
       title({'Phase Margin at f_o vs. N_T_u_r_n_s';'Open, Short, Thru De-embedding (Wide
vs. Narrow Wire)'})
       ylabel('Phase Margin (degrees)');
       xlabel('N_T_u_r_n_s');

       %semilogx(Freq_W,Zang_W,'-sb',Freq_N,Zang_N,'-sr');
       %title({'Z Angle vs. f_o';'Open, Short, Thru De-embedding (Wide vs. Narrow
Wire)'})
       %xlabel('f_o (Hz)');
       %ylabel('Z Angle (degrees)');

figure(12)
        set(gcf, 'color', 'white');
       plot(N_W,L_W,'-sb',N_N,L_N,'-sr');
       title({'L at f_o  vs. N_T_u_r_n_s';'Open, Short, Thru De-embedding (Wide vs.
Narrow Wire)'})
       xlabel('N_T_u_r_n_s');
       ylabel('L at f_o (nH)');

figure(13)
        set(gcf, 'color', 'white');
       semilogx(Fsrmax_W,N_W,'-sb',Fsrmax_N,N_N,'-sr');
       title({'f_s_r vs. N_T_u_r_n_s';'Open, Short, Thru De-embedding (Wide vs. Narrow
Wire)'})
       xlabel('f_s_r (Hz)');
       ylabel('N_T_u_r_n_s');

figure(14)
        set(gcf, 'color', 'white');
       %Q1=Q1/2
```

```
       %QT= Q+Q1
       %semilogx(freq(y),Q(y),'-sb',freq(y),Q1(y),col,freq(y),QT,'-sr')
       title({'Dual Q-Peaking vs. Freq';'Raw (Wide Wire)'})
       xlabel('Q (Hz)');
       ylabel('Freq (Hz)');

figure(15)
        set(gcf, 'color', 'white');
       plot(Zmag_W,Qmax_W,'-sb',Zmag_N,Qmax_N,'-sr')
       title({'Q_m_a_x vs. |Z|';'Open, Short, Thru De-embedding (Wide vs. Narrow )'})
       xlabel('Q_m_a_x ');
       ylabel('|Z| (Ohms)');

       %plot(Qmax_W,Zmag_W,'-sb',Qmax_N,Zmag_N,'-sr')
       %t%tle({'|Z| vs. Q_m_a_x';'Open, Short, Thru De-embedding (Wide vs. Narrow )'})
       %x%abel('|Z| (Ohms)');
       %ylabel('Q_m_a_x ');

leg=[]
h_legend=legend(leg);
figure(5)
     legend('Wide Wire OSTD', 'Narrow Wire OSTD');
     grid on
    set(h_legend, 'FontSize', 18)

 figure(6)
     legend('Wide Wire OSTD', 'Narrow Wire OSTD');
    grid on
     set(h_legend, 'FontSize', 18)

 figure(7)
     legend('Wide Wire OSTD', 'Narrow Wire OSTD');
    grid on
     set(h_legend, 'FontSize', 18)

figure(8)
     legend('Wide Wire OSTD', 'Narrow Wire OSTD');
    grid on
     set(h_legend, 'FontSize', 18)

figure(9)
     legend('Wide Wire OSTD', 'Narrow Wire OSTD');
    grid on
    set(h_legend, 'FontSize', 18)

    figure(10)
     legend('Wide Wire OSTD', 'Narrow Wire OSTD');
    grid on
    set(h_legend, 'FontSize', 18)

figure(11)
     legend('Wide Wire OSTD', 'Narrow Wire OSTD');
    grid on
    set(h_legend, 'FontSize', 18)

figure(12)
     legend('Wide Wire OSTD', 'Narrow Wire OSTD');
    grid on
    set(h_legend, 'FontSize', 18)

figure(13)
```

```
     legend('Wide Wire OSTD', 'Narrow Wire OSTD');
    grid on
    set(h_legend, 'FontSize', 18)

figure(14)
     legend('Wide Wire OSTD', 'Narrow Wire OSTD');
    grid on
    set(h_legend, 'FontSize', 18)

figure(15)
     legend('Wide Wire OSTD', 'Narrow Wire OSTD');
    grid on
    set(h_legend, 'FontSize', 18)
  end
end
%Setup the array for the legend
leg=[]
if  isequal('deembed OPD',mode) | isequal('deembed OSD',mode)  | isequal('deembed',mode)|
isequal('deembed RAWOSTD',mode)
    leg=[];
    for j = 1:numinductors;
        leg=[leg,strcat(l(1,j),'\_RAW'),l_D(1,j)]; %BOTH
    end
elseif isequal('deembed ALL',mode)
    leg=[];
    for j = 1:numinductors;
        leg=[leg,strcat(l_DA(1,j),'\_RAW'),l_DO(1,j),l_DOS(1,j),l_DOST(1,j)];
    end
elseif isequal('raw',mode)
    leg=[];
    for j = 1:numinductors;
        leg=[leg,strcat(l(1,j),'\_RAW')];
    end
elseif isequal('deembed OSTD',mode)
    for j = 1:numinductors;
        leg=[leg,l_D(1,j)];
    end
end

% Insert the legend into each figure
figure(1)
    h_legend=legend(leg);
    set(h_legend, 'FontSize', 18)
         grid on

figure(2)
    h_legend=legend(leg);
    set(h_legend, 'FontSize', 18)
         grid on

figure(3)
        h_legend=legend(leg);
    set(h_legend, 'FontSize', 18)
         grid on

figure(4)
        h_legend=legend(leg);
    set(h_legend, 'FontSize', 18)
     grid on

OK = 1;
```