Inductor Modeling and Optimization for Integrated Circuits

Alia Briglia | aliabriglia31@icloud.com | ACME Lab
Girls Academic Leadership Academy, Class of 2025
USC Viterbi Department of Electrical Engineering, SHINE 2023

Introduction

Inductors are radio-frequency structures that can be used as electrical components that generate magnetic fields. They can be integrated directly into CMOS integrated circuits and used as low-cost, highly-sensitive, and ultraportable biosensors, and also find applications in magnetic communications, wireless power transfer, and localization.

Due to the lack of analytical solutions to Maxwell's equations, inductors must be simulated numerically. If a simulation doesn't yield the desired results, it has to be slightly tweaked and run again. This process can be very tedious and time-consuming.

The goal of my project was to develop techniques for optimizing an inductor's geometry given desired performance metrics, without requiring iterated field simulations to obtain an optimal result.

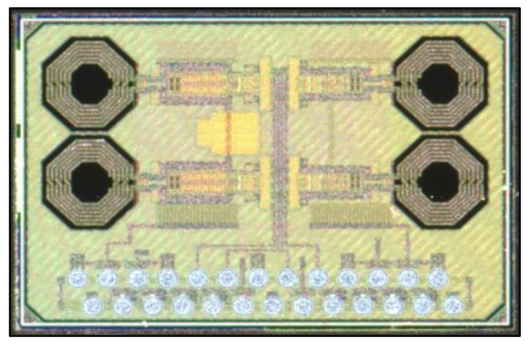

Fig 1: CMOS integrated circuit (J. Sun ESSCIRC 2022)

Fig 2: Inductor in a guitar pedal (PC: Alia Briglia)

Objective & Impact of Professor's Research

The Analog/RF Integrated Circuits, Microsystems, and Electromagnetics (ACME) Lab broadly encompasses two areas of research: analog integrated circuits for biomedical applications and modeling and optimization of electromagnetic devices.

- The analog and RF integrated circuits are used for point-of-care biosensors because they can be highly sensitive and portable.
- The development of computational techniques to simulate Maxwell's equations more efficiently enables the design of high-performance RF structures ranging from waveguide couplers to antennas.

My project contributes to Dr. Sideris' work because it focuses on optimizing electromagnetic devices. This could help with both areas of Dr. Sideris' research. The model I made can predict the performance of inductors without requiring lengthy EM simulations during design.

Methods

The open-source finite-difference time domain (FDTD) solver OpenEMS was used for field simulations of different inductor geometries. For simplicity, single-turn planar inductors appropriate for implementation in a modern CMOS process were considered. A lumped port excitation with a gaussian source and a PEC ground were used. From this simulation we also obtained the inductance (L), series resistance (R_s), and quality factor (Q) values for the inductor.

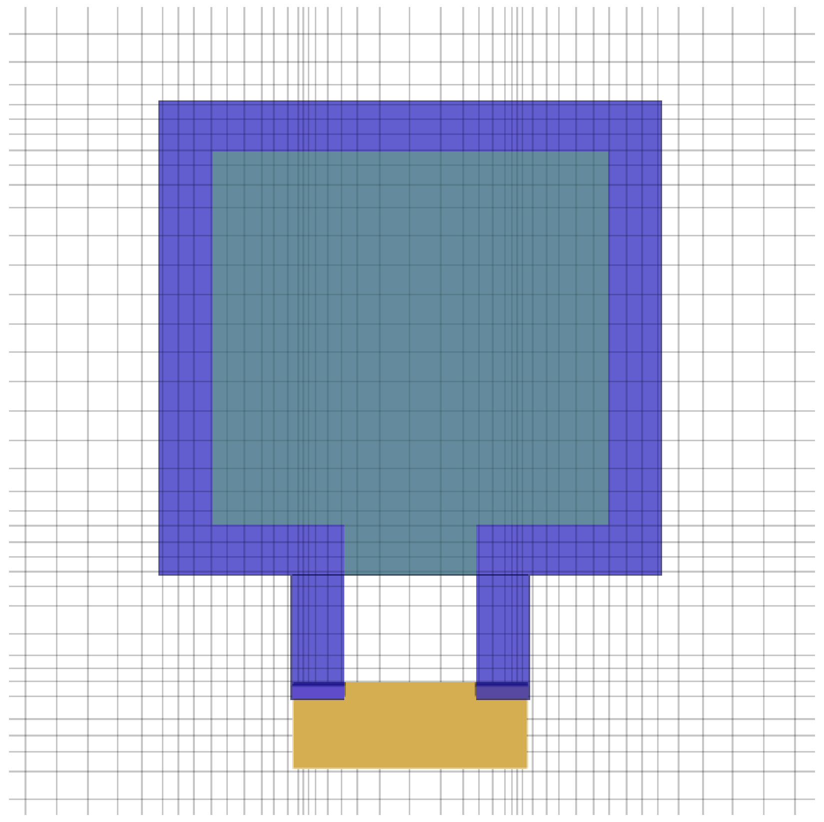

Fig 3: Inductor simulation in OpenEMS (diameter: 150, width: 20)

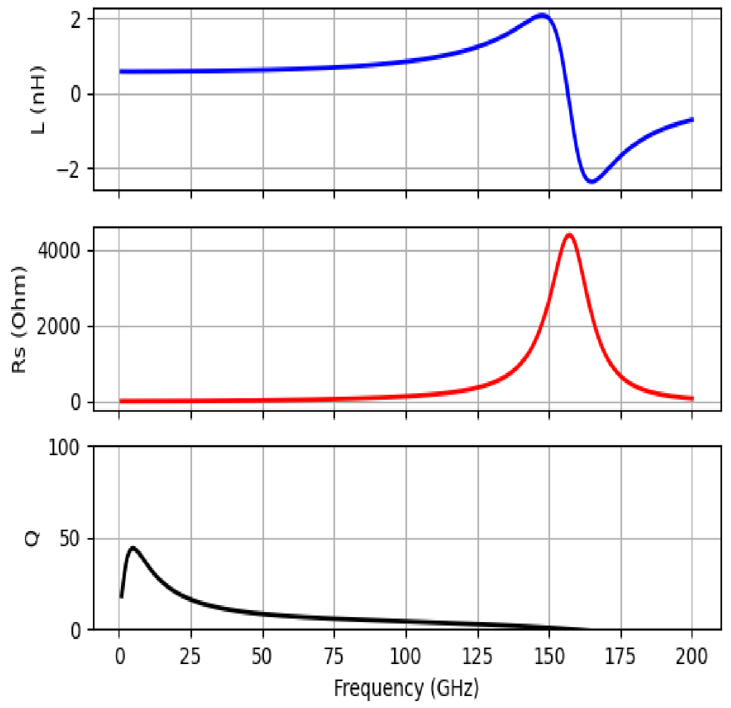

Fig 4: Graphs of L, Rs, and Q values (diameter: 150, width: 20)

In the optimization, the L, R_s, and Q were used as performance metrics. For a given inductor, these quantities may be computed from the real and imaginary parts of the input impedance Z_{in}.

$$Z_{\text{in}} = \text{Re}(Z_{\text{in}}) + j\text{Im}(Z_{\text{in}}) = R_s + j\omega L$$

$$L = \frac{\text{Im}(Z_{\text{in}})}{\omega}$$

$$R_s = \text{Re}(Z_{\text{in}})$$

$$Q = \frac{\omega L}{R_s} = \frac{\text{Im}(Z_{\text{in}})}{\text{Re}(Z_{\text{in}})}$$

Fig 5: Equations for inductor properties

The next step was to create a Python script that could generate multiple diameters and widths for the inductors to create a data set for the machine learning models to train and practice on. To reduce simulation runtime, the bandwidth of the source was reduced around the frequency of interest.

Neural networks were constructed using PyTorch. The networks use an optimization algorithm, so they can maximize certain values and minimize others. I was unable to use this algorithm to its full potential, but with more time, it could make the networks even better.

Two models were trained, one that uses the inductor geometry as input and predicts performance (forward model), and one that uses the performance metrics as input and predicts the corresponding geometry (backward model). The neural networks used have one hidden layer and the ReLu activation function was used on each layer.

Results

For both models, the loss function used during training was the Mean Squared Error (MSE).

To evaluate the accuracy of the model, I used an OpenEMS script to obtain the L, R_s, and Q values for an inductor with a "diameter" of 150 and a width of 20. I then inputted the L, R_s, and Q values into the backward model, to predict the diameter and width of the inductor.

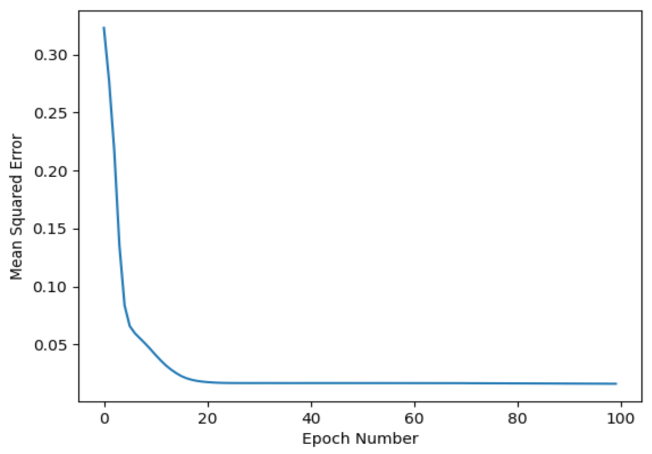

Fig 6: MSE graph for the backward model

Table 1

Table of "diameters" and widths generated by the model

Run	"Diameter" (um)	Width (um)
openEMS	150	20
1	152.89	21.07
2	149.89	20.45
3	150.83	19.07

The machine learning model generated results for the inductors' "diameter" and width. This data, is quite accurate and demonstrates that the models can be used to predict the geometry for an inductor with any desired performance metrics.

Skills Learned

- Coding in Python
- Using OpenEMS (Open source Finite-Difference Time Domain (FDTD) simulator)
- Understanding and coding machine learning algorithms
- How to debug and troubleshoot issues in code
- General knowledge of electromagnetics and physics
- Soldering
- Preparing for college applications
- Solving mathematical puzzles

Next steps for Research

- Develop optimization to use the models to generate optimal inductor geometries subject to area constraints
- Rewrite OpenEMS script to accommodate inductors with more than one turn
- Retrain model with a data set that includes multi-turn inductors

Acknowledgements

I would like to thank Professor Sideris and Ray Sun for helping me with my research. I would also like to thank my parents for supporting my interests in math and engineering.