Design of Spectral Filtering Wireless Transmitters

Bonjern Yang

Electrical Engineering and Computer Sciences
University of California, Berkeley

Technical Report No. UCB/EECS-2021-227
http://www2.eecs.berkeley.edu/Pubs/TechRpts/2021/EECS-2021-227.html

December 1, 2021

Design of Spectral Filtering Wireless Transmitters

by

Bonjern Yang

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering - Electrical Engineering and Computer Sciences

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Elad Alon, Chair
Professor Borivoje Nikolic
Professor Martin White

Fall 2019

Design of Spectral Filtering Wireless Transmitters

Abstract

Design of Spectral Filtering Wireless Transmitters

by

Bonjern Yang

Doctor of Philosophy in Engineering - Electrical Engineering and Computer Sciences

University of California, Berkeley

Professor Elad Alon, Chair

A frequency-flexible radio-frequency (RF) front end has long been desired, but faces a myriad of obstacles to its realization. In recent years, the use of switching power amplifiers (PA) as part of digital PAs and RF digital-to-analog converters (RFDACs) has become more common. The primary motivation of these RFDACs is to directly convert from digital baseband bits to RF output. This is useful in the realization of a frequency-flexible RF front end, but this is prevented by the generation of significant spectral emissions in the form of harmonics and quantization noise by RFDACs. These issues are both typically remedied with the usage of high-order fixed filters which are inherently not frequency flexible.

In this dissertation, we will discuss an approach to implement a frequency-flexible digital PA-based transmitter using programmable integrated filtering to suppress spectral emissions without the use of external filters. Two filtering techniques will be discussed, as well as their requirements and limitations. Additionally, design automation of core blocks within the transmitters using the Berkeley Analog Generator (BAG) framework will be discussed in detail. We will demonstrate two prototypes implemented in 65nm and 28nm processes achieving state of the art filtering performance at a peak power level of > 23 dBm across at least a 1 GHz - 2 GHz frequency range.

To my parents

Contents

List of Figures

List of Tables

Acknowledgments

Graduate school has been a longer and more arduous journey than I could have ever imagined. It's been a thoroughly humbling experience constantly interacting with extremely intelligent and hardworking people on a regular basis, but the same people have pushed me to grow both as a researcher and a person. I'd like to take this space to convey my gratitude towards all these people and those who supported me throughout my graduate studies.

First, I would like to thank my advisor Elad Alon. My path towards conducting research began when I first approached Elad during office hours to ask for research work as an undergraduate. I didn't think at the time that it would lead to years of working together. I'm always impressed with how quickly you can assess a problem that you've never seen before and come up with a reasonable solution, or at least something that would lead to a solution. There were at least a few times where I hit a wall in research which was resolved by one of your suggestions. Lastly, I'd like to thank you for being encouraging and keeping my spirits up even when I felt like I wasn't making meaningful progress. After writing this dissertation, I can definitely look back and be proud of what I accomplished over these last 7.5 years, largely in part to the research guidance you've given me.

Borivoje Nikolic and Ali Niknejad both provided me invaluable advice both in research topics and more broadly for my future career, as well as serving on my quals committee. I'd especially like to thank Bora for providing extra advising when I really needed it, especially in the final stretch of my PhD, and for serving on my dissertation committee as well. Both professors gave me valuable alternative perspectives, pushing me to become a more well-rounded researcher. I would also like to thank Professor Martin White for serving on my quals and dissertation committees.

Next, I'd like to acknowledge the people who've contributed most directly to the chips that I've worked on, Nai-Chung Kuo and Eric Chang. Nai-Chung is simultaneously one the hardest working and kindest people I've met in graduate school. The fact that he is always willing to help others even when he is swamped with his own work is truly admirable and reflects upon his character. I appreciate all the discussions and help he's given me on various RF topics and measurement methodologies even long after our collaboration on the first RFFPGA chip.

Eric is the person I've met who most closely embodies that mythical engineer who does 10 times the work of the average engineer. The first version of this transmitter would have been significantly delayed without your contributions. The story that best encapsulates working with you was how after one day of using the existing measurement framework for testing the first chip, you went home and completely rewrote a much better framework in a single night. It was so useful I even continued to use to test the second version of the transmitter. Thanks for fixing all of my issues with BAG in a speedy fashion, and for all the random trivia you've dispensed upon me throughout the years.

I want to thank Andrew, Lorenzo, Emily, Meng, Angie, and Charles for their help in tapeouts either in reusing their blocks, measurement setup, or overall general aid. Special thanks to Andrew for helping me, along with many other graduate students, in the deep,

dark hours leading to the tapeout deadline, whether it was through helping to run tools or providing another set of eyes to debug last minute issues.

I would not have made it through graduate school without the friends I've made along the way. First, I want to thank Sameet and Phil for being loyal friends from undergraduate through graduate school. I always could rely on you two to have discussions about anything, whether it be technical discussion or rambling about life struggles. The countless hours of talking and laughing about incredibly inane topics are irreplaceable memories that took the edge off of the stress of graduate school. I know there's been a lull recently, but I look forward to years of continued friendship.

Nathan, I couldn't have asked for a better cubicle neighbor. Not only have you helped me out countless times with research, you've been a great friend who bore all my venting. Even through your own struggles, I could rely on you to help me out in times of need, and I hope I can return the favor in the future.

Ozzy, thanks for being a great friend and my designated basketball (watching) buddy. Our technical discussions throughout the years has been extremely helpful, and I am indebted to you for your help on the second version of the transmitter. Thanks for all the moral support, and dealing with (or enjoying) my random rants. I appreciate the weirdness you bring to BWRC and into my life, even if I don't participate in your schemes often.

Luke, you are an odd person in the best possible way. It's been great discussing music with you throughout the years, and it's definitely helped to push the boundaries of what I enjoy, even if not everything you suggest clicks. Your out of the box thinking rubbed off on me just a little bit, making me a little more open to very outwardly outlandish ideas.

Thanks to the Antonio, John, Emily, Keertana, Krishna, and George for being part of the lunch squad and contributing to a variety of odd lunch discussions. Being able to lighten things up everyday kept me chugging along. I'd like to thank my BWRC colleagues Ali Moin, Nima, Sashank, Pengpeng, Seobin, Zhongkai, Yi-An, Filip, Nick, Sean, Kourosh, Ayan, and Bob for various technical discussions and their friendship.

I may not have even been on this path if it wasn't for my graduate student mentors Lingkai Kong and John Crossley. Thank you for taking a confused undergrad under your wing and nurturing my interest in research, eventually leading me to continue into graduate school. I am thankful for all the advice in both research and navigating graduate school in general. Lingkai, I'm glad I listened to your advice all those years back to stick with BAG - even though I've never been a primary developer, it's been amazing to see it grow and move closer to the lofty initial ideal of what it could be. I would also like to thank Hanh-Phuc, Alberto, Matt Spencer, Yue, Yida, and Chintan for being welcoming and helpful seniors in the EEIS group during my undergraduate and early PhD years.

I want to thank the BWRC staff for doing an incredible amount of work behind the scenes to ensure that students can focus on research. In particular, the support of Candy, Ajith, and James have been absolutely critical to me finishing my graduate studies, whether it came in the form of pushing and organizing tapeouts, helping with tech and lab issues right before a major deadline, or miscellaneous issues in the case of Candy. I think the three of you have looked out for the students far beyond what anyone could reasonably expect. I'd

also like to thank Fred, Sarah, Olivia, Yessica, Mikaela, Bira, Brian, and Greg for helping to keep BWRC afloat.

I'd like to thank Aritra Banerjee for the helpful discussions regarding cellular transmitters, as well as providing the LTE data I've used for measurements of both chips. The insight from someone currently in industry was extremely valuable, and gave me a sense of validation of my research work. Also on the industry side, I'd like to thank Simone Gambini for mentoring me during my internship at Apple. Thanks for making my first experience working a pleasant one, and for giving an idea of what to expect working in industry.

My graduate studies would not have been possible without the funding provided by DARPA RFFPGA, DARPA CRAFT, and the GAANN fellowship.

I want to thank two friends outside of BWRC who were constants in my life, and helped me through my toughest times in graduate school. Ryan, we've known each other and lived together for so many years that you're essentially family to me. Those who know me know how seriously I value family, and how much weight that statement carries. Despite our divergent paths in life, I always appreciated how we stay connected based on our starting points. Your activism in undergrad and your ongoing focus on helping underserved communities inspired me to learn more about my own roots. I wish you luck in completing your PhD, and hope you'll be back in California soon.

Steven, you're one of my oldest and most loyal friends. Thank you for constantly taking trips to come hang out throughout the years. I know I don't make enough of an effort to stay connected with friends, so thank you for bearing with that and putting in the work to keep our friendship strong. The most extreme example of this when you flew to Japan to sit in on my VLSI talk to show your support. You did this without being prompted - if that's not true friendship, I don't know what is.

I want to thank my extended family and my brothers for their support. Wilton, thanks for always being my rival in fighting games, and being my equal, even though I'm the older brother. Thanks for taking care of the family the last few years - hopefully soon I can step in and help out. Hans, I'm always impressed with how sharp your wit is. I'm excited to read your future works, whether it be in the form of a novel or a script. Hudson, it's been great seeing you grow into a passionate and principled young man. I know you're gonna reach your dream of making amazing video games. The three of you drive me to be the best person I can be - I have to set the standard high as the eldest brother.

None of this would have been possible without the support of my parents. Despite having a tough life, you fought through it to carve out a decent life for us. Growing up poor wasn't easy, but you did everything you could to make it feel alright. Thanks Dad, for being my role model growing up. You worked hard every day for the sake of our family so that we could live a better life. Thanks Mom, for dealing with four lazy boys. I don't know how you didn't go crazy dealing with our antics, but you did it so we could focus on education for the final goal of living a more comfortable life than you did. I know you wished you could have completed your education, but I hope my PhD will be an adequate substitute. It would have been impossible to get through graduate school without the constant and unwavering love and support from the both of you. I hope I've made you proud.

Chapter 1

Introduction

The current cellular landscape of the world consists of a plethora of coexisting standards with varying bandwidths, center frequencies, and specifications. Standards from different generations such as 3G (GSM) and 4G (LTE) exist within the same regions despite starkly different requirements in output power and channel bandwidth. Even within a single standard such as long term evolution (LTE), frequency bands of operation can vary from country to country. For example, LTE band B2, centered at 1900 MHz, is used in the USA and not the UK, while LTE band B7, centered at 2600 MHz, is used in the UK and not the USA [1]. This means that the same product will be designed with different hardware for different parts of the world, increasing engineering costs. The ubiquity of smart phones and the powerful functionality they provide also requires the inclusion of hardware supporting non-cellular standards such as WiFi and Bluetooth.

It makes sense to have separate radios for cellular, WiFi, and Bluetooth so they can operate simultaneously. However, there may even be multiple cellular radios for extra coverage and functionality, such as if the phone is intended to support multiple standards like CDMA and LTE. All of these additional radios increase the overall cost of the system, since separate hardware will be required to implement a radio compliant with each standard.

Each of these radios include a transmitter which will generate spectral emissions, primarily harmonics and quantization noise, which must be suppressed in order to meet spectrum mask requirements. This is typically done using a fixed, high-order passive filter, such as surface acoustic wave (SAW), bulk acoustic wave (BAW), or film bulk acoustic resonator (FBAR). These filters have very good out of band rejection, but are physically bulky and add insertion loss. These filters are relatively large, comparable in size to the TX integrated circuits (IC). Two examples covering different LTE bands measure 3 mm x 3 mm x 1.02 mm [2] and 1.1 mm x 1.4 mm x 0.85 mm [3]. These also add 2.0 dB and 2.3 dB of insertion loss respectively. Since these filters are not configurable, each new band will require a filter in addition to the transmitter ICs, increasing the cost and complexity of the product.

The goal of DARPA's RFFPGA program was to propose commercial and government solutions which would tackle the issue of many different standards by implementing a single design which could be reconfigured to meet different standards. Our approach to achieving

this configurability is to use a wideband digital PA combined with integrated, programmable filtering techniques to reduce harmonics and quantization noise. The purpose of the wideband PA is to allow for operation across a wide frequency range using a single TX. An integrated solution to the spectral issues is needed to avoid the bulky, off-chip filters. We will define a digital PA and its benefits, as well as their drawbacks in generating strong harmonics and quantization noise.

Digital PAs have become increasingly attractive in recent years due to advances in semiconductor processes, with many works demonstrating competitive results [4][5][6][7][8][9]. The core of a digital PA is a switching PA topology, which requires a square wave input with relatively sharp edges, or short rise and fall times. Slow edge rates will degrade efficiency or even fundamentally alter operation in the extreme case. Improved transistors in modern CMOS processes can easily generate these square waves at cellular frequencies, enabling the operation of efficient digital PAs. Transmit data is generally digitized for ease of use, allowing for easy modification of input levels as well as bandwidth. Digital PAs also combine DAC and PA functionality into a single block, directly converting from digital bits to RF output, while systems with linear PAs require a separate DAC.

However, these digital PAs have two major issues when it comes to spectral emissions: harmonics and quantization noise. Digital PAs generate very strong harmonics due to fundamentally operating as switching PAs. Switching PAs typically operate by generating square waves either in voltage or current which drives a tuned load to achieve a sinusoidal output. A square wave $x(t)$ with amplitude α can be written as a sum of harmonically related sinusoids with Fourier series decomposition (Eq. 1.1). The k^{th} harmonic in a square wave only falls off as $1/k$ in voltage relative to the fundamental, which places strict requirements on the output network. For example, without any output filtering, the 3^{rd} harmonic is only 9.5 dB lower than the fundamental.

$$x(t) = \frac{\alpha}{j\pi} \sum_{|k| \ odd}^{\infty} \frac{e^{jk\omega_0 t}}{k} = \frac{2\alpha}{\pi} \sum_{k=1,3,5...}^{\infty} \frac{sin(k\omega_0 t)}{k} \tag{1.1}$$

This is a major issue in our case since we cannot rely on the output network alone to suppress the harmonics since we are opting to implement a wideband PA. A different and programmable solution is required to suppress these harmonics.

Quantization noise in digital PAs is generated due to DAC behavior, and is a consequence of the input data being digital. This quantization noise is an issue for nearby out-of-band channels, and could potentially desensitize receivers in these bands in extreme cases. The plot in Fig. 1.1 shows a case in which the quantization noise from a TX can bleed into a receiver (RX) channel spaced 40 MHz away. The quantization noise floor can be lowered either by adding additional bits to the digital PA or using a larger oversampling ratio, but this is very expensive from a power perspective. If we know what frequency band a nearby receiver may be operating in, it can be more energy efficient to reduce the quantization noise only in that band.

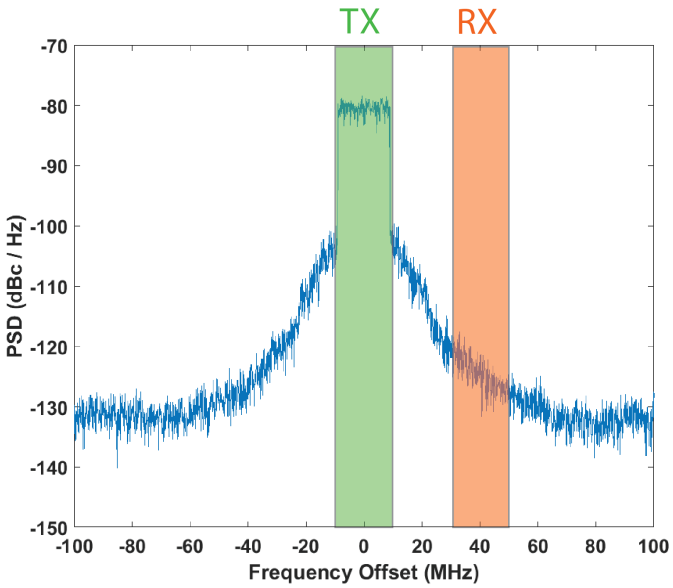


Figure 1.1: Effects of TX quantization noise on nearby channels

Our solution to these two problems of harmonics and quantization noise is to use filtering techniques which divide the PA into smaller pieces, driving each piece with different input signals, and combines the pieces at the output. Depending how the input signals are modified, different parts of the spectrum can be cancelled. We will implement two types of filtering using this general technique, which we will call harmonic cancellation and mixed-signal filtering. These notch out the harmonics and specific bands of nearby spectrum, respectively. In particular, we will focus on the third harmonic since it will be the largest harmonic in our differential TX. Both filtering techniques are frequency flexible, meaning they are able to operate across a wide frequency range, and are programmable on chip. These do not depend on off-chip components to operate, and all our prototypes do not use external components in the output network.

Another reason digital PAs were chosen were due to how technology scaling-friendly they are. These advanced processes give us transistors with better switching performance, but come with the drawback of rapidly growing design complexity. In addition to this, a great deal of effort is expended porting designs from one process to another. Design automation is critical in helping to manage this growing design complexity, which is already commonplace in the case of large scale digital design [10]. This is much less common with regards to analog, mixed-signal, and RF design, with much design and layout still being done completely manually. Several attempts have been made at trying to automate analog design [11][12][13], but in our case we felt the generator-based approach of Berkeley Analog Generator (BAG) [14][15] fit our needs best. Originally, this framework was designed with the goal of process portability in mind, which was further refined in later versions.

BAG is not intended to be analog synthesis, but is a framework in which users can

codify their designs into schematic and layout generators written in the Python programming language. Once complete, these generators take in user parameters to generate specific instances of these circuits. These generators have been used as key parts of fabricated prototypes, including serial links [16][17] and RF transceivers [18] across 16 nm and 65 nm processes. The latter is the first RFFPGA chip prototype which had the PA layout generated by BAG. This first chip is not the focus of this work, but we will give background on it to motivate the use of design automation in this work.

The initial approach for the first RFFPGA system was to create a transceiver capable of operating from 800 MHz - 6 GHz. This was to be done by splitting this into three bands of 800 MHz - 1.5 GHz, 1.5 GHz - 3 GHz, and 3 GHz - 6 GHz, with each band covered by a wideband PA. These would then be connected to an interposer containing a transformer for each frequency band (1.2). Optimizing each PA for each band would require significant effort, so the approach we took was to build a single BAG generator and supply it with three sets of parameters to generate a different instance for each band. Writing a generator is more effort than manually generating a single instance of a PA, but the generator saves time and effort once multiple differing instances are required, such as in this case. In the end, the first RFFPGA system [18] consisted of only a single chip used for multiple bands instead of multiple chips with different PA designs, but a fully functioning class D^{-1} PA generator was implemented and used to create the layout of the PA used in this single chip.

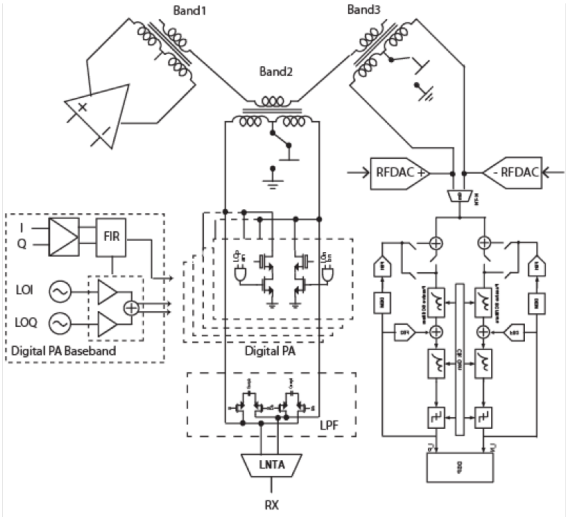


Figure 1.2: Conceptual RFFPGA top level block diagram

For this work, our primary reason for using BAG is to allow more rapid design iteration. The primary goal in terms of design automation is to capture the core PA's design in schematic and layout generators capable of implementing a variety of instances based on sets of parameters provided by the user. Other key blocks are also automated using BAG, which will be discussed in detail. The usage of BAG also allows for design reuse of smaller blocks once a large enough library of generators has been developed.

To summarize, this work aims to tackle the issue of a reconfigurable TX by implementing an integrated, wideband PA with on-chip harmonic cancellation and mixed-signal filtering to reduce undesired spectral emissions. In order to implement such a complex system, we use the BAG framework to automate large portions of our design. These transmitters are demonstrated to be effective with both simulation results and measurements from fabricated designs.

1.1 State of the Art

There are a variety of approaches to more versatile transmitters, capable of meeting more than a single standard. Integrated multi-standard CMOS transmitters have been presented as a way of implementing a frequency-flexible transmitter [19][20][21]. However, these transmitters generally have output power levels < 10 dBm, well below what is required for cellular handsets. These transmitters are generally intended to be used with external PAs. Even if the transmitters produce low distortion and out-of-band noise, the external PA will generate its own distortion, which can create a need for an external filter in addition to the PA's output network. This is further exacerbated by the fact the transmitters supporting multiple bands have physically separate output ports for different bands, meaning multiple external PAs are required.

Multiple integrated circuit (IC) multi-standard PAs outputting cellular handheld levels of power have also been demonstrated [22][23][24]. While these works have very good performance and reach our power target, they do not achieve the level of integration targeted in this work. For example, the PA cores of [22][23] are implemented a InGaP/GaAs heterojunction bipolar transistor process instead of a standard CMOS process. Both [22][23] use multiple ICs on a single printed circuit board (PCB) and utilize a large number of off-chip passive components to implement the output filtering. In contrast, [24] doesn't use off-chip components explicitly, but has a separate die for the two power amplifiers, for passives consisting of power combining and filtering, and for a RF power switch, integrated onto a single package. While this is an impressive level of integration, it does not achieve the single die level of integration we are aiming for.

Another approach in building more versatile transmitters is by implementing high power multi-mode / multi-mode CMOS PAs on a single IC [8][25][26][27]. These meet our target power levels and are relatively well integrated, but don't quite satisfy our goals for different reasons. The work in [8] demonstrates separate 2 GHz and 5 GHz digital PAs on a single die, with WiFi, LTE, and Bluetooth measurements shown for the 2 GHz PA. The 2 GHz PA is shown to operate at least across a 2.3 GHz - 2.6 GHz band given the standards measured, but we want to implement a TX covering an even wider band. The other three works [25][27][26] present integrated PAs reaching cellular handheld levels of output power, but all three do not implement integrated amplitude control. In particular, [26] includes many off-chip passive components in its output network. However, [27] is highly frequency flexible design due to implementing an integrated, programmable output network with a center frequency tuning

range of 1.8 GHz - 2.2 GHz resulting in a 3-dB bandwidth ranging at least between 1.6 GHz - 2.6 GHz. A drawback to this approach is that it requires high power switches and supply voltages for the programmable output network.

Effective techniques to suppress harmonics in switching PAs have been demonstrated in recent years. These utilize a variety of techniques, such as conduction angle calibration [28], duty-cycle control [29], and harmonic cancellation [30]. Several of these works [28][30] implement PAs targeting low power standards (< 10 dBm) which do not employ amplitude modulation. Due to this, they only demonstrate results with either single tone, continuous wave (CW) tests or simple modulations like BSPK at a fixed frequency. The work of [30] utilizes the same technique we seek to implement in this work, but we wish to demonstrate that this technique is effective at higher powers, with modulated data, and across a wide frequency range when implemented carefully.

The mixed-signal filtering technique has been demonstrated in CMOS TXs utilizing digital PAs to varying degrees of effectiveness [31][32]. In particular, [32] demonstrates good filtering results at watt-class output power. Both demonstrate high levels of integration, with both using only a few off-chip passive components. Our goal is to demonstrate that this technique can work in conjunction with harmonic cancellation.

1.2 Scope of the Dissertation

This dissertation will discuss the implementation of two prototype TXs which combine two filtering techniques, harmonic cancellation and mixed-signal filtering, in order to control spectral emissions in a programmable fashion. First, we will introduce the switched-capacitor power amplifier (SCPA), the topology used in both versions of the TX. An overview of the TX system will be given.

The two filtering techniques will be described in detail, with both the mathematical representation discussed as well as practical implementation. We will cover the overall TX architecture and how the final choice of a cartesian architecture was driven by the implementation requirements of the two filtering techniques. The choice of PA topology will also justified by examining the key requirements of the filtering techniques and how the SCPA satisfies those.

The delay and phase shift elements of the filtering techniques will be covered, with special attention paid to the implementation of the harmonic cancellation. We will discuss the why the phase shift implementation matters, and the gilbert-cell based phase interpolator (PI) was chosen. An analysis of the relationship between PI phase resolution to harmonic cancellation will be presented.

The usage and purpose of IQ combining using 25% duty cycle local oscillator (LO) signals in our cartesian TXs will be discussed. We will also look into how 25% duty cycle LO signals can cause nonlinearity in the TX output constellation which is code dependent, which will be demonstrated both mathematically and with simulation results. Duty cycle

control is presented as a way of fixing this nonlinearity, backed by simulation results and with a practical implementation shown and implemented.

We will transition into an depth analysis on designing the TXs to maximize efficiency while meeting filtering specifications. The underlying mechanisms which link the PA switch sizing to the efficiency and filtering specifications will be explained in detail. This culminates in a design algorithm being presented for the PA and its output network given high level specifications and system level parameters.

We will present a prototype along with a wide variety of measured results verifying the effectiveness of the techniques. However, the implementation of the harmonic cancellation was not very robust in the first version, and was extremely reliant on symmetry in the output network. This will be analyzed in detail, with the proposed hypothesis validated by measurements and simulation setups. A solution to this issue will be proposed, analyzed, and verified in the prototype of the second version of the TX. The measured results of this version will be presented to verify the efficacy of the changes between the first and second versions.

In the final portion of this dissertation, the design automation which was core to this work will be covered. The Berkeley Analog Generator (BAG) framework will be introduced, with benefits and disadvantages clearly outlined. Layout generation of key blocks in both versions of the TX will be discussed in great detail, consisting of the SCPA generator for version 1, and the SCPA and PI generators for version 2. The SCPA generator section covers both array level and unit cell designs, as well as floorplanning for more optimal layout. Design methodology within BAG will be elaborated upon, with examples for both layout sizing and circuit design demonstrated.

An introduction to the changes of BAG 2.0 as well as its benefits will be presented, as well as the way in which manufacturable designs can be ensured. Changes and improvements to the 2nd version of the SCPA for the second (revised) version of the TX will be discussed, with specific details regarding modifications to the unit cell in order to improve overall layout and circuit performance. The PI generator will be discussed, with the current DAC (IDAC), and integrator covered in this work. In particular, an algorithm is presented to size the IDAC unit cells to meet resolution requirements, which are critical to harmonic cancellation performance.

Finally, we will conclude with a summary of the work presented, along with key contributions of this work. We close with a list of potential future work that builds upon the presented dissertation.

Chapter 2

Filtering Power Amplifiers

This chapter will begin by giving an introduction to the switched-capacitor power amplifier (SCPA), the PA topology used in the transmitter (TX). The overall TX system will then be introduced, after which harmonic cancellation and mixed-signal filtering will be discussed in detail, as well as why the SCPA is a natural choice for these techniques. A deep analysis of how to design and size the SCPA and overall TX will be presented, covering specifications of output power, efficiency, and linearity. Measured results from a 65 nm prototype will be shown, validating the techniques. In the latter half of the chapter, the effect of output network symmetry on the filtering techniques in the first version of the TX will be examined, and a solution will be proposed to make the techniques less dependent on this symmetry. The chapter concludes with measurements from a second prototype, this time in a 28 nm process, which validate the solution to the symmetry issue.

2.1 Switched Capacitor Power Amplifier

The switched-capacitor power amplifier (SCPA) topology was first demonstrated in [9]. The SCPA operates as a DAC with unit cells composed of an inverter driving a series capacitance, with the other capacitor node being shorted together between all cells. The unit cell of a stacked version with buffers is shown in Fig. 2.1. A key characteristic of the SCPA is its linear input code to output voltage transfer function under ideal operation. This stands in contrast to topologies like the Class D^{-1} [4], which are inherently nonlinear even with perfect switches. The SCPA's linearity is practically limited by device nonlinearity and device and capacitor matching, but it remains significantly more linear than other topologies.

This linearity can be shown with a model of the SCPA (Fig. 2.2a) where the transistors are treated as ideal switches with a non-zero on resistance. Here, n is the number of switching unit cells while N is the total number of unit cells. These groups of unit cells are equivalent to a single cell with an impedance reduced by n. If the pull-up and pull-down resistances are the same ($R_p = R_n$), the model can be further simplified (Fig. 2.2b) by replacing the switches and supplies with a single square wave voltage source with a source impedance. The

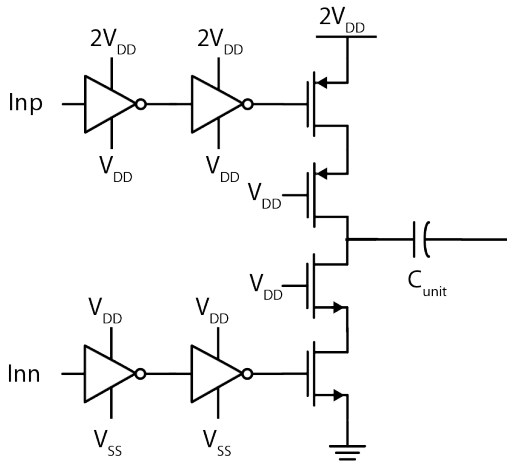


Figure 2.1: SCPA unit cell schematic with device stacking and buffers

effects of this assumption not being met will be analyzed further in section 2.5. This square wave can be decomposed into harmonics allowing for frequency domain analysis. Since this is a linear circuit, it can be further simplified using Thevenin's theorem into the form shown in Fig. 2.2c. The overall transfer function assuming a load Z_L is shown in Eq. 2.1.

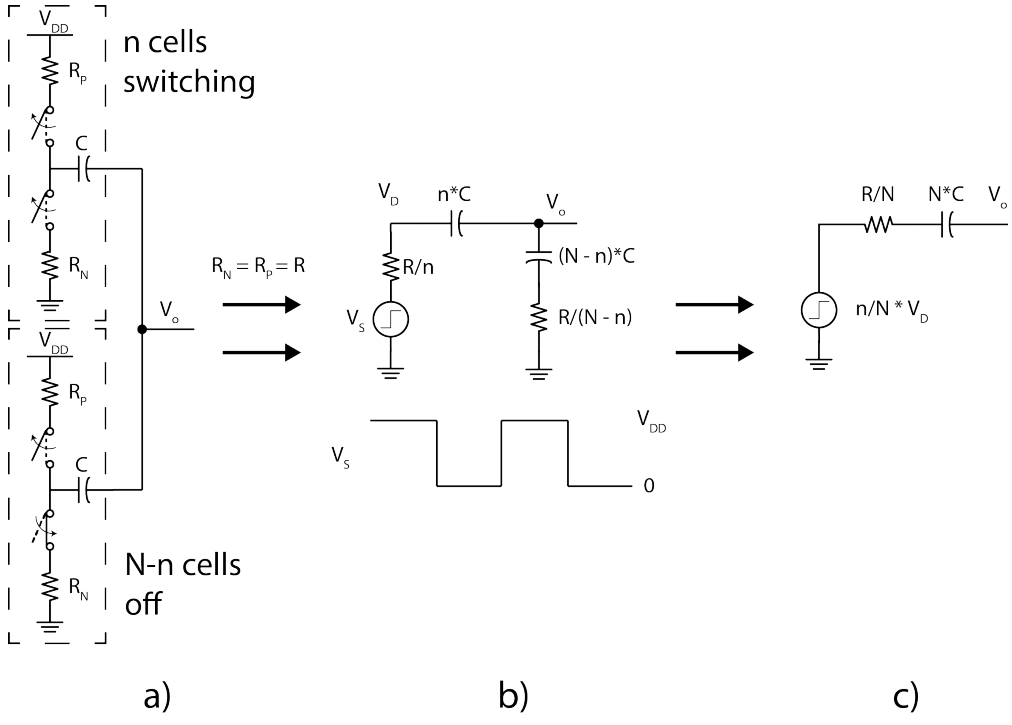


Figure 2.2: Ideal SCPA operation

$$\frac{v_o}{v_s} = \frac{n}{N} \cdot \frac{NZ_L}{NZ_L + Z_s} \tag{2.1}$$

$$Z_s = R + \frac{1}{j\omega C}$$

This analysis can be extended to multiple sub-PAs which are drain combined, which is modeled in Fig. 2.3. For two sub-PAs with n_1 and n_2 switching cells and N_1 and N_2 total cells, we can derive the transfer function in Eq. 2.2. If we have the input V_s shown in Fig. 2.3, we can also compute a time-domain expression for V_o (Eq. 2.3). This demonstrates that under ideal operation of the SCPA, the summation is completely linear.

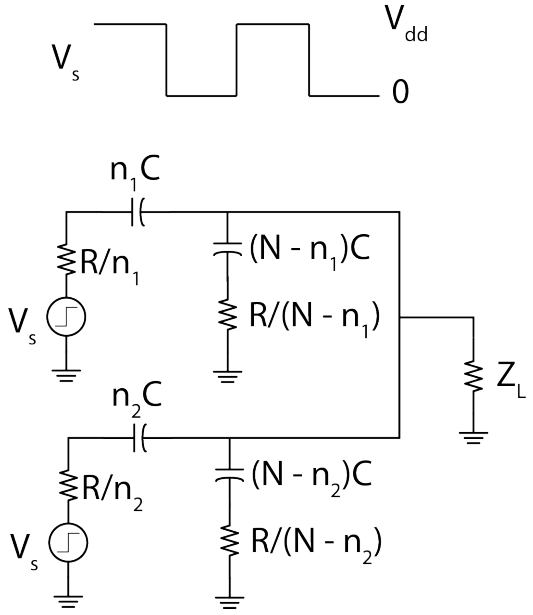


Figure 2.3: SCPA summation operation

$$\frac{v_o}{v_s} = \frac{n_1 + n_2}{2N + \frac{Z_S}{Z_L}} v_s = \frac{n_1 + n_2}{2N} \cdot \frac{2NZ_L}{2NZ_L + Z_s} \tag{2.2}$$

$$V_o(t) = \frac{V_{DD}}{2} + \frac{2V_{DD}}{\pi} \sum_{k=1,3,5,\ldots}^{\infty} \frac{1}{k} \cdot \frac{n_1 + n_2}{2N + \frac{Z_s(jk\omega_0)}{Z_L(jk\omega_0)}} \cdot sin(k\omega_o t) \tag{2.3}$$

This linearity is a key trait of the SCPA topology, and is critical to the effectiveness of the filtering techniques presented in Section 2.3. The class D^{-1} topology was initially used, but the extremely nonlinear behavior made these techniques ineffective. Previous work utilizing a Class E/F_{odd} topology [31] has shown relatively poor filtering results, due largely to the nonlinearity of the topology.

2.2 TX System Block Diagram

The overall system block diagram will be introduced, with the filtering techniques and design decisions explained in later sections. The top level block diagram of the TX is shown in Fig. 2.4, with all blocks depicted implemented on chip. The SCPA is split into 2 PA arrays, each consisting of 4 sub-PAs for each of the mixed-signal filter taps. There are a total of 2 PA arrays for the 2 LO phases, which means the system has a total of 8 sub-PAs. Each sub-PA is implemented as a 9-bit segmented DAC with 4 binary, 4 thermometer, and 1 sign bit. The PA arrays are summed at the output using a series stacked 1:2 transformer to provide tank inductance and increase output power.

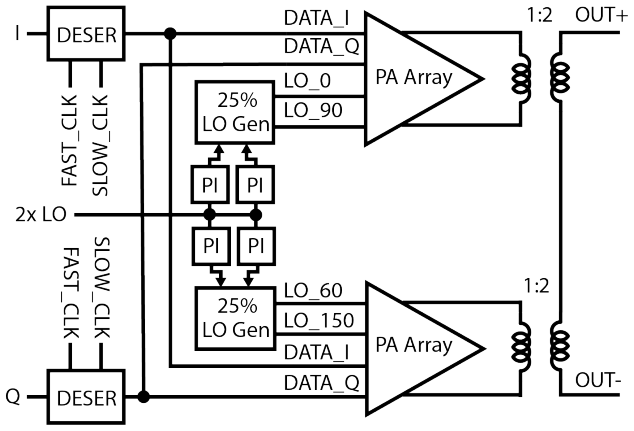


Figure 2.4: SCPA v1 top level block diagram

The TX is implemented using a cartesian architecture, which is also referred to an IQ system since it contains real and imaginary data represented as I and Q, respectively. The I and Q data are received differentially at $f_{clk,fast} = 2.5GHz$, which is deserialized by a factor of 10:1. 9 of 10 bits are used, with the last bit discarded. This 10:1 ratio was chosen to allow for a simple relationship between $f_{clk,fast}$ and $f_{clk,slow}$. The deserializer uses a double data-rate DDR scheme in order to allow for a lower $f_{clk,fast}$, and sets $f_{clk_slow} = 500MHz$. A scan chain is also implemented to allow for external configuration of various settings, such as bias currents for various blocks.

Clock receivers take the sinusoidal fast data clock, slow data clock, and $2f_{LO}$ local oscillator (LO) signals and generate digital signals for use on chip. The $2f_{LO}$ signal goes through a divider to generate I, Ib, Q, Qb LO signals at f_{LO}. These then are fed into pairs of phase interpolators (PIs) to generate the four sets of phases associated with a PA array, which are then used to generate 25% duty cycle versions of the same phases. This LO network comprises the harmonic cancellation portion of the TX. The harmonic cancellation will be configured to target the 3rd harmonic, as this will be the largest harmonic in our system since we use a differential PA.

Both the SCPA capacitors and transformer were implemented on chip, with no off chip passives used in the output network in order to meet the goal of having a single integrated

TX. Both passive components are sized for an overall low loaded quality factor (Q) so that the overall TX is wideband.

A specific standard was not targeted, but the general specifications for the TX included a peak output power $P_{out} > 24dBm$, a center frequency f_{LO} of around 1 GHz with at least several hundred MHz of bandwidth. Additionally, the TX is designed to support HD3 reduction of 40 dB, with the HD3 reduction being defined as the difference in HD3 from the case where the PAs sum in phase to the cancelled case.

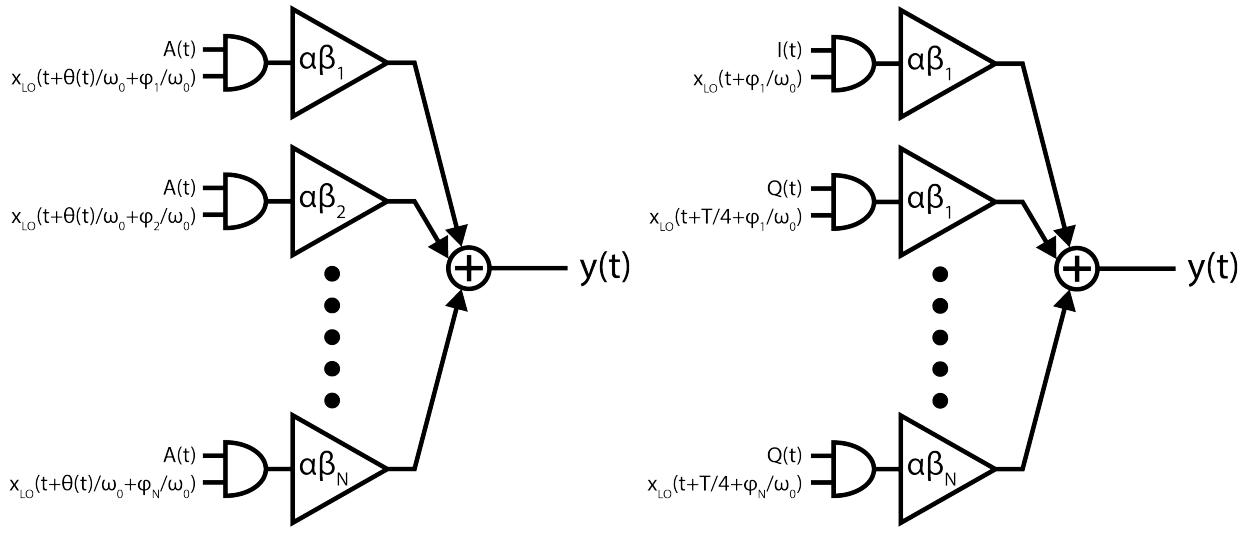
Now that a rundown of the TX system is given, we will discuss the implementation of the harmonic cancellation and mixed-signal filtering, starting at a high level and moving to a lower level. The choice of architecture was made based on the implementation requirements of these filtering techniques.

2.3 Filtering Techniques

Harmonic cancellation and mixed-signal filtering can be used to tackle the harmonic emission and quantization noise issues. Both of these techniques are active cancellation schemes which follow the same general guideline: the PA is partitioned into several sub-PAs, driven with different inputs, and summed at the output. The sub-PA inputs take in the upconverted IQ data, which has already been mixed with the LO. Harmonic cancellation is implemented by feeding sub-PAs different LO phases, while mixed-signal filtering is implemented by feeding sub-PAs different baseband data. Modern transmitters send complex-valued data, which can be represented either in a real-imaginary format (cartesian) or amplitude-phase format (polar). Since these filtering techniques rely on manipulating the input signals of each sub-PA, the implementation will differ in cartesian and polar architectures. We will first discuss the harmonic cancellation, then the mixed-signal filtering, and finally combine the two into a single system.

The block diagram for the harmonic cancellation technique is shown in Fig. 2.5 for both polar and cartestian architectures. This harmonic cancellation technique was primarily used in mixers for RF receivers [33][34], but has seen usage with switching PAs recently [30]. The polar architecture consists of amplitude $A(t)$ and phase $\theta(t)$, while the cartesian architecure has in-phase $I(t)$ and quadrature $Q(t)$ componenets. The signals $A(t)$, $\theta(t)$, $I(t)$, $Q(t)$, and x_{LO} are digital, with $x_{LO}(t)$ being a square wave with period T, with $\omega_0 = \frac{2\pi}{T}$. The digital AND gate acts as a mixer (multiplier) for the data and LO. We are assuming that the switching PAs generate a square wave output from x_{LO} with a gain α, though this analysis still holds as long as the sub-PAs outputs are periodic with period T. Each sub-PA is weighted with a scaling factor β_l and driven with an LO with a phase shift ϕ_l at the fundamental frequency. This scheme is represented mathematically in Eq. 2.5 for the polar case and in Eq. 2.6 for the cartesian case.

$$x_{LO}(t) = \sum_{k=-\infty}^{\infty} a_k e^{jk\omega_0 t} \tag{2.4}$$



(a) Polar architecture (b) Cartesian architecture

Figure 2.5: Harmonic cancellation block diagram

$$y(t) = \sum_{l=1}^{N_\phi} A(t) \cdot \alpha\beta_l x_{LO}\left(t + \frac{\theta(t)}{\omega_o} + \frac{\phi_l}{\omega_o}\right) \tag{2.5}$$

$$y(t) = \sum_{l=1}^{N_\phi} \alpha\beta_l \left[I(t)x_{LO}\left(t + \frac{\phi_l}{\omega_o}\right) + Q(t)x_{LO}\left(t + \frac{T}{4} + \frac{\phi_l}{\omega_o}\right)\right] \tag{2.6}$$

Using the fourier series decomposition of $x_{LO}(t)$ (Eq. 2.4), the output $y(t)$ can be rewritten as Eq. 2.8 (polar case) and Eq. 2.9 (catesian case). Proper choices of β_l and ϕ_l can set $\gamma_k = 0$ for certain values of k, cancelling specific harmonics with a small reduction of the fundamental. A common harmonic cancellation scheme uses three sub-PAs weighted by $1, \sqrt{2}, 1$ with phase shifts of $0, 45°, 90°$ respectively. This perfectly cancels the $3^{rd}, 5^{th}, 11^{th}, 13^{th}$, etc. harmonics while reducing the fundamental by 1.6 dB.

$$\gamma_k = \sum_{l=1}^{N} \beta_l e^{jk\phi_l}$$

$$y(t) = \alpha A(t) \sum_{k=-\infty}^{\infty} a_k e^{j(k\omega_0 t + \theta(t))} \sum_{l=1}^{N} \beta_l e^{jk\phi_l} = \alpha A(t) \sum_{k=-\infty}^{\infty} a_k \gamma_k e^{j(k\omega_0 t + \theta(t))} \tag{2.7}$$

$$y(t) = \alpha A(t) \sum_{k=-\infty}^{\infty} a_k \gamma_k e^{j(k\omega_0 t + \theta(t))} \tag{2.8}$$

$$y(t) = \alpha \sum_{k=-\infty}^{\infty} a_k \gamma_k e^{jk\omega_0 t} \left[I(t) + Q(t) e^{\frac{jk\pi}{4}} \right] \tag{2.9}$$

There are several ways of generating the phase shift assuming we have a base LO signal either brought in externally or generated by an on chip oscillator. The phase shift can either be generated directly using a phase interpolator (PI) or using a time delay implemented by a delay line.

If we want the harmonic cancellation to work across a frequency range $f_{min} \leq f \leq f_{max}$ with a minimum resolution kT_{min}, this imposes two major constraints on the delay line implementation. The maximum delay required depends on f_{min}, with a maximum delay $t_{dmax} = \frac{1}{6f_{min}}$ to implement a 60° phase shift. However, the minimum time delay Δt depends on f_{max}, with $\Delta t = \frac{k}{f_{max}}$. This means that we need a much higher effective resolution if the delay line was designed for a single operating frequency. For operation across a 1 - 2GHz range, we require $t_{dmax} = 166.7ps$ and $\Delta t = 1.4ps$. This increased resolution can be relaxed by using a coarse-fine delay line, such as in [35][36].

For our requirements however, and since the LO is a steady-state periodic signal, a phase interpolator (PI) provides much better phase noise performance for a given power specification than a delay line. We used a gilbert-cell based phase interpolator [18], with an N-bit implementation shown in Fig. 2.6. The output phase is set by the signals I_PHI, I_PHI_B, Q_PHI, and Q_PHI_B. This topology requires input integrators to shape the input LO phases into triangular waves. Frequency flexibility can be easily achieved with this topology by providing a programmable bias to the integrators in the form of a current DAC. The integrator's dominant pole can then be moved to different frequencies by modifying g_m by changing the bias current. Prior work utilized gilbert-cell based PIs operating across a 0.4 - 4.0 GHz range in a polar architecture [37], demonstrating the frequency flexibility of this PI topology.

A separate PI is required for each phase shift in both the cartesian and polar architectures. However, the polar architecture has the phase input of the PI on the high speed data path, and requires adders to implement the fixed phase shifts ϕ_l. In a cartesian architecture however, the phase input of the PI can be statically set to the desired phase shift.

The phase shift ϕ_l can be applied directly to the LO signal, but for a standard switching PA, the size of the PA must be scaled in order to implement the β_l. The LO input can't be scaled due to being a digital square wave signal. This direct scaling has been implemented in previous harmonic cancellation PA works [30]. However, this problem becomes much more difficult when each sub-PA is a DAC, as special care has to be taken to properly implement the scaling even for the smallest binary cell to ensure good matching. The layout overhead becomes even larger if the ratios between any weights are not a whole number. As an example, we can examine the harmonic cancellation setup with sub-PA weights 1, 1.4, 1 with phase shifts of 0, 45°, 90°. The value of $\sqrt{2}$ is rounded to 1.4 to simplify the implementation of the multiplier.

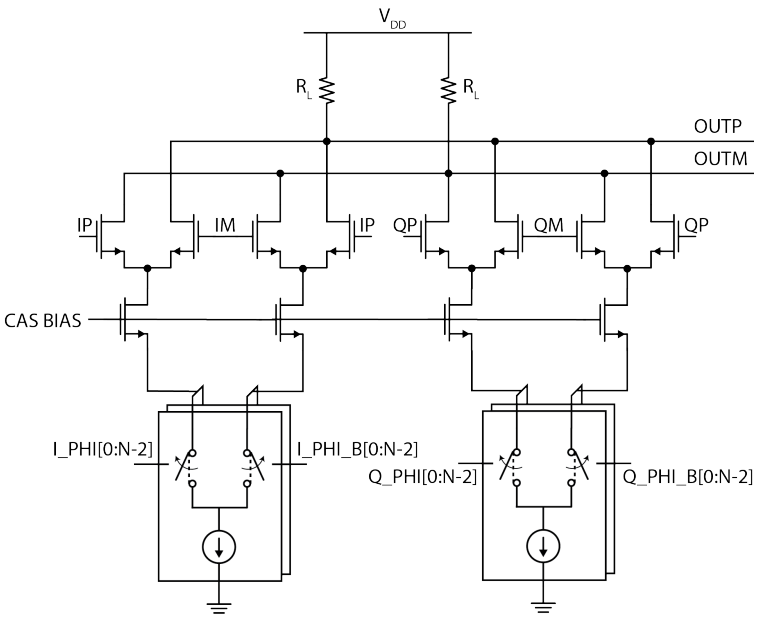


Figure 2.6: Gilbert-cell based phase interpolator

In order to scale the output stage transistors, we must rely on modifying the number of transistor fingers. Though width scaling is simpler, it does not provide reliable matching in advanced planar technologies. The weights of 1, 1.4, and 1 can be implemented with ratios of 5, 7, 5 fingers. However, we generally want a finger pair to be the unit element in order to balance out biases in current flow based on direction [38], which raises this minimum number of fingers to 10, 14, 10. This sets the minimum total width of the smallest device, which forces the unit cells to be upsized if the desired size of the minimum smallest binary cell is below this.

Mutiplying the digital inputs of each sub-PA by its appropriate weight β_l is another way of implementing the scaling, assuming each sub-PA is a DAC. Digital multipliers would be added in the data path, either for the IQ signals or the amplitude for a polar implementation. If any of the scaling factors are not whole numbers, the input codes must be rounded after multiplication, introducing input code dependent quantization error. This quantization error generates nonlinearity (DNL, INL) and input code dependent harmonic cancellation. In particular, we will use the metric of HD3 reduction ΔHD_3, which is defined as the difference in HD3 under cancelling conditions and when the sub-PAs sum in phase.

In order to quantify this, we'll look at an example with a fixed-point multiplier with 4 bits of precision (down to $1/16^{ths}$). Using the same harmonic cancellation scheme as the previous two paragraphs, this rounds $sqrt(2)$ to 1.4375. Fig. 2.7 demonstrates the HD3 reduction across input code with and without rounding after multiplication, when we assume a maximum input code of 255. Quantization limits the HD3 reduction to 15 dB in the worst case, though we have $\Delta HD_3 \geq 30dB$ for $n \geq 6$. Additionally, quantization introduces a max ≥ 0.4 LSB of INL, and DNL of -0.58 to 0.25 LSB. This quantization-induced nonlinearity

combines with the random mismatch induced nonlinearity to degrade the effective resolution of the DAC.

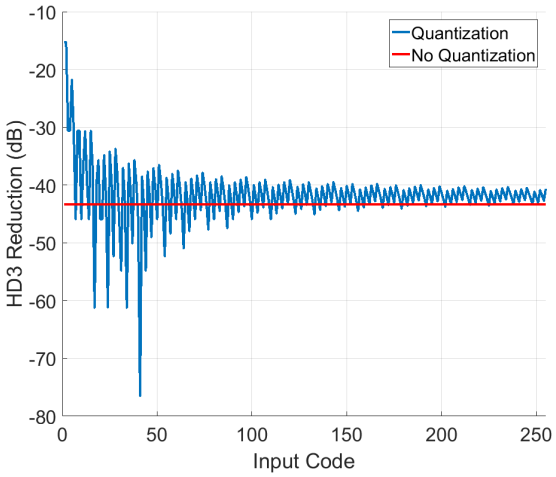


Figure 2.7: HD3 reduction with and without quantization

Given all these issues, we opted go for a more simple configuration which would still provide harmonic cancellation without requiring sub-PA scaling. The PA is split into two equally weighted sub-PAs which are fed the same IQ data but LO signals phase shifted by 60°. This shifts the fundamental of the output by 60° and the third harmonic by 180°. If the sub-PAs are perfectly matched, the $3^{rd}, 9^{th}, 15^{th}$, etc. harmonics will be completely cancelled while the fundamental is reduced by 1.2dB.

The block diagram for the mixed-signal filter is shown for a polar architecture (Fig. 2.11a)

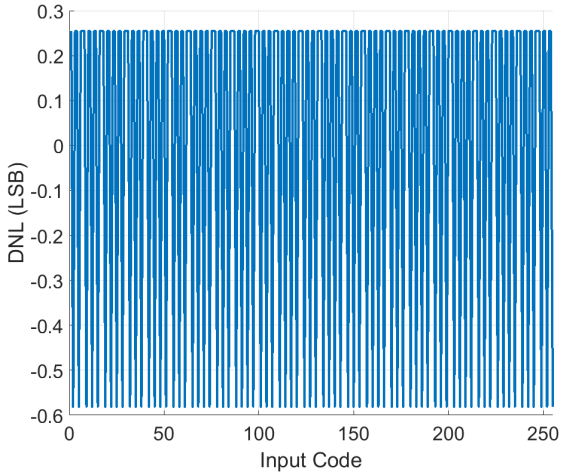


Figure 2.8: DNL with quantization

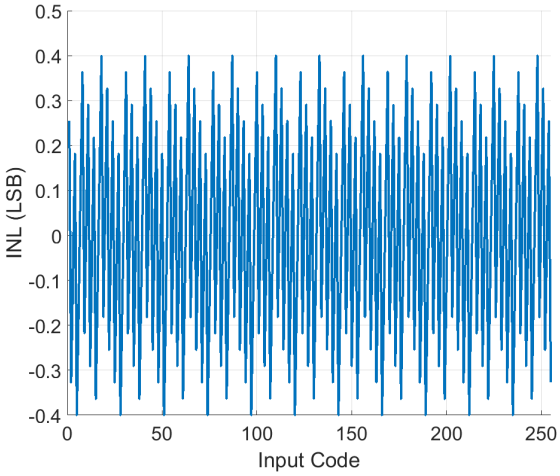


Figure 2.9: INL with quantization

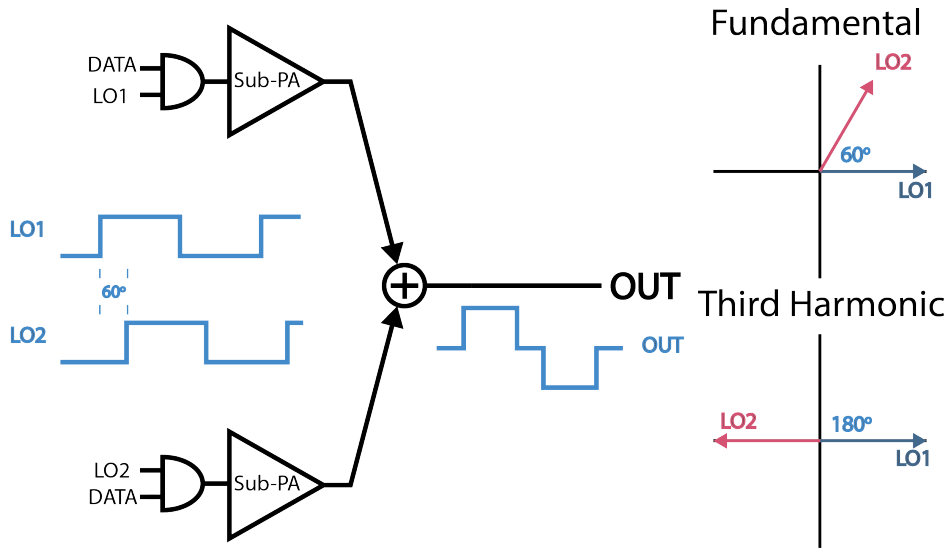
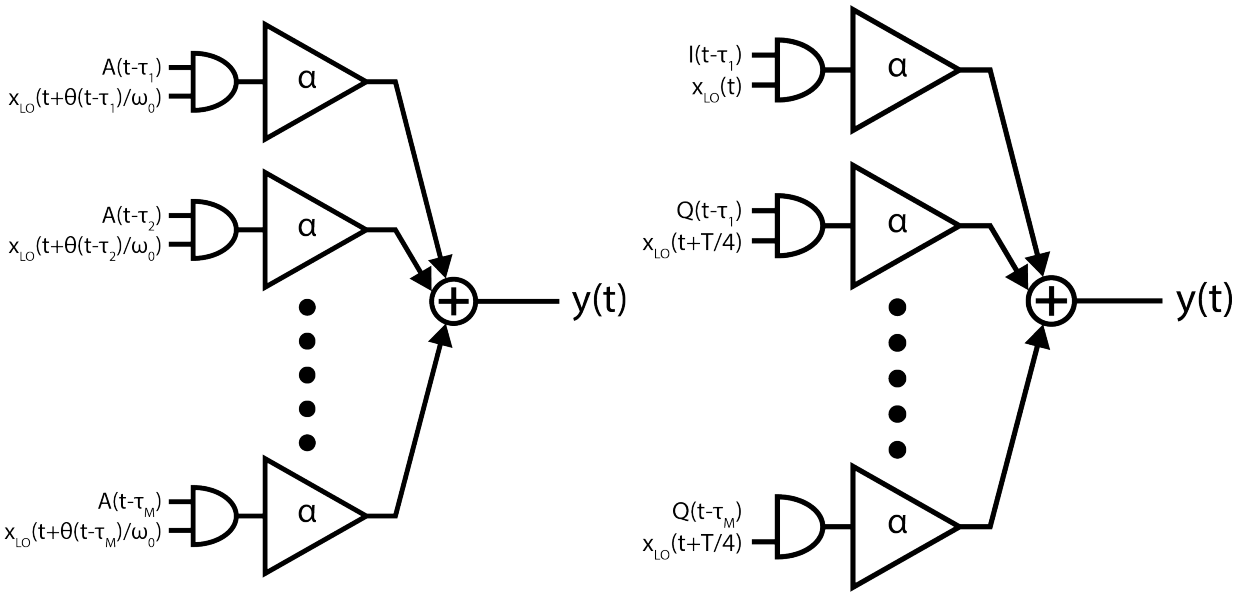


Figure 2.10: Harmonic cancellation block diagram

and cartestian architecture (Fig. 2.11b). This technique has been previously demonstrated using digital PAs [31][32]. In both polar and cartesian architectures, the input data is delayed by τ_m for the m^{th} sub-PA. This implementation makes the key restriction of implementing only real-valued filter coefficients. Additionally, the sub-PAs are equally weighted in order to avoid similar sub-PA scaling issues discussed in the harmonic cancellation section. In this case, there is a significant difference in the implementation of the mixed-signal filter in the two architectures.

The mathematical representation of this filter implementation in the polar architecture is shown in Eq. 2.11. Each sub-PA must be able to take in different delayed versions of the amplitude and phase data. The different phase data requirement causes the number of PIs needed to scale linearly with the number of sub-PAs since each sub-PA needs an LO with an arbitrary phase $\theta(t - \tau_m)$. In contrast, the cartesian case (Eq. 2.10) only requires delays on $I(t)$ and $Q(t)$, with two versions of the LO signal x_{LO} required regardless of the number of sub-PAs. The number of PIs is fixed and does not scale with the number of sub-PAs in the cartesian implementation.

This reduced number of PIs is the primary reason we chose to implement the TX using a cartesian architecture instead of a polar one. This is key because PI performance is critical to the effectiveness of our harmonic cancellation, which is analyzed more thoroughly in Section 2.4. This requirement also drives the PIs to be relatively large, so reducing the total number of PIs saves significant area. It should be emphasized that the relative simplicity of cartesian implementation only holds when we restrict the filter coefficients to being real-valued. It is comparable with the polar implementation if complex-valued filter coefficients are allowed.



(a) Polar architecture

(b) Cartesian architecture

Figure 2.11: Mixed signal filter block diagram

$$y(t) = \alpha \sum_{m=1}^{M} I(t-\tau_m)x_{LO}(t) + Q(t-\tau_m)x_{LO}\left(t+\frac{T}{4}\right) \tag{2.10}$$

$$y(t) = \alpha \sum_{m=1}^{M} A(t-\tau_m)x_{LO}\left(t+\frac{\theta(t-\tau_m)}{\omega_o}\right) \tag{2.11}$$

The mixed-signal filtering implemented using this technique implements an FIR filter with several restrictions. For a configuration with a total of M equally weighted sub-PAs of weight α, each tap coefficient must be a multiple of α, with the sum of the absolute values of the coefficients totaling αM. For example, with $M = 4$, $y(n) = 2\alpha x(n) + 2\alpha x(n-n_1)$ and $y(n) = \alpha x(n) - 2\alpha x(n-n_1) - \alpha x(n-n_2)$ are valid configurations while $y(n) = \frac{3\alpha}{2}x(n) + \alpha x(n-n_1)$ is not.

The choice of delay line topology is set primarily by the maximum delay needed. For the cellular standards targeted in our designs, the nearest channel will likely be 20 MHz or 40 MHz away, such as in LTE. With 4 sub-PAs, we can implement the double notch filter in Eq. 2.12, with the time-domain implementation shown in Eq. 2.13. A maximum delay of $1/f_{notch}$ is needed to implement a double notch at f_{notch}. This means that we need 50 ns of delay to implement two notches at 20 MHz. Given this requirement, we chose to implement the delay line with a chain of flip-flops due to the relative simplicity as well as the relatively low power cost for a given jitter specification. This chain of flip-flops is clocked using the data clock, with the specific delayed output selected using a MUX (Fig. 2.12). In our TX,

the MUX select is set using the TX's scan chain. The delay line consisted of a total of 25 flip flops clocked at $f_{clk,slow} = 500MHz$, resulting in the desired maximum delay of 50 ns. An inverter based delay line requires either a very large number of stages or large load capacitors to achieve this delay with reasonable jitter performance, which is costly both in area and power.

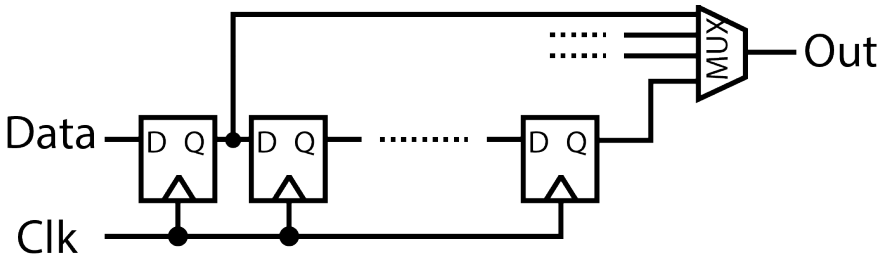


Figure 2.12: Delay line schematic

$$Y(\omega) = X(\omega) \cdot \left(1 + e^{-j\omega\tau_1}\right)\left(1 + e^{-j\omega\tau_2}\right) = X(\omega) \cdot \left(1 + e^{-j\omega\tau_1} + e^{-j\omega\tau_2} + e^{-j\omega(\tau_1+\tau_2)}\right) \quad (2.12)$$

$$y(t) = x(t) + x(t - \tau_1) + x(t - \tau_2) + x(t - \tau_1 - \tau_2) \quad (2.13)$$

Harmonic cancellation and mixed-signal filtering can be implemented simultaneously, both in cartesian (Eq. 2.14) and polar (Eq. 2.15) architectures for a generic periodic signal x_{LO}. These equations assume N LO phases for harmonic cancellation and M sub-PAs for mixed-signal filtering.

$$y(t) = \alpha \sum_{l=1}^{N} \sum_{m=1}^{M} I(t - \tau_m) x_{LO}\left(t + \frac{\phi_l}{\omega_0}\right) + Q(t - \tau_m) x_{LO}\left(t + \frac{T}{4} + \frac{\phi_l}{\omega_0}\right) \quad (2.14)$$

$$y(t) = \alpha \sum_{l=1}^{N} \sum_{m=1}^{M} A(t - \tau_m) x_{LO}\left(t + \frac{\theta(t - \tau_m)}{\omega_0} + \frac{\phi_l}{\omega_0}\right) \quad (2.15)$$

A critical requirement for these techniques is that the summation of the sub-PAs must be linear. Nonlinearities can generate spectral content that "fills in" the notches generated by the cancellation. Toplogies with inherently nonlinear input code to output voltage transfer functions (such as Class D^{-1}) are not suited to these techniques. However, even topologies with perfectly linear operation in the ideal case can exhibit nonlinear summation. The most obvious source is from nonlinearity in the switch devices used, but nonlinear summation can also be casued by not matching the resistance of pull-up and pull-down networks. The latter will be discussed in more detail in Section 2.5. First, we will discuss how PI phase resolution impacts harmonic cancellation.

2.4 Phase Resolution and Harmonic Cancellation

Even with perfect amplitude matching, harmonic cancellation will be limited by phase matching, which is impacted primarily by phase interpolator resolution and phase noise. In order to understand how the phase resolution affects HD3, we first need to look at the transfer function with harmonic cancellation scheme, shown in Eq. 2.16, where ϕ is the phase shift. This can be used to compute the HD3 in Eq. 2.17, which can then be normalized to the HD3 reduction ΔHD_3 in Eq. 2.18, with the latter plotted in Fig. 2.13. $HD_{3,0}$ is defined as the HD3 measured when the PAs sum in phase ($\phi = 0$).

$$H(\omega) = 1 + e^{-j\omega t_0} = 1 + e^{-j\phi} \tag{2.16}$$

$$HD_3(\phi) = \left|\frac{H(3\phi)}{H(\phi)}\right| = (HD_{3,0}) \cdot \left|\frac{1 + e^{-j3\phi}}{1 + e^{-j\phi}}\right| \tag{2.17}$$

$$\Delta HD_3(\phi) = \frac{HD_3(\phi)}{HD_{3,0}} = \left|\frac{1 + e^{-j3\phi}}{1 + e^{-j\phi}}\right| \tag{2.18}$$

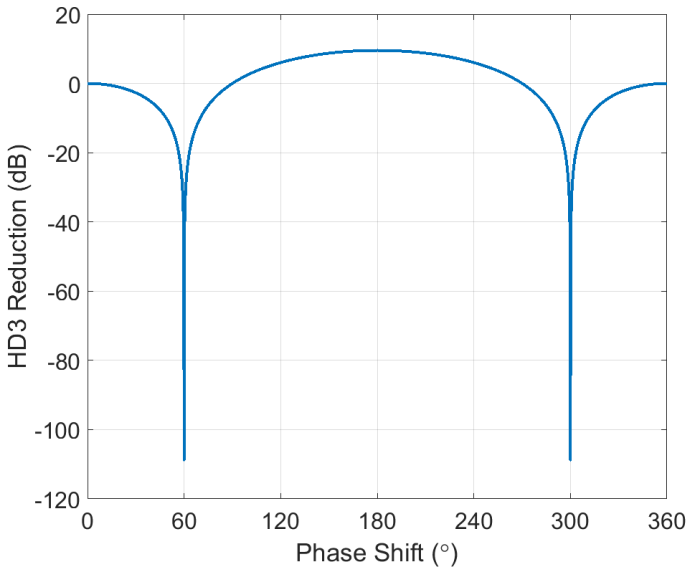


Figure 2.13: Ideal HD3 reduction vs phase shift

Perfect cancellation can be attained with a 60° phase shift but this impossible to guarantee in practice. Instead, we can design to ensure a minimum HD3 reduction, assuming perfect amplitude matching. This can be done by noting that there are exactly two phase shifts ϕ_a and ϕ_b that correspond with a given HD3 reduction, and that any value of phase shift between those two will exceed that HD3 reduction. If we have a phase step $\phi_{step} \leq \phi_b - \phi_a$,

Table 2.1: HD3 reduction vs number of bits

Min HD3 Reduction	Phase Step	n_{bits}
20 dB	6.623°	5.8
30 dB	2.092°	7.4
40 dB	0.661°	9.1
50 dB	0.209°	10.7
60 dB	0.066°	12.4

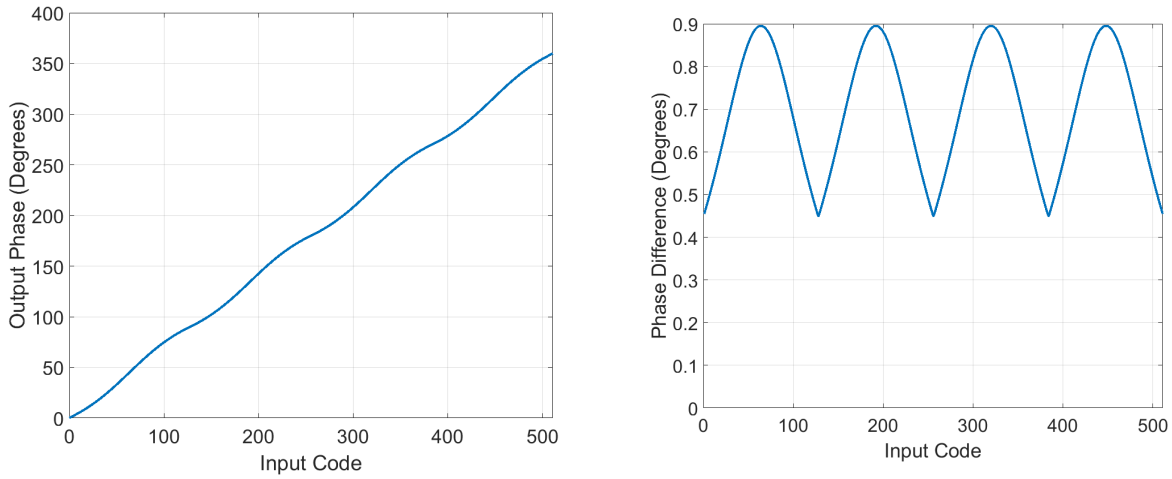
we will be able to guarantee that we can land within the desired phase shift interval and guarantee our minimum HD3 reduction. This is the phase step required for our PI, which can then be mapped to the required number of bits (or phase resolution) in the PI for a given full range.

Table 2.1 lists the phase steps and number of bits required to ensure different minimum HD3 reduction values. A full scale of 360° is assumed here, which is the case for a gilbert-cell based PI. The number of bits required to ensure higher HD3 reduction grows quickly, which is problematic because high resolution DACs are costly in area. Given this, we have chosen to target an HD3 reduction of 40 dB, which corresponds to about 9 bits of phase resolution for the PI.

This TX implements gilbert-cell based PIs, which operate by weighing the four phases (I, IB, Q, QB) and summing them in current (Fig. 2.6). Though the current summation is generally very linear, the ideal output phase vs input code transfer function is inherently nonlinear since it relies on a tan^{-1} function. This relationship is plotted for a 9-bit PI in Fig. 2.14a, with the phase step for each code given in Fig. 2.14b.

$$\phi(n) = \begin{cases} tan^{-1}\left(\frac{\frac{N}{2}-2n-1}{2n+1}\right) & 0 \le n < \frac{N}{4} \\ tan^{-1}\left(\frac{-2\left(n-\frac{N}{4}\right)-1}{\frac{N}{2}-2\left(n-\frac{N}{4}\right)-1}\right) & \frac{N}{2} \le n < \frac{N}{2} \\ tan^{-1}\left(\frac{-\frac{N}{2}+2\left(n-\frac{N}{2}\right)-1}{-2\left(n-\frac{N}{2}\right)-1}\right) & \frac{N}{2} \le n < \frac{3N}{4} \\ tan^{-1}\left(\frac{2\left(n-\frac{3N}{4}\right)+1}{-\frac{N}{2}+2\left(n-\frac{3N}{4}\right)+1}\right) & \frac{3N}{4} \le n < N \end{cases} \tag{2.19}$$

From Fig. 2.14b, we can deduce that the phase steps are largest at $(2k-1)\cdot 45°$ and smallest at $k \cdot 90°$ for gilbert-cell based PIs. The phase step ranges between 0.45° and 0.9° across input codes for an ideal 9-bit PI. The phase steps near 60° are about 0.84° as compared to 0.7° for a perfectly linear PI. This reduction in phase resolution causes a slight drop in minimum HD3 reduction from 39.5 dB to 37.9 dB, 2 dB less than the target 40 dB cancellation. Since practical issues such as DAC nonlinearity will degrade the minimum HD3 reduction, we deem this to be an acceptable loss.



(a) Phase vs input code

(b) Phase step vs input code

Figure 2.14: Ideal phase and phase step for a gilbert-cell based PI

2.5 Resistance Mismatch and SCPA Linearity

The introductory analysis of SCPA operation in Section 2.1 assumed equal pull-up resistance R_p and pull-down resistance R_n. Mismatch in R_n and R_p will generate nonlinearity in the SCPA, degrading the linear summation which is critical to the filtering techniques discussed in Section 2.3. Furthermore, it is difficult to ensure $R_n = R_p$ across voltage swing in practical implementations, since the switches will be implemented with transistors.

To analyze the effect of this nonideality on the output, we can analyze the schematic in Fig. 2.15, in which we assume non-switching cells are pulled to ground. The eventual goal is to compute the fundamental frequency component of the output $V_o(t)$. The load is being omitted to simplify the analysis. We have a time-varying resistance, which makes frequency domain analysis difficult, as it would require the use of Volterra series. In this case, it is much simpler to compute the time domain output and apply a Fourier series decomposition. The results of the first step give us the piecewise time domian expression in Eq. 2.20 for a single switching period. From this it is clear that any mismatch in the R_n and R_p introduces an error term which causes deviation from the ideal $\frac{n}{N}V_{DD}$, and that this error decays over time as well. Mismatch in R_n and R_p also cause the output impedance becomes time-varying and dependent on input code n.

$$V_o(t) = \begin{cases} \frac{nV_{DD}}{N}\left[1 + \epsilon \cdot e^{-\frac{t}{\tau_1}}\right] & 0 < t \leq \frac{T}{2} \\ 0 & \frac{T}{2} < t \leq T \end{cases} \tag{2.20}$$

$$\tau_1 = \frac{\big((N-n)R_p + nR_n\big)C}{N} \qquad \tau_2 = R_nC \tag{2.21}$$

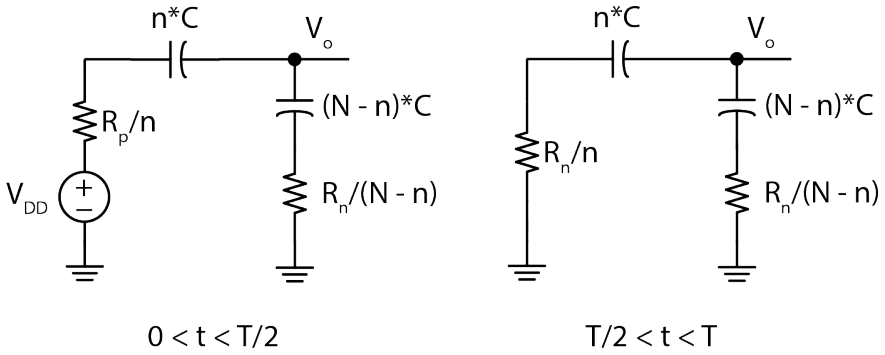


Figure 2.15: SCPA operation during across a single period

$$\epsilon = \left(\frac{1 - e^{-\frac{T}{2\tau_2}}}{1 - e^{-\frac{T}{2\tau_1} - \frac{T}{2\tau_2}}}\right)\left(\frac{R_n - R_p}{\frac{n}{N-n}R_n + R_p}\right) \tag{2.22}$$

Eq. 2.22 demonstrates that the error depends on the switching code used, with the worst case occurring for $n = 1$. For the other part of the error, the worst case error comes in the case where T is sufficiently larger than τ_1 and τ_2 (i.e. $T > 4.6\tau_1$, $T > 4.6\tau_2$), which gives the maximum value as shown in Eq. 2.23. For $N = 255$ and $R_p = 2R_n$, we find that $\epsilon = -0.499$.

$$|\epsilon| \leq \left|\frac{R_n - R_p}{\frac{1}{N-1}R_n + R_p}\right| = \left|\frac{1 - \frac{R_p}{R_n}}{\frac{1}{N-1} + \frac{R_p}{R_n}}\right| \tag{2.23}$$

We can apply a Fourier series decomposition on the time domain expression for $V_o(t)$ (Eq. 2.20) to write $V_o(t)$ as a sum of complex exponentials, with coefficients a_k shown in Eq. 2.25. This form can be further manipulated into Eq. 2.26, which splits $V_o(t)$ into real-valued DC, odd and even harmonic components. The amplitudes of the harmonics are expressed as b_k (Eq. 2.27) for odd harmonics and c_k (Eq. 2.28) for even harmonics, except for $k = 0$.

$$V_o(t) = \sum_{k=-\infty}^{\infty} a_k e^{jk\omega_0 t} = a_0 + \sum_{k=1}^{\infty}\left(a_k e^{jk\omega_0 t} + a_{-k}e^{-jk\omega_0 t}\right) \tag{2.24}$$

$$a_k = \begin{cases} \frac{nV_{DD}}{2N}\left[1 + \frac{\epsilon\tau_1}{T}\left(1 - e^{-\frac{T}{2\tau_1}}\right)\right] & k = 0 \\ \frac{nV_{DD}}{N}\left[\frac{1}{j\pi k} + \frac{\epsilon\left(1+e^{-\frac{T}{2\tau_1}}\right)}{\frac{T}{\tau_1}+j2\pi k}\right] & |k|\ odd \\ \frac{nV_{DD}}{N} \cdot \frac{\epsilon\left(1-e^{-\frac{T}{2\tau_1}}\right)}{\frac{T}{\tau_1}+j2\pi k} & |k|\ even \end{cases} \tag{2.25}$$

$$V_o(t) = a_0 + \sum_{k=1,3,5\ldots}^{\infty} b_k cos\left(k\omega_o t + \theta_k\right) + \sum_{k=2,4,6\ldots}^{\infty} c_k cos\left(k\omega_o t + \phi_k\right) \tag{2.26}$$

$$b_k = \frac{2nV_{DD}}{\pi kN}\sqrt{1+\frac{\epsilon^2\left[1+e^{-\frac{T}{2\tau_1}}\right]^2+4\epsilon\left[1+e^{-\frac{T}{2\tau_1}}\right]}{4\left[1+\gamma_k^{-2}\right]}} \tag{2.27}$$

$$c_k = \frac{nV_{DD}}{\pi kN}\cdot\frac{\epsilon\left[1-e^{-\frac{T}{2\tau_1}}\right]}{\sqrt{1+\gamma_k^{-2}}} \tag{2.28}$$

$$\theta_k = tan^{-1}\left(-\frac{\left(\epsilon\left[1+e^{-\frac{T}{2\tau_1}}\right]+2\right)\gamma_k^2+2}{\epsilon\left[1+e^{-\frac{T}{2\tau_1}}\right]\gamma_k}\right) \tag{2.29}$$

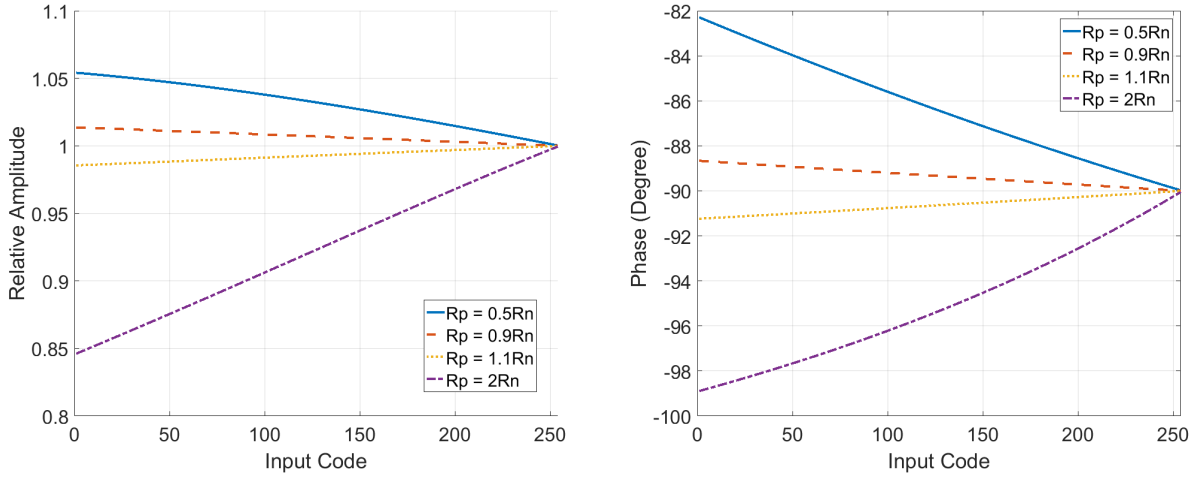
$$\phi_k = tan^{-1}\left(\gamma_k\right) \tag{2.30}$$

$$\gamma_k = \frac{2\pi k\tau_1}{T}$$

Mismatch between R_p and R_n can cause a variety of issues, such as generating even harmonics (Eq. 2.28) and causing AM-PM distortion (Eq. 2.29). The former is clear from $c_k \neq 0$, and the latter is clear due to the dependence of the phase θ_k on n for odd harmonics. Our primary concern however is how the mismatch affects the fundamental. From Eq. 2.27, we can see that b_k has a nonlinear dependence on n because ϵ and τ_1 are functions of n, generating nonlinearity in the input code to output voltage transfer function. Resistance mismatch causes the SCPA to sum nonlinearly even if the switches are perfectly linear, reducing the effectiveness of the filtering techniques used. Given this, it is critical to ensure $R_n = R_p$.

The amplitude (Fig. 2.16a) and phase (Fig. 2.16b) of the fundamental across code is plotted across several difference resistance mismatch cases to show the relative effects. Fig. 2.16a shows the amplitude as a multiple of the ideal case ($R_n = R_p$) - ideally, this should be 1 across all codes. The amplitude of the ideal case ($R_n = R_p$) and the case of $R_n = 2R_p$ is plotted in Fig. 2.17. In all of these plots, we are assuming $V_{DD} = 1V$, $R_n = 1k\Omega$, $C = 100fF$, $N = 255$, $T = 1ns$. R_n remains fixed and the value of R_p is modified.

From these plots, it can be seen that limiting the mismatch to a relatively small amount such as 10% will have a relatively small effect on AM-AM and AM-PM distortion. Even order harmonics can be removed by using a differential PA, but the AM-PM and AM-AM distortion remain issues. Given all this, it is important to match R_p and R_n, but a small amount of mismatch will not significantly degrade performance. We will now move to discussing the implementation of the TX architecture.



(a) Amplitude (relative) (b) Output phase

Figure 2.16: Fundamental amplitude and phase vs input code with resistance mismatch

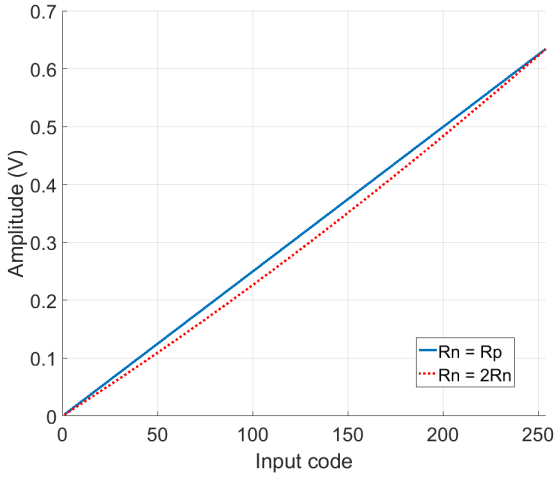


Figure 2.17: Fundamental amplitude across input code for matched resistance and a mismatched resistance case

2.6 Carestian (IQ) Architecture

Analysis in Section 2.3 has shown that when implementing harmonic cancellation and mixed-signal filtering with constraints, cartestian (IQ) systems are significantly simpler than polar ones. The primary argued drawback to using a cartesian architecture with switching PAs is a loss in efficiency compared to polar architectures. With linear PAs, the RF IQ input can be combined into a single waveform. However, this voltage summing is not possible for switching PAs expecting digital input waveforms. Traditional cartesian architectures for

switching PAs require separate I and Q PAs which are combined at the output, which sum 90° out of phase.

However, input combining can be implemented by using 25% duty cycle LO signals instead of traditional 50% duty cycle LO signals, as shown in previous works [39][7][32]. I and Q LO waveforms with 25% duty cycle no longer overlap, so the frequency translated I, Q data can be combined in time by using a simple digital OR gate (Fig. 2.18). This final waveform is fed to the digital PA, with the input signal looking identical to a polar architecture operating at peak power. This allows the benefits of a cartesian system without the efficiency penalty, at least in the case where the I and Q input codes are equal.

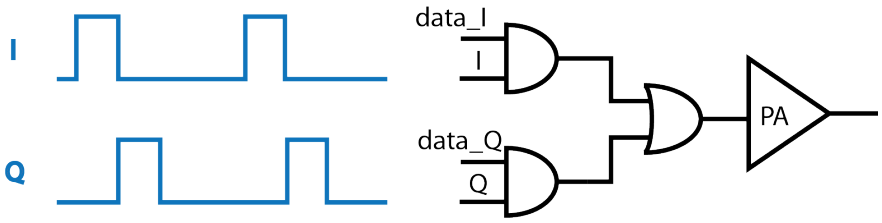


Figure 2.18: IQ input combining scheme with 25% duty cycle

In order to compute the total efficiency η_{tot} of the two cases, we will examine the case where we take two identically sized PAs, but drive one each using 25% and 50% duty cycle LOs. This setup is shown in Fig. 2.19. The PAs are split into 2 smaller PA halves, with $\phi = 0$ for the 25% case and $\phi = \pi/4$ for the 50% case. The max power case in which the same data is sent for I and Q will be analyzed. A 1:1 series stacked transformer is used to implement summation, and the PA drives a load resistance R_L.

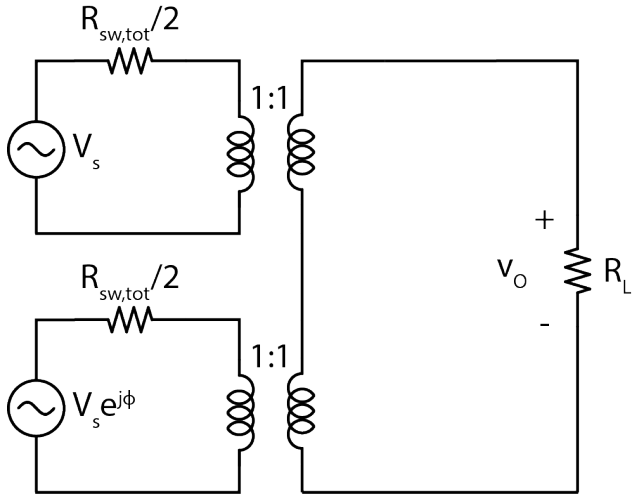


Figure 2.19: Schematic for computing TX efficiency

The total efficiency η_{tot} is defined in Eq. 2.31 and considers loss from the PA, its final

drivers, and the output network. $P_{L,R}$ and $P_{L,C}$ represent resistive and capactive losses respectively. The next step is to compute these terms as well as P_O using this model.

P_O is defined in Eq. 2.32, where $I_{O,rms}$ is the root-mean-square (RMS) current flowing through the load resistance. $P_{L,R}$ comes from sinusoidal current $I_{R,rms}$ flowing through the power transistors and dissipating power, with the total effective resistance for power dissipation represented by $R_{sw,tot}$. Due to transformer coupling, we can relate $I_{R,rms}$ to $I_{O,rms}$ with a scalar a, since some fraction of the load current will flow through the power transistors. $R_{sw,tot}$ and R_L can be related with a constant b, resulting in Eq. 2.33 for $k = a^2b$. $P_{L,C}$ is represented in the typical switching loss equation shown in Eq. 2.34, assuming a supply voltage of V_{DD} for each PA half and a total effective switching capacitance of $C_{sw,tot}$.

$$\eta_{tot} = \frac{P_O}{P_O + P_{L,R} + P_{L,C}} \tag{2.31}$$

$$P_O = I_{O,rms}^2 R_L \tag{2.32}$$

$$P_{L,R} = I_{R,rms}^2 R_{sw,tot} = a^2 I_{O,rms}^2 R_{sw,tot} = a^2 b I_{O,rms}^2 R_L = k P_O \tag{2.33}$$

$$P_{L,C} = \alpha C_{sw,tot} V_{DD}^2 f_{LO} \tag{2.34}$$

We will look at the ratio of η_{tot} for the two cases (Eq. 2.35). Several observations can be made to simplify this expression. Ignoring nonlinearity, $R_{sw,tot}$ and $C_{sw,tot}$ should be same in both the 25% and 50% cases since the PA and drivers are sized exactly the same. In both cases, the capacitors are switched once per cycle, meaning α is identical. The two PAs are also operating using the same V_{DD}. These factors mean that the capacitive losses in the both cases are the same, i.e. $P_{L,C,50} = P_{L,C,25} = P_{L,C}$. Since a is set by the output network, we have $k_{50} = k_{25} = k$ because a and $R_{sw,tot}$ are the same in both cases. Combining these relations with dividing the numerator and denominator by $(1+k)P_O$ results in the simplified expression in Eq. 2.36.

$$\frac{\eta_{tot,50}}{\eta_{tot,25}} = \frac{P_{O,50}}{P_{O,25}} \frac{P_{O,25}\,(1 + k_{25}) + P_{L,C,25}}{P_{O,50}\,(1 + k_{50}) + P_{L,C,50}} \tag{2.35}$$

$$\frac{\eta_{tot,50}}{\eta_{tot,25}} = \frac{1 + k + \frac{P_{L,C}}{P_{O,25}}}{1 + k + \frac{P_{L,C}}{P_{O,50}}} \tag{2.36}$$

Using the model in Fig. 2.19, and knowing that the PA halves sum 90° out of phase in the 50% duty cycle case, we can compute $P_{O,25} = 2P_{O,50}$ (Eq. 2.37). The general result of $P_{O,25} > P_{O,50}$ can be observed by looking at the output waveforms of the 25% and 50% duty cycle cases (Fig. 2.20). The outputs in this figure ignore the output filtering network for the sake of visual clarity, but the general conclusion remains true since any linear filtering will scale the output signal by the same factor.

$$\frac{P_{O,25}}{P_{O,50}} = \left| \frac{2}{1 + e^{-j\frac{\pi}{2}}} \right|^2 = 2 \tag{2.37}$$

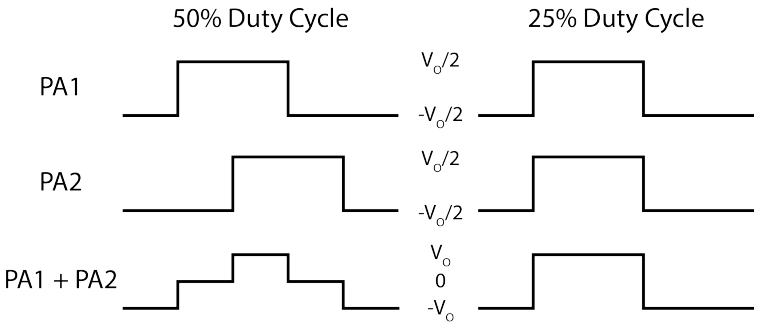


Figure 2.20: Output waveforms of PAs using 25% and 50% LOs

Knowing $P_{O,25} > P_{O,50}$, we can derive the final result of $\eta_{tot,50} < \eta_{tot,25}$ through the steps shown in Eq. 2.38. More detailed analysis in Section 2.7.1 later demonstrates that this analysis holds true for the SCPAs using this configuration.

$$P_{O,25} > P_{O,50} \implies 1 + k + \frac{P_{L,C}}{P_{O,25}} < 1 + k + \frac{P_{L,C}}{P_{O,50}} \implies \frac{\eta_{tot,50}}{\eta_{tot,25}} < 1 \tag{2.38}$$

Though the use of 25% duty cycle LO waveforms increases the peak efficiency, it also comes with some drawbacks. Most obviously, these 25% duty cycle waveforms need to be generated. All phases of the 25% duty cycle I, Q can be generated using the PI generated 50% duty cycle I, Q LOs and their complements in combination with simple logic, as shown in Eq. 2.39. A less obvious but significant issue causes by the use of this technique is that it can degrade the linearity of the TX constellation.

$$I_{25} = I_{50} \cdot \overline{Q_{50}}, \qquad \overline{I_{25}} = \overline{I_{50}} \cdot Q_{50}, \qquad Q_{25} = I_{50} \cdot Q_{50}, \qquad \overline{Q_{25}} = \overline{I_{50}} \cdot \overline{Q_{50}} \tag{2.39}$$

2.6.1 Nonlinearity from 25% Duty Cycle LOs

Ideally, the output amplitude of a 50% duty cycle waveform should be 3 dB (or $\sqrt{2}$) higher than the amplitude of a 25% duty cycle waveform. However, this is not necessarily true if the PA has a nonzero rise/fall time, leading to AM-AM distortion which distorts the output constellation. Fig. 2.21 shows one case in which this can arise. Ideally, the outputs of the IQ case should match the sum of the separate I and Q cases. The IQ output, denoted I+Q (actual), should ideally have the small dip in I+Q (ideal). In practice, the IQ output will have this dip partially or completely filled in due to circuitry not switching instantly, whether it comes from the IQ combining or the SCPA output.

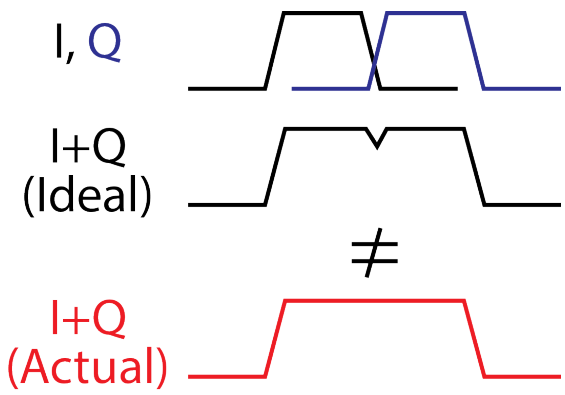


Figure 2.21: 25% / 50% duty cycle nonlinearity

This is further exacerbated by the fact that the rise time of the drain of the SCPA transistor devices fundamentally varies across input code, even without considering parasitic capacitance. Time domain expressions for the voltage at the SCPA "drain", V_D, can be computed using the model in Fig. 2.2. This results in the piecewise expression in Eq. 2.40, where τ is defined in Eq. 2.41. These equations make the assumption that $R_p = R_n$. The input code n sets the initial value in each time region, with higher values of n resulting in initial values closer to the final value (either 0 or V_{DD}). From this, it's clear that the rise / fall times depends on and is reduced with increasing n.

$$V_D(t) = \begin{cases} V_{DD}\left[1 - \left(\frac{1-e^{-\frac{T}{2\tau}}}{1-e^{-\frac{T}{\tau}}}\right)\left(\frac{N-n}{N}\right)e^{-\frac{t}{\tau}}\right] & 0 < t \leq \frac{T}{2} \\ V_{DD}\left(\frac{1-e^{-\frac{T}{2\tau}}}{1-e^{-\frac{T}{\tau}}}\right)\left(\frac{N-n}{N}\right)e^{-\frac{t}{\tau}+\frac{T}{2\tau}} & \frac{T}{2} < t \leq T \end{cases} \tag{2.40}$$

$$\tau = RC \tag{2.41}$$

Another way to view this is by looking at the frequency-domain expression of V_D (Eq. 2.42). There is a zero present in $V_D(s)$, which helps to speed up the rise / fall time by setting an initial condition. The dependence of this zero on the input code n means that the SCPA rise / fall time also depends on the input code n. From this it is clear that even if the pull-up and pull-down resistances are perfectly matched, there will still be variation in rise time across input code.

$$V_D = \frac{1 + s\frac{n}{N}RC}{1 + sRC}V_S \tag{2.42}$$

This analysis ignores the effects of both a load and drain capacitance for the sake of simplicity, but the general trend holds true even when both of these are included. This was verified in a transistor-level simulation with the SCPA operating at an LO frequency of 1.4GHz, simulated with extracted transistor level unit cells, an extracted transformer model, and an ideal 50Ω load. The drain voltage for two codes across a single period is shown in Fig. 2.22, and the rise and fall times across codes are shown in Fig. 2.23. Though the rise / fall

times are not strictly monotonic, the general trend is that the rise / fall times decrease with higher input code. More significantly, the variation of rise and fall times across input codes remains true even with the addition of the extra elements of parasitic drain capacitance, inductance, and a load.

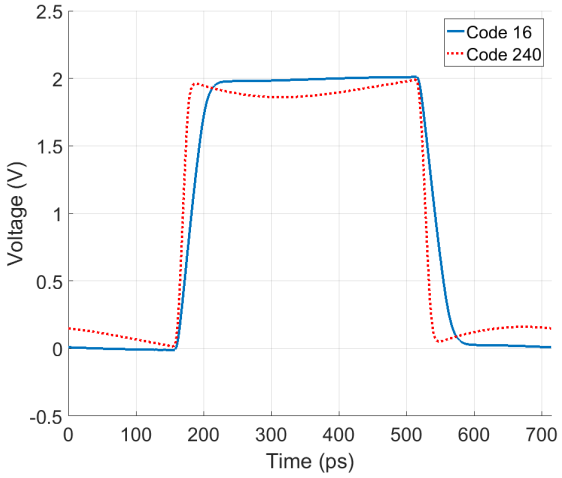


Figure 2.22: Simulated SCPA drain voltage vs time for two codes

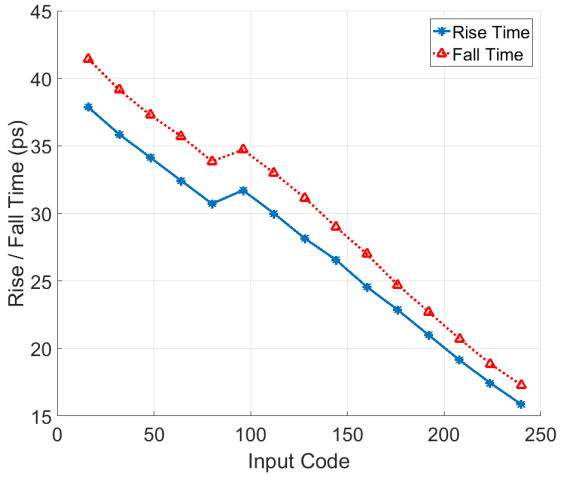


Figure 2.23: Simulated SCPA drain rise and fall time across codes

There are a variety of metrics that can be used to quantify the distortion of the constellation such as EVM, but we will examine the static ratio of the amplitudes of IQ / I across input codes n. This is the ratio of the output power of the PA when transmitting $I = n, Q = n$ to the output power when the PA is transmitting $I = n, Q = 0$. Ideally, this ratio should be $\sqrt{2}$, or about 3.01 dB. We find that this varies significantly across code, but

it can be controlled by sizing to some degree, particularly by changing the ratio of cascode to input device width for both the NMOS and PMOS stack. The IQ / I vs code is shown in Fig. 2.24 for different sizing ratios, where we see a variation of IQ / I of about 0.7 dB for the equally sized case, and a variation of 0.4 dB when the cascodes are 3x wider than the input devices. This is a relatively large increase in area and capacitance for a small improvement of 0.3 dB, so we ended up to keeping the cascode and input devices sized the same for the sake of efficiency.

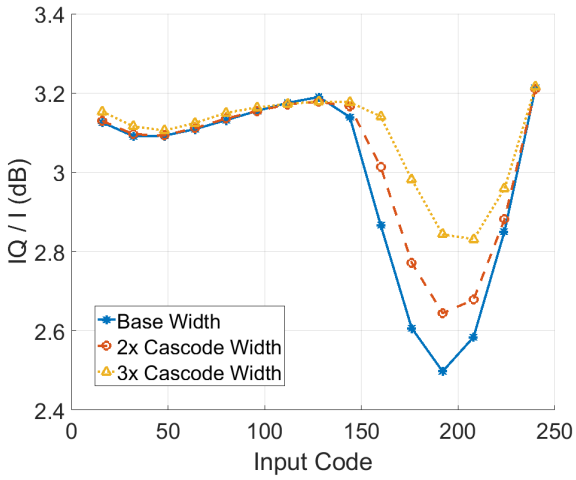


Figure 2.24: Simulated IQ / I vs codes for different sizing

Though we can flatten the IQ / I vs code curve, the curve may not be centered around 3 dB. This is further exacerbated by process variation, with IQ / I across input codes plotted for multiple corners in Fig. 2.25. We see that different corners will generally maintain the same shape but shift the curve up and down, with the exception of the SF corner. The IQ / I ratio is expected to change across corners as this will affect the rise / fall times, which we have argued affects IQ / I.

A tunable solution with sufficient range is necessary to set IQ / I to 3 dB across corners. One method to control IQ / I is to modify the duty cycle of the nominally 25% LOs. Assuming we start from 25%, a 1% increase in the LOs leads to a 1% increase in the IQ case but a 2% increase in the I+Q case. Since the amplitudes of the I and IQ cases are modified by different amounts, the IQ / I ratio will change.

The effect of modifying the duty cycle can be more thoroughly analyzed by computing the fundamental component of square waves with varying duty cycles, to model the ideal output of the SCPA. The square waves are assumed to have infinitely fast edges to simplify the analysis. The fraction of the duty cycle modified is represented with ϵ. The I, Q, and IQ waveforms are shown in Fig. 2.26 under both cases of $\epsilon \leq 0$ and $\epsilon > 0$.

For the nominally 25% (I) case, a single expression for the amplitude of the k^{th} harmonic can be computed (Eq. 2.43). The 50% (IQ) case is more complicated, and the two cases

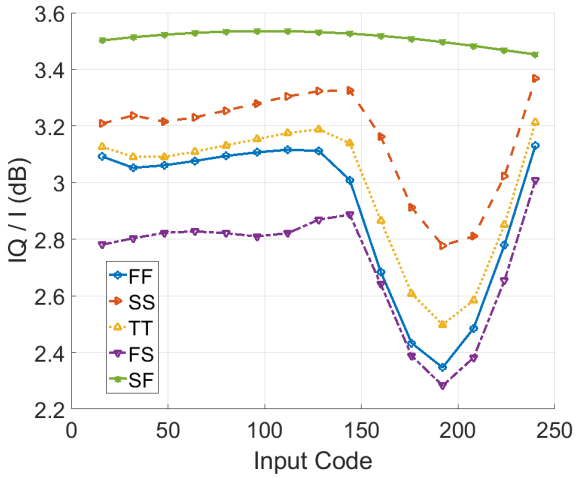


Figure 2.25: Simulated IQ / I vs codes for across corners

of $\epsilon \leq 0$ and $\epsilon > 0$ must be considered separately. The main difference here is that the IQ waveform is exactly the sum the the I and Q waveforms when $\epsilon < 0$, which is not true for the $\epsilon > 0$ case. The overlapping portion does is not doubled in amplitude since the signals are combined using a logical OR gate - there exists only two digital voltage levels. The final expression for the k^{th} harmonic in the IQ case is shown in Eq. 2.44, which is valid for $-0.25 \leq \epsilon \leq 0.25$ and odd values of k.

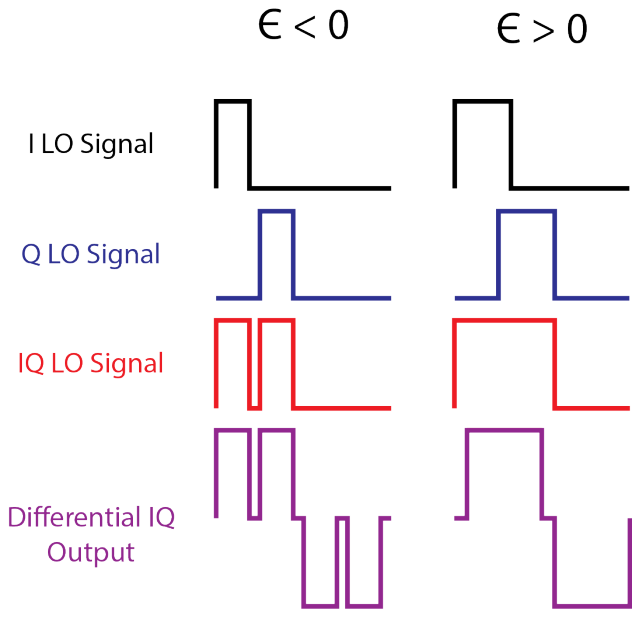


Figure 2.26: I, Q, and IQ waveforms with duty cycle control for $\epsilon < 0$ and $\epsilon > 0$

$$V_{I,k} = \begin{Bmatrix} \frac{\alpha}{j\pi k}\left(1 + j^k e^{-j2\pi\epsilon k}\right) & k\ odd \\ 0 & k\ even \end{Bmatrix} \tag{2.43}$$

$$V_{IQ,k} = \begin{Bmatrix} \frac{\alpha}{j\pi k}\left(1 - j^k\right)\left(1 + j^k e^{-j2\pi\epsilon k}\right) & \epsilon \leq 0,\ k\ odd \\ \frac{\alpha}{j\pi k}\left(1 + e^{j2\pi\epsilon k}\right) & \epsilon > 0,\ k\ odd \\ 0 & k\ even \end{Bmatrix} \tag{2.44}$$

$$\left|\frac{V_{IQ,k}}{V_{I,k}}\right| = \begin{Bmatrix} \sqrt{2} & \epsilon \leq 0,\ k\ odd \\ \sqrt{\frac{[1+cos(2\pi\epsilon k)]^2 + sin^2(2\pi\epsilon k)}{\left[1-(-1)^{\frac{k+1}{2}} sin(2\pi\epsilon k)\right]^2 + cos^2(2\pi\epsilon k)}} & \epsilon > 0,\ k\ odd \end{Bmatrix} \tag{2.45}$$

The IQ / I ratio can be computed by taking $|V_{IQ,k}/V_{I,k}|$ (Eq. 2.45). The IQ / I ratio has the ideal value of $\sqrt{2}$ for $\epsilon \leq 0$. This fits with the observation that the IQ waveform is exactly equal to the sum of the I and Q waveforms in this case. Given this, the exact value of ϵ should not affect IQ / I when $-0.25 < \epsilon \leq 0$. This contrasts with the $\epsilon > 0$ case, which is reflected in IQ / I being a function of ϵ.

In practical cases where the fall and rise times are non-zero, the formula in Eq. 2.45 for $\epsilon > 0$ is valid for small negative values of epsilon due to the circuitry not having sufficient time to return to zero. This assumption is used to generate the plot in Fig. 2.27, which shows how IQ / I (Eq. 2.45) varies as we modify the duty cycle by ϵ given this assumption. The duty cycle control has a strong effect, with a 4% change in duty cycle causing a 1.1 dB change in IQ / I.

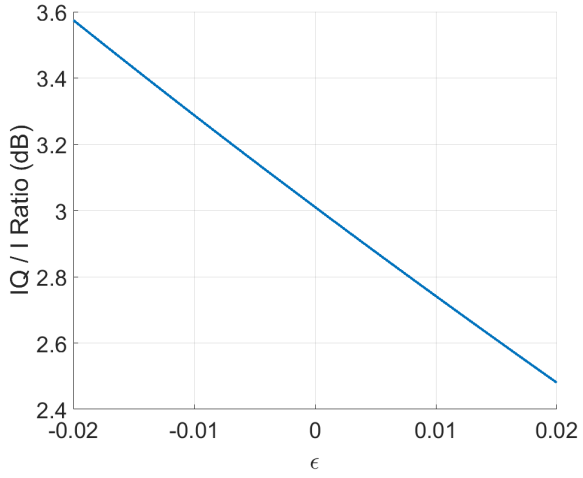


Figure 2.27: IQ / I ratio vs duty cycle modification ϵ

To verify these calculations, the effect of the duty cycle control on the IQ / I ratio was simulated with the overall SCPA using layout extracted unit cells assuming ideal duty cycle

control (Fig. 2.28). Simulations with duty cycle variation of -2% and $+2\%$ correspond to a change in IQ / I of 0.585 dB and -0.531 dB respectively, averaged across input codes. This matches closely with the calculations in Eq. 2.45, which expect changes in IQ / I of 0.565 dB and -0.53 dB respectively. The change in IQ / I for different duty cycle settings is relatively constant across codes, only shifting the IQ / I vs input code curve up and down but not changing its shape.

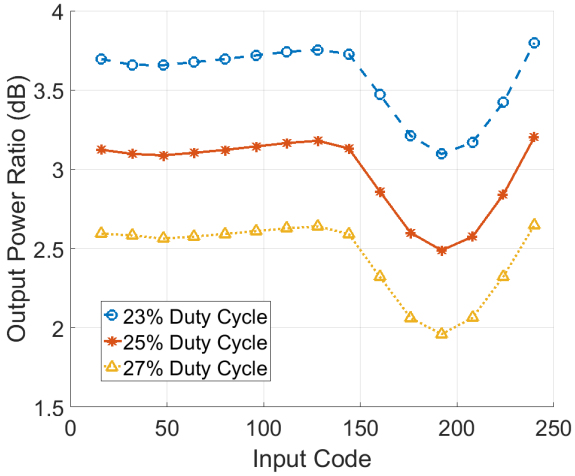


Figure 2.28: Simulated IQ / I ratio across input code

The duty cycle control was designed and implemented by Eric Chang. The block was integrated into the 25% LO generation circuitry, with the circuit shown in Fig. 2.29. The high level idea is to slow down one edge of the LO signal, moving the midpoint of that edge. Since the duty cycle is defined from midpoint to midpoint for the two edges, further slowing the edge will extend the duty cycle. This signal drives a buffer chain to generate a final LO signal with low rise and fall times, with the first stage being a skewed inverter to even out the rise and fall times. This does have the drawback of introducing additional phase noise since we are purposely slowing down one of the clock edges.

The rising edge is slowed down by current starving the pull up network, with the bias voltage set using a current mirror structure fed by a bias current DAC (IDAC) with both coarse and fine control. The IDAC is set using the scan chain, since this is a "DC" setting which only needs to be modified when the f_{LO} is changed. The IDAC should be designed with enough range and resolution to ensure fine duty cycle control.

A plot the tuning range of the duty cycle control over a small code range is shown in Fig. 2.30. The worst case tuning ranges from -3 ps to 200 ps which correspond to a -0.2% to 14% change in the duty cycle. The resolution is not evenly spread out across duty cycle extension, with finer resolution at lower duty cycle extensions. This is reflected in the plot with the steps becoming finer for higher codes. This chapter will now analyze the sizing of the SCPA

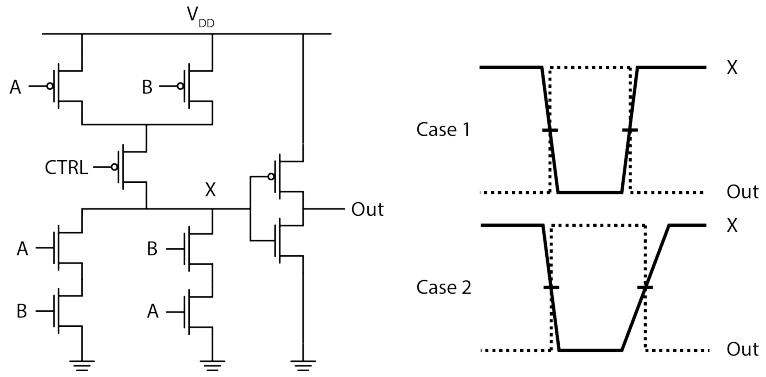


Figure 2.29: Duty cycle control circuit with waveform diagrams

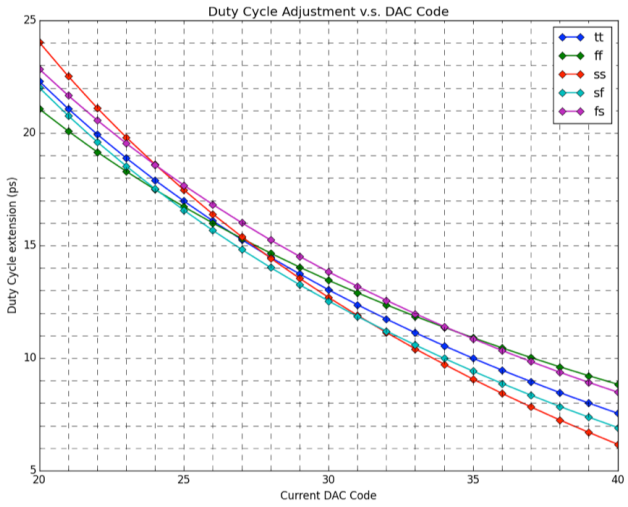


Figure 2.30: Duty cycle extension versus code

and TX since system level topics such as the filtering techniques and TX architecture have been discussed.

2.7 SCPA and Output Network Design

This section will discuss how SCPA and TX parameters are chosen to meet high level specifications. The high level specifications consist of peak output power P_O, center frequency f_{LO}, bandwidth f_{BW}, and HD3 reduction. Mixed-signal filtering is also implemented but harmonic cancellation is the primary focus. First, the output network will be partially designed to meet f_{LO} and f_{BW} specifications. Afterwards, remaining values for the passives components in the output network and the SCPA sizing are chosen to meet the P_O and HD3 reduction requirements. This section relies on simulations using layout extracted schematics in order to make the design as close as possible to the measurement results of the fabricated

prototype. The numbers used for various parameters come from a representative technology, but the general methodology in this section translates across CMOS processes.

The output network consists of the transformer and SCPA capacitors. The transformer provides the tank inductance to resonate at the desired f_{LO} as well as impedance transformation to increase P_O. The transformer presents both a primary side inductance L_p, which can be transformed to a secondary side inductance L_s. Here, we are defining the primary to be the side connected to the SCPA arrays, while the secondary side is connected to the load.

The secondary side inductance L_s is first chosen to meet the f_{BW} specification, written in terms of Q using Eq. 2.47. Once L_s has been computed, we can compute L_p. For a 1:n transformer, $L_s = n^2 L_p$ ideally, but isn't exactly true due to leakage inductance. In practice, the transformer will be laid out and extracted once L_s and n are known, with the value of L_p simulated using this extracted model. The total series capacitance for each PA array C_s can be computed using Eq. 2.48 based on the primary side inductance L_p.

$$Q = \frac{f_{LO}}{f_{BW}} = \frac{R_L}{2\pi f_{LO} L_s} \tag{2.46}$$

$$L_s = \frac{R_L f_{BW}}{2\pi f_{LO}^2} \tag{2.47}$$

$$C_s = \frac{1}{\omega_{LO}^2 L_p} \tag{2.48}$$

The SCPA unit cell capacitance C_{unit} can be computed from C_s by dividing by twice the total number of effective thermometer cells in the overall SCPA. The extra factor of two comes from us using a pseudo-differential SCPA. This causes us to see two unit capacitors in series, cutting the effective capacitance in half. Once we've computed C_{unit}, we can use this generate the SCPA unit cell capacitor, extract it, and get an estimate of the bottom plate capacitance of the unit cell capacitor. The bottom plate capacitance on the output side will add to the capacitance seen by the transformer, which affects f_{LO}.

Our final design has $L_p = 620pH$, $L_s = 2.64nH$, and $C_{unit} = 310fF$, corresponding to a total $C_s = 39.8pF$ per SCPA half. These numbers set $f_{LO} \approx 1.4GHz$, $Q \approx 2$, and $f_{BW} \approx 700MHz$.

For our TX design, the primary goals are meeting a specific target output power P_O and ensuring a certain amount of harmonic cancellation while maximizing efficiency η under these specifications. Other important specifications like EVM and ACLR exist but the aforementioned specifications will be the focus of the design. We will first discuss sizing for maximum system efficiency η.

2.7.1 SCPA Design for Efficiency

A model is needed to estimate the output power P_O and resistive losses, which is shown in Fig. 2.31. This model assumes operation at the center frequency f_{LO}, in which all reactive

components have been resonated out. This model also assumes perfect transformer coupling ($k = 1$). If we wish to model imperfect coupling, it can be grouped into the PA amplitude aV_{DD}. This model sums k_{PA} PA arrays together using transformer combining, with each PA array having output phase ϕ_i.

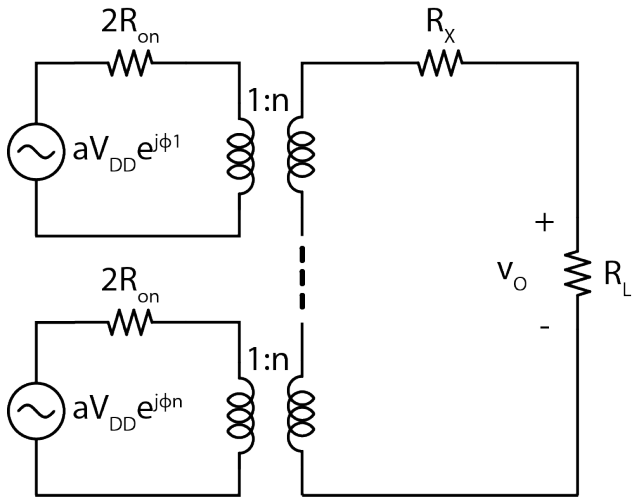


Figure 2.31: Schematic to compute output current and resistive losses

$$I_O = \frac{an\gamma_\phi V_{DD}}{R_L + R_{XMFR} + 2n^2 k_{PA} R_{on}} = \frac{an\gamma_\phi V_{DD}}{R_L + R_{XMFR} + 2n^2 k_{PA}\left(\frac{R_u}{W}\right)} \tag{2.49}$$

$$\gamma_\phi = \left| \sum_{i=1}^{k_{PA}} e^{j\phi_i} \right| \tag{2.50}$$

Critical design parameters of the TX consist of the transistor widths, transistor channel lengths, output amplitude per PA array aV_{DD}, the transformer turns ratio n, the number of PAs power combined k_{PA}, and PA array phases ϕ_i. This is a large set of variables to optimize over, but we can reduce the number of free variables by setting values which will maximize efficiency or are set due to other specifications or desired functionality.

System level considerations set or restrict both ϕ_i and k_{PA}. The harmonic cancellation technique used in this TX sets two PA arrays with phase shifts of $\phi_0 = 0°$ and $\phi_1 = 60°$ to cancel the 3rd harmonic. This sets k_{PA} to be a multiple of 2, and combined with the phase shifts, sets $\gamma_\phi = \sqrt{3} * k_{PA}/2$.

The minimum channel length L_{min} for a given device type (thin oxide, thick oxide, etc.) is chosen to minimize transistor on-resistance R_u and capacitance C_u. Additionally, we want to match the pull-up and pull-down resistance as shown by the analysis in Section 2.5. All NMOS transistors in the stack have width W, and PMOS transistors have width W_p. The use of the same width is to reduce the size of the design space. W_p is related to W with

PN ratio α_{pn} such that $W_p = \alpha_{pn}W$. These factors reduce transistor size optimization to a single variable W, representing the width of the NMOS pull down network.

This leaves a, V_{DD}, n, k_{PA}, and W as the free variables left to optimize over. We'll first analyze P_O and how it is impacted by these parameters. This can be done by first computing the load current I_O, which is related to the output power by $P_O \propto I_O^2$. The expression for I_O in 2.49 demonstrates that the critical parameters for setting P_O are the numerator terms consisting of aV_{DD}, n, and k_{PA}. A $P_O \approx 26dBm$ was targeted for TX to demonstrate the effectiveness of harmonic cancellation at power levels for handheld cellular specifications. There are a variety of tradeoffs in choosing different sets of parameters, but we have chosen $a = 8/\pi$, $V_{DD} = 1V$, $k_{PA} = 2$, $n = 2$ to allow for a maximum $P_O = 28.9dBm$. Each decision is explained further in the following paragraphs.

The overall output of each PA array is set by choosing V_{DD} and a. V_{DD} corresponds to the gate swing of the input devices of the SCPA. The main choice in our technology was between standard thin oxide transistors with $V_{DD} = 1V$ and thick oxide transistors with $V_{DD} = 2V$. a is the output amplitude as a function of V_{DD}, and is affected by the PA topology (including single ended / differential) and PA supply voltage $V_{DD,high}$.

Increasing the supply voltage will increase a but can cause device breakdown. The primary mechanism is gate oxide breakdown [40], which occurs when $|V_{GS}| > V_{break}$ or $|V_{GD}| > V_{break}$, where V_{break} is a technology specific parameter. Device stacking has been used in CMOS PAs to limit the maximum $|V_{GS}|$ and $|V_{GD}|$ [41][42][9][43] of each device to allow for operation with a higher $V_{DD,high}$, while thick oxide transistors have a larger V_{break} than thin oxide transistors.

A single thick oxide transistor would allow us to reach the same effective supply as that of a stack of 2 thin oxide devices, henceforce referred to a 2-stack. However, the parasitics of the thick oxide transistors are significantly worse, being much lossier at our desired center frequency f_{LO} than even the 2-stack of thin oxide devices. The thick oxide transistors also have the additional drawback of requiring thick oxide drivers, which necessitates level shifting between thin oxide logic and these drivers. Stacking also requires level shifters, but these are significantly simpler since only a level shift is required and not a change in voltage swing. These two factors drove the decision to use a 2-stack of thin oxide transistors with $V_{DD} = 1V$ and $V_{DD,high} = 2V_{DD}$, giving us $a = 2 \cdot 4/\pi = 8/pi$.

k_{PA} and n both influence the complexity of the transformer design, with k_{PA} increasing the total number of ports of the transformer. The minimum $k_{PA} = 2$ is chosen to keep the transformer as simple as possible, with the n chosen based on final output power requirements. The choice of $n = 2$ also helps reduce the overall area of the transformer, since the secondary side transformer inductance L_s will be proportional to the loop area and n^2.

Though transistor width W also influences P_O, the presence of R_L and R_X significantly reduce its effect when $R_L + R_X >> R_u/W$. This makes it generally less significant than aV_{DD}, n, and k_{PA} for setting P_O unless W is small enough that R_u/W is comparable to $R_L + R_X$. There will always be loss incurred by both the PA transistors and the transformer, so our parameters choices give us a couple dB of headroom to achieve $P_O \approx 26dBm$. Once aV_{DD}, n, and k_{PA} are set, the main parameter left is W, with an optimal value balancing resistive

and capacitive losses to maximize efficiency [9]. Balancing these two losses is a common theme in optimizing switching PAs for efficiency.

$$\eta = \frac{P_O}{P_O + P_X + P_{L,R} + P_{L,C}} \tag{2.51}$$

$$R_{on} = \frac{R_u}{W} \qquad C_{sw} = WC_u \tag{2.52}$$

$$\eta = \frac{\frac{I_O^2 R_L}{2}}{\frac{I_O^2(R_L+R_X)}{2} + n^2 I_O^2 k_{PA} R_{on} + 2k_{PA} V_{DD}^2 f_{LO}\left(C_{sw} + C_P\right)} \tag{2.53}$$

R_{on} is the switch on-resistance for one differential half of a PA array, while C_{sw} is the effective capacitance for one differential half of a PA array referenced to V_{DD}^2. C_{sw} is computed by dividing the total capacitive loss by $V_{DD}^2 f_{LO}$. The per width versions, R_u and C_u, are required to compute the optimal width W_{opt} (Eq. 2.52). These have units of $\Omega \cdot \mu m$ and $fF/\mu m$ respectively. These will be taken as set parameters for now, with the method for computing them discussed later.

The final remaining unknown variable in Eq. 2.53 is C_p, which represents any parasitic shunt capacitance in the PA array. This includes bottom plate capacitance for the SCPA capacitor, wiring capacitance between blocks, and routing channel parasitic capcacitance.

Making the simplifying assumption that $C_{sw} >> C_p$ results in a relatively simple closed form expression for the optimal width W_{opt} (Eq. 2.55) as well as the maximum total efficiency η. This simplified case will be analyzed for design insight, though the design algorithm which is later proposed will include C_p. W_{opt} and η are written in terms of β_η, defined in Eq. 2.56. β_η is a function of the number of PA arrays and how in phase their summation is, as well as the LO frequency f_{LO}.

$$\eta = \frac{R_L}{R_L + R_X + 2n^2 k_{PA} R_{on} + \frac{4k_{PA} f_{LO} C_{sw}}{a^2 n^2 \gamma_\phi^2}\left[R_L + R_X + 2n^2 k_{PA} R_{on}\right]^2} \tag{2.54}$$

$$W_{opt} = \frac{2n^2 k_{PA} R_u}{R_L + R_X}\sqrt{1+\beta_\eta} \tag{2.55}$$

$$\beta_\eta = \frac{a^2 \gamma_\phi^2}{8k_{PA}^2 f_{LO} R_u C_u} \tag{2.56}$$

$$\eta_{opt} = \frac{R_L}{R_L + R_X} \cdot \frac{1}{1 + \frac{1}{\sqrt{1+\beta_\eta}} + \frac{\sqrt{1+\beta_\eta}}{\beta_\eta}\left[1 + \frac{2}{\sqrt{1+\beta_\eta}} + \frac{1}{1+\beta_\eta}\right]} \tag{2.57}$$

The peak efficiency as a function of β_η is given in Eq. 2.57 and plotted in Fig. 2.32 when $R_X = 0$. If $R_X \neq 0$, the plot will be scaled by $\frac{R_L}{R_L+R_X}$. β_η will increase with lower transistor parasitics and lower operating frequency f_{LO} (Eq. 2.56). It makes intuitive sense that η_{opt}

would increase with increasing β_η, since we would expect a higher peak efficiency for lower transistor parasitics and at lower operating frequencies.

β_η depends on the phase between PA arrays through γ_ϕ, meaning W_{opt} and η also depend on phase. The efficiency vs phase can be plotted assuming some maximum β_η occurring when $\gamma_\phi = k_{PA}$. This is shown for two different maximum β_η values in Fig. 2.33. As we would expect, summing out of phase reduces the maximum achievable efficiency. Interestingly though, this effect is not uniform for different values of β_η, with effect being more severe for smaller values. For example, η decreases from 53.5% to 42% for $\beta_\eta = 10$ but only from 75% to 67% for $\beta_\eta = 50$.

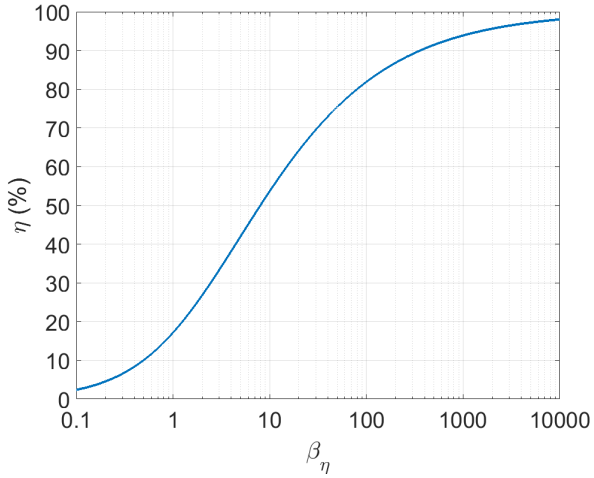
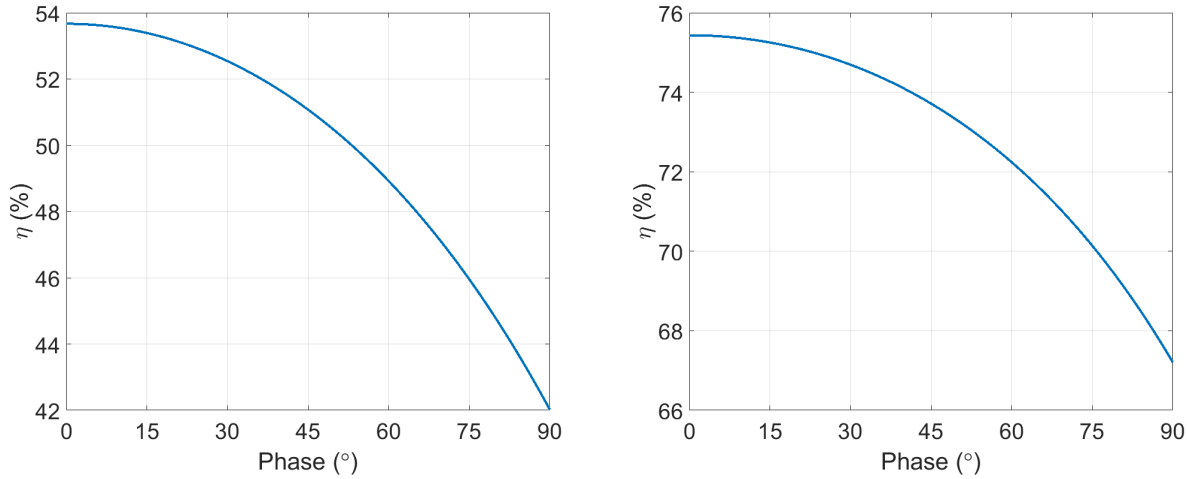


Figure 2.32: Maximum system efficiency vs β_η

A major assumption of the preceding analysis is that R_u and C_u are not functions of W, which was made to simplify the math. This isn't valid since both depend on transistor V_{GS} and V_{DS}, the latter of which has a dependence on W. The optimal width W_{opt} can still be computed by using an iterative design which updates the values of R_u, C_u for each iteration when W_{opt} is found. The loop's convergence will depend on the output drain swing $V_{sw} = I_o R_{on}$, which will set the transistor V_{DS} values. The methods for computing R_u, C_u for each iteration are shown below.

Both R_u and C_u are computed by extracting either the entire SCPA unit cell or portions of it. This is done instead of the extracting the entire SCPA to have a tractable design - extraction and simulation time would be far too large to be useful if the entire SCPA was extracted. Most of R_u and C_u can be captured with unit cells, but complete routing channels parasitics will not be properly captured. The layout generation of the unit cell is discussed in detail in Chapter 3.

The value of R_u is simulated using the extracted SCPA unit cell power transistors using the schematic shown in Fig. 2.34. The pull-down transistors are sized with width W_{test}. We set $V_{GN} = V_{DD}$ and $V_{GP} = 2V_{DD}$ to only turn on the pull-down network, with the



(a) $\beta_{\eta,max} = 10$ (b) $\beta_{\eta,max} = 50$

Figure 2.33: Maximum system efficiency vs 2nd PA phase for a fixed peak β_η

SCPA output bias set with V_{DS}. We set $V_{DS} = V_{sw}/\sqrt{2}$ to correspond with the rms value of the swing drain node, and the AC resistance is computed with V_{test}/I_{test}. The rms value is chosen to approximate the "average" drain swing across the period. This method is repeated for the pull-up network by setting $V_{GN} = 0$, $V_{GP} = V_{DD}$, $V_{DS} = 2V_{DD} - V_{sw}/\sqrt{2}$. The AC resistances of the pull-down and pull-up networks are averaged to compute R_{on}, and R_u is computed by dividing R_{on} by W_{test}.

The R_u computed takes into account the transistor on-resistance as well as wiring resistance to the SCPA series capacitor. The resistance of the capacitor itself is not modeled, though this can generally be made to be significantly smaller than the transistor on-resistance. Much more significant sources of resistance which are not modeled include the output and supply network. The supply network is difficult to estimate due to the complex connection to the supply grid from outisde the PA and from bumps. The output network could be estimated using layout dimensions and grouped into R_x, but this was not implemented in this version of the design algorithm.

The nonlinearity of the parasitic capacitances makes calculating C_{sw} based on individual voltage swings and capacitance values per unit width tedious and inaccurate. Instead, we simulate a schematic (Fig. 2.35) using the extracted layout schematic of the SCPA unit cell, with the switch pull-down network sized with width W. This will capture most of the voltage swings accurately, except for the output swing. The buffers are sized with a set fanout per stage, with the exact widths depending on the switch device width. The test inverters are required to capture the switching loss of gate capacitance of the first stage, though the test inverter's switching loss will also be included. This switching loss can be decoupled either by using inverters with ideal switches or using a replica to simulate the switching power directly. The total power consumed by all supplies is divided by $V_{DD}^2 f_{LO}$ to

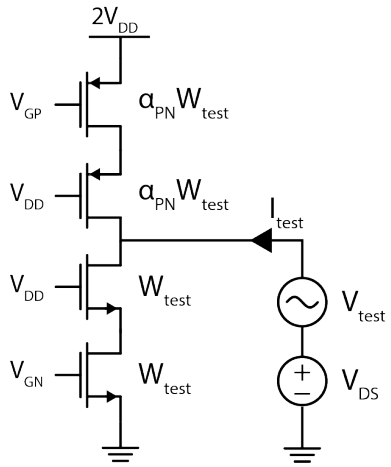


Figure 2.34: Schematic to simulate R_{on}

compute $C_{sw,tot} = C_{sw} + C_p$, and C_u is computed by dividing C_{sw} by W. This still leaves us with two variables to solve for.

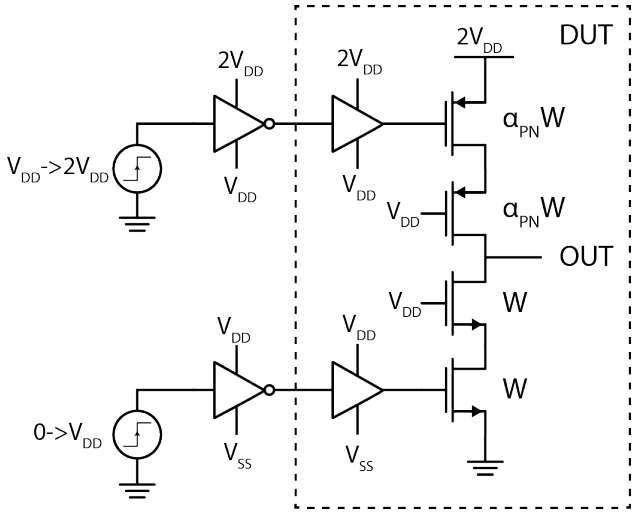
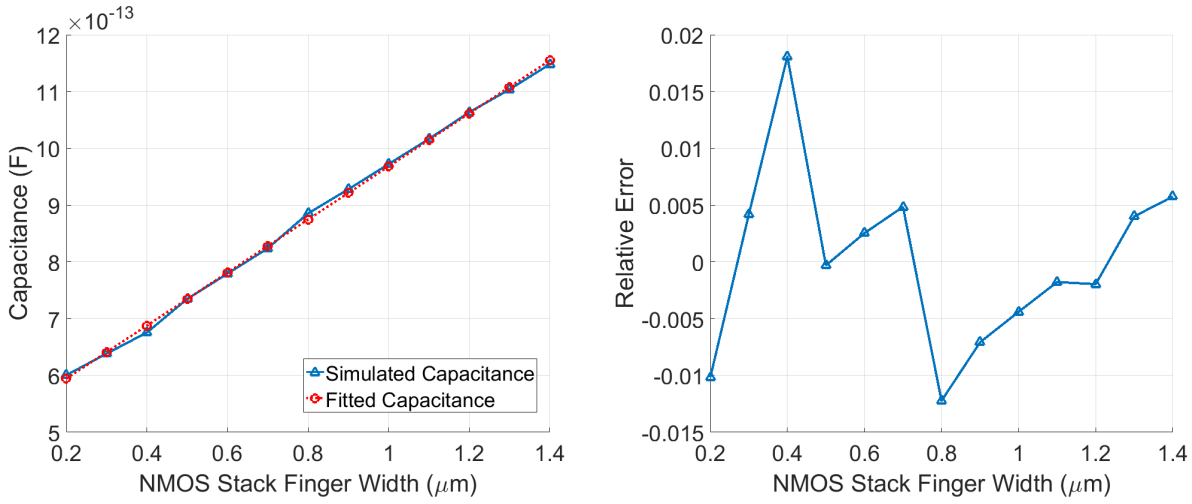


Figure 2.35: Schematic to simulate C_{sw}

$$\eta = \frac{R_L}{R_L + R_X + 2n^2 k_{PA}\frac{R_u}{W} + \frac{4k_{PA}f_{LO}(WC_u+C_p)}{a^2 n^2 \gamma_\phi^2}\left[R_L + R_X + 2n^2 k_{PA}\frac{R_u}{W}\right]^2} \tag{2.58}$$

A custom layout generator for the SCPA unit cell (Chapter 3) has been written, allowing us to easily create several instances of the unit cell layout with different values of W. Each instance is extracted and simulated, and a linear fit is applied to the set of $C_{sw,tot}$ values with W as the independent variable. This fit is in the form of $C_{sw,tot,fit} = WC_u + C_p$, giving



(a) Capacitance (b) Simulated vs Fitted Error

Figure 2.36: Simulated vs Fitted Capacitance vs Width for nf = 32

us our values of C_u and C_p. Fig. 2.36 shows the simulated and fitted capacitance versus W as well as the error between these two. In this case, we fix the number of fingers (nf) and sweep the finger width. The worst case error across the data points is $< 2\%$, making this a relatively good fit. The C_u and C_p should fit well as long as the number of transistor fingers remains constant. If this is satisfied, wires in layout either scale with finger length, such as wires connecting to the drain, or remain constant, such as wires connecting the input devices to the cascode devices. The latter of these normally scale with nf, but will be constant if nf is fixed. If the nf does change, we must repeat the layout generation, extraction, simulation, and linear fit with the new value of nf.

One issue with this method is that the nonlinearity of the shunt capacitors at the output nodes are not properly captured, due to it not having the correct swing in this model. Fig. 2.35 does not capture the sinusoidal current that flows back through the devices. Due to this, this method will tend to overestimate C_u.

As mentioned earlier, the dependence of R_u, C_u on W necessitates the use of an iterative design loop to compute W_{opt}. The SCPA drain swing V_{sw} is the main parameter which is used to check for loop convergence. If the nonlinearity is relatively weak, this loop should converge with only a few iterations. The iteration loop is as follows:

1. Start with a base size (W, nf, etc.) for R_u, C_u, C_p simulations.
2. Start with an estimated drain swing V_{sw}.
3. Simulate R_u assuming this V_{sw} and current width W.
4. Look up value of C_u, C_p in a table if it exists for the current set of transistor nf. If not, simulate C_u, C_p and record this in the table.

5. Compute W_{opt} using R_u, C_u, C_p.
6. Compute I_O using W_{opt} and R_u. Compute the new V_{sw}^* using I_O, W_{opt}, R_u.
7. Check if the newly computed V_{sw}^* is within some tolerance, i.e. 5% - 10%, of the old V_{sw}. If not, repeat steps 3-6.
8. Use the final value of W_{opt} computed.

This entire design algorithm can be implemented in a script as a part of design automation. The simulation of R_u, C_u and C_p can be included in the iteration loop since the unit cell layout is generated using BAG. The extraction and simulation can also be included as a part of the script. The simulations are run using the computed W_{opt} and V_{sw}^* in each iteration. W_{opt} is computed in step 5 using the full expression for η, including the previously omitted C_p (Eq. 2.58). The denominator is differentiated with respect to W and the roots are computed using a numerical solver to find the values of W which minimize the denominator. The smallest positive root is chosen to be W_{opt}. With the optimization for maximum efficiency done, the next step is to analyze the impact of SCPA parameters on linearity.

2.7.2 SCPA Design for Linearity

Linearity of summation is a critical requirement for the effectiveness of the filtering techniques, especially harmonic cancellation, as discussed in Section 2.3. The analysis in Sections 2.1, 2.5 has demonstrated that the SCPA operates linearly assuming the switches themselves are linear, and the pull-up and pull-down resistances are matched. The nonlinearity of the switches, implemented by CMOS transistors, then becomes the limiting factor in linearity of summation.

In order to understand the source of switch nonlinearity, we will examine how the power transistors behave during operation. In ideal operation the drain of the SCPA is a square wave which goes from 0 to $2V_{DD}$. However, the current flowing through the output load to generate the output power P_O also eventually flows back through the SCPA power transistors. This current may be scaled or phase shifted in some fashion due to the output network and parasitics, but some form of it flows through the power transistors. This causes a sinusoidal voltage ripple on top of the ideal square wave at the drain. This voltage ripple increases with increasing current and causes a reduction in linearity, reducing the effectiveness of these filtering techniques. In order to quantify the effect of voltage ripple on resistance, the relationship between the normalized large signal R_{on} and $|V_{DS}|$ is shown in Fig. 2.37. In this plot, all R_{on} curves are normalized to the smallest value of R_{on} across $|V_{DS}|$. From this plot, we can see that the nonlinearity gets more severe with a larger voltage ripple.

The voltage ripple can be reduced by reducing the effective R_{on} of the pull-up and pull down-networks by increasing W. Reducing the voltage ripple will directly reduce the nonlinearity of the R_{on}. The transistor sizing thus has a direct effect on the device linearity and linearity of summation, setting a floor on the maximum attainable cancellation. This

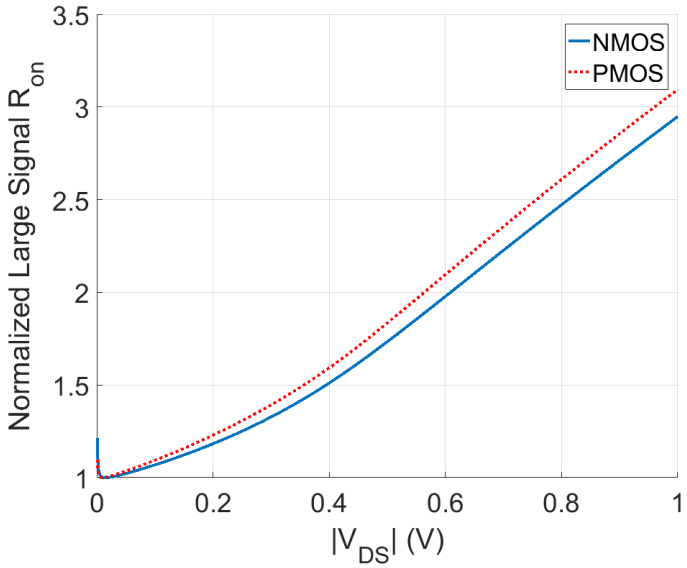


Figure 2.37: Normalized large signal R_{on} vs V_{DS}

in turn sets a constraint on the minimum W, known as W_{min}, in order to achieve a desired cancellation value.

Limited device linearity during operation causes these filtering techniques to be difficult at higher output powers. At higher output powers, the device current and thus the voltage ripple will increase, worsening the linearity. The primary ways to increase output power are to use a higher supply voltage, transform the output impedance, or power combine. Generally, the load current which flows back into the power transistors will be roughly proportional to $\sqrt{P_O}$. An exception is if the output impedance transformation is done only using a transformer, in which case it is roughly proportional to P_O. The transistors must be upsized (W increased) to counteract the degradation in linearity at higher output power levels.

To demonstrate the effects of sizing on cancellation, we simulated three extracted SCPA designs with the same PN ratio and different widths. Computing the best case HD3 reduction from simulated R_{on} is complicated, so we opted to instead directly simulate the SCPA during operation and sweep the phase shift and find the maximum HD3 reduction. The resolution of the phase shift is set to be fine enough to ensure that it does not limit cancellation.

The simulations use the layout extracted SCPA unit cells for the same reasons as described in the previous section. These unit cells are arranged into four separate arrays, to form two pseudo differential pairs and to implement the two LO phases. Input signals are generated using a DC input voltage connected to ideal ADCs and binary-to-thermometer decoders implemented in Verilog A. The SCPA arrays drive a layout extracted model of the output transformer, which was generated using EMX [44]. This transformer is connected to an ideal 50Ω load on the other side. Fig. 2.38 shows the resulting simulated HD3 vs phase

Table 2.2: Simulated HD3 reduction for different sizings

Size	HD3 reduction
1	46.8 dB
0.5	19.6 dB
0.25	11.4 dB

Table 2.3: Simulated peak drain voltage swing for NMOS and PMOS stacks

Size	NMOS V_{sw} (from V_{SS})	NMOS V_{sw}	PMOS V_{sw} (from $2V_{DD}$)	PMOS V_{sw}
1.0	80 mV	73 mV	70 mV	65 mV
0.5	323 mV	273 mV	276 mV	236 mV
0.25	722 mV	600 mV	524 mV	431 mV

shift with the SCPAs operating with $f_{LO} = 1.4GHz$.

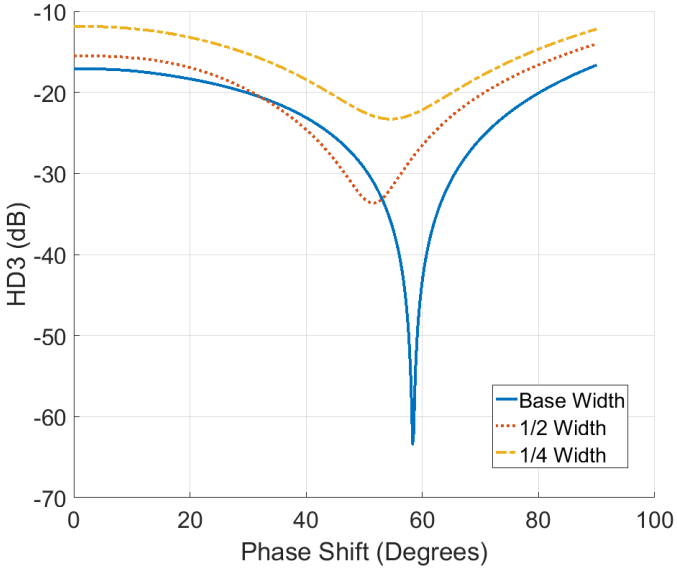


Figure 2.38: Simulated HD3 vs phase shift

There is a significant difference in the effectiveness of the harmonic cancellation between the three sizings, summarized in Tab. 2.2. The cancellation is significantly higher in the base width (1x) case, demonstrating that the choice of transistor width W can significantly limit HD3 reduction. In order to examine the hypothesis that this effectiveness is due to lowered drain swing, we need to examine the drain node of the SCPA. The SCPA operates using $V_{DD} = 1V$ and $2V_{DD} = 2V$.

The periodic drain voltage during operation is shown in Fig. 2.39, using the same simulation setup as the previous plot, but with the two PA arrays operating in phase. The first

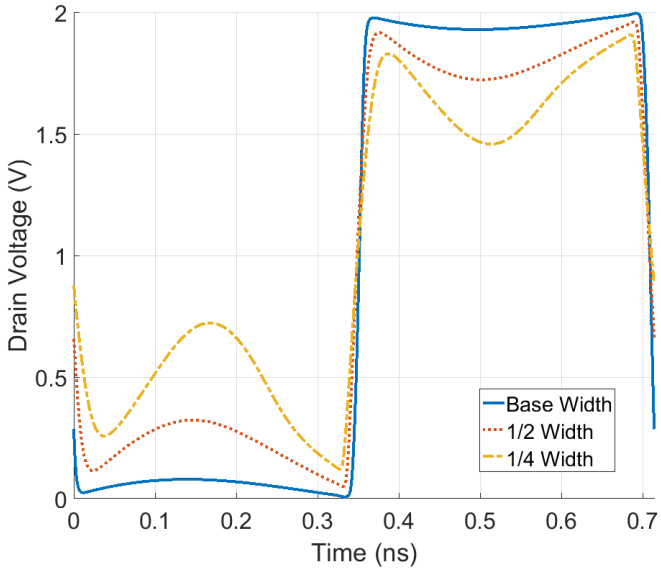


Figure 2.39: Simulated SCPA drain voltage vs time (1 period)

half of the period coincides with pull-down (NMOS) network, while the second half corresponds with the pull-up (PMOS) network. This simulation verifies our hypothesis that larger widths will reduce the voltage swing on the drain node, with the base size having swings of less than 100 mV (Tab. 2.3). The simulations also establish a relationship between lower voltage swing and improved HD3 reduction.

The different transistor types and configurations have different linearity characteristics. In Section 2.7.1, 2-stacks of thin oxide transistors and thick oxide transistors were compared as options for increasing output power. Fig. 2.40 compares the large signal normalized R_{on} across drain swing V_{DS} for 2-stack and thick oxide NMOS and PMOS transistors. The transistors or stacks are sized to have the same R_{on} at $V_{DS} \approx 0V$, assuming normal gate bias.

The thick oxide transistors have better linearity than the 2-stacks, with the PMOS being significantly more linear. However, both NMOS configurations and the PMOS stack display similar linearity characteristics for $V_{DS} < 200mV$. From Tab. 2.2 and 2.3, $V_{sw} < 80mV$ is required for an HD3 reduction of 47 dB. Since we are targeting HD3 reduction $> 40dB$, the NMOS 2-stack and NMOS thick oxide will have extremely similar linearity performance for our HD3 reduction specifications. Additionally, the R_{on} of the NMOS and PMOS stacks match very well within this region of operation. This contrasts significantly with the NMOS and PMOS thick oxide R_{on} values, which differ significantly across V_{DS} swing. These two factors demonstrate that the 2-stack has comparable or favorable linearity performance to the thick oxide transistors for our targeted specifications.

This section focused on how sizing affects harmonic cancellation through its effects on device nonlinearity, but sizing can also affect harmonic cancellation through mismatch caused

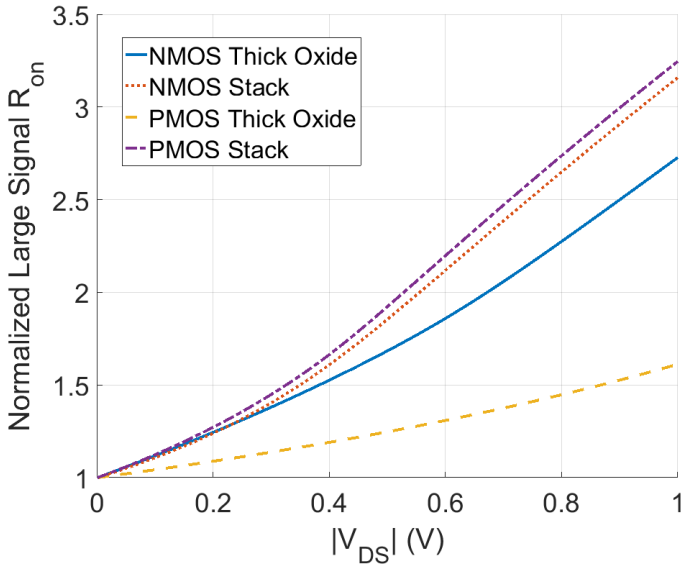


Figure 2.40: Normalized large signal R_{on} vs V_{DS} for thick oxide and 2-stack

by random variation in V_{TH}. This is normally a major concern when designing DACs, however our SCPA differs from most cases in two important ways. First, we choose the maximum possible gate voltage of $V_{GS} = V_{DD}$ for the input devices, and use low threshold devices to maximize $V_{GS} - V_{TH}$ in order to minimize R_{on}. This V_{GS} is significantly higher than what we would expect of other DACs, such as a current DAC. Secondly, the SCPA will generally be sized relatively large based on efficiency and linearity requirements. Using the Pelgrom model [38], we have $0.2\ mV \leq \sigma_{V_{TH}} \leq 2.9\ mV$ for each overall PA array, with the exact value of $\sigma_{V_{TH}}$ depending on input code. Even the worst case of $6\sigma_{V_{TH}} = 17.4mV$ is still at least 30x smaller than our overdrive voltage $V_{GS} - V_{TH}$. These two factors combine to make the drain-swing induced nonlinearity the dominant source of nonlinearity when sizing the SCPA, and is the reason we have mostly ignored the impact of V_{TH} variation.

The final choice of SCPA nmos stack width W is will be the larger of W_{min} and W_{opt}. The SCPA efficiency will be degraded in the case where $W_{min} > W_{opt}$. We can compute the efficiency as a function of W (Eq. 2.58) to quantify the impact of choosing $W > W_{opt}$. A plot of efficiency vs W is shown in Fig. 2.41, with R_u, C_u, and f_{LO} values resulting in $\beta_\eta = 31.5$. This plot uses the formula in Eq. 2.57. The width shown in this plot corresponds to the width of the pull-down network of each differential half of a PA array. For these parameters, $W_{opt} = 1416\mu m$ with a corresponding $\eta = 70.2\%$, but even doubling the width to $2831\mu m$ corresponds to $\eta = 66\%$, which is 94% of the peak value. Quadrupling the width corresponds to 77% of the peak efficiency. The conclusion is that while the efficiency optimization is important, the optimum is relatively shallow and even upsizing by a factor of 2 from the optimal width results in less than a 10% reduction in peak efficiency.

We seed the HD3 reduction simulation with the W_{opt} value from the efficiency optimiza-

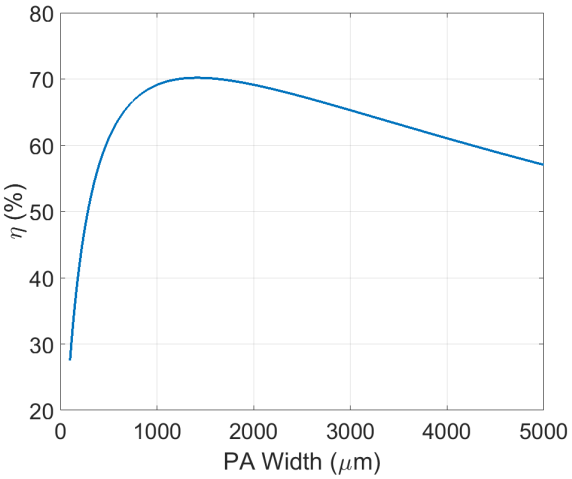


Figure 2.41: System efficiency vs PA array width with $\beta_\eta = 31.5$

tion algorithm, and increase until needed to meet the HD3 reduction specification. The final W chosen corresponds with the total width of SCPAs connected to a single transformer port, also known as a PA array. There are 4 sub-PAs per PA array in order to implement mixed signal filtering, so the total width per sub-PA is $W/4$.

1. Start with P_O, f_{LO}, f_{BW}. Compute $Q = \frac{f_{BW}}{f_{LO}}$.
2. Compute $L_S = \frac{R_L}{\omega_{LO} Q}$.
3. Choose a, V_{DD}, n based on P_O and other concerns like area.
4. Design and lay out the transformer given L_S and n. Extract to get actual L_P and L_S values.
5. Compute the total SCPA capacitance $C_s = \frac{1}{\omega_{LO}^2 L_P}$. Compute $C_{s,unit} = \frac{2 k_{PA} C_s}{n_{cells}}$.
6. Compute W_{opt} using the algorithm in 2.7.1.
7. Simulate HD3 reduction at f_{LO} by finely sweeping phase shift around 60°.
8. If this does not meet the specification, increase W by some amount and repeat step 7. Otherwise, use this final value of W.

Currently, the design of the transformer starts with a generated layout which is then manually modified by the user. Steps 6-8 all happen within a scripted loop, in which layout is generated, extracted, and simulated all within the Berkeley Analog Generator (BAG) framework. In a sample run, step 8 increases W by 1.1x per iteration until either the user specified maximum number of iterations is reached or the HD3 reduction specification is

met. The BAG loop is fully functional, but has not been used to design a fabricated TX since it was only completed after both versions of the TX were fabricated.

2.8 First Prototype Measurements (65 nm)

The first version of the spectral filtering transmitter was taped out in a TSMC 65nm process, measuring 2.6 mm x 3.4 mm with an active area of 3.6 mm^2 including the transformer (Fig. 2.42). A custom printed circuit board (PCB) was designed and fabricated in order to test the prototype. The chip was attached to the board through a process called flip-chip on board (FCOB), in which the bare die is flipped and attached to the board [45]. There were two versions of the PCB made, though all plots shown in this section will only be from the second board. The necessity of a second version of the PCB will be explained in Section 2.9. All measurements were performed using with $V_{DD} = 1.2V$ for the analog and digital supplies and $2V_{DD} = 2.4V$ for the PA supply.

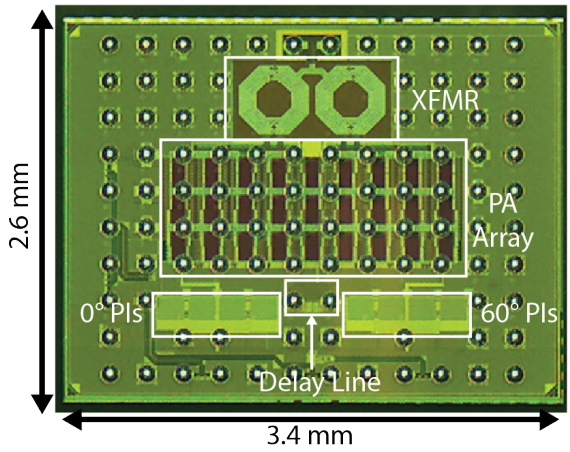


Figure 2.42: TX v1 die photo

The testing setup is shown in Fig. 2.43. I/Q data and scan data are controlled by an FPGA, which is controlled using a PC. Signal generators provide fast and slow data clocks, the 2x LO signal, and the FPGA clock. The chip output is taken differentially to two ports of an oscilloscope using a matched pair of cables.

First, continuous wave (CW) measurements were performed. All measurements spanning multiple frequencies range from 0.7 GHz to 2.0 GHz using steps of 100 MHz. From Fig. 2.44 and Fig. 2.45, this design achieves a peak output power $P_{out} = 26.8dBm$ and $\eta_{sys} = 25\%$ at a center frequency of $f_{LO} = 900MHz$ without harmonic cancellation enabled. With harmonic cancellation enabled, $P_{out} = 25.6dBm$ and $\eta_{sys} = 20\%$. The 1.2 dB reduction in P_{out} matches the theoretical expectation. The design demonstrates an efficiency backoff better than class B (Fig. 2.46), which is consistent with previous work utilizing the SCPA topology [9][32].

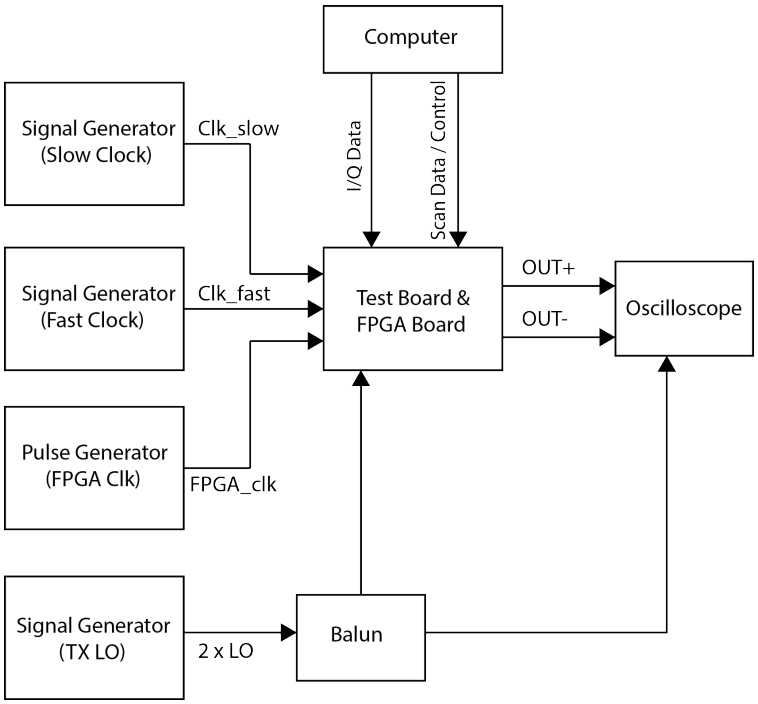


Figure 2.43: Measurement block diagram

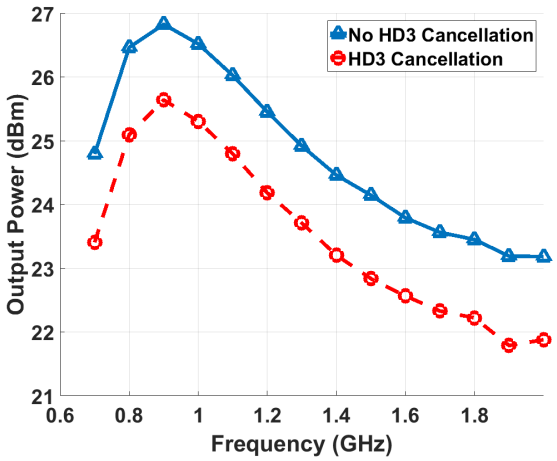


Figure 2.44: Peak P_{out} vs frequency

The HD3 reduction is measured by first fixing the PA input code and sweeping the PI pair phase of one PA array using the scan chain while fixing the PI pair phase of the other. The fundamental and third harmonic are measured at each code, and the HD3 is computed and recorded. This is done at each frequency for a few different PI settings, until we find one which maximizes HD3 reduction. The optimal setting for the previous frequency point

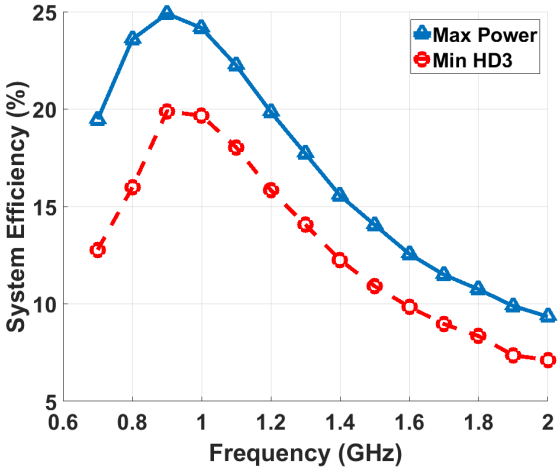


Figure 2.45: System efficiency vs frequency

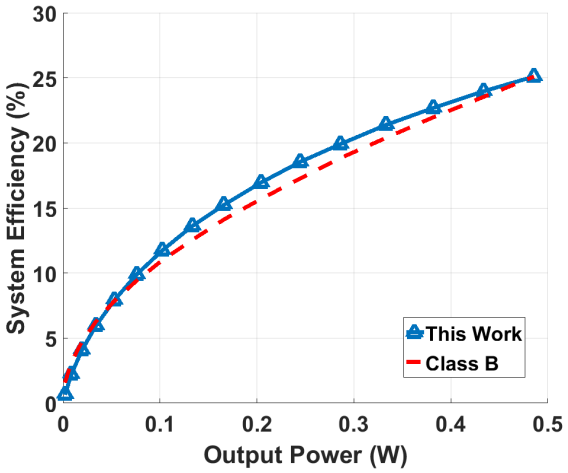


Figure 2.46: System efficiency vs code

is used as an initial seed, with that bias setting as well as the ones above and below it tested.

We measured a HD3 reduction ranging from 24 dB to 42 dB across our 700 MHz - 2GHz range (Fig. 2.48), with a post cancellation HD3 ranging from -44 dB to -58 dB. A HD3 reduction of 42 dB and post cancellation HD3 of -57 dB are achieved at the center frequency of $f_{LO} = 900MHz$ (Fig. 2.47). These measurements are performed at peak P_{out} by transmitting I = Q = 255. This demonstrates the effectiveness of this technique across a wide frequency range, making it suited for a frequency flexible front end.

The IQ/I linearity vs input code was measured with two different duty cycle settings (Fig. 2.49), with one setting being the smallest and the other being the optimal setting. Though we still have code variation in the optimal setting, the IQ/I ratio is clustered more closely

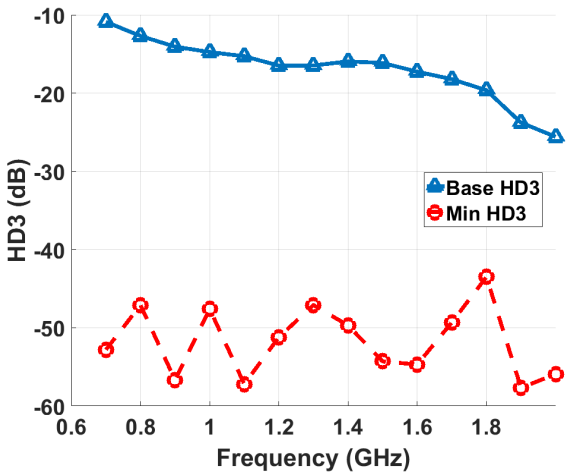


Figure 2.47: HD3 vs frequency

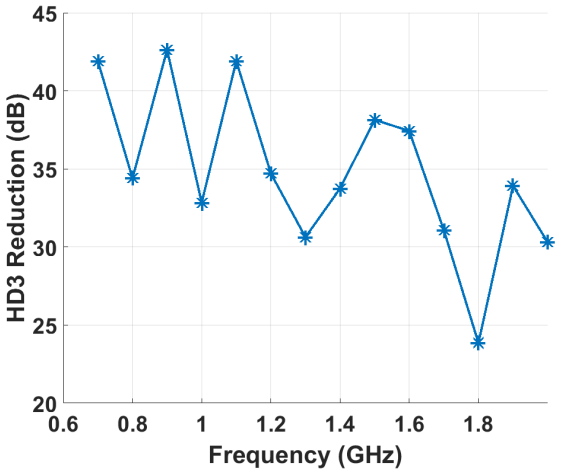


Figure 2.48: HD3 reduction vs frequency

to the ideal 3 dB value. Another way to visualize this is to measure the constellation from the chip output, which is measured by sending a large data packet to the chip and measuring the oscilloscope output. We will first clarify our method of transmitting modulated data.

Data is sent from the FPGA to the chip using a memory on the FPGA, with custom FPGA code written to handle reading, writing, and the various interfaces. This FPGA memory has separate read and write ports with separate clocks, allowing these to occur at different rates. A Xilinx Virtex-707 FPGA [46] is used for measurements, and has high speed GTX transceivers [47] which are able to send data at our required 5 Gb/s data rate. The GTX takes in a data bus input from an FPGA memory at a lower data rate, which is then serialized to generate a serial stream at 5 Gb/s. The GTX constantly reads from the

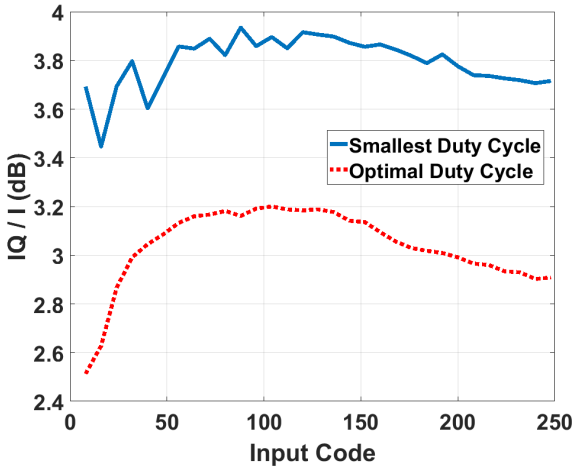
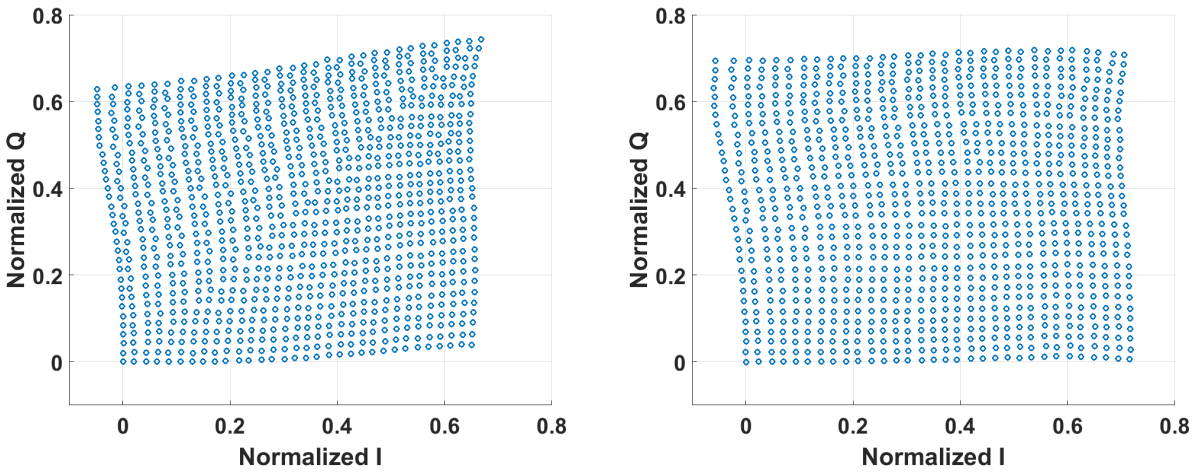


Figure 2.49: IQ / I ratio vs code for different duty cycle settings



(a) Smallest duty cycle

(b) Optimal duty cycle

Figure 2.50: Top right constellation quadrant with different duty cycle settings

memory, looping through from the starting address to a final address specified by the user. In contrast, the PC writes data to the memory at a much lower data rate set by the baud rate of the PC to FPGA serial interface. Writing at this lower data rate was done to simplify the PC to FPGA interface.

The measured 1^{st} quadrant of the constellations under the two duty cycle settings are plotted in Fig. 2.50a and Fig. 2.50b to qualitatively demonstrate the effect of the duty cycle on the shape of the constellation. These measurements are taken by sending a long string of I, Q codes in which we step by 8 codes each time. The data changes at a rate of 15.625 MHz to allow for transients to settle. The entire packet is recorded at once,

which is then post-processed on a PC. This post-processing includes demodulation and phase shifting the constellation, using the point at $I = 32$, $Q = 0$ as the reference for phase equal to 0°. The constellation quadrant data is spread over a relatively short $65.536\mu s$ in order to minimize phase drift within the measured codes. Measurements show qualitatively that the constellation is much more square in the optimal duty cycle case compared to the smallest duty cycle case. No predistortion was used either in these constellation or IQ/I measurements, demonstrating the high linearity of the SCPA consistent with prior art [9][32].

Measurements using modulated data is also critical for the cellular standards that this TX targets. The modulated data used consists of a 20 MHz bandwidth LTE signal at a 500 MS/s data rate with 9 dB PAPR. All spectrum plots in this section are normalized to the maximum CW output power at 900 MHz, meaning that integrating the PSD across the 20 MHz bandwidth around the fundamental will result in -9 dBc.

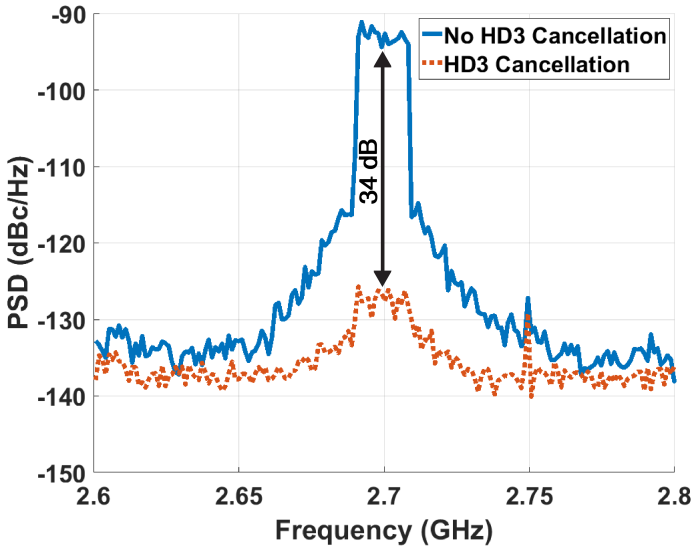


Figure 2.51: Normalized 3^{rd} harmonic spectrum with HD3 cancellation enabled, $f_{LO} = 900MHz$

Harmonic cancellation was measured with modulated signals, where an HD3 reduction of 32 dB is achieved, with a post-cancellation HD3 of -45 dB (Fig. 2.51). This is a good result, but less than the 42 dB reduction from the CW case. This could be due to mismatch in the DAC itself, as the matching between codes will differ from the max power case due to random variation.

The spectrum under different mixed-signal filter settings is shown in Fig. 2.52, which includes the setting implementing no filter shown in blue. These settings are programmed by the scan chain, where n_k is the number of slow data clock cycles to delay the input for the k^{th} tap. The setting of $n_0 = 0, n_1 = 6, n_2 = 7, n_3 = 13$ places notches at 35.71 MHz and 41.67 MHz offset from the LO to notch out a 20 MHz bandwidth at a 40 MHz offset.

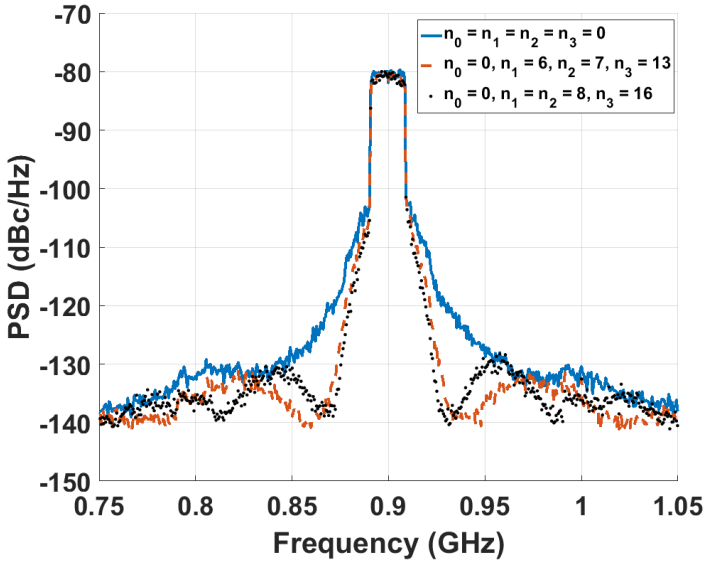


Figure 2.52: Normalized spectrum with mixed-signal filtering enabled with $f_{LO} = 900MHz$

This specific filtering band is chosen to mimic the nearest LTE channel. This configuration achieves a peak notch of 18 dB within this band and reduces the power of the overall band by 15 dB while reducing in band power by 0.45 dB. An ideal filter with the same configuration predicts a 23.5 dB reduction across this channel with a 0.47 dB reduction in band. This leaves a relatively large gap of 8.5 dB between measured and ideal performance, which could be due to nonlinearity in summation and DAC mismatch.

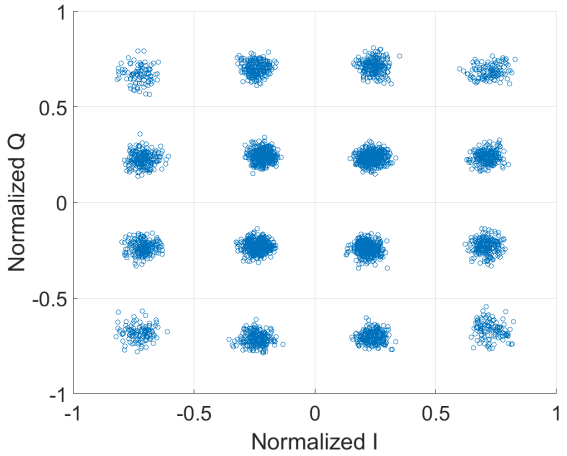


Figure 2.53: EVM measurement of 16QAM constellation at 125 MS/s data rate

EVM measurements were also taken for a 16QAM constellation with a maximum code of 224 (Fig. 2.53). The EVM measurement consisted of transmitting points in the 16QAM

constellation in a long, random sequence at 125 MS/s and capturing the TX output using the oscilloscope. This output was then taken and post-processed on a PC, measuring an EVM of 4.94% (-26.1 dB) without any predistortion applied.

Table 2.4: Comparison table for the TX v1

	Huang, RFIC 2016 [30]	**Ba, RFIC 2014 [28]**	**Mizokami, VLSI 2017 [29]**	**Bhat, JSSCC 2017 [32]**	**Bhat, RFIC 2014 [31]**	**This Work**
Architecture	HR SCPA	Conduction Angle Calibration	33% Duty Cycle PA	Mixed-Signal Filtering	Mixed-Signal Filtering	HR and Filtering SCPA
Technology	40 nm	40 nm	40 nm	65 nm	65 nm	65 nm
Peak Power	8.9 dBm	1.2 dBm	20 dBm	30.3 dBm[1]	29.9 dBm	25.6 dBm[2]
Frequency	0.9 GHz	2.4 GHz	< 1GHz	2.2 GHz	2.4 GHz	0.7–1.6 GHz
Peak Efficiency	43%[3]	39%[3]	43%[4]	34%[4]	38.3%[4]	20%[4]
HD (CW)	HD2: -65 dB HD3: -46 dB HD4: -42 dB	HD2: -50 dB	HD3: -57.7 dB	N/A	N/A	HD3: -57 dB
HD Reduction (CW)	HD2: 48dB HD3: 17 dB HD4: 24 dB	HD2: 23 dB	N/A	N/A	N/A	HD3: 42 dB
Notch Depth @ f_{offset}	N/A	N/A	N/A	24 dB @ 140 MHz	8 dB @ 100 MHz	18 dB @ 40 MHz
Noise Floor @ f_{offset}	N/A	N/A	N/A	-149 dBc/Hz @ 140 MHz	-134 dB/Hz @ 100 MHz	-137 dBc/Hz @ 40 MHz
Modulation Bandwidth	N/A	N/A	N/A	1.4 MHz	20 MHz	20 MHz
EVM @ Sample Rate Symbol Rate	N/A	N/A	N/A	-29.2 dB[5] 560 MSa/s 20 MSym/s	-20 dB[5] 200 MSa/s 20 MSym/s	-26.1 dB[6] 125 MSa/s 125 MSym/s

[1]23Ω load, [2]100Ω load, [3]Drain efficiency, [4]System efficiency, [5]64QAM, [6]16QAM

This work is compared to prior art in Tab. 2.4, with this work shown in the rightmost column, which is highlighted in grey. We could not find existing works that combined techniques to suppress harmonics and implement mixed signal filtering into a single design at the time, so works which implemented these separately were compared against. All works compared against specifically use switching PAs or digital PAs.

The TX achieves a comparable CW HD3 to [30][28][29] while operating at the highest output power reported of 25.6 dBm. As mentioned before, the harmonic cancellation technique used in this work has been demonstrated in [30], but this work achieves comparable cancellation at an output power 15 dB higher, and also reports results with 20 MHz modulated data. The HD2, HD3, and HD4 numbers reported in [30] are measured using a single tone (CW) tests.

Previous work also only demonstrated the harmonic distortion numbers at a single frequency, since they were focused on meeting a specific standard. This work demonstrates the viability of this technique across a wide range of frequencies. It should be noted that [28][30] focused on cancelling the 2nd harmonic for single ended implementations of low power standards such as Zigbee. In comparison to other works reducing quantization noise in nearby channels, this work has superior mixed-signal notch performance to [31] but worse than [32]. Both this work and these two works implement integrated, programmable filters.

The results of the first prototype are good overall, but these results only hold up for a specific type of board output network. Two iterations of the PCB design were required, with the first version having very poor results. This will be further elaborated below, and was the driving push, along with increased design automation, for a second version of the filtering TX.

2.9 Filtering Techniques and Output Network

The first version of the filtering TX chip had two board designs, which differed primarily in the off-chip output network. The original board had a single ended output in which one side of the differential output was connected to a single ended transmission line to an SMA, while the other side is connected to the SMA ground, shown in Fig. 2.54. Simulations with the SMA and chip grounds shared had significantly degraded harmonic cancellation, so the SMA ground was explicitly isolated from the shared chip / board ground, with cutouts beside and below SMA ground to reduce capacitive coupling between the two grounds.

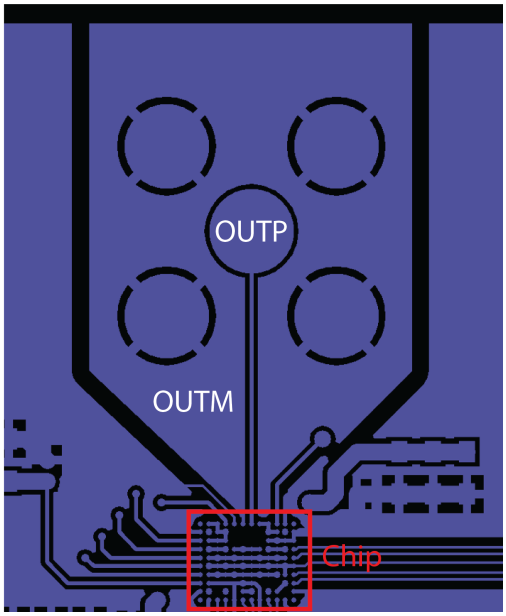
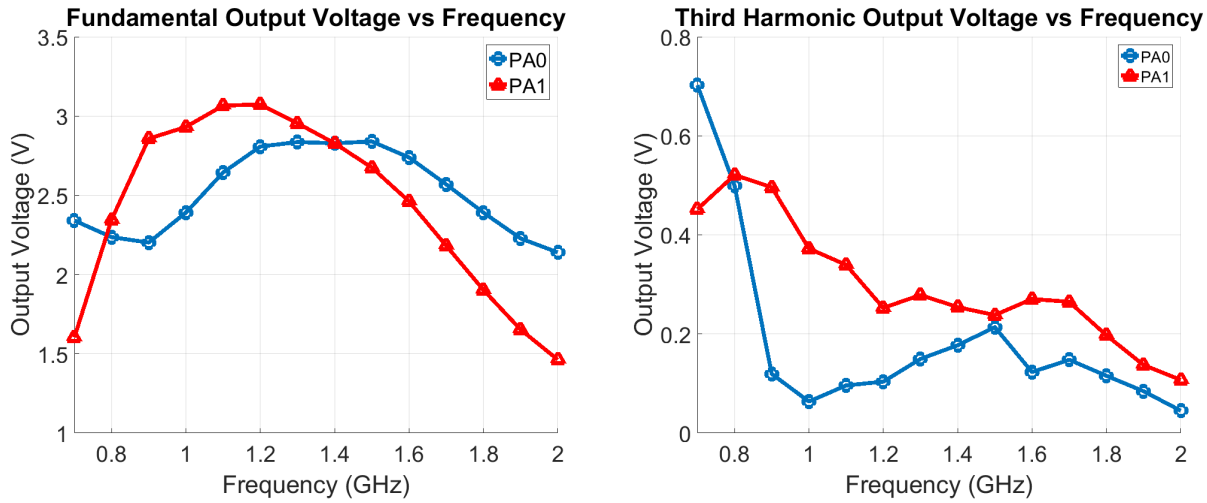


Figure 2.54: Output network for 1st board for SCPA v1



(a) Fundamental (b) Third Harmonic

Figure 2.55: PA array transfer functions

A major issue with this board design is that the measured harmonic cancellation was extremely frequency dependent, ranging from 3 to 35 dB across a 0.7 to 2.0 GHz range. These numbers were significantly worse than numbers expected from simulation, which did not include any model of the off-chip output network.

The cause of the reduced effectiveness of the cancellation is due to the transfer function from the two PA arrays (sets of 4 sub-PAs grouped into 2 phases) to the final output differing. We measured the fundamental and third harmonic output voltage of each PA array by sending the max code to one PA array while turning off the other PA array, shown in Fig. 2.55a and Fig. 2.55b. The transfer functions are completely different across frequency for both the fundamental and harmonic, and the couple of points which have good cancellation are essentially due to coincidence.

We can demonstrate that this mismatch in the amplitudes is the primary limiting factor for the harmonic cancellation. We compare the HD3 reduction from measurement to summing the fundamental and third harmonics using PA transfer functions from Fig. 2.55 and summing assuming they are exactly 60° and 180° out of phase, respectively. This setup is shown in Fig. 2.56, with the HD3 reduction overlaid in Fig. 2.57. The two figures nearly overlap, meaning that the measured output sums with sufficient phase resolution, and that poor amplitude matching between the two PA arrays is the limiting factor for harmonic cancellation.

The significant difference in transfer functions implies a difference, or asymmetry, in the output network. This asymmetry is clear in the first version of the board, as outputs of different polarity are connected to fundamentally different networks. To verify that the asymmetry of the board output network degrades the HD3 reduction, HD3 cancellation simulations were run with a model of the board. The board output network was modeled

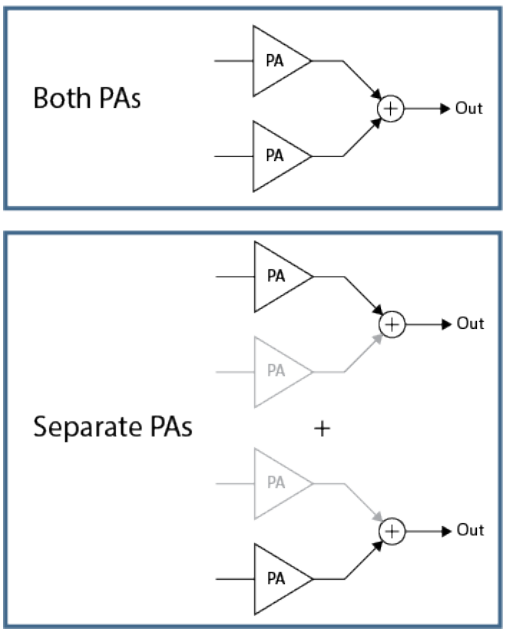


Figure 2.56: Setup for measured vs perfectly summed HD3 reduction

and simulated in HFSS [48], which was simulated along with the extract SCPA unit cells. The simulation results are shown in Fig. 2.58, which is overlaid with the measured HD3 reduction. Though the exact values differ, the general trend and shape match relatively well across frequency especially from 0.6 GHz to 1.4 GHz, supporting the theory that the HD3 reduction is degraded by the output network.

Analysis using a complex output network like this board output network will be difficult and may not produce a very clear result. Instead, we need to come up with a relatively simple model to better understand the effects of asymmetry on the output voltage and power. Assuming we have the linear summation required for the filtering techniques employed, we can model two PAs being summed using a series stacked transformer with the schematic in Fig. 2.59, which is divided into on-chip and off-chip sections. V_1 and V_2 each represent the output voltage of a distinct PA. For example, for 3rd harmonic cancellation, these represent the third harmonic output voltage of the PAs. The off-chip output network is modeled with shunt impedances Z_B and Z_T. The output network is likely much more complicated, but even this simplified model gives sufficient insight about the effects of asymmetry on a residual output.

$$v_{OP} - v_{OM} = K\left[(V_1 + V_2)\left(1 + \frac{Z_C}{Z_T} + \frac{Z_C}{Z_B}\right) + n^2 Z_S\left(\frac{V_1(1+\epsilon)}{Z_B} + \frac{V_2}{Z_T}\right)\right] \tag{2.59}$$

The output voltage v_O is shown in Eq. 2.59, where K is a coefficient that depends on the turns ratio n and the various impedances. This is written to demonstrate that residual output can come from several sources. The first term $(V_1 + V_2)$ shows that mismatch in V_1

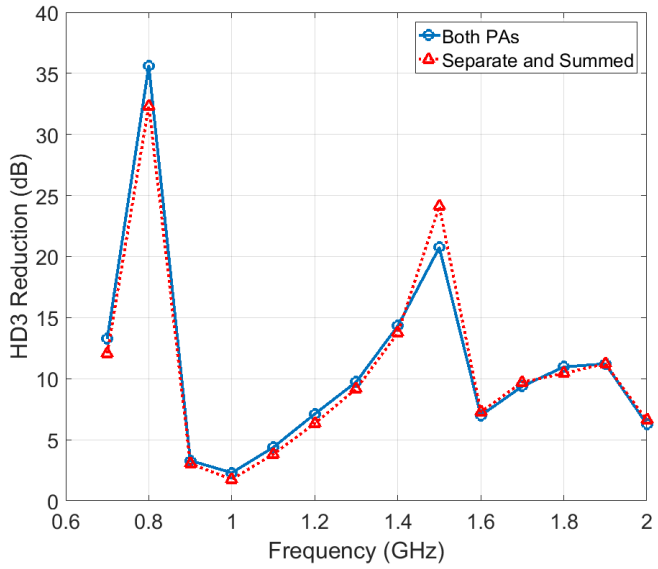


Figure 2.57: HD3 reduction for both PAs vs separate and summed PAs

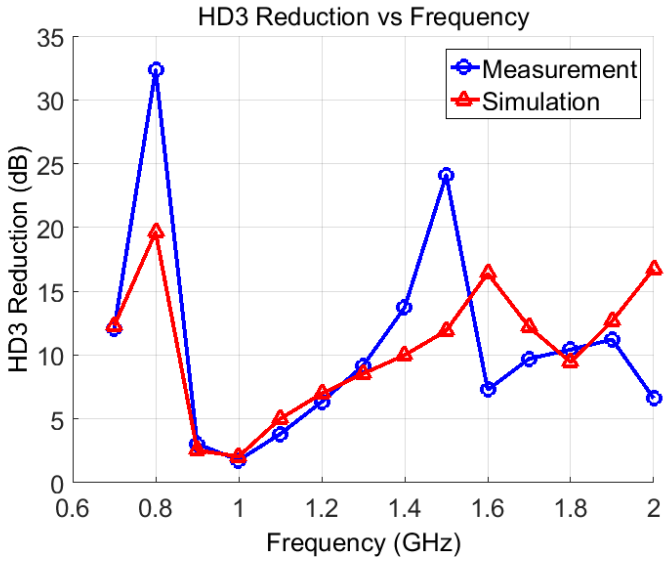


Figure 2.58: Simulated HD3 reduction with board model vs measured HD3 reduction

and V_2 ($V_1 \neq -V_2$) will result in residual output, which is relatively intuitive. The second term starting with $n^2 Z_S$ shows that mismatch in either PA output impedance or impedances in the output network ($Z_B \neq Z_T$) will result in residual output even if $V_1 = -V_2$. From this, it is clear that we need to match both the PAs (output voltage and output impedances) and ensure symmetry in the output network to ensure effective cancellation.

The chip was designed to maximize symmetry in the SCPA, which fulfills the first con-

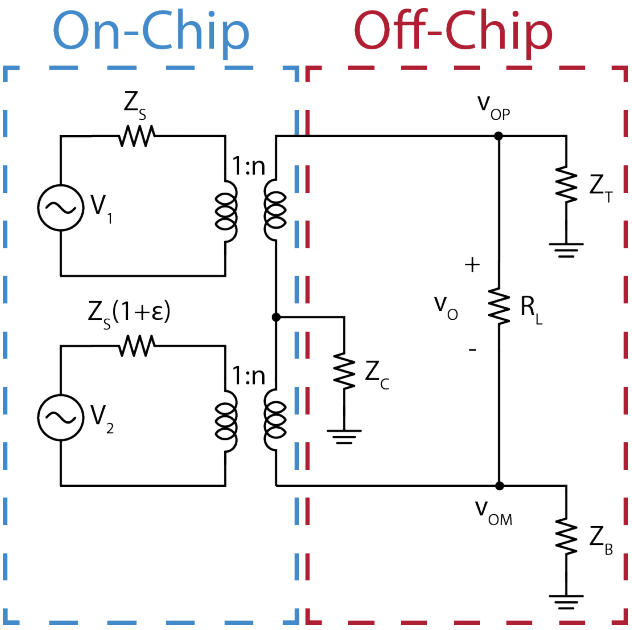


Figure 2.59: Series-stacked transformer summing

dition for both versions of the board. However, it is clear that $Z_B \neq Z_T$ in the first board design due to the completely physically distinct nature of the output network. This exact issue was the motivation for the second board which tries to ensure $Z_B \approx Z_T$ by making the output network as symmetric as possible to improve HD3 reduction across frequency.

The second version of the board has the output connected to a differential transmission line which connects to a pair of SMA connectors. This is then connected using a matched pair of SMA cables into two ports of an oscilloscope in order to preserve the output symmetry. We take the difference of the two ports to get the differential output, meaning we have $R_L = 100\Omega$ instead of $R_L = 50\Omega$. The harmonic cancellation is significantly improved compared to the first board as seen in Section 2.8, Fig. 2.48. The downside to this is that we require either a highly symmetric differential output network or a balun to convert the output from differential to single ended. The measurements in Section 2.8 use the former approach.

We also took measurements for the balun case, in which the matched pair of SMA cables connect to an external (off-board) balun which is then connected to a spectrum analyzer. The HD3 reduction also ranges between 24 to 42 dB (Fig. 2.61) like in the case using an oscilloscope (Fig. 2.48), though the shape of the HD3 reduction vs frequency curve differs. These measurements support the qualitative analysis of our simple model which emphasizes the critical condition of $Z_T = Z_B$ for harmonic cancellation.

One observation made during measurement was that the effectiveness of the mixed-signal filtering varied little between the two different boards, in contrast to the harmonic cancellation. The major difference between the implementation of the two techniques is the

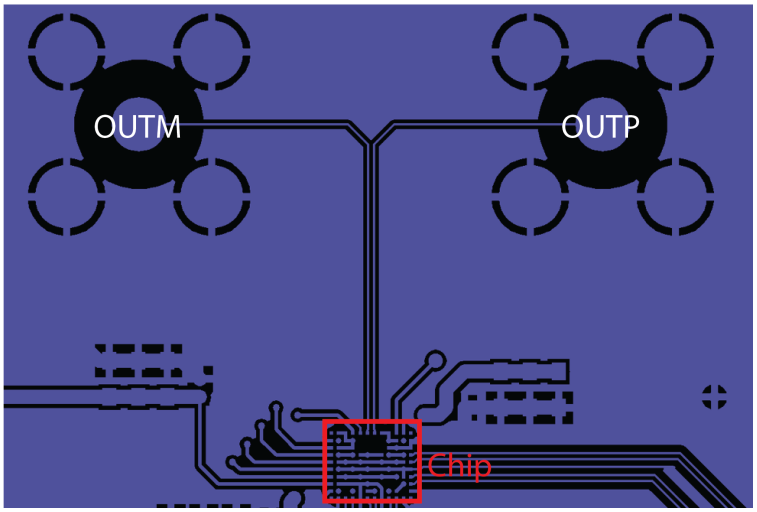


Figure 2.60: Output network for 2nd board for SCPA v1

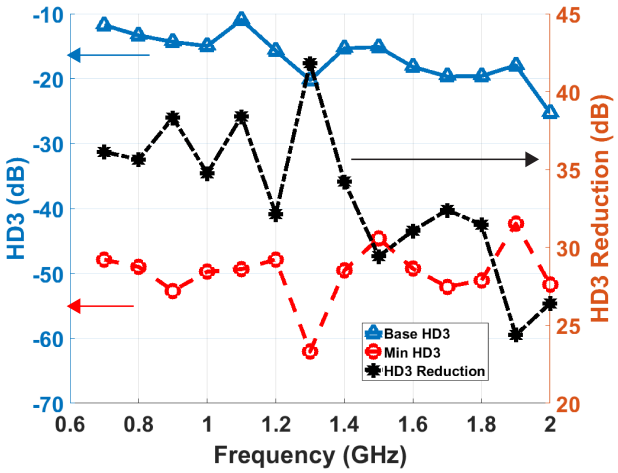


Figure 2.61: HD3 reduction vs frequency for 1st board with balun

mixed-signal filtering summation is implemented by drain combining instead of using the series-stacked transformer for summation. The drain combining along with a series stacked transformer is modeled in Fig. 2.62 to analyze the differences in summation from these two networks.

$$v_O = K\left[(V_1\Gamma_1 + V_3\Gamma_2)\left(1 + \frac{Z_C}{Z_T} + \frac{Z_C}{Z_B}\right) + V_1\frac{n^2\Gamma_1 Z_1 Z_2}{Z_B(Z_1+Z_2)} + V_3\frac{n^2\Gamma_2 Z_3 Z_4}{Z_T(Z_3+Z_4)}\right] \tag{2.60}$$

$$\Gamma_1 = \frac{Z_2 - Z_1}{Z_1 + Z_2} \tag{2.61}$$

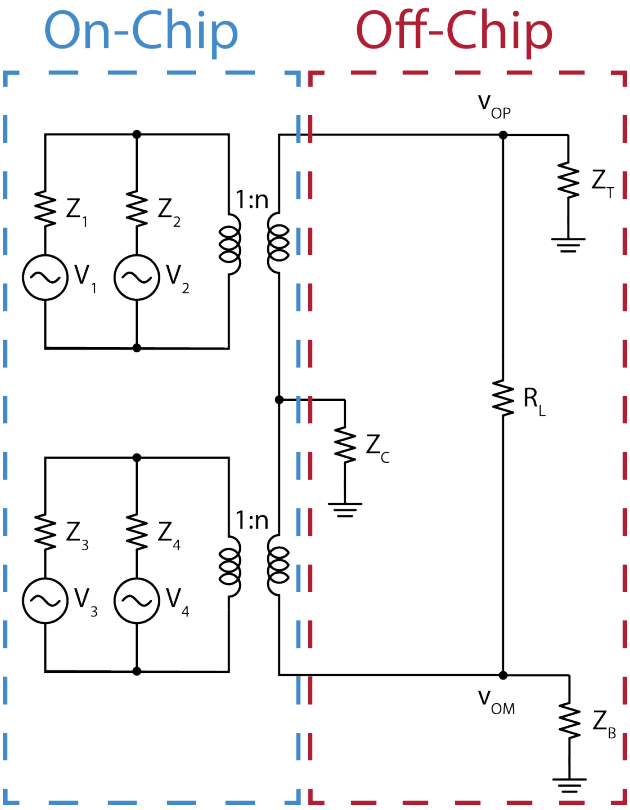


Figure 2.62: Drain combining with a series-stacked transformer

$$\Gamma_2 = \frac{Z_4 - Z_3}{Z_3 + Z_4} \tag{2.62}$$

The output of the sub-PAs with drain combining and a stacked transformer is shown in Eq. 2.60. Drain combined sub-PAs are represented by pairs of voltage sources, grouped into V_1, V_2 and V_3, V_4. This equation assumes that the voltage source pairs are of equal amplitude and opposite polarity, i.e. $V_1 = -V_2$ and $V_3 = -V_4$. The critical aspect of this equation is that while asymmetry in the network ($Z_T = Z_B$) can scale the output, ensuring matching of the output voltage source pairs and output impedances of the drain combined sub-PAs ($Z_1 = Z_2$, $Z_3 = Z_4$) will cancel the output regardless of asymmetry in the off-chip network. This analysis is consistent with mixed-signal filtering measurements being very similar between the first and second board.

$$v_O = \frac{nR_L(V_1Z_2 + V_2Z_1)(Z_T + Z_B)}{(Z_1 + Z_2)(Z_T + Z_B)R_L + n^2Z_1Z_2(R_L + Z_B + Z_T)} = K'\left[V_1Z_2 + V_2Z_1\right] \tag{2.63}$$

The series stacked transformer is no longer needed for summation if both harmonic cancellation and mixed-signal filtering are implemented with drain combining. A standard

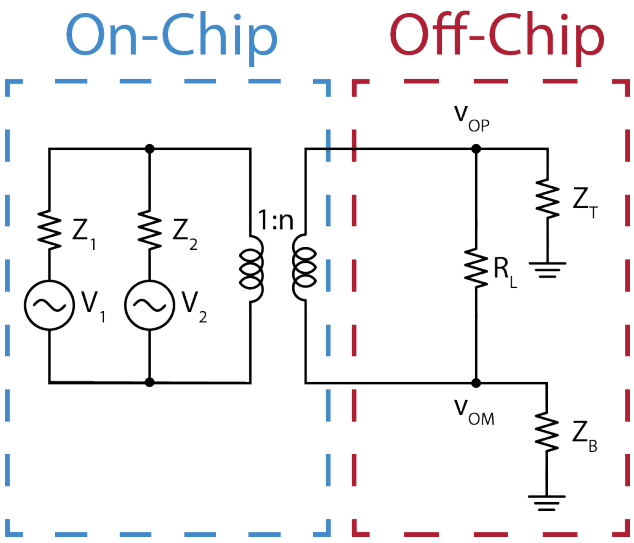


Figure 2.63: Drain combining with a single transformer

transformer can be used if the power combining provided by the series stacked transformer is no longer needed. We can model the TX and output network with the schematic shown in Fig. 2.63, which gives the output expression in Eq. 2.63. The second form of the equation with K' demonstrates a key benefit of using a single transformer: any off-chip output network asymmetry does not factor into the output. The specific values of Z_B, Z_T can affect the scale the output voltage but $Z_B = Z_T$ is no longer required to ensure good cancellation.

This makes intuitive sense, since Z_B, Z_T are the only components shunted to ground, meaning we can view ground as the middle node between the two impedances. Z_B and Z_T are seen in series looking from V_{OP} to V_{OM}, meaning we can view the load R_L as having a single parallel impedance $Z_B + Z_T$. Any contribution from either Z_B or Z_T will be seen equally by both PAs, which was not true in previous networks due to the Z_C impedance. This differs from the series stacked transformer case because the presence of Z_C prevents the simplification of the contributions of Z_B, Z_T to a single parallel impedance.

This model suggests that we can use a simple but highly asymmetric output network to convert the chip output from differential to single ended without significantly impacting the harmonic cancellation. We ran two different simulations with different setups in order to verify this. The first setup has one of the TX outputs tied to ground, with the other being the single ended output connected to $R_L = 50\Omega$. The second setup has the output network of first board (shown in Fig. 2.54) used earlier to generate the data in Fig. 2.58. The simulations use BAG generated SCPA unit cells and an EMX extracted transformer model of the transformer used in the second version of the TX. The HD3 reduction across frequency is plotted in Fig. 2.64.

The simulated HD3 reduction does not significantly degrade even in the presence of the highly asymmetric output network, achieving an HD3 reduction $> 30dB$ from 1 - 2 GHz. The second version of the TX uses only drain combining for both techniques and a single

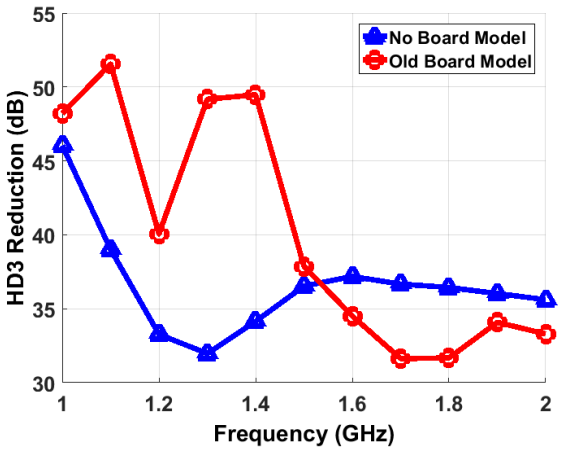


Figure 2.64: Simulated HD3 reduction for simple transformer drain combining with different output networks with single ended output

transformer in order to make the filtering techniques as robust as possible. The board output network connects one of the differential chip output ports to ground and takes the other port as a single ended output (Fig. 2.65). This implementation of differential-to-single ended conversion was used to avoid the use of an external balun.

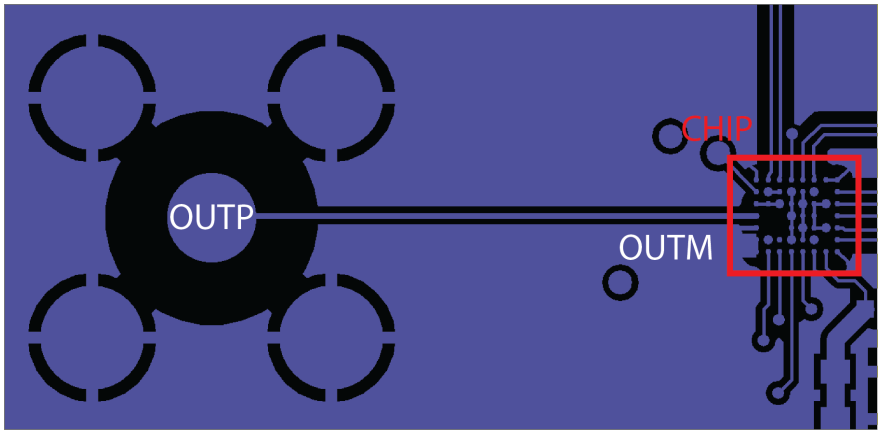


Figure 2.65: TX v2 board output network

2.10 Revised TX System Block Diagram

A 2nd version of the TX was designed to implement a version in which the filtering techniques would be insensitive to the off-chip output network. A secondary goal of this version was to implement more of the overall design in BAG, since the 1st version only implemented the

SCPA core using BAG. This 2nd version is overall similar (Fig. 2.66) to the 1st version. The TX has 2 LO phases and 4 mixed-signal filter taps for a total of 8 sub-PAs. Each sub-PA is a 9 bit DAC with 4 thermometer, 4 binary, and 1 sign bit.

The primary system level changes from the first version consist of drain combining the PA arrays, a simpler 1:2 transformer instead of a series stacked transformer, and a single pair of inputs for the input I/Q data. In order to reduce the number of I/O pins on the chip, a single differential input is used for both the I/Q data. A single 20:1 deserializer gives us the 9-bit I and Q input codes, as well as 2 extra bits which are unused. The deserializer uses a DDR scheme, allowing us to operate with $f_{clk,fast} = 10 f_{clk,slow}$.

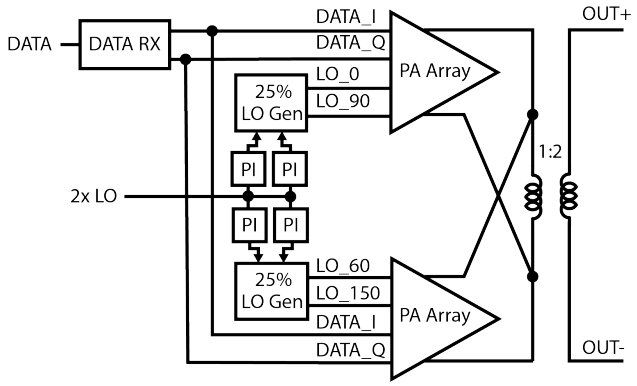


Figure 2.66: TX v2 top level block diagram

This version also has more blocks generated using BAG, specifically a newer version called BAG 2.0 [15]. The SCPA array and PI layouts were completely BAG generated, with the transformer and data RX being partially BAG generated. In the case of the transformer, a generated layout was used as a base and manually modified, while the data RX consisted of manually connected BAG generated layouts in conjunction with manually generated layouts. The implementation details of the generators are further discussed in Chapter 3.

The prototype was implemented in a 28 nm process, in contrast to the 65 nm process for first TX. This switch was motivated by the expected improvement in switch devices due to lower channel lengths. These switches should have reduced parasitics and much more balanced PMOS and NMOS performance with $\alpha_{pn} = 1.2$ instead of $\alpha_{pn} = 2.7$, where α_{pn} is the ratio of PMOS to NMOS width. We expect a lower output power however since the 28 nm process operates using a lower supply voltage. The output power is further reduced since the 2nd version does not have power combining implemented using a stacked transformer like in the first version.

The last major system difference is that there no duty cycle control was implemented, which was mainly due to time constraints. Though we found that previously this could have a large effect on the linearity of the constellation, the SCPA is still significantly more linear than other topologies even without the use of predistortion.

The sizing methodology in the revised TX is the same as that discussed in section 2.7.1 and 2.7.2, with the primary change being in some of the parameter values. We've chosen to

target a lower power and forgo power combining, giving us $k_{PA} = 1$. This choice is also made to make the harmonic cancellation as robust as possible with respect to the asymmetry of the off-chip output network, as described in Section 2.9. This change also sets $\gamma_\phi = \sqrt{3}/2$, half of the value from TX v1. This results in the optimal W will be divided between 8 sub-PAs instead of 4 sub-PAs.

The values of R_u and C_u differ significantly from TX v1 due to the change from a 65 nm to 28 nm process. We expect R_uC_u to decrease, meaning that the optimal efficiency should increase. R_u and C_u are computed via extracted layout simulation in the same fashion as TX v1. R_u remains exactly the same, and while C_u still uses the extracted SCPA unit cell layout, this includes the data mixer and level shifters in revised TX. In the case of this design, we ended up being very linearity limited, with the final W chosen to be larger than the W_{opt} corresponding with peak system efficiency.

2.11 Revised Prototype Measurements (28 nm)

The second version of the TX was implemented in a 28nm TSMC process, measuring 1.86 mm x 1.98 mm (Fig. 2.67). Similar to TX v1, flip-chip bumps were used, which allowed for a more distributed connection for the SCPA's power grid. Supply bumps sit directly above the SCPA, with a different row reserved for each of the three supplies.

The measurement block diagram is shown in Fig. 2.68. This setup is largely similar to that in the first version, with the major difference being that the output is taken single-ended into a signal analyzer instead of being taken differentially to an oscilloscope. Due to this, all of this version's measurements are taken with a 50Ω load instead of a 100Ω load. GPIB was also used more heavily in order to automate measurements.

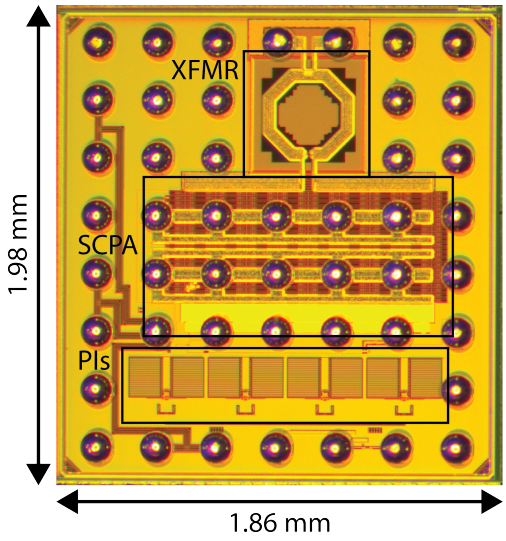


Figure 2.67: TX v2 Die Photo

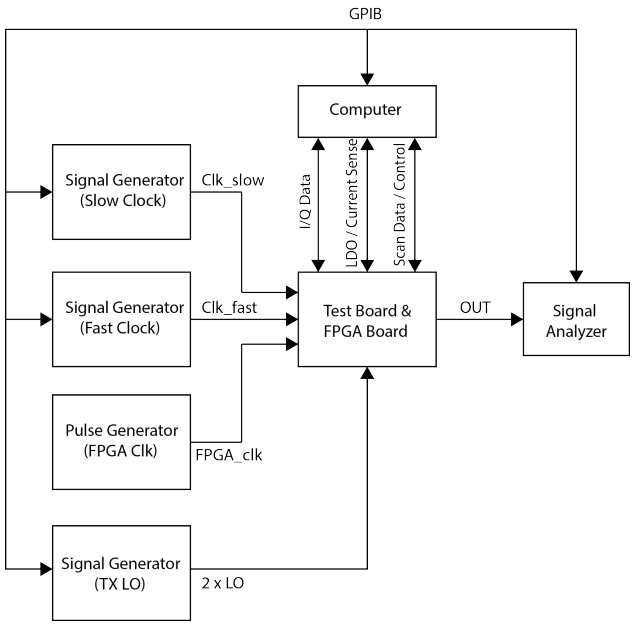
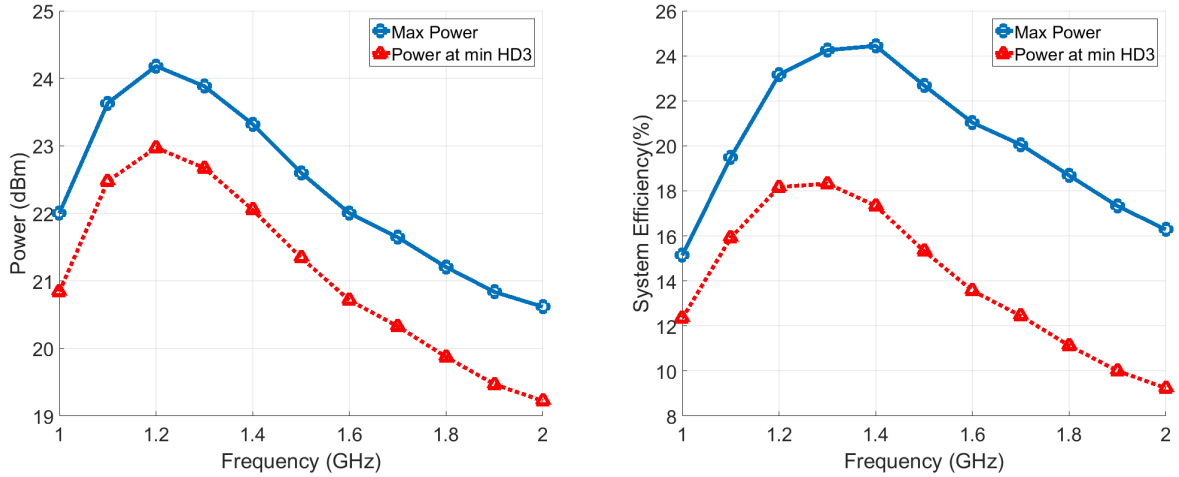


Figure 2.68: TX v2 testing block diagram

All measurements taken with frequency sweeps are between 1 GHz - 2GHz in 100 MHz frequency steps. The chip outputs a peak output power at $f_{LO} = 1.2GHz$ of $P_{out} = 24.2dBm$ without cancellation enabled and $P_{out} = 23dBm$ with cancellation enabled. A system efficiency of $\eta_{tot} = 23.2\%$ and $\eta_{tot} = 18.2\%$ were measured at $f_{LO} = 1.2GHz$ without and with cancellation, respectively. The TX is relatively wideband, with a 1 dB bandwidth of 1.1 - 1.4 GHz and 3 dB bandwidth of 1 - 1.8 GHz. The 1.2 dB loss in output power from enabling cancellation is exactly what is expected for combining 60° out of phase.

The phase resolution of the PIs are measured by observing the demodulated output of the TX. We used the signal analyzer's built in demodulator, which decomposes the output into I and Q components. In order to measure each pair of PIs independently, we need a way to "shut off" the PA array not associated with the pair of PIs under test since there is only a single shared differential output. The bias currents of the final CML to CMOS converter of the two PIs are set to 0, which effectively shuts off one PA array by setting the 25% duty cycle LO signals to be constant voltage instead of a square wave. This was verified by observing a drop in output power of approximately 6 dB after setting the bias currents to 0.

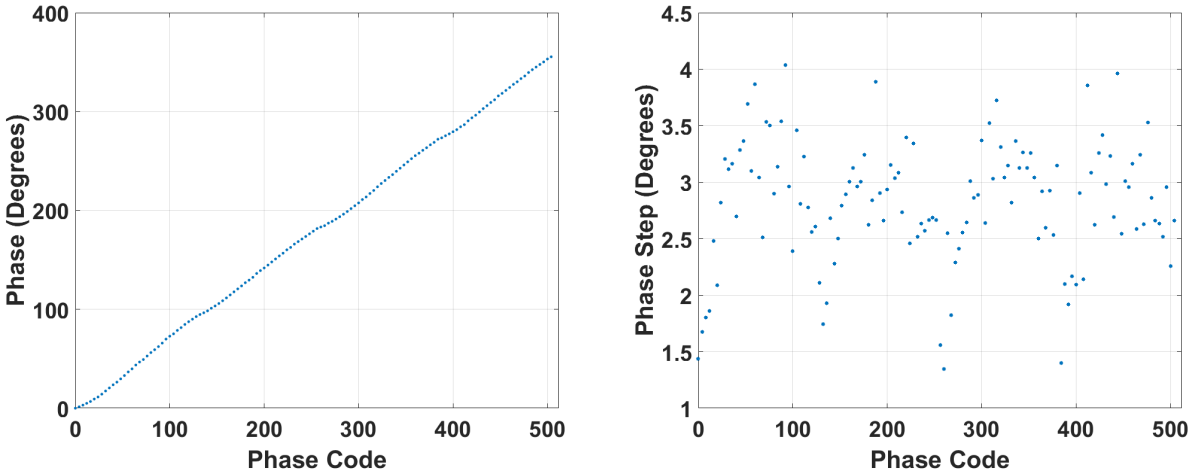
The pair of PIs have their phase codes stepped together with $PI_I = n$ and $PI_Q = (n + 128)\ \%\ 512$ to set a nominal phase shift of 90° between the PIs. The PC uses GPIB to step these phase codes and record the demodulated I, Q components. The output phase is computed using the I, Q components and phase shifted so that the phase of $PI_I = 0, PI_Q = 128$ corresponds with 0°. This process is repeated for several PI integrator bias current values until we find a setting which minimizes the phase standard deviation σ_ϕ across all measured



(a) Peak P_{out} vs frequency

(b) System efficiency vs frequency

Figure 2.69: CW output power and efficiency vs center frequency



(a) Output phase

(b) Output phase code

Figure 2.70: Phase and phase step vs phase code for $f_{LO} = 1.2GHz$, $I_{PI,int} = 100\mu A$

points.

The output phase measurements are taken with phase code steps of 4 across the range of [0, 508]. The input code to output phase (Fig. 2.70a) and output phase step (Fig. 2.70b) transfer functions are plotted with $f_{LO} = 1.2GHz$ and PI integrator bias current $I_{PI,int} = 100\mu A$. As expected of the architecture, the input code to output phase transfer function is not perfectly linear. In particular, the phase resolution near 60° and 300° is approximately 0.78° and 0.85° respectively, close to the predicted value of 0.84° from Section

2.4.

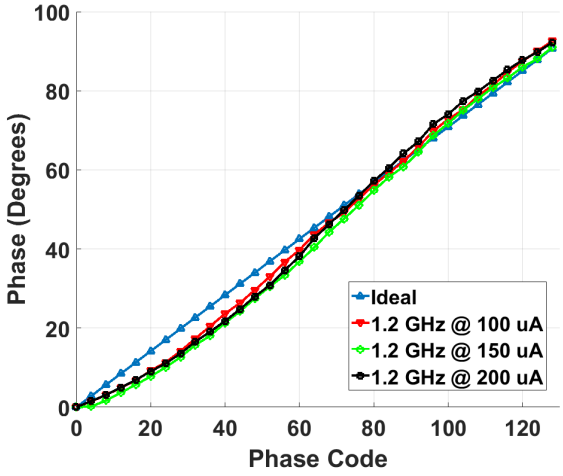


Figure 2.71: Output phase vs phase code for $f_{LO} = 1.2GHz$ with different $I_{PI,int}$

The output phase vs phase code is plotted with different $I_{PI,int}$ values (Fig. 2.71) when operating with $f_{LO} = 1.2GHz$. The ideal linear transfer function is also plotted, and only the first quadrant shown to highlight the differences between the bias settings. From this, we can see the effect of different bias currents on the input code to output phase transfer function, with the optimal setting producing relatively linear results.

The CW HD3 was measured with a static I, Q data input under the maximum power case ($I = Q = 255$). The PC sweeps the phase codes of one pair of PIs while holding the other pair constant, recording the fundamental and third harmonic for each phase code setting. The phase codes of the swept PI pair are swept in the same way as the output phase measurement. This is repeated for several PI integrator bias current settings. All phase codes were swept across the entire range of $[0, 511]$ for these measurements.

The HD3 vs phase code plot with $f_{LO} = 1.2GHz$, $I_{PI,int} = 200\mu A$ is shown in Fig. 2.72, which is the bias setting that produced maximum HD3 reduction of 42 dB. As expected, we see notches near $\pm 60°$, though we have a much deeper notch with $+300°$ $(-60°)$ than with $+60°$. The actual phase where the notches occur are closer to 53° and 304°, based on PI measurements.

This process was repeated across the 1 GHz - 2 GHz range, giving us a measured post cancellation CW HD3 of -51 to -70 dB and HD3 reduction of 35 to 57 dB. This is a significant improvement over the 24 to 42 dB HD3 reduction and post-cancellation HD3 ranging from -44 dB to -58 dB across a 700 MHz to 2 GHz range demonstrated in the first version. A HD3 reduction of 42 dB and post-cancellation HD3 of -58 dB were measured at the center frequency, which is very similar to the performance at the center frequency of the first version. The improved HD3 and HD3 reduction measurements from the first version verify the theory presented in Section 2.9. These improved numbers were achieved even with a

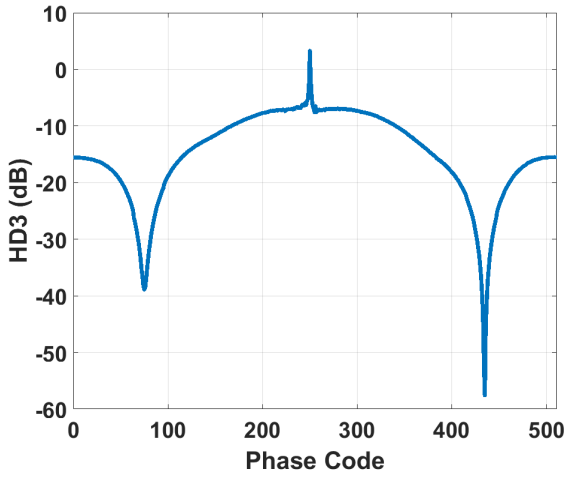


Figure 2.72: HD3 vs phase code with $f_{LO} = 1.2GHz$, $I_{PI,int} = 200\mu A$

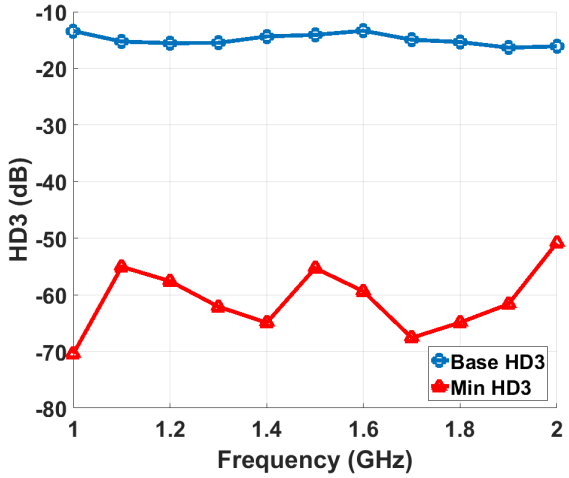


Figure 2.73: CW HD3 for TX v2

highly asymmetric output network, in contrast to the great care taken to preserve symmetry on board and with matched cables in the first version.

Measurements were also taken with modulated data, using 20 MHz LTE data at 7 dB PAPR. These were taken at the LO frequency corresponding to peak output power $f_{LO} = 1.2GHz$. All spectrum plots are normalized to the peak output power at 1.2 GHz, meaning that integrating the 20 MHz bandwidth around the fundamental will result in -7 dBc. The same scheme to send input data used in the first version of the TX (Section 2.8), involving a PC and FPGA, was used in this version.

For HD3 measurements with modulated data (Fig. 2.75), we manually sweep phase codes around the optimal phase setting from CW measurements. The measured 3rd harmonic

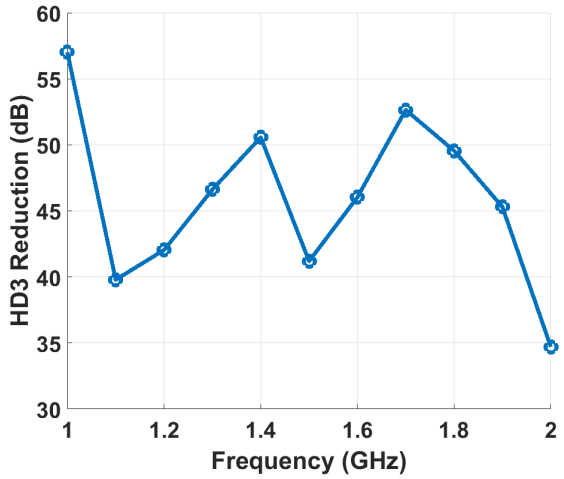


Figure 2.74: CW HD3 Reduction vs f_{LO} for TX v2

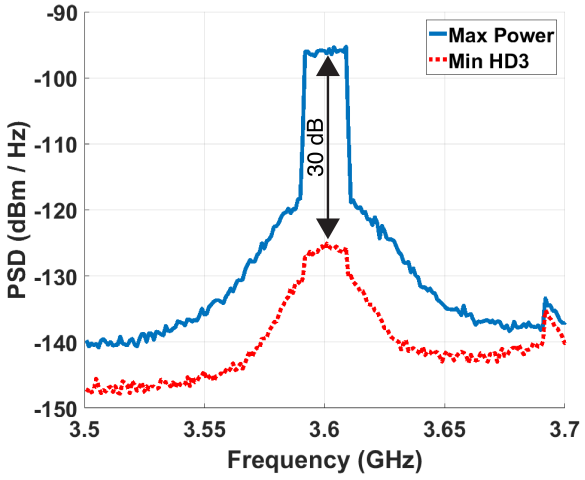


Figure 2.75: Normalized spectrum of 3^{rd} harmonic with 20 MHz LTE data

is reduced by 30 dB across a 20 MHz bandwidth, meaning approximately a 29 dB HD3 reduction considering the 1.2 dB loss in the fundamental. This result is slightly worse than the 32 dB HD3 reduction measurement from the previous chip taken at 9 dB PAPR. The difference between the CW and modulated data case is likely due to variation in matching across different input codes, similar to the first version.

The same modulated data setup was used to take mixed-signal filtering measurements (Fig. 2.76). Notches were placed at 35.71 MHz and 41.67 MHz to filter a 20 MHz channel at a 40 MHz offset, which reduces the channel by a 17 dB while reducing the main channel power by 0.44 dB. Overall, this is a small improvement over the 15 dB channel reduction in the first version. In comparison, an ideal filter with the same configuration with perfectly

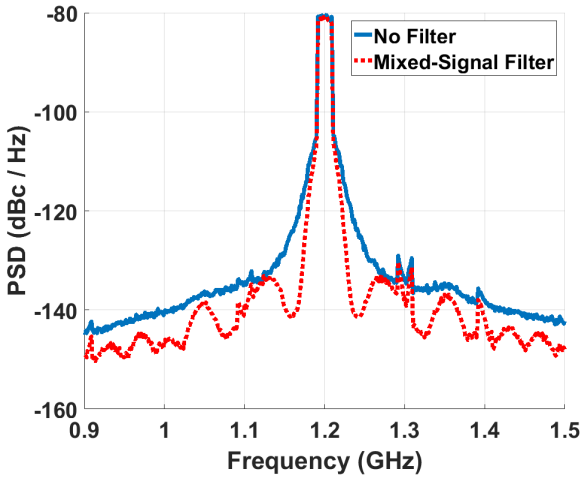


Figure 2.76: Normalized spectrum mixed-signal filtering with 20 MHz LTE data

linear summation and no mismatch predicts a 23.5 dB reduction across this channel with a 0.47 dB reduction in band. Though there is still a relatively large difference of 6.5 dB, the implemented filter still works relatively well.

Overall, the revised TX was able to meet both of its major goals of increased robustness in filtering performance and increased automation. The revised TX matched or surpassed HD3 reduction and mixed-signal filtering performance at a similar output power level compared to the first version. This was done even with an incredibly asymmetric output network connected to a single ended load, demonstrating the robustness of harmonic cancellation with properly implemented summation. The revised TX also used more generated blocks, discussed more thoroughly in the next chapter. The revised TX was implemented in an entirely new process from the original TX, demonstrating that this TX design scales well with modern processes.

Chapter 3

RF Circuit Generators

Core portions of both prototypes were designed and laid out using various versions of the Berkeley Analog Generator (BAG) framework [14]. This is a framework designed at UC Berkeley for the purposes of capturing a designer's methodology into schematic and layout generators. These generators take in user parameters to generate specific instances suited for their purposes. The primary goal of the generators are to promote reusability of not just set instances but entire circuit types, with the goal of even writing generators capable of working across multiple processes.

The first version of the TX used a version predating BAG 1.0, while the second (revised) version of the TX used BAG 2.0 [15]. The features of each different version will be elaborated upon in the respective generator sections. All layout generators in this section were written assuming a technology with at least two thick metal layers and several thin metal layers, with the former being assumed in order to have at least one layer per direction for top level power and signal routing.

This section opens with discussion of the SCPA generator, which is the most critical and complex block generated in BAG for the first TX. After this, we discuss the updated SCPA generator for BAG 2.0, as well as the PI generator in 2.0. Both of these were used in the second version of the TX, in which increased design automation was a major goal.

3.1 SCPA Generator

The first version of the TX used a version of BAG prior to BAG 1.0 to generate portions of the layout. This version of BAG generated layout using Ciranova PyCells [49] which are instantiated and used like PCells. These can be modified live in Virtuoso by modifying the parameters of a placed instance. A key feature of the PyCells is the fgPlace function, which places an object next to another object using DRC rules, ensuring no DRC rules are violated. The objects can range from simple wires to complex objects like instances including transistors and complex routing. The SCPA core and the delay line were implemented using BAG, though we will only be discussing the SCPA core in this section.

The main block implemented in BAG in the first version of the TX, referred to as TX v1, was the SCPA core. The SCPA PyCell core consists of several levels of hierarchy. The top cell is an array which instantiates SCPA unit cells with the proper parameters and places them. Array level routing is included in the SCPA unit cell PyCell, with connections between unit cells made by touching them together at their edges. Given all this, the bulk of the effort and functionality is implemented at the unti cell level, which will be discussed in further detail.

3.1.1 SCPA Unit Cell Generator

Each unit cell consists of input, output, and power routing, the SCPA capacitor, power devices, and buffers for the power devices. The unit cell takes in inputs which have already been IQ combined, mixed up to RF, and level shifted to the appropriate level for the NMOS and PMOS input devices. These inputs drive the power device buffers which then drive the input power devices. The IQ combining, RF mixing, and level shifting were implemented with manually laid out column drivers, external to the SCPA core.

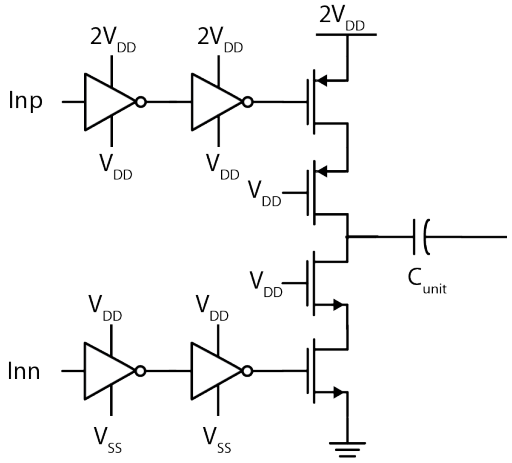


Figure 3.1: SCPA unit cell schematic

The first step to creating a useful layout generator is designing a floorplan that works across a wide set of input parameters. The general layout of the SCPA unit cell is shown in Fig. 3.2. The relative placement of the blocks is constant regardless of parameters. This floorplan is split into two groups based on vertical location in the stackup, where the top layers consist of thick metal routing layers and bottom layers consist of thin metal layers and active layers like oxide diffusion and polysilicon. All blocks besides the supply decoupling capacitance were implemented in the PyCell. The decoupling capacitance was added manually post-generation. First, we will discuss the top layers, consisting of the input routing channels, output wiring, and the horizontal and vertical rails.

The output, horizontal and vertical power rails are implemented on thick metal layers to reduce resistance and improve power handling. The width of each power rail can be set

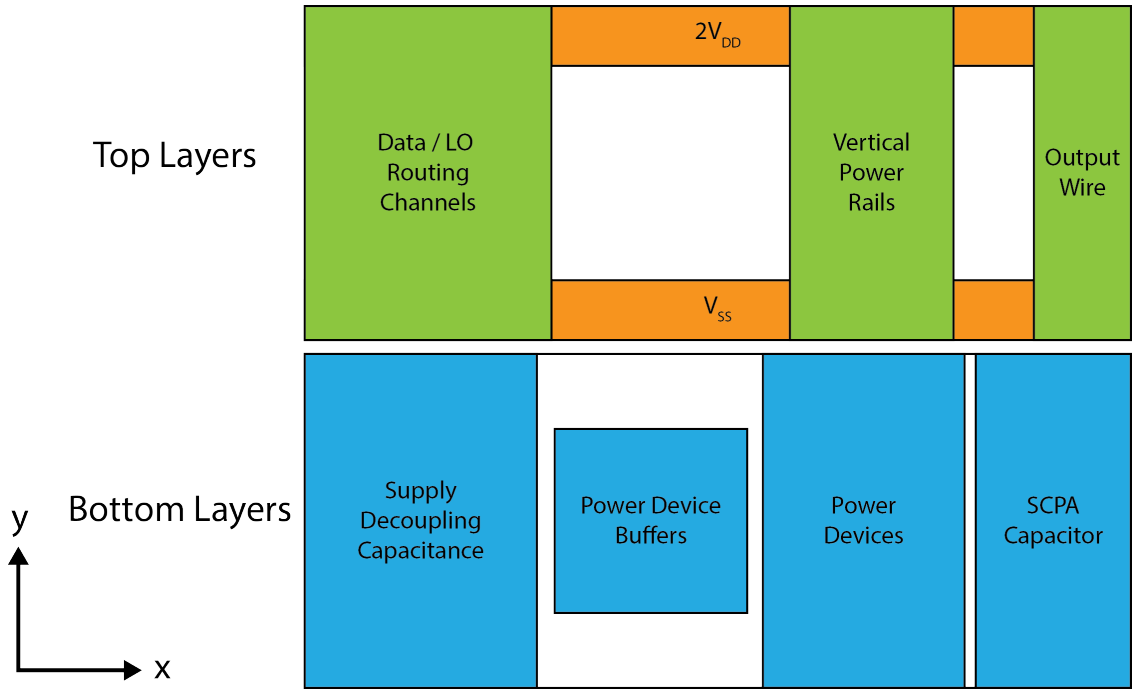


Figure 3.2: SCPA unit cell layout floorplan split into top and bottom layers

independently. Power routing consists of horizontal $2V_{DD}$ and V_{SS} rails and vertical $2V_{DD}$, V_{DD}, and V_{SS} rails. The unit cells form a continuous power grids for V_{SS} and $2V_{DD}$, while V_{DD} needs to be connected across columns external to the SCPA core. A horizontal V_{DD} rail was omitted to save area and simplify routing, and is justified by the significantly lower current draw of V_{DD} supply relative to the other two supplies.

We chose to draw the top level power grid within each SCPA unit cell to give the user maximum control and to ensure matching between unit cells. Since the power grid has a significant impact on the performance of the SCPA, its dimensions are left as a parameter for the user. It is possible to use an external power grid, but this would require the SCPA unit cell to be snapped to that power grid pitch to ensure good DAC performance.

The routing channels consist of a bus of wires which is used to connect SCPA unit cell inputs from outside the array. Key parameters include number of input wire pairs, wire width and spacing. Pairs of N and P inputs run adjacent to each other, separated from other pairs by shield wires tied to V_{SS}. Each non-dummy unit cell connects to exactly one pair of N and P inputs. The N and P inputs are nominally identical signals with the same swing, with the only difference being that the P input is shifted up by V_{DD}. This results in any coupling capacitance between the N and P inputs being effectively nullified. This obviates the need for a shield wire between the N and P input is not needed, reducing capacitance and area.

An issue with this relatively large array is that we will pick up skew on the input and output moving vertically. Since the input and output parameters are set independently, we cannot rely on the output to make up for any skew introduced by the input. The most reliable

option is to minimize the absolute skew by reducing the RC time constant of the wire. The wires are generally quite long, on the order of hundreds of μm, so a thick metal layer was used for the routing channels to reduce resistance and overall wire RC time constant.

The power device buffers are implemented as inverter chains, with one buffer each for the NMOS and PMOS input power devices. Key parameters include the number of stages, fanout per stage, and final buffer width, length, and number of fingers. Since the buffers operate on different supplies (Fig. 3.1), they are placed in separate triple wells to reduce the body effect. Body effect's impact on V_{TH} reduces system efficiency by increasing leakage current in the buffers, and causing crowbar current in the power devices due to mismatches in the delay of the two buffer paths. Though not strictly necessary, the NMOS power device buffer also placed into its own triple well in order to match the PMOS power device buffer. Each triple well has a relatively large minimum spacing requirement from other triple wells. For a smaller unit cell, this area overhead becomes extremely costly, potentially even dominating the area of the unit cell. This was the driving factor in our implementation of unit cell scaling.

Each of the four power devices are placed in separate diffusions. The power devices are collectively surrounded by a single grounded substrate guard ring to reduce substrate coupling into and from the power devices. The transistor width, length, and number of fingers can be set completely independently for each power switch. In practice, the same length and number of fingers was used for every device to simplify the layout.

The SCPA capacitor is implemented as an array of unit capacitor cells in order to allow for robust scaling. Key parameters of this include the unit cell parameters, an array pattern file, and the number of outer dummy cell rings. Dummy cells from both the outer ring and unused cells to implement unit cell scaling are re-purposed as decoupling capacitance between $2V_{DD}$ and V_{SS}. There is potentialy for oxide breakdown between the metal capacitor fingers, but this is not an issue in the technologies used in the two versions of the TX.

The unit capacitor cell consists of a metal-oxide-metal (MOM) finger capacitor with vertical and horizontal routing to connect to the rest of the array. Key unit cell parameters include target capacitance, the top and bottom finger capacitor layers, and metal layers of horizontal and vertical routing channels. Most of the array level routing is included in the unit capacitors, with a grid of connections formed by placing unit capacitors next to each other.

Horizontal routing channels serve as the array level connections to the rest of the SCPA unit cell. The vertical routing consists of two pairs of vertical wires, with one pair connecting to the capacitor terminals and the other pair connecting to the supplies. The unit capacitor's horizontal fingers connects to a single pair, with the specific pair depending on whether the unit capacitor is a normal or dummy cell.

Initially, all wiring was identical between normal and dummy capacitors with only the placement of vias differing in order to maximize matching. However, we found that the total desired capacitance was not linear but affine with the number of unit capacitors according to $C_{tot} = C_{fixed} + \frac{n}{N} C_{unit}$. This directly degrades the linearity of the entire SCPA. The source of this C_{fixed} was found to mostly be overlap capacitance between vertical and horizontal

routing channels. Our compromise was to cut the vertical routes in dummy caps short of running under the horizontal routes, which significantly reduced C_{fixed}. In the case of the used capacitors, this overlap capacitance became part of the C_{unit}. This worsens mismatch between unit and dummy cells due to more differences in layout, but we anticipated that this would be less than the nonlinearity from the large value of C_{fixed}. The normal and dummy unit capacitors are shown next to each other in a 2x1 array in Fig. 3.3.

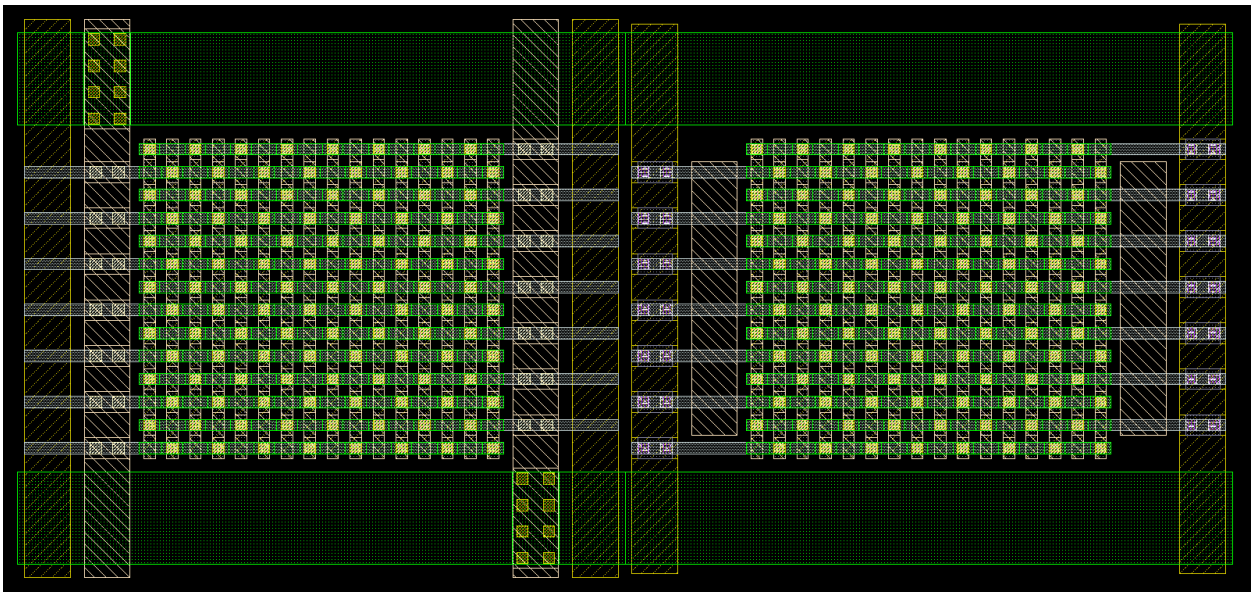


Figure 3.3: Unit capacitor layouts for normal (left) and dummy (right) cells

This is all put together to create the SCPA unit cell, with a specific layout instance shown in Fig. 3.4. Here, "Buffers" refers to the power device buffers, and "Switch + Cap" refers to the power devices and SCPA capacitor. We can verify that this layout follows the floorplan illustrated in Fig. 3.2.

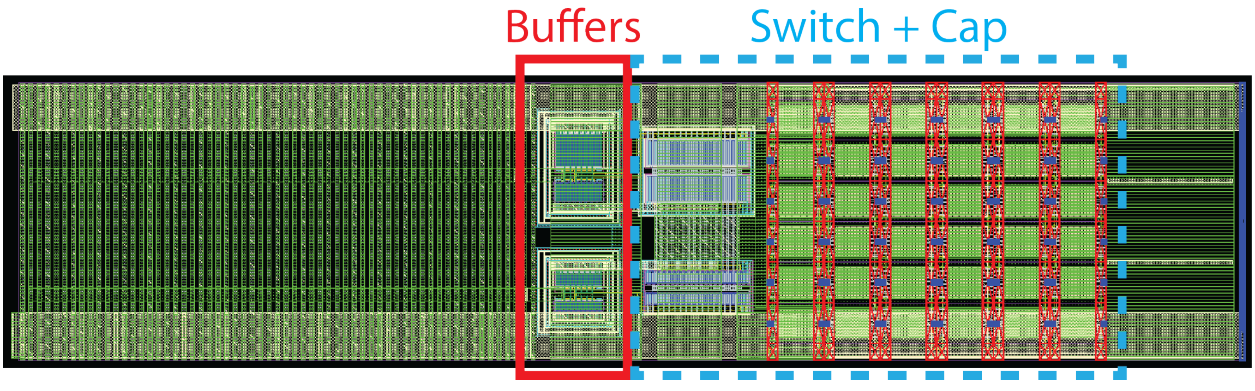


Figure 3.4: SCPA unit cell layout instance

Lastly, we will discuss how the SCPA unit cell implements thermometer, binary, and dummy cells in order to implement a segmented DAC. Each unit cell contains all the power device fingers and unit capacitors associated with the thermometer weight, but only a fraction of these are connected for binary cells. Dummy cells consist of all power device fingers and unit capacitors connected as dummies. This is done in contrast to sizing the unit cell as the smallest binary cell and connecting multiple cells together to implement cell weighing for larger binary and thermometer cells. This latter option has better matching but comes with significant area overhead. This area overhead was primarily due to the implementation of the power switch buffers, which will be elaborated upon later. This area cost drove us to implement the first option as a compromise of maintaining acceptable matching performance while saving area.

Each cell has the same number of total fingers and capacitor unit cells present to maintain good matching, with unused fingers and cells connected as dummies. This restricts the minimum number of fingers to be even multiples of $2^{n_{bin}+1}$, and integer multiples for the number of unit capacitors. The buffers are not scaled because this requirement on the number of fingers would cause a large area overhead. The lack of buffer scaling causes nonlinearity by generating differing input rise times for the power devices of different cell weights.

3.1.2 SCPA Array Generator

The array level PyCell sets parameters for SCPA unit cells and places them in pseudo-differential pairs. An example of a layout instance of this array level generator is shown in Fig. 3.5. Most unit cell parameters are the same across unit cells, with the primary differing parameters being the routing channel pair to use and the unit cell weight. The array level PyCell will generate the correct value of these parameters based on a parameter called an array pattern file.

The array pattern file is a plain text file with the placement and type of each unit cell. Each pseudo-differential unit cell pair is named in the format of $< type >< number >$ where $< type >$ is either 'D', 'B', or 'T' corresponding with dummy, binary, or thermometer weighted cells. The $< number >$ is associated with a specific input, and is required for all types besides dummy cells. The array pattern file used for the SCPA array in TX v1 is shown in Fig. 3.6. The location of each name in the array pattern file corresponds to the location of the unit cell pair within the array. For example, B14 will be in the bottom left of the SCPA array layout.

The parameter *frac_width_used* is a fraction which sets the weight of a unit cell. Computing weights for thermometer and dummy cells is straightforward since these are always 1.0 and 0 respectively, but handling binary cell weighing is less straightforward, especially when multiple sub-PAs exist. Cell weighing is determined with the use of the input number n with the formula $frac\_width\_used = 2^{(n \,\%\, n_{bin})-n_{bin}}$ where n_{bin} corresponds with an additional array-level parameter *bits_binary*. The values of *frac_width_used* for each SCPA unit cell in TX v1 are shown in Fig. 3.7.

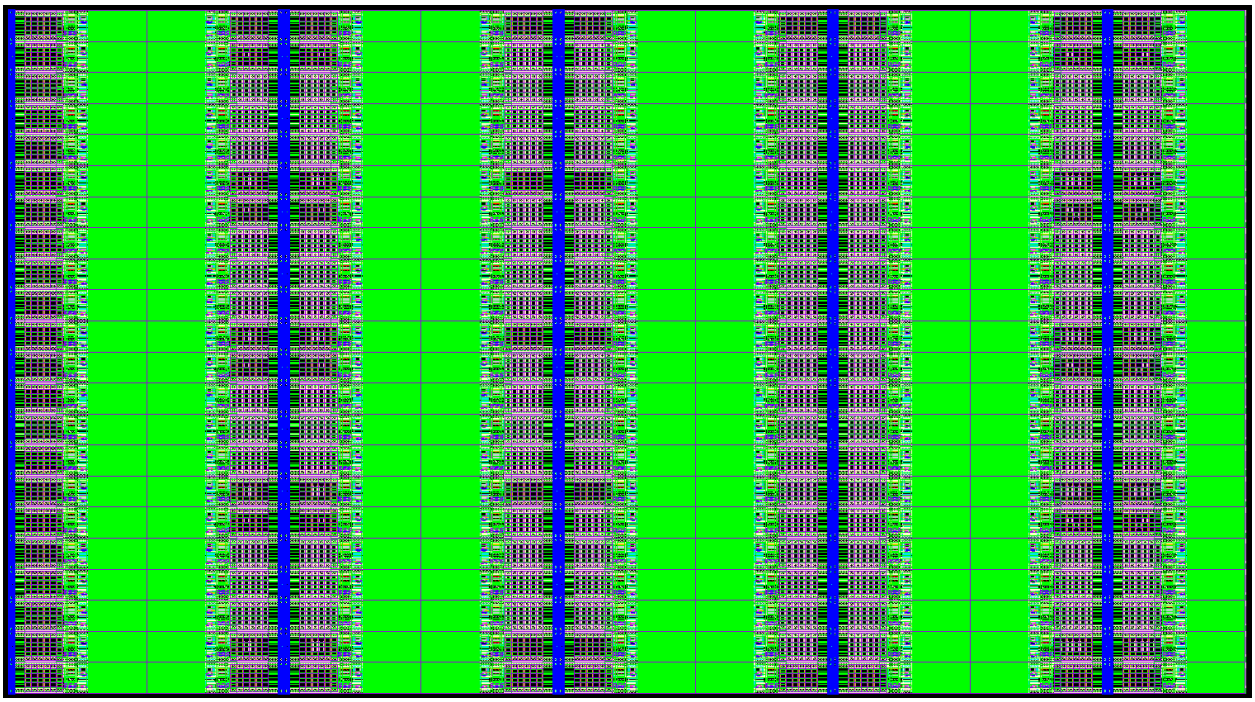


Figure 3.5: SCPA array layout instance

The array pattern in Fig. 3.6 only describes a PA array corresponding to either the 0° or 60° LO phase. Another array needs to be generated for the other phase, after which the PA arrays are placed next to each other to form the overall SCPA core.

The bottom-most cell in each column is connected to the leftmost pair of routing channels, with cells further up the column connecting to further right routing channels. The set of routing channels used depends only on vertical position of a cell in the column, with the exception of dummy cells. Dummy cell have their buffer inputs tied to the rightmost V_{SS} shield wire regardless of position. This means that any duplicate inputs for a given column will not be connected by default, and will need to be connected externally. The array pattern used in TX v1 is set with this restriction is mind, with no duplicate inputs in the array.

3.2 SCPA Generator for BAG 2.0

BAG saw significant developments in the time between the design of the first and second versions of the TX with the release of BAG 2.0. The changes in BAG necessitated a complete rewriting of the SCPA generator, making it a good time to change the functionality and layout of the SCPA generator. The SCPA unit cell now includes additional functionality, and generation of the external column drivers was added to the SCPA generator. All layout generators in TX v2 were implemented using the BAG 2.0 framework [15] in contrast to the pre BAG 1.0 version used in TX v1.

Figure 3.6: SCPA array pattern file

```
B0   T12  T14  B1
T11  T2   T5   T7
T8   T0   T1   T9
T6   T4   T3   T10
B2   D    T13  B3
B4   T28  T29  B5
T26  T17  T20  T22
T23  T15  T16  T24
T21  T19  T18  T25
B6   D    T27  B7
B8   T43  T44  B9
T41  T32  T35  T37
T38  T30  T31  T39
T36  T34  T33  T40
B10  D    T42  B11
B12  T58  T59  B13
T56  T47  T50  T52
T53  T45  T46  T54
T51  T49  T48  T55
B14  D    T57  B15
```

Figure 3.7: *frac_width_used* values for SCPA unit cells

Cell names	*frac_width_used*
D	0
B0, B4, B8, B12	0.0625
B1, B5, B9, B13	0.125
B2, B6, B10, B14	0.25
B3, B7, B11, B15	0.5
T0, T1, ..., T59	1.0

The layout in BAG 2.0 is drawn directly by the BAG framework using layout generators instead of using Ciranova PyCells. Most users will extend classes such as AnalogBase or StdCellBase which draw transistors, handle routing at the lowest several metal layers, and provide a bounding box to meet spacing requirements for these layers. Routing on metal layers above those layers are restricted to a user specified grid, in which each metal layer has a direction and set quantified track width and track spacing values [15]. This routing grid quantizes the layout into tracks, which are spaced by the sum of a single track width and track spacing for each layer. These width and spacing values are generally significantly coarser than the minimum grid resolution of the technology. This loss in resolution is a constraint which simplifies routing, and is a technique which is often used in manual layouts in advanced technologies.

DRC clean layouts generated in BAG 2.0 by enforcing two sets of routing grids, split into user level grid layers and other layers. User level grid layers can be ensured to be DRC clean with the built in BAG 2.0 RoutingGrid class, which has functionality for computing the minimum spacing required for a given layer and wire width in number of tracks. The remaining layers, such as the lower metal layers and front of the line layers like oxide diffusion, are handled in classes like AnalogBase which draw the transistors and their connections, including to the substrate.

BAG layout templates inherit from a set of classes which implement some core functionality. These classes are broadly called XBase, but AnalogBase and TemplateBase are the specific classes heavily utilized in this generator. AnalogBase includes functionality to draw rows of transistors based on a variety of parameters, with additional functions to connect

up groups of transistor fingers into separate devices. Almost all of the leaf cells, defined as the lowest level custom generator cells in our generators, in the SCPA generator hierarchy are AnalogBase cells. In contrast, TemplateBase is a "higher level" template which is used to place and route other instances together.

Unlike in PyCells, wires cannot be moved after placement, and instances cannot have their parameters modified. Though this initially seems very restrictive, wire spacing and locations can generally be computed with user parameter before placement. Additionally, instances can still be moved and their size can be probed prior to placement. This limitation eases implementation of some useful functions such as power and dummy fill.

BAG schematic templates corresponding to the layout templates have been written, but the focus in this section will be on the layout generators and sizing algorithms. For these templates, the layout template is treated as the source template. Users provide layout parameters, after which layout generators create a python dictionary of relevant parameters to pass into the schematic generator to create a schematic. Sample parameters of schematic parameters include widths and lengths of transistors, number of dummy fingers per dummy device. There are situations in which there are separate schematic parameters not tied to the layout, in which case those parameters will be user supplied and added to the schematic parameters generated by the layout.

There are several other benefits to this switch to BAG 2.0. Layout templates are full fledged Python classes, and do not have the restriction of having input parameter types compatible with with open access standards like in the case of PyCells. For example, python dictionaries cannot be used as an input type in PyCells, meaning input lists can get unwieldy. Another issue is that within PyCells, other instances generated by a PyCell are treated like any other layout, with no internal access to their python functions. In BAG 2.0, these functions can be freely accessed, which can useful for accessing design functions of lower level cells.

Layout templates in BAG 2.0 directly generate the layout each time a script to generate the layout is run, in contrast to the PyCells which will generate layout for each new set of parameters provided in Virtuoso. Though this may seem convenient, it also comes with the drawback that the PyCell must generated each time it is opened and not already cached. Layout generation is significantly faster in BAG 2.0 compared to layout generation with pre BAG 1.0, with at least a 20x speed increase from generating the entire SCPA between the two verisons.

The change in the method of layout generation and introduction and enforcement of routing grids meant that the SCPA generator had to be completely rewritten. However, many ideas were carried over such as the overall floorplanning and hierarchy of layout generators. Most of the overall hierarchy and structure of the SCPA core in TX v2 is very similar to TX v1, with a couple key changes.

One major change from the first version is that the data mixing, IQ combining, and level shifting have been moved into each unit cell to reduce the number of input routing signals, saving a significant amount of area in each unit cell. The old design required that each unit cell have $3n_{in,col} + 1$ routing channels while the new design requires $3n_{lo,pair} + n_{in,col} + 1$,

where $n_{in,col}$ is the number of maximum number of unique data inputs in a column for the array. These numbers also include shield wires to isolate different input wires. We opted for an array pattern with $n_{in,col} = 20$ in both designs, meaning the number of routing channels was reduced from 61 to 33 per unit cell, or a total reduction from 976 to 528 routing channels across the entire SCPA. The routing channel wire capacitance dominates the SCPA input capacitance so any reduction in the number of channels helps reduce power consumption. The reduction of routing channels also had a significant effect on the unit cell area in the final instance used in TX v2, with a reduction of routing channel area by 54% lead to an overall reduction of the unit cell area by 23.3%.

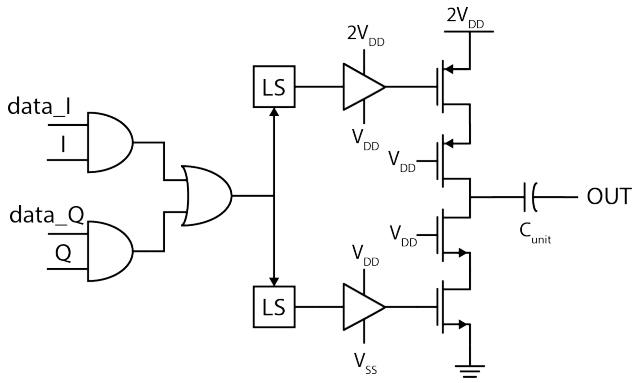


Figure 3.8: SCPA v2 unit cell schematic

Another key change in the SCPA generator is the generation of the external SCPA column driver along with the SCPA core. This external driver selects the proper I, Q LO phase, and drives the long routing channels for the I, Q data and LO signals. We will begin the discussion of the SCPA generator for BAG 2.0 at the array level, followed by the column drivers, and concluding with the unit cell level.

3.2.1 SCPA Array Generator for BAG 2.0

Each SCPA unit cell takes in I and Q data inputs as well as corresponding LO phase pairs. The data input and LO signals are specified using two user supplied array pattern files, which are text files describing how the data input and LO signals associated with each differential SCPA unit cell. The data input file has the same input naming conventions as the array pattern file from Section 3.1. These two files are necessary to exactly specify the sub-PAs for both harmonic cancellation and mixed-signal filtering. This was chosen to allow maximum flexibility to the user in placing different sub-PAs and controlling the aspect ratio of the overall array. It should be noted that the mixed-signal filter taps will differ both in data and LO signals since they take on a delayed version of the sign bit, which means that the LO could be phase shifted by 180° from non delayed versions.

The array pattern files are plain text files, with the specific patterns used in TX v2 shown in Fig. 3.9a for data and Fig. 3.9b for phase. In the data pattern, D, B, and T

```
T28  T18  T16  T27  T58  T48  T46  T57
T24  B4   B6   T23  T54  B12  B14  T53
T20  T15  D    T21  T50  T45  D    T51
T22  B7   B5   T25  T52  B15  B13  T55
T26  T17  T19  T29  T56  T47  T49  T59
T28  T18  T16  T27  T58  T48  T46  T57
T24  B4   B6   T23  T54  B12  B14  T53
T20  T15  D    T21  T50  T45  D    T51
T22  B7   B5   T25  T52  B15  B13  T55
T26  T17  T19  T29  T56  T47  T49  T59
T13  T3   T1   T12  T43  T33  T31  T42
T9   B0   B2   T8   T39  B8   B10  T38
T5   T0   D    T6   T35  T30  D    T36
T7   B3   B1   T10  T37  B11  B9   T40
T11  T2   T4   T14  T41  T32  T34  T44
T13  T3   T1   T12  T43  T33  T31  T42
T9   B0   B2   T8   T39  B8   B10  T38
T5   T0   D    T6   T35  T30  D    T36
T7   B3   B1   T10  T37  B11  B9   T40
T11  T2   T4   T14  T41  T32  T34  T44
```

(a) Data inputs

```
5 5 5 5 7 7 7 7
5 5 5 5 7 7 7 7
5 5 5 5 7 7 7 7
5 5 5 5 7 7 7 7
5 5 5 5 7 7 7 7
1 1 1 1 3 3 3 3
1 1 1 1 3 3 3 3
1 1 1 1 3 3 3 3
1 1 1 1 3 3 3 3
1 1 1 1 3 3 3 3
4 4 4 4 6 6 6 6
4 4 4 4 6 6 6 6
4 4 4 4 6 6 6 6
4 4 4 4 6 6 6 6
4 4 4 4 6 6 6 6
0 0 0 0 2 2 2 2
0 0 0 0 2 2 2 2
0 0 0 0 2 2 2 2
0 0 0 0 2 2 2 2
0 0 0 0 2 2 2 2
```

(b) LO phase

Figure 3.9: SCPA array pattern text files

represent dummy, binary, and thermometer cells respectively, with the numbers afterwards representing different unique inputs. In the phase file, each number represents a unique phase quad (I, IB, Q, QB). Dummy cells connect the data inputs and LO inputs to V_{SS}, so unique data inputs don't need to be specified. Phase inputs for dummy cells are specified but not used. Overall, these example pattern files implement a SCPA with 2 LO phase quads and 4 mixed-signal filter taps.

Even though there are 2 LO phase quads generated by PIs, there are 8 unique phase quads for the SCPA unit cells. Each sub-PA in the mixed-signal filtering takes in different data inputs consisting of a sign bit and amplitude bits. The sign bit sets whether the base LO phase or the 180° phase-shifted version will be used for a given sub-PA. Since each of these data inputs can be completely independent, a separate phase quad is needed for each filter tap, for a total of 8 phase quads for the overall array.

The array patterns used can significantly impact the size of the SCPA unit cell. The number of routing channels in each unit cell depends directly on the maximum number of unique data and phase inputs across columns. The template is flexible enough to handle general sets of inputs, but clever arrangements of sub-PAs can help to reduce the total number of routing channels and simplify the wiring into the array.

We want to place sub-PAs (filter taps) with the same input data but different LO phases (0°, 60°) physically near each other to mitigate differences due to process gradient effects and improve harmonic cancellation. Placing these sub-PAs in the same columns reduces the

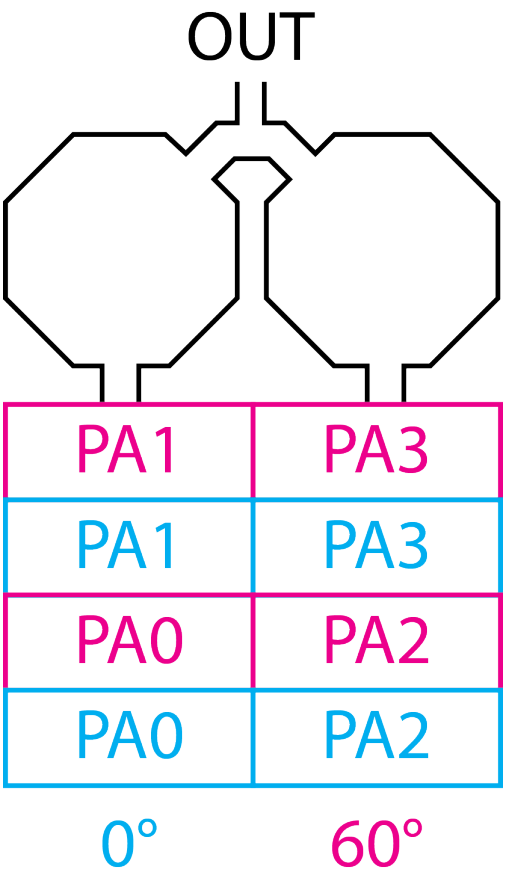


Figure 3.10: SCPA array floorplan

total number of routing channels since the data inputs are shared between the sub-PAs. The SCPA is then constructed by placing these columns next to each other horizontally.

In TX v2, the SCPA was designed to take up most of width of the chip, leaving room for routing on either side. This mostly sets the aspect ratio of the entire SCPA, with the only remaining choices being in how to distribute the sub-PAs. Excluding the outer dummy ring, the SCPA is set to be a 20 x 8 array given the unit parameters used. This means the optimal pattern uses 10 x 2 arrays for the sub-PAs, which is not used in the array pattern used in TX v2 (Fig. 3.9). In TX v2, the sub-PAs were arranged into 5 x 4 arrays were used in hopes of a more square aspect ratio, though the effectiveness of this depends on the dimensions of the unit cell. In the 5 x 4 arrangement, we have the same number of data wires but double the LO phase wires as compared to the 10 x 2 arrays. The final layout instance of the SCPA array (core) with column drivers used in TX v2 is shown in Fig. 3.11.

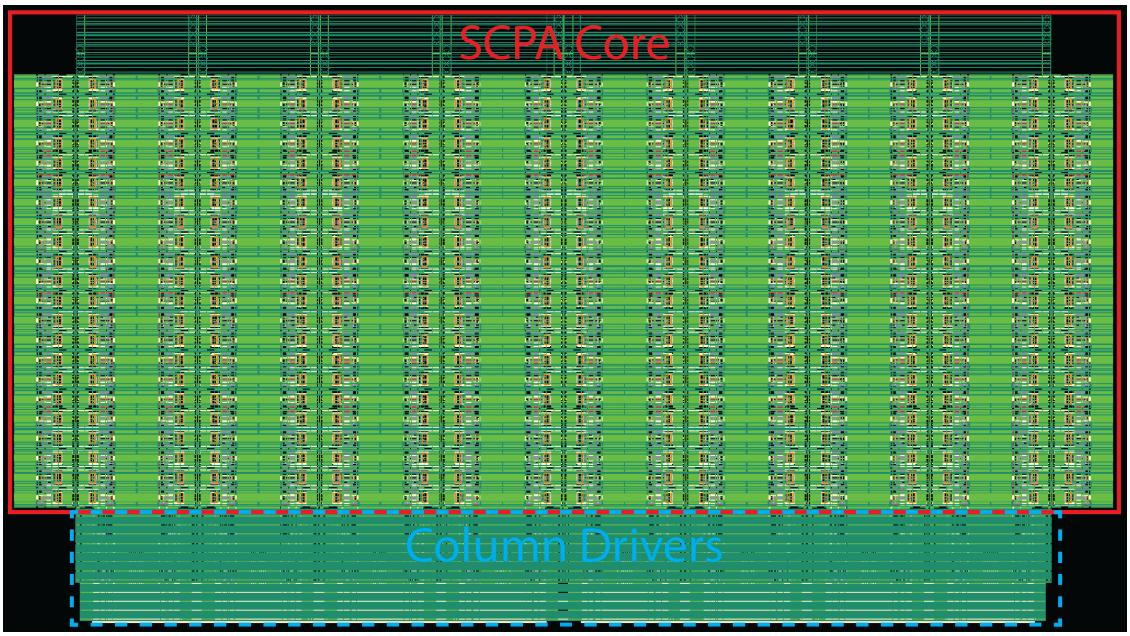


Figure 3.11: SCPA top level layout instance with core and column drivers

3.2.2 SCPA Column Driver Generator for BAG 2.0

The main purpose of the external column drivers are to buffer the IQ data and LO signals into the unit cell inputs. In addition to these signals, column drivers take in CLK_SLOW and RSTB signals. CLK_SLOW is used to retime data signals using flip flops, and RSTB is a synchronous reset used to set the SCPA I, Q data signals to 0 for debugging purposes. The external column driver array consists of a row of columns aligned to the SCPA unit cells for data inputs, LO inputs, and vertical supply wires.

The column driver array takes in SCPA array parameters in order to compute the placements of these signals and supplies, the total number of columns, and the number of data and LO drivers required for each column. The number of data and LO drivers per column can also be set directly to aid in debugging. The column driver array is implemented as an array of individual column drivers, with a layout instance of a single column shown in Fig. 3.12.

The data inputs represent signed integers in a sign-magnitude format, where the MSB represents the sign. The sign is implemented by swapping the LO and LOb signals by using butterfly switches controlled by the MSB of the data input. For these 25% duty cycle LO signals, the LOb signal is not the logical complement of LO, but instead the LO signal phase shifted by half of an LO period. A separate LO signal is needed for each mixed-signal filter sub-PA in a given column due to the influence of the sign bit on the LO signals.

The column driver (Fig. 3.13) retimes and buffers the magnitude portion of the IQ data using an inverter chain. The LO driver is more complex, with the driver schematic for a signal LO signal pair shown in Fig. 3.14. The key function of this LO driver is to swap

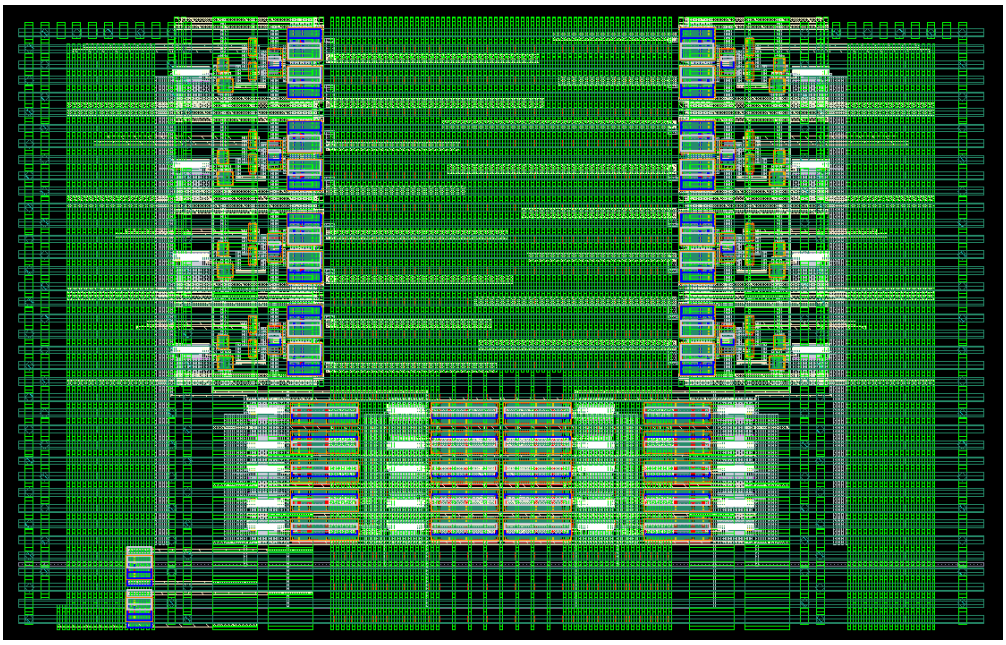


Figure 3.12: SCPA column driver layout instance

LO and LOb when the sign bit is 1, and to buffer the LO signals. Swapping of LO phases is implemented using transmission gate based multiplexers (MUX). The sign bit is retimed and used to generate a buffered version of itself and its complement are generated using a 2-3 splitter, which are labeled FLIP and FLIPb.

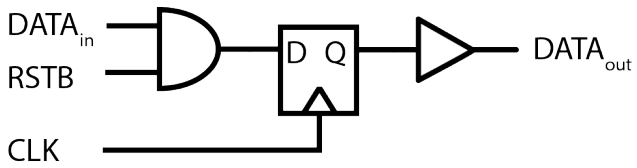


Figure 3.13: Column driver data driver schematic

The data is retimed using flip flops clocked by the data clock to ensure that each column of the sub-PA receives the input data at the same time. This avoids any skew in data signals from routing differences to the external driver. Each column includes buffers for the CLK_SLOW and RSTB signals which drive the individual data and LO drivers.

The flip flops are not implemented a template inheriting from AnalogBase or TemplateBase unlike all other cells in the column driver. Instead, the flip flop used is a custom fixed layout imported to BAG as the StdCellBase template. The main parameter here is the path to a YAML file [50] containing information such as the specific layout cell, the size of the cell, and each port along with its associated layer and location. Additionally, the YAML file contains information on both the local private routing grid along with the bottom layer of the user routing grid and the expected user set routing grid. A major restriction of using

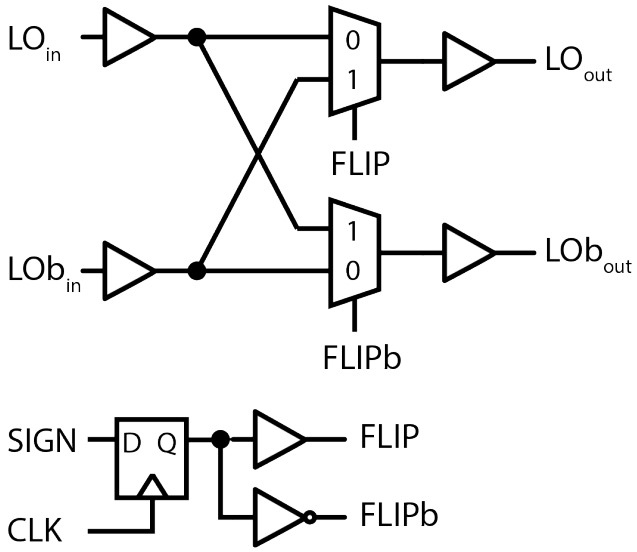


Figure 3.14: Column driver LO driver schematic

StdCellBase is that the ports must be on the routing grid of their specified layer. The best practice is to ensure all ports are on the top private layer and on grid, as this allows the cell to be used across a wider range of user specified routing grids.

Like in the pre BAG 1.0 version, each unit cell is to be placed next to another mirrored unit cell to form a psuedo-differential pair, with the routing side of the unit cell shared. Previously, no routing channel inputs were shared since all inputs were already mixed up to RF. However, the data inputs can be shared between the cells in the pseudo-differential pairs in the second version to save area and reduce power consumption by reducing the total number of long routing channels. We also route the I and Q LO signals in signal phase pairs (phase shifted by $T/2$), allowing easy access of both phases for the positive and negative unit cells of the pseudo-differential pair.

3.2.3 SCPA Unit Cell Generator for BAG 2.0

Similar to the discussion in Section 3.1.1, we begin design of the generator with a floorplan. The floorplan of the SCPA unit cell BAG layout template (Fig. 3.15) is very similar to the previous version in TX v1. The main difference is that the power device buffers have been expanded into the data driver, which includes more functionality that just input buffering.

The SCPA unit cell core consists of three main components: the data driver, the power switches, and the capacitor, placed from left to right. The data driver consists of the IQ mixer / combiner and two level shifters to drive the power switches. The IQ mixer / combiner is relatively simple and consists of a single AnalogBase. The level shifter consists of an AC coupling capacitor, a feedback inverter, and inverter chain placed from left to right. A wrapper cell instantiates the level shifter along within a deep n-well delimited by a substrate

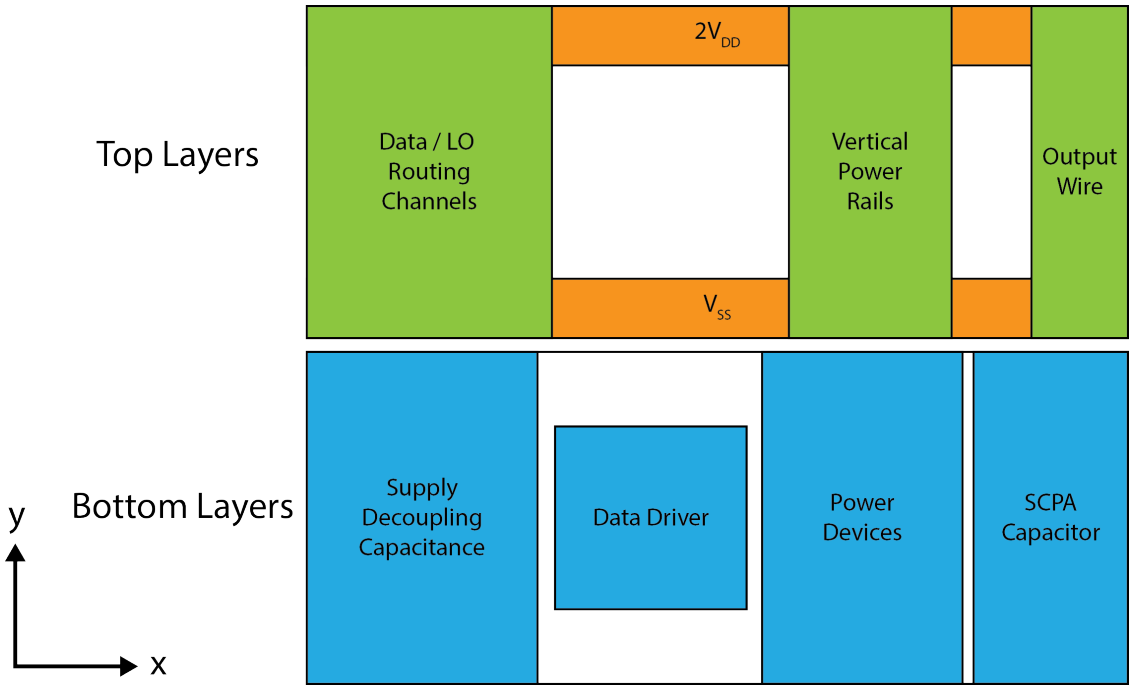


Figure 3.15: SCPA unit cell layout floorplan split into top and bottom layers

ring. The purpose of this is to have a separate bulk node to reduce the body effect on the NMOS devices. The data driver also has a wrapper with an optional substrate ring to prevent substrate coupling to and from nearby cells. This is meant to isolate the substrate of the power switches, which draw significantly more current than the driver.

The data driver consists of two level shifters for the NMOS and PMOS inputs of the power switches, which will be referred to as the N and P level shifters respectively. The P level shifter is placed above the N level shifter, with the left side of their bounding boxes (BBox) aligned. The IQ mixer / combiner is placed to the left of these level shifters, with the V_{DD} rail aligned to the V_{SS} rail of the P level shifter. We connect these rails together by drawing a horizontal wire connecting the two. This alignment also ensures that the horizontal IQ mixer / combiner output will not be aligned with the P level shifter horizontal input, and will be sufficiently far to ensure no DRC issues when connecting the two with a single vertical wire.

Similar to the first version, the final buffers driving the input power devices consist of two inverter chains placed in separate triple wells. In this version of BAG 2.0, deep n-well guard rings are implemented using the DeepNWellRing BAG template. This template takes in the bounding box of the cell to surround and returns the lower left coordinate to place that cell to ensure the layout is DRC clean.

Most of the difficulty in generating the layout of the unit cell core and data driver comes from routing the inputs, outputs, and supplies of the lower level instance (data driver, power switches, capacitor) together, with a variety of approaches used at different levels. In

particular, routing the level shifter outputs to the power switch inputs is challenging due to how the parameters for these instances have no fixed relationship.

We must implement this routing in a way that is ensured to work for a reasonable range of input parameters. This is simple if the two wires to be connected have different directions, such as a vertical wire connecting to a horizontal wire, separated by 1 layer. However, the outputs of the level shifters and input of the power devices are wires on the same layer with the same direction.

In the simplest case where the wires are of the same width and on the same track, the wires can be extended until they touch or overlap. This generally will only happen if the BAG layout cells are written specifically to ensure this, since it is unlikely for both wires to be aligned and of the same width. A more robust approach using one extra layer is to connect to a wire of the opposing direction, such as horizontal wires connecting to each other using an intermediate vertical wire. However, this can cause DRC issues with via spacing or even overlap in extreme cases, as well as DRC issues from small corners or kinks on a given metal layer.

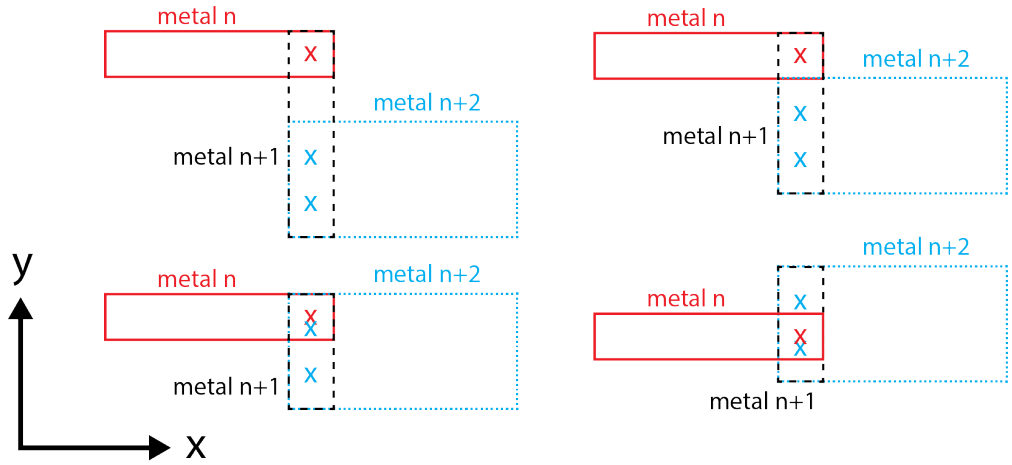


Figure 3.16: Technique to connect two wires on the same layer (direction)

We can use 2 extra layers (for a total of 3 layers) to avoid these DRC issues. Assuming both wires are originally on metal layer n, we will bring one wire up to n+2, and connect the two wires in a middle using a wire on layer n+1. Several different cases are shown in Fig. 3.16 assuming metal layer n is horizontal, but the same idea applies regardless of the layer's direction. Since the vias are now on different layers, we eliminate potential via spacing errors, and also can avoid any alignment and width mismatch issues for the wires. This was the technique used to connect the level shifter outputs to the power device inputs. This technique is useful when attempting two connect wires from two separate BAG instances without a fixed relationship between the connecting ports. The main downside to this technique is that it requires a 3 layers, which may not be viable in processes with a small amount of metal layers.

Another difficult set of connections in the unit cell are between the horizontal power rails of the sub-blocks and the unit cell's power grid. The way these connections are made differs depending on if the vertical thick metal are below or above the horizontal one. This version of the unit cell only implements connections for metal grids where the vertical thick metal is above the horizontal one. For both the $2V_{DD}$ and V_{SS} supplies, the PA driver and switch rails are connected together on a thin metal layer. The switch rails are brought up to the highest thin metal layer, which runs horizontally, which connects to the corresponding thick metal vertical rail. For the V_{DD} supplies, the cascode bias is connected to both horizontal rails in the PA driver. The PA driver rails are brought up to the top thin metal layer which connects directly to the thick metal vertical V_{DD} rail.

Like in the first version of the SCPA generator, the SCPA capacitor is implemented as an array of unit cells. The capacitor unit cell includes a MOM capacitor with additional horizontal and vertical routing. The capacitor's physical size is set by the capacitance value, array pattern, and metal layers used. The physical size of the MOM capacitor is specified at a lower level in the number of horizontal and vertical metal fingers. This version of the generator adds an algorithm to physically size the capacitor, shown in Fig. 3.17.

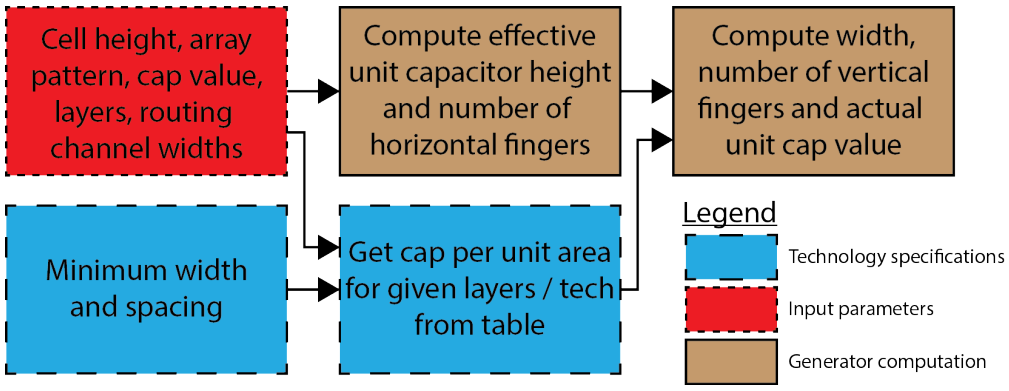


Figure 3.17: Capacitor sizing flowchart

The effective unit capacitor height is computed by taking the SCPA unit cell height, dividing by the number of number of capacitor rows (set by the array pattern and number of dummy rings), and subtracting the height taken up by horizontal routing channels and spacing (Eq. 3.1). We then divide this by the MOM finger pitch and take the floor to get the number of horizontal fingers (Eq. 3.2). The number of vertical fingers is computed by using Eq. 3.3, and the predicted pre-extraction capacitance C_u is computed using Eq. 3.4.

$$h_{eff,unit} = \frac{h_{total}}{n_{rows}} - h_{route} \tag{3.1}$$

$$f_h = \left\lfloor \frac{h_{eff,unit}}{p_h} \right\rfloor \tag{3.2}$$

$$f_v = \left\lceil \frac{C_{u,des}}{C_{den}} \cdot \frac{1}{f_h p_v p_h} \right\rceil \tag{3.3}$$

$$C_u = f_v f_h p_h p_v C_{den} = WHC_{den} \tag{3.4}$$

The capacitance per area C_{den} depends on the technology and layers used. A table of C_{den} should be generated once for each technology, and should be generated for all possible combinations of top and bottom metal layers, excluding thick metal layers. C_{den} is computed by generating the layout of a relatively large MOM capacitor, getting the capacitance using layout extraction, and dividing by the area of the MOM capacitor.

Before the capacitor can be sized, the SCPA unit cell height must be determined. This makes determining the unit cell size (width and height) somewhat complicated and involves both the unit cell and the unit cell core. The algorithm to determine the size is shown in Fig. 3.18, which splits the steps between the unit cell and the core. The main thrust of this requires the height to be computed first, which is then used to compute the width. The most important thing about this algorithm is that everything can be computed before placement. Like all instances, the unit cell core size can be looked up before placement, and all routing can be computed before placement using track widths and BAG functions to determine spacing given those widths.

All these changes combine to form the SCPA unit cell generator. Two different instances of the unit cell are shown below in Fig .3.19, with widely varying input parameters. Everything from the number of routing channels, device sizes, and output capacitance values differs between the two.

These changes in the unit cell and the external driver culminate in a 1.9x area reduction for the SCPA core and a 2.2x reduction for SCPA with column drivers. Some of this can attribute this to technology scaling from a 65 nm to 28 nm process, which gives smaller devices and denser MOM capacitors. However, a lot of the area does not scale with technology, since much of it is set by routing on thick metal layers. These layers do not scale very much, and similar widths to the TX v1 to minimize resistance and improve SCPA output power and efficiency. Most of the area saved here came from reducing the number of routing channels in each SCPA unit cell.

3.3 Phase Interpolator Generator for BAG 2.0

The phase interpolator generator is implemented as a gilbert-cell based PI with phase controlled achieved using current steering DACs (IDACs) (Fig. 3.20). This PI topology requires that the input signals be triangle waves in order for proper operation, so the generator also includes input integrators to convert the square wave LO input signals into triangle waves. The integrators have their bias currents set by an IDAC, which allows for operation across a wide frequency range. The output of this PI is fed into a CML to CMOS converter to get full swing digital signals, which used an existing fixed layout, and was not BAG generated.

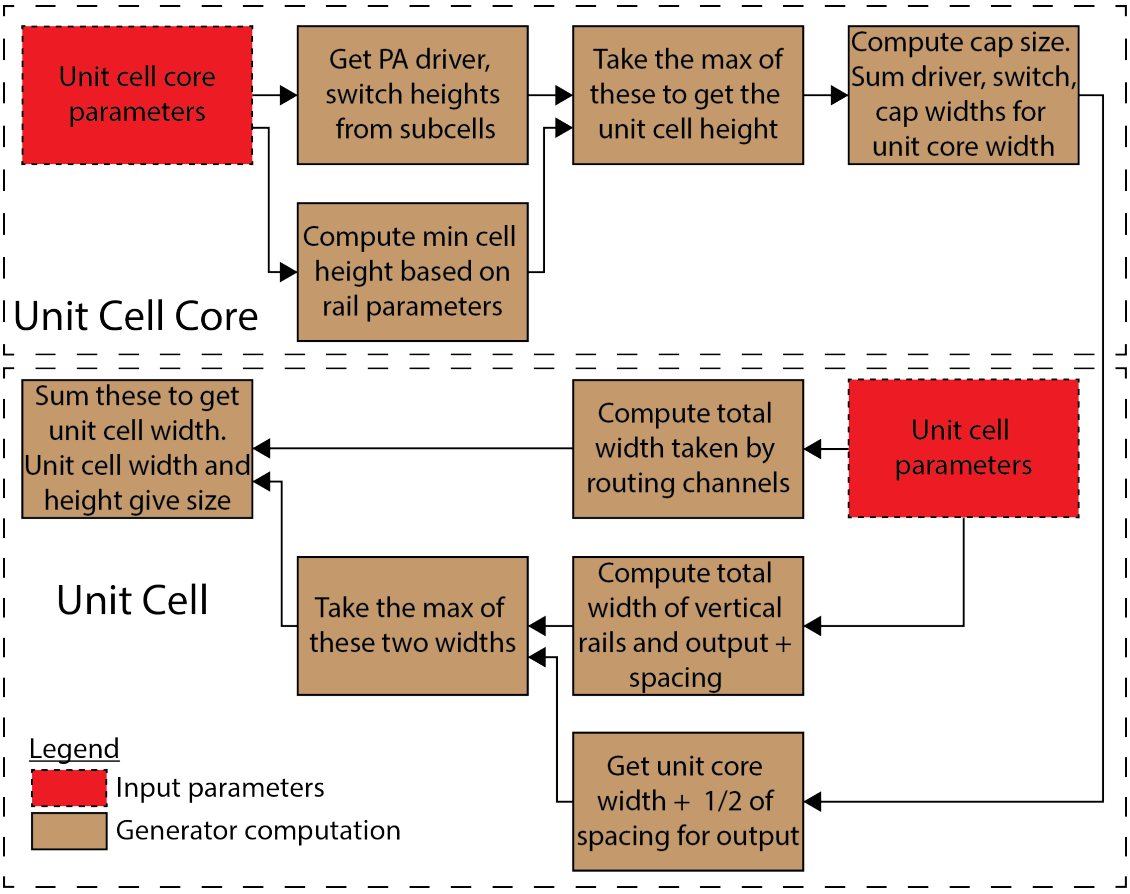


Figure 3.18: SCPA unit cell sizing flowchart

We chose to implement this generator for a couple of reasons. The first reason is that this block's performance is critical to the effectiveness of the harmonic cancellation technique, as discussed in Section 2.4. The second is that this is a useful block used not only in RF applications, but also for mixed-signal circuits such as in serializer and deserializer (SERDES) blocks. The PI also has consists of many analog circuits, in contrast to the heavily digital or hard switching circuits comprising the SCPA. Though the PI core is generated, we will omit discussion on this since the generator is very simple. We will begin discussion of this generator on the most critical block which sets the minimum guaranteeable harmonic cancellation, the PI IDAC. This is followed by discussion of the PI input integrator.

3.3.1 Phase Intepolator Current DAC Generator for BAG 2.0

The most critical component of the PI are the IDACs which control the output phase because their resolution and linearity are tied to the PI performance. For relatively high resolution DACs, care needs to be taken layout to ensure matching. The IDAC is implemented as an array of unit cells configured either as normal, mirror, or bias cells. Differences between the

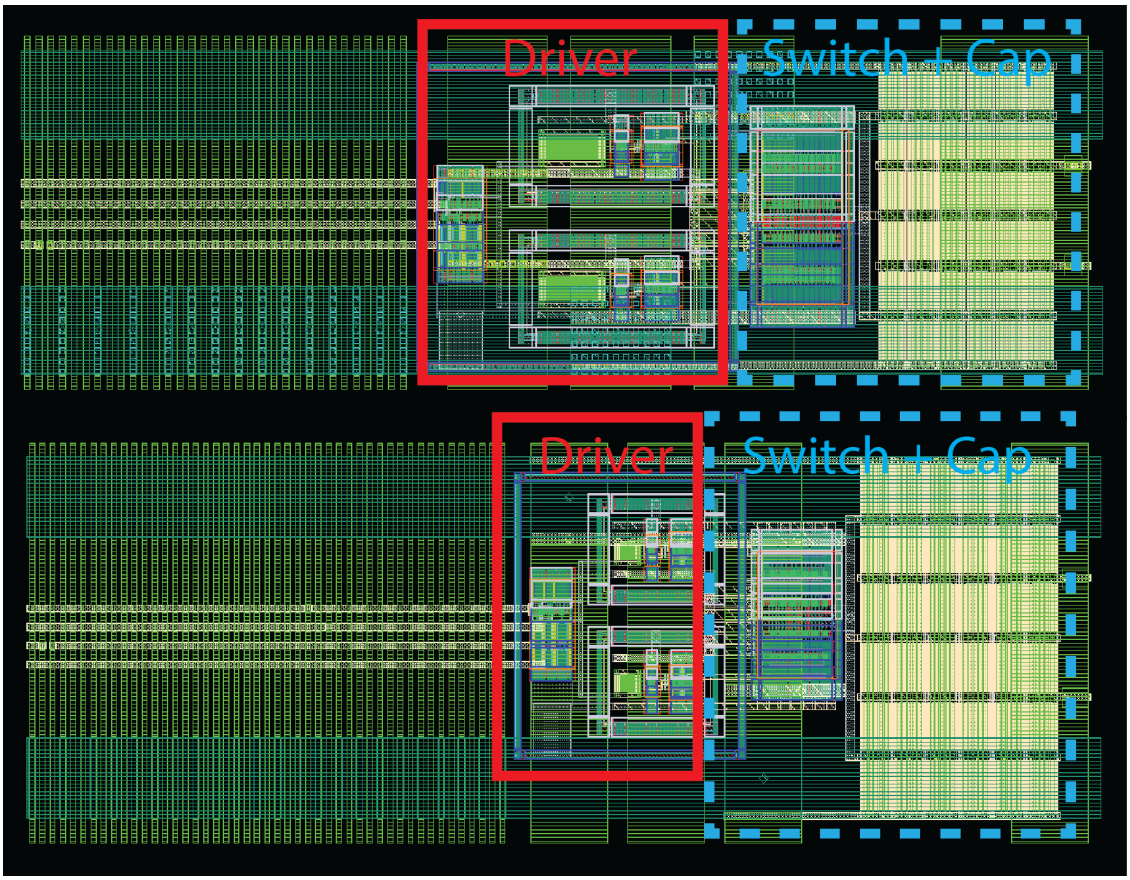


Figure 3.19: SCPA unit cell layout instances with two different set of parameters

three types of cells are minimized in order to maximize matching (Fig. 3.24).

In an earlier section, we discussed how harmonic cancellation requirements set the phase resolution and thus total number of bits needed for the PI. A critical issue for DACs is ensuring that random V_{th} variation [38] does not significantly degrade the effective number of bits (ENOB) of the DAC, which is usually set by the differential nonlinearity (DNL) of the DAC. Depending on the application, a variety of techniques may be used to mitigate the effect of mismatch, but this design deals with this solely by upsizing the current source device in the unit cell. We want to ensure that $2k\sigma_{DNL} < I_u$, where k is the number of standard deviations we desire, and I_u is the current of a single unit cell. M is the maximum number of unit cells switched in a single transition across code. The value of M depends on the structure of the DAC (thermometer, binary, or segmented), shown in Eq. 3.7. W and L represent the effective width and length of the current source device and includes multiple fingers and device stacking, respectively.

$$\frac{I_u}{2k} \geq \sigma_{DNL} = \sqrt{M}\sigma_{I_u} \tag{3.5}$$

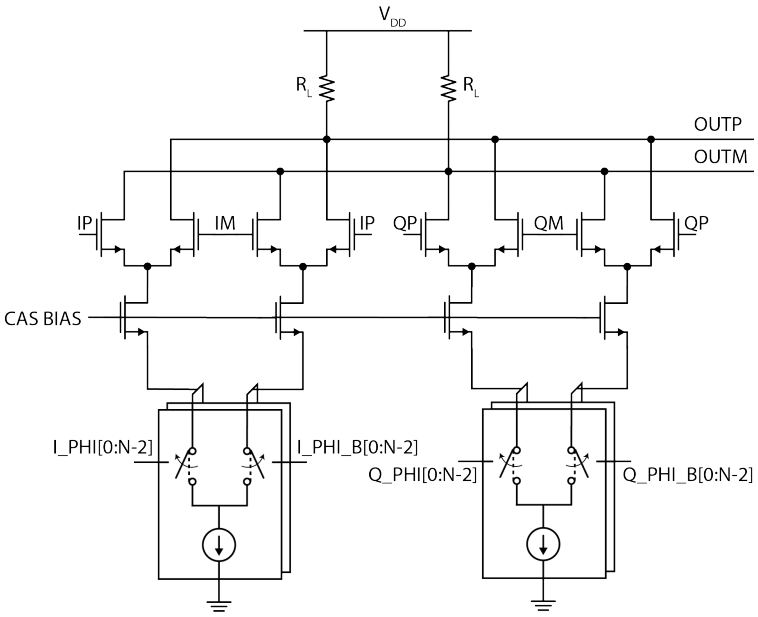


Figure 3.20: Phase interpolator core

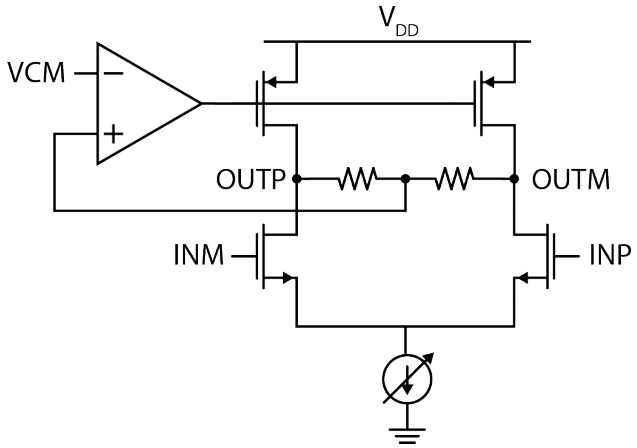


Figure 3.21: Phase interpolator input integrator

$$\sigma_{I_u} = g_m \sigma_{V_{TH}} = \frac{2I_u}{V^*}\sigma_{V_{TH}} = \frac{2I_u}{V^*} \cdot \frac{A_{VT}}{\sqrt{WL}} \tag{3.6}$$

$$M = \begin{Bmatrix} 1 & Thermometer \\ 2^{n_b} - 1 & Binary \\ 2^{n_b+1} - 1 & Segmented \end{Bmatrix} \tag{3.7}$$

The equations above can be manipulated into our main constraint based on transistor area (Eq. 3.8) to meet the DNL specification, with A_{min} specified in Eq. 3.10. Additionally,

each process has a minimum transistor width constraint (Eq. 3.9), which can force the unit cell to be upsized to be manufacturable. This can result a $W = W_{min}$, even if the DNL specification could be met with a smaller value of W in theory.

$$WL \geq A_{min} \tag{3.8}$$

$$W \geq W_{min} \tag{3.9}$$

$$A_{min} = M \left(\frac{4kI_u A_{VT}}{V^*} \right)^2 \tag{3.10}$$

$$W = \frac{I_u}{i_D\left(V^*, L\right)} \tag{3.11}$$

Here, i_D is the current per unit width, which will be referred to as the current density, and is a function of V^* and L. This allows us to write W as a function of L and allows us to solve for the single variable L. A plot of i_D vs V^* for a given L can be generated by simulating the circuit in Fig. 3.22. The transistor length is set to the input L. The transistor width W_{test} is set to a fixed value, and V_{DS} is set to a constant value such as V_{DD} or $V_{DD}/2$. We then sweep V_{GS} over a reasonable range of values (such as 0 to V_{DD}), compute $V^* = 2I_D/g_m$, where I_D is the drain current the test transistor, and generate a plot of $i_D = I_D/W_{test}$ vs V^*. The specific value of i_D for the desired V^* can then be interpolated from this data.

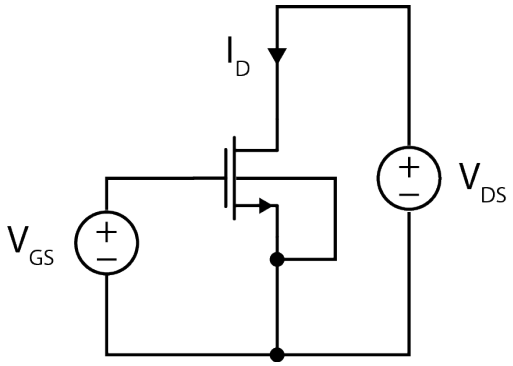


Figure 3.22: Test circuit to simulate i_D vs V^*

Combined with the constraints from Eq. 3.8 and Eq. 3.9, we can solve for L and W using the algorithm shown in Fig. 3.23. The value of L can be incremented by a set value, or chosen from a list of possible lengths in technologies which have tightly quantized length values. The algorithm starts with L_{min} and will find the smallest manufacturable W, L which results in the desired current and meets DNL specifcations. In our design, it should be noted that we enforce an even number of transistor fingers, so W_{min} is actually twice the minimum width of a single finger.

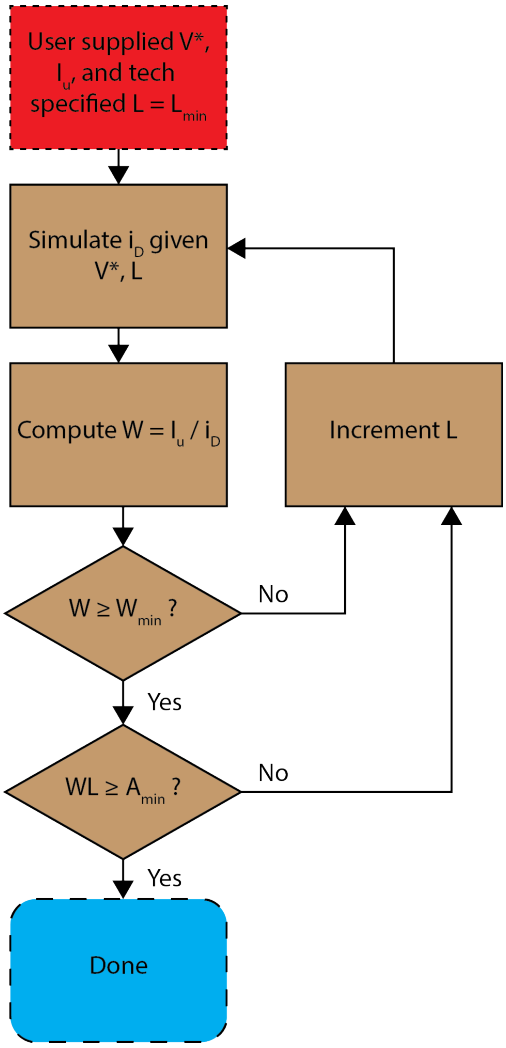


Figure 3.23: PI IDAC current source device sizing flowchart

The PI IDAC unit cell consists of a current source device with two switches to steer the current. The layout of the unit cell is implemented using the AnalogBase class in BAG. The current source device is placed in its own diffusion, with the two switch devices placed in a separate diffusion above the current source device. One restriction of the AnalogBase class is that all transistor fingers must have the same length. This is an issue since the switches want a shorter channel length for faster operation, while the current source requires a longer channel for higher output impedance and better matching. To get around this, the current source device uses device stacking to achieve a higher effective length. This stack factor is a user supplied parameter.

The number of device fingers is the same for every row in AnalogBase. In order to ease routing, each row requires at least 2 dummy fingers each on the left and right edges of the switch and current source rows. Leftover dummy fingers are reconfigured as decoupling

capacitance for the voltage bias input to allow for distributed decoupling capacitance (decap). Depending on the user parameters, it is possible to have either no decap, only switch decap, only current source decap, or both switch and tail decap. Additional decap can be added by increasing the number of dummy finger pairs, a user specified parameter.

Input weighing for binary and thermometer cells is implemented by connecting multiple normal unit cells to the same input, with the lowest input being connected to a single unit cell. Each row of the IDAC array shares the same input in order to reduce the space for input routing in each unit cell. However, multiple rows can be driven by the same input in order to allow for control of the array aspect ratio. Unused cells in binary weighted rows are configured as mirror cells, with the user able to specify additional mirror rows to more finely control the mirror ratio.

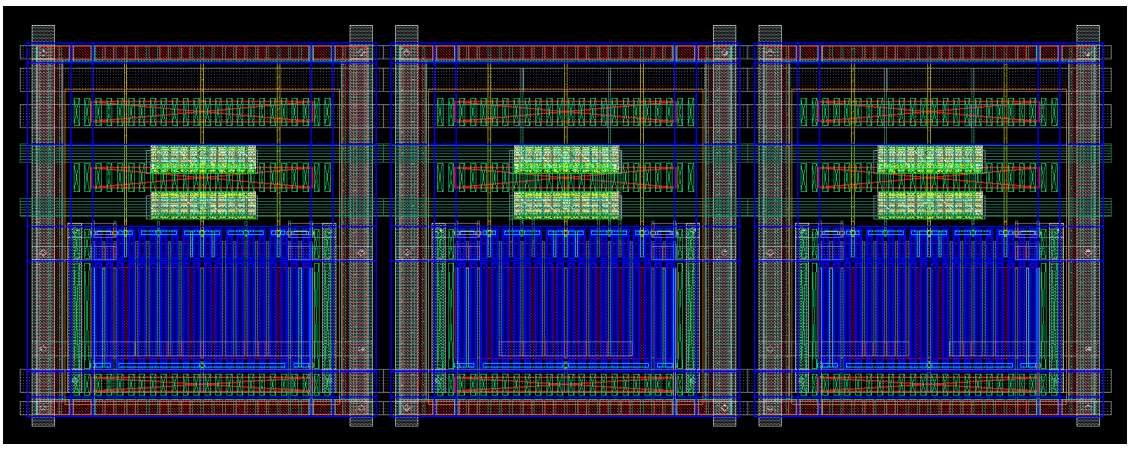


Figure 3.24: PI IDAC normal, mirror, and dummy unit cells from left to right

In contrast to the SCPA array generator, the array pattern is not set by a user specified pattern but generated using an algorithm taking in user parameters such as the total number of cells per row (excluding dummies) n_{cpr}, number of total bits n_{tot}, number of binary bits n_{bin}, and number of extra mirror rows. The array pattern generation algorithm is as follows:

1. Compute the number of binary rows with $\sum_{k=0}^{n_{bin}-1} \lfloor 2^k / n_{cpr} \rfloor$
2. Compute the number of thermometer rows with $(2^{n_{tot}-n_{bin}} - 1) \cdot 2^{n_{bin}} / n_{cpr}$
3. Place extra mirror rows (if any) in the center of the array.
4. Place binary rows (if any) above and below the mirror rows (or center if there are no extra rows) in an alternating fashion, from lowest to highest input weight.
5. Place thermometer rows in the same fashion as the binary rows. Start where the binary rows left off, where the extra mirror rows, or the center of the array, in that order of precedence.

6. Place the outer rings of dummy cells (if any).

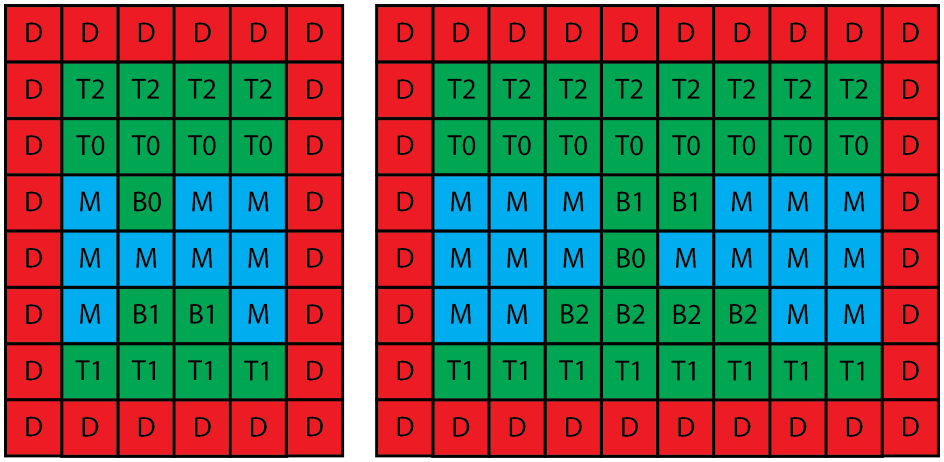


Figure 3.25: PI IDAC patterns with different number of bits and bias rows

Two sample array patterns generated using this algorithm are shown in Fig. 3.25. One extra caveat of this algorithm is that multiple rows may be connected to the same input depending on user supplied parameters. In order to reduce gradient effects on the array, the rows are split into two equally sized groups with one group each above and below the existing array. n_{cpr} is restricted to a power of 2 (excluding dummy cells) to ensure that each input occupies a number of rows that is a power of 2. This ensures that the number of rows is even (as long as there is more than 1 row) and can thus be split evenly into two groups. An example pattern with 6 total bits, 4 binary bits, and 8 columns is shown in Fig. 3.26. In this case, each thermometer weighted input requires 2 rows to implement the weighing. An example layout for an entire IDAC is shown in Fig. 3.27.

3.3.2 Phase Interpolator Integrator Generator for BAG 2.0

The gilbert-cell based PI requires inputs in the form of triangular waves, rather than digital square waves. Integrators are used to convert the square, digital LO signals into triangular waves. The integrator gain depends on the period of the input signal, and thus the input frequency, meaning that our gain will vary across operating frequency. In the extreme case, the output waveform may clip, no longer giving us a triangular wave. The dominant pole of the integrator must then be adjusted to allow for operation across a wide frequency range, which is done by modifying the bias current of the integrator. Common mode feedback (CMFB) is implemented to ensure proper biasing for the PMOS load resistors for a desired output common mode voltage. Both the bias current and CMFB output common mode voltage are controlled using the scan chain.

The PI integrator layout is split into three sub-blocks: the integrator core, the CMFB network, and the tail source IDAC. The CMFB network is further split into the common

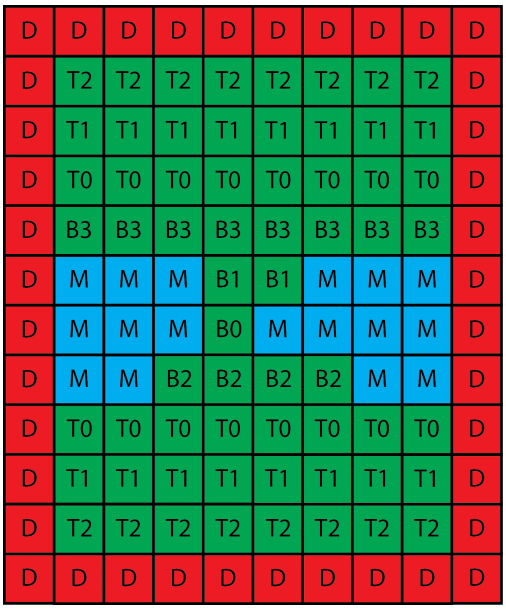


Figure 3.26: PI IDAC pattern with multi-row inputs

mode sense resistors and CMFB amplifier. The full extent of the hierarchy is shown in Fig. 3.28, where common templates like AnalogBase and SubstrateRing are omitted.

The bias DAC is implemented as a gate controlled IDAC made of normal, mirror, and dummy cells (Fig. 3.29). The NMOS implementation is shown here but an analogous PMOS version is also implemented in the BAG generator. Key parameters include a pattern array file, the unit cell parameters, number of outer dummy rings, and a flag for either an NMOS or PMOS DAC. Like the SCPA generator, this uses a pattern array file to place unit cells within the array. This choice was made to allow for very fine user control of cell placements, and with the expectation that this will be used for relatively small DACs on the order of 4 or 5 bits. This specific instance implemented a 4-bit thermometer DAC. Like the PI IDAC, the unit cell represents the smallest input weight, with higher weight binary and thermometer weighted inputs implemented by connecting multiple unit cells to the same input.

The mirror and dummy cells use nearly the same layout as the normal cell, but with different node connections as shown in Fig. 3.29. The purpose of connected BIAS to MID for the dummy cell is to reuse it as decap for the BIAS node. Like the PI IDAC, the channel length for each finger is the same. The MSRC device has a stack parameter to enable a longer effective channel length.

The PI integrator common mode sense resistors and CMFB amp are placed within a substrate guard ring. The CMFB amp is implemented as a single stage differential-to-single-ended amplifier with NMOS inputs. A high gain for this amplifier is not required due to extra loop gain coming from the integrator itself, particularly through the PMOS loads.

The resistor values of the sense resistors should be set to be large to minimize loading effects on the output of the CMFB amp. The resistors are implemented using the custom

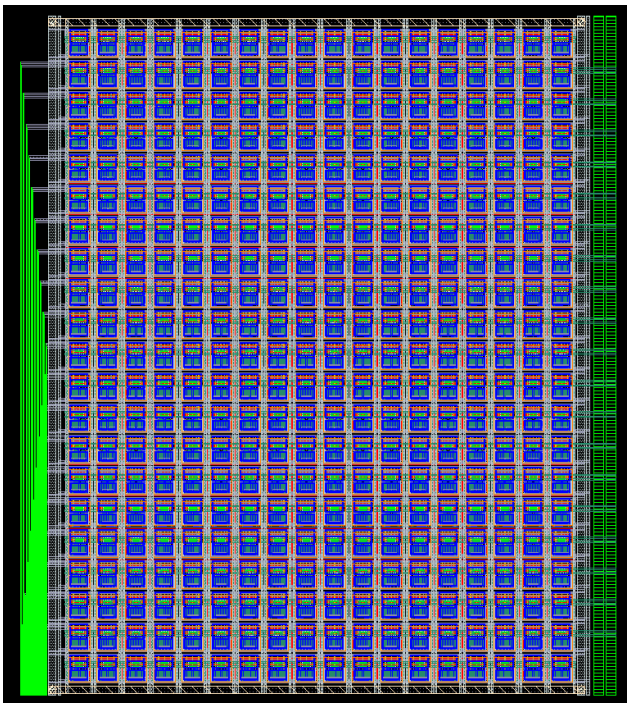


Figure 3.27: PI IDAC layout

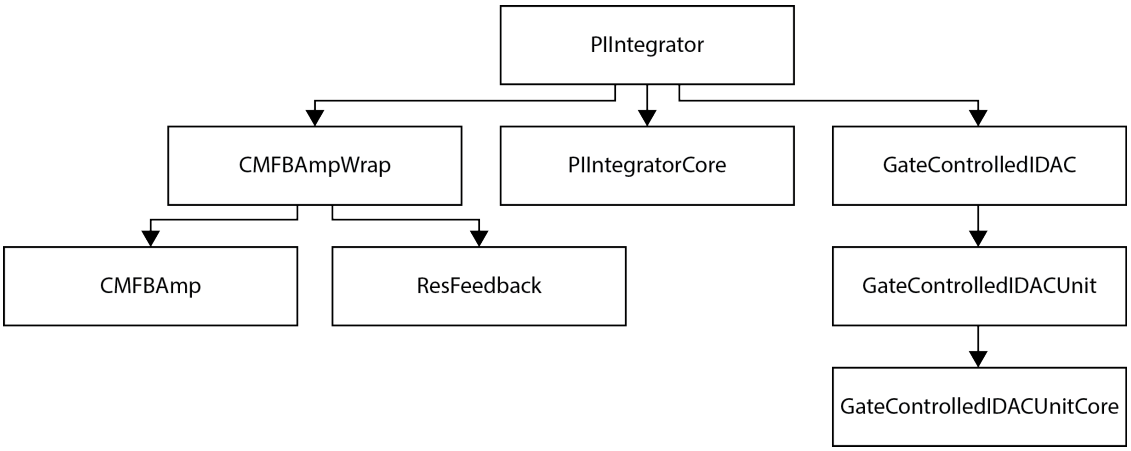
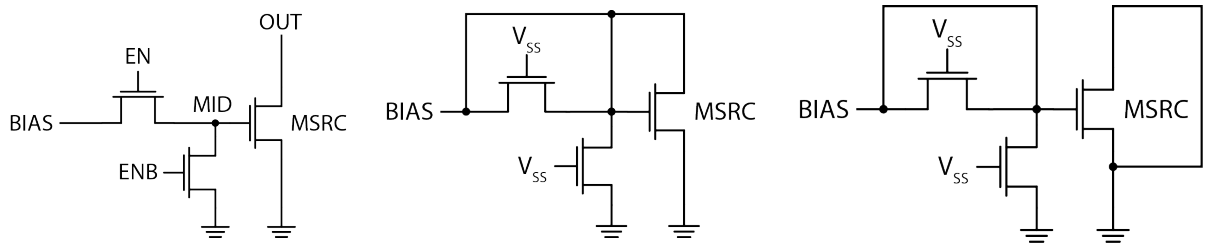


Figure 3.28: PI integrator layout template hierarchy

written ResFeedback class. This class inherits from the ResArrayBase class, which draws the resistor segments and and an outer ring of dummy resistors. These segments are drawn in a symmetric fashion to ensure good matching between resistor segments. The ResFeedback adds extra code to connects the resistor segments together in the desired fashion, and creates the ports which interface with other circuits.

Multiple resistor segments are tied in series to allow for better control of its aspect ratio.



(a) Normal cell (b) Mirror cell (c) Dummy cell

Figure 3.29: PI integrator bias IDAC unit cell types (NMOS)

Users can set the length and width of individual segments, as well as the total number of segments in series. All resistor segments have the same width and length. The segment width is chosen to be the minimum value to meet EM requirements. The sense current resistors will experience no DC current across a single period, meaning the EM requirements are very relaxed, and the miminum width for the process can be used to minimize area.

Finally, it should be noted that the CMFB loop was compensated using narrowbanding at the output of the CMFB amplifier. This was not implemented in the PI integrator generator, due to the required capacitor consisting of a fixed MOM + MOS capacitors. At the time of the design, MOS capacitors with the requisite amount of connections were not yet implemented in this technology. The capacitors were manually placed and connected to the PI integrator after generation, though the MOM cap portion is an instance of the metal finger capacitor used in the SCPA unit cell.

Chapter 4

Conclusion

4.1 Summary

This work presents two iterations of a fully integrated, frequency-flexible transmitter with programmable filtering to reduce unwanted spectral emissions in order to tackle the issue growing complexity and cost of radios within cellular handsets. The choice of topology was given and justified for the harmonic cancellation and mixed-signal filtering techniques presented. We demonstrated that these filtering techniques could be made frequency flexible as long as the core blocks generating the delayed and phase shifted versions of the data and LO signals also remain frequency flexible.

A deep analysis of the design considerations of the SCPA core as well as the overall TX was presented, considering traditional metrics like output power and efficiency, as well as metrics specifically important to our transmitter like HD3 reduction. Two versions of this TX were fabricated, including a revised version with more robust harmonic cancellation. Measurement results are shown for both TXs demonstrating the effectiveness of these cancellation techniques at cellular levels of output power, and showing that they can remain frequency flexible.

The usage of Berkeley Analog Generator (BAG) to implement circuit generators in both versions of the TX was discussed at length, primarily focusing on the layout generation. However, a couple of simple sizing algorithms were presented for components like current DACs and capacitors. These generators are discussed at multiple level of hierarchies, with robust floorplanning used to ensure manufacturable layouts even across a wide set of parameters. The circuit generation discussed two versions of the SCPA generator as well as a PI generator. Layout examples were also shown to demonstrate the flexibility of these generators.

4.2 Key Contributions

While this work covers various topics in detail, key contributions include:

- Designed a fully integrated transmitter in a TSMC 65 nm process outputting cellular handheld levels of output power. The transmitter implements programmable, integrated harmonic cancellation and mixed-signal filtering.
- Showed that the previously demonstrated harmonic cancellation and mixed-signal filtering techniques can coexist in a single transmitter.
- Presented analysis on how output asymmetry can degrade the filtering techniques depending on the output network. Provided analysis on why drain summing avoids these pitfalls and makes the techniques insensitive to the off-chip output network.
- Designed a second version of this transmitter in a TSMC 28 nm process, with improved harmonic cancellation, both in harmonic reduction numbers and robustness, verifying the analysis in the previous point.
- Demonstrated the effectiveness of the harmonic cancellation technique across at least a 1 GHz - 2 GHz frequency range, achieving a harmonic reduction of 24 - 42 dB for the first version and 35 - 57 dB for second version. Also demonstrated harmonic cancellation $> 29dB$ for 20 MHz modulated LTE data. Previous work has demonstrated this cancellation technique, but only for continuous wave (CW) measurements at a single frequency.
- Demonstrated that the usage of 25% duty cycle LO signals to implement IQ combining can introduce distortion in the constellation. Performed analysis showing that this is intrinsic to this technique when using nonideal switches, and that this distortion will fundamentally vary across code even with ideal switches.
- Proposed a technique to correct for the distortion by modifying the duty cycle, and validated this with measurements. The effectiveness of this technique is shown both using the newly defined metric of IQ / I as well as with measured constellations with different duty cycle settings.
- Further verified the effectiveness of design automation using the BAG framework for RF applications by generating key layout blocks using BAG. Previous works have demonstrated state of the art designs in the field of SERDES, but no previous work besides [18] targeted RF applications.
- Presented the methodology and algorithms used in writing custom generators for key blocks in the transmitters. Key blocks consisted of the SCPA in the first version, and the SCPA and phase interpolators (PI) in the second version.
- Presented a deep analysis of the design of the TX, with a focus on sizing the SCPA core. Proposed a sizing algorithm for optimizing for efficiency while meeting a HD3 reduction target using layout extracted schematics generated using BAG.

4.3 Future Work

Though this work has presented two prototypes with good results that have thoroughly demonstrated the effectiveness of these cancellation techniques, there are a myriad of possible options for future work.

- Implement 3^{rd} and 5^{th} harmonic cancellation simultaneously, and demonstrate it with modulated data.
- Explore the benefit of implement mixed-signal filtering with complex coefficients, and pursue this if it can be shown that significantly improved filters can be implemented.
- Improve the efficiency of this overall system, and determine if this is something due to poor design or intrinsic to the topology.
- Automate the transformer design and layout generation to allow for a design script able to size the key portions of the entire TX all within BAG, and not just the SCPA core.
- Develop a key metric for device linearity which maps to harmonic cancellation.
- Present a better solution to the variation of IQ / I across codes than sizing up cascode devices.
- Update the existing SCPA and PI generators to be more process portable, in preparation for a move to more advanced processes, such as those utilizing FinFETs.

Bibliography

[1] RFMW, *LTE Band Chart*, 2015 (accessed January 7, 2020). [Online]. Available: https://www.rfmw.com/data/rfmw_lte_band_chart.pdf

[2] Qorvo, *TQQ7301 Datasheet: Rev B*, Qorvo, 2 2016.

[3] Avago, *ACPF-7241 Bandpass Filter for Band 41 Product Brief*, Avago, 1 2016.

[4] D. Chowdhury, S. V. Thyagarajan, L. Ye, E. Alon, and A. M. Niknejad, "A Fully-Integrated Efficient CMOS Inverse Class-D Power Amplifier for Digital Polar Transmitters," *IEEE Journal of Solid-State Circuits*, vol. 47, no. 5, pp. 1113–1122, May 2012.

[5] D. Chowdhury, Lu Ye, E. Alon, and A. M. Niknejad, "A 2.4GHz mixed-signal polar power amplifier with low-power integrated filtering in 65nm CMOS," in *IEEE Custom Integrated Circuits Conference 2010*, Sep. 2010, pp. 1–4.

[6] L. Ye, J. Chen, L. Kong, P. Cathelin, E. Alon, and A. Niknejad, "A digitally modulated 2.4GHz WLAN transmitter with integrated phase path and dynamic load modulation in 65nm CMOS," in *2013 IEEE International Solid-State Circuits Conference Digest of Technical Papers*, Feb 2013, pp. 330–331.

[7] H. Jin, D. Kim, and B. Kim, "Efficient Digital Quadrature Transmitter Based on IQ Cell Sharing," *IEEE Journal of Solid-State Circuits*, vol. 52, no. 5, pp. 1345–1357, May 2017.

[8] H. Wang, C. Peng, Y. Chang, R. Z. Huang, C. Chang, X. Shih, C. Hsu, P. C. P. Liang, A. M. Niknejad, G. Chien, C. L. Tsai, and H. C. Hwang, "A Highly-Efficient Multi-Band Multi-Mode All-Digital Quadrature Transmitter," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 5, pp. 1321–1330, May 2014.

[9] S. Yoo, J. S. Walling, E. C. Woo, B. Jann, and D. J. Allstot, "A Switched-Capacitor RF Power Amplifier," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 12, pp. 2977–2987, Dec 2011.

[10] Y. Lee, A. Waterman, H. Cook, B. Zimmer, B. Keller, A. Puggelli, J. Kwak, R. Jevtic, S. Bailey, M. Blagojevic, P. Chiu, R. Avizienis, B. Richards, J. Bachrach, D. Patter-

son, E. Alon, B. Nikolic, and K. Asanovic, "An Agile Approach to Building RISC-V Microprocessors," *IEEE Micro*, vol. 36, no. 2, pp. 8–20, Mar 2016.

[11] G. Van der Plas, G. Debyser, F. Leyn, K. Lampaert, J. Vandenbussche, G. G. E. Gielen, W. Sansen, P. Veselinovic, and D. Leenarts, "AMGIE-A synthesis environment for CMOS analog integrated circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 20, no. 9, pp. 1037–1058, Sep. 2001.

[12] R. Harjani, R. A. Rutenbar, and L. R. Carley, "OASYS: a framework for analog circuit synthesis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 8, no. 12, pp. 1247–1266, Dec 1989.

[13] U. Choudhury and A. Sangiovanni-Vincentelli, "Automatic generation of parasitic constraints for performance-constrained physical design of analog circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 12, no. 2, pp. 208–224, Feb 1993.

[14] J. Crossley, A. Puggelli, H. . Le, B. Yang, R. Nancollas, K. Jung, L. Kong, N. Narevsky, Y. Lu, N. Sutardja, E. J. An, A. L. Sangiovanni-Vincentelli, and E. Alon, "BAG: A designer-oriented integrated framework for the development of AMS circuit generators," in *2013 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, Nov 2013, pp. 74–81.

[15] E. Chang, J. Han, W. Bae, Z. Wang, N. Narevsky, B. Nikolic, and E. Alon, "BAG2: A process-portable framework for generator-based AMS circuit design," in *2018 IEEE Custom Integrated Circuits Conference (CICC)*, April 2018, pp. 1–8.

[16] J. Han, Y. Lu, N. Sutardja, K. Jung, and E. Alon, "Design Techniques for a 60 Gb/s 173 mW Wireline Receiver Frontend in 65 nm CMOS Technology," *IEEE Journal of Solid-State Circuits*, vol. 51, no. 4, pp. 871–880, April 2016.

[17] J. Han, E. Chang, S. Bailey, Z. Wang, W. Bae, A. Wang, N. Narevsky, A. Whitcombe, P. Lu, B. Nikoli, and E. Alon, "A Generated 7GS/s 8b Time-Interleaved SAR ADC with 38.2dB SNDR at Nyquist in 16nm CMOS FinFET," in *2019 IEEE Custom Integrated Circuits Conference (CICC)*, April 2019, pp. 1–4.

[18] N. Kuo, B. Yang, C. Wu, L. Kong, A. Wang, M. Reiha, E. Alon, A. M. Niknejad, and B. Nikolic, "A frequency-reconfigurable multi-standard 65nm CMOS digital transmitter with LTCC interposers," in *2014 IEEE Asian Solid-State Circuits Conference (A-SSCC)*, Nov 2014, pp. 345–348.

[19] T. Georgantas, K. Vavelidis, N. Haralabidis, S. Bouras, I. Vassiliou, C. Kapnistis, Y. Kokolakis, H. Peyravi, G. Theodoratos, K. Vryssas, N. Kanakaris, C. Kokozidis, S. Kavadias, S. Plevridis, P. Mudge, I. Elgorriaga, A. Kyranas, S. Liolis, E. Kytonaki, G. Konstantopoulos, P. Robogiannakis, K. Tsilipanos, M. Margaras, P. Betzios,

R. Magoon, N. Bouras, M. Rofougaran, and R. Rofougaran, "9.1 A 13mm2 40nm multiband GSM/EDGE/HSPA+/TDSCDMA/LTE transceiver," in *2015 IEEE International Solid-State Circuits Conference - (ISSCC) Digest of Technical Papers*, Feb 2015, pp. 1–3.

[20] M. Fulde, A. Belitzer, Z. Boos, M. Bruennert, J. Fritzin, H. Geltinger, M. Groinig, D. Gruber, S. Gruenberger, T. Hartig, V. Kampus, B. Kapfelsberger, F. Kuttner, S. Leuschner, T. Maletz, A. Menkhoff, J. Moreira, A. Paussa, D. Ponton, H. Pretl, D. Sira, U. Steinacker, and N. Stevanovic, "13.2 A digital multimode polar transmitter supporting 40MHz LTE Carrier Aggregation in 28nm CMOS," in *2017 IEEE International Solid-State Circuits Conference (ISSCC)*, Feb 2017, pp. 218–219.

[21] J. Moreira, S. Leuschner, N. Stevanovic, H. Pretl, P. Pfann, R. Thringer, M. Kastner, C. Prll, A. Schwarz, F. Mrugalla, J. Saporiti, U. Basaran, A. Langer, T. D. Werth, T. Gossmann, B. Kapfelsperger, and J. Pletzer, "9.2 A single-chip HSPA transceiver with fully integrated 3G CMOS power amplifiers," in *2015 IEEE International Solid-State Circuits Conference - (ISSCC) Digest of Technical Papers*, Feb 2015, pp. 1–3.

[22] S. Kang, U. Kim, and J. Kim, "A Multi-Mode Multi-Band Reconfigurable Power Amplifier for 2G/3G/4G Handset Applications," *IEEE Microwave and Wireless Components Letters*, vol. 25, no. 1, pp. 49–51, Jan 2015.

[23] J. Lin, Y. Zheng, Z. Zhang, and G. Zhang, "A multi-mode multi-band power amplifier for quad-band GSM, dual-band TD-SCDMA, and TDD LTE band 39 cellular applications," in *2015 Asia-Pacific Microwave Conference (APMC)*, vol. 1, Dec 2015, pp. 1–3.

[24] M. Tsai, C. Lin, P. Chen, T. Chang, C. Tseng, L. Lin, C. Beale, B. Tseng, B. Tenbroek, C. Chiu, G. Dehng, and G. Chien, "13.1 A fully integrated multimode front-end module for GSM/EDGE/TD-SCDMA/TD-LTE applications using a Class-F CMOS power amplifier," in *2017 IEEE International Solid-State Circuits Conference (ISSCC)*, Feb 2017, pp. 216–217.

[25] C. Lee, J. J. Chang, K. S. Yang, K. H. An, I. Lee, K. Kim, Joongjin Nam, Y. Kim, and H. Kim, "A highly efficient GSM/GPRS quad-band CMOS PA Module," in *2009 IEEE Radio Frequency Integrated Circuits Symposium*, June 2009, pp. 229–232.

[26] S. He, F. Peng, L. Xu, H. Meng, and Y. Qian, "A Compact High Efficiency and High Power Front-end Module for GSM/EDGE/TD-SCDMA/TD-LTE Applications in 0.13um CMOS," in *2018 IEEE Asian Solid-State Circuits Conference (A-SSCC)*, Nov 2018, pp. 289–292.

[27] A. F. Aref and R. Negra, "A Fully Integrated Adaptive Multiband Multimode Switching-Mode CMOS Power Amplifier," *IEEE Transactions on Microwave Theory and Techniques*, vol. 60, no. 8, pp. 2549–2561, Aug 2012.

[28] A. Ba, V. K. Chillara, Y. Liu, H. Kato, K. Philips, and R. B. Staszewski, "A 2.4GHz class-D power amplifier with conduction angle calibration for 50dBc harmonic emissions," in *2014 IEEE Radio Frequency Integrated Circuits Symposium*, June 2014, pp. 239–242.

[29] M. Mizokami, T. Uozumi, Y. Yamashita, K. Shibata, and H. Sato, "A 43%-efficiency 20dBm sub-GHz transmitter employing rise-edge-synchronized harmonic calibration with 33.3% duty cycle," in *2017 Symposium on VLSI Circuits*, June 2017, pp. C304–C305.

[30] C. Huang, Y. Chen, T. Zhang, V. Sathe, and J. C. Rudell, "A 40nm CMOS single-ended switch-capacitor harmonic-rejection power amplifier for ZigBee applications," in *2016 IEEE Radio Frequency Integrated Circuits Symposium (RFIC)*, May 2016, pp. 214–217.

[31] R. Bhat and H. Krishnaswamy, "A watt-level 2.4 GHz RF I/Q power DAC transmitter with integrated mixed-domain FIR filtering of quantization noise in 65 nm CMOS," in *2014 IEEE Radio Frequency Integrated Circuits Symposium*, June 2014, pp. 413–416.

[32] R. Bhat, J. Zhou, and H. Krishnaswamy, "Wideband Mixed-Domain Multi-Tap Finite-Impulse Response Filtering of Out-of-Band Noise Floor in Watt-Class Digital Transmitters," *IEEE Journal of Solid-State Circuits*, vol. 52, no. 12, pp. 3405–3420, Dec 2017.

[33] J. A. Weldon, J. C. Rudell, Li Lin, R. Sekhar Narayanaswami, M. Otsuka, S. Dedieu, Luns Tee, King-Chun Tsai, Cheol-Woong Lee, and P. R. Gray, "A 1.75 GHz highly-integrated narrow-band CMOS transmitter with harmonic-rejection mixers," in *2001 IEEE International Solid-State Circuits Conference. Digest of Technical Papers. ISSCC (Cat. No.01CH37177)*, Feb 2001, pp. 160–161.

[34] C. Andrews and A. C. Molnar, "A Passive Mixer-First Receiver With Digitally Controlled and Widely Tunable RF Interface," *IEEE Journal of Solid-State Circuits*, vol. 45, no. 12, pp. 2696–2708, Dec 2010.

[35] G. H. Li and H. P. Chou, "A high resolution time-to-digital converter using two-level vernier delay line technique," in *2007 IEEE Nuclear Science Symposium Conference Record*, vol. 1, Oct 2007, pp. 276–280.

[36] H. Huang and C. Sechen, "A 14-b, 0.1ps resolution coarse-fine time-to-digital converter in 45 nm CMOS," in *2014 IEEE Dallas Circuits and Systems Conference (DCAS)*, Oct 2014, pp. 1–4.

[37] N. Kuo, B. Yang, A. Wang, L. Kong, C. Wu, V. P. Srini, E. Alon, B. Nikoli, and A. M. Niknejad, "A 0.4-to-4-GHz All-Digital RF Transmitter Package With a Band-Selecting Interposer Combining Three Wideband CMOS Transmitters," *IEEE Transactions on Microwave Theory and Techniques*, vol. 66, no. 11, pp. 4967–4984, Nov 2018.

[38] M. J. M. Pelgrom, A. C. J. Duinmaijer, and A. P. G. Welbers, "Matching properties of MOS transistors," *IEEE Journal of Solid-State Circuits*, vol. 24, no. 5, pp. 1433–1439, Oct 1989.

[39] L. Calderin, S. Ramakrishnan, A. Puglielli, E. Alon, B. Nikoli, and A. M. Niknejad, "Analysis and Design of Integrated Active Cancellation Transceiver for Frequency Division Duplex Systems," *IEEE Journal of Solid-State Circuits*, vol. 52, no. 8, pp. 2038–2054, Aug 2017.

[40] M. Kimura, "Field and temperature acceleration model for time-dependent dielectric breakdown," *IEEE Transactions on Electron Devices*, vol. 46, no. 1, pp. 220–229, Jan 1999.

[41] S. Pornpromlikit, J. Jeong, C. D. Presti, A. Scuderi, and P. M. Asbeck, "A Watt-Level Stacked-FET Linear Power Amplifier in Silicon-on-Insulator CMOS," *IEEE Transactions on Microwave Theory and Techniques*, vol. 58, no. 1, pp. 57–64, Jan 2010.

[42] J. G. McRory, G. G. Rabjohn, and R. H. Johnston, "Transformer coupled stacked FET power amplifiers," *IEEE Journal of Solid-State Circuits*, vol. 34, no. 2, pp. 157–161, Feb 1999.

[43] N. Kuo, B. Yang, A. Wang, L. Kong, C. Wu, V. P. Srini, E. Alon, B. Nikoli, and A. M. Niknejad, "A Wideband All-Digital CMOS RF Transmitter on HDI Interposers With High Power and Efficiency," *IEEE Transactions on Microwave Theory and Techniques*, vol. 65, no. 11, pp. 4724–4743, Nov 2017.

[44] Integrand Software, Inc., *EMX Users Manual*, Integrant Software, Inc., 2010.

[45] Zhuqing Zhang and C. P. Wong, "Recent advances in flip-chip underfill: materials, process, and reliability," *IEEE Transactions on Advanced Packaging*, vol. 27, no. 3, pp. 515–524, Aug 2004.

[46] Xilinx, Inc., *VC707 Evaluation Board for the Virtex-7 FPGA*, Xilinx, Inc., 2019.

[47] Xilinx, Inc., *7 Series FPGAs GTX/GTH Transceivers*, Xilinx, Inc., 2018.

[48] Ansoft Corporation, *user's guide - High Frequency Structure Simulator*, Ansoft Corporation, 2003.

[49] Synopsys, *PyCell Studio Tutorial*, Synopsys, 2016.

[50] O. Ben-Kiki and C. Evans and I. dot Net, *YAML Aint Markup Language (YAML) Version 1.2*, 2009 (accessed January 2, 2020). [Online]. Available: https://yaml.org/spec/1.2/spec.html