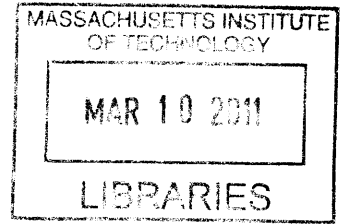# Key Receiver Circuits for Digital Beamforming in Millimeter-wave Imaging

by

Khoa Minh Nguyen

B.S., University of California, Berkeley (2004)
S.M., Massachusetts Institute of Technology (2006)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2011

Author . . . . .
Department of Electrical Engineering and Computer Science
December 10, 2010

Certified by.
Charles G. Sodini
LeBel Professor of Electrical Engineering
Thesis Supervisor

Accepted by . . . . .
Terry P. Orlando
Chairman, Department Committee on Graduate Theses

Key Receiver Circuits for Digital Beamforming in Millimeter-wave Imaging

by

Khoa Minh Nguyen

Abstract

Millimeter-wave (MMW) frequencies have wavelengths small enough to offer sufficient spatial resolution for certain imaging applications. Advances in silicon processes have developed devices that can operate at these frequencies, which has led to the potential for low cost MMW imaging, provided that the circuit design can meet the performance specifications for these applications. In this research, we investigate key components for an active phased array imaging system operating at 77GHz that performs purely digital beamforming. Each element in the phased array has an antenna and processor that measures the phase and amplitude of the received signal. The focus of this thesis is the quality of the phase measurement. Phase noise from the receiver's front-end circuits will degrade the spatial resolution and image integrity.

The system requires a MMW phase-locked loop (PLL) to generate the local oscillator. The PLL is a significant contributor to phase noise. A MMW PLL was designed in a 0.13μm silicon-germanium BiCMOS technology for low phase noise and power consumption while maintaining enough output power to robustly drive a mixer load. Measurement results show a de-embedded single-ended output power of -2dBm, a phase noise of -81dBc/Hz at 1MHz offset corresponding to 1ps of timing jitter at the carrier, and a total power dissipation of 107mW.

A new technique called digital phase tightening reduces phase noise from receiver front-end circuits to allow precise phase estimation for digital beamforming. This technique leverages the large ratio between the MMW carrier frequency and the low frame rates in imaging applications. By mixing down to an intermediate frequency (IF) and then averaging over many samples, we reduce phase error caused by phase noise. A test chip demonstrating the phase tightening concept was designed and characterized. We show that we can reduce RMS error from 450ps to 1.4ps at a 175MHz IF which corresponds to reducing 1ps of jitter to 3fs at a 77GHz carrier.

Thesis Supervisor: Charles G. Sodini
Title: LeBel Professor of Electrical Engineering

Acknowledgments

The past six years has been a great learning experience for me, and I could not have done it without the help of so many others. I would first like to thank my advisor, Professor Charles Sodini, for giving me the opportunity to work on the millimeter-wave imaging project. I appreciated his guidance and intuition throughout this work, and he always provided new perspectives for approaching problems.

I would also like to thank Helen Kim of MIT Lincoln Laboratory for her insight in millimeter-wave imaging technologies and design.

I want to thanks the members of the Millimeter-wave Imaging group including Johnna, Anthony, Ke, and Jack. Without their collaboration, my work would never have been possible.

The members of the Sodini, Lee, and Dawson research groups also deserve acknowledgement for providing an enjoyable and motivating work environment.

I want to acknowledge the support of the C2S2 Focus Center, one of six research centers funded under the Focus Center Research Program (FCRP), a Semiconductor Research Corporation subsidiary. I also wish to acknowledge the Massachusetts Institute of Technology Center for Integrated Circuits and Systems for funding.

Finally, I would like to thank my parents for all of their support throughout the past six years and providing me with the discipline to make it this far. And I would also like to thank my little sister for always being there to cheer me on.

Thanks everyone,

Khoa

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Imaging is a process in which one captures information from a scene to create a visual representation of it. There are various signals that can be captured from a scene and different ways to process this information into an image. The most common example is the conventional camera. It is an optical imaging system, in which a lens focuses visible light from a scene onto photographic film which reacts to the light and records an image.

The conventional camera creates an image by manipulating visible electromagnetic radiation, but imaging can be broadened to signals outside of the visible spectrum. Familiar applications include the use of X-rays for medical imaging within the human body and infrared for use in thermal imaging. A spectrum of particular interest for imaging is the millimeter-wave (MMW) spectrum. Imaging at these frequencies has been studied and used in certain applications. However, these systems are expensive to implement and limited in capability. It is not until recently that advances in silicon technologies have produced devices that will allow cost-effective and highly-integrated solutions that open up enormous possibilities for MMW imaging.

This thesis focuses on imaging with MMW signals. It will discuss different methods for imaging with MMW signals and delve into the circuit implementation for one particular approach. Chapter 2 will discuss imaging with MMW frequencies. Properties of this spectrum and previous methods and systems that operated at these frequencies will be described. With advances in technology, phased arrays at MMW

frequencies are possible. Beamforming and a study of the MMW imaging system of interest will be discussed in Chapter 3. We determine that a MMW phase-locked loop (PLL) will be a key component of the imaging receiver. Chapter 4 will describe the design and characterization of a MMW PLL. In order to use beamforming for imaging, we require accurate phase measurements in our system. Chapter 5 introduces and discusses the implementation of a phase estimation system called digital phase tightening which is used to measure phase and reduce error caused by phase noise. Concluding remarks will be made in Chapter 6.

Chapter 2

Millimeter-wave Imaging

Millimeter-wave (MMW) frequencies have wavelengths that range from 1mm to 10mm which correspond to frequencies from 30GHz to 300GHz. MMW frequencies, as well as terahertz frequencies, lie at a technological crossing point. Below them are microwave frequencies where coherent signal processing has been employed. Above them are infrared and visible frequencies where imaging is typically done through incoherent processing known as radiometry. MMW imaging has been demonstrated in the past and was done mostly through radiometry [1] [2]. However, the improvement of semiconductor technology has allowed coherent processing to be possible at MMW frequencies. [3]. As a result, there has recently been a surge in interest for MMW imaging. This chapter will discuss properties of MMW signals and how they have been used for imaging applications.

2.1   Properties of Millimeter-waves

MMW frequencies have certain characteristics that are amenable to imaging. With relatively large wavelengths on the order of millimeters, MMW signals have the ability to pass through inclement weather conditions such as rain, smoke, and dust [4]. They can also be used to image through some thin opaque items such as clothing. This capability means that MMW signals can be useful in low-visibility applications that would not be possible with visible or infrared wavelengths.

19
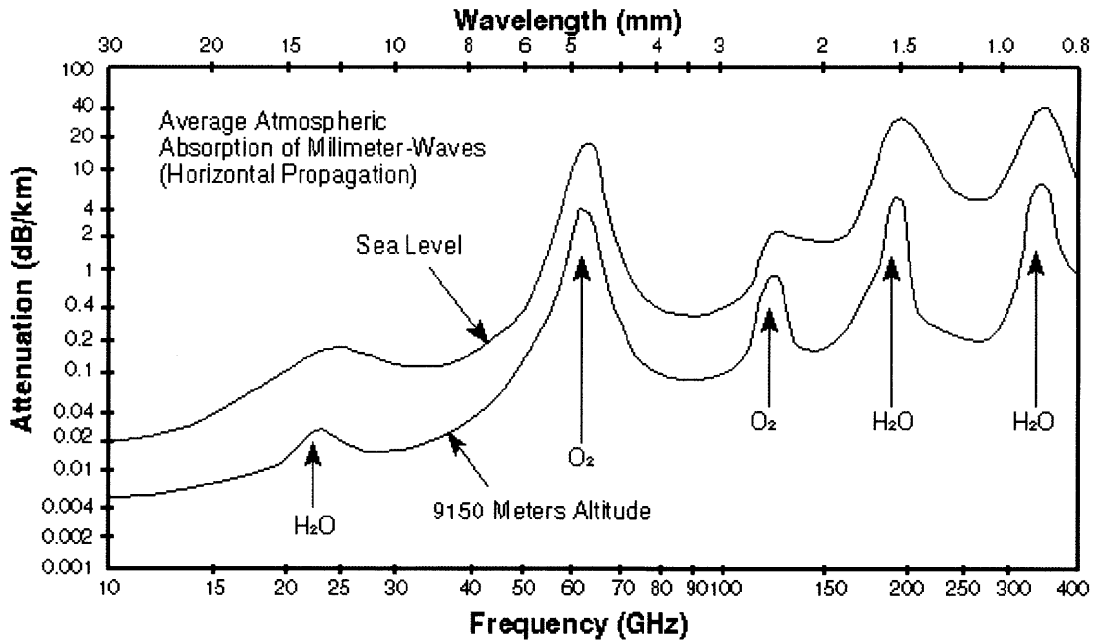
Figure 2-1: Atmospheric attenuation versus frequency.

Furthermore, there are bands within the MMW frequency spectrum that experience high attenuation due to O_2 and H_2O absorption in the earth's atmosphere. Between these bands are windows in which the attenuation is low. These windows are suitable for long-range applications which require signals to propagate longer distances while maintaining sufficient signal-to-noise ratio. As can be seen in Figure 2-1 from [5], these windows occur around 94GHz, 140GHz, and 220GHz. Imaging applications tend to operate in these windows, so the system can image over longer ranges.

Spatial resolution for imaging is proportional to the operating wavelength [6]. Although MMW wavelengths are much larger than optical wavelengths, they are still small enough that they can provide suitable spatial resolution for many imaging applications. Furthermore, compared to microwaves, MMW frequencies require smaller antennas and can be assembled more compactly. This allows for higher resolution for a given aperture size.

Ionization is the process of turning a molecule into an ion by adding or removing

20

a charged particle like an electron. High frequency radiation, such as X-rays, have enough energy to ionize molecules in the human body, which can damage cells and potentially cause illness. Lower energy MMW frequencies, however, have the property of being non-ionizing and therefore pose no known adverse health risks when exposed to humans. This fact makes them suitable for imaging applications that involve people since the radiation does not harm the subjects being imaged [7] [8].

2.1.1 Radiometric Properties

Since we are creating images by measuring MMW radiation, it is important to understand the radiometric properties of the materials being imaged. Radiometric properties describe the behavior of electromagnetic radiation that is incident upon the surface of the material. The radiation will be reflected, absorbed, and transmitted by the material. The distribution of the energy between these three behaviors depends on the material. We describe the reflectivity of a material by assigning a value to its reflectance ρ. Similarly, a material will have an absorptance α and transmittance τ. Since energy is conserved, the sum of these three parameters must equal one.

Any material above a temperature of 0K will radiate energy. In steady-state, an object's temperature is not changing so the the amount of energy it absorbs is equal to the amount it emits. Therefore, the emittance of a material ϵ is equal to the absorptance α. For any material that is not translucent, the transmittance is zero, so the emittance and reflectance are directly complementary to each other. Emittance and reflectance are key properties since they determine how much radiation will be observed from an object. Materials with different levels of emittance and reflectance will have a measurable difference in radiation that will allow them to be distinguished in an image [9].

A blackbody is an idealized material that absorbs all electromagnetic radiation, and therefore has an emissivity of one. The amount of radiance by a blackbody as a function of wavelength and temperature is described by Planck's law, which is shown in Equation 2.1 as a function of frequency ν.

$$L_{\nu,bb} = \frac{2h\nu^3}{c^2} \frac{1}{e^{\frac{h\nu}{kT}} - 1} \qquad (2.1)$$

L_ν is the emitted power per unit area of emitting surface, per unit solid angle, and per unit frequency. It has units of $W/(m^2 \cdot sr \cdot m)$. ν is the frequency, h is Planck's constant, k is Boltzmann's constant, c is the speed of light, and T is the temperature. If we fix the frequency and sweep the temperature, the exponential in the denominator decreases with increasing temperature, so the radiation L_ν increases with increasing temperature.

For real objects which are non-blackbody, emissivity must be considered. The emissivity represents the ratio of the emitted power per area for a material versus the emitter power per area of a blackbody object. The radiation from a material with emittance ϵ_x is the blackbody radiation from Planck's law scaled by ϵ_x as shown in Equation 2.2. Therefore, the radiation emitted from a material is dependent on its temperature and its emittance.

$$L_{\nu,x} = \epsilon_x \cdot L_{\nu,bb} \qquad (2.2)$$

2.2 Active and Passive Imaging

Millimeter-wave imaging can be divided into two types: passive imaging and active imaging. The main difference between the two is that active imagers use a radiating source to illuminate a scene while passive imagers do not. Passive imagers rely on the emitted radiation from the objects in the scene based on their temperature and also reflections from the environment. As described by Planck's law, the amount of energy emitted by an object depends on its temperature and its emissivity. Metallic objects have very low emissivity and will appear cool compared to their environment. The human body has a fairly high emissivity, approximately 65% at 100GHz, which is much higher than the emissivity of metals and, to a lesser extent, plastics and ceramics. A person will therefore appear warmer than his surroundings, and likewise, low emissivity objects carried on a person can be readily distinguished. The passive

imager primarily measures apparent temperature differences between objects in the scene [7].

An active imager uses a radiating source to illuminate a scene and then measures the reflected energy to produce an image. This is analogous to a camera that captures the light from a scene illuminated by a camera flash. Active imaging relies on the reflectivity of material rather than emissivity.

There are certain key differences between the two types of imaging. Active imagers are narrow-band systems that transmit and receive at a particular frequency. Passive imagers have no radiating source, so the received energy will be lower than in their active counterparts. These systems will generally have wider bandwidths to capture as much energy as possible. Temperature sensitivity is another issue. Passive imaging is limited in the level of temperature contrast it can detect. While outdoor environments are illuminated naturally and can have significant temperature differences, indoor environments tend to have a smaller temperature range. Active imaging is more effective than passive imaging for indoor applications where temperature differences are small and an additional source is required to illuminate the scene or object [10]. However, active imaging can still be useful for outdoor applications given the increase in received power.

Active imagers can have large dynamic ranges, especially when the illuminating radiation is sourced from a single angle, which is known as single-mode illumination. Because millimeter-waves are larger than visible wavelengths, there is more specular reflection than diffusive reflection. So surfaces that are not angled properly between the radiation source and receiving array will reflect very little back to the imaging array. The problem is a lack of angular diversity that is needed to better illuminate the scene so more signal can be received. This problem can be alleviated by using a flood light rather than focused illumination. By flooding the scene with radiation, we illuminate from multiple directions which increases the angular diversity, at the cost of increased power dissipation. This is an issue only for active imagers since passive imaging uses native radiation from the scene, which is inherently more diverse [11].

23

2.3 Applications

Applications for MMW imaging fall into several categories. The first is outdoor vehicular guidance which takes advantage of the ability to image in all-weather conditions. Aircraft landing guidance systems have been demonstrated operating at the 94GHz atmospheric window [12]. These systems image aircraft runways and detect any obstructions that may exist on the runway. They are often implemented as passive systems that take images based on the natural temperature contrast of outdoor environments [3].

Contraband detection is another prevalent use of MMW imaging. Security scanners have been employed at airports to detect concealed weapons. Operating at 94GHz, they are capable of observing objects hidden by clothing. Active imagers [10] [13] and passive imagers [14] have been demonstrated for these purposes. Active MMW imaging also has potential in detecting land mines. Land mines can be constructed from a variety of materials, and although metallic land mines can be detected using a number of methods, plastic mines are more difficult to detect. With MMW imaging, one can distinguish size and shape of different materials including metals and plastics in the ground, which makes it promising for this application. MMW transmission through the soil depends on the size of the soil particles since large particles compared to the wavelength will cause scattering [15].

Automotive radar systems are commercially available at 77GHz. These are active systems whose purpose is to accurately measure speed and distance of objects or vehicles in the lanes ahead even in low-visibility conditions due to inclement weather. They are used to aid drivers with adaptive cruise control and active brake assist for collision mitigation. MMW imaging is currently being explored to operate in conjunction with the range finding capabilities of these radar systems [16] [17] [18].
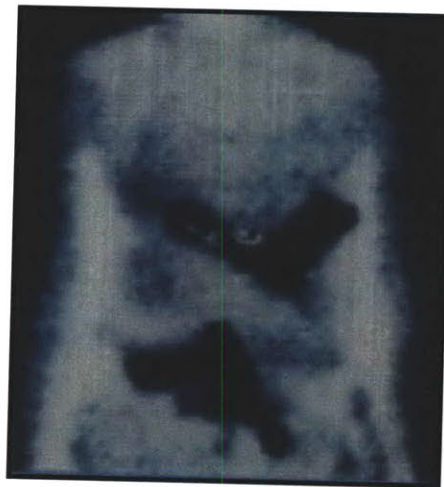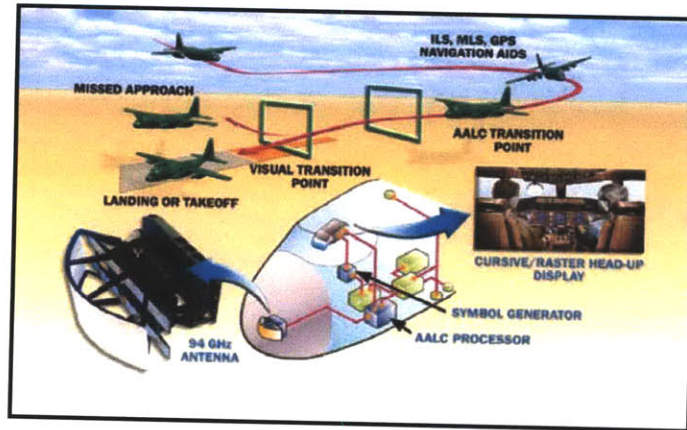
Figure 2-2: Current millimeter-wave imaging applications including aircraft guidance landing systems, concealed weapons detection, and automotive radar.

2.4 Prior System Architectures for Millimeter-wave Imaging

Early MMW imaging systems were passive imagers that used radiometers to measure the radiation of the target scene. There are different types of radiometers, but in its simplest form, a radiometer consists of an antenna, a detector, and an integrator to measure the power of the incoming radiation. A low noise amplifier (LNA) is often required to amplify the received signal. Due to the cost and difficulty of implementing MMW devices, these systems would consist of a single radiometer. The scene was scanned by mechanical steering, which means that the receiver is mechanically adjusted to focus on a certain pixel at one point in time as shown in Figure 2-3. The drawback of this approach is that it is slow and the moving parts bring reliability concerns [19] [20].

Scene

Mechanically-steered single receiver
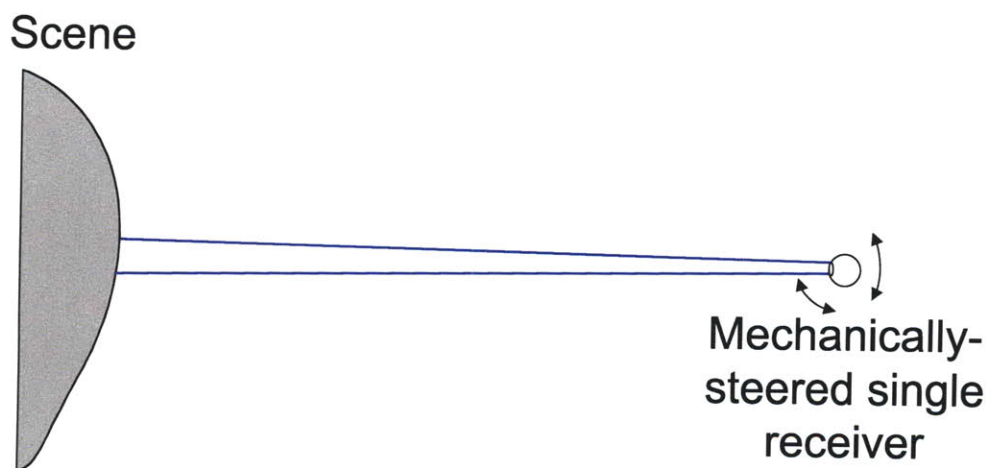
Figure 2-3: Diagram of mechanically-steered imaging system.

In [3], Goldsmith et al. discuss MMW focal plane imaging systems. Unlike the previous systems which utilized a single receiver that was mechanically steered, these imagers consisted of a lens that focuses radiation from the scene onto a planar array of radiometers which measures the received energy as shown in Figure 2-4. Each of the elements in the array represents a single pixel. The advancements in technology allowed for individual radiometers to be created cost-effectively to form these planar

arrays, which were prohibitively expensive before.
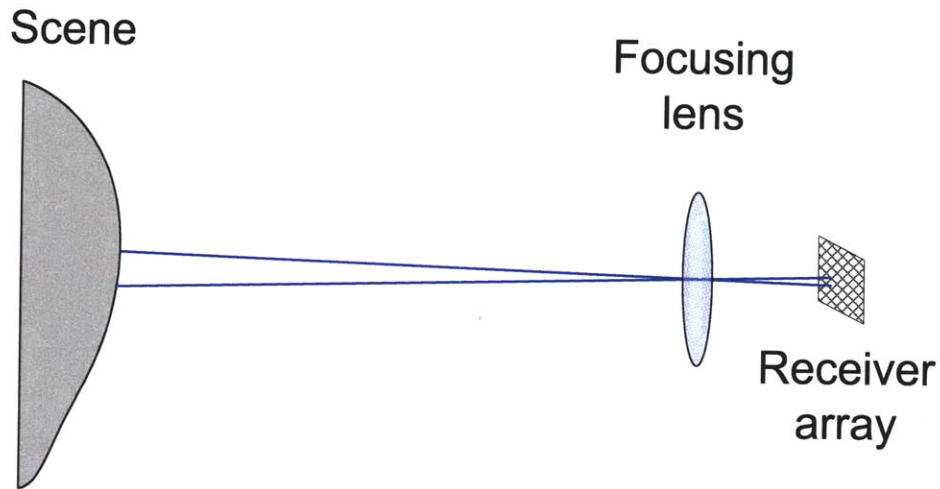
Figure 2-4: Diagram of focal plane array imaging system.

These techniques used radiometry to measure the MMW radiation propagating from the scene. This is essentially incoherent imaging which relies only on the amplitude information of the received signal. However, a trend has been moving towards coherent imaging which utilizes both the amplitude and phase information. Coherent processing has been prevalent in microwave applications and has only recently become possible at MMW frequencies [3].

The push into coherent processing is due largely to advances in III-V, and more recently silicon, technologies which have developed devices that can operate at MMW frequencies. Silicon has become relevant with ft's and fmax's that are now pushing in the hundreds of gigahertz range, which has allowed for higher integration and more cost-effective solutions for MMW systems. Devices are now fast enough to not only manipulate the amplitude, but also the phase of MMW signals. Electronic steering using array beamforming is now feasible. Beamforming will be discussed further in Chapter 3.

2.5 Published MMW Imaging System Implementations

2.5.1 Systems Using III-V Technologies

Due to the high frequency operation of MMW imaging, technologies such as Gallium Arsenide (GaAs) and Indium Phosphide (InP) semiconductors have traditionally been used for these applications. These semiconductors have higher electron mobility compared to similarly doped silicon. The higher mobility results in higher transit frequencies, usually in the hundreds of gigahertz even without the aggressive scaling of silicon CMOS, and therefore makes them suitable for MMW operation [21]. This section will describe MMW imaging systems that have been implemented in these III-V technologies.

In 1993, Raffaelli et al. describe a monolithic transmitter operating at 77GHz using $0.25\mu m$ GaAs MESFETs. The transmitter was intended for automotive collision avoidance radar. The architecture of the design was a single 38GHz VCO which was amplified and then frequency multiplied by two. A phase noise of -90dBc/Hz at 1MHz offset was achieved and the output power was greater than 10dBm over the tuning range. The small size of the monolithic design was a cost effective solution at the time that made MMW collision avoidance systems more affordable [22].

Shoucri et al. published a passive MMW camera for low-visibility aircraft landing assistance operating at 94GHz. The system was composed of a 40x26 receiver focal plane array with an 18" aperture. The receivers were implemented in GaAs MMW monolithic integrated circuits to perform the low noise amplification and measure the MMW radiation. The receivers had a 10GHz bandwidth and achieved a noise figure of 5.5dB [1] [23].

In 2003, a passive MMW camera operating from 75 to 93GHz was presented for security applications. The camera consisted of an array of 232 elements. Each element had an LNA that was implemented using InP and achieved a noise figure of 4dB. The camera had an angular resolution of 0.35° and it achieved a temperature resolution

of 2.1° [14].

An active MMW radar system was used for imaging concealed weapons. The system used a GaAs Gunn oscillator which allowed operation of 94GHz. The system obtained a range resolution of 3 inches. The antenna had a beamwidth of 1.7° and was mechanically scanned to achieve ±4.5° [24].

2.5.2 Silicon-based Imaging Systems

III-V technologies have proven to be viable solutions for MMW imaging systems. However, they are expensive to implement and lack the level of integration of silicon IC. More recently, imaging systems have been implemented in silicon-based technologies.

Tomkins et al. published a passive W-band imager in 65nm CMOS in 2009 using a direct-detection architecture. The system was composed of a single pole, double throw (SPDT) switch between a RF input and a 50Ω reference resistor which toggled between the two to cancel low frequency fluctuations. The output of the switch led to an LNA, which was implemented as a cascade of five cascode stages. The output of the LNA went to a square-law power detector. The receiver has a peak gain of 27dB with a 3dB bandwidth of 8GHz. The noise figure was measured to be 6.8dB [25].

Another passive imaging receiver operating at 94GHz was published using $0.18\mu m$ SiGe BiCMOS technology. It also has a direct-detection architecture. A balanced LNA was used with an embedded Dicke switch. The LNA was followed by a square-law power detector, implemented with a common-emitter heterojunction bipolar transistor, which converts the output to a DC voltage that is proportional to the input signal's power level. The receiver achieved a 3dB bandwidth of 26GHz with a pre-detection gain of 30dB and a noise figure of 12dB [26].

2.6   Summary

MMW frequencies have properties that make them suitable for certain imaging applications. In this chapter, we described how images can be formed from MMW

radiation. We explained the differences between passive and active imagers and then reviewed previous implementations of these systems. We found that MMW imaging systems began as radiometric systems, and now coherent imaging systems are possible using phased arrays in silicon.

Chapter 3

Beamforming

Improvements in modern silicon technologies have led to a push in cost-effective phased arrays operating at MMW frequencies [27]. In a phased-array, we can apply weights to signals from individual sensors such that the overall array amplifies signals from a target direction while suppresses signals from other directions. This process is called beamforming and allows us to electronically steer an array.

3.1   Delay-and-Sum Beamformer

The simplest beamformer is the conventional delay-and-sum beamformer. In order to understand its function, we consider a one-dimensional array of antenna elements, which can be readily extended to two-dimensional arrays. In Figure 3-1, we have an array of N elements that are evenly spaced by distance d. The aperture size is the length of the array, $(N-1)d$. A plane wave x(t) with amplitude A, frequency ω_c, and wavelength λ propagates towards the array at angle θ.

The wavefront of x(t) arrives at each element at separate time intervals, and the received signal at the kth element is denoted as $r_k(t)$. Based on the angle θ and the array spacing d, we determine the distance the wavefront must travel to reach the kth element to be $kd \cdot cos\theta$. Given the signal's propagation speed to be the speed of light c, we calculate the time delays for the kth element to be $kd \cdot cos\theta/c$. The expression for r_k is the following.
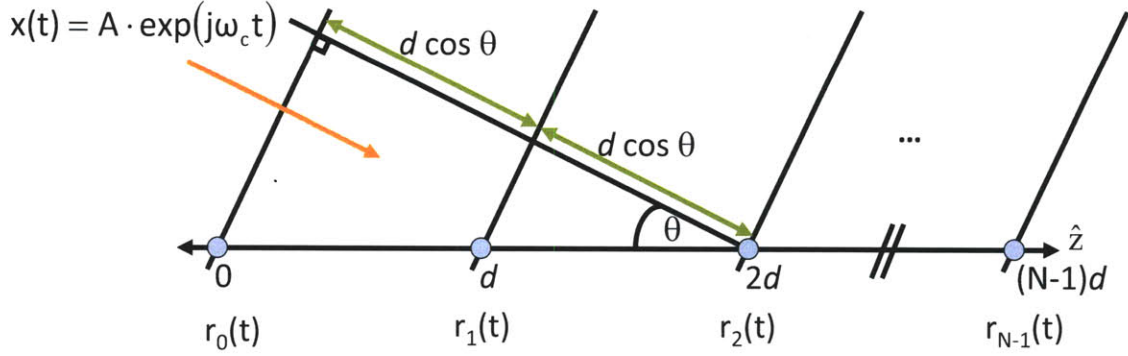
Figure 3-1: Signal wavefront approaching an array with aperture length (N-1)d.

$$r_k(t) = A \cdot exp\left(j \left[\omega_c \left(t - \frac{kd}{c}cos\theta \right) \right] \right) \qquad (3.1)$$

With the time delays known, we can apply our own delays τ_k to correct for the delay offsets at each element and align the wavefronts in the time domain. The sum of these signals will add constructively such that the output of the beamformer is x(t) amplified by the number of elements N.

In practice, signals may be propagating towards the array from all directions, and the purpose of the beamformer is to focus in one direction while suppressing signals from all others. We want to characterize the performance of the beamformer over all angles when it is steered towards a target angle θ_T. Since we know the desired direction, we apply delays τ_k that would align the wavefront of a hypothetical signal coming from the θ_T direction. The received signals r_k are summed after applying the delays and then normalizing by the number of elements N as shown in Figure 3-2. The output is y(t) which is expressed in Equation 3.2.

$$y(t) = \frac{1}{N} \sum_{k=0}^{N-1} r_k(t + \tau_k) \qquad (3.2)$$

We assume that the array's received signal comes from an angle θ that is not necessarily equal to θ_T. We substitute r_k from Equation 3.1 with amplitude A set to one into Equation 3.2. The delays τ_k are then set equal to $kd \cdot cos\theta_T/c$. Noting that c is equal to $f \cdot \lambda$, the simplified output y(t) is shown in Equation 3.3.
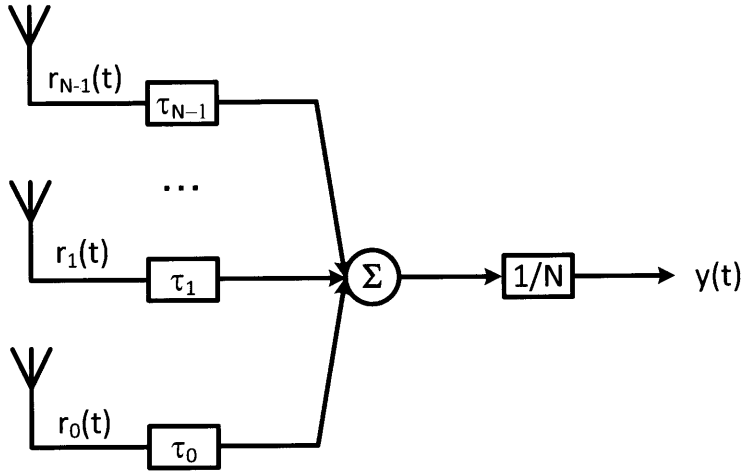
Figure 3-2: Block diagram of a delay and sum beamformer.

$$y(t) = A \cdot exp\left(j\omega_c t\right) \frac{1}{N} \sum_{k=0}^{N-1} exp\left(j\left[k\frac{2\pi d}{\lambda}\left(cos\theta_T - cos\theta\right)\right]\right) \qquad (3.3)$$

The beam pattern B_θ is the time independent component of y(t). It describes the gain of the array for a signal approaching the array at angle θ when the array is pointed in the θ_T direction.

$$B_\theta = \frac{1}{N} \sum_{k=0}^{N-1} exp\left(j\left[k\frac{2\pi d}{\lambda}\left(cos\theta_T - cos\theta\right)\right]\right) \qquad (3.4)$$

Figure 3-3 shows the magnitude of a beam pattern plotted against θ. The beam pattern is plotted for a 10 element array with each element spaced half wavelength apart $(d = \lambda/2)$, and the target angle θ_T set to $\pi/2$.

We observe that the beam pattern has a main lobe that peaks at $\theta = \theta_T$. The width of the main lobe determines the angular resolution of the beamformer. The half-power beamwidth (HPBW) is defined as the angular width in which the main lobe maintains half its peak power, and it is dependent on the size of the array. A longer array aperture will have narrower HPBW, which is required for finer steering resolution. Assuming $\lambda/2$ spacing between elements, the relationship between the HPBW and array size is described in [28] to have the relation in Equation 3.5.
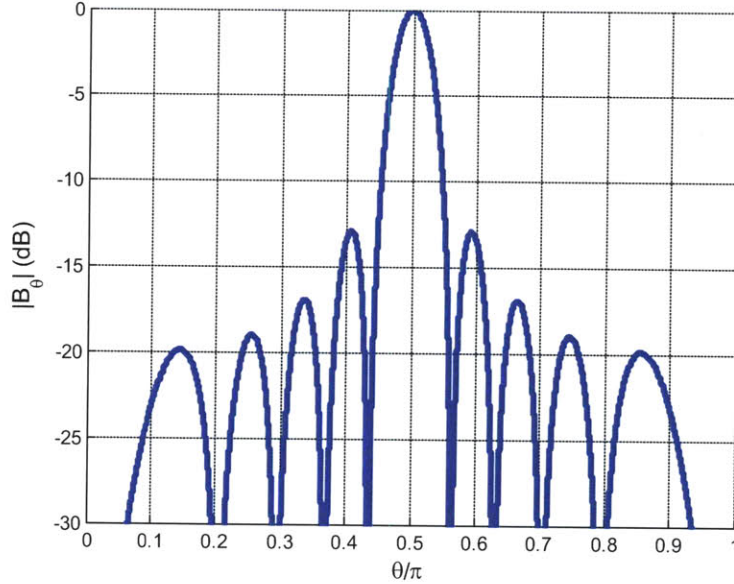
Figure 3-3: Beam pattern for 10 element array with d $= \lambda/2$ and $\theta_T = \pi/2$.

$$HPBW_\theta = 2sin^{-1}\left(0.891\frac{1}{N}\right) \tag{3.5}$$

The lower peaks in Figure 3-3 are known as side lobes and are inherent to the beamforming algorithm. The side lobes determine the amount of gain that interferers, which are undesired signals from directions other than the target direction, will see. Ideally, we want the side lobes to be as low as possible since they represent contributions from unwanted directions that corrupt the beamforming output.

3.2 Types of Beamforming

3.2.1 Analog Beamforming

Beamforming consists of applying delays, or phase shifts in the narrow-band case, to each receiver in an array and then summing the signals across the array. Beamforming is generally done in the analog domain. Analog phase shifters are used to apply the delays and power combiners perform the summation. There are several approaches to analog beamforming relating to the placement of these components in the receiver's
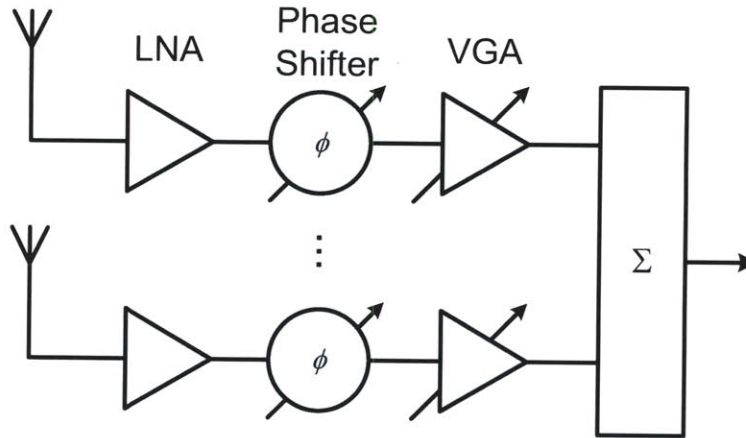
Figure 3-4: Beamforming with RF phase shifting and power combining.

signal chain, and this section will compare the advantages and disadvantages of each.

The first approach is radio frequency (RF) beamforming, which is the most straightforward approach to beamforming and is a direct translation of the beamforming algorithm. It is depicted in Figure 3-4. Each antenna element is followed by a low noise amplifier (LNA) to increase the signal strength. The phase shifts are applied at the carrier frequency. Variable gain amplifiers are then used to correct for mismatches between the different element paths and can be used for alternative beamforming algorithms that weight the individual element's amplitudes separately. Finally a summation is performed with a power combiner operating at the carrier frequency.

An advantage to this method is that interferers are canceled before the subsequent down-conversion. This alleviates the dynamic range specifications for the mixer and the following blocks. However, implementing phase shifts at the carrier frequency is a challenge since the time delays are so small. Furthermore, the lossy nature of operating at the RF makes efficient power combining difficult to achieve.

Alternate approaches to RF beamforming require down-converting the received RF signal to an intermediate frequency (IF) before manipulating them. Mixing to an IF preserves the phase information of the RF signal in radians, but increases the time delays by ω_c/ω_{IF}. There are two options: applying phase shifts at the local oscillator
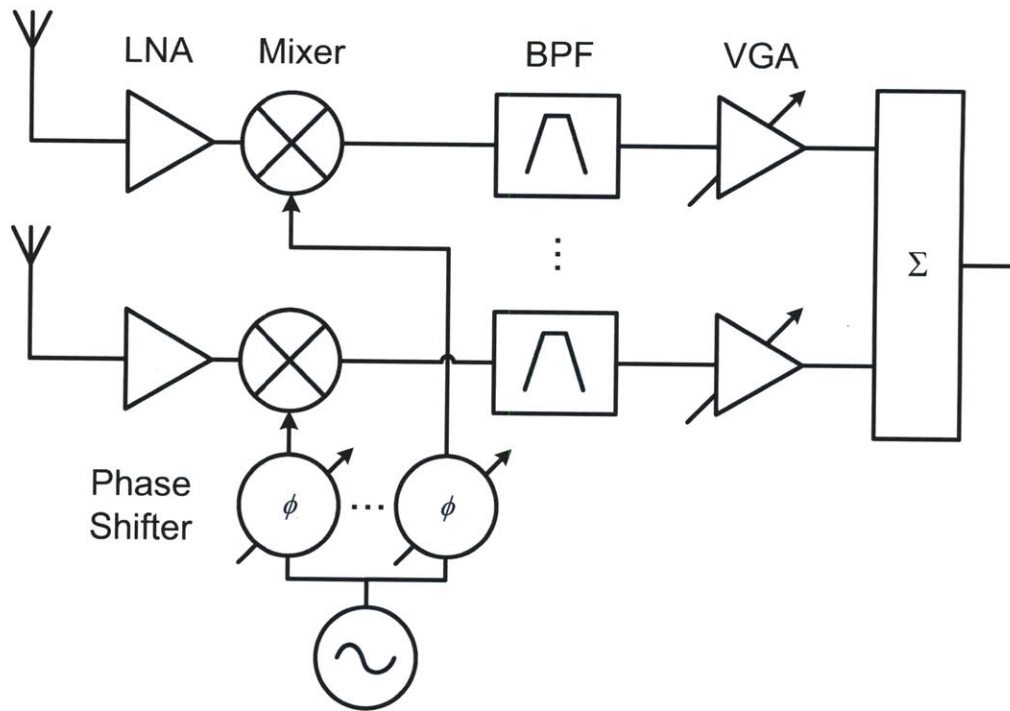
Figure 3-5: Beamforming with LO phase shifting and IF power combining.

(LO) shown in Figure 3-5 and applying phase shifts at the IF shown in Figure 3-6. In both cases, the phase shifting has been removed from the RF signal path. Nonlinearity and losses in the phase shifter have a lesser impact on the system. Also, the power combining is performed at the IF, which is less lossy and easier to implement than at the RF. The phase shifting in IF beamforming occurs at the lowest frequency and is therefore the easiest to implement among the three. LO phase shifting still has the advantage of being removed from the signal path. Since the LO is generally used to strongly switch a mixer, nonidealities in the LO phase shifting are not as severe as in RF phase shifting [29].

Both LO and IF beamforming require higher constraints on the mixer. Unlike RF beamforming in which interferers have been removed before the mixing process, these methods allow interferers to pass through the down-conversion. Therefore, the mixer must have enough dynamic range to handle interferers from all directions. Another drawback of these methods is the requirement of an LO distribution network. Every element in the array requires an LO, and routing and buffering the LO can be costly
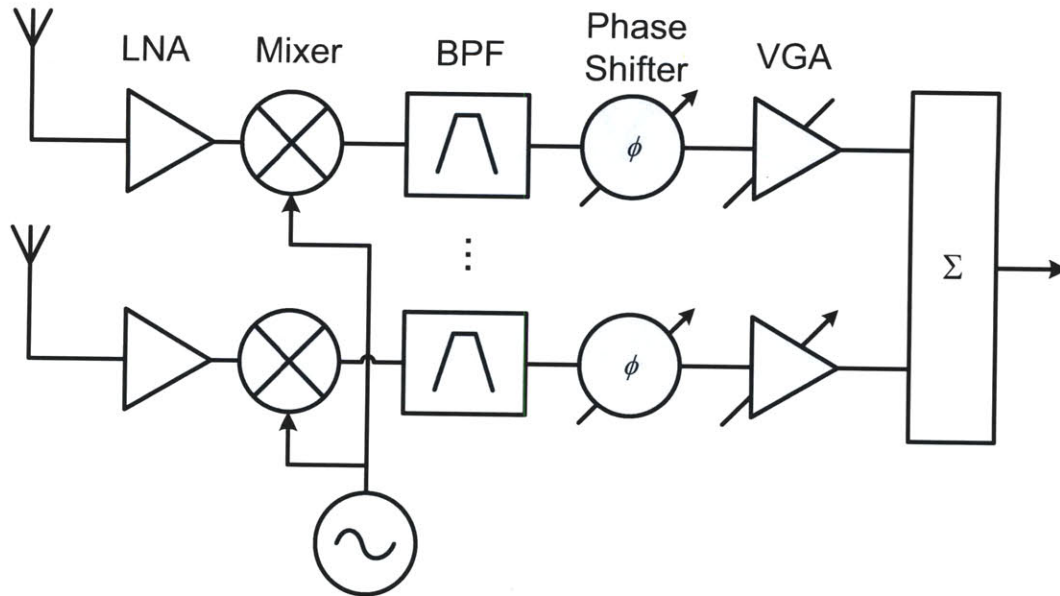
Figure 3-6: Beamforming with IF phase shifting and power combining.

and cumbersome [29].

3.2.2 Digital Beamforming

The focus of this research is to explore a system that uses digital beamforming. This method requires converting the phase and amplitude information at each antenna element into digital data and then applying the delays and summation in the digital domain. By performing all of the processing digitally, we push most of the receiver complexity into the digital domain and remove challenging analog components such as the phase shifters and power combiners.

Digital beamforming provides more flexibility that is not possible with the previously described analog architectures. The phase shifters in analog beamformers are generally implemented as programmable circuits that have quantized phase steps. This quantization constrains the sets of phase delays that are possible. With analog beamforming, we can only apply one set of phase shifts at a time, so we can direct the beam at only one target location at a time. With the digital beamforming approach, rather than scanning the scene pixel by pixel, we capture raw data that includes in-

formation from all directions. We then apply different sets of phase shifts on the data to calculate the pixels. Since this is done digitally, we are not constrained to calculate each pixel one at a time but can process multiple pixels in parallel. Using the computational power of modern CPUs, more sophisticated beamforming techniques and imaging algorithms can be implemented.

Although we push most of the complexity into the digital domain with digital beamforming, we do place more constraints on the remaining analog circuitry. For example, since the beamforming occurs after the mixer, we have not filtered out interferers which is similar to the case of LO and IF beamforming. The mixer must have the dynamic range to handle signals from all directions including interferers. We also require additional circuitry that can convert the phase and amplitude information into digital signals.

3.2.3   Digitizing Phase and Amplitude

The standard method for measuring the phase ϕ and amplitude A of a signal is to decompose it into its in-phase (I) and quadrature (Q) components. The phase and amplitude are then calculated as follows.

$$A(t) = \sqrt{I(t)^2 + Q(t)^2} \tag{3.6}$$

$$\phi(t) = tan^{-1}\left(\frac{Q(t)}{I(t)}\right) \tag{3.7}$$

In Figure 3-7, a homodyne architecture is considered in which I and Q can be obtained by mixing down the input RF signal to baseband where they are sampled directly. Two identical signal paths require two mixers, two low-pass filters, and two analog-to-digital converters (ADC). This comes as additional power and area costs for the system. The homodyne approach requires generation of a quadrature MMW LO which places more complexity in the MMW PLL and LO distribution. LO path losses and quadrature phase offsets are common challenges for RF transceivers that are exacerbated when considered at MMW frequencies. With wavelengths on the

38

order of millimeters, mismatched LO routing lengths have more bearing on the fixed phase offset. Furthermore, the quadrature LO amplitude levels are more susceptible to mismatches between the quadrature LO paths due to increased sensitivity to parasitic losses. I/Q imbalance limits the accuracy of the phase and amplitude measurements.

Since the homodyne topology down-converts to baseband, there are also issues of DC offsets and LO leakage. Poor isolation of the LO can cause leakage into the RF signal. Self-mixing then creates a DC component that appears in the baseband signal. This will corrupt the measurement data and may even saturate the ADC [30].
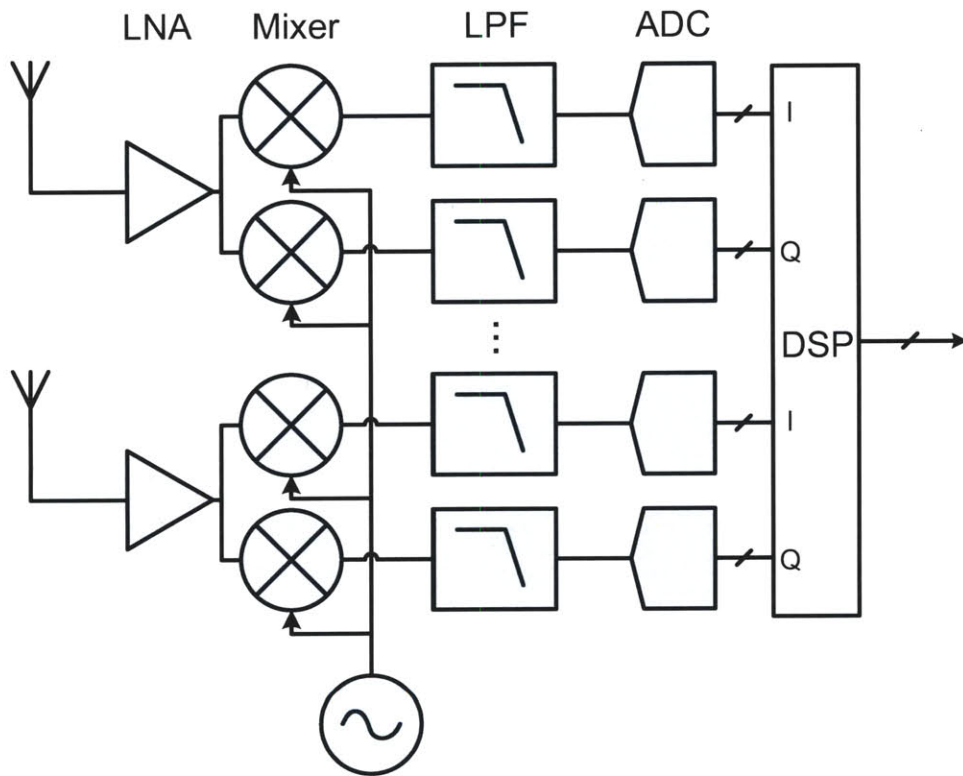
Figure 3-7: Block diagram of system which mixes down to baseband to measure I and Q.

Another method for determining the I and Q is the digital-IF architecture shown in Figure 3-8. Rather than mix to baseband, we mix the RF signal to an IF where it is sampled by a single ADC. The I and Q components are then calculated in the digital domain. Since there are no separate I and Q branches, I/Q imbalance is no longer an issue. And there is no need for a quadrature LO. However, the ADC sampling

frequency must be at least twice the Nyquist rate of the IF, and I and Q are obtained by multiplying the digitized input signal with digital representations of quadrature sinusoids.

The main challenge of this design is the requirements on the ADC. The ADC must sample at least four times the IF, which is on the order of hundreds of megahertz to a gigahertz for the imaging receiver. The ADC must have sufficient resolution to accurately measure the phase and amplitude. These constraints require an ADC that is difficult to achieve even with modern technologies [30].
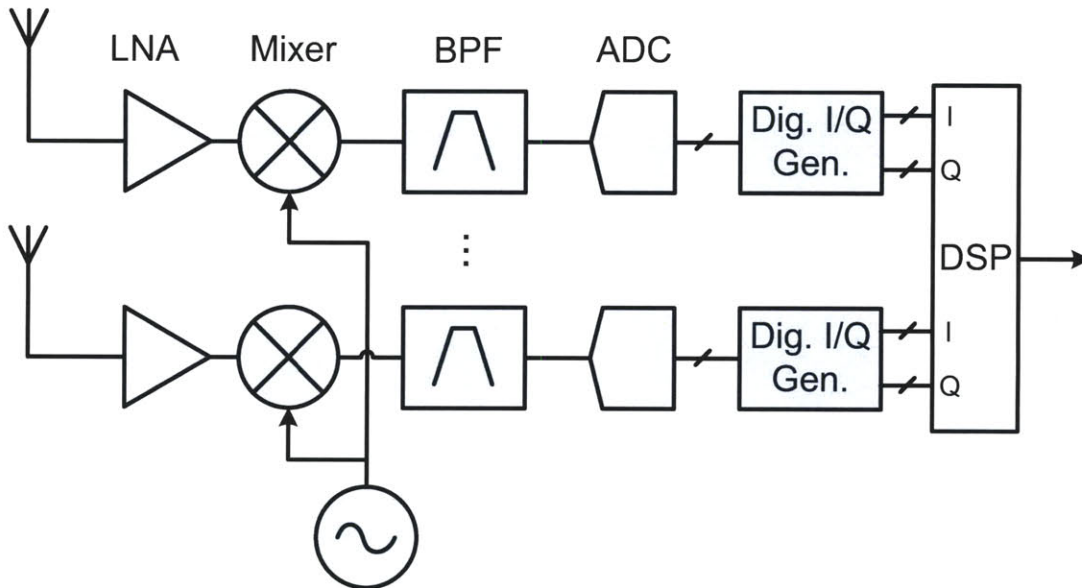
Figure 3-8: Block diagram of system which mixes down to IF where the signal is digitized and I and Q are generated in the digital domain.

3.2.4 Separation of Phase and Amplitude Estimation

In this project, we aim to separate the phase and amplitude estimation. Rather than compute these parameters by measuring the I and Q components, we will measure the phase and amplitude of the input signal directly. This approach is illustrated in Figure 3-9, in which the MMW signal is mixed to an IF that has two branches that are independent of one another. One measures phase, and the other measures amplitude. The phase estimation branch will use a method that we have coined Digital Phase
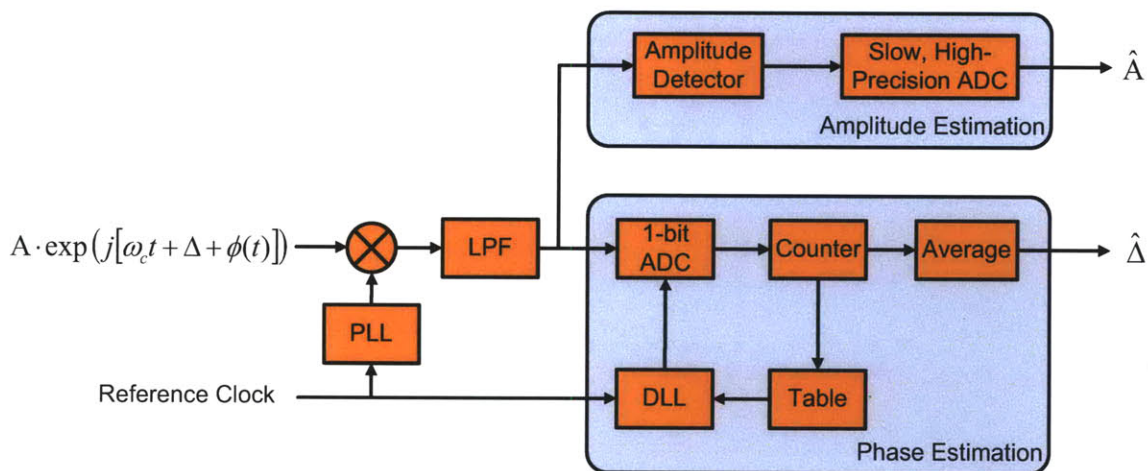
Figure 3-9: Block diagram of system with separate phase and amplitude estimation.

Tightening, and it will be discussed further in Chapter 5.

The amplitude estimation branch consists of an amplitude detector and an ADC. Two examples of amplitude detectors are shown in Figure 3-10. Both contain a parallel resistor and capacitor pair which sets a time constant for the charging and discharging of the output node. The time constant is designed based on the rate of change of the scene. This will be much longer than the period of the IF, so that the output does not follow the IF signal but only captures its amplitude. The amplitude detector performs analog averaging.

The first example uses a simple diode. When the diode is forward biased, current will flow into the output node, and the output will follow the input with a diode drop. The second example is a source follower. In this topology, the output node will follow the input by a V_{gs} drop, which is equal to the threshold voltage, V_T, plus the overdrive voltage. Typical V_T's for current silicon technologies are about 0.3V, and even with an overdrive voltage of 0.4V, the V_{gs} drop will be smaller than the diode drop, so the source follower will have a larger range given the same supply voltage. In order to maintain a constant voltage difference for the source follower case, the source and bulk should be tied together to prevent variations in V_T due to body effect. This may require a triple-well process. Another important difference is that the source follower acts as a buffer that drives current into the output from the

41
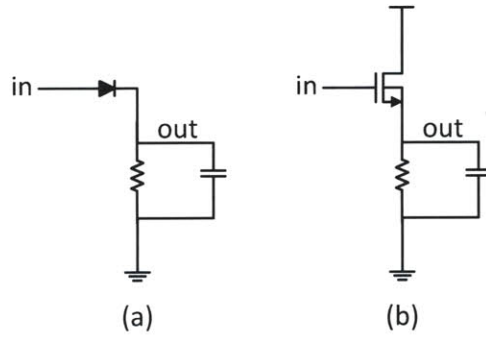
Figure 3-10: (a) Amplitude detector using diode. (b) Amplitude detector using source follower.

supply. The diode approach requires the input signal to drive the amplitude detector output, which loads the IF signal. If a triple-well process is available, the source follower which acts as a buffer is preferred.

The second component of the amplitude detector is the ADC. Since imaging systems have relatively slow frame rates, it is not necessary to measure the amplitude at the same rate as the IF. Instead, we can use the extra time to obtain a more accurate measurement. A slow high precision ADC, such as a dual-slope or delta-sigma ADC, can be used for this purpose.

Compared to the baseband I/Q method, the separate phase and amplitude estimation also requires two paths. However, the phase and amplitude estimations are independent of each other, so there is no requirement for matching the paths. Also, only one mixer is needed in this case, although additional buffers following the mixer may be required to isolate the two paths. The baseband I/Q approach requires two ADCs that will sample the I and Q at baseband. We expect I and Q to be narrow-band signals, so these ADCs only need to operate fast enough to supply information to achieve the system framerate. They will need to be high resolution to accurately measure I and Q for the phase and amplitude calculations. These ADCs have similar requirements as the ADC in the separate phase and amplitude approach, except two of them are required.

In order to make a fair comparison with the IF I/Q architecture, we assume that
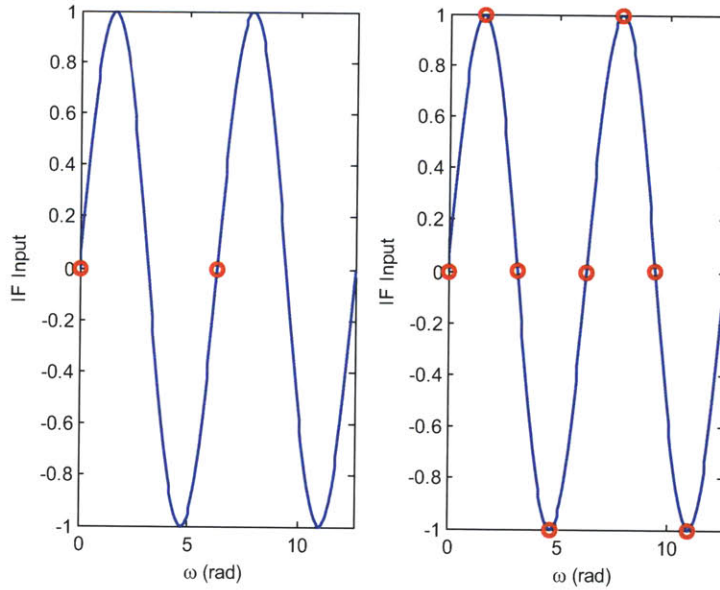
Figure 3-11: (a) Digital Phase Tightening requires sampling at the IF. (b) Digital IF I/Q architecture requires sampling at twice the Nyquist rate of the IF.

the IF frequencies are the same for both cases and that the same amount of averaging can be used. With Digital Phase Tightening for separate phase and amplitude estimation, the phase estimation branch can sample at the same rate as the IF. The IF I/Q architecture requires a higher performance ADC operating at twice the Nyquist sampling rate. This difference in sampling is shown in Figure 3-11. The specifications for the ADC have been prohibitively demanding even without power or cost constraints [30]. However, recent developments in ADC designs including zero-crossing based ADCs may make this approach feasible. These ADCs operate in the hundreds of megahertz range with up to 12-bit resolution while still maintaining very low power dissipation [31] [32]. The performance of these ADCs show the feasibility of the IF I/Q architecture. We consider this approach a viable alternative in the future, but in this thesis, we will consider a system with separate phase and amplitude measurements which avoids I/Q mismatch.
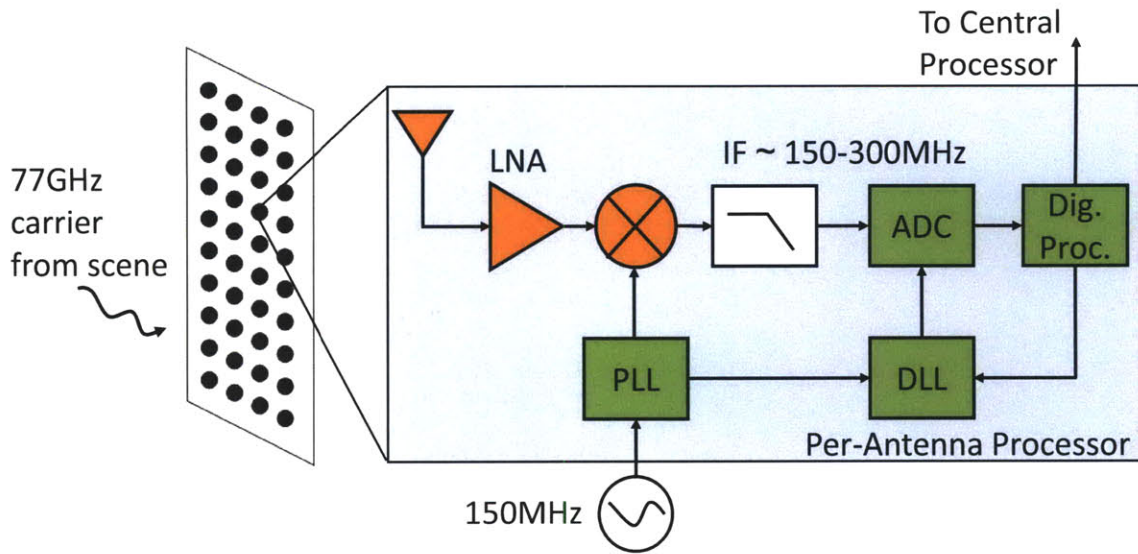
Figure 3-12: Active imaging system using purely digital beamforming.

3.3 Imaging System

Figure 3-12 shows a block diagram for an active imaging receiver operating at 77GHz. The system consists of a planar antenna array, and each antenna has a receiver. The receiver mixes down the received signal to an intermediate frequency where the phase and amplitude are measured. This information is sent to a CPU where digital beamforming is done to create an image. For simplicity, only the phase estimation portion of the receiver is shown in the figure.

A reference clock of 150MHz is routed to each element in which a PLL generates an LO to down-convert the received signal. This architecture requires a PLL at every node, which may seem redundant since an alternative architecture could have a single PLL that generates one MMW LO that would route to each element. However, routing this LO to every receiver is nontrivial, especially for a large array. Losses and variations in phase due to routing are primary concerns. Buffering the LO would be inefficient and power consuming. Instead, this design avoids these issues by distributing the reference clock which is a more manageable task.

There were several considerations that went into the selection of 150MHz for the reference clock. We wanted a divide ratio that could be achieved using a cascade

of divide-by-2 frequency dividers, which simplifies the implementation of the divider chain. In order to allow easier distribution of the reference clock among array elements, we wanted to keep the reference clock on the order of 100MHz. An even lower reference clock would ease the distribution but requires a larger divide ratio in the PLL. A larger divide ratio requires more frequency dividers in the PLL's divider chain which costs area and power, although the area and power overhead can be small if CMOS frequency dividers are implemented for the additional low frequency dividers. More importantly, the divide ratio scales the phase noise contribution from the reference clock and the charge pump. We aimed for timing jitter below 1ps at the output of the PLL. Given the noise of the charge pump and the simulated phase noise from the VCO in [33] to be -76dBc/Hz at 1MHz offset, we expected that we could achieve -85dBc/Hz of phase noise at 1MHz offset with a divide ratio of 512. This phase noise would correspond to jitter of 830fs. The design of the PLL will be discussed in detail in Chapter 4. Given the 77GHz carrier, divide ratio of 512, and requiring the reference clock to be on the order of 100MHz for routing, the reference clock was chosen to be 150MHz, so a divide-by-512 PLL could use it to generate a 76.8GHz LO.

A single transmitter will act as a source and is used to flood the scene with 77GHz radiation, which will be reflected back to the receiver array. The transmitter will illuminate the scene in a similar manner as a flash does for an optical camera. In order to maintain frequency synchronization between the transmitter and receiver, the transmitter will use the same reference clock as the receiver.

The system targets a framerate of 10fps. The target pixel resolution is 6cm at 30m distance. This corresponds to an angular resolution of 0.115°. Based on Equation 3.5, we calculate that we need 1000 elements to achieve this resolution.

The design of the PLL will be discussed in Chapter 4. The feedback loop consisting of the delay-locked loop (DLL), ADC, and digital logic will be discussed in Chapter 5. The IF is specified as a range from 150MHz to 300MHz. There are a number of tradeoffs that are considered for the selection of the IF. These will be discussed in Section 5.7 after further understanding of this loop.

3.4 Phase Noise Requirement

3.4.1 Element-to-Element Delay

The target specification for the imaging system is a pixel resolution of 6cm at 30m distance. This is represented in Figure 3-13, which shows an array of evenly spaced elements with spacing $\lambda/2$. Two plane waves are shown approaching the sensor array with the spacing from the target resolution. The angle they produce with the array is θ, which is about $0.115°$. A signal propagating towards the array from broadside, or an angle of incidence of $0°$, will hit every element at the same time. If the signal propagates from incident angle θ, there will be a time delay between the elements. The time delay $\Delta t_{n+1,n}$ is defined to be the difference in arrival time between element n and n+1. With the element spacing known and the propagation of the plane wave to be the speed of light, c, we can calculate $\Delta t_{n+1,n}$.

$$\Delta t_{n+1,n} = \frac{d}{c} sin\theta \qquad (3.8)$$

For a 77GHz carrier, $\Delta t_{n+1,n}$ is about 13fs. This suggests that receivers at each sensor element must be able to measure down to at least 13fs in order to beamform properly.

3.4.2 Effect of Phase Noise on Beam Pattern

Beamforming utilizes the phase and amplitude information of the received signal at each array element to focus the array in one direction while attenuating signals coming from other directions. However in practice, the analog circuits at each element will contribute noise that will corrupt the signal quality. In the imaging system, phase noise is particularly important because the periods of MMW signals are so small - on the order of ten picoseconds with time delays between elements being even smaller. Reducing phase noise levels down to this level is a challenge even with modern technologies.

Phase noise will cause inaccuracies in the phase information and is equivalent
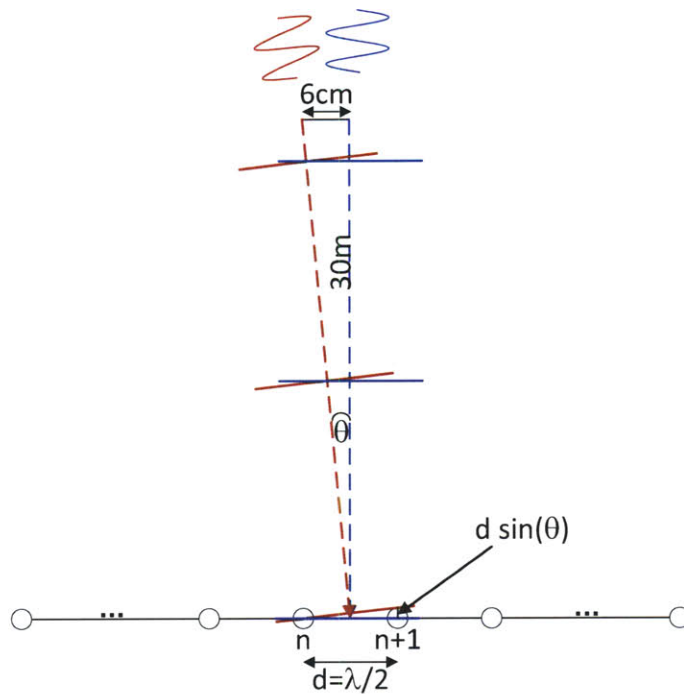
Figure 3-13: Diagram of minimum time delay between elements for plane waves arriving with spacing from target pixel resolution. The blue wave approaches the array from broadside and hits each element at the same time. The red wave is propagating from θ which is the next pixel over from the blue wave and path distance between elements is $d \cdot sin\theta$

to adding incorrect time delays during the beamforming process. Intuitively, the incorrect time delays will degrade the constructive addition in the target direction, which essentially causes the beamformer to lose focus. In order to obtain a fuller understanding of the effect of phase noise on beamforming, we analyze the effects of increasing phase noise on the beampattern.

In [34], Accardi describes key parameters in the beampattern that are corrupted by phase noise: the main lobe beamwidth, the tilt of the main lobe, and the side lobe suppression. These parameters are portrayed in the beampattern diagram of Figure 3-14. The main lobe beamwidth, or half-power beamwidth, is defined in Section 3.1 and refers to the width of the main lobe in which the lobe is 3dB below the peak value. The tilt of the main lobe is defined as the offset from the angle of the beampattern's peak value to the target angle. The side lobe suppression is the difference in gain between the main lobe peak and the highest side lobe peak.

47
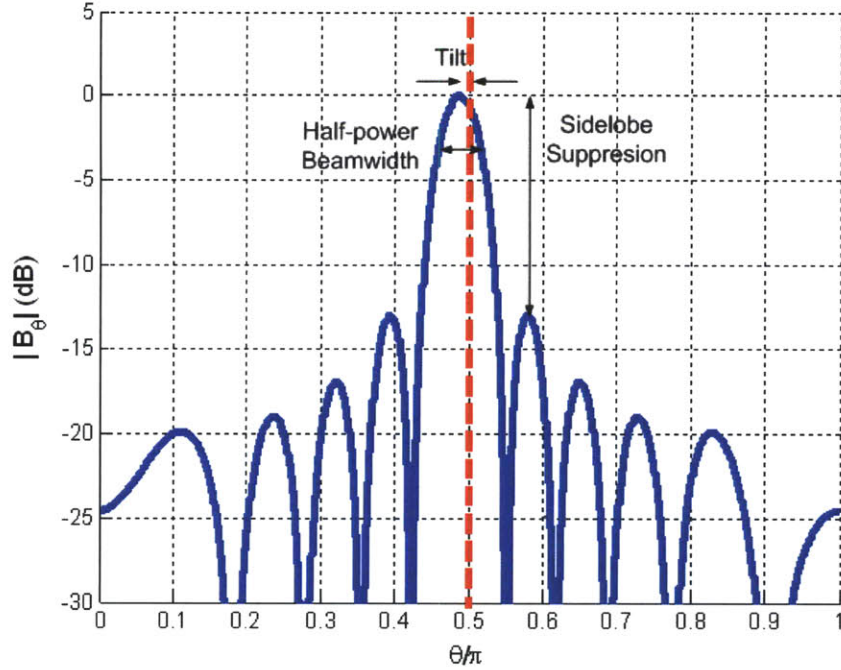
Figure 3-14: Beamformer performance metrics labeled on the beampattern of a 10 element array.

Accardi describes different stages of beampattern degradation as the phase noise level is increased and these metrics begin to break down. For low levels of noise, the beampattern is stable and the main lobe and side lobes are well-defined. As phase noise increases, the beampattern enters another stage in which the side lobe levels begin to increase. He defines a side lobe suppression break point, σ_ϕ^{SB}, which is the amount of phase noise that can be tolerated before the side lobe levels begin to rise dramatically. Empirically, this occurs when the side lobe suppression level is approximately -12dB. He fitted the following equation to approximate the side lobe suppression break point based on the number of elements in the array.

$$\sigma_\phi^{SB} \approx 0.698 \cdot log_{10}N - 0.250 \tag{3.9}$$

If the phase noise is raised further, the beampattern's tilt begins to degrade as the main lobe wobbles. The phase noise level that marks the beginning of this region is the tilt break point, σ_ϕ^{TB}. The expression that describes this value in terms of the

number of elements in the array is shown in Equation 3.10.

$$\sigma_\phi^{SB} \approx 0.584 \cdot log_{10}N - 0.681 \tag{3.10}$$

Increasing the noise even further causes the beampattern to collapse, in which a main lobe can no longer be distinguished. In this region, the beamformer has no steering capability and is no longer functional.

We are interested in the level of phase noise at which the beamformer performance degrades as this will reduce the pixel quality for the imaging system. This would be the side lobe suppression break point σ_ϕ^{SB}. We substitute N = 1000 into Equation 3.9 to obtain a σ_ϕ^{SB} of 1.844 radians. Converting this value into timing jitter at the carrier frequency of 77GHz gives 3.8ps. This analysis suggests that the beamformer can withstand up to 3.8ps of jitter at the carrier before the side lobe levels begin to encroach into the main lobe level. This is a significant amount of jitter given that the period of the carrier is only 13ps.

An important characteristic of beamforming is that the beam pattern becomes increasingly robust to phase noise as we increase the aperture size of the array. Increasing the aperture size refers to adding more elements while holding the element spacing constant. Since the phase noise at each element is independent, the error due to phase noise is averaged out over the array. As the number of elements in the array is increased, more averaging takes place. This increases the tolerance for phase noise of the beamformer.

Another effect of phase noise on the beampattern is the reduction in gain of the main lobe. This occurs at lower phase noise levels than the break down of the side lobe suppression. As the phase noise increases, the main lobe level will drop, but the side lobe level will lower accordingly, maintaining the side lobe suppression level. Since the side lobe suppression is maintained, it is possible to apply additional gain to the beamformer output in the digital domain to compensate for the lowered array gain. However, the beamformer's dynamic range is reduced, and compensating for varying peak gains can be difficult and will degrade the image quality.
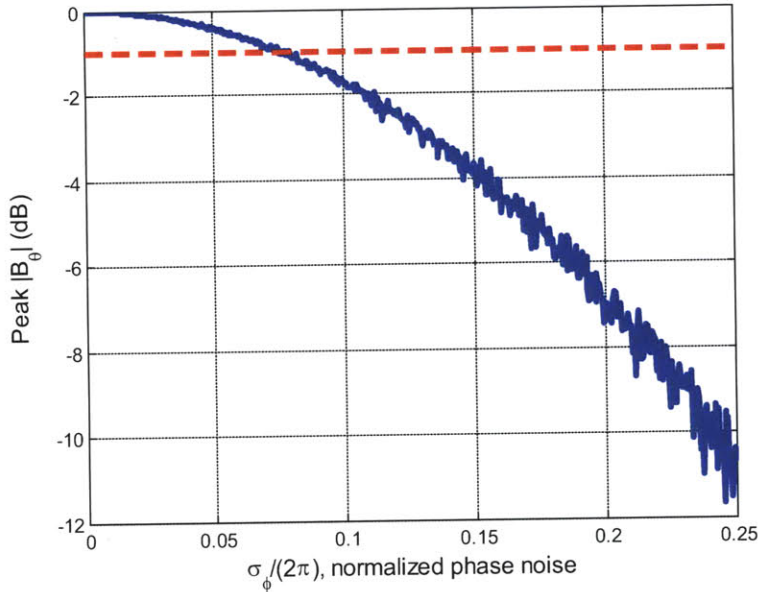
Figure 3-15: Plot of main lobe peak level versus normalized phase noise for a 1000 element array. The array is steered towards $\theta = \pi/2$. The red dashed line represents -0.5dB attenuation. This level of main lobe degradation occurs when the phase noise is $0.075 * 2\pi$ radians.

The reduction in peak gain is plotted in Figure 3-15. An array of 1000 elements evenly spaced by $\lambda/2$ is steered towards the direction $\theta = \pi/2$. Gaussian phase noise with zero mean and standard deviation σ_ϕ is applied to each element of the array independently. The peak level of the beampattern's main lobe is then plotted against the phase noise level σ_ϕ. With zero noise, the main lobe's peak level is 0dB. As the noise increases, the peak level drops since the noise causes misalignment in the received signals. Therefore, the summation is not fully constructive, and the array's gain has been lowered.

If we allow a reasonable tolerance of 1dB in peak gain degradation, then the allowable level of phase noise is $0.075 \cdot 2\pi$ radians according to Figure 3-15. For a 77GHz carrier, this corresponds to 975fs of jitter. In other words, the beamformer can tolerate up to 975fs of jitter at each element for an array gain deviation to be less than 1dB. This is a tighter specification than the point at which the side lobe suppression breaks down, but it is considerably higher than the element to element delay of 13fs. The system was designed with the 13fs element to element delay in mind, which

turned out to be an overdesign given the robustness of the array to phase noise. We will continue to use the 13fs specification as motivation for the circuit design while acknowledging that this level of accuracy is more than necessary for this application.

3.4.3 Evaluation of Far-field Assumption

In Section 3.3, we found that an array size of 1000 elements is needed to achieve the target pixel resolution of 6cm at 30m distance. This is a large array, and we must evaluate the assumptions we made. One concern is that the plane wave assumption for beamforming changes as the array aperture size increases. As the array becomes larger, the far-field region, which allows for the plane wave approximation, is pushed further away. We consider the incoming signals to propagate from radiating sources with circular wavefronts. If the source is far away, the circular wavefronts appear to be plane waves as they approach the sensor array. However, as the array aperture is made larger, the array becomes more sensitive to the curvature of the wavefronts.

Figure 3-16 shows a source radiating circular waves towards the array. It is located at a distance r and angle θ from the array. The time delays required at each element is dependent on the distance between the source and each element and the angle. This distance is r_n and is calculated in Equation 3.11 using the Law of Cosines.

$$r_n = \sqrt{r^2 - 2rx_n cos\theta + x_m^2} \qquad (3.11)$$

The time delay for element n, τ_n, is difference in time it takes the wavefront to reach the center of the array and to reach element n. The variable c is the propagation speed of the wavefront.

$$\tau_n = \frac{r_n - r}{c} \qquad (3.12)$$

This formula contrasts the time delay for a plane wave which was described in Section 3.4.1. The time delays in the plane wave case does not depend on r but only on the propagation angle of the wavefront and the element spacing of the sensors.

If the source is close enough that it is in the near field of the array, then the

51
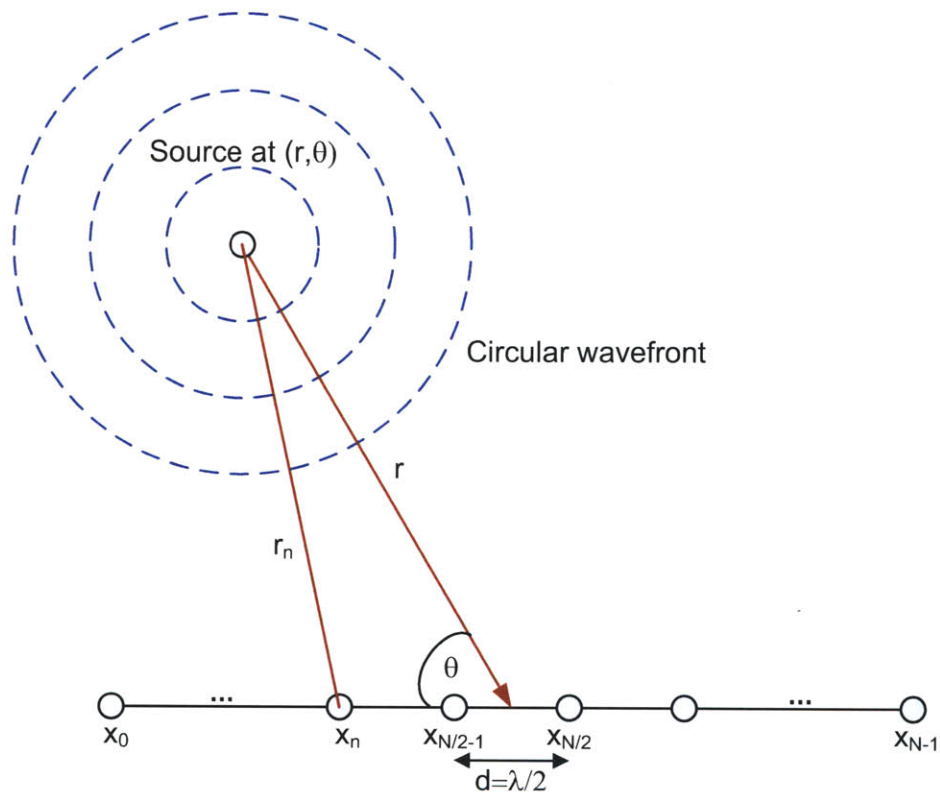
Figure 3-16: This is a diagram of radiating source located at distance r and angle θ from the array. The source is emitting circular waves towards the array.
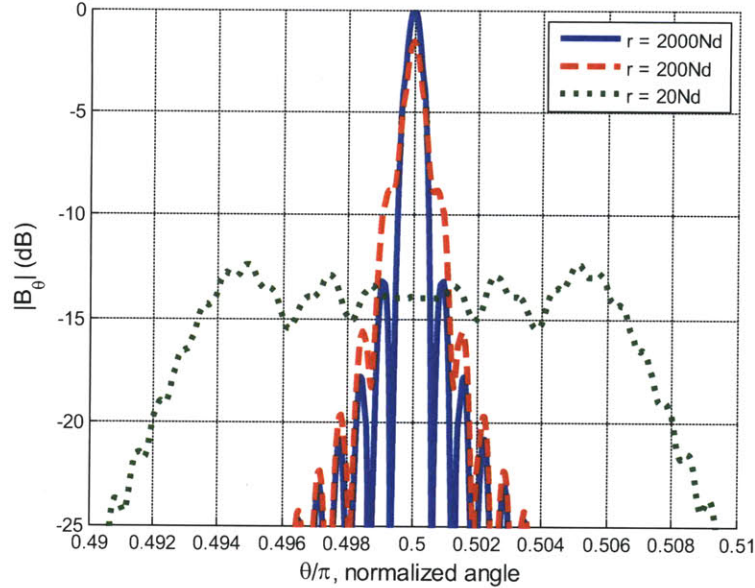
Figure 3-17: Zoomed-in plot of the beampattern main lobe when plane wave delays are applied for different source distance r. This was simulated for a 1000 element array spaced $\lambda/2$ apart and is steered towards $\pi/2$.

curvature of the wavefront affects the arrival time at each element, so the delays at each element must be adjusted accordingly. If we apply the same delays for a plane wave as a circular wave, the beamformer's response can be severely degraded. This is demonstrated in Figure 3-17, which shows the main lobe of the beampattern when plane wave delays are used to beamform a circular wave sources from varying distances of r. These beampatterns were simulated for a 1000 element array. The r distances are displayed as scaled factors of Nd, which is the approximate length of array aperture. For r equal to 2000Nd, the source is well into the far field, and the main lobe is well-defined and the peak is close to the ideal case of 0dB. For r equal to 200Nd, the sidelobes become partially absorbed into the main lobe. The main lobe peak has dropped to about -1.5dB and the beamwidth is larger. For even lower r equal to 20Nd, the source is in the near field of the array. There is large degradation of the beampattern in which the main lobe has become very wide and the gain has dropped significantly to about -14dB.

The difference between the circular and plane wave responses also depends on the

angle of arrival. If the radiation source is located at endfire with respect to the array, then the delays will be identical for a circular or planar wavefront. However, as the source moves away from endfire towards broadside, then the time delays between the circular and plane waves will be increasingly different. Therefore, we expect deviation in the beampattern due to incorrect plane wave delays to be higher near broadside and lesser near endfire.

In order to analyze this deviation, we define an error ratio between the two beampatterns. We assume that a radiating source is located at θ_T at a distance r from the array and is emitting circular waves. $B_p(\theta_T)$ is the beampattern evaluated at θ_T when the array is steered towards the source at angle θ_T by applying plane wave delays. $B_c(r, \theta_T)$ is the beampattern evaluated at θ_T when the array is steered towards the source at angle θ_T and distance r by applying circular wave delays. $B_p(\theta_T)$ and $B_c(\theta_T)$ are the values of their respective beampatterns when steered directly at the source. The error ratio ϵ is defined to be the amplitude of the ratio between $B_p(\theta_T)$ and $B_c(r, \theta_T)$ and represents the error caused by using plane wave delays for an incoming circular wave [35].

$$\epsilon(r, \theta_T) = 20 \cdot log\left(\frac{B_p(\theta_T)}{B_c(r, \theta_T)}\right) \tag{3.13}$$

In Figure 3-18, the error ratio is plotted in decibels against θ_T for different values of r in a 1000 element array. At angles 0 and π where the signal source is at endfire, the error ratio is one for all cases because there is no difference between delays applied for circular and plane waves. Between the ends, there is a drop in the error ratio which shows that there is a discrepancy in the beampattern's main lobe peak when plane wave delays are used for circular wavefronts. For r equal to 2000Nd, the error appears flat in this plot and is at most -0.015dB, which shows that the plane wave approximation holds strongly when the signal source is at least this distance away. For decreasing values of r, the error becomes increasingly significant. For r equal to 200Nd, the worst case error is about -1.6dB. For r equal to 20Nd and 2Nd, the error is so large that the beamforming is unusable.
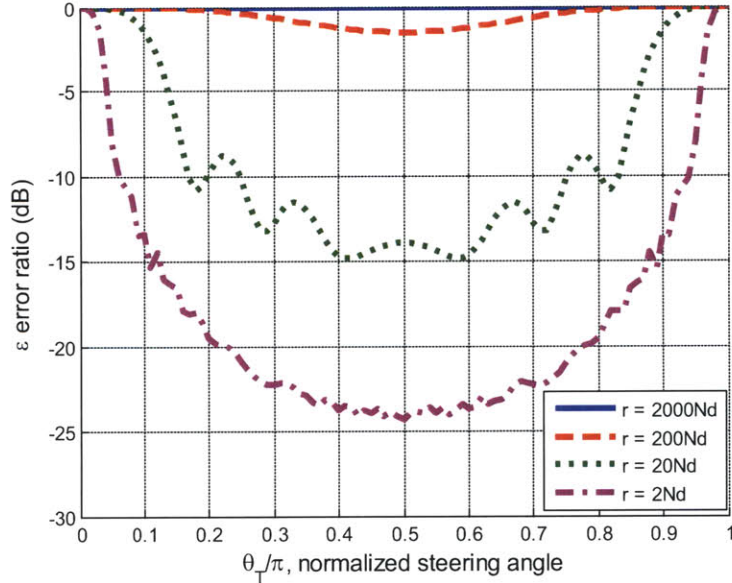
Figure 3-18: Plot of error ratio versus normalized steering angle for different source distance r. The array is steered directly at the source, θ_T, but plane wave delays are applied regardless of the true distance of the source. The y-axis shows the error ratio ϵ which is the deviation from the ideal beampattern, when correct circular wave delays are applied. This was simulated for a 1000 element array spaced $\lambda/2$ apart.

For this project's imaging system, the target pixel resolution is 6cm at 30m distance. An r of 30m distance is about 15Nd, which according to Figures 3-17 and 3-18 is strongly affected by the circular delays and is therefore very much in the near field. In order to beamform properly, the source distance must be determined, so the correct delays can be calculated. The imaging system would work in conjunction with a range finding system. The range finding system can be built as a separate system, or it can be implemented as a time-interleaved mode for the imaging array. Since the imaging system requires time to perform the digital beamforming and imaging processing, we can operate the imaging array during this time to measure the distance of target sources. MMW radar systems and techniques have been demonstrated to measure range and velocity of target objects [36] [37] [38].

One possible approach to measuring range is described by Kuroki et al. in [37], in which a pulse train is emitted and the receiver measures the reflected pulses. The time difference between the transmitted and received pulses can be used to determine

the distance of a target object. A distance of 30m corresponds to a time difference of 200ns for a signal propagating at the speed of light. The authors achieved accuracy within 1% for distances of 2m to 40m by modulating a signal with pulse width of 5ns and repetition frequency of 2MHz with a 59GHz signal. This is a fairly straightforward technique for range finding that could be used in this imaging system.

An alternate approach to measuring the range of a target and then calculating the appropriate delays is simply applying delays for all angles and all ranges of interest. From a top-view perspective, we can view the area of interest in front of the array as a grid as shown in Figure 3-19 that is dependent on the range r and angle θ. We can calculate and apply delays to focus the array for all points in the grid and then measure the radiation from each point. This method allows us to image the scene in a manner similar to ultrasound imaging in which beamforming is used to focus sound waves at different angles and depths in a human body [39]. Since delays are calculated for all angles and ranges, this eliminates the need for a range finding system.
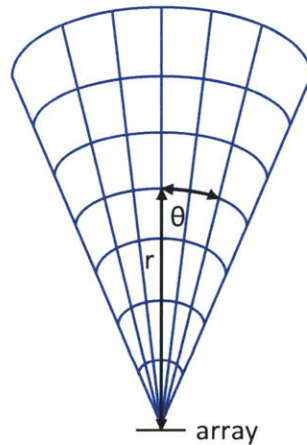
Figure 3-19: A top-view grid of the area in front of an array. Delays can be calculated to measure radiation from all points in the grid giving a two-dimensional image of the region

3.4.4 Sensitivity to Mismatch in Distance r

Near field beamforming requires measurement of the source distance, r, in order to calculate the appropriate time delays for the incoming circular wavefronts. For this

project, we will assume that there is a range finding system that will assist the image processing. The range finding system will provide an estimate of the source distance r', which will have some error from the true value. We want to know how much error the beamformer can tolerate.
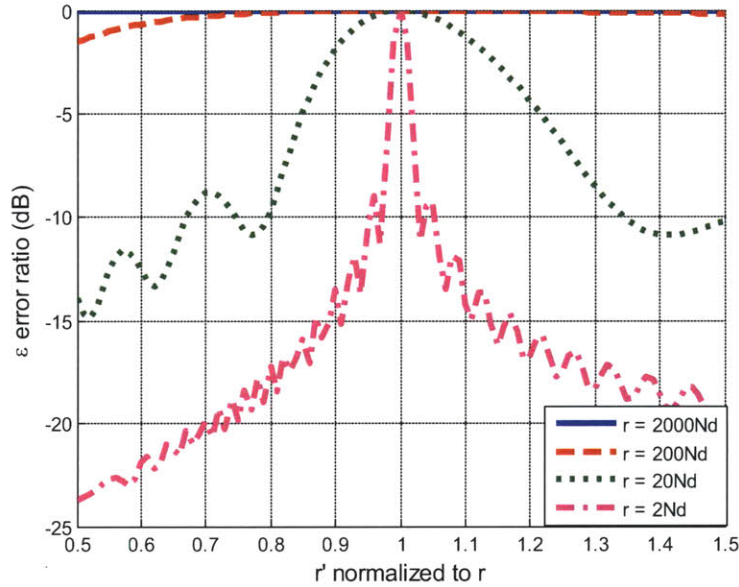
Figure 3-20: Plot of beampattern error ratio versus the estimated source distance r' normalized to true source distance r. The x-axis represents the percentage error between the estimated and true source distance. The y-axis shows the error ratio ϵ which is the deviation from the ideal beampattern when the estimated source distance differs from the true distance. This plot was made for a 1000 element array spaced $\lambda/2$ apart steered towards $\pi/2$.

Since the main lobe's peak height is the first metric that degrades with phase error, we plot the reduction of the main lobe peak against the percentage error with respect to r. Figure 3-20 shows the error ratio between beamforming with a mismatched distance and beamforming with the correct distance. The x-axis is the range finder's estimate r' normalized to r, which represents the true source distance. This is plotted for a 1000 element array with various r while directing the array in the $\pi/2$ direction. For large r equal to 2000Nd, the source is approaching far field, and error in the r' estimation has little effect as can be seen in the error ratio which is nearly constant at 0dB. For decreasing r, main lobe peak becomes increasingly sensitive to mismatch

in r and we observe more rapid degradation of the main lobe for smaller mismatches. For r equal to 20Nd, a $\pm 10\%$ mismatch between r and r' causes degradation of at most -2dB. The array becomes more robust for r of 200Nd and above, as the error ratio is no worse than -0.02dB for $\pm 10\%$ mismatch.

3.5   Summary

This chapter presented a MMW imaging system using digital beamforming with a frame rate of 10fps and a pixel resolution of 6cm at 30m distance, corresponding to 13fs element to element delay. Due to the large aperture size of the array, the array can tolerate more phase noise than we had initially anticipated. The digital phase tightening technique that will be analyzed in Chapter 5 targeted the element-to-element delay, which we acknowledge was an overdesign.

Chapter 4

Millimeter-wave Phase-locked Loop

The MMW imaging system requires the phase information at each antenna receiver
for digital beamforming. Measuring the phase of a MMW signal is achieved by first
down-converting to an IF, which requires a PLL to generate a stable LO. The PLL
is a crucial component of the receiver because it directly impacts the accuracy of the
phase measurement. Any phase noise from the PLL will translate into error in the
phase measurement. Furthermore, each array element will have its own PLL because
distributing a MMW LO across a large antenna array is not practical.

A PLL is a feedback loop that compares and then locks the phase of a voltage
controlled oscillator's (VCO) output signal to the phase of an input reference clock.
Placing frequency dividers in the feedback path allows the output to lock to a multiple
of the reference clock's frequency. This allows for the generation of a MMW LO from
a lower frequency reference. However, designing a MMW PLL is a nontrivial task
that pushes the capability of modern silicon technologies. This chapter will discuss
the design of the MMW PLL and its role in the imaging receiver.

4.1   PLL Overview

This section will provide a brief overview of a PLL's basic operation and its compo-
nents. Figure 4-1 shows the block diagram of a PLL in its most basic form. It is a
feedback loop that consists of a phase detector, a loop filter, and a VCO. A reference

clock operating at frequency f_{ref} provides the input phase θ_{ref} to which the system will lock. Since f_{ref} is a fixed frequency, the input phase θ_{ref} is $2\pi f_{ref}t$ where t is time. The output is a signal generated by the VCO with frequency f_{out} and output phase θ_{out}. The phase detector output v_d is the difference of the two phases (θ_{ref}-θ_{out}) scaled by the phase detector gain K_d. The signal v_d is fed into a loop filter. The loop filter has the transfer function F(s) which controls the dynamics of the loop and produces the control signal v_c for the VCO. The last major component of the PLL is the VCO, whose output frequency f_{out} is the control voltage v_c scaled by the VCO gain K_{vco}. The phase of the VCO output θ_{out} is the integration of f_{out} with respect to time. The feedback tries to minimize the error signal between θ_{out} and θ_{ref}. When this occurs, the output of the VCO is phase-locked to the reference clock. And because frequency is the time derivative of the phase, phase-locking the VCO output to the reference clock implies that the VCO output frequency f_{out} is equal to the reference clock's frequency f_{ref} [40].

4.1.1   Charge-Pump PLL

Figure 4-2 shows an implementation of a standard integer-N charge-pump PLL. This PLL compares the phase of the reference clock θ_{ref} and the phase of the frequency divider output θ_{div}. The feedback locks θ_{div} to θ_{ref}, and this causes the frequencies f_{div} and f_{ref} to be equal. The frequency divider relates f_{div} to the VCO output frequency f_{out} by the divide ratio N. This divide ratio also implies that the phase θ_{div} is equal to θ_{out}/N. When the PLL locks the reference and divider phases together causing f_{div} and f_{ref} to be equal, the VCO will output a signal whose frequency is N times f_{ref}. This PLL is a frequency synthesizer that generates an output frequency that is a multiple of the input reference clock's frequency.

Like the basic PLL shown in Figure 4-1, the integer-N charge-pump PLL contains a phase detector, loop filter, and VCO. However, a phase frequency detector (PFD) is used in place of a simple phase detector. A simple phase detector assumes that its two input signals' frequencies are equal, and it outputs an error signal based on their phase difference. The PFD outputs UP and DN signals, and the effective error
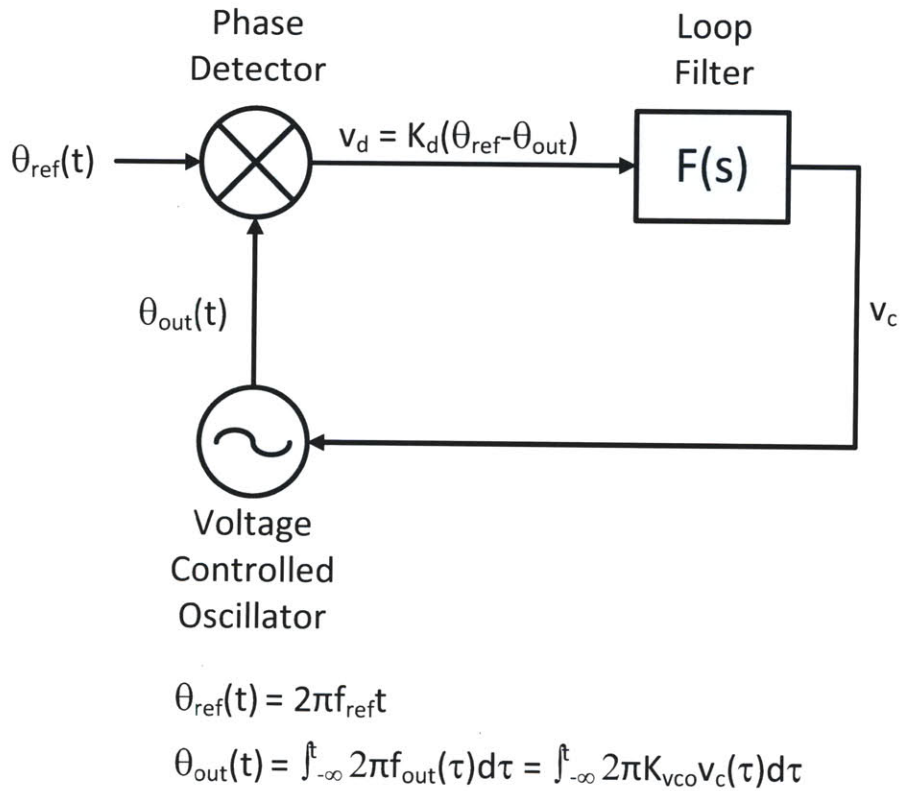
Figure 4-1: Basic block diagram of a PLL.

signal is the difference between UP and DN. The error signal is based not only on the phase relation between the inputs, but also on their relative frequencies. The PFD's frequency detection can be observed by viewing its characteristic plot shown in Figure 4-3, which describes the average error signal versus the phase difference between the reference clock and divider signal. The characteristic plot of the common XOR phase detector is shown for comparison. The PFD's average error signal is always positive for any positive phase differences, which includes positive frequency differences. And the average error signal is always negative for negative phase differences, which includes negative frequency differences. This feature allows the PFD to distinguish positive and negative frequency differences between the reference and divider signals. The phase detector does not have this attribute, and its error signal will not correctly reflect differences in frequency [41] [42].

In addition, a charge-pump is used in conjunction with the phase frequency detec-
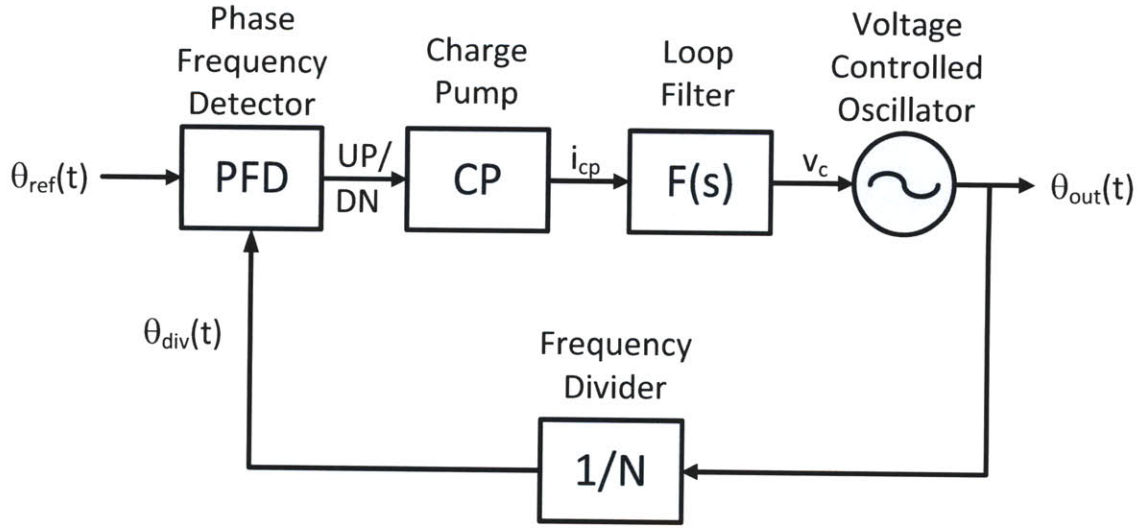
$\theta_{ref}(t) = 2\pi f_{ref}t$

$\theta_{out}(t) = \int_{-\infty}^{t} 2\pi f_{out}(\tau)d\tau = \int_{-\infty}^{t} 2\pi K_{vco}v_c(\tau)d\tau$

$\theta_{div}(t) = \theta_{out}(t)/N = \int_{-\infty}^{t} 2\pi f_{div}(\tau)d\tau$

Figure 4-2: Block diagram of a PLL with charge pump and frequency divider.

tor to produce the input to the loop filter. The charge-pump consists of a switched current source and switched current sink that are toggled by the UP and DN signals. The charge-pump current and the duration that each switch is closed determines the amount of charge that is sent to the loop filter.

From the block diagram, we derive the closed-loop transfer function between the output and input phases. G(s) is the forward gain, and f is the feedback factor.

$$H(s) = \frac{\theta_{out}(t)}{\theta_{ref}(t)} = \frac{G(s)}{1 + G(s)} = \frac{K_d I_{cp} K_{vco} F(s)}{s + K_d I_{cp} K_{vco} F(s)(1/N)} \qquad (4.1)$$

4.1.2   PLL Phase Noise

The transfer function from Equation 4.1 describes the PLL's frequency response and its dynamic behavior. For the imaging receiver, the PLL is used to generate an LO that will run indefinitely. It is not used to modulate or track any signal. Therefore, the dynamic behavior of the PLL expressed in this equation is less of a concern. However, phase noise is critical in the system since it corrupts the phase information
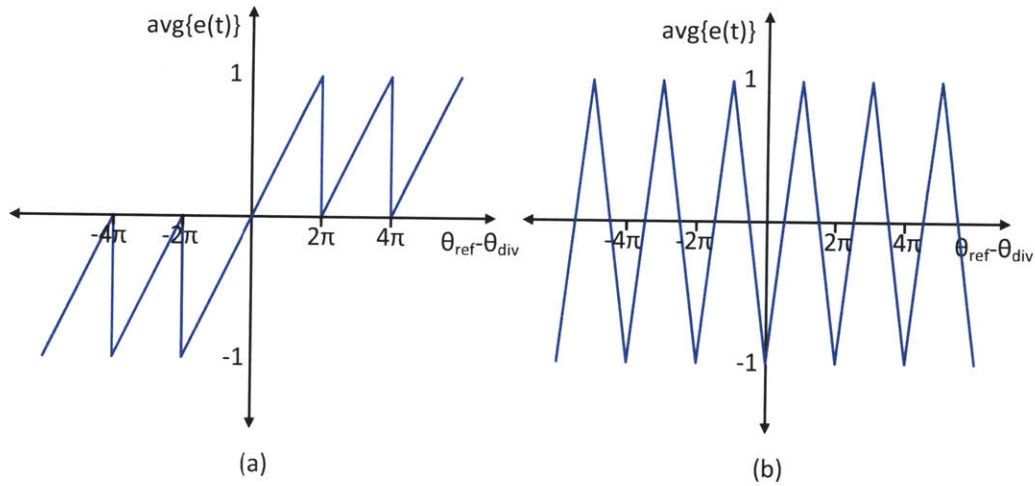
Figure 4-3: The transfer characteristics between the average error signal e(t) and the phase difference between reference and divider signals for (a) a PFD and (b) XOR phase detector.

used in beamforming, so we are interested in the transfer functions for the phase noise sources in the PLL.

These sources include the reference clock and divider jitter, the charge pump noise, and the reference clock feedthrough spur. Any jitter from the reference clock or frequency divider will cause errors in the phase detector's error signal. Noise from the active devices in the charge pump causes fluctuations in the VCO's control signal and translates into phase noise. Feedthrough of the reference clock onto the VCO control signal causes a reference clock spur in the PLL's output spectrum. The noise from these components can be referred to the phase detector output and would then share the same transfer function H(s) and is shown in Equation 4.2. These noise sources experience a low-pass characteristic in which the bandwidth is set by the loop filter.

$$\frac{\theta_o(t)}{\theta_{n,pfd}(t)} = H(s) = \frac{G(s)}{1 + G(s)f} = \frac{K_d I_{cp} K_{vco} F(s)}{s + K_d I_{cp} K_{vco} F(s)(1/N)} \tag{4.2}$$

The VCO is another important source of noise. Although much research has been devoted to understanding the phase noise of a VCO, we will use a simple linear time-invariant model to gain a basic intuition of where this noise comes from. For

63

an LC tank VCO, the current noise generated by the transistors sees the impedance of the LC tank. The output noise spectrum can be calculated by multiplying the noise power density with the square of the tank's effective impedance. From the equipartition theorem of thermodynamics, half of this noise translates into phase noise while the other half becomes amplitude noise. The noise spectrum is frequency dependent due to the filtering of the LC tank and will roll off as the frequency moves further away from tank's resonant frequency. In order to obtain lower noise density, a higher quality factor (Q) for the tank is required. This becomes increasingly difficult at MMW frequencies as devices become more lossy and high Q passive devices are harder to implement [43].

VCO phase noise occurs at the output of the PLL and has the transfer function shown in Equation 4.3. The transfer function has a high-pass characteristic. Outside of the bandwidth set by the loop filter, the VCO noise is passed, but within the bandwidth, the VCO noise is attenuated.

$$\frac{\theta_o(t)}{\theta_{n,vco}(t)} = \frac{1}{1 + G(s)f} = \frac{s}{s + K_d I_{cp} K_{vco} F(s)(1/N)} \tag{4.3}$$

The order of the loop filter transfer function F(s) determines the roll off of the transfer function H(s). In [44], Demir analyzes the effect of white noise and flicker noise driving the VCO in a PLL. He finds that over time 1/f noise causes unbounded jitter variance unless a second order loop filter is used. Having unbounded jitter means the PLL is not actually locked. Continually increasing jitter, when mixed with the carrier signal, will cause increasing phase error of the IF signal. This motivates the requirement for a second order loop filter in this design because 1/f noise is inherent to real devices and unbounded jitter cannot be tolerated.

4.2 Prior Published Work

4.2.1 Silicon Technologies

There have been a number of MMW frequency synthesizers that have been reported in the literature. PLLs at 60GHz have been designed in both BiCMOS and CMOS technologies for intended use in wireless communications.

A 60GHz PLL was presented by Hoshino et al. using 90nm CMOS. The architecture was an integer-N divide-by-1024. The VCO consisted of a cross-coupled core with transmission lines for inductance. A ring oscillator based injection-locked frequency divider which consumed high power at higher fequency but removed the need for an inductor which saved area. The PLL's locking range was 61.1 to 63.1GHz. It had an output power of greater than -7dBm and a power dissipation of 78mW. The phase noise was -80dBc/Hz at 1MHz offset [45].

A 60GHz PLL has also been demonstrated at 45nm. This was a quadrature design which had a wide bandwidth of 9GHz that was achieved by using two separate oscillators which relaxed the tunability for each one. Each quadrature VCO consisted of two parallel LC VCOs. A binary weighted bank of varactors was used for coarse tuning and analog control was used for fine tuning. This PLL has a power consumption of 76mW. It has a phase noise of -75dBc/Hz at 1MHz offset [46].

77GHz PLLs for automotive applications have been reported. Lee et al. presented an intenger-N divide-by-32 PLL in 90nm CMOS technology. Using a LC cross-coupled VCO with $3\lambda/4$ transmission lines to distribute the load and boost the oscillation frequency. The PLL's locking range was 73.4 to 73.72GHz. It consumed 88mW and had an output power of -28dBm at the spectrum analyzer without accounting for losses. It achieved a phase noise of -88dBc/Hz at 100kHz offset [47].

A 77GHz PLL was demonstrated in 0.18μm silicon-germanium (SiGe) BiCMOS. The design consisted of a LC emitter degenerated Colpitts oscillator with resistive biasing for lower phase noise contribution. An injection-locked frequency divider was used for high operating frequency and low phase noise. The locking range was 75.67 to 75.8GHz with -17.8dBm output power after de-embedding losses while dissipating

75mW. The phase noise was -103.5dBc/Hz at 1MHz offset [48].

4.2.2 Alternative Technologies

Integrated MMW PLLs have also been designed using alternative technologies to silicon. Although it is difficult to make a fair comparison between these PLLs to silicon-based designs, it is useful to see the performance difference between the technologies.

In [49], a PLL operating at V-band was designed using $0.15\mu m$ GaAs pHEMT technology, which has an ft of 90GHz and an fmax of 160GHz. The output frequency was 60GHz and the reference clock was 5GHz. The PLL had a divide-by-3 frequency divider but also used a fourth sub-harmonic mixer as a phase detector to achieve an effective divide ratio of 12. The measured PLL had a locking range of 840MHz around a center frequency of 60.1GHz. The output power was 0dBm, and the total power dissipation was 370mW. The free-running VCO had a phase noise of -87.7dBc/Hz at 1MHz offset, and the locked PLL had phase noise of -112.4dBc/Hz at 1MHz offset.

4.3   System Phase Noise Analysis

At MMW frequencies, estimating the phase is a major challenge, and the phase noise contribution from each component in the receiver must be considered. Rather than estimate the phase at the carrier frequency, we first mix down to an intermediate frequency which will scale up the phase information and ease the requirements on the phase estimation. When the input signal is mixed with the LO generated by the PLL, the noise from the mixer, LNA, and PLL are added at the IF.

For imaging applications, we expect a single tone input as opposed to a broadband signal from traditional communications systems. This simplifies the noise analysis and allows us to estimate the timing jitter contributed by the LNA, mixer, and PLL before the signal is digitized.

From the work presented in [33], the LNA for this system has a gain of 20dB and a noise figure of 6dB at 77GHz. The bandwidth is approximately 5GHz. We can calcu-

late the increase in noise floor caused by this circuit. The noise floor is -174dBm/Hz and is defined as kTB where k is Boltmann's constant, T is room temperature of 290K, and B is the bandwidth.

$$P_{Noise} = -174 + Gain + NoiseFigure + 10log_{10}(Bandwidth)[dBm] \qquad (4.4)$$

Using Equation 4.4, we determine the noise power to be -51dBm or 7.92nW. For a 50Ω impedance, we estimate the RMS noise voltage to be 0.628mV. Assuming the noise is applied equally over the period of the sinusoid input, we divide the voltage noise by the expected slope at the zero-crossing to obtain an estimate of the timing jitter caused by the LNA. We assume that the LNA and mixer are operating at their 1dB compression point, which is an optimistic approximation but should give us an idea of the expected slope. The 1dB compression point at the mixer output is 2dBm, and the mixer has a conversion gain of 26dB and a noise figure of 14dB. The output of the LNA is then -24dBm which corresponds to an amplitude of 0.014V.

$$Slope = \frac{d}{dt}A \cdot sin(\omega t)\bigg|_{t=0} = A\omega \cdot cos(\omega t)\bigg|_{t=0} = A\omega \qquad (4.5)$$

For a 77GHz tone with amplitude 0.014V, the slope is 6.77GV/s. We then calculate the jitter as follows.

$$t_{jitter} = V_{n,RMS}/Slope \qquad (4.6)$$

In this case, the jitter was approximately 90fs at the 77GHz carrier frequency. Assuming an IF of 1GHz, we scale this value to obtain 6.9ps of jitter at the IF.

We use a similar method for the mixer which has a conversion gain of 26dB and a noise figure of 14dB that is outputting at its 1dB compression point of 2dBm. We assume the same 5GHz bandwidth of noise from the LNA has been folded down at the IF of 1GHz. Substituting in these values gives an RMS noise voltage 3.15mV and a slope of 1.77GV/s. The jitter is then calculated to be 1.8ps at the IF.

Based on the phase noise measurements from the MMW PLLs reported in the previous sections, we expect the PLL to have significantly more jitter than the contributions from these blocks. We will see in Section 4.6 that the PLL contributes about 1ps of jitter at the carrier. For a 1GHz IF, this jitter will scale to 77ps for a 1GHz IF which is significantly more than the jitter from the LNA or the mixer. We require that the system, particularly the PLL, has low jitter because jitter will corrupt the signal sent to the beamformer and degrade the image quality.

4.4 PLL Architecture

The objective of this PLL design is to obtain low phase noise while maintaing low power consumption for use in the imaging array system. Although power dissipation is a concern, we still require sufficient output power to drive the mixer load.

The PLL design uses an integer-N architecture shown in Figure 4-4. The reference clock was designed for a 148MHz crystal oscillator with an output frequency of 76GHz.[1] The feedback path consists of a frequency divider with divide ratio of 512. A tri-state phase frequency detector compares the reference clock and frequency divided output of the VCO and toggles the current source and current sink in the charge pump. A passive second-order loop filter controls the bandwidth and stability of the feedback loop.

4.4.1 Voltage Controlled Oscillator

The schematic of the VCO is shown in Figure 4-5. The VCO core is a cross-coupled LC tank oscillator based on the design presented in [33]. Capacitive coupling is used in the oscillator's feedback path. These capacitors act as capacitive dividers with the C_π of the core transistors and allow for higher operating frequency and output power. Due to the capacitors in feedback, the transistor base voltages could be set independently and were biased externally with resistors. Transmission lines

---

[1]The imaging system shown in Figure 3-12 was updated to a 150MHz reference clock and a 76.8GHz LO after analysis of the Digital Phase Tightening System in Chapter 5, which occured after the design of this PLL.
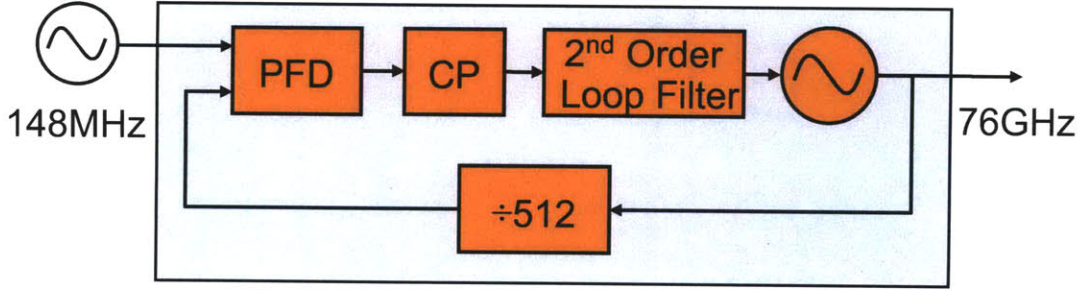
Figure 4-4: Block diagram of PLL.

were used to implement the inductance of the core. The varactors were designed as a series connection between the process's accumulation mode MOS capacitor and metal-insulator-metal (MIM) capacitors with the internal node resistively biased. Since the VCO is very sensitive to parasitic capacitances and inductances which can shift the center frequency, locking is a concern. By allowing one terminal of the varactor to be independently biased, we can set that voltage to maximize the capacitance range of the MOS capacitor for a given voltage range of the charge pump and passive loop filter, increasing the locking range.

The VCO core is followed by a buffer stage that consists of an emitter follower and an emitter-coupled pair. Transmission lines were used for inductive peaking and to maximize output power to a 50Ω load. The buffer stage has a separate supply and is biased independently from the core to allow for better control of the output voltage.

4.4.2 Frequency Divider Chain

The divider chain was constructed with nine divide-by-2 static frequency dividers, and the block diagram of one divider is illustrated in Figure 4-6. The divider consists of two latches in a master-slave D flip-flop configuration, which is then placed in negative feedback. The signals out_p and out_n are complementary to one another. In one clock cycle of in_p, the current values of out_p and out_n will propagate through the latches, and output values will flip due to the negative feedback. In the next
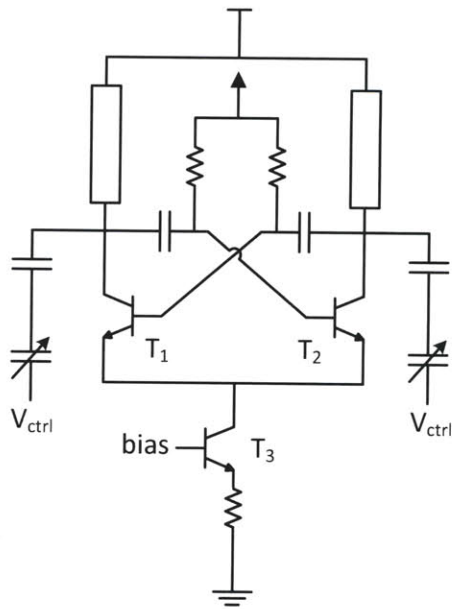
Figure 4-5: Schematic of cross-coupled VCO with capacitor feedback.

cycle of in_p, the values will flip again, returning them to their original state. It takes two cycles of the inputs for the output to complete one cycle, which performs the frequency divide operation. This architecture using a D flip-flop is known as the static frequency divider. Alternative topologies include the injection-locked frequency divider and regenerative frequency divider, which only require one latch or mixer but have much narrower locking ranges. Static frequency dividers were chosen for their wide bandwidth operation and robustness which came at the cost of increased power dissipation.
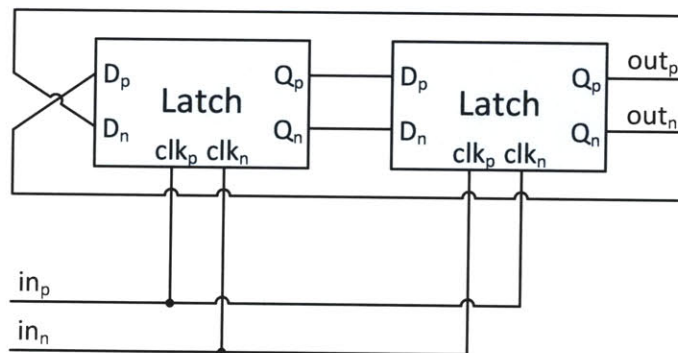
Figure 4-6: Block diagram of D flip-flop used as divide-by-2 static frequency divider.

The first seven stages were created in emitter-coupled logic and the last two were designed using CMOS devices. The highest frequency divider utilizes inductive peaking for increased operating frequency. The inductors are planar spiral inductors created from the topmost analog metal layer. The inductors were simulated using RFDE Momentum [50] and were sized with consideration given to the mutual inductance due to their close proximity to each other. Figure 4-7 shows the circuit diagram of the latch used in the first divider stage. Each divider stage was sized separately to optimize for power consumption, and the cascaded ECL divider stages were capacitively coupled.
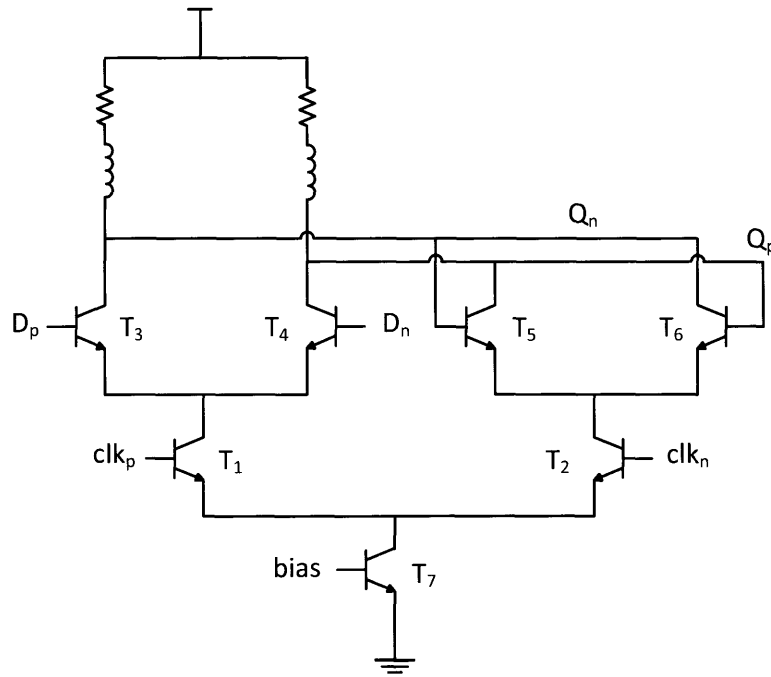
Figure 4-7: Schematic of the ECL latch with inductive peaking used in first frequency divider stage.

An ECL-to-CMOS conversion circuit is used to transition between the logic levels in the divider chain. This circuit consists of a differential pair with current mirror load using MOSFET devices, which converts the differential ECL signal into a single-ended CMOS logic signal. Then, a single-ended-to-differential circuit was employed to generate complementary signals for the CMOS static frequency dividers. This was achieved by matching the delays of a transmission gate and an inverter followed by

71

cross-coupling transistors to help align the transitioning edges.

The frequency dividers using CMOS devices were composed of the latches shown in Figure 4-8. The latches used source-coupled logic, and the PMOS devices provided a resistive load. This topology has inherently complementary outputs, unlike a purely CMOS frequency divider which does not have truly complementary outputs and would have an inverter delay between the differential outputs. This improves the function of subsequent divider stages which require differential outputs. However, for low operating frequencies, purely CMOS frequency dividers may have been more suitable and would have reduced power consumption by 3.4mW.
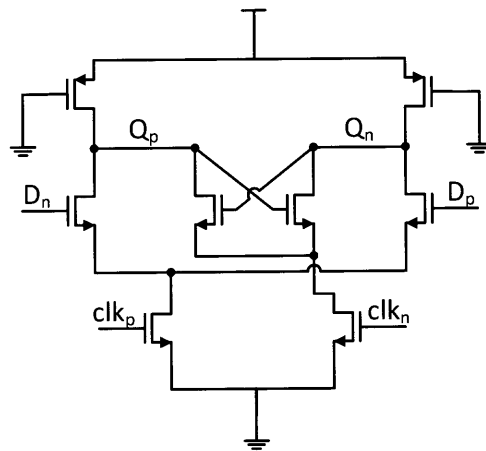
Figure 4-8: Schematic of the source-coupled logic latch used in the final two stages of the divider chain.

4.4.3 Phase Frequency Detector/Charge Pump/Loop Filter

A standard tri-state phase frequency detector was used to compare the divider output and reference clock at 148MHz. It is shown in Figure 4-9. In the initial state, REF, DIV, UP, and DN are low, and both UP_b and DN_b are high. When the PFD sees a rising edge from REF, the signal UP_b will be pulled low causing UP to be toggled high. The circuit will stay in this state until a rising edge from DIV is detected. When this occurs, DN_b will be pulled low. With both UP_b and DN_b being low, the NOR gate output RST toggles high and resets the circuit causing UP and DN to

both return low. The low-to-high transition of DIV will cause the DN signal to be momentarily high before the reset forces it low again. Similarly, a rising transition in DIV during the initial state will cause DN to be high before a rising transition in REF causes the RST signal to force both UP and DN to be low again.
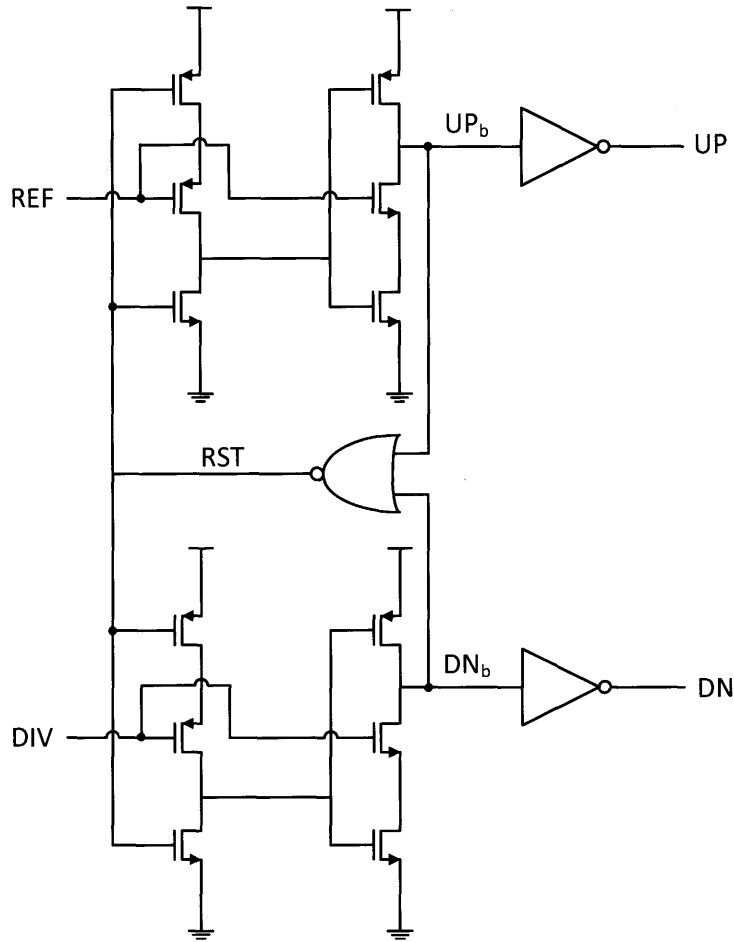
Figure 4-9: Schematic of phase detector.

The charge pump is a single-ended design based on work presented in [51]. The schematic is shown in Figure 4-10. Transistors M1 and M4 act as switches placed at the source nodes of the current source transistor M2 and current sink transistor M3. The up and down currents are designed for 1mA. In this design, single transistors rather than cascodes were used to allow for a larger output swing to drive the VCO's varactors. This was a design decision that reduced the linearity of the charge pump but was chosen to improve robustness by allowing a larger tuning range for the VCO.

Transistors Mc1 and Mc2 act as capacitors to hold the bias nodes of the current sources stable while the switches were toggled by the UP and DN control signals. Both capacitors were designed to each be 5pF.
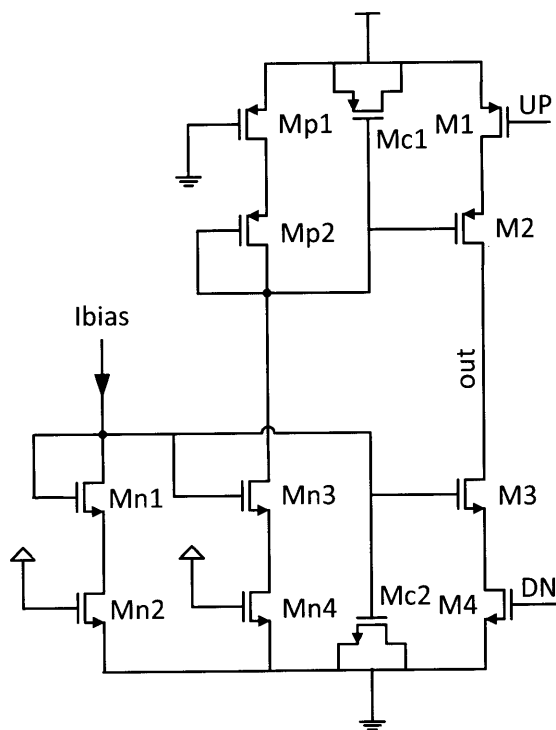
Figure 4-10: Schematic of PLL charge pump.

The phase detector, charge pump, and CMOS divider stages were implemented using the process's thick oxide transistors, which allowed these circuits to operate with the same voltage supply level as the VCO and ECL dividers of 2.5V. This simplified the ECL to CMOS transition in the divider chain, and also increased the voltage range of the charge pump.

4.4.4 Loop Filter

As described in Section 4.1.2, a second-order loop filter is required to ensure that the timing jitter is bounded when there is flicker noise in the system. A passive second-order loop filter was implemented on chip. The bandwidth of the loop filter was set to be 15MHz to balance the noise between the charge pump and VCO. CppSim and

its ancillary PLL Design Assistant were employed to determine the system dynamics and loop filter characteristics [52]. The loop filter's pole was designed for 23.5MHz, and the zero was designed for 1.875MHz. The PLL was designed for a phase margin of 55.5°.
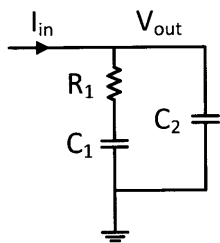
Figure 4-11: Schematic of second-order passive loop filter.

The passive loop filter shown in Figure 4-11 has the following transfer function.

$$\frac{V_{out}}{I_{in}} = \frac{1 + sR_1C_1}{s(C_1 + C_2)(1 + s\frac{C_1C_2}{C_1+C_2}R_1)} \tag{4.7}$$

In order to achieve the desired pole and zero locations with appropriate gain, the component values were calculated to be $6.9k\Omega$ for R_1, 15pF for C_1, and 1.08pF for C_2.

4.5 Simulation Results

High-level full closed-loop simulations were performed using CppSim. Verilog-A and SpectreRF were used for simulation of individual blocks and smaller transient closed loop simulations.

The transmission lines and interconnects in the VCO core were simulated using a 2.5D EM simulator, RFDE Momentum. The inductors in the first stage frequency divider were also simulated in RFDE Momentum. Due to challenges in incorporating the generated S-parameter models for transient analysis in this simulation environment, the critical components were replicated by using the design kit transmission line models and adjusting their parameters until the S-parameters matched the simu-

lated ones from Momentum. These replicated models were then substituted into the extracted design for more accurate simulations.

The VCO has a simulated output frequency range of 72.7GHz to 76.4GHz. The output power of the PLL is -3dBm. The VCO core consumes 18mW while the VCO buffer consumes 34mW from a 2.5V supply. The remaining PLL circuitry consumes an additional 45mW. The total simulated power dissipation is 97mW. Figure 4-12 shows the simulated phase noise spectrum using CppSim. The phase noise was plotted using simulated noise values of critical blocks from Spectre and inputting these values into the CppSim simulator. At 1MHz offset from the carrier, the phase noise is simulated to be about -85dBc/Hz.
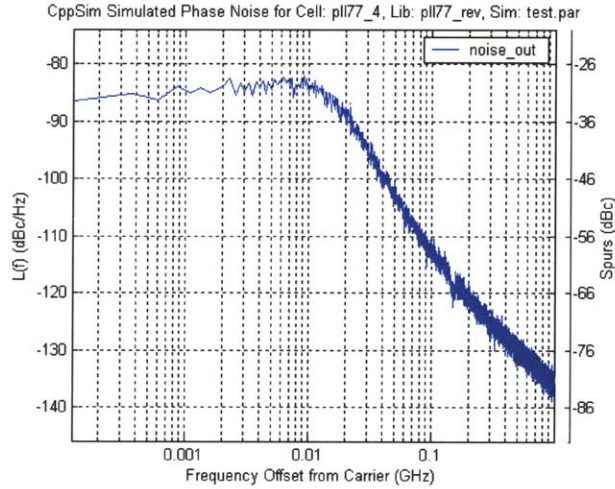
Figure 4-12: Phase noise simulation of PLL using CppSim.

Phase noise causes variations in the period timing jitter, which is the time difference between ideal period and the actual period of the oscillator signal. The jitter is represented by an RMS time value, $J_{RMS,PER}$. We represent the phase noise spectrum as L(f), where f is the frequency offset from the carrier frequency. The relationship between the RMS period jitter and phase noise is shown in Equation 4.8 [53].

$$J_{RMS,PER} = \frac{1}{2\pi f_c}\sqrt{2\int_{f_1}^{f_2} 10^{\frac{L(f)}{10}}\,df} \tag{4.8}$$

The integration of the phase noise spectrum L(f) inside the square root computes

76

the RMS phase error. Since L(f) represents the phase noise on one side of the carrier, we assume that the other side is symmetric and scale the integration of L(f) by two. The limits of integration, f_1 and f_2, depend on the frequencies of interest. In this system, we are interested in phase noise down to the frame rate of 10fps for the lower limit. But for practical purposes regarding simulation time, we used 100kHz as the lower bandwidth. This is a reasonable approximation because integrating this profile lower than 100kHz adds very little contribution to the timing jitter (< 2fs) because the frequency range is small compared to the rest of integration. We choose the upper frequency limit to be 1GHz which is 2 orders of magnitude higher than the loop bandwidth and sufficiently captures out-of-band phase noise contribution. Integrating phase noise higher than 1GHz adds negligible jitter (< 1fs) because the phase noise level is so low. The RMS phase error is then divided by $2\pi f_c$, which converts the RMS phase error into RMS timing jitter. The PLL Design Assistant from CppSim was used to calculate the jitter from this phase noise spectrum and gave a timing jitter of 830fs.

4.6   Measurement Results

Figure 4-13 is a die photo of the PLL designed in IBM's 0.13μm SiGe BiCMOS process. Particular care was placed in the layout of the VCO core to ensure symmetry and to minimize parasitic capacitances and inductances. Via resistances connecting between the devices and the top analog metal used for transmission lines were also considered and minimized. Traces connecting the VCO output and the first stage frequency divider were designed to be as symmetrical as possible to ensure proper phase relation. This was also true for the differential traces along the divider chain. The area of the PLL excluding bondpads is 0.6mm by 1.4mm.

The test measurement setup is shown in Figure 4-14. Testing was performed at a probe station where high-frequency ground-signal-ground (GSG) probes could directly probe the differential outputs of the PLL. Measurements were taken using an Agilent E4440A PSA Series Spectrum Analyzer with the Agilent 11970W Waveguide
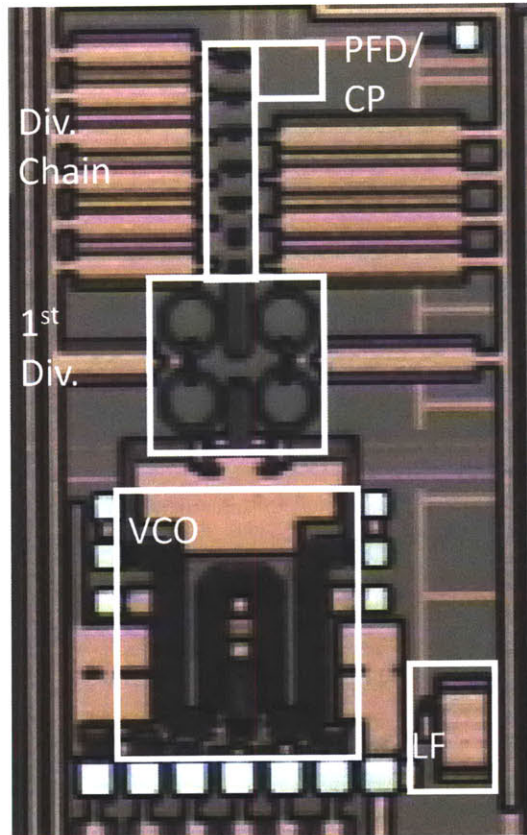
Figure 4-13: Die photo of PLL.
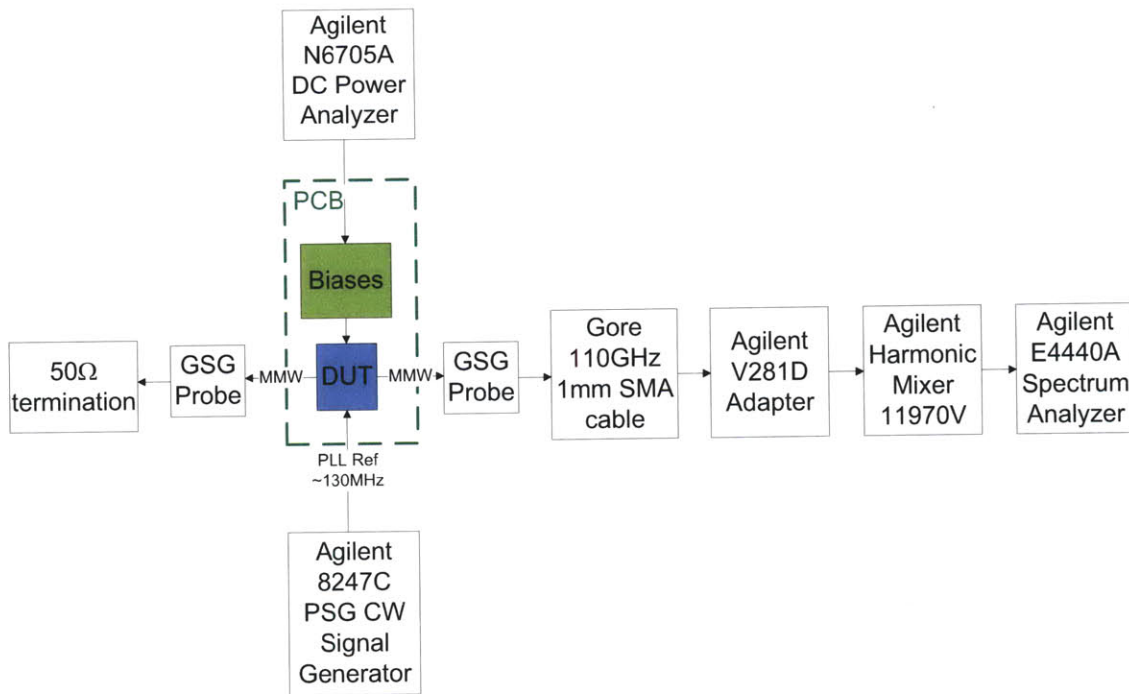
Figure 4-14: Block diagram of test measurement setup for PLL.

Harmonic Mixer. The dice were packaged in an open cavity QFN to minimize lead inductance for the DC signals while allowing for direct probing of the high frequency signals. This packaging also allowed for easier swapping of dies that are prone to damage due to probing. Voltage and current sources were generated externally via circuitry on a printed circuit board. Particular care was taken to bypass the supplies and bias signals.

The operating range of the PLL is 64.25 to 68.84GHz. Due to the change in frequency range, the reference clock is driven by an Agilent 8247C signal generator. The measured output spectrum of the PLL is shown in Figure 4-15. The PLL is locked at 66.9GHz with an output power of -13.5dBm at the spectrum analyzer. The reference spurs and their harmonics are visible. The reference spur, located at an offset of 130.6MHz from the carrier, was measured to be -33.8dBc.

Figure 4-16 shows phase noise measurement taken with a carrier frequency of 66.9GHz. This result is a single-ended measurement taken from one of the differential outputs. The phase noise is -80.99dBc/Hz at 1MHz offset. We integrate the phase
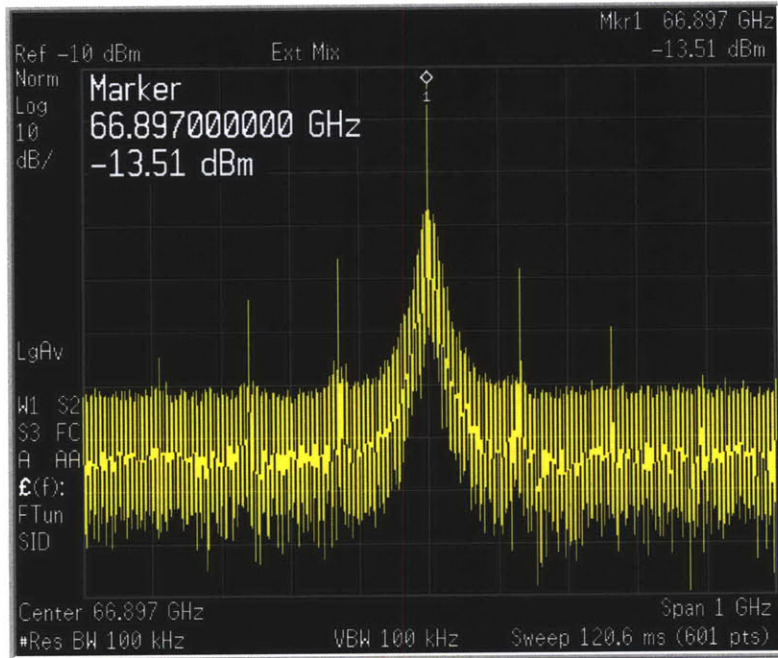
Figure 4-15: Measured output spectrum of the PLL.
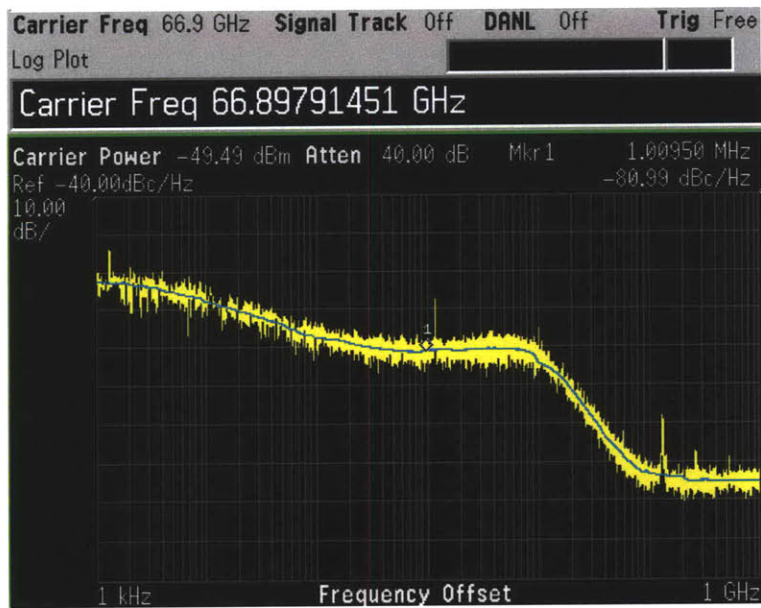
Figure 4-16: Measured phase noise of the PLL.

noise to produce a timing jitter of approximately 1.0ps. The reference spur at 130MHz can be observed in the figure, although the level appears lower due to the resolution of the phase noise measurement. The spur at around 1.1MHz was most likely due to an AM radio broadcasting station that was operating near the test measurement laboratory [54].

The measured output power is -9.5dBm for a single ended output. Cables and waveguides account for an additional 7.5dB of loss. Accounting for a differential output, the total output power is 1dBm. The VCO core and buffer dissipate 60.5mW. The remainder of the PLL consumes 46.1mW yielding a total power dissipation of 106.6mW. The power consumption is larger than simulated because the output buffer was boosted to provide enough power for accurate phase noise measurements from the spectrum analyzer.

The operating frequency is lower than simulated. This is due to larger than expected parasitics from the components and interconnects in the varactors and cross-coupling capacitors near the operating frequencies. Most of the rigorous EM simulations were focused on the transmission lines of the VCO core rather than these capacitors. The routing to the cross-coupling capacitors and the vias required to reach the MIM capacitors in the varactors contributed significantly to reducing the oscillation frequency of the VCO. These nodes are highlighted red in the schematic and layout in Figure 4-17. Accounting for these additional parasitics gave a simulated oscillation frequency of 64.7 to 67.5GHz.

4.7   Comparison to Previously Published PLLs

Table 4.1 compares this PLL with previously published silicon-based PLLs operating around 60 to 70GHz. This work is comparable to most of these PLLs in terms of performance. However, the power dissipation is noticeably higher, and this was the result of tradeoffs that were made to have higher output power and increased robustness. The VCO output buffers drew a lot of current in order to provide sufficient power to drive a mixer load. Static frequency dividers were used to have wider locking
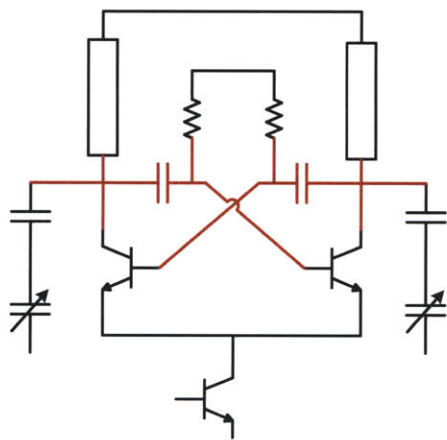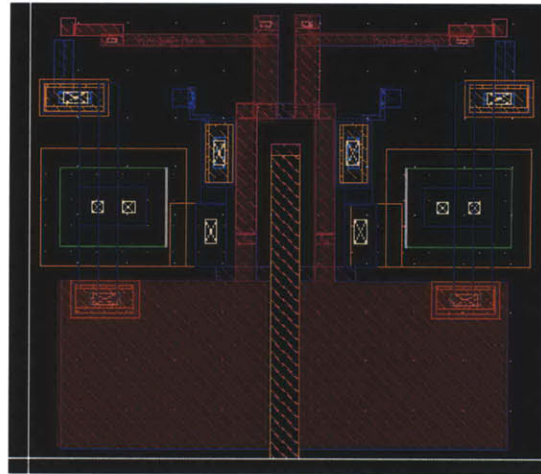
(a)   (b)

Figure 4-17: The key nodes that contributed to the reduction in oscillation frequency were the cross-coupling capacitors and their interconnects. These nodes are highlighted in red in the schematic (a) and the layout (b).

ranges to account for shifts in frequency due to unexpected parasitics. The static frequency dividers have two current branches as opposed to alternatives, such as regenerative or injection-locked frequency dividers that consume less power but have smaller locking ranges. Furthermore, thick oxide MOSFETs were used to allow the same Vdd voltage level between the MOSFET and bipolar transistors. This increased robustness by allowing the two transistor types to interface directly which each other and removed the need for level shifting circuits. However, this came at the cost of increased power consumption by the MOSFETs.

4.8 Summary

This chapter described the design and characterization of a MMW PLL that could be used to generate the LO for each receiver in the imaging system. The PLL was designed for low phase noise and power consumption but with sufficient output power to drive a mixer load. Due to unexpected parasitics, the center frequency was lower than the target of 76GHz. However, we located the source of the problem, and a future

	C. Lee et al, ISSCC '07	J. Lee et al, JSSC '08	Tsai et al, RFIC '08	Hoshino et al, IEICE '09	Jain et al, JSSC '09	This work
Process	90nm CMOS	90nm CMOS	0.13um CMOS	90nm CMOS	0.18um BiCMOS	0.13um BiCMOS
Operating Freq.	58 to 60.4GHz	73.4 to 73.72GHz	64.5 to 66.2GHz	61.1 to 63.1GHz	75.67 to 78.5GHz	64.25 to 68.84GHz
Output Power	N/A	N/A	N/A	> -7dBm	-17.8dBm	1dBm
Phase Noise at 1MHz	-85.1 dBc/Hz	-88 dBc/Hz at 100kHz offset	-84.1 dBc/Hz	-80 dBc/Hz	-103.5 dBc/Hz	-81 dBc/Hz
Power Consumption	80mW	88mW w/o output buffers	72mW w/o output buffers	78mW	75mW	106.6mW

Table 4.1: Table comparing this work with previously published silicon-based PLLs.

iteration could be designed with improved modeling of the critical interconnects taken into account. We determined that the PLL is the dominant source of phase noise in the receiver, and this PLL gives a reference for the amount of timing jitter that will be mixed into the received signal at each antenna receiver. From the measured phase noise profile, we found that the PLL will contribute 1ps of jitter with a bandwidth of 10MHz at the carrier.

Chapter 5

Digital Phase Tightening

A major challenge for digital beamforming in the imaging system is measuring the phase of the received MMW signal. For a 77GHz carrier frequency, the period is only 13ps which is smaller than a single inverter delay in the $0.13\mu m$ BiCMOS process. Furthermore, phase noise from the analog circuits corrupts the signal and causes error in the phase measurement. We use a technique that we call digital phase tightening, which leverages the large ratio between the MMW carrier signal and the slow operating frame rate to estimate the phase and reduce the phase error.

5.1 Description

We model the received input signal, $x(t)$, as a narrow-band signal at the carrier frequency ω_c and a time delay τ expressed in Equation 5.1. We want to estimate the signal phase Δ, equivalent to $\omega_c\tau$, relative to the reference clock. The term $\varphi(t)$ is zero-mean phase noise that has been added to the signal by the front-end analog circuitry.

$$x(t) = exp\left(j\left[\omega_c\left(t + \tau\right) + \varphi(t)\right]\right) \tag{5.1}$$

We first mix the MMW signal with a local oscillator (LO) operating at ω_{LO} to an intermediate frequency ω_{IF}. The result $y(t)$ is expressed in Equation 5.2. The
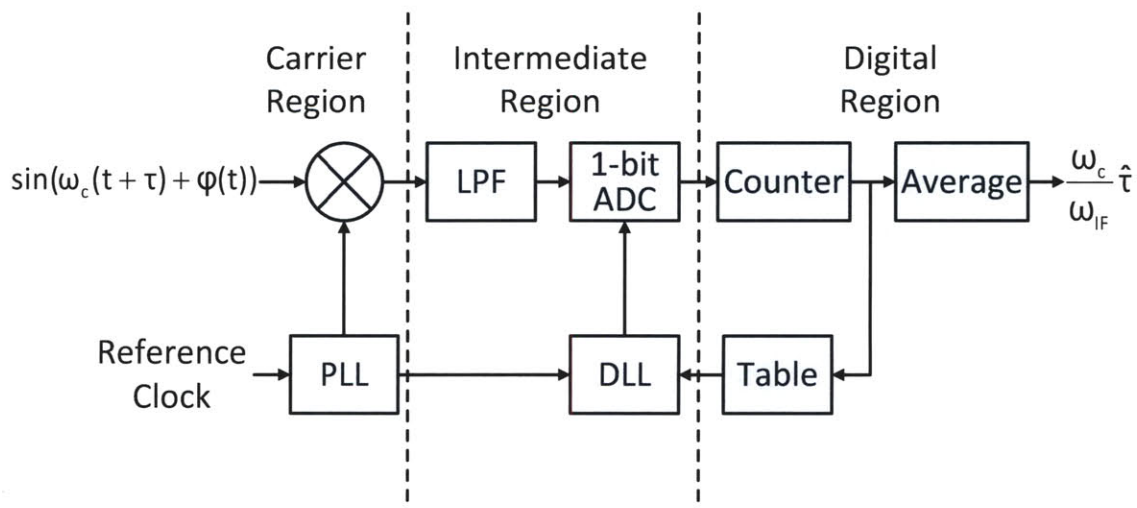
Figure 5-1: Block diagram of digital phase tightening system.

time delay τ has been scaled up by ω_c/ω_{IF}, which is more manageable and easier to estimate. However, this benefit comes at the cost of increased noise, which is introduced by the mixer and imperfect LO and is represented as $\zeta(t)$.

$$y(t) = exp\left(j\left[\omega_{IF}\left(t + \frac{\omega_c}{\omega_{IF}}\tau\right) + \varphi(t) + \zeta(t)\right]\right) \tag{5.2}$$

The IF signal is sent to a feedback loop that consists of an analog-to-digital converter (ADC), a delay-locked loop (DLL), and some digital logic. By locking to the DLL clock, the DLL generates evenly-spaced rising edges of the clock signal throughout its period. Based on the output of the ADC, the digital logic selects the rising edge from the DLL that triggers the ADC near the IF signal's zero crossings. The digital control signal represents the phase of the input signal relative to the DLL clock. Assuming an IF of about 100MHz, we can oversample from the IF to the frame rate to capture about one million samples, which are averaged in the digital domain. This reduces the zero-mean noises by the square root of the number of samples and essentially tightens our estimate to the true phase of the received signal.

86
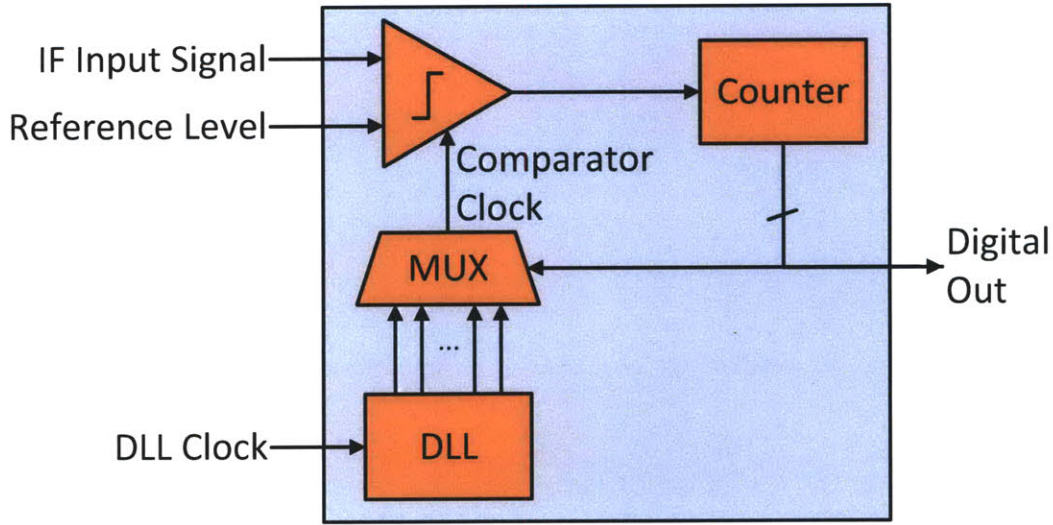
Figure 5-2: Block diagram of chip implementation of phase tightening system.

5.2   Implementation

A test chip was designed and fabricated using the CMOS devices in IBM's 0.13μm BiCMOS8HP process. The block diagram of the circuits in the implemented system is presented in Figure 5-2. These blocks operate after the received signal has been mixed down to the IF. Since we focus on the phase and not the amplitude, the ADC has been replaced with a simple comparator to determine the zero-crossings by checking whether the IF input signal is above or below the zero reference level. The digital logic, which consists of a counter, either increments or decrements based on the output of the comparator. The counter controls a MUX which selects one of the delay taps from the DLL to be the comparator clock. For this implementation, the DLL clock and input signal are the same frequency, and the system locks onto the rising zero-crossings of the input signal.

5.2.1   Delay-locked Loop

Figure 5-3 shows a DLL, which is composed of a variable delay-line in a feedback loop with a phase detector, charge pump, and loop filter. A DLL is similar to a PLL except a delay-line is used in place of a VCO, and a reference clock drives the
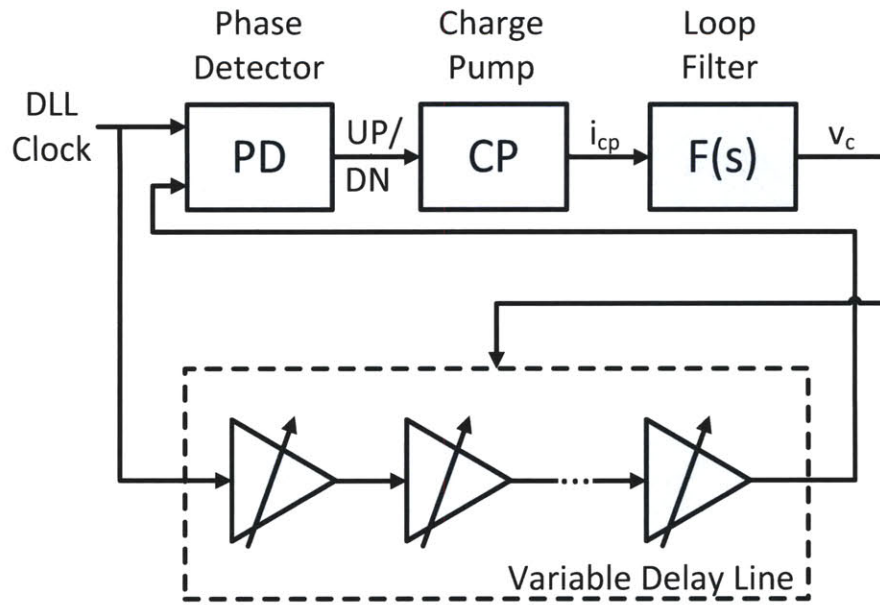
Figure 5-3: Block diagram of DLL.

input of the delay-line. A phase detector compares the edges from the reference clock input and the output from the delay-line. The same tristate phase detector topology described in Section 4.4.3 for the PLL was used in this DLL design. It outputs UP and DN signals to control the amount of current from the charge pump that is driven into the loop filter. The schematic of the charge pump is shown in Figure 5-4. The charge pump was biased at 0.5mA. The charge pump has switches M1 and M6 at the sources of a cascoded current source (M2 and M3) and a cascoded current sink (M4 and M5). Unlike the MMW PLL's charge pump which required a large tuning range for increased robustness, this charge pump was designed with cascodes for improved output resistance, which reduces variations in current due to the output potential at the cost of limited output swing.

The loop filter then produces the control signal of each delay cell. The feedback loop adjusts the variable delay-line such that these edges are locked in time. Since a DLL's delay-line is not an integrator like the VCO in a PLL, a DLL is inherently stable. The loop filter was implemented with a single capacitor sized at 6pF.

The delay-line is composed of 128 delay cells and is represented in Figure 5-5a, and the schematic for the individual delay cell is shown in Figure 5-5b. Each stage

88
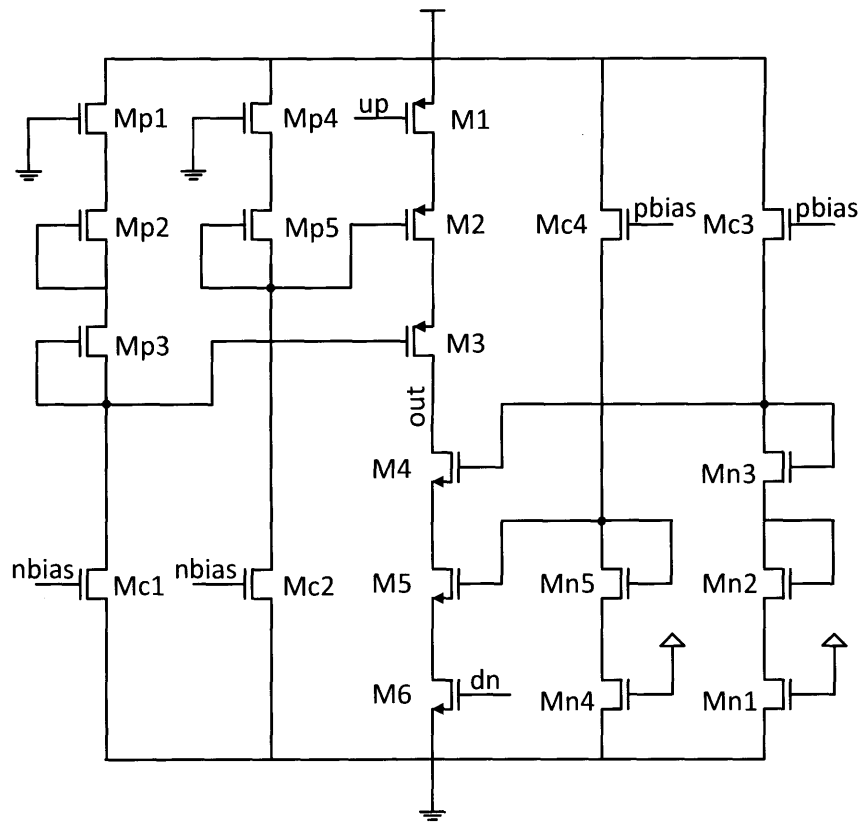
Figure 5-4: Schematic of charge pump used in DLL.

in the delay-line consists of an inverter pair, I_1 and I_2, which takes complementary inputs. The delay cells are cascaded such that the I_1 and I_2 from each cell form two inverter chains that operate in parallel and switch with opposite polarity. Since the inverter chains are complementary, the outputs are alternately selected from the two chains, so only rising edges from the delay line are chosen to clock the comparator. These rising edge outputs, which are highlighted in red in Figure 5-5a, are buffered out and routed to the MUX. Dummy buffers were placed on the falling edge outputs to match the capacitive load on the rising edge outputs.

The cross-coupling inverters I_3 and I_4 ensure that each stage's rising and falling edges are in sync. They also ensure that the output nodes from the two inverter chains are complementary. Without the cross-coupling inverters, the two inverter chains would be independent, and their edges may drift apart. I_3 and I_4 were sized to be one-fourth the size of I_1 and I_2 to limit the loading on the output nodes while

89
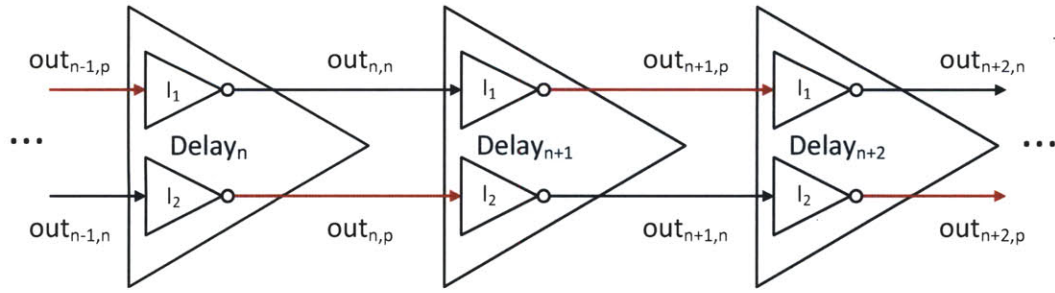
(a)
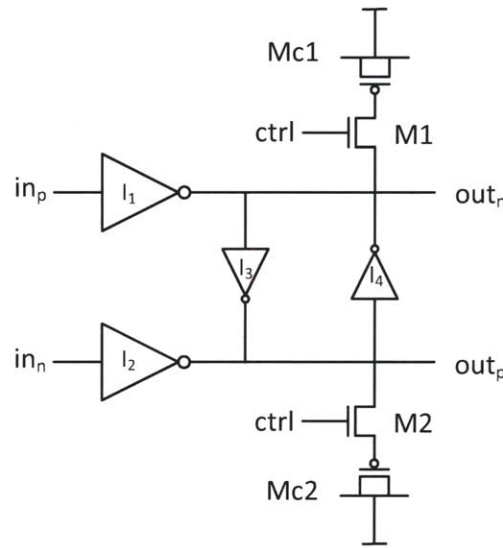
(b)

Figure 5-5: (a) Diagram of delay line in which alternating outputs are selected for the rising edge. (b) Schematic of individual delay cell.

still providing sufficient cross-coupling.

The variable delay was achieved by controlling the resistance of transistors M1 and M2 to shunt capacitors, Mc1 and Mc2, as described in [55]. The resistors determine the loading that the output nodes see from the capacitors. Figure 5-6 shows the equivalent circuit for the rising edge portion of the delay cell. We assume the input to inverter I_1 is a falling edge, so its output is a rising edge. The NMOS in the inverter is off and is shown as an open switch. The resistor $R_{I1,p}$ represents the pull-up resistance of the PMOS in the inverter cell I_1. Capacitor C_L represents the total capacitance at the output node of inverter I_1. R_{M1} is the resistance value of M1 and
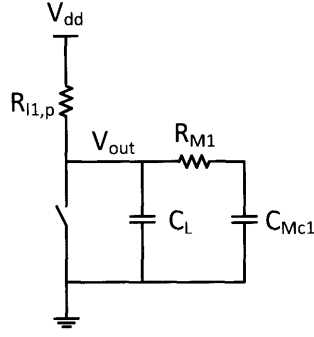
Figure 5-6: Equivalent circuit diagram of the pull-up side of the delay cell.

C_{Mc1} is the capacitance of capacitor Mc1. The transfer function from Vdd to Vout is an impedance divider and can be simplified into the following.

$$\frac{V_{out}}{V_{dd}} = \frac{1 + sC_{Mc1}R_{M1}}{1 + s(R_{I1,p}C_L + R_{I1,p}C_{Mc1} + R_{M1}C_{Mc1}) + s^2 R_{I1,p}R_{M1}C_L C_{Mc1}} \tag{5.3}$$

In the extreme case when R_{M1} is infinity, the transfer function becomes Equation 5.4. In this case, there is no loading from capacitor C_{Mc1}, and the time constant for charging Vout is $R_{I1,p}C_L$.

$$\frac{V_{out}}{V_{dd}} = \frac{1}{1 + sR_{I1,p}C_L} \tag{5.4}$$

In the opposite extreme when R_{M1} is zero, the transfer function simplifies into Equation 5.5. With R_{M1} equal to zero, the capacitance of Mc1 adds directly to C_L. The time constant for charging Vout is $R_{I1,p}(C_{Mc1}+C_L)$. Compared to case when R_{M1} is infinity, the larger capacitance causes a longer time constant for the rise time, and this increases the delay time between delay cells.

$$\frac{V_{out}}{V_{dd}} = \frac{1}{1 + sR_{I1,p}(C_L + C_{Mc1})} \tag{5.5}$$

Between these two cases, the time constant depends on the value of R_{M1}. The resistor was implemented with an NMOS transistor operating in triode region, and the resistance value is a function of M1's Vgs, which is determined by the control

signal from the DLL feedback loop. As the control voltage is raised, the resistance becomes smaller which increases the loading from the shunt capacitor and increases the delay time. The shunt capacitors, C_{Mc1} and C_{Mc2}, were implemented using PMOS transistors with the source and drain tied together. The capacitors were sized to be 50fF. The resistor was implemented with an NMOS transistor operating in triode region. The resistance and shunt capacitance could have also been implemented with a PMOS resistor and an NMOS capacitor with similar functionality. However, an NMOS resistor was chosen because a smaller sized device could be used to achieve the same resistance compared to a PMOS resistor, which helped make a more compact design with less parasitics. The PMOS capacitor has its source and drain tied to Vdd which provides isolation from substrate noise.
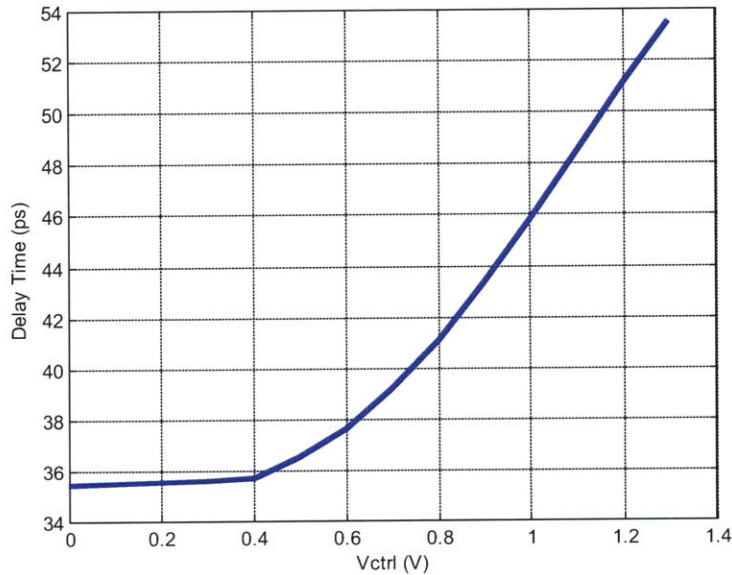
Figure 5-7: Plot of loaded delay-cell delay time versus control voltage.

The delay time with an identical delay cell load is simulated versus the control voltage in Figure 5-7. Compared to current-starving the inverters, this architecture has a more gradual transfer function between the control voltage and delay time, which reduces sensitivity to noise on the control voltage at the cost of lower tuning range. The positive output of each delay cell was buffered and routed to the MUX.
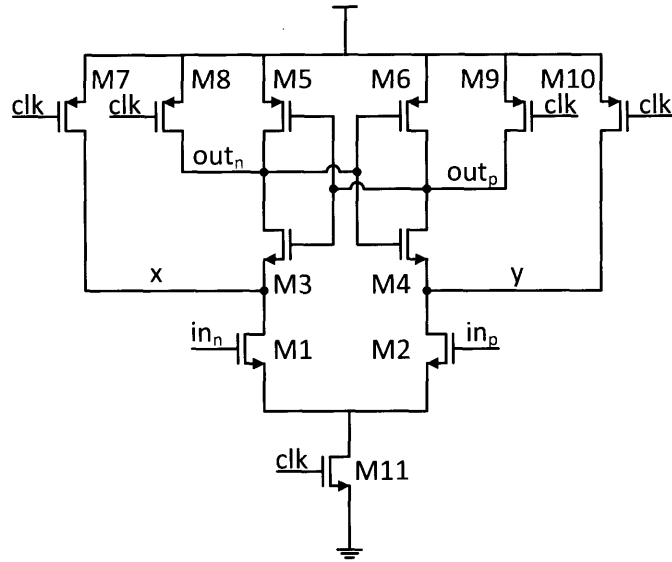
.Figure 5-8: Schematic of comparator cell.

The negative output was loaded with a dummy buffer to ensure better symmetry between the falling and rising edges of the delay line.

5.2.2 Comparator

The comparator determines whether the IF input signal is above or below a zero reference level. Figure 5-8 shows the schematic of the comparator used in this design. Transistors M3 to M6 form a cross-coupled latch. The comparator clock is used as the enable signal. When the clock is high, the latch will rail due to positive feedback and the polarity is determined by the input to the differential pair, M1 and M2. A low clock signal will reset the circuit by turning on transistors M7 to M10 which will pull the output nodes and nodes x and y to Vdd. Resetting the comparator every clock cycle removes hysteresis, so data dependent nonlinearities are removed.

Noise in the comparator can cause incorrect triggering which degrades the phase estimation. For accurate measurements, we want the input referred voltage noise from the comparator to be much smaller than the voltage increment ΔV during one DLL time step Δt. The voltage increment of interest occurs is dependent on the slope of the input signal near the zero crossings. Assuming the input signal has amplitude

A, the slope can be calculated as follows.

$$x(t) = A \cdot sin(\omega t) \tag{5.6}$$

$$\frac{dx(t)}{dt} = A\omega \cdot cos(\omega t) \tag{5.7}$$

$$\Delta V = \Delta t \cdot \frac{dx(t)}{dt}\bigg|_{t=0} \tag{5.8}$$

$$\Delta V = \Delta t \cdot A\omega \tag{5.9}$$

We want the comparator noise to be much smaller than ΔV. A worst case noise analysis was performed by simulating the input referred noise while all devices were in saturation to emulate the transition period of the comparator. The input referred noise was simulated by integrating the input noise's power spectral density up to the bandwidth of the comparator which was 5.3GHz. This evaluated to 0.41mV.

The time step Δt if approximately 50ps and the operating frequency is designed for 150MHz. The amplitude will vary depending on the input signal. We apply the restriction that ΔV must be 10 times larger than the input referred noise. This will ensure that even three standard deviations of noise will not cause an incorrect trigger. Given these parameters, we solve for the amplitude and find that we can tolerate an amplitude as low as 87mV before comparator noise can potentially be an issue.

5.2.3 MUX and Digital Counter

The multiplexer (MUX) circuit taps out each delay step in the DLL's delay line and and selects one of them to clock the comparator. It has a hierarchical structure, in which there were three layers of 4-to-1 MUX's followed by a single 2-to-1 MUX to achieve to the 128 selection. Each MUX was composed of parallel transmission gates with a single inverter buffer to restore the signal. The layout was designed to be as symmetrical as possible.

The digital counter was a 7-bit ripple carry adder that had a plus or minus one input based on the output of the counter. This topology was chosen for its simplicity, which was a consideration due to time constraints. However, the ripple carry adder is slow due to the propagation of the carry bits, which can limit the speed of the phase tightening system if the operating frequency is pushed higher. A possible alternative for a future implementation is a carry look-ahead adder which reduces the computation time of the carry bits.

5.3   Simulation Results

Figure 5-9 shows the results of a Spectre transient simulation that demonstrates the system's operation as it locks to the zero-crossings of an input sine wave. The top plot shows the red IF input sine wave relative to the blue comparator clock, which is the clock edge that has been multiplexed from the DLL's delay-line. The bottom plot shows the digital code output of the counter from Figure 5-9, which is labeled as phase index, and each increment represents 1LSB. The glitches in this plot are the result of transitioning bits and are removed in the chip implementation by latching the outputs.

Initially the IF signal's rising edge lags the clock's rising edge. When the comparator is triggered by the clock edge, its output increments the counter and selects the next delayed clock edge to trigger the next cycle. While the IF signal is lagging, the counter continues to increment and select more delayed versions of the DLL clock. The plots in Figure 5-10 zoom in on this behavior. We observe the comparator clock's rising edge is increasingly delayed relative to the IF signal's rising zero crossing as the phase index increments. Eventually, the IF signal will lead the comparator clock. When this occurs the comparator decrements the counter and selects the previous delayed clock edge, causing the IF signal to lag the clock again.

This process will repeat, and if there is no phase noise, the digital output will alternate between these two values as it selects between the two DLL clock edges that trigger the comparator around the IF signal's zero crossings. We see that when

the phase tightening system is locked, the comparator clock and the IF signal are in phase. We average these digital phase index values to obtain the phase estimate.

5.4   Measurement of Phase-to-Digital Conversion

Figure 5-11 is a die photo of the digital phase tightening system with the layout. It was fabricated using the CMOS devices IBM's $0.13\mu m$ BiCMOS8HP process on the same chip as the MMW PLL. It has an active area of $0.2mm^2$.

The measurement setup is shown in Figure 5-12. Two synchronized signal generators were used to create the DLL clock and IF signal. These signals had the same frequency but could be programmed with different phases. The phase tightening system was characterized by sweeping the phase of one signal relative to the other via a computer and General Purpose Interface Bus (GPIB). The phase difference was ramped in increments of 0.5°. Since these were analog inputs and variations were observed in the phase increments, an oscilloscope was used to measure and record the actual phase difference. For each phase input increment, a logic analyzer was used to sample and store the digital phase index outputs. This data was then post-processed in Matlab and Excel, and a digital phase estimate was calculated for each phase input. A pattern generator was used to control all digital inputs into the test chip.

The phase tightening system was tested with DLL clock and IF signals at 175MHz. The DLL produces 128 rising clock edges with 45ps delay steps. The blue data points in Figure 5-13 show the measured digital output plotted against the input phase. For each phase increment, 45875 phase index samples were taken and averaged to produce each blue point. The red curve shows the ideal transfer curve for 360° phase sweep with 128 delay steps. The quantization steps of the phase to digital conversion can be observed. Taking the difference between these signals gives the error plot shown in Figure 5-14. This plot shows the accumulation of nonlinearities in the delay-line and mismatches in the 128-to-1 MUX. However, these nonlinearities are static errors that are consistent through multiple measurement runs, so they are calibrated out in the digital domain. The calibration was performed by taking a running average with
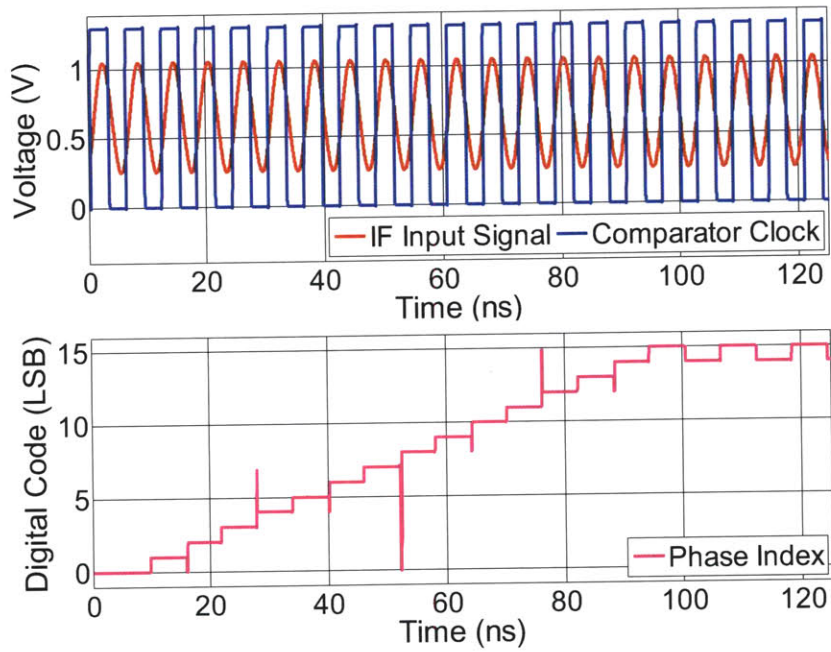
Figure 5-9: Simulation of the comparator clock locking to the IF signal's zero crossings with the counter's digital code output.
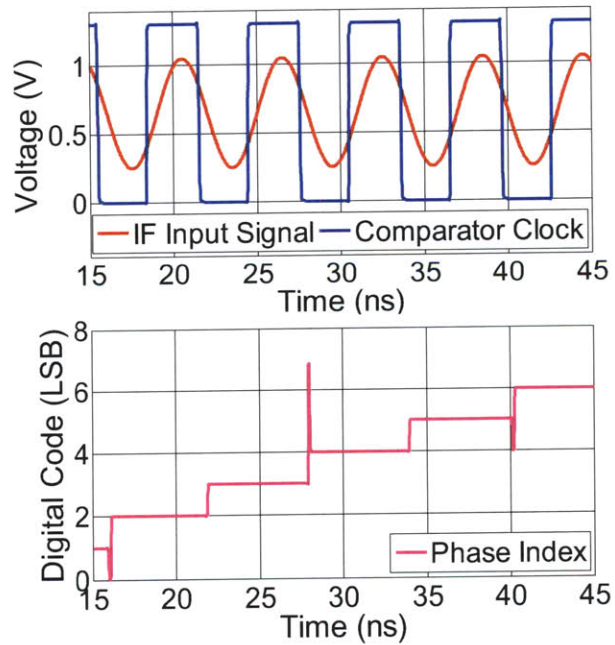
Figure 5-10: Zoomed-in simulation of phase tightening system operation demonstrating the incrementally delayed comparator clock.
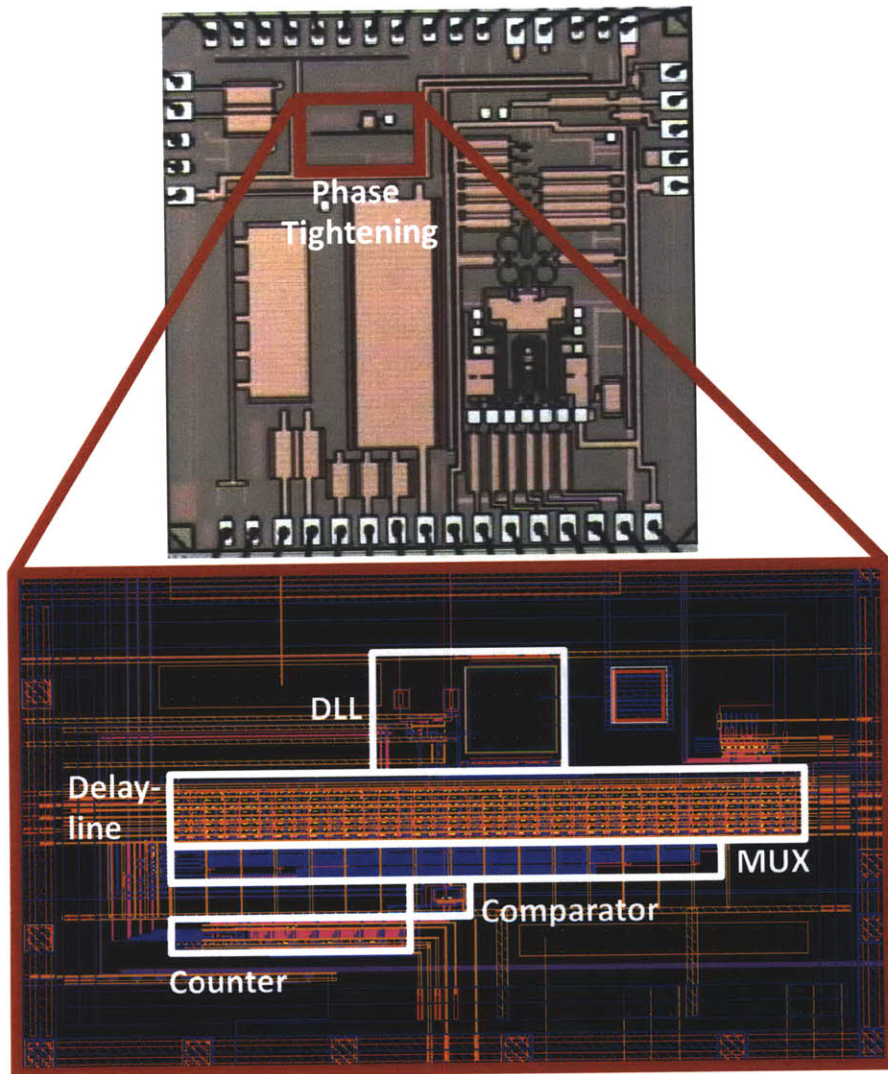
Figure 5-11: Die photo and layout of the phase tightening system. The layout shows key blocks of the design.
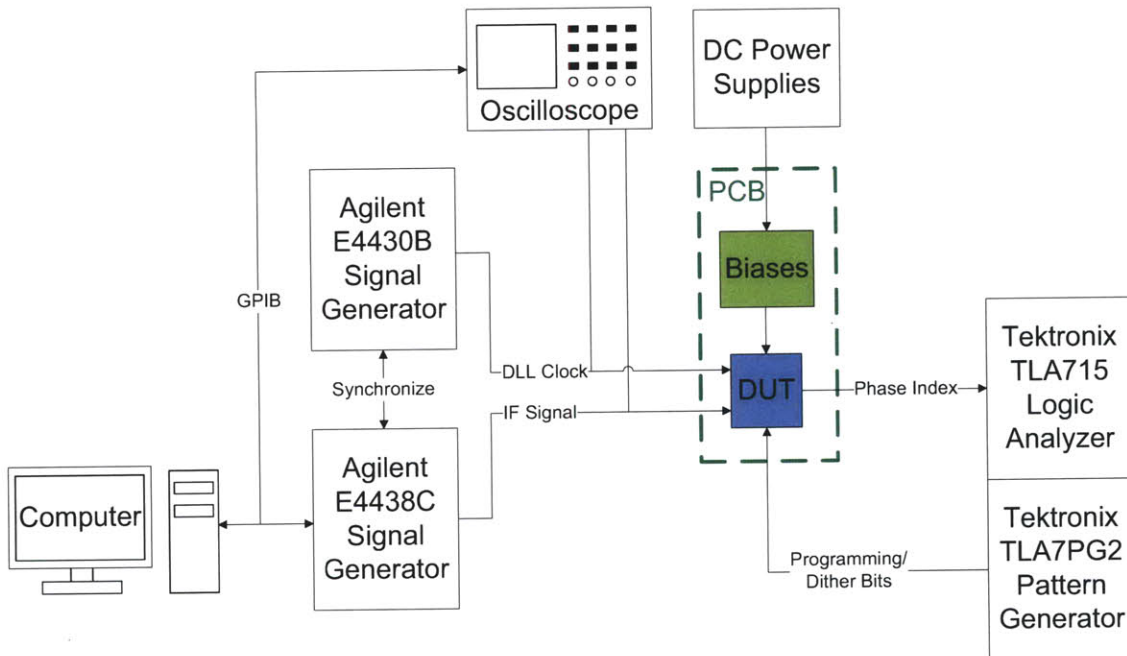
Figure 5-12: Test measurement setup for phase tightening system.
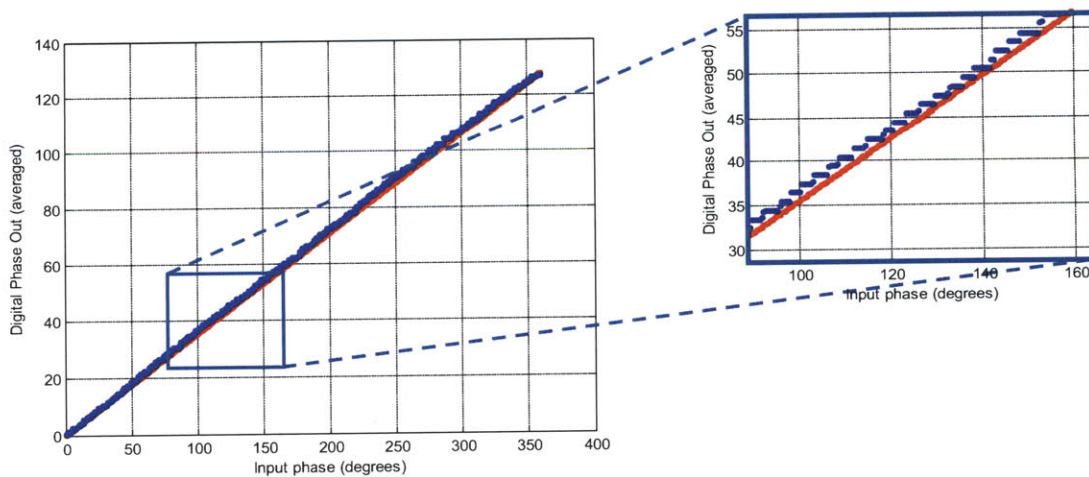
Figure 5-13: Measured digital phase output versus input phase.
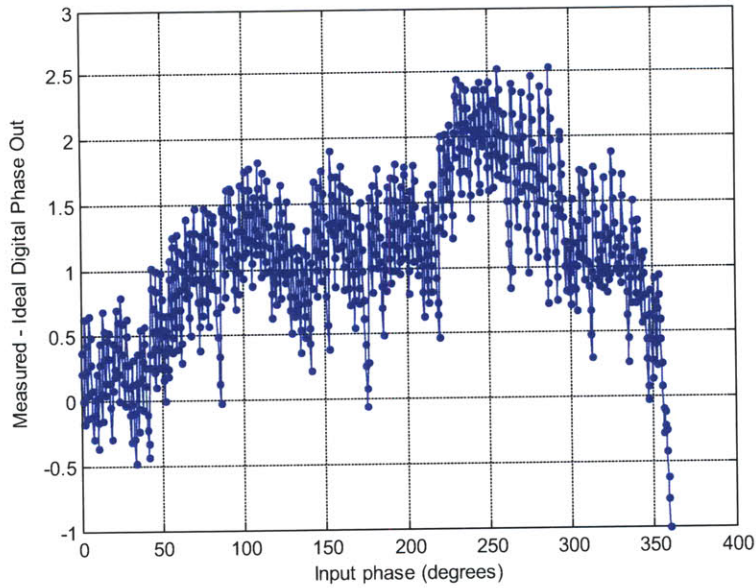
Figure 5-14: Measured error between phase output and ideal transfer curve.

a window size of 5° and then subtracting the averaged curve from the error plot. The resulting calibrated error plot is shown in Figure 5-15. The RMS error of this plot is 0.31LSB, corresponding to 13.8ps at the IF or 31.3fs when referenced to a 77GHz carrier. This demonstrates the performance of the phase-to-digital conversion.

It should be noted that the calibration is a nontrivial task. For an array, each receiver would need its own calibration. The implemented receiver would require a means to sweep the input signal phase while recording the digital outputs in order to determine the nonlinearities of the DLL's delay-line. Since the nonlinearities are not simply gain or offset errors, the most feasible approach is to calculate a look-up table during the calibration process that would correct each digital code. The look-up table would need to be an array of length 128, and each element would require 10 bits, in which 3 bits are allocated for the decimal value, to sufficiently correct the nonlinearities. The use of 3 bits was determined empirically in which further increase in bits for the decimal gave only 0.001LSB improvement in the RMS error. Each receiver in the array would require a look-up table, but given the relatively small size of the look-up table that can be implemented all digitally, this is not unreasonable.
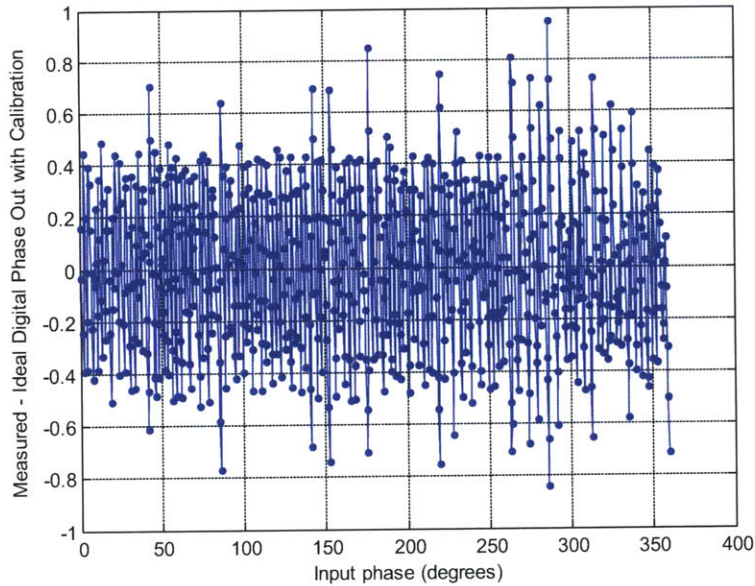
Figure 5-15: Measured error plot after calibration.

5.5 Reducing Error by Averaging with Noise

Although it may be counterintuitive, having some noise in the system can be useful. With digital phase tightening, the digital output will continue to alternate even when the system is locked. If there is no noise present, then the values will alternate between two values, say M and M+1. This is illustrated in Figure 5-16.a. The system is trying to measure the true value μ by taking samples. These samples are averaged to form $\bar{x}[n]$ which is an estimate of μ. No matter how many samples are taken, averaging M and M+1 will always produce M+1/2. However, if there is noise, the digital outputs will span more than just two bins, so more information is taken into account during the averaging. Figure 5-16b shows that the averaged result $\bar{x}[n]$ including noise is much closer to the true value than the case in Figure 5-16.a which had no noise. Averaging with noise allows for more accurate estimation than that set by the quantization level.

The previous example gives a qualitative demonstration of how noise can be beneficial to the system. However, there are limits to how much noise can be added and how much it improves the estimation. Figure 5-17 plots the RMS error between

101
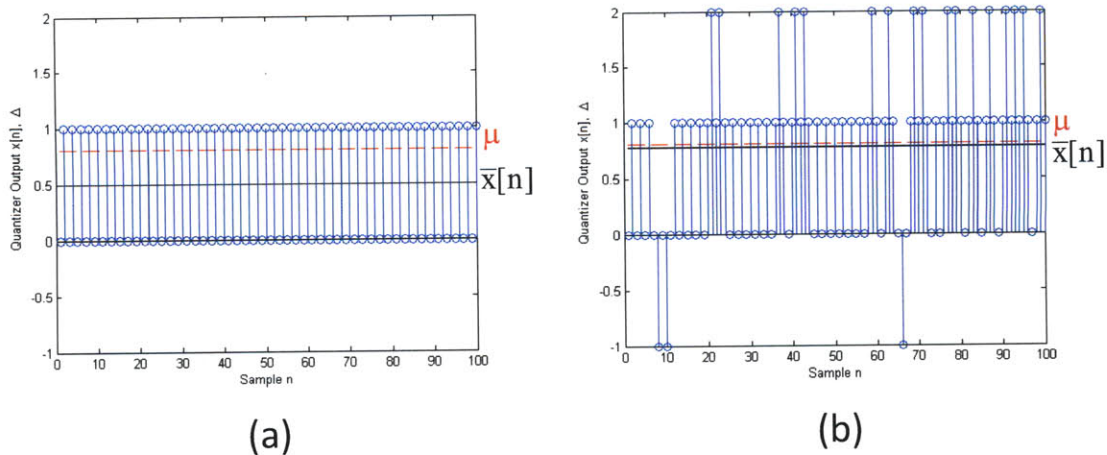
Figure 5-16: This is an example of averaging data samples with noise can produce a better estimate. The data samples are shown and the averaged value and true value are shown. (a) With no noise is added, the averaged value is always half the step size. (b) With noise added, the averaged value is closer to the true value.

the averaged result and the true value versus the standard deviation σ of Gaussian noise for different averaging window sizes, N. Both the noise level and RMS error are normalized to Δ which is one LSB step. These curves are calculated plots based on the work in [56]. If the distribution of the noise is known, we can calculate the RMS error for different window sizes and noise standard deviation σ.

We first focus on the blue "N=100" curve. This curve corresponds to the RMS error that we obtain when we average 100 samples while sweeping the noise level. When the noise σ is zero, the RMS error is $1/\sqrt{12}$LSB, which is the quantization error. As the noise level increases from zero, the RMS error decreases because the noise is causing the outputs to span more bins, and averaging these values lowers the error caused from quantization. As σ continues to increase, the noise will start to increase and then become so large that it will swamp out the improvement from averaging and will cause the RMS error to increase. For the "N=100" curve, we see that this occurs at σ equal to 0.37LSB in which there is a minimum RMS error of 0.05LSB. RMS error of 0.05LSB is the best that can be achieved for a sampling window of 100, and it occurs when the noise is at an optimum level of 0.37LSB. For noise levels larger than 0.37LSB, the error is no longer limited by the quantization.
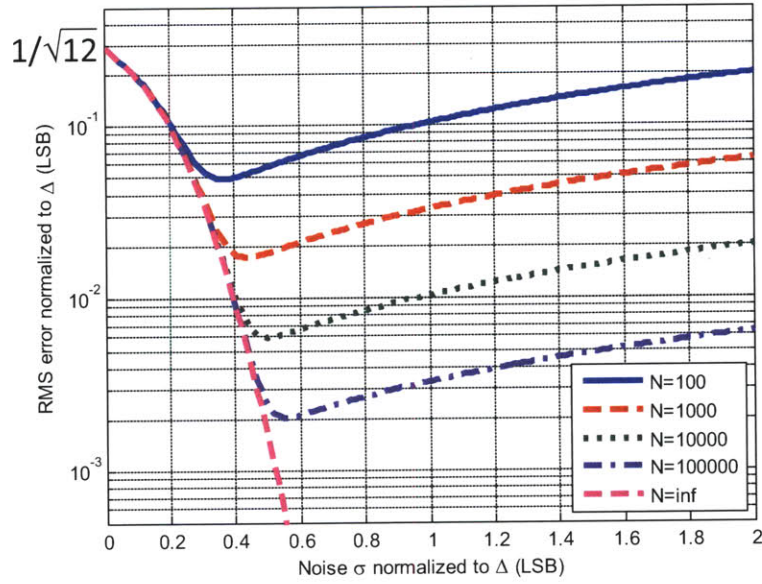
Figure 5-17: Plot of RMS error versus standard deviation σ of noise for different number of averaged samples. The RMS error and σ have been normalized by the LSB step size Δ. In our system, one LSB corresponds to 45ps.

It is limited by by the added noise and is approximately equal to $\sigma/\sqrt{100}$. When the noise level is below 0.37LSB, the RMS error is mostly due to quantization, and when the noise if above 0.37LSB, the added noise will dominate the RMS error.

Figure 5-17 also shows the curves for N values of 1000, 10000, and 100000. When σ is equal to zero, we see that all of these curves have an RMS error of $1/\sqrt{12}$ LSB. If there is no noise at all, no amount of averaging will improve the quantization error. As σ is increased from zero, these curves follow the same roll-off as the "N=100" curve, but they reach lower RMS error, and the minimum RMS value occurs for larger σ. As the averaging window becomes larger, higher noise levels can be averaged out and a lower RMS error can be achieved. When the RMS error is dominated by the noise and not the quantization, the RMS error is approximately $\sigma/\sqrt{N}$. For a fixed σ, increasing the averaging window will improve the RMS error by the square root of that increase. This reduction is evident in the figure when we look at any σ greater than 0.6LSB, in which the RMS error is clearly dominated by the added noise for the "N=100 to 100000" curves. If we hold a noise level σ constant in this region, the reduction in RMS error for each curve is $\sqrt{10}$ since N is increased in factors of 10.

A final observation from this figure is that there is a limit to the amount of reduction in RMS error for a given noise level. The magenta curve is the curve for N equal to infinity. Like the other curves, it has RMS error of $1/\sqrt{12}$LSB when σ is zero. As σ increases, the RMS error rolls-off, but it does not reach a minimum. Instead, it continues to drop indefinitely for increasing σ. For a given noise σ with infinite averaging, all of the added noise will be averaged out such that RMS error is limited by the quantization. From this curve, we see that there is a minimum achievable RMS error for a given σ even with infinite averaging.

5.6 Digital Phase Tightening in MMW Imaging

In the MMW imaging system, we want phase estimates on the order of femtoseconds at the carrier frequency. The quantization level is too large for direct use in a MMW imaging system, even when the signal is mixed down to an IF. However, our target frame rate is 10fps, which is low compared to the IF. This allows ample time to capture and average a large number of samples to produce a better phase estimate. For each frame, we assume a generous allocation of 90ms for beamforming and image processing in the digital domain, which leaves 10ms of time remaining for data sampling. Our current test setup has an IF of 175MHz which gives a window of approximately one million samples to average for each frame.

Noise was introduced into the system using the phase modulation feature of the Agilent E4438C signal generator to modulate the generator's internal noise source into the IF signal. The phase noise level could be controlled and measured on a spectrum analyzer, and the corresponding timing jitter could be measured using an oscilloscope. We measure the RMS error for different levels of averaging and jitter and observe the improvement in phase estimation. Figure 5-18 shows measured RMS error versus the number of averaging samples, N, for different levels of timing jitter. Due to limitations in the logic analyzer, the maximum number of stored data for averaging was limited to 225ksamples. We observe that the RMS error level is directly related to the amount of jitter introduced, which we expect because more noise should result
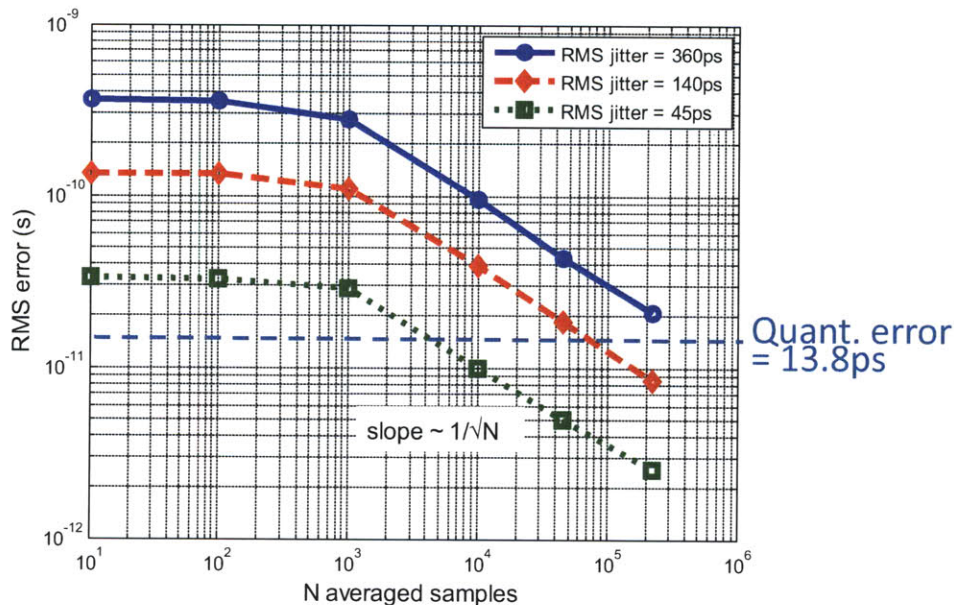
Figure 5-18: Measured RMS error plotted against N averaged samples for various levels of timing jitter.

in larger error given the same amount of averaging. We also observe that the noise stays constant and then rolls off for increasing N. Once it rolls off, the error is reduced by the expected $1/\sqrt{N}$.

The flat portion of the curves in Figure 5-18 suggests that there is no improvement in RMS error until a large enough sample size is averaged. This characteristic is a result of applied phase noise. Figure 5-19 is the measured frequency spectrum of the input signal to the phase tightening system. The input tone of 175MHz can be observed with phase noise level of 61.5dBc/Hz and a single-sided bandwidth of 100kHz. The different levels of jitter shown in Figure 5-18 were obtained by adjusting the phase noise level with the signal generator. However, the single-sided phase noise bandwidth remained constant at 100kHz.

The phase noise bandwidth determines the rate at which the jitter is changing. The 100kHz bandwidth is significantly slower than the sampling rate of 175MHz. A sample sequence of the digital phase index outputs is plotted against time in Figure 5-20. This data was obtained by phase modulating 360ps of timing jitter from the
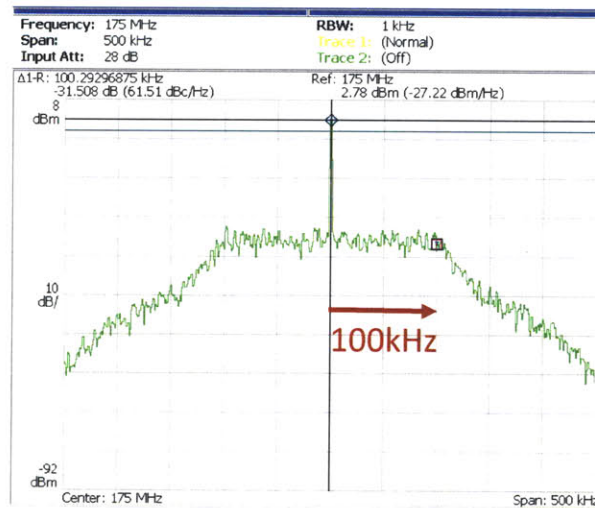
Figure 5-19: Measured spectrum of IF input signal with added phase noise from signal generator. The phase noise has a 100kHz bandwidth.

signal generator. The jitter causes variations in the phase index outputs. However, because the phase noise bandwidth is much smaller than the operating frequency, the phase index output is varying slowly compared to the samples that are taken.

Figure 5-21 shows the same data but windowed with different sample lengths N. For N equal to 10000, the phase indices show significant variation due to the noise. We expect that averaging out the noise in this sequence will improve the estimate of the true phase. When N is reduced to 1000, there is not enough variation due to noise for averaging to give its full benefit. When the sampling window is further reduced to 100 or 10, there is even less variation such that averaging gives almost no benefit at all. This explains the RMS error profile seen in Figure 5-18. For N less than 1000, there is very little improvement in RMS error because there is little change in the phase indices due to noise within that window. The result is a flat portion in the profile up to around N equal to 1000. After this point, we see the $\sqrt{N}$ reduction because the sampling time is sufficiently large to capture variation from the noise that can be averaged out to improve the phase estimate. The requirement of window size N being 1000 is consistent with the phase noise bandwidth being approximately 1000 times slower than the operating frequency of 175MHz.
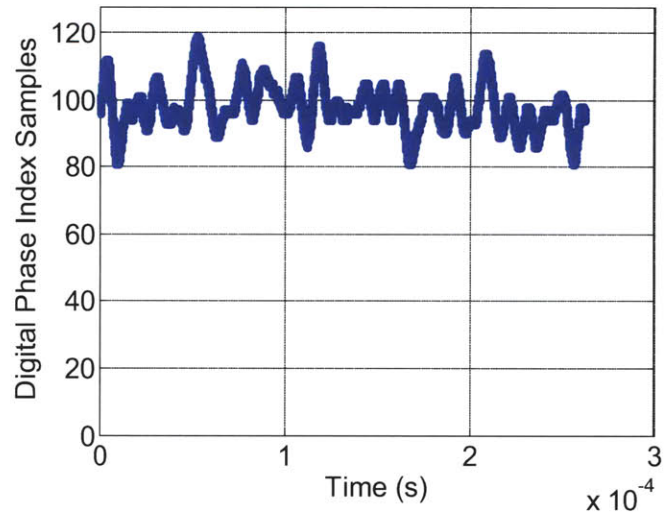
Figure 5-20: Sample phase index plot with applied jitter from signal generator.
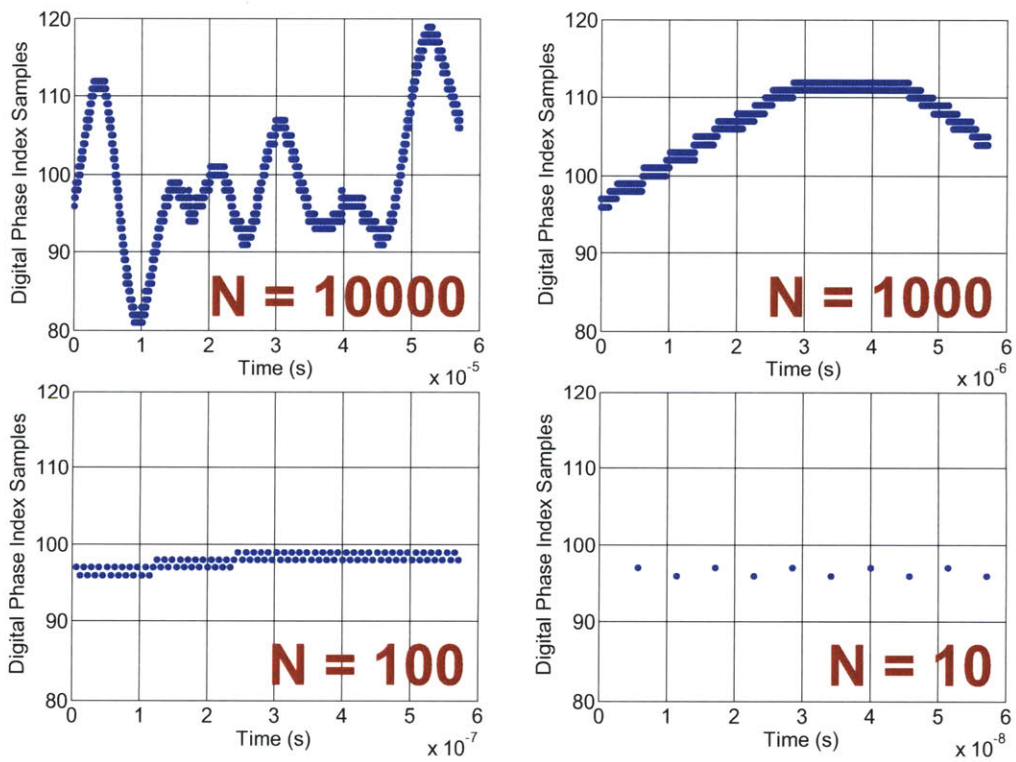
Figure 5-21: Zoomed-in sample phase index plot with applied jitter from signal generator.

5.6.1 An Alternate Perspective to Phase Noise Bandwidth

An alternate approach to understanding the effect of the phase noise bandwidth is to note that correlation between adjacent samples reduces the effectiveness of averaging. If the window size is too small, all of the samples in the window are correlated and averaging these samples gives no improvement. We want to know how long in time we must wait until the samples are no longer correlated, which will tell us how many samples we must take before the averaging will take effect. For simplicity, we model the phase noise with a brick wall roll-off, so the the phase noise spectrum $X(\omega)$ has a rectangular profile with a 100kHz single-sided bandwidth as shown in Figure 5-22. Since we are only concerned with phase, the signal center frequency is not relevant, and $X(\omega)$ is centered at zero. We denote the height of the phase noise spectrum as a constant A.
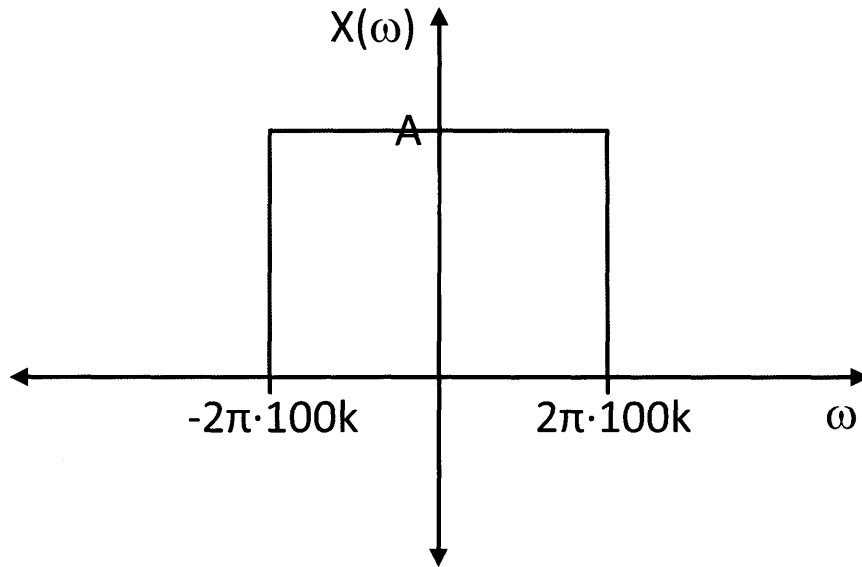
Figure 5-22: $X(\omega)$ represents the phase noise spectrum with 100kHz single-sided bandwidth and a brick wall roll-off.

We represent $X(\omega)$ as follows.

$$X(\omega) = A \cdot rect\left(\frac{\omega}{2\pi(2 \cdot 100e3)}\right) \tag{5.10}$$

In the time domain, the autocorrelation of x(t), $\phi_{xx}(t)$, is the convolution of x(t)

108

with itself. Fourier transform of the autocorrelation, $\Phi_{xx}(\omega)$, is the multiplication of $X(\omega)$ with itself, which is still a rectangular function.

$$\Phi_{xx}(\omega) = A^2 \cdot rect\left(\frac{\omega}{2\pi(2 \cdot 100e3)}\right) \tag{5.11}$$

The inverse Fourier transform is taken to calculate the time domain signal of the autocorrelation which is a sinc function.

$$\phi_{xx}(t) \propto sinc(2 \cdot 100e3 \cdot t) \tag{5.12}$$

The coefficient of the sinc function is the constant A^2 scaled by a factor from the inverse Fourier transform of the rectangular profile. However, this constant does not effect the roll-off of the sinc function, so we remove it and replace the equal sign with a proportionality symbol. The autocorrelation $\phi_{xx}(t)$ is plotted against time in Figure 5-23. We assume that the sample is no longer correlated once the autocorrelation reaches its first zero, and this occurs at $5\mu s$ for the 100kHz bandwidth rectangular spectrum. For a 175MHz sampling frequency, 875 samples are taken before the signal is no longer correlated. 875 samples matches closely with the measurement results in Figure 5-18, which show that 1000 samples are needed before the averaging reduces the RMS error.

5.6.2   PLL Phase Noise

The roll-off point in the RMS error profile depends on the phase noise bandwidth. For a noise source with a higher bandwidth, we expect that the roll-off point would occur sooner. The bandwidth in the previous measurements was set by the phase modulation feature of the signal generator. As described in Section 4.3, we expect the dominant phase noise source in the imaging receiver system is the MMW LO generated by the PLL.

In Chapter 4, we implemented and analyzed a MMW PLL. Although the PLL operated slightly lower than the target frequency, we can use this as a basis for the PLL noise we expect to see in the imaging system. We found that the timing jitter
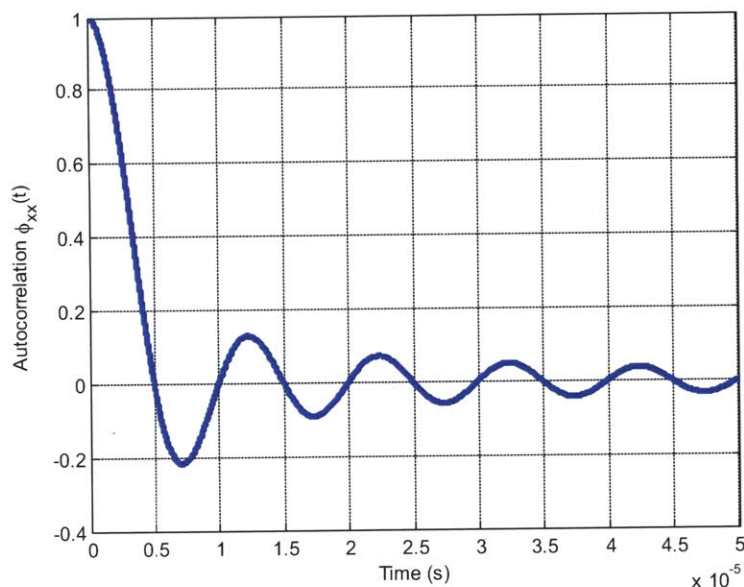
Figure 5-23: Autocorrelation $\phi_{xx}(t)$ of 100kHz single-sided bandwidth phase noise with brick wall roll-off. The amplitude has been normalized to one.

from this PLL is about 1ps.

We recall from Equation 5.2 that phase noise from the PLL is additive for the IF signal, which means that the PLL timing jitter is scaled up by the ratio of ω_{LO}/ω_{IF}. Scaling the jitter from this PLL to the IF would give a jitter of about 450ps. However, the bandwidth of the PLL's phase noise is 10MHz, which is two orders of magnitude higher than the bandwidth of the noise source used in these measurements. Given this noise, we expect the RMS error to roll off two orders of magnitude sooner at approximately N equal to 10, and the RMS error to improve by approximately $\sqrt{100}$. In the system, we expect to average about one million samples which reduces the error further.

Figure 5-24 shows a Monte Carlo simulation of the RMS error given a jitter of 450ps but changing at a more rapid rate to match the PLL's 10MHz phase noise bandwidth. The Monte Carlo simulation was performed by adding noise to a constant signal using a zero-mean random number generator with the expected standard deviation. The noise was held constant, and a new noise value was generated at the same rate as the expected phase noise bandwidth compared to the IF. Samples from
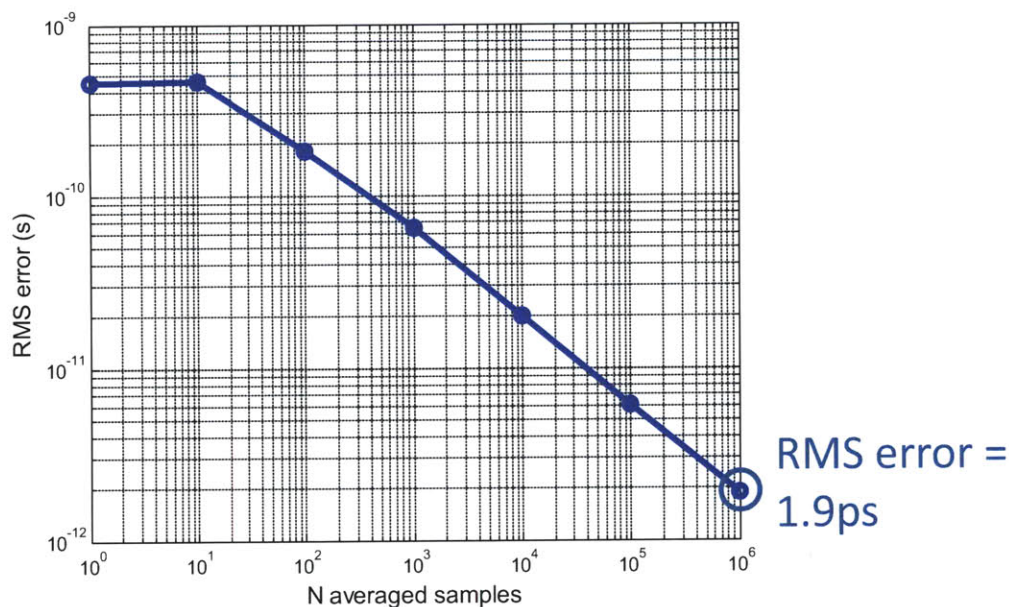
Figure 5-24: Monte Carlo simulation result of expected RMS error given PLL phase noise with 10MHz bandwidth.

this signal were quantized and then averaged over different window lengths. The RMS error was calculated for the different averaged values given the true value of the constant signal and plotted in Figure 5-24. From this plot, we see that the roll-off occurs about two orders of magnitude sooner given the higher bandwidth noise. The RMS error after averaging one million samples is 1.9ps at the IF of 175MHz. Scaling this to the 77GHz carrier gives 4.4fs.

This simulation was verified with measurement shown in Figure 5-26. The proper amount of noise was created by using the signal generator's I/Q modulation feature with a noisy signal generated in Matlab. Given the signal generator's modulation frequency to be 100MHz, a random sequence was generated to be 10ms which is long enough such that there is no repetition in a single measurement. The standard deviation was set to achieve 450ps of jitter for a 175MHz signal. For a bandwidth of 10MHz, the random sequence was upsampled with an oversample ratio of five and then digitally filtered to remove the spectral replicas. In order to obtain phase modulation, the I and Q signals were generated as the cosine and sine of the noisy

signal. The measured spectrum of this signal is shown in Figure 5-25. We observe the single-sided bandwidth is 10MHz as designed. The timing jitter was measured to be 450ps using an oscilloscope.

Using this noisy signal as the IF input to the phase tightening system gives the result shown in Figure 5-26. The roll-off point occurs roughly at N equal to 10 which is sooner than the roll-off with the 10kHz noise and is in agreement with the Monte Carlo simulation. The RMS error is slightly lower in the measured result than in the Monte Carlo simulation, particularly when N is 10 or less. This disparity can be attributed to the inaccuracy of averaging over very small window sizes. The incremental behavior of the phase tightening system has some filtering effect on the 10MHz noise which lowers the RMS error. As the averaging window size is increased, the measured result matches more closely to the Monte Carlo simulation in terms of the slope and the absolute levels of the RMS error. The maximum window size in this measurement was 90ksamples. At 90ksamples, the RMS error was 4.5ps at the IF. The red dashed line in Figure 5-26 shows the extrapolation of the data to one million samples which gives 1.4ps. Scaling this value from the 175MHz IF to the 77GHz carrier gives 3.2fs of RMS error. This value is within the range of accuracy that we are seeking for the imaging system.

One concern with the plot from Figure 5-26 is the validity of extracting the measured result to 1 million samples. We know from Section 5.5 that there is a limit to the reduction in RMS error for a given level of jitter, even with infinite averaging. Figure 5-27 shows calculated RMS error versus the standard deviation of noise for different window sizes for averaging. The curve for infinite N is shown in magenta in this plot. For a step size (1LSB) of 45ps, 450ps of jitter corresponds to 10LSB in the plot. We see for 10LSB, averaging window of N equal to 1 million and many orders of magnitude beyond 1 million still shows $1/\sqrt{N}$ behavior.
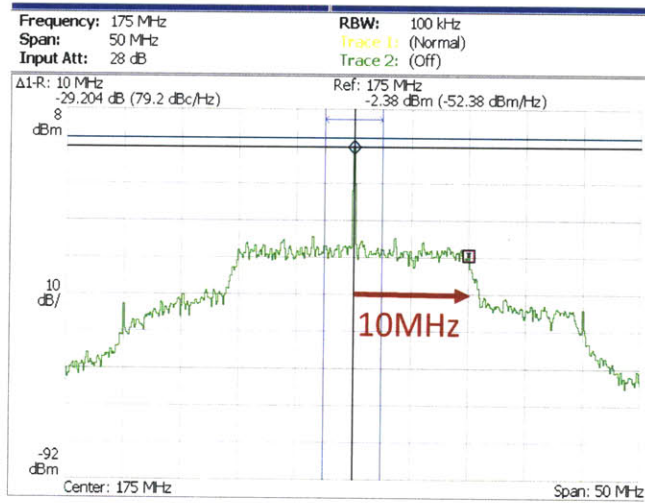
Figure 5-25: Measured spectrum of the IF input signal with phase noise generated from I/Q modulation. The phase noise has a 10MHz bandwidth.
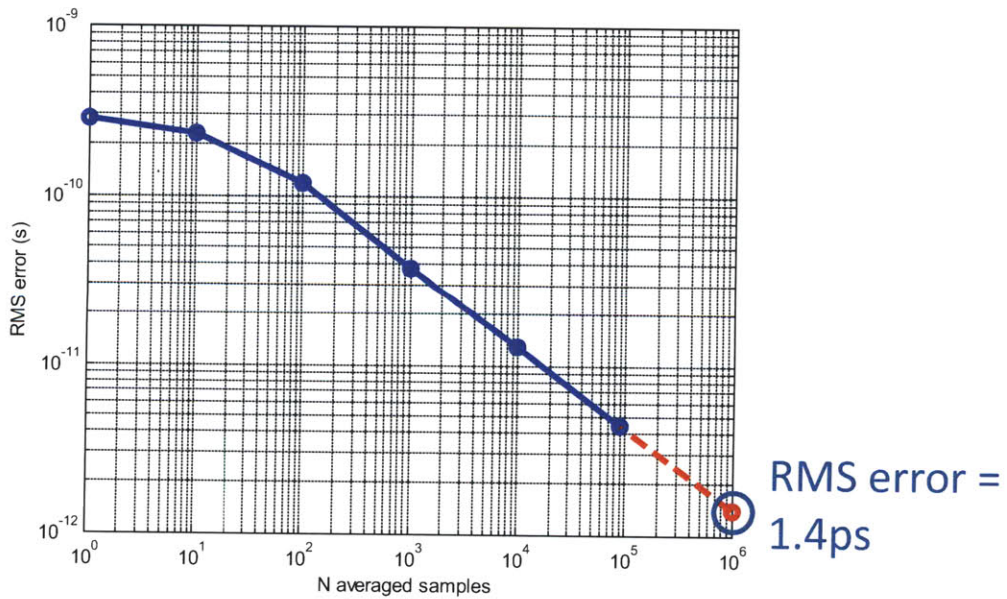
Figure 5-26: Measured RMS error plotted against N averaged samples for an IF signal with 450ps jitter and 10MHz bandwidth phase noise. Extrapolating the measurement to 1 million samples give 1.4ps of RMS error.
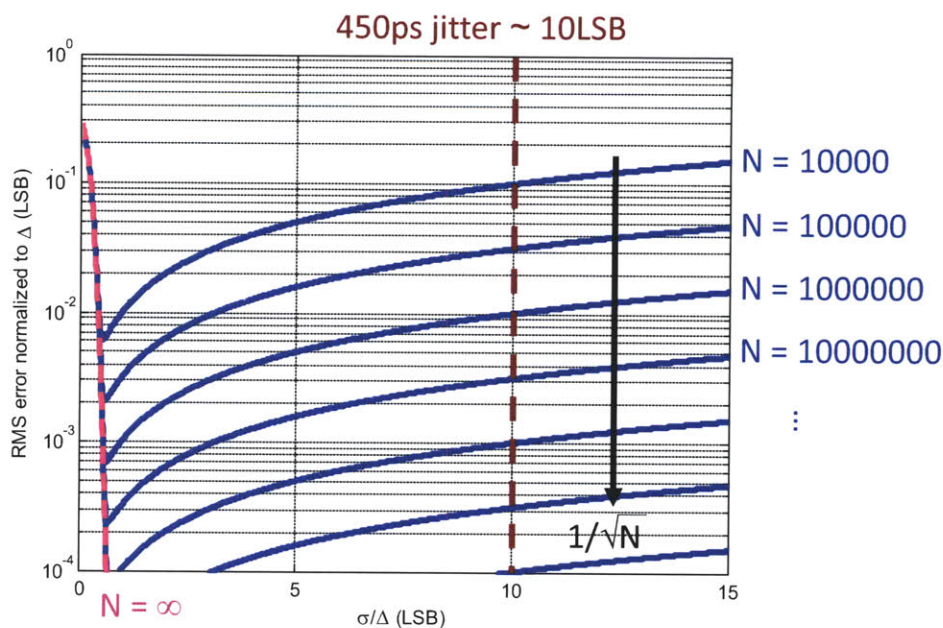
Figure 5-27: Plot of RMS error versus standard deviation σ of noise for different number of averaged samples. For LSB step size Δ equal to 45ps, 450ps of corresponds to 10LSB. For jitter of 10LSB, $1/\sqrt{N}$ reduction in RMS error holds for well over 1 million samples of averaging.

5.7 Intermediate Frequency for Digital Phase Tightening

The selection of the IF is an important design choice with various tradeoffs. One concern which is general to all heterodyne receivers is the image frequency. During the down-conversion process, signals at $\omega_{LO} + \omega_{IF}$ and $\omega_{LO} - \omega_{IF}$ are mixed down to the same frequency ω_{IF}. This is problematic because the image power is added to the signal power, which degrades the SNR. To ameliorate this effect, an image filter can be used, which is typically a bandpass filter that passes the signal while attenuates the image. The image filter becomes more difficult to implement as ω_{IF} decreases because it requires increasingly narrow passband filters. This is particularly challenging at MMW frequencies since the bandwidths are generally small compared to the MMW LO. Furthermore, high-Q passive devices that would be necessary for these narrow-band filters are difficult to achieve at MMW frequencies [30].

114

Another tradeoff occurs due to the limited resolution in the phase tightening system's delay-line imposed by the minimum inverter delay. This essentially fixes the time steps for a given technology, which does not scale as we select the IF. A higher IF will have fewer delay steps so there will be coarser phase resolution. A lower IF will have a longer period, so there will be more delay steps and finer phase resolution.

In the imaging system, we assume there will be a fixed sampling time for the image processing given the target framerate. Using digital phase tightening, the number of samples we obtain is proportional to the number of zero-crossings that are measured in that time frame. The IF is equivalent to the sampling rate, so a higher IF will have more samples while a lower IF will have fewer samples for the same window size. Therefore, the selection of the IF has a direct impact on the number of samples the system can average. The purpose of digital phase tightening is to reduce the error caused by noise sources to improve the phase measurement of the RF signal. We want to determine how the selection of the IF affects the error we obtain at the RF.

As described earlier in this chapter, the time delays scale up by ω_c/ω_{IF} when the received signal is mixed down to the IF. This includes the phase noise. A higher IF will have a lower scaling factor so the jitter caused by phase noise will not be increased as much as with a lower IF. We can expect lower levels of timing jitter if we increase the IF. This combined with the greater number of averaging for a given sampling window suggests that having a higher IF would be beneficial. However, if we use too high of an IF, not only do the circuits become more challenging to design, but the scaled phase noise may become small compared to the time delay step. At the extreme, the noise will become so small that no amount of averaging will reduce the error.

Figure 5-28 shows a Monte Carlo simulation of RMS error in seconds at the IF for varying IFs. In this simulation, the carrier frequency, sampling window, and time delay step were fixed to be 77GHz, 10ms, and 45ps, respectively, which are the values that will be used in the imaging system. The jitter at the carrier frequency and its bandwidth were set to be 1ps and 10MHz. From the plot, we can see that the RMS error decreases rapidly for increasing IF up to the phase noise bandwidth. The
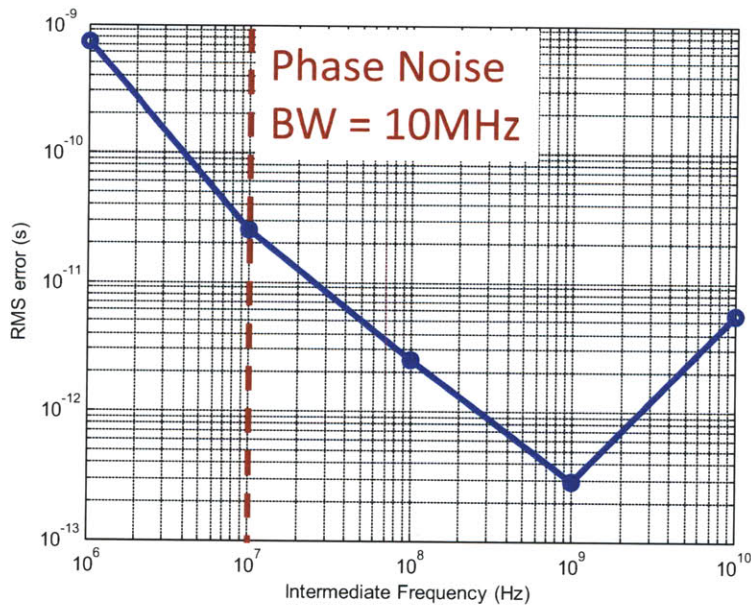
Figure 5-28: RMS error in seconds at the IF versus IF frequency.

decrease is expected because a higher IF means that the timing jitter at the carrier has a smaller scale ratio when it is down-converted to the IF. Increasing the IF also means more samples are taken to be averaged. At frequencies above the phase noise bandwidth up to around 1GHz, the RMS error decreases but at a lesser rate. The RMS error continues to drop because jitter from the carrier has a lower scale ratio for increasing IF, but the effect of averaging has been reduced because the noise is varying slowly compared to the sampling rate. Although increasing the IF allows more samples to be averaged, as discussed in the previous section, sampling much faster than the rate the noise is varying does not provide any benefit. After 1GHz, we see a sharp increase in the RMS error. This is due to scale ratio between the carrier and IF becoming so small that the jitter at the IF is now significantly smaller than the 45ps quantization level. The RMS error is now dominated by quantization noise. As mentioned before, the quantization steps are limited in a technology by the minimum inverter delay, so the quantization steps do not scale as the IF increases.

Figure 5-29 shows the RMS error from Figure 5-28 after it has been scaled back to the 77GHz carrier frequency. For IF below the phase noise bandwidth, we see the

116
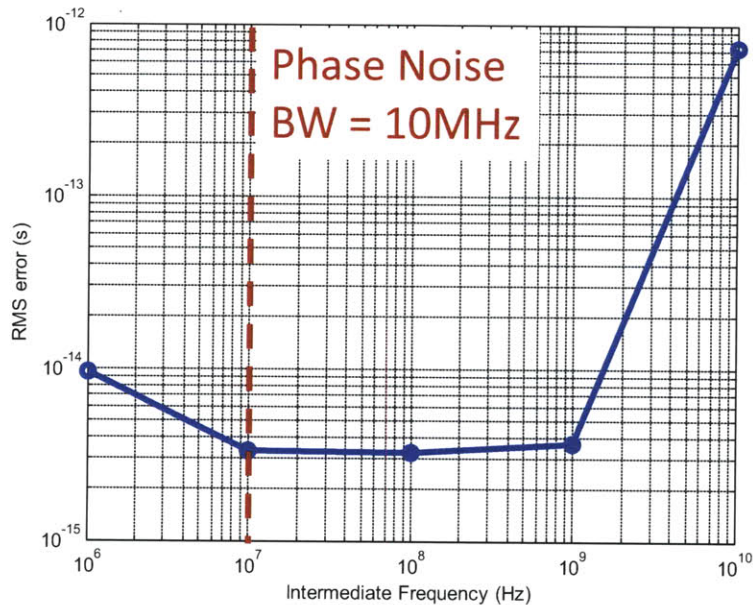
Figure 5-29: RMS error in seconds at the carrier frequency versus IF frequency.

reduction in RMS error due to averaging more samples for higher IF. After the IF exceeds the phase noise bandwidth, the plot becomes approximately flat because the averaging is no longer effective. The reduction in error that we saw in this region in Figure 5-28 is offset when we scale back to the carrier. After 1GHz, the RMS error at the RF increases significantly due to quantization noise.

Based on these simulations, the minimium RMS error at the carrier occurs when the IF is set within the region between the phase noise bandwidth and the point when the scaled jitter becomes smaller than the quantization level. Since the phase noise is dominated by the PLL, the IF should be set with respect to the bandwidth of the PLL. However, consideration must be taken for the fact that the PLL bandwidth will be very small compared to the MMW carrier, and designing a narrow image filter will be increasingly difficult as the IF is chosen to be smaller. Given the reference clock is 150MHz in the imaging receiver, the IF should be selected to be a multiple of this frequency for easier generation of the DLL clock. Therefore, the IF should be set to be between 150 to 300MHz.

5.8 Tracking Capability of Phase Tightening System

In this work, we made the assumption that the scene is changing slowly compared to the frame rate of the system, such that the received signal at each element has nearly constant phase and amplitude. However, it is instructive to consider the case in which the phase is not constant. We want to know how quickly the system can recover when the phase changes and how rapidly a signal can change and still be tracked by the system. If the received signal is varying rapidly, there will not only be less time to take samples for averaging, but the phase tightening system may lose the ability to track the signal altogether. For a 175MHz IF with 128 delay steps, it takes $0.73\mu s$ for the system to incrementally loop a full $360°$. The system cannot track signals that are varying more rapidly than $0.73\mu s$. We are interested in signals which are uncorrelated in this time frame, and we assume that the signal has a certain bandwidth. As described in Section 5.6.1, a signal with a rectangular frequency spectrum will have a sinc function for its time domain autocorrelation. We calculate that a signal with a single-sided bandwidth of 68MHz will be uncorrelated at $0.73\mu s$. If the signal has a bandwidth higher than 68MHz, then the phase tightening system's incremental adjustments will fail to track.

5.9 Summary

In this chapter, we presented a technique called digital phase tightening which is used to measure phase and reduce error caused by phase noise. We discussed the design of this system and its individual blocks. A prototype IC implementation was fabricated, and we measured and characterized its performance. We then analyzed digital phase tightening in a MMW imaging system, in which we take advantage of the slow imaging frame rate to average out the inherent noise from the system's PLL to improve the phase estimate. We achieved phase estimates with femtosecond level of accuracy at the MMW carrier.

Chapter 6

Conclusion

In this thesis, we discuss the advantages and usefulness of MMW imaging. With the advances in modern silicon technologies, it is possible to perform coherent imaging at MMW frequencies. This led to the proposal of an imaging system that uses digital beamforming to create an image. We then analyze the key circuit components in this system. The first was a MMW PLL that is needed at each element in the imaging array to generate an LO. The design and measurement of this PLL gave us an idea of the phase noise we can expect in the system. The second component was a technique that allowed us to measure the phase of a MMW signal. The technique was called digital phase tightening, and it leverages the large ratio between the MMW carrier frequency and the slow imaging framerate to estimate the phase of a MMW signal and reduce error caused by phase noise. Digital phase tightening allows for the separation of phase and amplitude estimation, both of which are needed for beamforming. Traditionally, these parameters are measured using I/Q demodulation either at the IF or at baseband. These methods require challenging specifications on the analog circuits and also significant digital processing. With digital phase tightening, the separation of the phase and amplitude measurements relaxes matching requirements and specifications on the analog circuits. We fabricated a chip that implemented this technique, which showed the phase estimation functionality and demonstrated how our estimates improved. They agreed with our analysis.

6.1 Contributions

There are several main contributions of this thesis. The first was the design and characterization of a MMW PLL. We examined key tradeoffs that were necessary to design a robust PLL with enough output power to drive a mixer in the imaging system. This work was published in [57]. The design and implementation of the digital phase tightening system was another contribution, which was presented in [58]. We studied its role in the antenna receiver and how it leads to an alternate architecture from the standard I/Q demodulation architectures. We demonstrated that we can reduce errors down to the femtosecond level and this technique may pave the way for future digital beamforming and MMW imaging systems.

6.2 Future Work

Comparisons can be made between digital phase tightening and $\Delta - \Sigma$ modulation. Both techniques use oversampling and a coarse quantizer with the intention of obtaining a higher resolution. It is feasible to imagine digital phase tightening that would use $\Delta - \Sigma$ modulation scheme. In Figure 5-1, the comparator output controlled a counter and table which were used to select the next DLL clock edge. In the chip implementation, the counter drove the MUX select directly such that the DLL clock edges were always incremented by plus or minus one. The table was simply a decoder that converted the counter's digital code into MUX select signals. However, we could use an alternate scheme that could replace the counter such that the next DLL clock edge is selected more intelligently. The averaging block could be replaced with a digital filter. We can imagine that the scheme used to select the next DLL clock edge performs some noise shaping that is attenuated by the digital filter, and this would give a more accurate measurement.

6.3   Concluding Remarks

We conclude this thesis with remarks about the effectiveness of digital phase tightening at the system level. In Section 3.2.4, we considered the difference between an architecture with separate phase and amplitude estimation and traditional I/Q demodulation. The main advantage of the separate estimation is the elimination of the I and Q branches which increases circuit complexity by requiring two identical paths and a quadrature LO. Furthermore, any mismatch between I and Q reduces the accuracy of the measurement. We then chose to separate the phase and amplitude measurements, which inevitably had its own set of challenges. Measuring the amplitude was clear, but measuring the phase was not so straightforward.
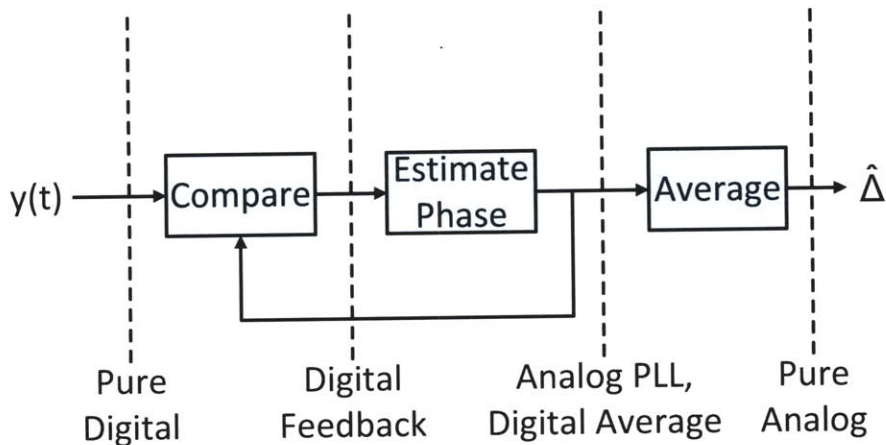
Figure 6-1: Block diagram of classical phase-locked loop architecture with interfaces for analog to digital conversions shown.

Digital phase tightening was our approach to estimate the phase while reducing phase noise error. In [34], Accardi mentions several architectures for phase estimation using a phase-locked loop, one of which was used as the reference for the digital phase tightening system. These architectures share the same basic components shown in Figure 6-1: a comparison block, a phase estimation block, and an averaging block. The comparison and phase estimation blocks form a feedback loop which takes in the input signal and outputs its phase. This output is then processed by the averaging block which reduces error in the estimate. These architectures shared the same components but differed in the placement of the analog to digital conversion among these

121

blocks. In the purely digital case, the input signal is digitized immediately and all of the steps are performed in the digital domain. Applying the digital conversion between the comparison and phase estimation blocks enables digital feedback and is the basis for digital phase tightening. Placing the ADC after either the phase estimation block or the averaging block requires a classical analog PLL followed by either digital or analog averaging of the phase information.

We have shown that digital phase tightening is a viable architecture for phase estimation. Unlike the alternate architectures, digital phase tightening does not require a demanding ADC with high speed or high resolution. A simple comparator is sufficient for the comparison operation. Most of the circuitry is digital which lends itself to technology scaling. Assuming we have already performed the down-conversion, digital phase tightening does not require a classical analog PLL at the IF, which reduces complexity as there is no need for a challenging VCO or consideration for the dynamics of a PLL. However, a PLL may be necessary for the down-conversion, but this PLL would be required in all four architectures. In terms of performance, the phase tightening system's resolution is determined by the time delay step in the delay line, which is limited by the minimum inverter delay of the process and should improve with scaling. These quantization steps can be large, and the system relies on the inherent noise of its components and averaging to improve the resolution. Depending on the application, the phase noise may be insufficient or there may not be enough time for averaging. In these cases, an alternative approach to phase estimation may be more suitable. But for MMW imaging where significant phase noise is added by the front-end and the frame rate is slow which allows adequate time for averaging, digital phase tightening is a very reasonable choice.

Bibliography

[1] M. Shoucri, G. Dow, B. Hauss, and L. Yujiri, "Passive millimeter wave camera for vehicle guidance in low visibility conditions," *Proceedings of SPIE, Synthetic Vision for Vehicle Guidance and Control*, vol. 2463, 1995.

[2] P. Racette, L. Dod, J. Shiue, R. Adler, D. Jackson, A. Gasiewski, and D. Zacharias, "Millimeter-wave imaging radiometer for cloud, precipitation, and atmospheric water vapor studies," *IEEE International Geoscience and Remote Sensing Symposium*, 1992.

[3] P. F. Goldsmith, C.-T. Hsieh, G. R. Huguenin, J. Kapitzky, and E. L. Moore, "Focal plane imaging systems for millimeter wavelengths," *IEEE Transactions on Microwave Theory and Techniques*, vol. 41, pp. 1664–1675, Oct. 1993.

[4] H. Wallace, "Millimeter-wave propagation measurements at the ballistic research laboratory," *IEEE Transactions on Geoscience and Remote Sensing*, May 1998.

[5] rfcafe.com, "Rf atmospheric absorption/ducting." [Online]. Available: http://www.rfcafe.com/references/electrical/ew-radar-handbook/rf-atmospheric-absorption-ducting.htm

[6] M. Born and E. Wolf, *Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light*, 8th ed. Cambridge University Press, 1999.

[7] National Academy of Sciences, *Assessment of Millimeter-wave and Terahertz Technology for Detection and Identification of Concealed Explosives and Weapons.* The National Academies Press, 2007.

[8] G. Gallerano, "Tera-hertz radiation in biological research, investigations on diagnostics and study on potential genotoxic effects," *IEEE Transactions on Instrumentation and Measurement,* June 2004. [Online]. Available: http://www.frascati.enea.it/THz-BRIDGE/reports/THz-BRIDGE%20Final%20Report.pdf

[9] J. Palmer and B. Grant, *The Art of Radiometry.* Society of Photo-Optical Instrumentation Engineers, 2010, ch. 3.

[10] L. V. Volkov, A. I. Voronko, N. L. Volkova, and A. R. Karapetyan, "Active MMW imaging technique for contraband detection," *Microwave Conference, 2003, 33rd European,* pp. 531–534, 2003.

[11] D. T. Petkie, C. Casto, F. C. De Lucia, S. R. Murrill, B. Redman, R. L. Espinola, C. C. Franck, E. L. Jacobs, S. T. Griffin, C. E. Halford, J. Reynolds, S. OBrien, and D. Tofsted, "Active and passive imaging in the THz spectral region: phenomenology, dynamic range, modes, and illumination," *Journal of Optical Society of America,* vol. 25, pp. 1523–1531, 2008.

[12] B. Systems, "Autonomous approach landing capability."

[13] E. Grossman and A. Miller, "Active millimeter-wave imaging for concealed weapons detection," *Proceedings of SPIE, Passive Millimeter-Wave Imaging Technology VI and Radar Sensor Technology VII,* 2003.

[14] S. Clark, J. Lovberg, C. Martin, and V. Kolinko, "Passive millimeter-wave imaging for airborne and security applications," *SPIE Conference on Passive Millimeter-Wave Imaging Technology VI and Radar Sensor Technology VII,* Apr. 2003.

[15] T. W. Du Bosq, J. M. Lopez-Alonso, and G. D. BoremanIbbott, "Millimeter wave imaging system for land mine detection," *Applied Optics*, vol. 45, pp. 5686–5692, Aug. 2006.

[16] R. H. Rasshofer, "Functional requirements of future automotive radar systems," *Proceedings of the 4th European Radar Conference*, pp. 259–262, Oct. 2007.

[17] M. Schneider, "Automotive radar - status and trends," *German Microwave Conference*, Apr. 2005.

[18] R. Troppmann and A. Bosch, "An introduction to adaptive cruise control (ACC)," *EE Times India*, Oct. 2008.

[19] W. J. Wilson, R. J. Howard, A. C. Ibbott, G. S. Parks, and W. B. Ricketts, "Millimeter-wave imaging sensor," *IEEE Transactions on Microwave Theory and Techniques*, vol. MTT-34, pp. 1026–1035, Oct. 1986.

[20] J. P. Hollinger, J. E. Kenney, and B. E. Troy, Jr., "A versatile millimeter-wave imaging system," *IEEE Transactions on Microwave Theory and Techniques*, vol. MTT-24, pp. 786–793, Oct. 1976.

[21] G. Packeiser, "III/V semiconductors: Substrate materials and epitaxy," *Advanced Materials*, pp. 464–471, 1990.

[22] L. Raffaelli, E. Stewart, R. Quimby, J. Borelli, A. Geissberger, and D. Palmieri, "A low cost 77 GHz monolithic transmitter for automotive collision avoidance systems," *IEEE Microwave and Millimeter-Wave Monolithic Circuits Symposium*, pp. 63–66, 1993.

[23] M. Shoucri, G. Dow, B. Hauss, and L. Yujiri, "Passive millimeter wave camera for enhanced vision systems," *Proceedings of SPIE, Enhanced and Synthetic Vision*, vol. 2736, 1996.

[24] Y.-W. Chang and M. Johnson, "Portable concealed weapon detection using millimeter wave fmcw radar imaging," *National Institute of Justice Office of Science and Technology*, Aug. 2001, grant No. 98-DT-CX-K001.

[25] A. Tomkins, P. Garcia, and S. Voinigescu, "A passive W-band imager in 65nm bulk CMOS," *IEEE Compound Seminoconductor Integrated Circuit Symposium*, Oct. 2009.

[26] L. Gilreath, V. Jain, H.-C. Yao, L. Zheng, and P. Heydari, "A 94-ghz passive imaging receiver using a balanced LNA with embedded Dicke switch," *IEEE Radio Frequency Integrated Circuits Symposium*, May 2010.

[27] A. Hajimiri, A. Komijani, A. Natarajan, R. Chunara, X. Guan, and H. Hashemi, "Phased arary systems in silicon," *Communication Magazine, IEEE*, vol. 42, no. 8, pp. 122 – 130, Aug. 2004.

[28] H. Van Trees, *Optimum Array Processing. Part IV of Detection, Estimation, and Modulation Theory.* John Wiley & Sons, Inc., 2002, pp. 47–49.

[29] G. M. Rebeiz and K.-J. Koh, "Silicon RFICs for phased arrays," *IEEE Microwave Magazine*, pp. 96–103, May 2009.

[30] B. Razavi, *RF Microelectronics.* Prentice Hall PTR, 2008.

[31] L. Brooks and H.-S. Lee, "A 12b, 50 MS/s, fully differential zero-crossing based pipelined ADC," *IEEE Journal of Solid-State Circuits*, Dec. 2009.

[32] J. Chu, L. Brooks, and H.-S. Lee, "A zero-crossing based 12b 100MS/s pipelined ADC with adaptive biasing," *IEEE VLSI Circuits, 2010 IEEE Symposium on*, 2010.

[33] J. Powell, H. Kim, and C. Sodini, "SiGe receiver front ends for millimeter-wave passive imaging," *IEEE Transactions on Microwave Theory and Techniques*, pp. 2416–2425, Nov. 2008.

[34] A. Accardi, "Generating pictures from waves: Aspects of image formation," Ph.D. dissertation, Massachusetts Institute of Technology, Feb. 2010.

[35] J. G. Ryan, "Criterion for the minimum source distance at which plane-wave beamforming can be applied," *Journal of Acoustics Society of America*, pp. 595–598, July 1998.

[36] F. Folster, H. Rohling, and U. Lubbert, "An automotive radar network based on 77GHz FMCW sensors," *IEEE International Radar Conference*, pp. 871–876, May 2005.

[37] F. Kuroki, K. Yamaoka, Y. Murata, S. Ishikawa, and K. Masaki, "Range finding by using NRD guide pulse radar front-end at 60GHz," *Proceedings of the 3rd European Radar Conference*, pp. 292–295, Sept. 2006.

[38] H. Rohling and C. Moller, "Radar waveform for automotive radar systems and applications," *IEEE Radar Conference*, pp. 1–4, May 2008.

[39] K. Thomenius, "Evolution of ultrasound beamformers," *IEEE Ultrasonic Symposium*, pp. 1615–1622, Nov. 1996.

[40] F. Gardner, *Phaselock Techniques*, 3rd ed. John Wiley & Sons, Inc., 2005, ch. 2.

[41] ——, *Phaselock Techniques*, 3rd ed. John Wiley & Sons, Inc., 2005, ch. 10.

[42] M. Perrott, "Massachusetts institute of technology. 6.776 high speed communication circuits course notes lecture 21," 2005.

[43] T. Lee and A. Hajimiri, "Oscillator phase noise: A tutorial," *IEEE Journal of Solid-State Circuits*, Mar. 2000.

[44] A. Demir, "Computing timing jitter from phase noise spectra for oscillators and phase-locked loops with white and 1/f noise," *IEEE Transactions on Circuits and Systems*, pp. 1869–1884, Sept. 2006.

[45] H. Hoshino, R. Tachibana, T. Mitomo, N. Ono, Y. Yoshihara, and F. R., "A 60-GHz phase-locked loop with inductor-less wide operation range prescaler in 90-nm CMOS," *IEEE Journal of Solid State Circuits*, vol. 44, pp. 785–791, June 2009.

[46] K. Scheir, G. Vandersteen, Y. Rolain, and P. Wambacq, "A 57-to-66GHz quadrature PLL in 45nm digital CMOS," *IEEE International Solid-State Circuits Conference*, 2009.

[47] J. Lee, M. Liu, and H. Wang, "A 75-GHz phase-locked loop in 90-nm CMOS technology," *IEEE Journal of Solid State Circuits*, vol. 43, pp. 1414–1426, June 2008.

[48] V. Jain, B. Javid, and P. Heydari, "A BiCMOS dual-band millimeter-wave frequency synthesizer for automotive radars," *IEEE Journal of Solid State Circuits*, vol. 44, pp. 2100–2113, Aug. 2009.

[49] J. Jeong and Y. Kwon, "A fully integrated V-band PLL MMIC using 0.15-μm GaAs pHEMT technology," *IEEE Journal of Solid-State Circuits*, pp. 1042–1050, May 2006.

[50] A. Technologies, "Rf design environment momentum 2008 update 1."

[51] W. Rhee, "Design of high-performance CMOS charge pumps in phase-locked loops," *Proceedings of the IEEE International Symposium on Circuits and Systems*, pp. 545–548, June 1999.

[52] M. Perrott, "Cppsim behavioral system simulator." [Online]. Available: http://www.cppsim.com

[53] M. I. Products, "Clock (clk) jitter and phase noise conversion," 2004, application Note 3359.

[54] M. Cross, 2009, private communication.

[55] M. G. Johnson and E. L. Hudson, "A variable delay line PLL for CPU-coprocessor synchronization," *IEEE Journal of Solid State Circuits*, vol. 23, pp. 1218–1223, Oct. 1988.

[56] R. Skartlien and L. Oyehaug, "Quantization error and resolution in ensemble averaged data with noise," *IEEE Transactions on Instrumentation and Measurement*, pp. 1303–1312, June 2005.

[57] K. Nguyen, H. Kim, and C. Sodini, "A 76GHz PLL for mm-wave imaging applications," *IEEE Microwave Symposium Digest*, May 2010.

[58] K. Nguyen, A. Accardi, H. Kim, G. Wornell, and C. Sodini, "Digital phase tightening for millimeter-wave imaging," *IEEE Custom Integrated Circuits Conference*, Sept. 2010.