

UC Santa Barbara

UC Santa Barbara Electronic Theses and Dissertations

Title

A Versatile Mixed-Signal Controller for Optoelectronic Frequency Synthesis

Permalink

<https://escholarship.org/uc/item/38k2z5kj>

Author

Bluestone, Aaron James

Publication Date

2019

Peer reviewed|Thesis/dissertation

University of California
Santa Barbara

A Versatile Mixed-Signal Controller for Optoelectronic Frequency Synthesis

A dissertation submitted in partial satisfaction
of the requirements for the degree

Doctor of Philosophy

in

Electrical and Computer Engineering

by

Aaron James Bluestone

Committee in charge:

Professor Luke S. Theogarajan, Chair
Professor John E. Bowers
Professor Forrest D. Brewer
Professor James F. Buckwalter
Professor João P. Hespanha

December 2019

The Dissertation of Aaron James Bluestone is approved.

Professor John E. Bowers

Professor Forrest D. Brewer

Professor James F. Buckwalter

Professor João P. Hespanha

Professor Luke S. Theogarajan, Committee Chair

June 2019

Acknowledgements

First and foremost, this work would not have been possible without the unwavering support and guidance from my advisor, Professor Luke Theogarajan. When I started this arduous process I had no idea what I was getting into, I just knew from his classes that he would push me past my limits any time I got comfortable. He has become the greatest mentor in my life and I still strive for his knowledge, passion, and work ethic. Some of my favorite memories from UCSB will always be the times we sat in his office bouncing ideas off of each other, coming up with outrageous circuits and architectures. He is a true friend that cares about my well being, and even a therapist at the times that I've needed one most. I will always be grateful for the relationship I've formed with him.

I would like to thank my committee members for their time and consideration throughout my doctorate. Professor John Bowers has been an exemplary PI on numerous optoelectronics projects I've been a part of. He often made his lab space, equipment, and students available to me in order to perform extensive measurements. Professor Forrest Brewer has always had an open door to provide feedback on my research and his valuable discussions never cease to amaze me. Professor Jim Buckwalter's expertise provided a fresh perspective on the architecture I was aiming to design. I thoroughly enjoyed his courses and his insight to the RF industry. Despite having never met me, Professor João Hespanha graciously accepted an offer to join my committee. He helped me create an image of what the complex controls of this system could someday look like. While I was not able to see this to fruition, I am thankful for his input and I think he will be a vital asset in the future of this project.

Throughout the years I have relied upon the support of countless staff members in the ECE Department and the Institute for Energy Efficiency. In particular I would like to thank Val De Veyra, Kelsey Ibach, Libby Straight, Beth English, Joelle Dohrman,

Paul Gritt, Bear, Avery Juan, Kaitlyn LeGros, Katherine Grayson, and Jane Allen.

I sincerely appreciate all of my labmates in the Biomimetic Circuits and Nanosystems Group. Luis Chen, Advait Madhavan, and Melika Payvand provided valuable guidance when I was just starting out. Sarah Grundeen, Avantika Sodhi, Justin Rofeh, Michael Isaacman, Shahab Mortezaei, Mitra Saeidi, Alex Nguyen-Le, Rebecca Hwang, and Wyatt Rodgers all contributed to a lively office atmosphere filled with stimulating discussions. In particular I would like to single out Akshar Jain, my right-hand man on this project for the past several years. Over countless all nighters and DARPA deadlines I have enjoyed watching his skillset skyrocket, and I'm excited to see where he takes this project. I'd also like to thank our undergraduate intern, Nancy Kaveh, for her assistance on numerous occasions.

My colleagues on the E-PHI and DODOS projects - including collaborators from NIST Boulder, Aurrion (Juniper Networks) and the Bowers group - were instrumental in the development of my photonic skillset. I'd like to acknowledge Tin Komljenovic, Daryl Spencer, Sudharsanan Srinivasan, Erik Norberg, Greg Fish, Laura Sinclair, Travis Briles, Jordan Stone, Scott Papp, and Scott Diddams.

On a personal note, I am forever thankful for my parents who have been my biggest supporters in all my academic endeavors. I'd also like to acknowledge my close friends Stef and DJ for going through every step of grad school with me. Lastly, Jessica Hai has been my Dwayne Johnson along this emotional roller coaster. She has kept me sane during the hardest of times, she has become my closest companion, and her drive continues to be a constant source of inspiration. I couldn't have reached this milestone without her.

Curriculum Vitæ

Aaron James Bluestone

Education

- 2019 Ph.D. in Electrical and Computer Engineering (Expected),
University of California, Santa Barbara.
- 2016 M.S. in Electrical and Computer Engineering,
University of California, Santa Barbara.
- 2012 B.S. in Electrical Engineering,
University of California, Santa Barbara.

Awards

- Outstanding Teaching Assistant of the Year, 2012-2013
- Outstanding Teaching Assistant of the Year, 2014-2015

Patent

- K. Veeder, **A. Bluestone**, N. Dhawan, and C. Trzebiatowski, “Dynamic Resistance Element Analog Counter,” *U.S.P.T.O. Patent Pending*, 2018.

Publications

- D. Spencer, T. Drake, T. Briles, J. Stone, L. Sinclair, C. Fredrick, Q. Li, D. Westly, R. Illic, **A. Bluestone**, N. Volet, T. Komljenovic, L. Chang, S. Lee, D. Oh, M. Suh, K. Yang, M. Pfeiffer, T. Kippenberg, E. Norberg, L. Theogarajan, K. Vahala, N. Newbury, K. Srinivasan, J. Bowers, S. Diddams, and S. Papp, “An optical-frequency synthesizer using integrated photonics,” *Nature*, vol. 557, no. 7703, pp. 81-85, 2018.
- **A. Bluestone**, A. Jain, N. Volet, D. Spencer, S. Papp, S. Diddams, J. Bowers, and L. Theogarajan, “Heterodyne-based hybrid controller for wide dynamic range optoelectronic frequency synthesis,” *Optics Express*, vol. 25, no. 23, pp. 29086-29097, 2017.
- D. Spencer, T. Briles, T. Drake, J. Stone, R. Ilic, Q. Li, L. Sinclair, D. Westly, N. Newbury, K. Srinivasan, S. Diddams, S. Papp, **A. Bluestone**, T. Komljenovic, N. Volet, L. Theogarajan, J. Bowers, M. Suh, K. Yang, S. Lee, D. Oh, K. Vahala, M. Pfeiffer, T. Kippenberg, and E. Norberg, “Full stabilization and control of an integrated photonics optical frequency synthesizer,” *IEEE Photonics Conference (IPC)*, pp. 341-342, 2017.
- **A. Bluestone**, R. Kaveh, and L. Theogarajan, “An analog phase prediction based fractional-N PLL,” *IEEE International Symposium on Circuits and Systems (IS-CAS)*, pp. 1-4, 2017.

- D. Spencer, **A. Bluestone**, J. Bowers, T. Briles, S. Diddams, T. Drake, R. Ilic, T. Kippenberg, T. Komljenovic, S. Lee, Q. Li, N. Newbury, E. Norberg, D. Oh, S. Papp, M. Pfeiffer, L. Sinclair, K. Srinivasan, J. Stone, M. Suh, L. Theogarajan, K. Vahala, N. Volet, D. Westly, and K. Yang, “Towards an integrated-photonics optical-frequency synthesizer with <1 Hz residual frequency noise,” *Optical Fiber Communications Conference and Exhibition (OFC)*, pp. 1-3, 2017.
- J. Bowers, A. Beling, D. Blumenthal, **A. Bluestone**, S. Bowers, T. Briles, L. Chang, S. Diddams, G. Fish, H. Guo, T. Kippenberg, T. Komljenovic, E. Norberg, S. Papp, M. Pfeiffer, K. Srinivasan, L. Theogarajan, K. Vahala, and N. Volet, “Chip-scale optical resonator enabled synthesizer (CORES) miniature systems for optical frequency synthesis,” *IEEE International Frequency Control Symposium (IFCS)*, pp. 1-5, 2016.
- **A. Bluestone**, “An Analog Phase Interpolation Based Fractional-N PLL,” *UC Santa Barbara Masters Thesis*, 2016.
- **A. Bluestone**, D. Spencer, S. Srinivasan, D. Guerra, J. Bowers, and L. Theogarajan, “An ultra-low phase-noise 20-GHz PLL utilizing an optoelectronic voltage-controlled oscillator,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 63, no. 3, pp. 1046-1052, 2015.
- D. Spencer, S. Srinivasan, **A. Bluestone**, D. Guerra, L. Theogarajan, and J. Bowers, “A low phase noise dual loop optoelectronic oscillator as a voltage controlled oscillator with phase locked loop,” *IEEE Photonics Conference (IPC)*, pp. 412-413, 2014.

Abstract

A Versatile Mixed-Signal Controller for Optoelectronic Frequency Synthesis

by

Aaron James Bluestone

Self-referenced optical combs have proven pivotal for numerous metrology applications including precision navigation, LiDAR, and molecular spectroscopy. While plenty of research has improved and broadened the scope of this instrument, most implementations to date have been lab-scale setups that require kilowatts of power. The size, weight and power (SWaP) needs to shrink in order to fully realize the potential of this technology.

Silicon chip-based Photonic Integrated Circuits (PICs) provide a platform to exploit the same optical phenomena in a reduced SWaP. However, these miniature devices are inherently prone to fabrication variation and environmental fluctuations during operation. An associated issue is the coupling of precision and power. Where bench top implementations can circumvent many issues by increasing the laser power to mitigate downstream losses in optical elements, an integrated solution requires novel electronic signal estimation, detection and stabilization architectures to maintain precision under a low power budget.

This dissertation presents a mixed-signal controller designed to handle the challenges of achieving parts-per-trillion frequency stability in an integrated optoelectronic frequency synthesizer. I discuss the development of a heterodyne-based architecture and highlight experimental results from a PCB prototype using commercial-off-the-shelf electronics. The limitations present at the board-level are further mitigated by the development of a custom IC. Results from an application-specific integrated circuit (ASIC) designed in 55nm CMOS show the potential of integration in reducing the SWaP. Ul-

timately, this architecture achieves state-of-the-art performance, producing a 193 THz output with 5.6 mHz average deviation ($2.9\text{e-}17$ ADEV @ 1000s). The synthesizer is tunable >40 nm across the C-band with 745 mHz setpoint resolution, capable of full configurability in real-time via a custom Graphical User Interface (GUI).

Contents

Curriculum Vitae	v
Abstract	vii
1 Introduction	1
2 System Overview	5
2.1 Offset-Locking Architecture	5
2.2 Heterodyne Receiver	8
2.3 Tunable Laser	9
2.4 Communication Hierarchy	11
2.5 Loop Bandwidth	13
3 Board-Level Implementation	16
3.1 Digital Phase Detection and Error Correction	18
3.2 Low-Noise Tunable Laser Drivers	21
3.3 Unlocked Frequency Jitter	23
3.4 Scan-And-Lock Algorithm	25
3.5 Notable Results	27
3.6 Board-Level Limitations	34
4 Integrated Circuit Implementation	36
4.1 Widely Tunable Phase-Locked Loop	38
4.2 Gilbert Cell Mixer	42
4.3 Baseband Amplifier	43
4.4 Limiting Amplifier	45
4.5 Time-to-Digital Converter	46
4.6 Peak Detector	51
4.7 Flash ADC	53
4.8 Injection-Locked Frequency Divider	54
4.9 Digital Interfaces	56
4.10 IC Fabrication and Results	57

5	Future Work	62
6	Conclusion	66
A	Source Code	68
A.1	Verilog Code	68
A.2	Processor C Code	79
A.3	GUI VB Code	81
	Bibliography	82

Chapter 1

Introduction

Frequency synthesis is ubiquitous in modern electronics. A technology that gained traction for discrete wireless communications during World War II, quickly progressed to everyday use in radios, television receivers, and is now at the heart of standards we rely on daily (WiFi, Bluetooth, cellular networks, etc.). Frequency synthesizers are a core building block to numerous metrology systems, such as GPS and radar, which rely upon the frequency and timing certainty achieved through RF synthesis. The number of applications can only go up in this era of IoT.

All of these systems are made possible by upconverting the stability of a fixed reference oscillator to a desired output frequency, typically a much faster oscillation. This can be abstractly thought of as a mechanical gearbox. One full revolution of the reference - a full 2π of phase rotation - translates to many more at the output, dictated by the gear ratio. In a synthesizer, the gear ratio is often programmable so that the output frequency can be tuned to various frequencies, or channels. An electronic feedback loop, most commonly a phase-locked loop (PLL), will adjust the output oscillator faster or slower to achieve the desired gear ratio [1].

An optical frequency synthesizer operates off of the same principal, with a much

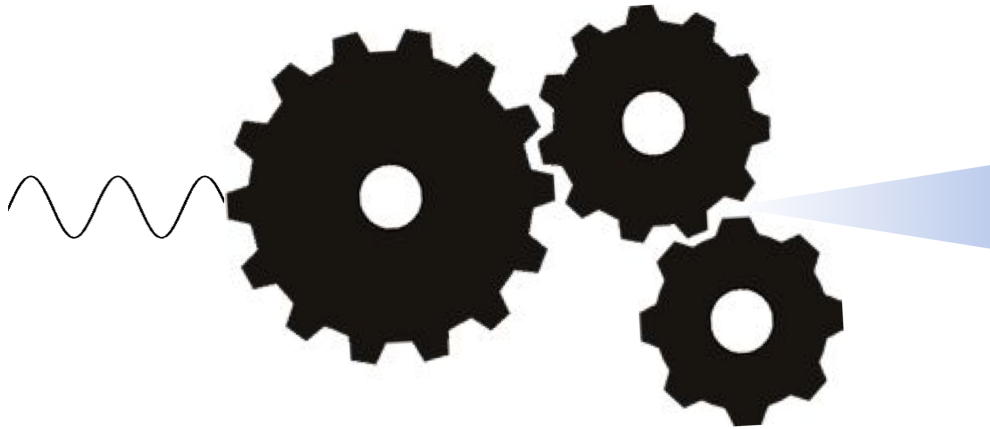


Figure 1.1: Abstract representation of optical frequency synthesis.

higher gear ratio leading to an output in the light spectrum (i.e. 10 MHz reference \rightarrow 200 THz output). Compared to radio frequency synthesis, this brings an added challenge in that the optical frequency cannot be directly processed by conventional electronics. Self-referenced laser frequency combs provide a method to translate frequency content from the optical to RF domain and vice versa [2, 3, 4]. Since this critical link was first exploited in 2000, numerous applications have appeared in metrology and surrounding fields.

Light detection and ranging (LIDAR) is used to map the topography of an area by sending beams of pulsed light and measuring the reflections [7, 8]. The precision of the

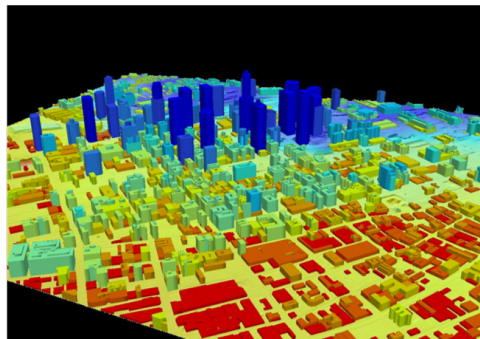


Figure 1.2: 3D map produced by light detection and ranging (LIDAR). Image taken from NASA Langley Research Center - Remote Sensing Branch [5].

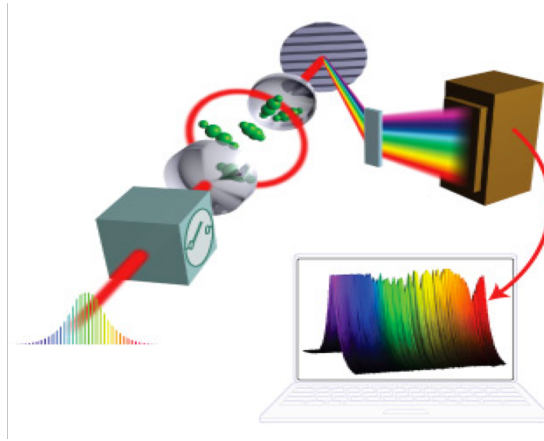


Figure 1.3: Diagram of molecular spectroscopy. Image taken from JILA - University of Colorado Boulder [6].

resulting 3D map is correlated to the wavelength and timing uncertainty of the optical source, meaning it can greatly benefit from the stability of a synthesizer. Molecular spectroscopy works by sweeping the wavelength of a laser pointed at an object, and measuring the optical power on the other side. Peaks and troughs will result from various atomic interactions that are known to take place at particular wavelengths. This can be used to identify potential toxins in gaseous environments as well as biomedical tracers, either noninvasively or via biopsy. Optical communications will also play a vital role in the advancement of positioning, navigation, and timing (P.N.T.). GPS is utilized in every facet of our movement, from airplanes all the way down wristwatches, though the protocol has several shortcomings that can be improved through optical communications. Precise control of the channel via frequency synthesis will allow discrete P.N.T. in military applications, robust against third-party listeners and jamming mechanisms (Low Probability of Intercept / Low Probability of Detection)[9]. Civilian uses may take advantage of the technology in areas where GPS communications are currently degraded or unavailable, whether it be underground, underwater, or in a thunderstorm. This is sure to improve next-gen applications such as self-driving cars and automated supply chain

management [10, 11].

Researchers have shown that all of these applications are possible through optical frequency synthesis, yet the implementations to date require kilowatts of power and often occupy entire lab spaces. Recent works have shown that waveguide microresonators can exploit a Kerr nonlinearity to generate frequency combs in a millimeter-sized footprint [12, 13, 14, 15]. Indeed these can be self-referenced to a microwave oscillator [16, 17, 18, 19]. Explained later in great detail, one major caveat in this frequency link comes from the optical comb spreading power across hundreds of comb teeth, leaving a small fraction of the power for the single comb tooth of interest in a given application. These systems use significant optical and electrical amplification to detect the desired signal, typically provided by server rack equipment. The goal of this research is to shrink the optical frequency synthesizer, creating a low-power implementation that leverages novel photonic integrated circuits and electronic integrated circuits. This will enable new applications for the synthesizer as well as improve those already known.

This dissertation will focus on the electronics used to detect and control various signals in the optical frequency synthesizer. Chapter 2 introduces some of the key challenges and the architecture designed to overcome them. Chapter 3 highlights a board-level prototype which was beneficial to gain familiarity with the synthesizer and its nuances. Building upon the lessons learned and limitations at the board-level, Chapter 4 details the integrated circuit design. Finally, I will discuss where this research is headed and conclude the paper.

Chapter 2

System Overview

2.1 Offset-Locking Architecture

An optical heterodyne detection is a common method for translating optical frequencies to the microwave domain [20]. This technique compares one light signal with a known reference source by combining their lights on a photodiode detector, and the difference in frequencies is detected in the electrical domain. This provides a demodulation scheme for numerous optical communication standards that utilize phase / frequency modulation to encode data.

When used in synthesis applications, the goal is to control a tunable laser (TL) emission to a fixed, desired wavelength. By heterodyning the TL light with an optical reference of precisely known wavelength, the difference in their frequencies, referred to as f_{offset} , and can be detected and stabilized to a fixed, desired frequency. Shifting the signal to the RF domain allows implementation of a phase-locked-loop (PLL), a well defined feedback loop that has been utilized for decades to provide parts-per-trillion frequency precision and stability. The main drawback to this form of optical offset-locking is a limitation in locking range to offset frequencies that can be detected and handled by the

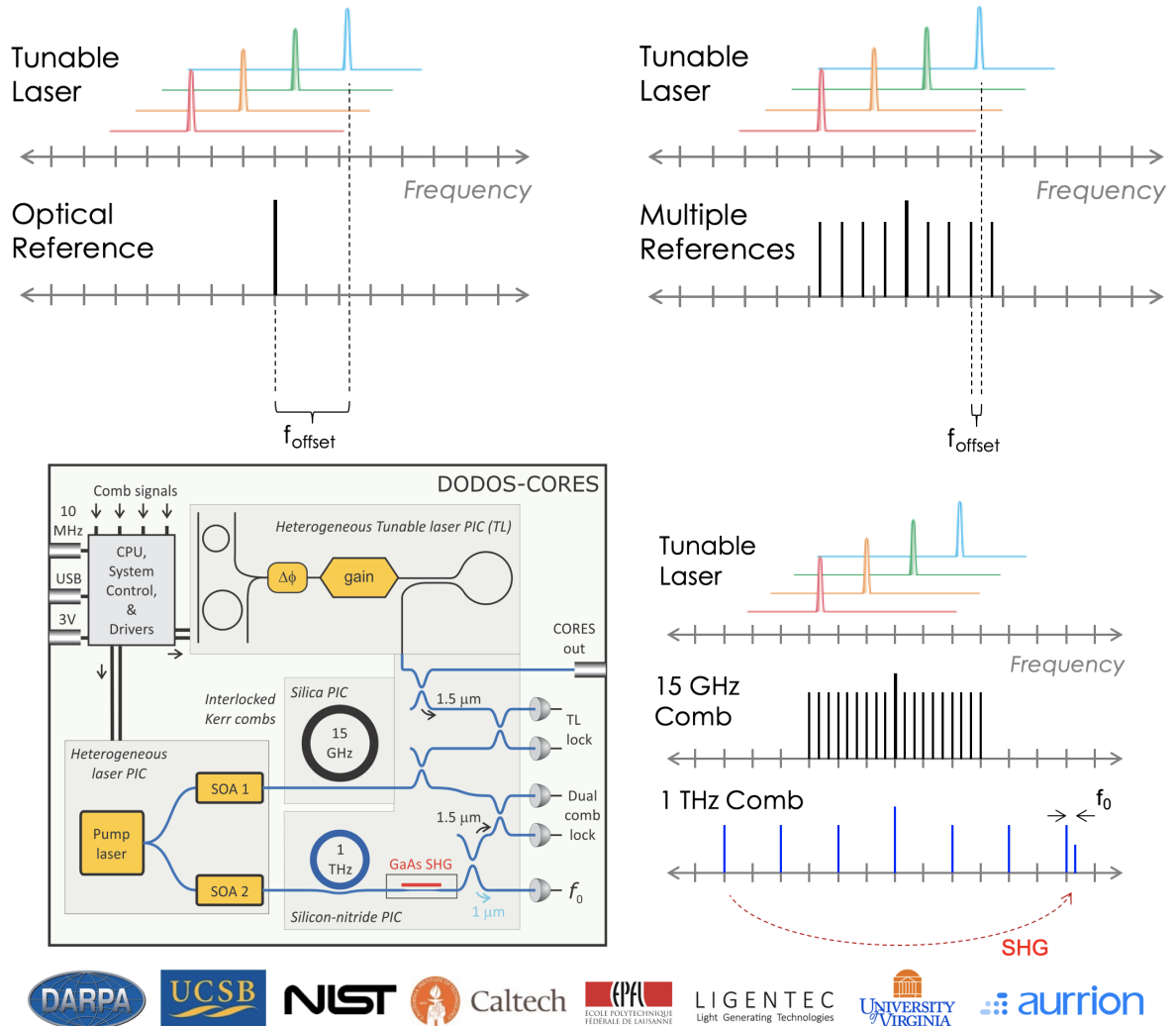


Figure 2.1: Overview of the offset-locking architecture, as well as the CORES self-referenced comb system.

underlying electronics. For modern CMOS technologies, the PLL design becomes non-trivial above several gigahertz. As mentioned previously, a Kerr frequency comb exploits a nonlinear phenomenon to produce a multitude of optical frequencies with fixed spacing. Illustrated in Figure 2.1, each of these comb teeth can be utilized as optical references, facilitating manageable offset frequencies over a much larger optical frequency range.

With funding from DARPA our research team targeted such a synthesizer, which also included locking the repetition rates of the combs (15 GHz and 1 THz) and self-

tracking the frequency of the comb's pump laser (f_{Pump}). The overall project (bottom of Fig. 2.1) is an immense collaboration involving 9 research teams across the globe. The directive involved fabrication of all of the low-power photonics including two wideband Kerr frequency combs, an optical frequency doubler, the pump laser and tunable laser, the broadband photodiodes, the electronics to detect and phase-lock the various RF signals, and most importantly, a collaborative effort to integrate everything in a low-power, minimal footprint. Our self-referencing scheme is published in great detail [21, 22, 23], and an abridged explanation follows.

Assuming it is known which comb-tooth the TL is closest to (the N^{th}), the frequency of the tunable laser will be:

$$f_{TL} = f_{Pump} + N(15GHz) + f_{offset} \quad (2.1)$$

The remaining challenge is finding f_{Pump} . The 1 THz comb, also powered by f_{pump} , spans all the way 999 - 1998 nm. A waveguide periodically poled lithium niobate performs second harmonic generation (SHG) with a nonlinear phenomenon, taking in the 1998nm comb tooth and producing another signal at 999 nm [24]. An optical heterodyne is used to compare the two tones at 999 nm and determine f_{Pump} . With the 999 nm and 1998 nm comb teeth defined as the K^{th} and M^{th} respectively:

$$\begin{aligned} f_{het} &= f_{Pump} + K(1THz) - 2(f_{Pump} - M(1THz)) \\ f_{Pump} &= (K + 2M)(1THz) - f_{het} \end{aligned} \quad (2.2)$$

2.2 Heterodyne Receiver

This dissertation will focus solely on the synthesis aspect, targeting a C-band TL that can lock to the 15 GHz spaced comb teeth across a >5 THz span. The frequency comb allows continuous detection and locking across that span although it brings along new architectural challenges, especially when operating in a low-power regime. Specifically the signal-to-noise ratio (SNR) of the f_{offset} signal becomes a critical design constraint. For a frequency comb that produces N number of comb teeth, each tooth has roughly $1/N$ optical power relative to the comb input pump. Furthermore, at any given frequency only one comb tooth is utilized for the optical heterodyne and all of the other comb teeth ($N-1$) contribute stray optical power into the photodetector. The shot noise of a photodetector is directly proportional to the optical power input and therefore the SNR degrades as N increases.

With a 15 GHz comb spacing, the nearest comb tooth (and therefore f_{offset}) could be anywhere DC to 7.5 GHz. If a detector were to look for f_{offset} in this bandwidth directly, the noise would swamp out the signal, and so filtering the noise is a necessity. Early approaches looked at analog filtering to reduce the noise bandwidth. This would need to be a tunable bandpass filter, such as an N-Path [25], to move anywhere that f_{offset}

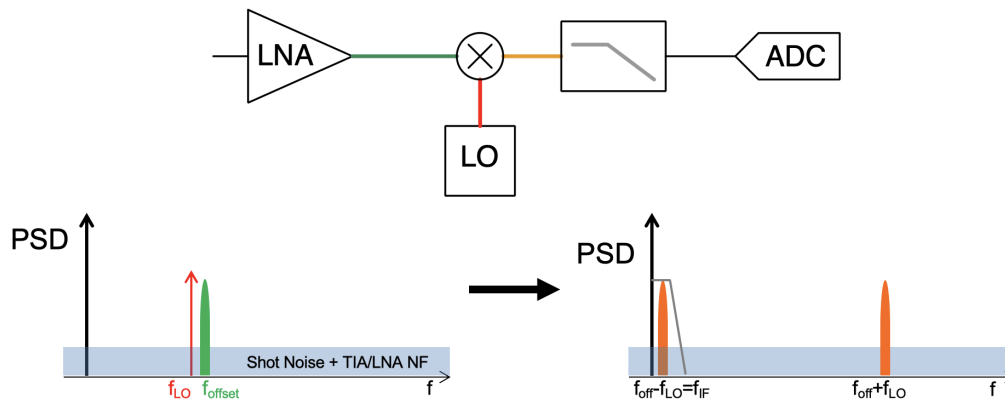


Figure 2.2: Heterodyne down-conversion receiver.

is desired to lock. However, this would still require a form of signal detection that works up to 7.5 GHz. Any phase detector or analog-to-digital converter at that speed would be unable to achieve the resolution desired in this system. Therefore, a heterodyne-based receiver is considered to address both the noise bandwidth and converter limitations.

Shown in Figure 2.2, a local oscillator signal, f_{LO} , is generated and combined with f_{offset} in an analog mixer. The mixer's properties (mathematically proven in Section 4.3), dictate that the output produces frequency content at $f_{offset} - f_{LO}$ and $f_{offset} + f_{LO}$.

f_{offset} is tuned to a frequency nearby f_{LO} , to enable tracking of the phase of an intermediate frequency, $f_{IF} = f_{offset} - f_{LO}$. By bringing this down to ~ 25 MHz, the noise bandwidth is easily reduced with a low-pass filter, and the conversion to the digital domain is obtained with high resolution.

2.3 Tunable Laser

Even with perfect detection of f_{offset} and precise calculation of the phase error, completing the feedback loop and controlling the tunable laser is not a trivial task. Consider a conventional voltage-controlled oscillator, where the voltage input, V_{tune} , is linearly proportional (approximately) to the output frequency. The tunable laser is capable of covering >5 THz, a single tuning knob like this could never have the dynamic range to cover the full span and still achieve the desired Hz-level precision. The accuracy would be fundamentally limited by voltage noise on the V_{tune} line. Instead, this laser utilizes several tuning knobs, both coarse and fine, to achieve its wideband performance.

The tunable laser, fabricated by Aurrion, Inc., emits up to 4 mW of CW light coupled into a fiber. As seen in Figure 2.4, the TL contains a gain section, a phase section and two micro-rings. An SOA also provides an on-chip small-signal gain >10 dB. The gain section and the SOA consist of InP-based quantum wells heterogeneously integrated on a silicon

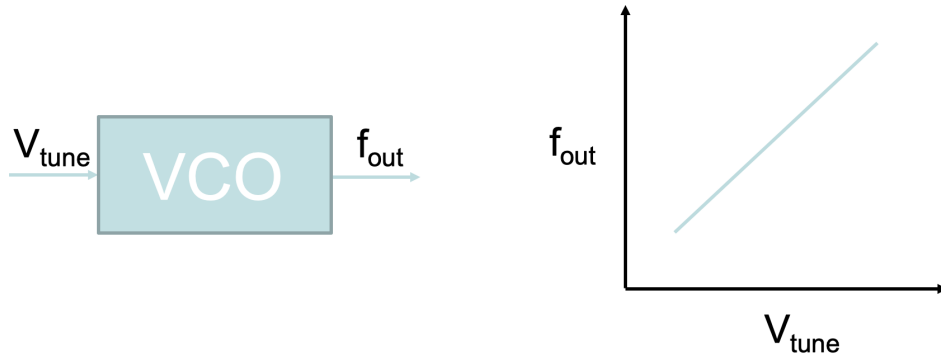


Figure 2.3: Simplified diagram of a conventional voltage-controlled oscillator.

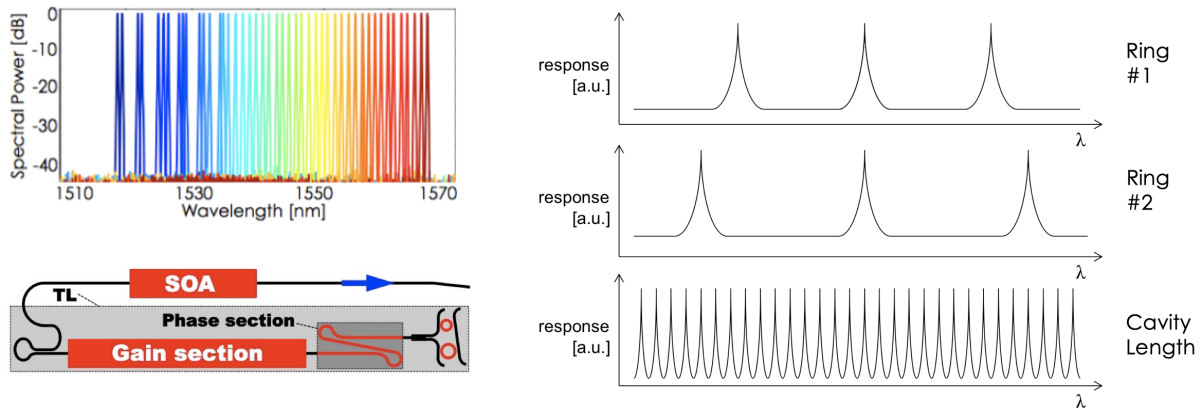


Figure 2.4: Overlapped optical spectra (top left) demonstrating the capabilities of the tunable laser (bottom left). The plots on the right abstractly describe the various filters which dictate the laser's output wavelength.

(Si) waveguide [26]. Current can also be injected in heaters implemented on the phase section. This allows modification of the refractive index and tuning of the laser emission wavelength. The two microring resonators are designed with a high quality factor and slightly different radii [27, 28]. This so-called Vernier effect provides a narrow-band optical filter, which allows for a large side-mode suppression ratio in the laser emission spectrum. The center of this filter can be thermally tuned by changing the current in heaters implemented on top of each ring. With careful adjustment of the current injected through the phase section and the ring heaters, the emission wavelength of this PIC can be tuned over 50 nm, centered around 1540 nm. The major caveat to heater control is

the long time constant associated with thermal settling, an unsuitable trait for the wide loop bandwidth and fast settling times desired in this system. Adjusting the current in the lasers gain section provides another tuning knob where the underlying physics allow \sim GHz modulation bandwidth. This is therefore used as the high bandwidth control point, with the ring and phase heaters providing coarse tuning. The digital controller facilitates control of the heaters as a DC operating point for which the synthesizer and gain section vary around.

2.4 Communication Hierarchy

The complexity of the loop is hidden from the end-user by seamless integration with a software interface. However, the large number of disparate components with vendor specific interfacing requirements makes this quite challenging. Shown in Figure 2.5, a central processing unit (an ARM Cortex-A9) is introduced to tackle this issue of hardware/software interfacing. Several layers of abstraction are crossed in designing a synthe-

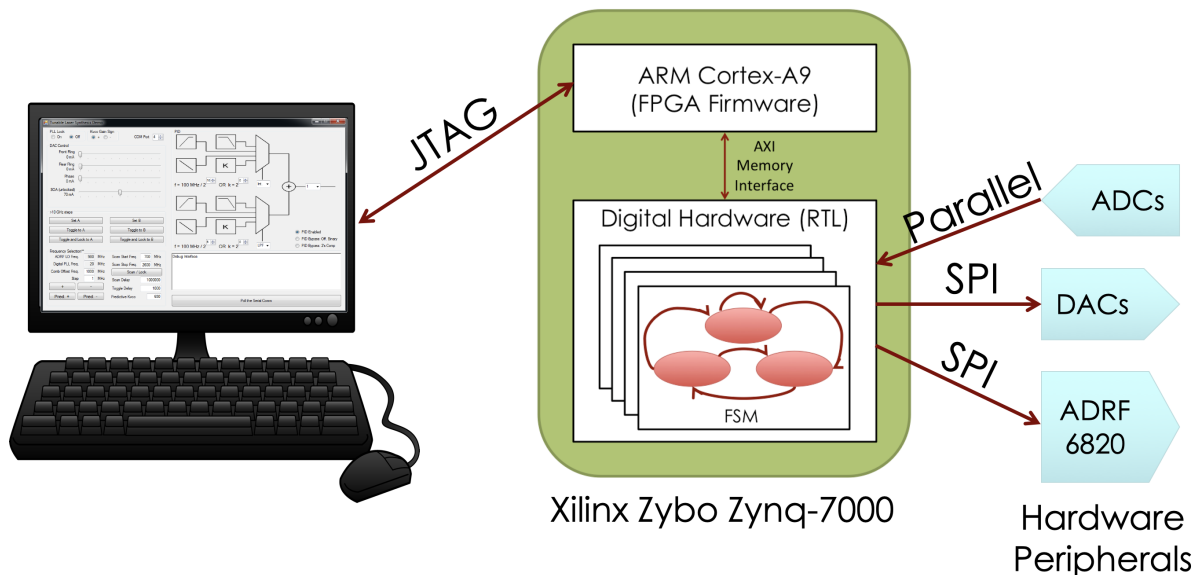


Figure 2.5: The communication hierarchy designed to interface from the custom GUI down to integrated circuits.

sizer from a Graphical-User-Interface (GUI) down to the IC level. Each of these layers required a custom protocol to communicate up and down the hierarchy.

The top-level GUI is a standalone executable I created in Microsoft Visual Basic. Commands from the user, such as *Turn Lock On* or *Change Wavelength*, are translated in the software and sent over USB to the FPGA's JTAG port. The FPGA features a hardwired ARM Cortex-A9 processor that runs a custom firmware.

When a GUI command is received, the firmware deciphers it and sends lower level requests via a Xilinx AXI Memory Interface [29]. This built-in protocol allows the processor to write to allocated memory blocks that custom HDL can read from, and vice-versa. The register transfer level logic, written in Verilog, is composed of several finite state machines (FSMs) where each handles processor requests for a major component in the system. The FSMs controlling the evaluation modules (DACs, ADCs, and the

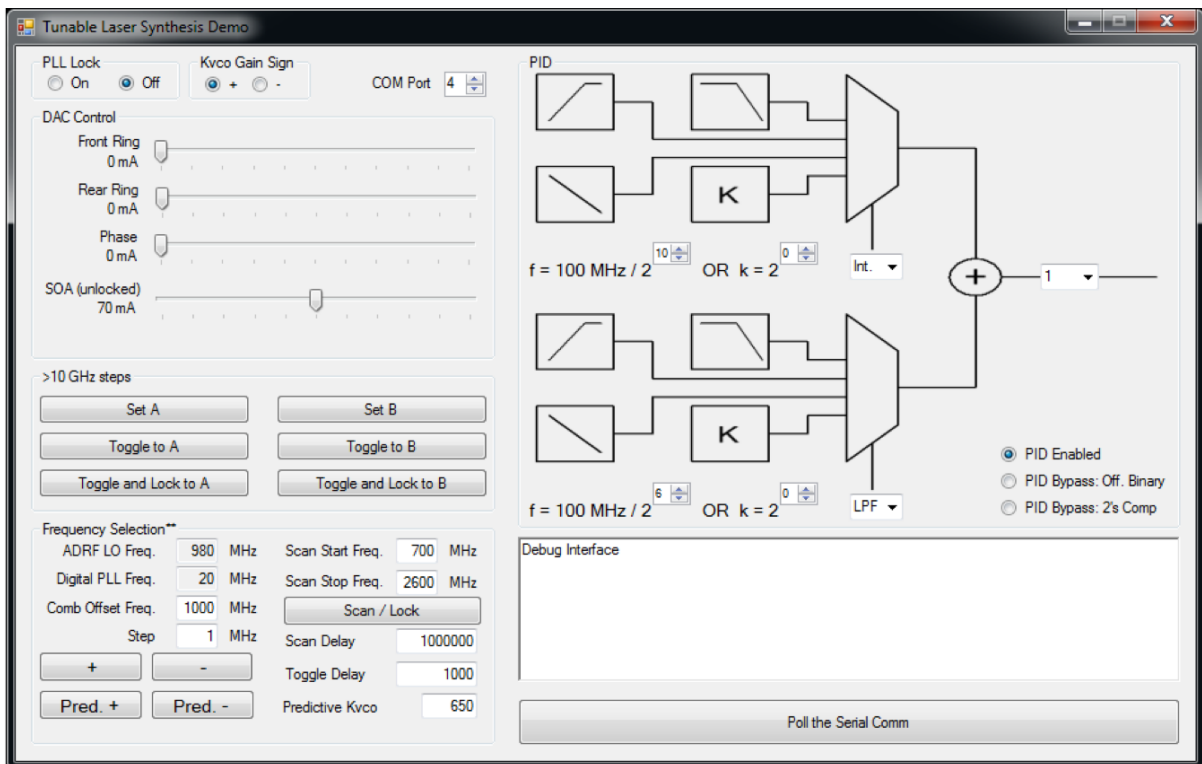


Figure 2.6: The optical frequency synthesizer's custom graphical user interface.

programmable down-converter) utilize a Serial Peripheral Interface or Parallel Bus to communicate between the FPGA and IC. Further explanation and critical code snippets are presented in Appendix A.

2.5 Loop Bandwidth

The dynamics of the feedback loop play an important role in minimizing the frequency instability of the locked tunable laser. Consider the phase domain representation in Figure 2.7.

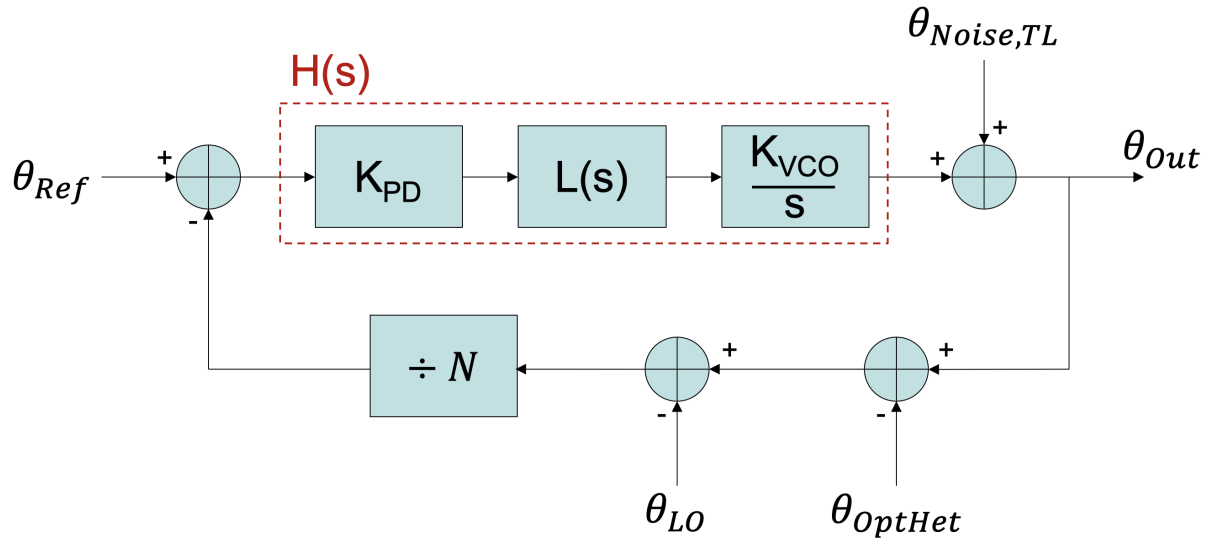


Figure 2.7: Phase domain representation of the optical frequency synthesizer.

$$\theta_{Out} = \left(\theta_{Ref} - \frac{\theta_{Out} - \theta_{OptHet} - \theta_{LO}}{N} \right) H(s) + \theta_{Noise,TL}$$

$$\theta_{Out} + \theta_{Out} \frac{H(s)}{N} = N\theta_{Ref} \frac{H(s)}{N} + \theta_{OptHet} \frac{H(s)}{N} + \theta_{LO} \frac{H(s)}{N} + \theta_{Noise,TL}$$

$$\theta_{Out} = \frac{T(s)}{1+T(s)} N\theta_{Ref} + \frac{T(s)}{1+T(s)} \theta_{OptHet} + \frac{T(s)}{1+T(s)} \theta_{LO} + \frac{1}{1+T(s)} \theta_{Noise,TL} \quad (2.3)$$

Several conclusions can be drawn from the derivation in Eq. 2.3. The intrinsic phase noise from the tunable laser ($\theta_{Noise,TL}$) will be reduced by the loop gain, $T(s) \equiv \frac{H(s)}{N}$. Maximizing the gain and bandwidth of this term will have the greatest effect on stability. Ultimately, the bandwidth is pushed as far as possible with a proportional-integral (PI) loop filter. The bandwidth limitation comes from numerous latencies (fixed delays) in the digital processing of our feedback loop. Roughly 100 clock cycles at 100 MHz are required to detect the instantaneous phase error and compute the feedback signal for the tunable laser (see Section 3.1). In order to remain stable, this one-microsecond delay should be slower than the response of the PLL, typically by one order of magnitude. This dictated a target bandwidth of ~ 100 kHz, and this is currently held constant under all operating conditions. The integral term of the filter in conjunction with the integral response of the TL (responds in frequency which integrates in the phase domain) led to a 40 dB / decade slope for close-in gain of the loop, see [30] for more detail. In addition to maximizing $T(s)$, the following sections will detail several techniques which helped lower the intrinsic noise.

Within the loop bandwidth, $T(s) \gg 1$ and therefore the PLL reference (upconverted by the divide ratio), the optical heterodyne, and the down-conversion LO all transfer their phase directly to the output. The adage “you're only as good as your reference” applies to all three of these signals, and this makes sense as the phase comparator will be unable to distinguish the source of any noise when those signals are summed together. For that reason, this architecture demands a low-noise local oscillator and a stable optical comb in addition to a precision reference.

This diagram assumes a perfect phase detector although that is often not the case. The board-level implementation described below is limited by finite ADC resolution and input voltage noise, which affect the conversion to instantaneous phase. Similarly, the ASIC implementation presented in Chapter 4 features a time-to-digital converter which

experiences cycle-to-cycle jitter and directly impacts the instantaneous phase measurement. In both cases the phase noise would enter the feedback loop at the same point as ϕ_{Ref} , and therefore its phase error is filtered by the $NT(s)/1+T(s)$ transfer function. The cycle-to-cycle noise in these cases essentially contributes all of its phase noise power at the reference frequency (100 MHz). Since this is 1000x faster than the loop bandwidth, $1 \gg T(s)$, the phase detector noise will be averaged by $NT(s)$, a low-pass filter at high frequency. Assuming a normal distribution for this error, the deviation from the expected mean decreases by the square root of the number of samples in the average (1000). Proof for “distribution of a sample mean” in [31]. In simpler words, the lower the loop bandwidth, the less this noise source affects the output. Unfortunately this trend is contradictory to all of the other intrinsic phase noise sources, which are minimized through a high bandwidth loop. Therefore, the loop filter provides a factor of $\sqrt{1000}$ cycle-to-cycle noise reduction, but we must rely on other design practices to create the best possible phase detector.

Chapter 3

Board-Level Implementation

The work presented in this chapter was previously published in *Optics Express* (OSA) [32]. Frequency synthesis is achieved at the board-level with the architecture shown in Figure 3.1. The two reference lasers imitate comb teeth spaced several modes apart for the preliminary stability and switching measurements. Light from the tunable laser output is coupled and the resulting beatnote is detected on a benchtop photodiode and transimpedance amplifier, the Agilent 11982A. Early synthesis experiments showed that the unlocked tunable laser periodically exhibits a frequency jitter larger than the acquisition range and faster than the loop bandwidth. To prevent the synthesizer from losing lock when this occurs, two high-speed frequency dividers are employed, reducing the speed and magnitude of beatnote's frequency jitter. Low noise amplifiers are also inserted to amplify the input and output signals to suit the full-scale range of each commercial component.

The digitally assisted analog front-end utilizes down conversion to limit the noise bandwidth and improve the beatnotes SNR. The ADRF6820 contains a local oscillator which can be tuned 600 - 2700 MHz, allowing synthesis of an offset lock over this entire range [33]. This chip outputs in-phase and quadrature (I/Q) signals with a nominal

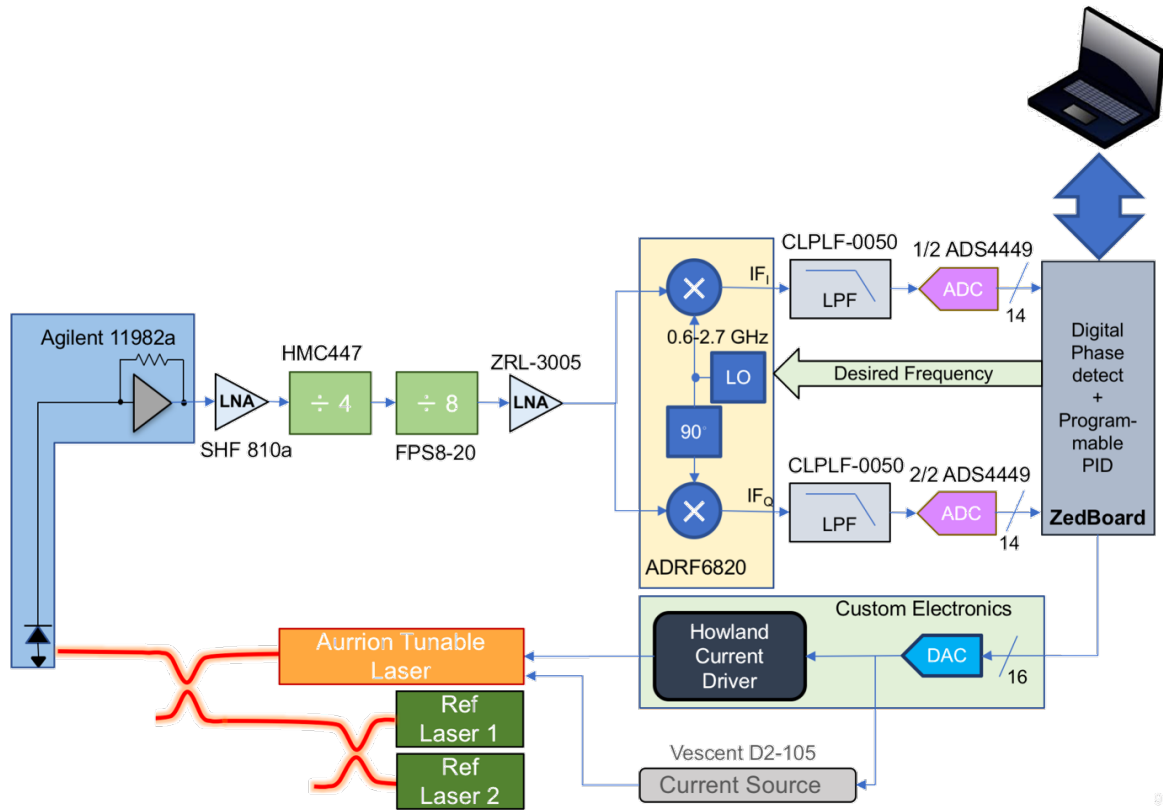


Figure 3.1: Board-level implementation of the optical frequency synthesizer.

intermediate frequency (IF) of 25 MHz. Commercial 5th order low-pass filters provide anti-aliasing before the ADS4449 samples I and Q simultaneously. Moving to the digital domain facilitates the complex control of the TLs multiple tuning knobs. This is performed on a Xilinx Zybo FPGA/ARM SoC. The FPGA implements a digital signal processor for phase error detection and correction, while the ARM processor manages user requests. The error correction —implemented as a programmable PID loop filter— outputs a feedback signal which is summed with the laser DC operating point bias currents.

A custom PCB implements all of the tunable lasers current drivers through 16-bit voltage DACs and voltage-to-current converter circuits. Stability of the PLL is achieved through proper selection of coefficients in the programmable loop filter. As described in

Section 2.5, the filter is currently configured for phase-lead compensation to provide the highest loop bandwidth with ample stability ($> 60^\circ$ phase margin). While this particular tunable laser has a high frequency conversion gain ($K_{VCO} = 300$ MHz/mA), the loop filter provides the ability to shift the overall gain of the loop up or down to accommodate a variety of lasers.

3.1 Digital Phase Detection and Error Correction

This section discusses the challenges and design methodologies for the digital signal processing. Note that Appendix A contains code snippets for the corresponding Verilog modules for those who wish to implement a similar system or gain a deeper understanding. My colleague, Akshar Jain, wrote the CORDIC algorithm implementation described below and I was the main designer for the other modules.

A key element to any frequency synthesizer is the component which computes the instantaneous phase error from a reference oscillator. While most classical PLLs utilize analog phase detectors or phase / frequency detectors, this architecture requires a digital calculation, in order to utilize the digital domain to facilitate the complex control of the TL. The in-phase and quadrature signals are sampled versions of $\sin(\omega_{IF}t)$ and $\cos(\omega_{IF}t)$. Dividing the I by the Q yields the tangential signal rotating in time. Therefore an arc tangent calculation can be performed to give the instantaneous phase (digitally sampled). The conventional division operation and arc tangent calculation are extensive in terms of the required footprint and switching power. The CORDIC algorithm described below takes in the same I and Q signals and yields the phase output in a successive approximation manner which does not require any multipliers or dividers[34].

The algorithm first looks at the signs of I and Q to determine which quadrant the signal is in. Then within a given 90 degree window, the CORDIC determines if the

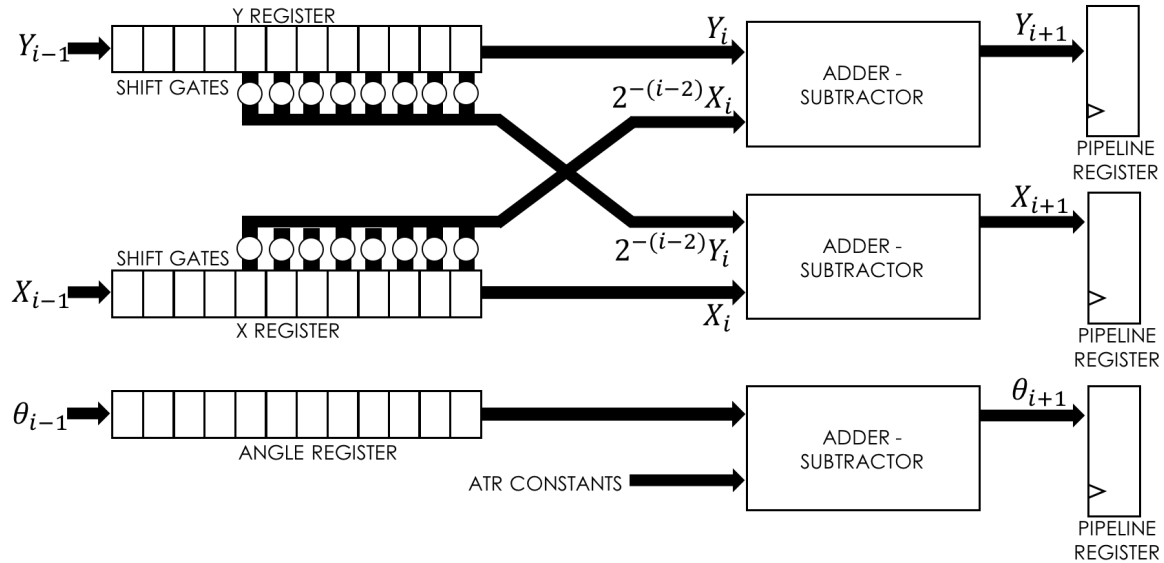


Figure 3.2: Pipelined CORDIC Arc Tangent Algorithm

signal is greater than or less than 45 degrees by looking at the magnitude of the X and Y signals. I and Q will be manipulated to take that result into account and then determine if the angle is greater than 22.5 degrees in that 45 degree window, and so on and so forth. All of the calculations are performed with shift registers and adders, and the phase output is a sum of precomputed angles stored in a lookup table. Furthermore, this iterative algorithm can be easily implemented in a pipelined fashion, improving the digital throughput and easing timing closure in the FPGA. A diagram of this CORDIC is shown in Figure 3.2, which achieves 32 bits of phase accuracy in a 30 stage pipeline.

The instantaneous phase of the IF signal is then compared to an ideal phase, which is generated with respect to the reference clock. Imagine a scenario where the reference clock is at 10 MHz and the IF is 11 MHz. One would expect the instantaneous phase to increase by 36 degrees every cycle, and wrap around every 10th cycle. Therefore, the ideal phase is constructed as a 32-bit accumulator of an expected rate of change - relative to f_{Ref} . Shown in Figure 3.3, this is then compared to the instantaneous phase computed by the arc tangent block to generate the overall phase error.

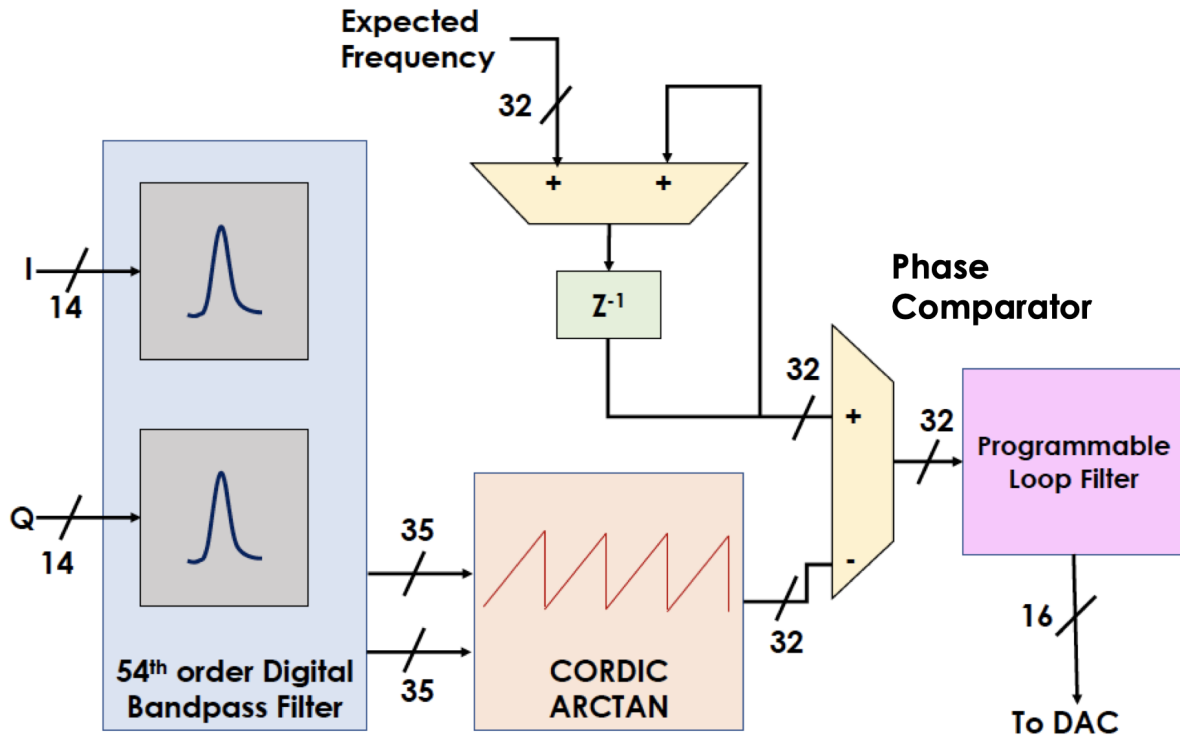


Figure 3.3: Digital phase detection and error correction. Signal path takes I/Q signals through a bandpass filter, CORDIC arc tangent, digital phase comparator, and then a programmable loop filter.

The output from the phase comparator is fed into the programmable loop filter which generates the control signal for the gain section of the laser. The two infinite impulse response (IIR) filters in the figure can be configured to be integrators, low-pass filters, differentiators, or proportional gain blocks. The parameters for these blocks are dynamically updated by the user via the GUI, which allows them to adapt to the laser's bandwidth requirements in real-time. An integrator and low-pass filter are most commonly chosen for the conventional phase-lead compensation. Section 2.5 detailed the selection of 100 kHz loop bandwidth, the maximum achievable while maintaining stability and minimizing phase degradation from digital pipeline delays. The integrator is set to a unity gain frequency slightly below the target loop bandwidth, so that the addition of the low-pass filter (with gain of 1) provides a zero in the response, and a higher phase

margin at crossover. The low-pass cutoff is set ~ 1 MHz for additional filtering of high frequency noise. Chapter 10 of [30] further explains this design methodology.

Similar to the design considerations with the instantaneous phase error calculation, conventional IIR filters require costly multiplication blocks. By constraining all of the IIR filter coefficients and gain parameters to powers of 2, this implementation can carry out the computations using simultaneous shift and add operations and does not use any multiplication or division operations. The resulting feedback loop has comparable stability and significantly lower area and power consumption.

3.2 Low-Noise Tunable Laser Drivers

The tunable laser requires several tuning knobs to be driven with quiet, high-resolution current sources. The ring resonator heaters have a 20 mA full-scale range while the phase section needs up to 10 mA and the lasers gain section requires high resolution but only several milliamps of tunability. The Improved Howland Current Pump drawn in Figure 3.4 is an ideal driver for all of these tuning knobs[35]. The feedback loop forces the current through the load to be V_{DAC} / R_5 . Most low-noise commercial off-the-shelf DACs are unbuffered voltage outputs so this is a practical conversion to the current domain. Furthermore, for a fixed voltage range DAC this gives the ability to change the current full-scale range by adjusting R_5 . This allows utilization of one 8-channel DAC for the numerous tuning knobs described above without sacrificing resolution or introducing the potential to violate compliance limits.

The PCB is shown in Figure 3.5, it contains identical copies of the Improved Howland Current Pump tied to various channels of the octal DAC, each with precisely selected R_5 values. The voltage DACs are programmable via a 3-wire SPI interface. As described in Section 2.2, the laser gain section is the high-speed plant for the feedback loop. In

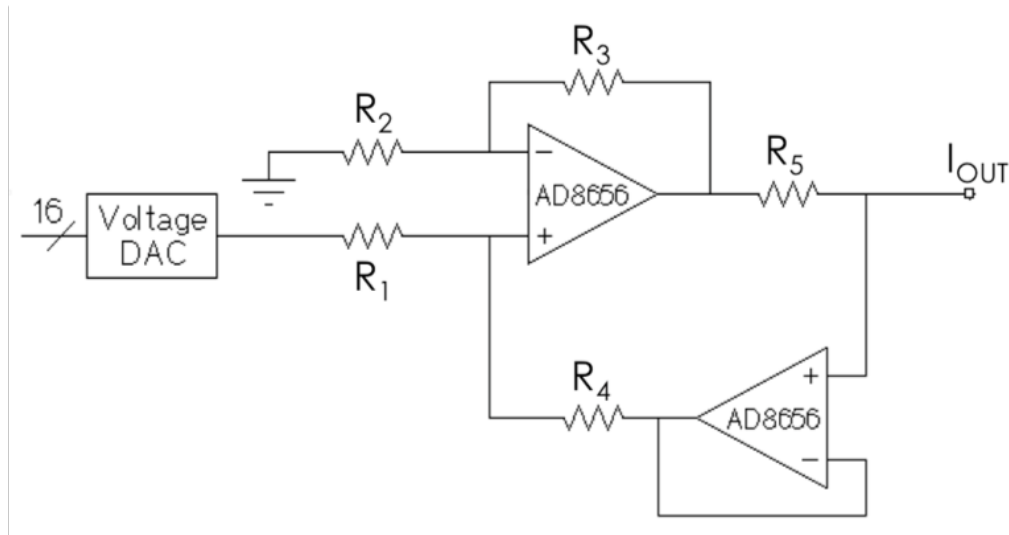


Figure 3.4: Improved Howland current pump schematic

order to minimize the loop latency, this signal comes from a separate high-speed DAC with parallel LVCMOS inputs. A full-scale range of 2 mA is chosen for this output — corresponding to 30 μA LSB steps from the DAC. The DAC current is summed with a separate DC bias current pump of 65 mA which sets the DC operating point of the laser’s gain section. The AD8656 op-amp was carefully chosen to meet the demands of our system (high-bandwidth, low-noise, >20 mA output current). The current driver exhibits 15.2 μA_{RMS} noise, limited by the output noise voltage DAC.

This circuit card was designed to directly plug into four fixed headers on the FPGA board. This kept the high-speed traces at minimum lengths which in-turn minimized reflections and inter-symbol interference, and attained the maximum specified update rates. The digital and analog domains, shown in red and blue respectively, were intentionally isolated to prevent digital noise from coupling to the outputs. These techniques along with proper supply regulation and ground loop minimization helped achieve performance on par with state-of-the-art commercial laser drivers, which are commonly server rack sized units with bulky battery supplies.

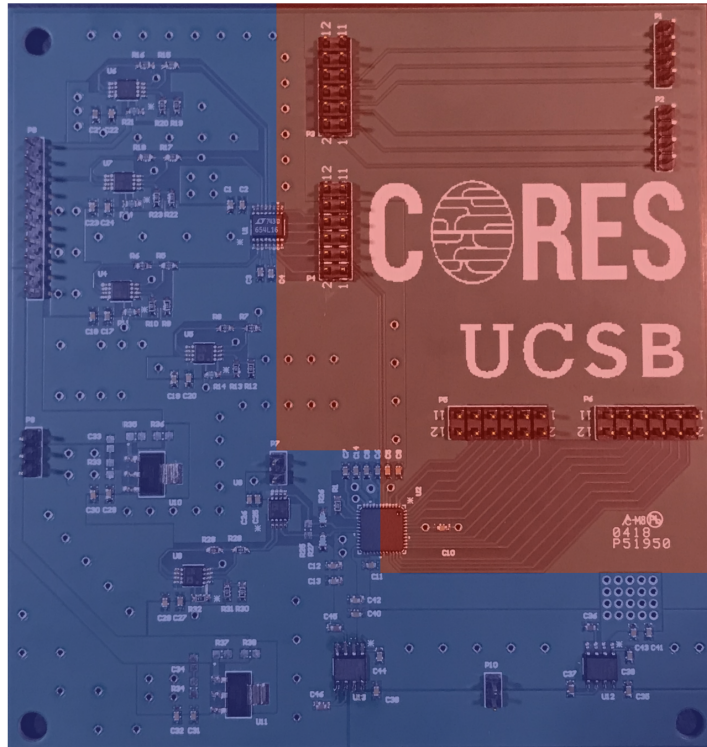


Figure 3.5: Printed circuit board featuring low-noise tunable laser drivers.

3.3 Unlocked Frequency Jitter

Figure 3.6 shows the TL as it was attached to a breakout board in the preliminary experimental phase. It quickly became evident that the tunable lasers wavelength, and consequently the IF beatnote, were prone to substantial frequency jitter when the synthesizer was unlocked. The top right subplot shows how fast and far the TL could move in a 10 second span. The heterodyne-based front end sets a limit on a subset of frequencies where the PLL will properly lock in a stable feedback loop. For a 25 MHz nominal IF with a 100 MSPS ADC, jitter greater than 25 MHz towards the LO (DC) will show the image frequency and go unstable in positive feedback, and moving greater than 25 MHz in the other direction will go above the Nyquist rate and produce erroneous phase detection due to aliasing. The frequency synthesizer could therefore only be turned on and

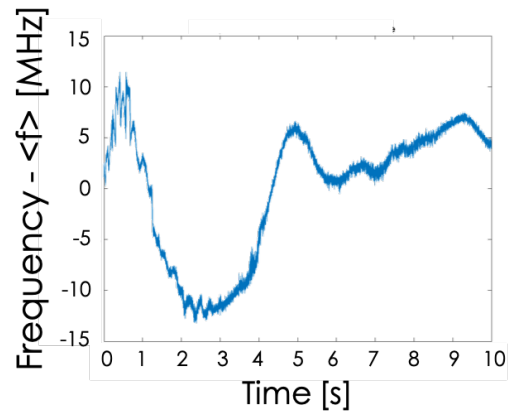
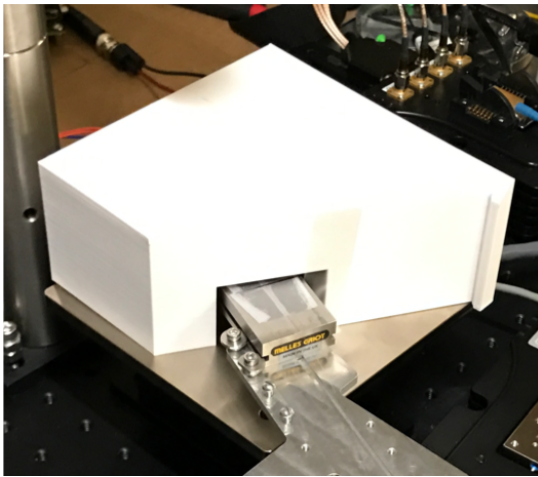
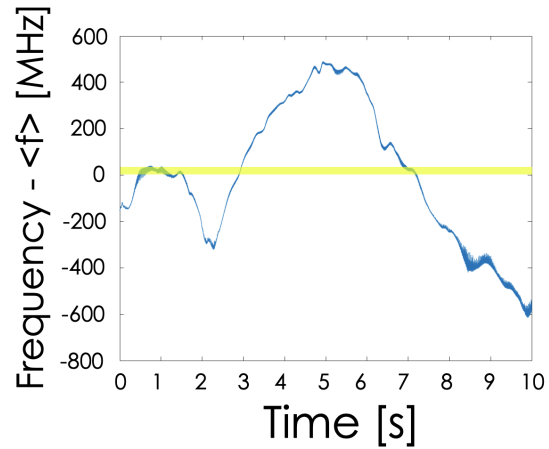
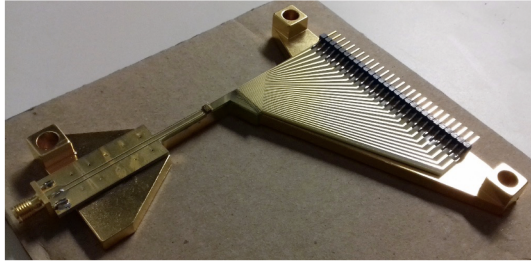


Figure 3.6: The tunable laser on its breakout PCB and copper heatsink (top left) exhibiting unlocked frequency jitter (top right). A 3D printed enclosure (bottom left) stabilizes the temperature and improves the intrinsic frequency stability (bottom right).

locked in a ± 25 MHz window, the yellow highlighted region of Figure 3.6 (top right).

After weeks of debugging, this frequency instability was eventually attributed to thermal fluctuations in the laboratory. Even with the TL on its own temperature feedback loop with a thermoelectric cooler (TEC), the slightest breeze in the room — from someone walking past the tabletop setup or opening a nearby door — could cause the jitter reported above. The enclosure shown in 3.6 (bottom right) was 3D printed to alleviate this, and yielded the drastic improvement shown in the bottom left subplot. The diag-

nosis and the resolution in this case were pivotal in moving this work from a theoretical architecture to a proven optical frequency synthesizer. The thermal stability is of strong importance for the ongoing task of packaging the cointegrated electronics and photonics.

3.4 Scan-And-Lock Algorithm

One key challenge is to find the beatnote before initial lock is acquired or after switching to a far-off wavelength. Predefined set points for the heaters allow reliably movement of the tunable laser to the correct comb tooth, though the absolute offset frequency can vary several hundred megahertz due to environmental variables — even with an isolated thermal chamber like presented above. Given a 100 Msps ADC and 25 MHz nominal intermediate frequency, we previously defined a ± 25 MHz acquisition range for the front-end. A custom algorithm utilizes the widely tunable local oscillator to overcome this acquisition range limitation. Whenever a user makes a request to enable the PLL or jump to a new comb tooth, the LO performs a sweep across its full range. Figure 3.7 animates this functionality. In the top row, the LO is far from the RF input and the outputs do not fall within the ADC bandwidth. Eventually the LO is brought close enough to the input (b), and the down-converted signal is detected on our ADC. The RMS voltage of the signal is recorded on the FPGA throughout the sweep, and following sweep completion the LO is set such that the beatnote is centered within the ± 25 MHz window. The algorithm then acquires lock at that frequency and subsequently tracks thermal drift or synthesizes any desired offset frequency.

The preliminary version of this algorithm swept linearly in 5 MHz steps, ensuring an optimal starting point was chosen within ± 2.5 MHz. Analysis showed that this algorithm was the largest factor in the time required for frequency switching (wavelength hopping). This is a fairly significant metric in numerous applications where information

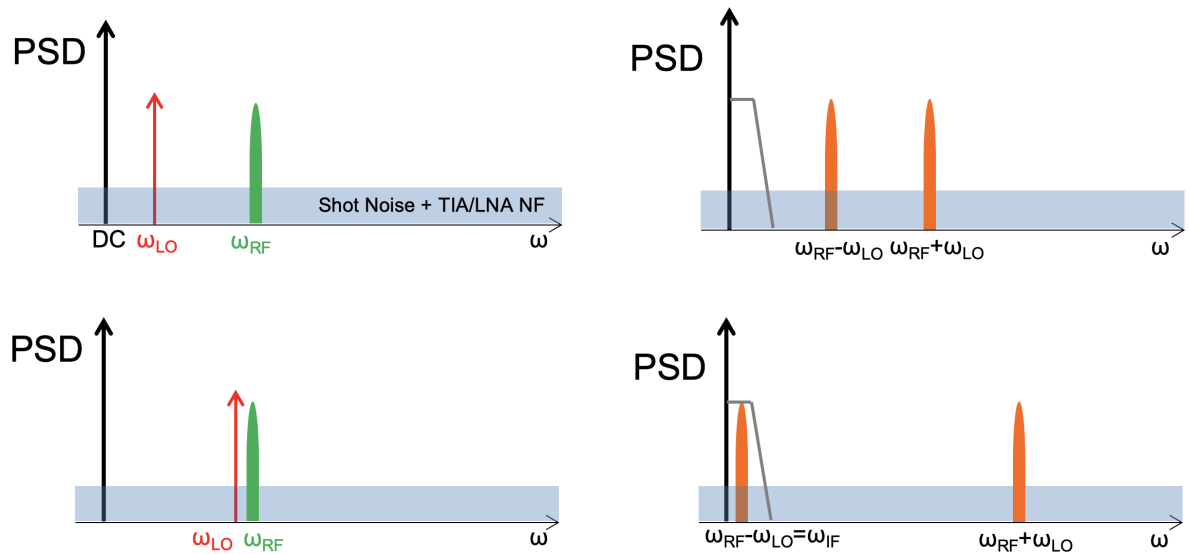


Figure 3.7: Heterodyne receiver’s response – inputs left, outputs right – when the LO is far from the input (top), and when the LO approaches the input (bottom) and the down-converted output falls within the IF bandwidth.

is gathered by sweeping the laser over a wide range, such as gas sensing or LiDAR. Performing a coarse-fine sweep reduces the scan time, requiring less LO steps to find the signal. Plotted in Figure 3.8, 50 MHz steps are initially taken and then the LO finely scans back over the predicted region of interest.

By default, the ADRF6820 performs a calibration of its VCOs internal settings each time the LO is stepped, requiring 2 ms per step. I reached out to the Analog Devices designers of this chip to develop a method to override the calibration and provide our own VCO internal settings. The slower sweep is done once at startup, and the ADRF6820s calibration results are stored in a lookup table (LUT) in the FPGA. Because this system operates in a fixed environment, these results should be valid for the remainder of operation. For any subsequent request to move the LO’s frequency, the FPGA explicitly cancels the re-calibration and sends the stored calibration values. Figure 3.8(b) shows how this made the sweep 256 times faster.

3.5 Notable Results

The measurement equipment used in the system is present in Fig. 3.1. A Yokogawa AQ6730C Optical Spectrum Analyzer (OSA) records the optical spectra with 20 pm resolution however RF analysis is necessary to report synthesizer performance. The divided beat note is down converted to a frequency that can be tracked by a Keysight 53230A Frequency Counter. This is used to measure Allan Deviation (ADEV) and switching times for larger frequency steps. The counter data is filtered through the frequency dividers in the signal path, so all plots are generated by multiplying the raw data by 32 to give an accurate representation of the stability of the beatnote and optical signal. The Keysight 53230A is known to perform internal filtering that makes it unsuitable for measuring fast frequency steps such as <1 GHz changes offset of the the same comb tooth. Therefore

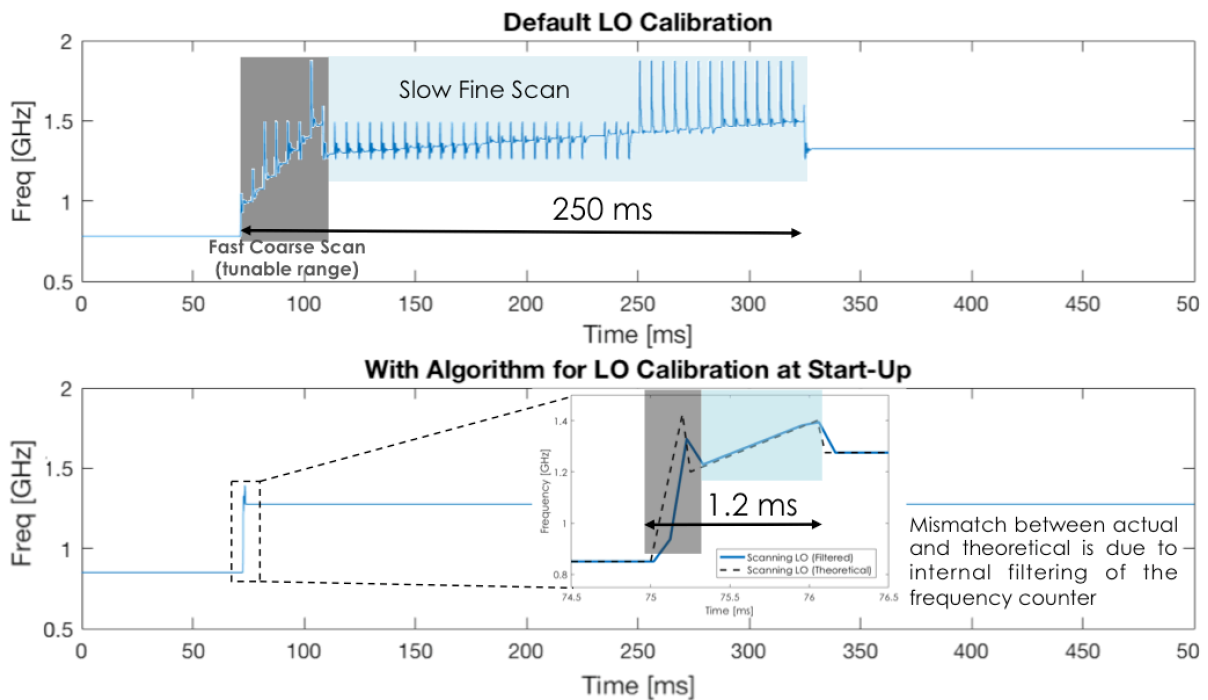


Figure 3.8: Coarse/fine scan-and-lock algorithm with the IC’s default LO calibration (top) and improved sweep after overriding the calibration step and providing pre-stored calibration constants (bottom).

the settling time of the DAC is measured on an oscilloscope.

The RF beatnote frequency is plotted as a transient in Fig. 3.10. With a user setpoint of 193,590,743,812,940.3 Hz, most of the data are contained within a noise floor of ± 75 mHz. This setpoint is held for over 5 hours without interruption. Long-term frequency stability is shown as an overlapped ADEV on the right. A 1 second gate time is used to analyze the trend out to 1E3 tau. This stability is several orders of magnitude better than the specifications set forth by DARPA to deem this technology viable. If this were to someday become a wearable timing reference, the clock would slip one second every 12.2 billion years. Furthermore, we believe this is the world record for an offset lock of any PIC based laser, and only beaten by NIST's optical lattice clocks which occupy entire laboratories and run on kilowatts of power.

The resolution of this synthesizer is demonstrated in Figure 3.11. A script was used to step the offset frequency up (and then down) by the smallest possible step every 10 seconds. Note that every 30th sample is discarded from the plot due to the frequency counters averaging function. The resolution is defined as 745 mHz from the equation

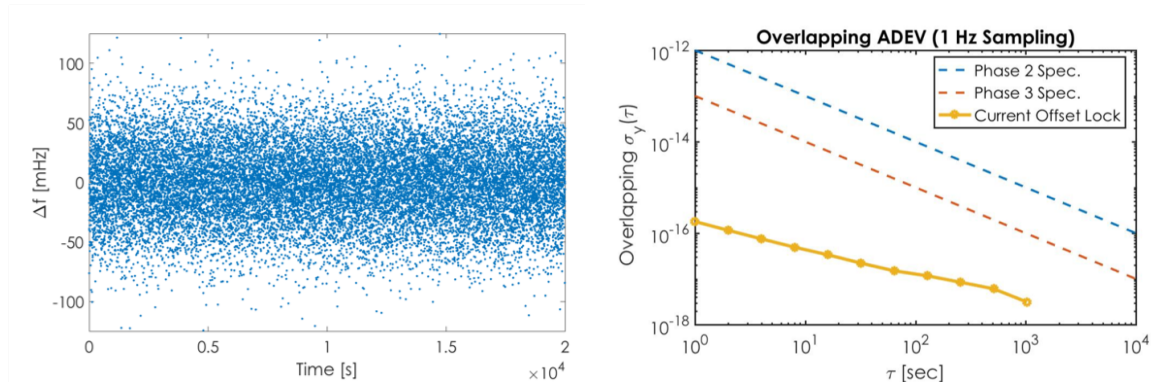


Figure 3.9: Stability of the optical frequency synthesizer plotted as a transient (left) and Allan Deviation (right).

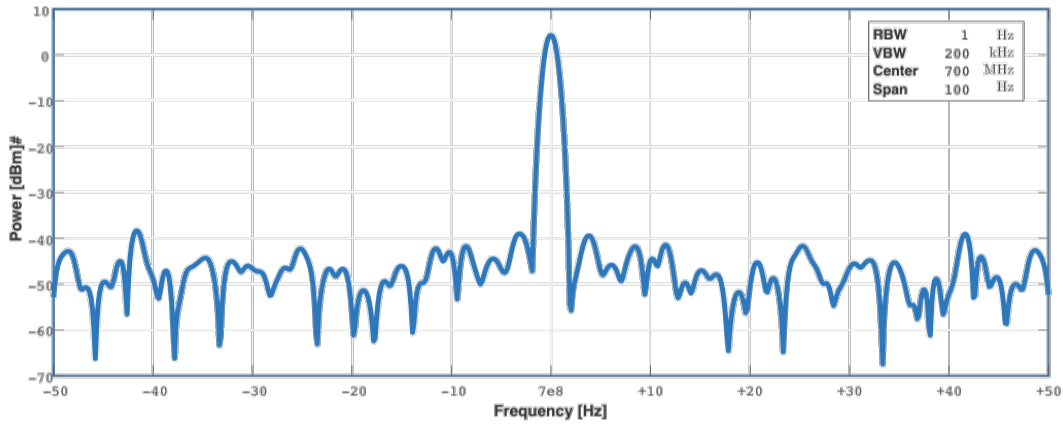


Figure 3.10: Power spectral density of the locked tunable laser plotted on an ESA. Note the resolution bandwidth of 1 Hz and span of 100 Hz.

below.

$$\begin{aligned}
 \text{Resolution} &= f_{Ref} \div (\text{Digital Bit Space}) \times (\text{Freq. Division}) \\
 &= (100 \text{ MHz}) \div (2^{32}) \times (32) \\
 &= 745 \text{ mHz}
 \end{aligned} \tag{3.1}$$

Frequency steps < 10 GHz are requested from the GUI and executed in several actions. The feedback loop is first broken and the LO is programmed to a new frequency so that the anticipated beatnote is within our IF bandwidth. The new desired phase rate of change is passed to the phase comparator and then the loop is re-closed. The switching transient that stutter-steps towards its new operating point is a characteristic response for classical PLLs pushed outside of their linear regime. However our digitally assisted architecture allows us to circumvent this. With the knowledge of how to tune of the gain section (units MHz/mA), one would expect we could calculate the expected operating point and command the DAC to jump there instantaneously. Unfortunately the operating point moves as a function of the temperature of the PIC, and so switching time is limited by the thermal settling of the chip, even for delta currents less than a milliamp. We

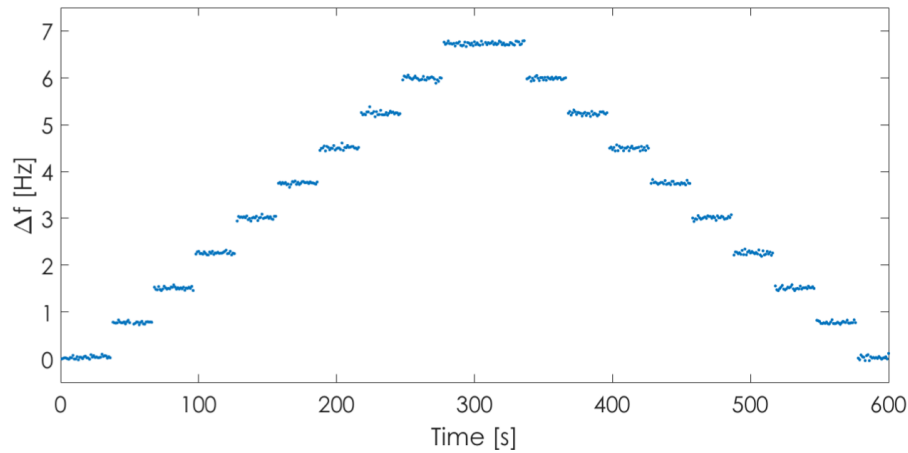


Figure 3.11: Resolution of the optical frequency synthesizer demonstrated through minimal (LSB) frequency steps.

therefore experimented with pre-emphasis drive — overheating the chip for a short period to reach the thermal steady state sooner. This is certainly an ongoing investigation that could be fully optimized in its own research project, but the early results suggest this could help bring switching times less than 50 microseconds.

Frequency switching times >100 GHz are also reported, outside the linear range of

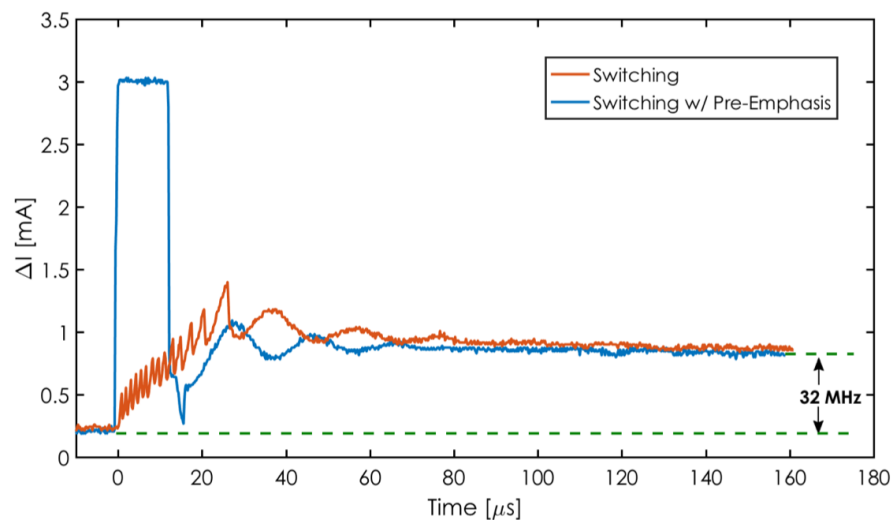


Figure 3.12: Small frequency transient steps with and without pre-emphasis.

the laser gain section tuning knob. A LUT adjusts the ring heaters to bring the tunable laser wavelength to the desired comb tooth. As is the case described above, thermal drift after changing the heaters powers can cause significant output drift ($14 \text{ GHz}/^\circ\text{C}$), so the current algorithm waits for the PIC to reach thermal stability before attempting lock. After which the scan-and-lock algorithm pulls the beatnote within the IF bandwidth and reacquires lock.

The optical spectra in Fig. 3.13 (left) show the tunable laser moving from an offset of one comb tooth to another. This is also captured in a transient measurement of the RF beatnote frequency (right). The laser moves wavelengths in $<100 \mu\text{s}$ though the thermal equilibrium is reached over 4 ms. The scan finds the new beatnote frequency and lock is achieved in $400 \mu\text{s}$. We envision the future system controlling the heaters with pre-

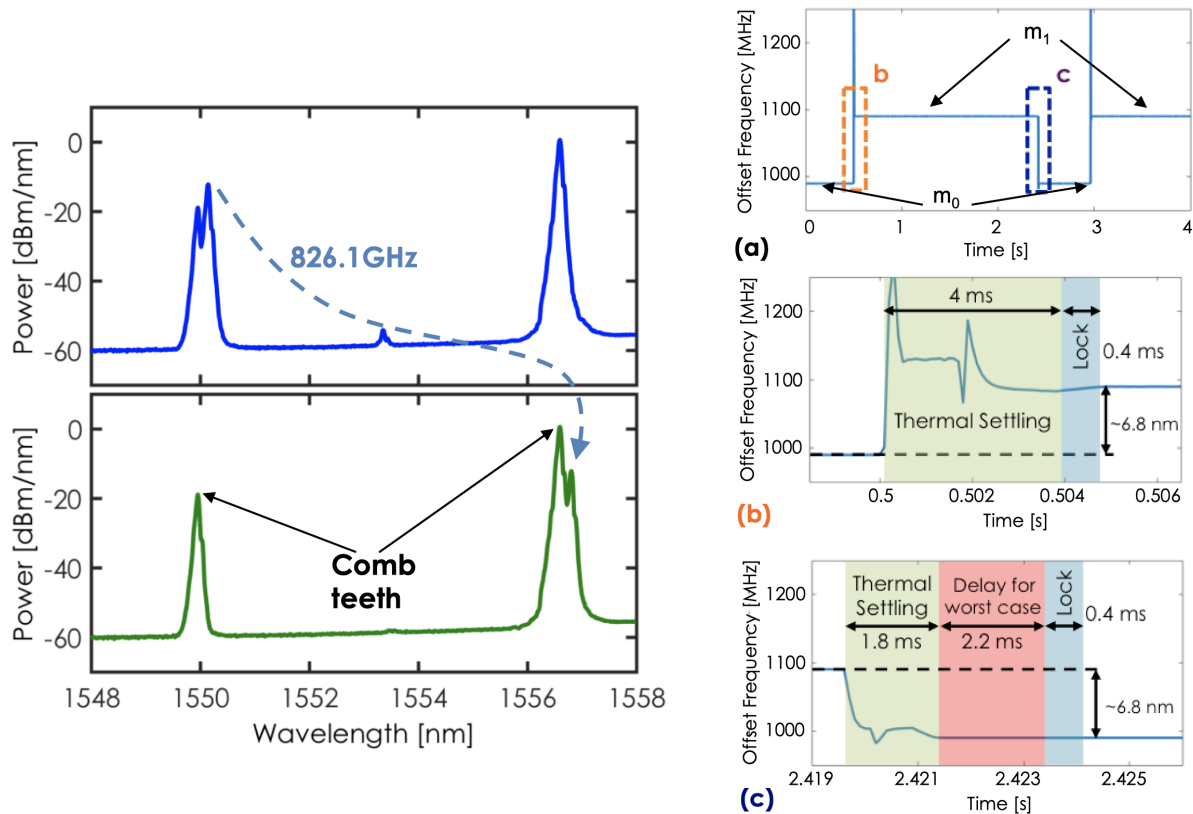


Figure 3.13: Large frequency hopping transient steps between two optical references.

emphasis drive in order to accelerate the thermal settling and eliminate the dead-time delay for the worst case scenario [36].

Finally, we sought to create a single measurement that could capture all of the capabilities of this synthesizer: stability, tunability, and versatility of the digital interface. A script was added to the GUI which could take in a .png image, convert it to a black and white image, and then map the frequency of the laser to the two-tone image in real time. The laser is commanded to hit each pixel sequentially in time, moving up a single column and then starting at the bottom of the next column. When a pixel is supposed to be "white", the laser is commanded to the frequency corresponding to the bottom row. Two example images are plotted below.

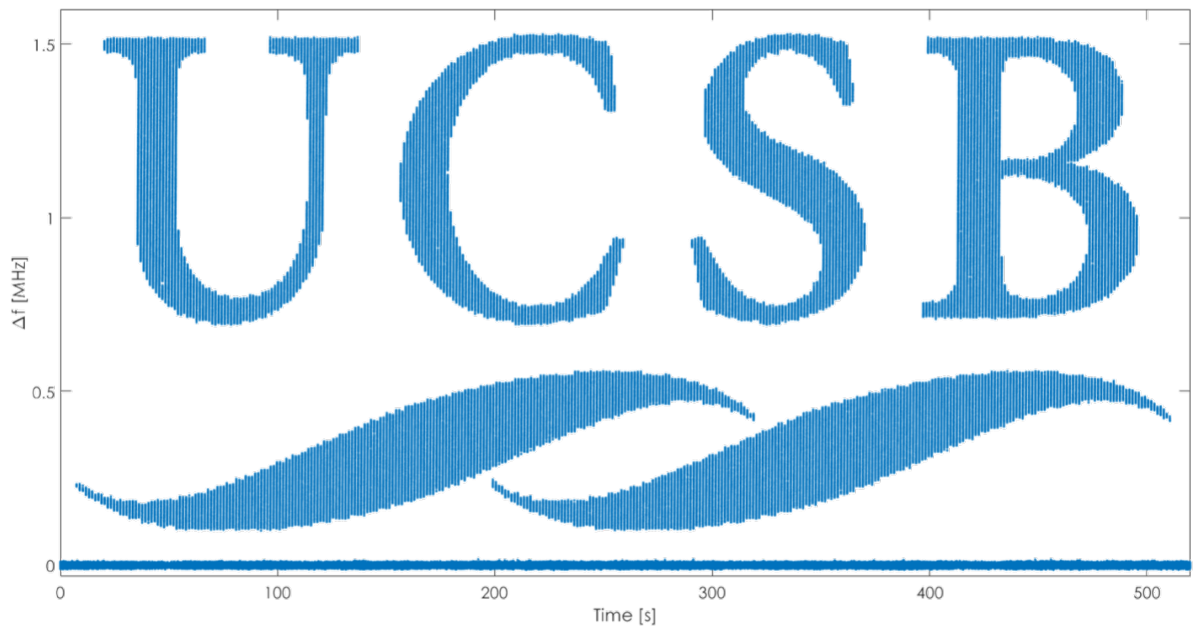


Figure 3.14: UCSB logo plotted time vs. frequency on a frequency counter via optical frequency synthesis.

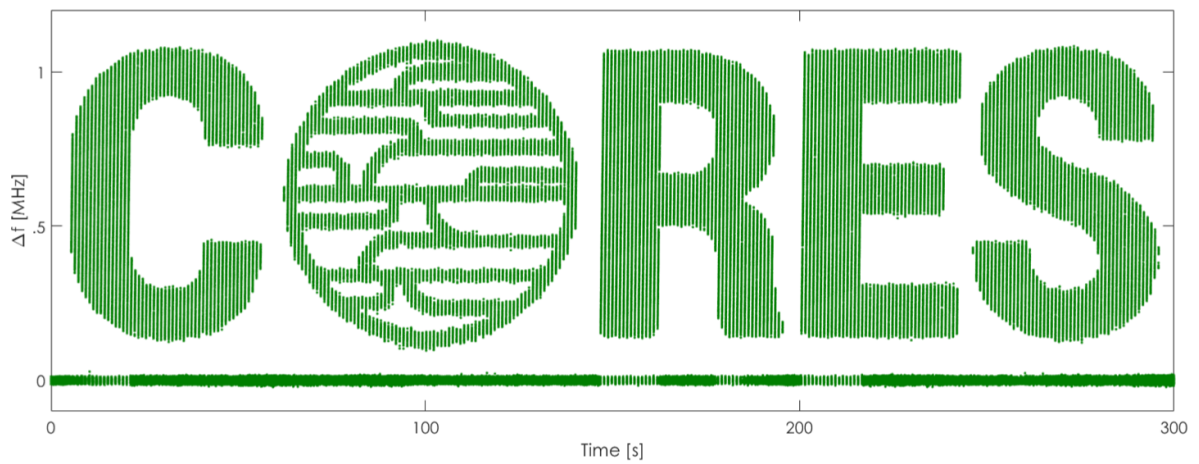


Figure 3.15: CORES team logo plotted time vs. frequency on a frequency counter via optical frequency synthesis.

3.6 Board-Level Limitations

Although it wasn't the initial plan, this implementation ended up being a multi-year project with numerous revisions and countless lessons learned. The performance greatly exceeded expectations and more importantly, I gathered where design efforts needed to be focused moving forward. Listed below are several areas in which an application specific integrated circuit (ASIC) could provide improvements:

1. The local oscillator's frequency range. With a comb tooth spacing of 15 GHz, it was previously described that the nearest beatnote could be anywhere from DC to 7.5 GHz. The heterodyne receiver therefore requires an LO that can cover that entire range. This implementation was limited to readily available COTS options, the best of which was the ADRF6820 with a 600 - 2700 MHz range.
2. The frequency divider's noise bandwidth. A common theme throughout this chapter was the effects of thermal drift on frequency instability ($14 \text{ GHz}/^\circ\text{C}$). It was concluded that frequency division was one of the mandatory remedies to track the beatnote. The dividers were inserted prior to the mixer as this was the only place in the signal path that would work with the existing implementation. However, this exposed the dividers' inputs to many gigahertz of noise bandwidth, which reduces the SNR and effectively cancels the very purpose of the heterodyne down-conversion. This would not work in the final system with a low-power optical comb. In the ASIC we were able to move the frequency division until after down-conversion; Chapter 4 will address this revision in great detail.
3. The size, weight, and power (SWaP). This implementation combined a significant amount of functionality (RF, low-noise analog, DSP) into a tabletop sized demo. However it is still several orders of magnitude away from the ultimate goal. Shrink-

ing this now-proven architecture to an ASIC footprint and cointegrating the electronics / photonics into a shared package will yield the greatest impact for existing applications and create many more.

Chapter 4

Integrated Circuit Implementation

The schematic in Figure 4.1 details the heterodyne receiver for detecting the optical beatnote. Modifications from the board-level implementation stemmed from the need for frequency division after the down-conversion to the intermediate frequency (IF). A dual-channel ADC was previously used to sample I/Q signals and digitally extract the beatnote’s instantaneous phase, however this was not feasible in the revised architecture, where the divided down IF had become a digital signal. A time-to-digital converter (TDC) was an ideal replacement. Later described in great detail, this TDC reports the instantaneous phase as the timing difference between the digital rising edges of the IF and reference. The ADC in the previous design also served a key role in finding the beatnote during the scan-and-lock algorithm. A separate signal path incorporates a peak detector and flash ADC for this purpose.

Another major component of this synthesizer — not previously mentioned — is the task of locking the 15 GHz repetition rate of the optical comb teeth, which is slightly tunable and requires its own low-noise feedback loop. That 15 GHz tone is present in the optical heterodyne which gets received on the photodetector, and therefore a path has been added in parallel to pick off this tone and detect it on its own TDC. 15 GHz is

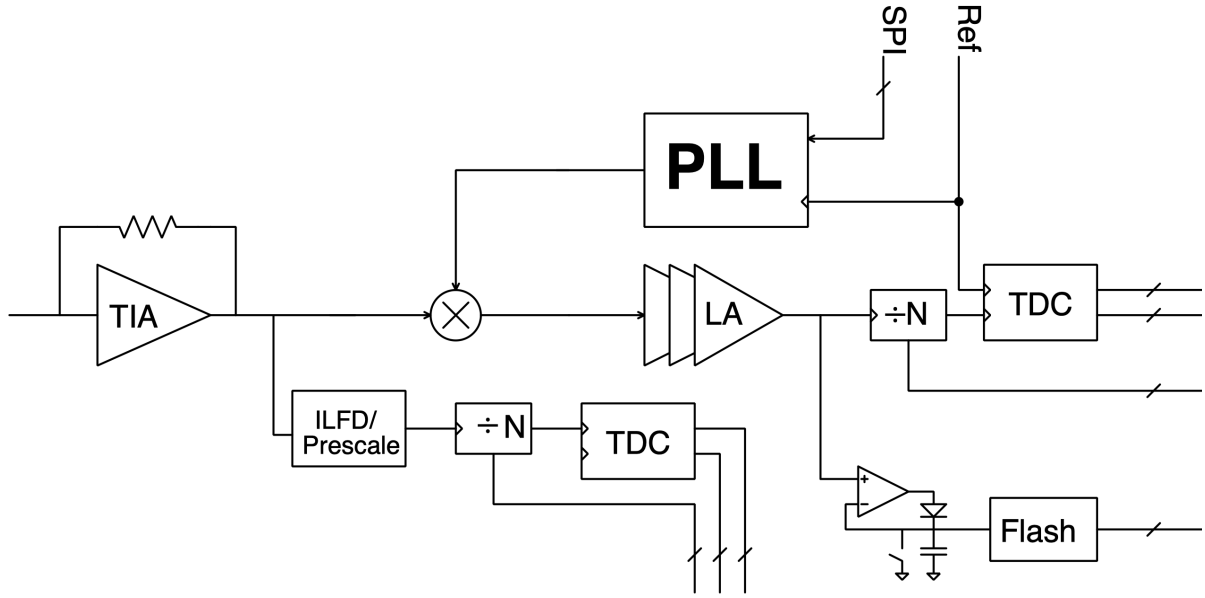


Figure 4.1: Architecture for the application-specific integrated circuit implementation of the heterodyne receiver.

near the limit of what can be processed in this 55 nm technology, and so power hungry and area intensive circuits were required. The injection-locked frequency divider (ILFD) was designed to simultaneously amplify and divide the frequency of this tone.

A common objective throughout this implementation was to design for post-fabrication flexibility. Rather than target a set of specifications for the optics, many of the components provide tuning knobs to suit future systems with different requirements (e.g. gain, bandwidth, resonant frequencies). These tuning knobs were brought off-chip to be controlled via external sources, but could feasibly be handled by on-chip DACs in a future implementation.

The broadband transimpedance amplifier (TIA) was a contribution from a colleague on this project from the University of Virginia, closely designed with the balanced photodetectors from a fellow team at the same institute. The TIA achieves $62 \text{ dB}\Omega$ gain with $>8 \text{ GHz}$ bandwidth. Rob Costanzo described the design in great detail in [37]. In addition, my advisor Prof. Theogarajan contributed extensively on the design of the

local oscillator, a widely tunable PLL. The following section outlines his architecture and highlights the challenges in designing this critical component.

4.1 Widely Tunable Phase-Locked Loop

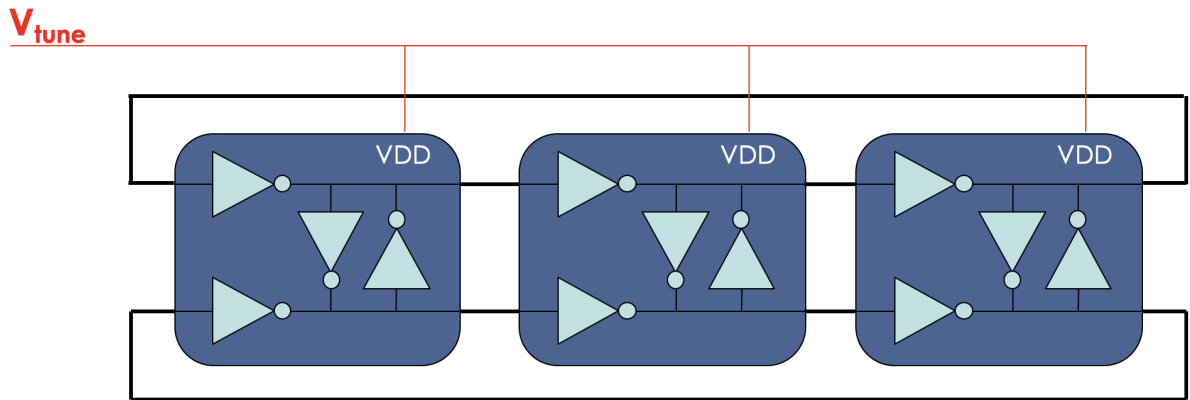


Figure 4.2: Voltage controlled 3-stage ring oscillator.

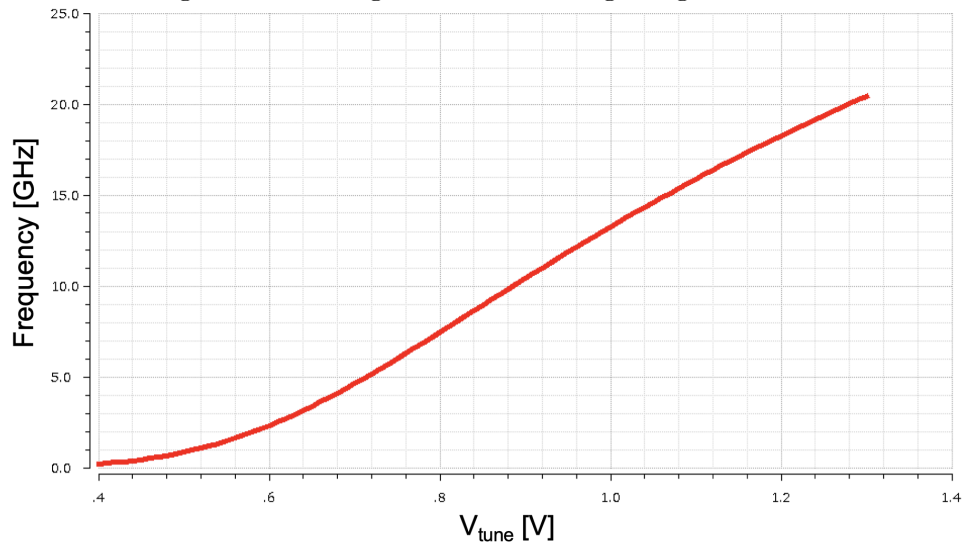


Figure 4.3: Control voltage (V_{Tune}) vs. output frequency.

The heart of this PLL is a 3-stage ring oscillator with psuedo-differential delay cells [38]. Shown in Figure 4.3, the circuit is voltage controlled with higher input voltage corresponding to a faster oscillation frequency, and a 0.1-15GHz full-scale range. Compare

this to most commercial solutions, such as the ADRF6820, which requires four separate oscillators to cover a 0.6-2.7 GHz range. Two neighboring circuits called for careful consideration in order to make this work: the voltage supply regulator and the voltage-level translator.

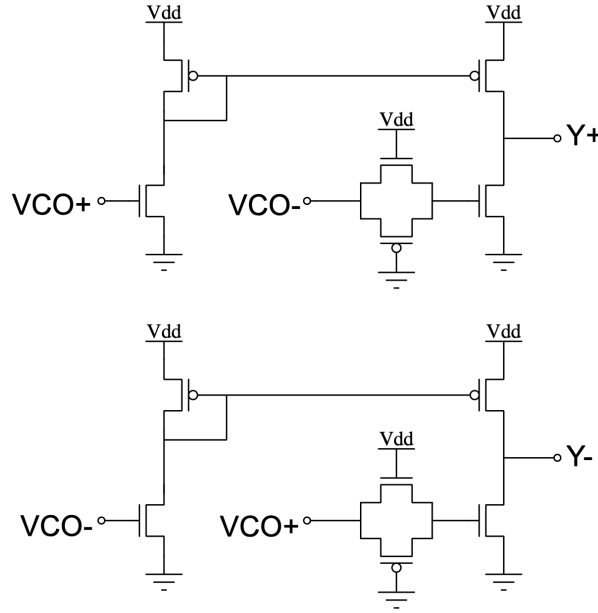


Figure 4.4: Voltage Level Translator

The supply regulator takes in the unbuffered V_{tune} voltage and creates an output capable of driving the power-hungry VCO. This also achieves $>10\text{MHz}$ bandwidth and minimal response time, in order to not become a bottleneck in the overall PLL loop bandwidth. The regulator functions from 100mV up to 1.3V , utilizing double gate FETs with 2.5V supply to operate above the chip's core voltage. The voltage-level translator in-turn must convert the oscillator's swing to the nominal core voltage for compatibility with the subsequent digital circuits. The design in Fig. 4.4 achieves up to 14GHz bandwidth while still having enough gain to create square-wave swings for slower 100mV inputs. It works by creating a push-pull drive with the differential inputs. The NMOS driver is delayed through pass-gate to match the timing of the inverting amplifier driving

the PMOS.

I designed the PLL's feedback frequency divider which had several trade-offs in play. First, and foremost, it requires a widely programmable divider in order to hit the full frequency range, and it was impossible to design a programmable divider that operates up to 14GHz across all process variation corners. For a traditional divide-by-N:

$$f_{VCO} = f_{Ref} \times N; \quad f_{Ref} = 100 \text{ MHz} \Rightarrow N = 1 - 140 \quad (4.1)$$

A pre-scale divider (M) is a high-speed, relatively straightforward design, and this subsequently clocks a fully programmable divider which runs at a slower maximum frequency. One might ask why not make the pre-scaler a large division such as 10 or 100, and follow it with an overly simplified divider such as a traditional frequency counter. As the pre-scale increases, the resolution (frequency steps) that f_{VCO} can lock to gets worse for a given f_{Ref} .

$$f_{VCO} = f_{Ref} \times N \times M \quad (4.2)$$

This could be counter-measured by designing a fractional-N PLL, typically associated with worse noise performance, or by lowering f_{Ref} to achieve a desired step (i.e, 1 MHz f_{Ref} with pre-scale M = 100 to achieve 100 MHz steps). However, the loop bandwidth of the PLL is limited by the phase-frequency detector's conversion rate – conventionally $f_{BW} \leq \frac{f_{PFD}}{10}$. A pulse swallow counter design overcomes this limitation with use of a dual-modulus pre-scaler, capable of dividing by 4 or 5 by toggling an input [39]. The slower counter clocks at divide by 5 for A cycles and then switches to 4, resetting after B cycles. The overall division is calculated below, with resolution of f_{Ref} . This allows

us to achieve 100 MHz steps with roughly 10 MHz PLL loop bandwidth.

$$\begin{aligned}
 f_{VCO} &= f_{Ref} \times (5A + 4(B - A)) \\
 &= f_{Ref} \times (4B + A)
 \end{aligned}
 \tag{4.3}$$

The dual-modulus is achieved with a Johnson ring topology [40]. Compared to a traditional counter, this increases the number of flip-flops and reduces the combinational logic in order to reach faster clocking rates. There were several possible locations for the divide-by-4 / divide-by-5 output, the one labeled was chosen for having the shortest setup time for the modulus input. A and B are implemented in a single counter with two sets of digital word comparators, each with several stages of pipeline delay in order to operate reliably >4 GHz. The 6-bit counter is segmented into two 3-bit counters to reduce the combinational logic and switching circuitry that had to run at the higher speed.

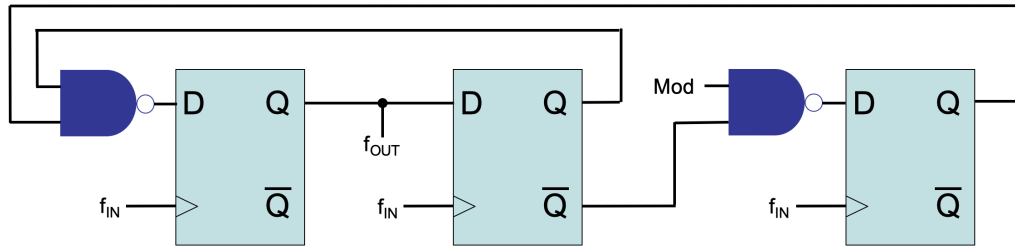


Figure 4.5: Johnson ring counter based dual-modulus frequency divider (4/5).

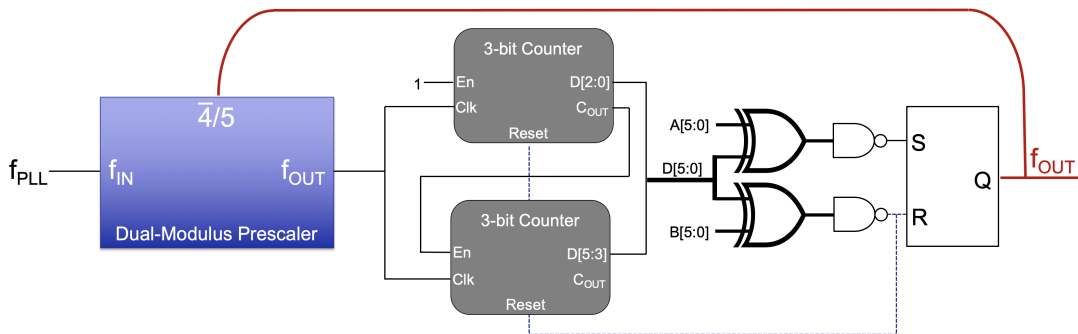


Figure 4.6: Swallow counter utilized for programmable frequency division.

4.2 Gilbert Cell Mixer

The heterodyne receiver works by multiplying the incoming and LO signals voltages, equivalent to a convolution in the frequency domain, to move the information of the beatnote to the intermediate frequency. The Gilbert cell was initially introduced as a BJT implementation that relied on logarithmic and exponential relations, however the same effect is now exploited with a similar CMOS topology [41]. The beatnote is converted to a differential current through the bottom differential OTA. The LO signals are driven rail-to-rail to act as differential switches for the current. For each side of the output this can be thought of as multiplying the input current by 1 and -1 at the rate of f_{LO} , which yields a frequency-domain convolution with the odd harmonics of f_{LO} . The first term in Eq. 4.5 is the desired intermediate frequency ($\omega_{IF} = \omega_{RF} - \omega_{LO}$) and the remaining terms can be filtered out. The circuit is designed to be double-balanced with a differential output to minimize unwanted non-linearities and switching effects (coupling capacitances, common-mode gain).

$$\begin{aligned}
 \text{Incoming Beatnote:} & \quad \cos(\omega_{RF}t) \\
 & \quad \times \\
 \text{Local Oscillator:} & \quad \begin{cases} 1 & \text{if } \text{mod}(t, \frac{1}{f_{LO}}) < \frac{0.5}{f_{LO}} \\ -1 & \text{otherwise} \end{cases} \quad (4.4) \\
 \text{(Fourier Transform)} & \quad \frac{4}{\pi} \sum_{n=1,3,5,\dots}^{\infty} \frac{1}{n} \sin(n\omega_{LO}t)
 \end{aligned}$$

$$\begin{aligned}
 & = \frac{2}{\pi} \sin((\omega_{LO} - \omega_{RF})t) + \frac{2}{\pi} \sin((\omega_{LO} + \omega_{RF})t) + \\
 & \quad \frac{2}{3\pi} \sin((3\omega_{LO} - \omega_{RF})t) + \frac{2}{3\pi} \sin((3\omega_{LO} + \omega_{RF})t) + \\
 & \quad \frac{2}{5\pi} \sin((5\omega_{LO} - \omega_{RF})t) + \frac{2}{5\pi} \sin((5\omega_{LO} + \omega_{RF})t) + \dots \quad (4.5)
 \end{aligned}$$

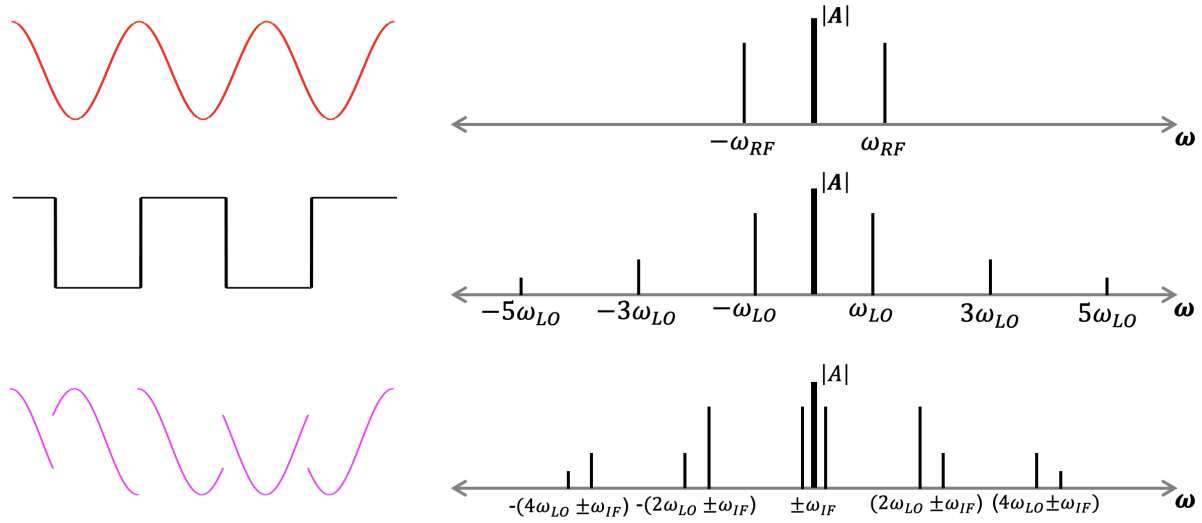


Figure 4.7: Transient signals (left) and amplitude vs. frequency (right) shown for the beatnote input, local oscillator input, and mixer output.

4.3 Baseband Amplifier

The output from the mixer is expected to swing several hundred microvolts. This circuit, jointly designed with my colleague Akshar Jain, intends to amplify the IF tone while removing all of the unwanted harmonics. One major design consideration was to make this circuit tunable post-fabrication, to ensure the ASIC will be capable of accommodating future potential applications with different IF bandwidths and signal strengths. The differential amplifier in Figure 4.8 accomplishes this. The differential gain is set by

$$g_m \times (r_{on} \parallel r_{op} \parallel R_{lin}/2)$$

where R_{lin} is shown in Figure 4.9. The amplifiers cutoff frequency is also set by this resistance and the output capacitance. Assuming $R_{lin} \ll r_{on} \parallel r_{op}$

$$\begin{aligned} Gain &\approx \frac{g_m R_{lin}}{2} \\ f_{3dB} &\approx \frac{1}{\pi R_{lin} C_{out}} \end{aligned} \tag{4.6}$$

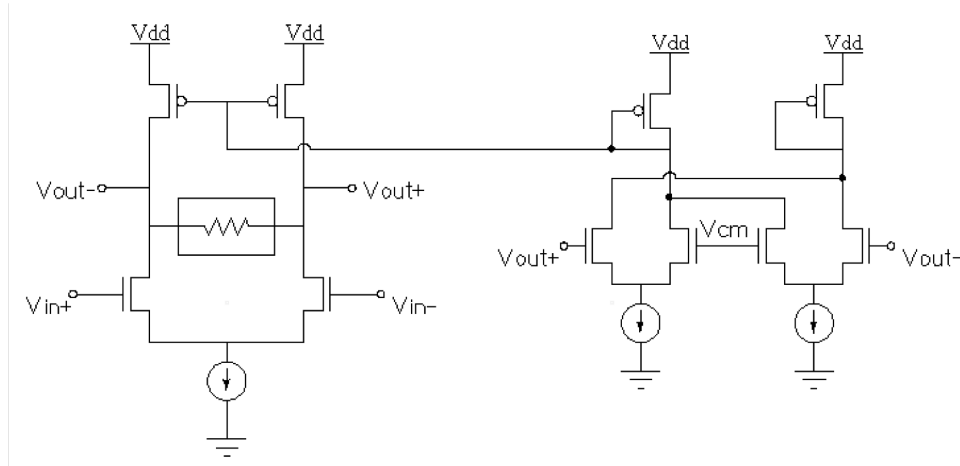


Figure 4.8: IF Amplifier with tunable g_m and tunable output impedance for variable gain and bandwidth.

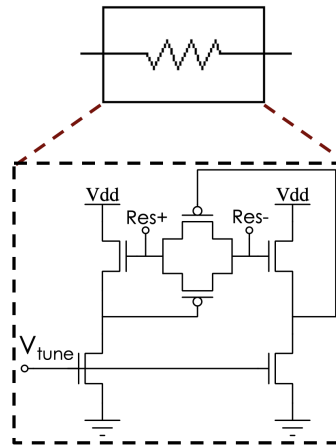


Figure 4.9: Tunable resistor from PMOS devices forced into linear regime.

This resistor is a pair of PMOS devices forced into the linear regime (since $V_{DS} = 0$ V with the DC output voltage on both sides). The n-bias on this circuit is controlled by a voltage source, which in turn pins the source of the upper NMOS devices / the gate voltage of the PMOS devices, and therefore controls the resistance.

Since R_{in} affects both the gain and the bandwidth, a second tuning knob is added to decouple the two. The n-bias current of the diff-amp is also controlled by an external source. Since the current is proportional to the transconductance, g_m , this allows adjustment of the gain of the amplifier independent of the resistor. Therefore this tun-

ing knob provides variable gain while the bias of R_{lin} provides the variable bandwidth. Traditionally, the DC voltage at V_{out} would vary as the operating current is adjusted. A common-mode feedback (CMFB) circuit is used to servo the output voltage to a desired value (V_{CM}). The CMFB does this by adjusting the gate voltage on the PMOS devices. The CMFB uses the same high-gain stage as the diff-amp to achieve a relatively low steady-state error from V_{CM} .

4.4 Limiting Amplifier

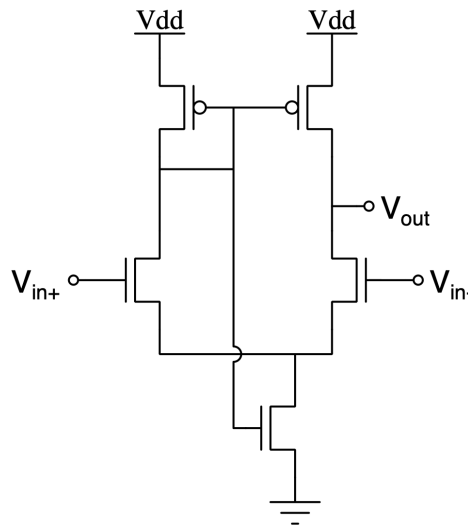


Figure 4.10: Chappell Amplifier

In order to perform digital frequency division, the sinusoidal IF signal must be converted to a digital square-wave. The Chappell Amplifier [42] is chosen as a simple yet robust solution. The circuit utilizes self-biasing as a form of gain boosting to provide a rail-to-rail output. As V_{in-} goes below the input's common mode, the bias node increases, which demands more current in the NMOS devices and less from the PMOS devices, helping V_{in+} pull the output node low faster. Similarly, V_{in-} going high forces less bias current and more current from the PMOS devices, helping the output charge

faster. Ultimately this can convert an IF up to 500 MHz with $\geq 100\text{mV}$ input swing into a full LVCMOS signal.

4.5 Time-to-Digital Converter

This system requires a versatile time-to-digital converter (TDC), capable of taking in the IF signal anywhere between 50-300 MHz, with picosecond resolution at a 10 Msps conversion rate. Conventional TDCs force design trade-offs between complexity, resolution, and full-scale range [43]. One of the prevalent methods for obtaining sub-picosecond resolution exploits a Vernier effect between two delay lines with slightly different stage delays [44]. While this allows picosecond resolution, a textbook linear TDC would dictate that the number of stages would be the full-scale range divided by the resolution. With a 10 Msps conversion rate, the instantaneous signal could theoretically be anywhere between 0-100 ns and would require 100,000 stages in order to not saturate. Yu *et al.* introduced a Vernier Ring TDC (VRTDC), which significantly reduces the number of stages by having the delay lines wrap around in a ring format[45]. This concept will be described in great detail as it forms a solid foundation for the architecture presented below. The original reference and presumably all implementations to-date compare rising edges between two signals of the same frequency, typically utilizing a frequency divider to bring a high-speed oscillator down to a crystal reference frequency. However, this optical synthesizer exhibits instantaneous jitter that may result in the IF signal quickly jumping to frequencies which cause the TDC to saturate for a fixed frequency division and a given reference frequency. In order to keep the TDC reporting phase error linearly in these situations — a desirable trait for the PLL — I introduce a novel modification to the VRTDC which dynamically adjusts the frequency divider to the optimal value, and reports it as an output for post-processing of the overall phase detection. The images

below illustrate the advantage in linearity as a function of timing difference.

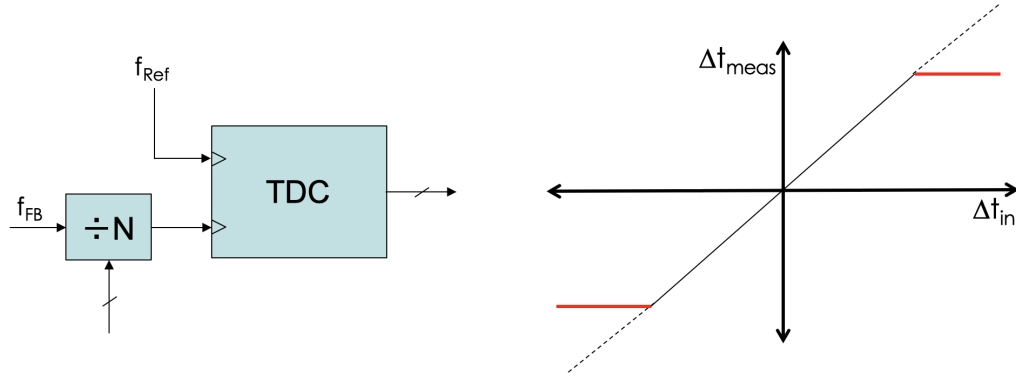


Figure 4.11: TDC implementation with traditional frequency division.

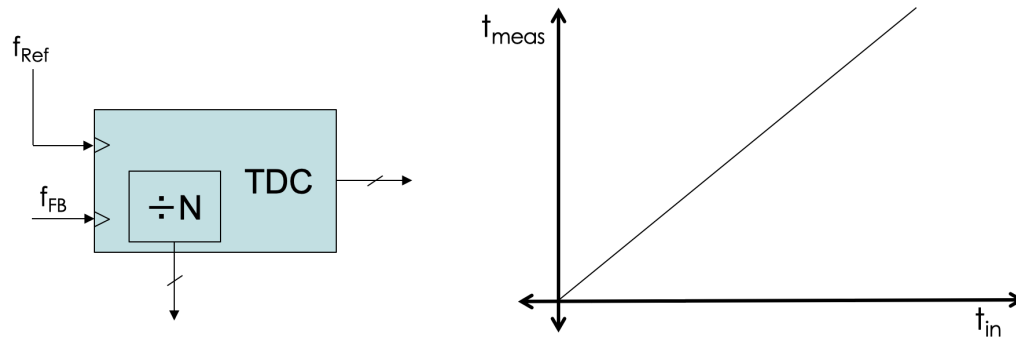


Figure 4.12: Extended range TDC designed to eliminate saturation.

The VRTDC core consists of two delay lines, each oriented as their own ring. After a rising-edge occurs, it propagates along the string of delay cells in a cyclical fashion. A digital counter is used to record how many laps this rising edge makes around the ring before the lagging signal enters the ring. This counter value, N_{coarse} , along with precise knowledge of the time it takes to complete one lap, t_{lap} , yields a coarse approximation of the timing difference. Once the lag signal has entered the ring, it will catch up to the leading signal at a controlled rate. The lag signals delay cells are slightly faster than those in the lead signals path, and the difference sets the resolution of the TDC (t_{res}). A separate counter, N_{fine} , records how many laps it takes for the lag signal to catch

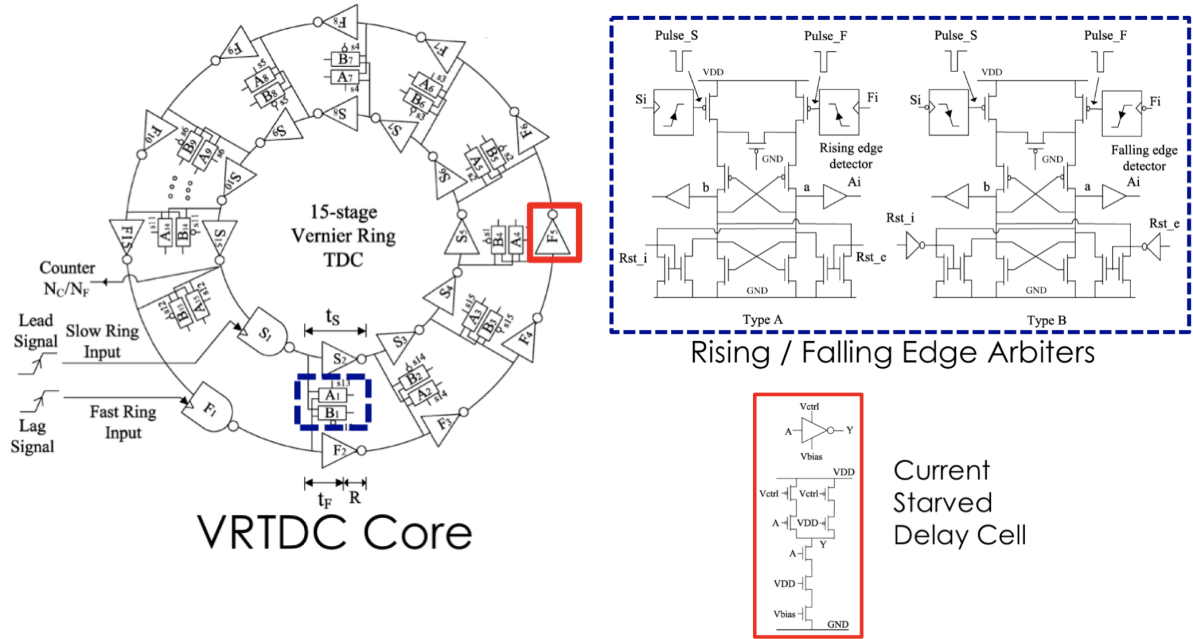


Figure 4.13: Vernier ring time-to-digital converter.

the lead. And finally, the number of arbiters prior to the lead signal being surpassed is captured in a thermometer code. The overall delay can be calculated as:

$$Delay = N_{coarse}t_{lap} + 2(\# \text{ delay cells})N_{finet_{res}} + N_{arbt_{res}} \quad (4.7)$$

The factor of two comes from the fact that an odd number of inverting stages requires two laps around the ring for a rising edge to repeat as a rising edge. For this reason, the architecture requires both rising edge and falling edge arbiters and the thermometer code encompasses twice the number of delay cells. Delay cells are current starved inverters shown in Figure 4.13. The NAND configuration allows the initial rising edge to enter the loop (see S_1/F_1) while maintaining the same parasitic capacitances and propagation delays as the other delay cells. Bringing n_{bias} and p_{bias} further from ground and V_{DD} respectively increases the speed of the inverter. These bias points are pulled out of the IC for characterization and tuning post-fabrication.

The rising and falling edge arbiters are shown in Figure 4.13. The edge detectors trigger a cross-coupled latch. By using the slower paths signal for the reset, it is guaranteed to not detect a false positive when the leading and falling edges are far away. The arbiters are followed by SR latches which will hold a value once the lag signal passes the lead. All 30 outputs go through bubble correction logic, which prevents a single false-positive from giving an incorrect conversion. Finally, the thermometer-to-binary converter rearranges the 30 arbiter outputs as a 5-bit binary word.

In the original reference and presumably all previous VRTDCs, an input arbiter is used to determine which rising edge arrived first and direct it into the slower ring (and the lagging signal is directed to the faster ring). The novelty of this implementation is highlighted in Figure 4.14 in order to accommodate a wide range of frequencies for the feedback signal. Rather than a fixed feedback divider, this dynamic counter records how many feedback pulses arrive before the reference rising edge. Therefore the reference is always treated as the lead signal and the subsequent feedback edge enters the ring as the lagging signal.

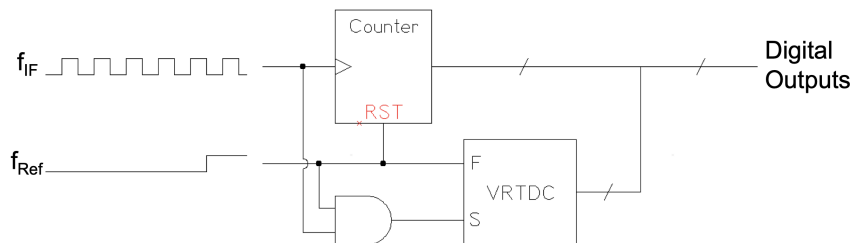


Figure 4.14: Diagram of the inputs to the TDC.

The simulation results in Figure 4.15 demonstrate this as well. The intermediate frequency (red) arrives several times before f_{Ref} . The reference rising edge enters the VRTDC as the lead pulse, and the subsequent f_{offset} triggers the VRTDC lag pulse. After several trips around the ring, the lag signal surpasses the lead and triggers arbiter 5, indicated by the inverter count.

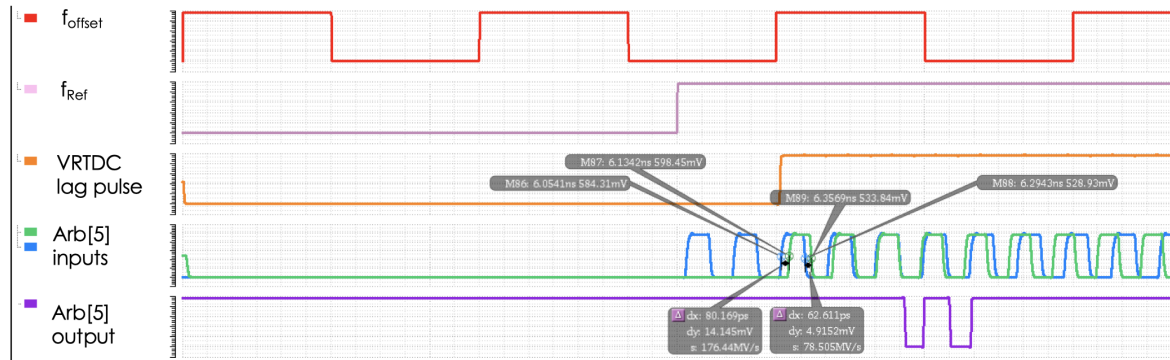


Figure 4.15: Extended range VRTDC simulation.

The physical layout design of the TDC was of critical importance. In order to realize the picosecond resolution theoretically possible in simulations, the layout had to have perfectly matched path lengths for minimized skew. The lead and lag signals needed to be completely isolated as to not affect each other’s arbiter inputs, as well as sufficiently decoupled power supplies. Shown in Figure 4.16, over 5,000 transistors were placed by

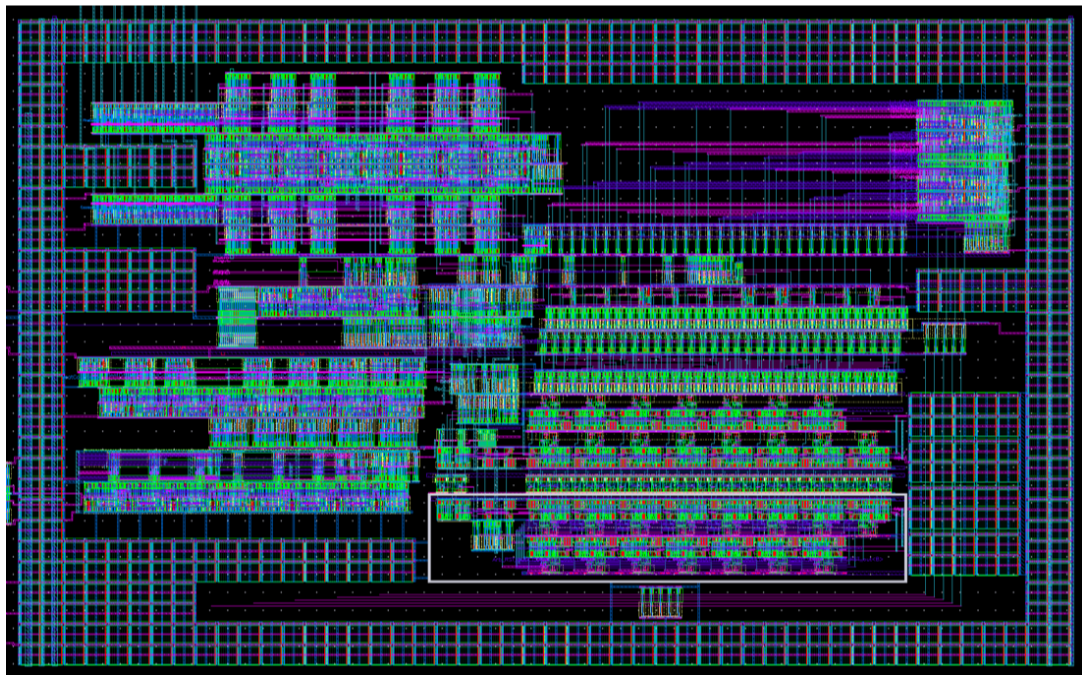


Figure 4.16: Extended range VRTDC layout.

hand to achieve this. The two rings are shown on the bottom right, with identical designs and separate triple well guard rings to minimize substrate coupling. The surrounding circuitry encompasses all of the various counters. Finally, all spare area was filled with FET-based decoupling capacitors, tied to the core digital supplies.

4.6 Peak Detector

As explained in Section 3.5, the scan-and-lock algorithm is critical to finding the beatnote when the OFS is first enabled or after switching to a new comb-tooth. The COTS solution utilized the existing ADC to create a digital peak detector and keep track of when the LO was properly aligned to the incoming optical beatnote. This method assumed an ADC that ran above the Nyquist frequency set by the IF, a requirement already met when the ADC was in the main signal path for I/Q phase detection. Instead, the circuit in Figure 4.17 performs a peak detection in the analog domain. The output settles to a DC voltage proportional the input signal's amplitude, and this significantly eases the ADC design constraints for quantizing this information.

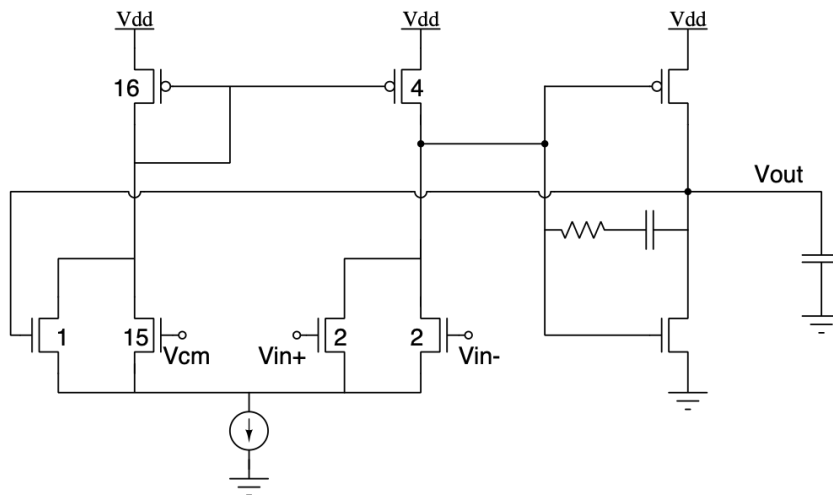


Figure 4.17: Peak detector with amplification.

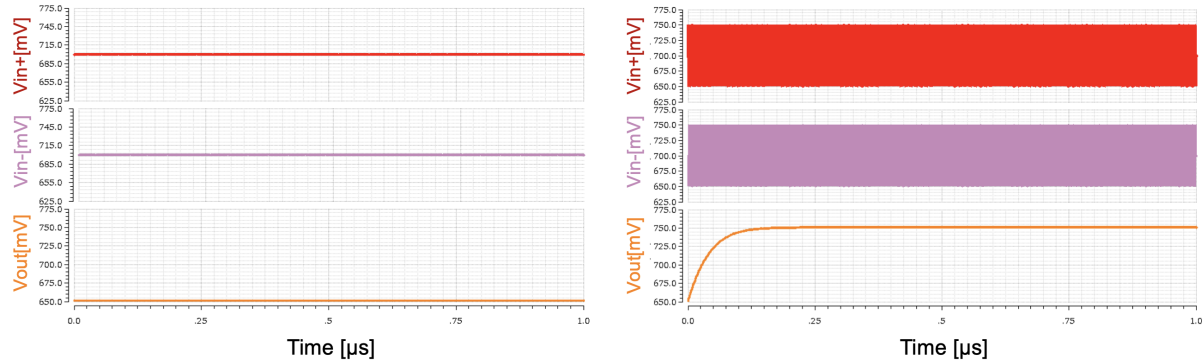


Figure 4.18: Peak detector simulation for 500 microvolt (left) and 60 millivolt (right) peak-to-peak inputs.

At first glance, this topology has a lot of similarities to a two-stage amplifier in a single-ended unity gain configuration, but with both differential inputs connected on the same side. When the input amplitude is a significant portion of the overdrive voltage ($V_{gs} - V_{th}$), all of the current will flow through whichever transistor ($V_{in+/-}$) has its input in a sinusoidal upswing, and the complimentary transistor will be cut-off (its input in a sinusoidal downswing). Since both currents are summed, this can be thought of as a full wave rectification (sum of the two half waves). The output, pinned through negative feedback, will produce a low-pass filtered version of the rectification, with an average value proportional to the amplitude of the input swing, V_{peak} .

$$\begin{aligned}
 V_{Avg} &= \frac{1}{\pi} \int_0^{\pi} V_{peak} \sin(\theta) d\theta \\
 &= -\frac{V_{peak}}{\pi} \cos(\theta) \Big|_0^{\pi} \\
 &= \frac{2V_{peak}}{\pi}
 \end{aligned} \tag{4.8}$$

The cutoff frequency — set by the second stage's output resistance and the large output capacitor — determines the trade-off between the voltage ripple and the settling time of the circuit. Quenching the rectified sinewave to a DC voltage would be ideal, but the settling time of such a filter could create a latency bottleneck in the overall time

required for the scan-and-lock algorithm. Therefore, the load capacitor was sized to set the cutoff frequency two orders of magnitude below the intermediate frequency. For the first order low-pass filter at the output:

$$\begin{aligned} \text{Attenuation} &= 20 \log \left(\frac{\omega_{cutoff}}{\omega_{IF}} \right) \\ &= 40dB \end{aligned} \tag{4.9}$$

The transistors on the left side of the circuit were uniquely sized to give this peak detector additional gain. A concept taken from [46], $\frac{15}{16}$ ths of the current on the left is biased by a DC source tied to the common-mode voltage of the input. With no input amplitude, V_{Out} will also sit at V_{CM} to make $I_{Left} = 4I_{Right}$. When the input amplitude is significant, the single transistor in feedback will have to climb faster to match the right side's current, and ultimately the output voltage will rise $\sim 4x$ higher than V_{avg} (given $I_{DS} \propto V_{gs}^2$). This led to a manageable full-scale range and subsequently facilitated a straightforward ADC design.

4.7 Flash ADC

The analog-to-digital converter only requires 4-bits of accuracy for peak detection and operation <10 Msps. The flash design in Figure 4.19 easily achieves this with a linear resistor ladder between a positive and negative reference [47]. The negative reference is pinned to the common-mode of the peak detectors input and the positive is externally controlled to match the full-scale range from the peak detector. I chose a StrongARM Latch for the comparator implementation [48]. This circuit consumes relatively small area and zero static power, both advantageous attributes since there are 16 comparators in parallel. The dynamic latch design also exhibits no hysteresis and ± 1 mV input-referred noise from input diff-pair. The comparators are sampled by a programmable clock source

that is generated as a divided down version of the reference input. The converter outputs (16) are reduced to 4 signals through a priority encoder, and finally these are latched to hold the data for a full cycle.

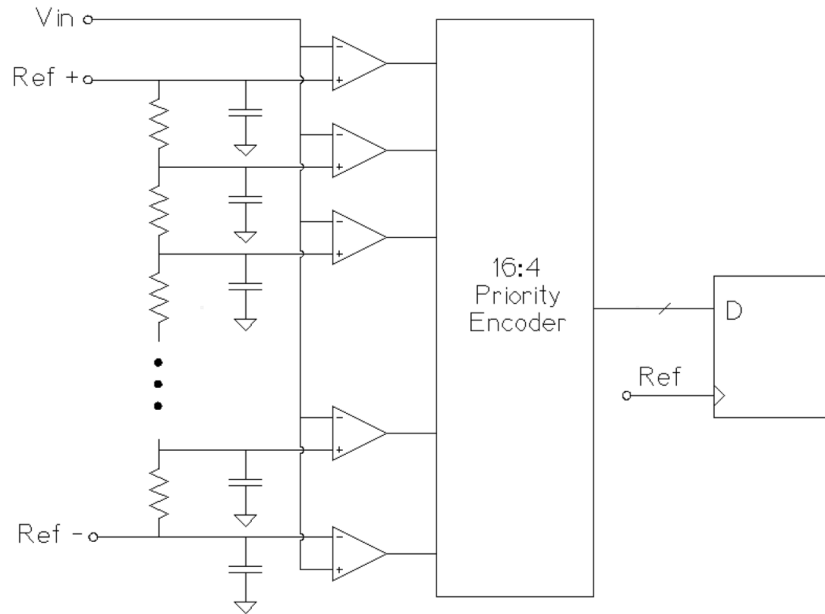


Figure 4.19: Flash ADC design.

4.8 Injection-Locked Frequency Divider

The incoming comb-tooth repetition rate is expected near 15 GHz with several millivolts of amplitude. Conventional techniques are too slow to process this signal directly or amplify it to a full digital square-wave. An injection-locked frequency divider (ILFD), shown in Figure 4.20, was chosen to provide square-wave rectification and frequency division, both of which made the subsequent signal processing more manageable[49, 50]. Without any RF input, this circuit acts as a differential delay cell ring oscillator. As the bias current is increased, the effective resistance of the pull down FETs decreases, resulting in a higher frequency for 2π phase shift around the entire loop (Barkhausen

criterion for oscillation [51]). The resistive loads were chosen as opposed to PMOS devices to give a faster intrinsic frequency range. The FETs were sized 10x larger than the standard digital inverter, so that loading the output with a digital buffer wouldn't diminish speed performance. The driven FETs are 3x larger than the cross-coupled FETs so that the positive feedback loop doesn't compete significantly against the next incoming edge, which slow down the transition time. Finally, the resistor value was designed for a Goldilocks value. Too large a resistance would have caused an RC limitation for rising edge transitions, and too small a value would have prevented the falling edge transitions from fully approaching GND.

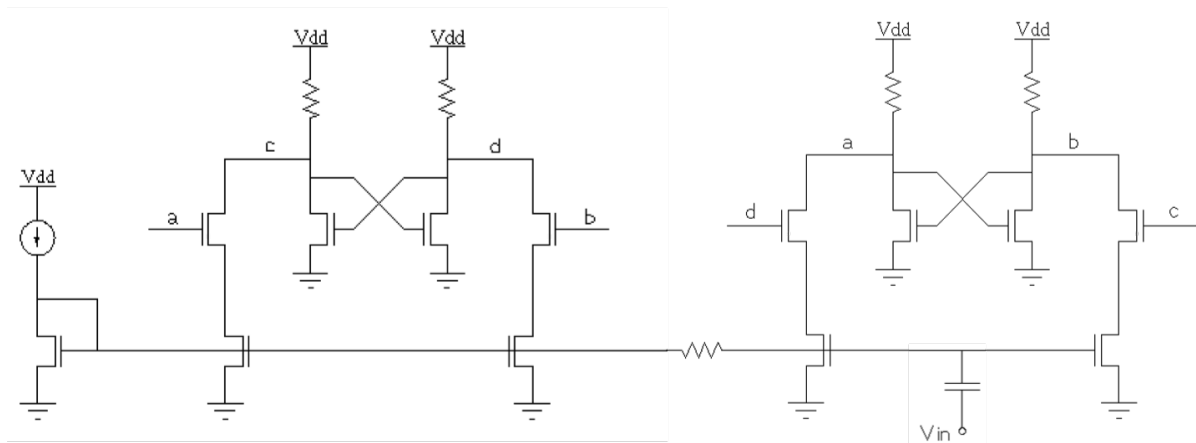


Figure 4.20: Inductorless injection-locked frequency divider.

Due to the regenerative cross-coupled FETs, the dynamic current draw of the circuit is highest at the zero-crossing point for the differential inputs, and the current reaches a minimum when the nodes settle near VDD / GND. The current draw essentially oscillates at twice the resonant frequency of the circuit and for this reason, an RF input can be coupled to the bias and the output will be pulled into phase and frequency alignment at $f_{in}/2$. Farazian *et al.* provide rigorous analysis to understand the limitations of the injection locking (pull-in range, injection current amplitude) for this particular topology [52]. In practicality, I chose to make the current source an external off-chip bias point

so that the natural resonance frequency could be set precisely post-fabrication. This is chosen as 7.5 GHz for the target 15 GHz comb-tooth repetition rate, but could be feasibly adjusted to fit other microresonators. Simulations (with parasitic extraction from the ASIC layout) show a ± 3 GHz pull-in range for a 1 mV input. This is significantly larger than the ± 100 MHz of worst case frequency deviation on the microresonator's repetition rate, and this circuit is therefore a suitable solution. Note that this topology is also inductorless, giving it a drastic area reduction compared to similar injection-locked designs.

4.9 Digital Interfaces

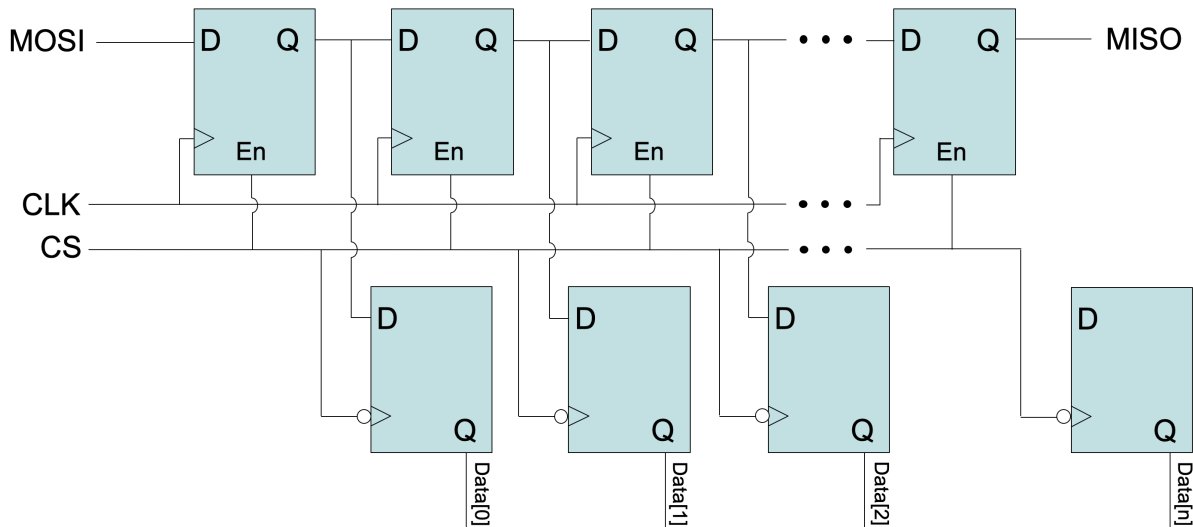


Figure 4.21: Serial Protocol Interface (SPI) implementation.

Several separate ports are used for digital inputs and outputs, each with a unique purpose. Most programmability (roughly 60 digital signals), excluding the PLL, is expected to be written once at start-up and remain constant throughout operation. A SPI interface is best suited to provide this low-speed functionality while occupying only four bondpads on the chip [53]. A master-out slave-in (MOSI) pin from the FPGA carries the

data, synchronized to a clock pin. An on-chip shift register retains this data in a local cascade of flip flops. A chip select (CS) pin lets the ASIC know when to begin shifting and when the full string has been read into the shift register, and subsequently latches the data and sends it across the chip to the various components with this programmability. The fourth pin, traditionally a master-in slave-out (MISO) with data from the ASIC, carries the data previously sent by the FPGA, which provides a loopback to self-check that the proper information was received.

The PLL has a combined 22 bits of data that need to be updated as different LO frequencies are requested. Therefore, the PLL is placed on its own SPI interface, so that the overall update latency can be minimized without having to send the overhead of the unchanging bits. Finally, the TDC outputs and flash ADC conversions need to be sent to the FPGA at rates upwards of 10 MHz. These are kept as parallel digital outputs, each assigned an ASIC bondpad and FPGA pin.

4.10 IC Fabrication and Results

The layout and interconnectivity of all components is shown in Figure 4.22 (left). Several of the top-metal layers are hidden which contain robust power gridding. Most of

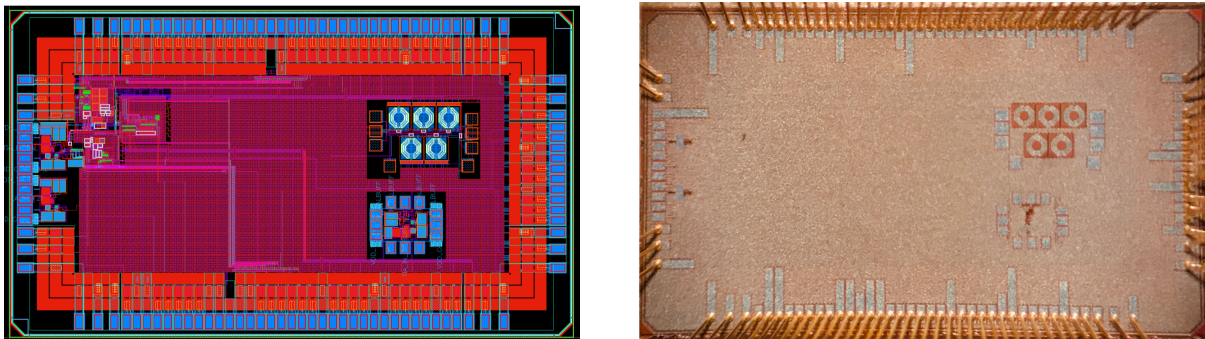


Figure 4.22: Top-level view of the ASIC layout (left) and the fabricated chip (right).

the processing takes place in the top-left corner, and the remainder of the chip is used for standalone test circuitry and routing of digital I/O. In order to maximize performance, the bond pads around the edge are ordered such that critical traces (reference clocks and power supplies) would have minimal bond-wire lengths and on-chip routing. The fabricated and wire-bonded IC is shown above as well.

A circuit board was designed to initially house the packaged IC and fully characterize its functionality. Similar to the synthesizer's communication hierarchy, all of the external sources (DACs) and digital I/O are displayed on a GUI interface with commands sent to and from the FPGA/SoC. The NB6L295M, a programmable delay chip from ON Semiconductor, yields 3-9 nanoseconds of delay in linear 11 picosecond steps [54]. This provided a great method for characterizing the TDC, shown in Figure 4.23. The delay was swept linearly and the inputs to the TDC were tuned ($N_{bias,fast}$, $N_{bias,slow}$, $P_{bias,fast}$, $P_{bias,slow}$) so that the output tracked linearly as well. Note that these four knobs set the individual stage delay (rising and falling edges matched), as well as the time required to complete one lap around the ring.

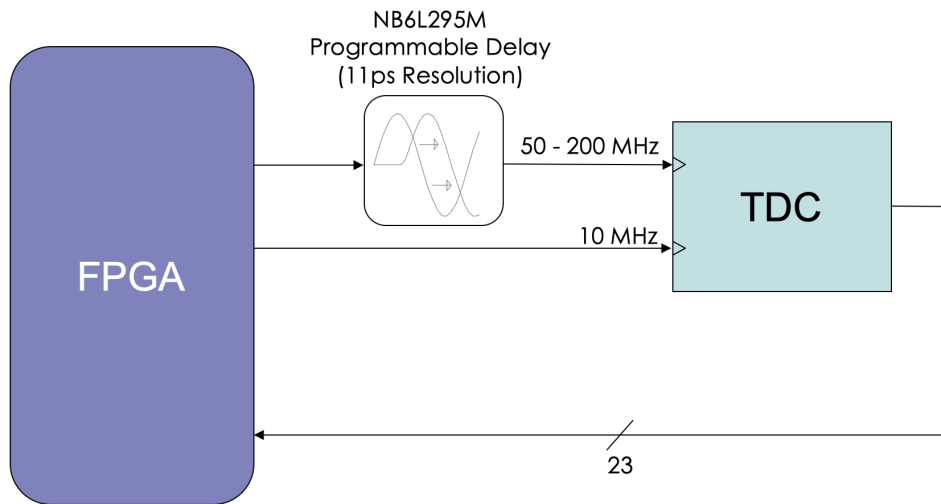


Figure 4.23: TDC test fixture.

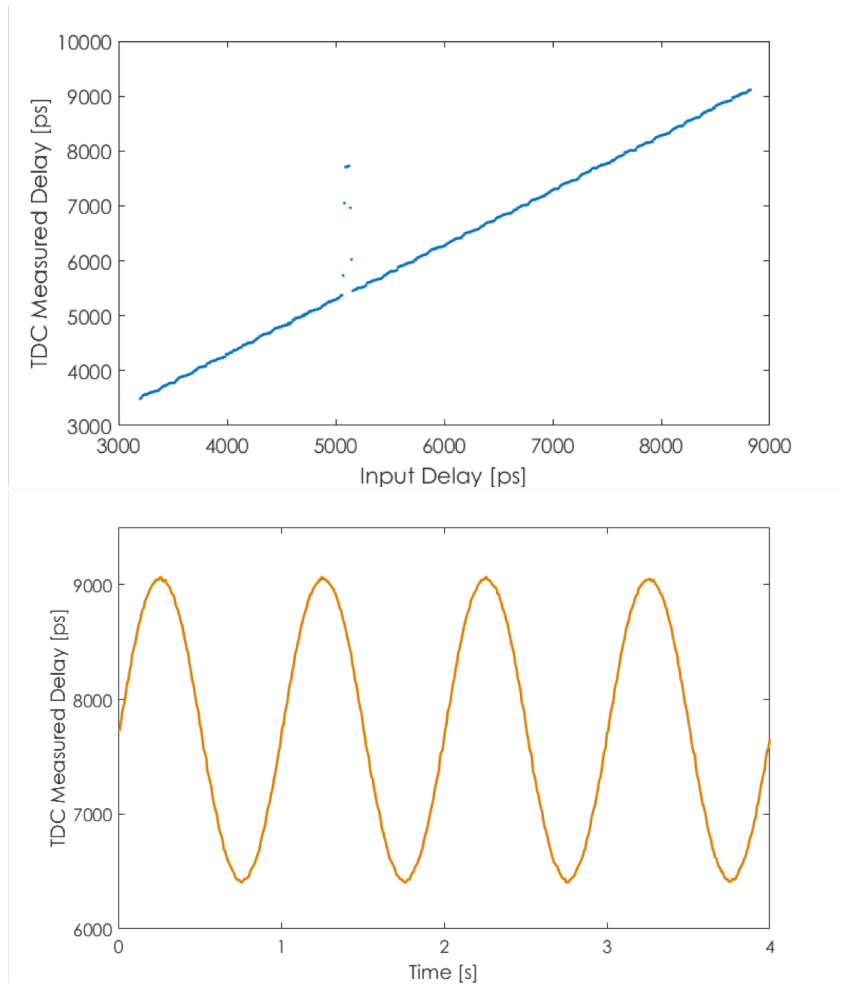


Figure 4.24: TDC results.

The glitches in the middle of the TDC output (around 5000 ps delay) are caused by a race condition where an extra N_{Coarse} is registered. Luckily this is a well defined case which can be simply corrected in the FPGA. In addition to the linear sweep, a function was added to modulate the delay of the input in a sinusoidal fashion, confirming there was no time-dependence in the TDC.

The on-chip local oscillator was characterized by pulling out the divided down version of the VCO, at the rep rate of $f_{PFD} = f_{Ref} = f_{VCO}/N$. The PLL was able to lock the oscillator up to 8.25 GHz, sufficient for the full range of optical beatnote. The lower

frequency limit was roughly 500 MHz. As the supply voltage of the VCO dropped to several hundred millivolts, the gain of the level translator was too low to bring the signal to full swing. If a desired optical frequency fell below 500 MHz from a comb tooth, this could be mitigated by utilizing the pump lasers feedback loop to move the optical comb further away so that the beatnote would be >500 MHz.

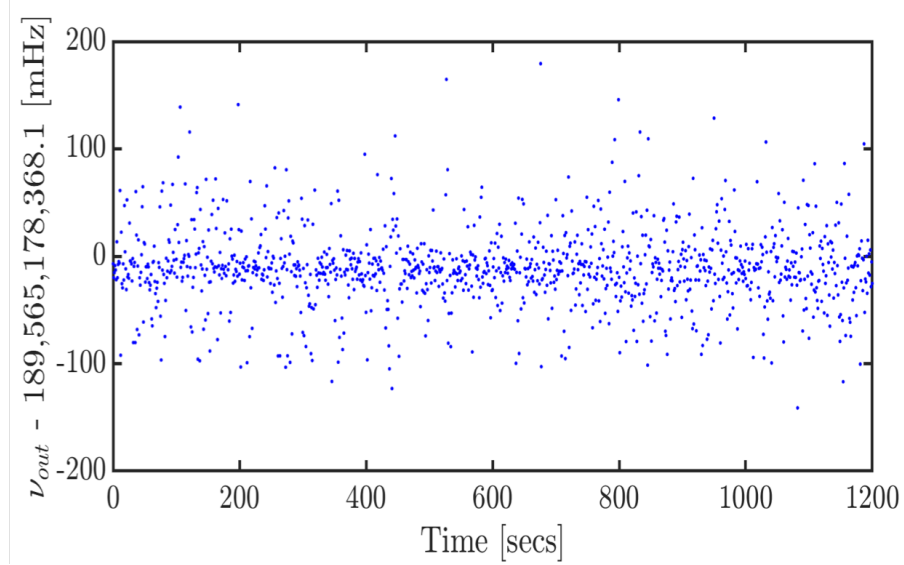


Figure 4.25: Optical frequency synthesis achieved with the VRTDC.

As a first demonstration, the VRTDC was used to complete the frequency synthesizer loop with a commercial off-the-shelf TIA, LNA, and LO/mixer. Figure 4.25 plots the stability recorded on a frequency counter for 20 minutes (1 second gating). With most data points contained within ± 50 mHz, this is approaching the same level of frequency uncertainty demonstrated at the board-level. An ESA measurement also plots the power spectral density to show the signal was contained within the 1 Hz resolution bandwidth (instrument limitation). Furthermore, the beatnote setpoint was adjustable in real-time and the TDC-based PLL corrected the phase errors accordingly. Several 3.2 kHz steps were manually requested over the course of 100 seconds in Figure 4.27. These results shows the promise of the architecture and affirm the decision to move to a heterodyne

receiver with frequency down-conversion and a digital phase detection. However, the full signal-path of the ASIC is still being debugged and a fully integrated synthesis has yet to be achieved.

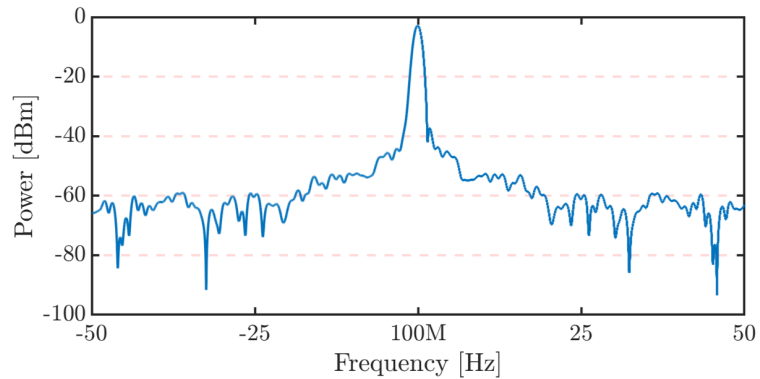


Figure 4.26: Power spectral density of the locked beatnote. 1 Hz Resolution Bandwidth and 100 Hz full span, centered at 100 MHz.

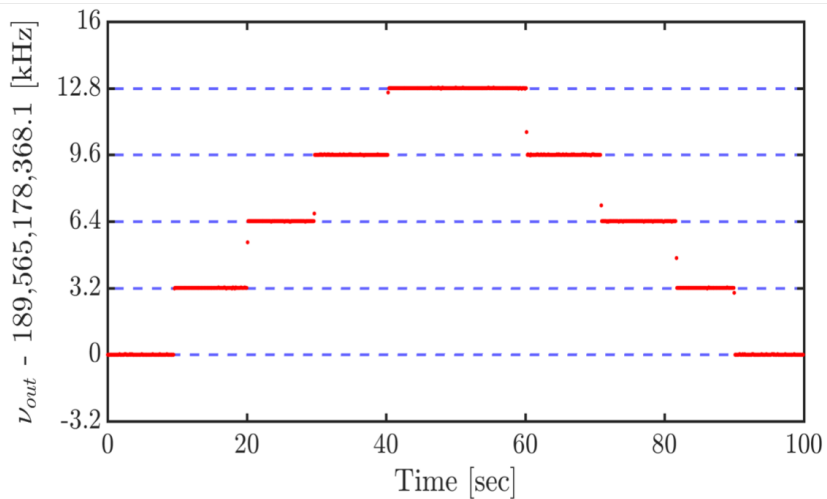


Figure 4.27: Frequency ramp of the locked beatnote.

Chapter 5

Future Work

This thesis focused primarily on locking the tunable laser to the frequency comb. However, as Section 2.2 alluded to, there are several other signals to detect and servo for true self-referenced synthesis. The 15 GHz rep rate, which the ASIC detects, gets locked to the crystal reference in a separate phase-locked loop. A heater under the ring resonator finely adjusts the waveguide length and therefore the repetition rate. The terahertz-spaced comb also needs its repetition rate locked to the 15 GHz comb (66rd tooth). The beatnote can be detected with the same heterodyne receiver used for the TL lock, and a heater under the ring resonator adjusts the spacing in the same mechanism. The final signal, f_0 , has too low of a signal-to-noise ratio for conversion to a digital squarewave. My colleagues have recently designed another ASIC which has the same architecture as the PCB prototype. The high-speed ADC will convert to the digital domain and the subsequent FPGA will perform a narrow bandpass filter (3MHz) to detect the signal. Having this done in the digital domain will allow multiple bandpasses in parallel, to accommodate frequency jitter and allow real-time tracking. The complete synthesizer will feature this new chip and two copies of the ASIC presented in Chapter 4.

The control loop described in this dissertation is a rudimentary take on continuous

control. The heaters are currently set to discrete values with control on the gain section giving instantaneous feedback. In an ideal implementation, the heaters would be moved in real-time as well to keep the laser in a stable regime across all 50nm. One method to do this would involve calculating the gain of each heater (K_{VCO}) and moving all of the tuning knobs together. I imagine the final solution could dither each heater simultaneously at different modulation frequencies and use the frequency content to calculate each gain. Furthermore, the settling time for wavelength hopping is limited by thermal settling of the PICs. Initial experiments show that this can be improved by overdriving the heaters with pre-emphasis. Fully understanding the characteristics of this could lead to improved switching performance. The high-speed controls theory of such PICs could be an entire thesis in and of itself.

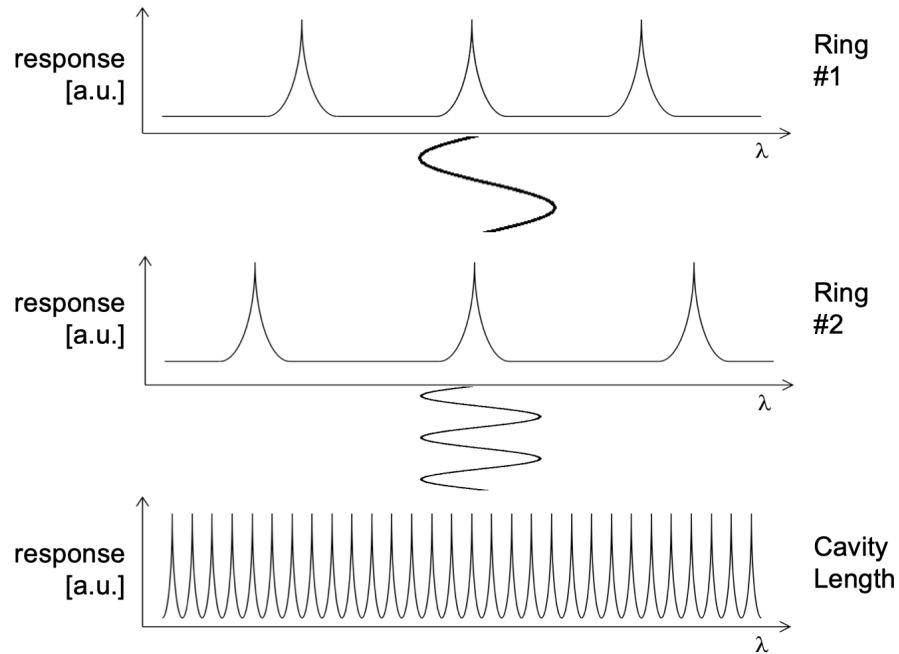


Figure 5.1: Image that depicts frequency modulation as a method for calculating the gain of the ring resonators.

Finally, the ultimate reduction in SWaP will incorporate all of the circuitry in a single footprint. This is an ongoing task with close collaboration between all of the chip designers and a packaging expert. The first step involves removing the current ceramic package housing the ASIC, an excessive footprint with lengthy wirebonds that hinder RF performance. My colleagues are developing a wafer-level chip-scale package (WLCSP) by implanting the chip in a silicon wafer and adding a redistribution layer on top of the chip to create ball grid array (BGA) connections. These chips will then be housed on an interposer board that wires to the 3 ASICs to the FPGA as well as the laser driver circuitry. The photonic ICs are mechanically mounted on the other side of the board. A 3D rendering of the design is shown in Figure 5.3. A miniaturized SMA carries the reference clock input and a USB micro provides 5 volt power and the PC interface. The tunable lasers output can be fiber spliced into any standard system.

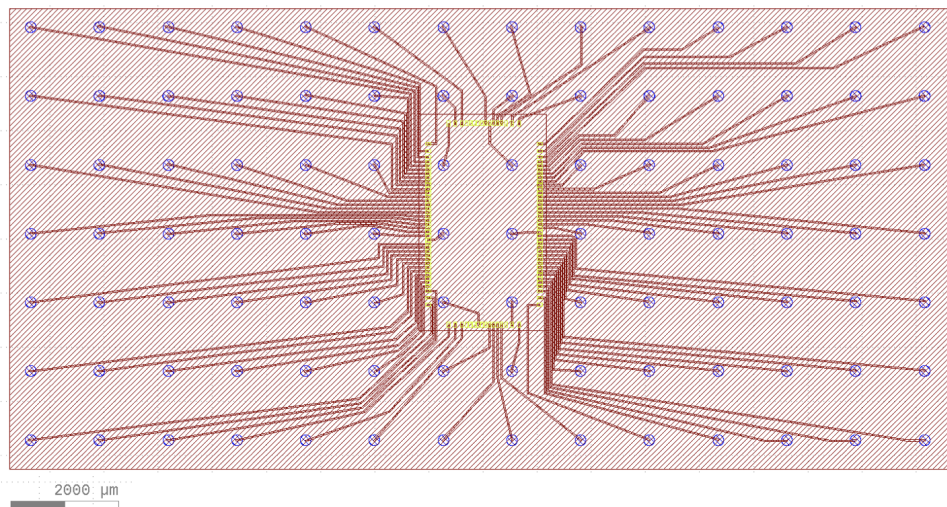


Figure 5.2: Mask of a redistribution layer used to develop a ball grid array footprint.

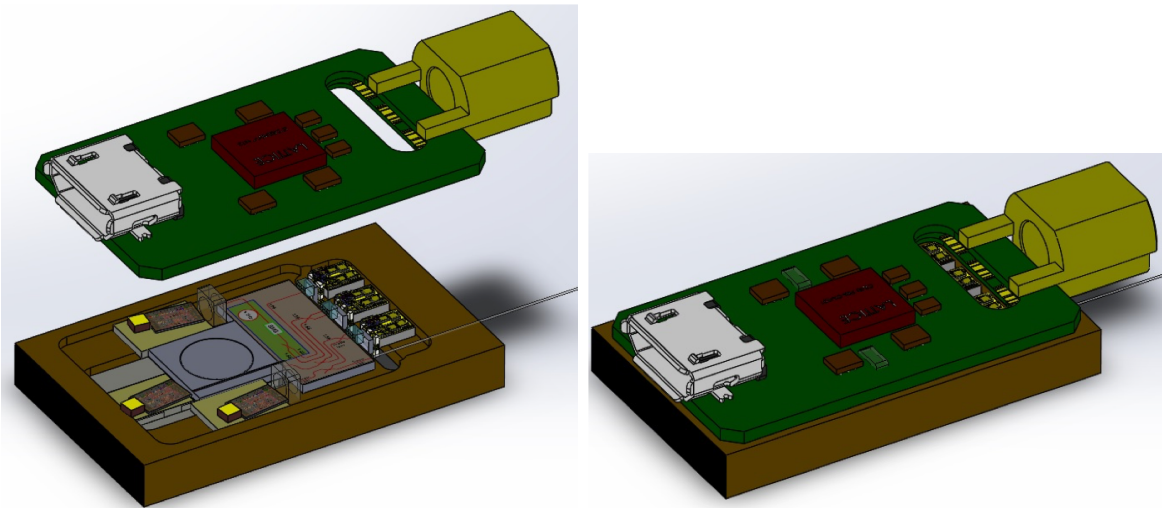


Figure 5.3: 3D rendering of the packaged optical frequency synthesizer.

Chapter 6

Conclusion

This project was motivated by the goal of creating a versatile optical frequency synthesizer, both miniaturized and widely tunable so that it could be used for any application imaginable. This was made possible by leveraging recent advances in the photonics community as well as the closely co-designed electronics described above. Shrinking the optical footprint to low-power PICs brought numerous challenges for frequency stability. Microwatts of comb tooth power translated to a signal-to-noise ratio limitation, which was overcome by a heterodyne down-conversion receiver. In order to achieve a wide dynamic range, the architecture utilized digital control to implement coarse and fine tuning knobs. The digital domain also facilitated communication to a custom user-friendly software. An initial prototype at the board-level achieved <1 Hz resolution with 50 nm of tuning range and state-of-the-art frequency stability. Ultimately, this influenced an ASIC design to improve upon the RF performance as well as provide the greatest SWaP reduction. The ASIC was highlighted by several novel circuits including a wideband local oscillator, a picosecond resolution extended range vernier ring time-to-digital converter, as well as a standalone peak detector signal path.

This work was funded by the Direct On-Chip Digital Optical Synthesizer (DODOS)

initiative from the Defense Advanced Research Project Agency, Microsystems Technology Office (DARPA MTO).

Appendix A

Source Code

This appendix highlights snippets of code that were critical to implementing the synthesizer described above. The first section details the various finite state machines, each operating independently while maintaining data channels to the processor and neighboring execution blocks. Section A.2 describes the processor's mechanism for communication with a PC interface, and Section A.3 shows the corresponding code which completes a custom protocol in a Microsoft Visual Studio GUI.

A.1 Verilog Code

Xilinx System Block Diagram (Figure A.1) is used to automatically configure the clocking and AXI memory interface between the hard-coded ARM microprocessor and the custom IP blocks described below. This also generates the wiring between the IP blocks as specified (arcTan \rightarrow Phase Comparator \rightarrow Loop Filter \rightarrow DAC Interface).

The code for the CORDIC arctangent (described in Section 3.1) is shown below. The case statements first check the I/Q (**Xin/Yin**) sign bits to determine which 90° quadrant the input lies within. The inputs are reoriented to an equivalent 0-90° window and then the CORDIC pipeline computes the remaining LSBs.

```

always @(negedge adc_clock) begin

    Xin <= chB_in;
    Yin <= chA_in;
end

always @(posedge adc_clock) begin
    case (quadrant)
        2'b00:
            begin
                if(Xin == 8'b0) begin
                    X[0] <= (8'b01111111 << 6);
                end else begin
                    X[0] <= Xin_neg[6:0] << 6;
                end
                if(Yin == 8'b0) begin
                    Y[0] <= (8'b01111111 << 6);
                end else begin
                    Y[0] <= Yin_neg[6:0] << 6;
                end
                Z[0] <= {2'b10, angle[29:0]};
            end
        2'b01:
            begin
                Y[0] <= ({1'b0, Xin[6:0]} << 6);
                if(Yin == 8'b0) begin
                    X[0] <= (8'b01111111 << 6);
                end else begin
                    X[0] <= ({1'b0, Yin_neg[6:0]} << 6);
                end
                Z[0] <= {2'b11, angle[29:0]};
            end
        2'b10:
            begin
                X[0] <= ({1'b0, Yin[6:0]} << 6);
                if(Xin == 8'b0) begin
                    Y[0] <= (8'b01111111 << 6);
                end else begin
                    Y[0] <= (Xin_neg << 6);
                end
                Z[0] <= {2'b01, angle[29:0]};
            end
        2'b11:
            begin
                X[0] <= ({1'b0, Xin[6:0]} << 6);
                Y[0] <= ({1'b0, Yin[6:0]} << 6);
                Z[0] <= {2'b00, angle[29:0]};
            end
    endcase
end

```

Figure A.2: CORDIC quadrant selection (Verilog code).

Figure A.3 highlights the CORDIC pipeline stage generation and execution. Recall that stage 1 determines if the instantaneous phase is greater or less than 45 degrees, and then subtracts or adds 22.5° accordingly, so stage 2 can determine the next bit in the next clock cycle in a successive approximation manner. The **Z** register stores the sum of these results, where the angles for each pipelined stage come from a pre-computed (fixed) lookup table, **atan_table**.

```

//-----
//                               generate stages 1 to STG-1
//-----
genvar i;

generate
for (i=0; i < (STG-1); i=i+1)
begin: XYZ
    wire          Z_sign;
    wire signed [XY_SZ:0] X_shr, Y_shr;

    assign X_shr = X[i] >>> i; // signed shift right
    assign Y_shr = Y[i] >>> i;

    //the sign of the current rotation angle
    assign Z_sign = Y[i][XY_SZ]; // Z_sign = 1 if Y[i] < 0

    always @(negedge adc_clock)
    begin
        // add/subtract shifted data
        X[i+1] <= Z_sign ? X[i] - Y_shr      : X[i] + Y_shr;
        Y[i+1] <= Z_sign ? Y[i] + X_shr      : Y[i] - X_shr;
        Z[i+1] <= Z_sign ? Z[i] - atan_table[i] : Z[i] + atan_table[i];
    end
end
endgenerate

//-----
//                               output
//-----
assign Xout = X[STG-1];
assign Yout = Y[STG-1];
assign Zout = Z[STG-1];
assign trig_wire = ~adc_clock_d2 & adc_clock_d3;

endmodule

```

Figure A.3: CORDIC pipeline generation and execution (Verilog code).

```

always@(negedge trigger) begin

    if(reset == 1) begin
        error <= 34'd0;
        meas_rot <= 1'b0;
        exp_rot <= 1'b0;
    end else begin

        case({meas_roll,exp_roll})

            2'b00: begin
                error <= {1'b0, exp_rot, ph_exp} - {1'b0, meas_rot, ph_meas_latched};
            end
            2'b10: begin
                if(exp_rot) begin
                    exp_rot <= 1'b0;
                    error <= {2'b00, ph_exp} - {2'b00, ph_meas_latched};
                    entered_here <= 1'b1;
                end else begin
                    meas_rot <= 1'b1;
                    error <= {2'b00, ph_exp} - {2'b01, ph_meas_latched};
                    entered_here <= 1'b0;
                end
            end
            2'b01: begin
                if(meas_rot) begin
                    meas_rot <= 1'b0;
                    error <= {2'b00, ph_exp} - {2'b00, ph_meas_latched};
                end else begin
                    exp_rot <= 1'b1;
                    error <= {2'b01, ph_exp} - {2'b00, ph_meas_latched};
                end
            end
            2'b11: begin
                exp_rot <= 1'b0;
                meas_rot <= 1'b0;
                error <= {1'b0, exp_rot, ph_exp} - {1'b0, meas_rot, ph_meas_latched};
            end
        endcase
    end

end

assign meas_roll = (ph_meas_latched < ph_meas_reg);
assign exp_roll = (ph_exp < ph_exp_reg);

```

Figure A.4: Digital phase comparator (Verilog code).

The phase comparator code is provided in Figure A.4. The instantaneous phase, **ph_meas_latched**, is compared to a digital accumulator (not shown) which sets the expected phase, **ph_exp**. The phase error should be a simple subtraction of the two linear phases (case **2'b00**), and yet most of the Verilog deals with the hazard cases when either of the two signals is rolling over from a completion of 2π phase rotation. Imagine a slight phase offset between the expected and measured signals, where the expected has rolled over to 0.1π , but the measured signal is at 1.9π . The phase error

should be 0.2π but a simple phase subtraction might report this as -1.9π . For this reason, **meas_rot** and **exp_rot** are extra bits added to maintain if either one of the signals has completed an extra rotation before the other. Case **2'b10** and case **2'b01** deal with either phase rolling over by itself in a given cycle, and case **2'b11** handles both phases rolling over. The resulting output vs. phase shows linearity up to $\pm 4\pi$ and maintains proper direction for all possible inputs, a notable improvement over analog detectors.

Following the phase comparator, the programmable loop filter manipulates the error signal to create the feedback signal, with reconfigurability in real time. Digital integrators, low-pass filters, high-pass filters, and gain blocks might be desired in different scenarios. Figure A.5 shows the snippet of code utilized to mux between these different operators. Currently, two separate filters are muxable in real-time through **slv_reg0[2:1]** and **slv_reg0[4:3]** and summed together to create an output. Two inputs are ideal for creating a lead-lag compensator (integrator plus DC or integrator plus low-pass). The Verilog code also contains bypass capabilities for routing the ADC inputs, ArcTan output, and phase comparator output directly to the DAC. Similarly, all are real-time programmable through bits **slv_reg0[10:8]** which could be written from the GUI. This debug functionality was absolutely pivotal throughout the bring-up process.

All of the loop filters are implemented as multiplierless, shift-register based operators. Compared to conventional designs this yields significantly lower power and area consumption, with minimal effects on the loop dynamics. The IIR-based integrator is presented as an example in Figure A.6. The unity gain frequency is set by **Ts**, a slave register variable controlled by the GUI. Before committing the filtered value to the main loop, the code checks for underflow and overflow conditions and if violated, saturates the output accordingly.


```

always @(posedge trigger) begin

    case( slv_reg0[2:1] )
        2'b00: filter1 <= int_out0;
        2'b01: filter1 <= diff_out0;
        2'b10: filter1 <= lpf_out0;
        2'b11: filter1 <= chA_in_latched <<< p0_ts;
    endcase

    case( slv_reg0[4:3] )
        2'b00: filter0 <= int_out1;
        2'b01: filter0 <= diff_out1;
        2'b10: filter0 <= lpf_out1;
        2'b11: filter0 <= chA_in_latched <<< p1_ts;
    endcase

    case( muxed_sum_temp[32:31])          // takes care of overflow and underflow
        2'b01: muxed_sum <= 32'h7FFFFFFF;
        2'b10: muxed_sum <= 32'h80000000;
        default: muxed_sum <= muxed_sum_temp[31:0];
    endcase

    //Bypass code (Rev 2.0)
    case( slv_reg0[10:8] )
        3'b000: shift_out <= muxed_sum; // Filter output
        3'b001: shift_out <= 32'h80000000 + {Xin,24'd0}; //direct from ADC, channel 1
        3'b010: shift_out <= 32'h80000000 + {Yin,24'd0}; //direct from ADC, channel 2
        3'b011: shift_out <= 32'h80000000 + atan_op; // Arc tangent result
        3'b100: shift_out <= chA_in; // Phase difference - remnant of some terrible naming convention
        default: shift_out <= muxed_sum;
    endcase

    if(slv_reg0[7]) begin                // 1 is for 2s compliment and 0 is for binary offset
        off_binary <= shift_out;
        off_binary_alt <= shift_out + 32'h01000000;
    end else begin
        off_binary <= shift_out + 32'h80000000;
        off_binary_alt <= shift_out + 32'h81000000;
    end
end

assign muxed_sum_temp = {filter0[31], filter0[31:0]} + {filter1[31], filter1[31:0]};

```

Figure A.5: Loop filter mux (Verilog code).

```
module Integrator(
    input trigger,
    input reset,
    input signed [31:0] chA_in,
    input signed [31:0] Ts,
    output reg signed [31:0] filter_out
);

    reg signed [31:0] filter_out_delay = 32'b0;
    reg signed [31:0] chA_delay = 32'b0;
    reg signed [32:0] filter_out_temp;

    always @(posedge trigger) begin

        if(reset) begin
            filter_out_delay <= 32'b0;
            chA_delay <= 32'b0;
            filter_out_temp <= 33'b0;
            filter_out <= 32'b0;
        end else begin
            filter_out_delay <= filter_out;
            chA_delay <= chA_in;
            filter_out_temp <= {filter_out_delay[31],filter_out_delay} + {chA_delay[31],($signed(chA_delay) >>> Ts)};
            case(filter_out_temp[32:31])
                2'b01: filter_out <= 32'h7FFFFFFF; //Overflow
                2'b10: filter_out <= 32'h80000000; //Underflow
                default: filter_out <= filter_out_temp[31:0];
            endcase
        end
    end
endmodule
```

Figure A.6: Digital IIR integrator (Verilog code).

Recall from Section 3.2, the analog current sources are driven by a single 8-channel DAC. All of the channels are programmed via a standard SPI interface. The code snippet in Figure A.7 sheds light on the method for updating the DAC channels one at a time. Each DAC channel has its own input word to the Verilog module (**dacN[15:0]**) which maintains the desired DAC value. The module also maintains a local history of the last word written to the DAC for each channel (**last_wrN[15:0]**). When update requests are passed to this module, the updates are queued in a priority encoded fashion. This solution was created to avoid the necessity of a handshake protocol, which typically requires multiple cycles to set and release write requests. The new word is written to **shiftout** along with several static command bits. The state machine exits the idle state and transitions to the write state.

```

if(lockEn == 0) begin
  shift_count <= 5'd0;
  if(dac0[15:0] != last_wr0[15:0]) begin
    ss <= 1'b0;
    shiftout[23:0] <= {cmd_wnu,4'd0,dac0[15:0]};
    last_wr0[15:0] <= dac0[15:0];
    state <= spi_wr;
  end else if(dac1[15:0] != last_wr1[15:0]) begin
    ss <= 1'b0;
    shiftout[23:0] <= {cmd_wnu,4'd1,dac1[15:0]};
    last_wr1[15:0] <= dac1[15:0];
    state <= spi_wr;
  end else if(dac2[15:0] != last_wr2[15:0]) begin
    ss <= 1'b0;
    shiftout[23:0] <= {cmd_wnu,4'd2,dac2[15:0]};
    last_wr2[15:0] <= dac2[15:0];
    state <= spi_wr;
  end
end

```

Figure A.7: Priority encoded update requests to the DAC SPI (Verilog code).

One caveat to the update queue is detailed in Figure A.8. When the feedback loop is enabled (**lockEn == 1'b1**), all other DAC requests are ignored, since this channel should be updated as fast as possible and all other bias points should be fixed values for proper operation. The GUI corroborates this rule by freezing the bias knobs (unselectable) during lock. The **fbVal[15:0]** is wired to the module through a separate input

and overwrites the DAC channel 5 input.

```

end else if(dac7[15:0] != last_wr7[15:0]) begin
    ss <= 1'b0;
    shiftout[23:0] <= {cmd_wnu,4'd7,dac7[15:0]};
    last_wr7[15:0] <= dac7[15:0];
    state <= spi_wr;
end
end

if(lockEn == 1) begin
    shift_count <= 5'd0;
    if(fbVal[15:0] != fbValHist[15:0]) begin
        ss <= 1'b0;
        shiftout[23:0] <= {cmd_wnu, 4'd5, fbVal[15:0]}; //check channel
        fbValHist[15:0] <= fbVal[15:0];
        state <= spi_wr;
    end
end
end
end
end

```

Figure A.8: Feedback loop update requests to the DAC SPI (Verilog code).

```

spi_wr: begin
if((count == 2'd3) & ~sclk_en) begin
    sclk_en <= 1'b1;
end

if((count == 2'd1) & sclk_en) begin
    shift_count <= shift_count + 1;
    shiftout[23:0] <= {shiftout[22:0], 1'b0};
    if (shift_count == 5'd23) begin
        sclk_en <= 1'b0;
        ss <= 1'b1;
        state <= idle;
    end
end
end
endcase
end

assign sclk = sclk_en ? count[1] : 1'b1;
assign mosi = shiftout[23];

```

Figure A.9: Write state for the DAC SPI state machine (Verilog code).

Finally, the write state (**spi_wr**) of the state machine is shown. A 2-bit register, **count**, is utilized to divide the FPGA's clock by 4 while allowing operations to take place on rising and falling edges of the SPI's clock, **sclk**. The first if statement provides

the slave select (`ss`) a half cycle of going low before the clock's first edge, satisfying a timing requirement of the DAC's interface. The code is written such that `shiftout` updates the `mosi` output on `sclk` rising edges, since the operation takes place on the clock edge following `count == 2'd1`. The `shift_count` register keeps track of how many bits have been shifted out, and the state machine returns to idle once the write is complete.

The ADRF6820 heterodyne down-converter, discussed in Section 3.4, is programmed via its own SPI interface. There are numerous calibration registers that may or may not need to be updated, depending on the previous LO frequency and the newly requested LO frequency. All of the desired register values are stored in a local lookup table and passed to this SPI module in a single string (`wrData`). In order to speed up the overall update rate, the module checks each of these against the last written value to the register (`lastWr_N`), and only performs a SPI write if the value has changed.

```

if(resetCount != 5'd0) begin
    ss <= 1'b0;
    shiftout[23:0] <= resetValues[24*resetCount - 1 -: 24];
    resetCount <= resetCount - 1;
    state <= spi_wr;
    mosiEn <= 1'b1;
end else if(wrData[263:240] != lastWr_1[23:0]) begin // reg 11
    ss <= 1'b0;
    shiftout[23:0] <= {8'h62, wrData[255:240]};
    lastWr_1[23:0] <= wrData[263:240];
    state <= spi_wr;
mosiEn <= 1'b1;
end else if(wrData[287:264] != lastWr_2[23:0]) begin // reg 10
    ss <= 1'b0;
    shiftout[23:0] <= {8'h60, wrData[279:264]};
    lastWr_2[23:0] <= wrData[287:264];
    state <= spi_wr;
mosiEn <= 1'b1;
end else if(wrData[527:504] != lastWr_4[23:0]) begin // reg 0
    ss <= 1'b0;
    shiftout[23:0] <= {8'h00, wrData[519:504]};
    lastWr_4[23:0] <= wrData[527:504];
    state <= spi_wr;
    mosiEn <= 1'b1;
end else if(intChange & fracChange) begin // mod div
    ss <= 1'b0;
    shiftout[23:0] <= {8'h08, 16'h0005};

```

Figure A.10: State machine for the ADRF6820 heterodyne down-converter (Verilog code).

A.2 Processor C Code

The code snippet in Figure A.11 provides a glance at the state machine running on the ARM microprocessor. Introduced in Section 2.4, the purpose of this layer is to handle user requests from the GUI and translate them to the lower level RTL modules described above. The PC/SoC serial port communicates at a relatively slow 115,200 baud. With the digital feedback loop running at 100 MHz, it was not feasible for multiple registers to update in a sequential fashion for a single request. It is this very reason that the firmware was designed to gather all information from the GUI for a given request, and then update all necessary RTL registers through the high-speed AXI Memory Interface [29].

The state machine sits idle in **case ‘a’**. The function **getInput** waits for a UART transmission from the GUI/PC, and returns the received byte as an array, **choice**. The GUI will transmit a letter corresponding to which type of request it is initiating. If the letter is a valid option, the state machine transitions to the according state (**state = choice[0]**).

Case ‘e’ captures the request to update the phase comparator’s expected phase accumulation rate. Similar to the previous function, **getIndex** will wait for a UART transmission, up to 10 bytes, and convert the ASCII to a numerical value. The expected phase is written to the RTL through the pre-defined function **ADRF_SPI_mWriteReg**, with the assigned base address **XPAR_PHASECOMPARATOR_S00_AXI_BASEADDR**, and the byte offset of 4 indicating this word maps to **slv_reg1**.

```

while (1)
{
    switch(state)
    {
        case 'a' :
        {
            if (promptVal == 0)
            {
                promptVal = 1;
            }

            getInput(choice);
            if (choice[0] == 'd' || choice[0] == 't' || choice[0] == 'e' || choice[0] == 'i' ||
                choice[0] == 'l' || choice[0] == 'p' || choice[0] == 'f' || choice[0] == 'b' ||
                choice[0] == 'u')

                state = choice[0];
            else {
                state = 'a';
                promptVal = 0;
            }
            break;
        }

        case 'e':
        {
            //expected dph/dz
            getIndex(lutVal,10);
            ADRF_SPI_mWriteReg(XPAR_PHASECOMPARATOR_0_S00_AXI_BASEADDR, 4, *lutVal);
            xil_printf("The expected phase is 0x%08x \n\r", *(ba_pc+1));
            state = 'a';
            promptVal = 0;
            break;
        }

        case 'l' :
        {
            // LUT index
            getIndex(lutVal,3);
            ADRF_SPI_mWriteReg(XPAR_ADRF_SPI_0_S00_AXI_BASEADDR, 4, *lutVal);
            ADRF_SPI_mWriteReg(XPAR_PROGRAMMABLELOOPFILTER_0_S00_AXI_BASEADDR, 16, 0x00000001);

            xil_printf("Write: 0x%08x to look up table (rev5) \n\r", *(ba_adrf+1));

            state = 'a';
            promptVal = 0;
            break;
        }

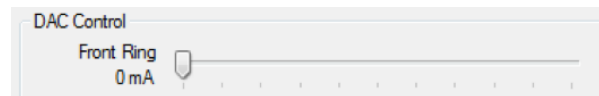
        case 'i': //case to test the integrator
        {
            getInput(choice);
            if(choice[0] == 'a'){
                ADRF_SPI_mWriteReg(XPAR_ARCTAN_0_S00_AXI_BASEADDR, 4, 0x00000101);
            }
        }
    }
}

```

Figure A.11: Snippet of the main while(1) loop on the ARM microprocessor (C code).

A.3 GUI VB Code

The trackbar for the Front Ring current (tied to DAC channel 0) is shown below. The code snippet **TrackBar18_Scroll** executes any time the slider is moved to a new value. SerialPort1 is an open COMM port that utilizes the USB to JTAG serial connection. The GUI/PC first sends the letter 'd' to inform the firmware of a the DAC update request. Once the firmware is in state 'd', the number '0' is sent to indicate this is an update to DAC channel 0. Finally, the update word is sent. The trackBar is initialized with a 0-65535 full scale range, and so the raw **TrackBar18.Value** is suited to the 16-bit DAC's full scale range. The serial port is only capable of sending strings and so the decimal value must be converted to ASCII and vice versa on the PC/firmware sides respectively. The TLDAC0.Text label provides a text readout of the current in milliamps, hiding the full-scale range and DAC voltage \rightarrow drive current equations from the end-user.



```
Private Sub TrackBar18_Scroll(sender As Object, e As EventArgs) Handles TrackBar18.Scroll
    SerialPort1.WriteLine("d")
    SerialPort1.WriteLine("0")
    SerialPort1.WriteLine(Convert.ToString(TrackBar18.Value))
    TLDAC0.Text = Convert.ToString(Math.Round((TrackBar18.Value / 65535 * 25), 2)) + " mA"
End Sub
```

Figure A.12: DAC tuning knob on the Graphical User Interface (top) and corresponding VB function (bottom).

Bibliography

- [1] G.-C. Hsieh and J. C. Hung, “Phase-locked loop techniques. a survey,” *IEEE Transactions on industrial electronics*, vol. 43, no. 6, pp. 609–615, 1996.
- [2] D. J. Jones, S. A. Diddams, J. K. Ranka, A. Stentz, R. S. Windeler, J. L. Hall, and S. T. Cundiff, “Carrier-envelope phase control of femtosecond mode-locked lasers and direct optical frequency synthesis,” *Science*, vol. 288, no. 5466, pp. 635–639, 2000.
- [3] T. Udem, R. Holzwarth, and T. W. Hänsch, “Optical frequency metrology,” *Nature*, vol. 416, no. 6877, p. 233, 2002.
- [4] S. A. Diddams, D. J. Jones, J. Ye, S. T. Cundiff, J. L. Hall, J. K. Ranka, R. S. Windeler, R. Holzwarth, T. Udem, and T. Hänsch, “Direct link between microwave and optical frequencies with a 300 thz femtosecond laser comb,” *Physical Review Letters*, vol. 84, no. 22, p. 5102, 2000.
- [5] [Online]. Available: <https://lasersdbw.larc.nasa.gov/files/2016/07/tutorials-lidar.jpg>
- [6] [Online]. Available: https://jila.colorado.edu/yelabs/sites/default/files/styles/image_700/public/images/publications/Comb%20dispersion.jpg
- [7] J. Ye, “Absolute measurement of a long, arbitrary distance to less than an optical fringe,” *Optics letters*, vol. 29, no. 10, pp. 1153–1155, 2004.
- [8] E. Baumann, F. R. Giorgetta, J.-D. Deschênes, W. C. Swann, I. Coddington, and N. R. Newbury, “Comb-calibrated laser ranging for three-dimensional surface profiling with micrometer-level precision at a distance,” *Optics express*, vol. 22, no. 21, pp. 24 914–24 928, 2014.
- [9] W. Rabinovich, C. Moore, R. Mahon, P. Goetz, H. Burris, M. Ferraro, J. Murphy, L. Thomas, G. Gilbreath, M. Vilcheck *et al.*, “Free-space optical communications research and demonstrations at the us naval research laboratory,” *Applied optics*, vol. 54, no. 31, pp. F189–F200, 2015.

- [10] H. G. Seif and X. Hu, “Autonomous driving in the icityhd maps as a key challenge of the automotive industry,” *Engineering*, vol. 2, no. 2, pp. 159–162, 2016.
- [11] A. A. Oloufa, M. Ikeda, and H. Oda, “Situational awareness of construction equipment using gps, wireless and web technologies,” *Automation in Construction*, vol. 12, no. 6, pp. 737–748, 2003.
- [12] P. DelHaye, A. Schliesser, O. Arcizet, T. Wilken, R. Holzwarth, and T. J. Kippenberg, “Optical frequency comb generation from a monolithic microresonator,” *Nature*, vol. 450, no. 7173, p. 1214, 2007.
- [13] T. J. Kippenberg, R. Holzwarth, and S. A. Diddams, “Microresonator-based optical frequency combs,” *science*, vol. 332, no. 6029, pp. 555–559, 2011.
- [14] M. A. Foster, J. S. Levy, O. Kuzucu, K. Saha, M. Lipson, and A. L. Gaeta, “Silicon-based monolithic optical frequency comb source,” *Optics Express*, vol. 19, no. 15, pp. 14 233–14 239, 2011.
- [15] S. Arafin, A. Simsek, S.-K. Kim, S. Dwivedi, W. Liang, D. Eliyahu, J. Klamkin, A. Matsko, L. Johansson, L. Maleki *et al.*, “Towards chip-scale optical frequency synthesis based on optical heterodyne phase-locked loop,” *Optics express*, vol. 25, no. 2, pp. 681–695, 2017.
- [16] J. D. Jost, T. Herr, C. Lecaplain, V. Brasch, M. H. Pfeiffer, and T. J. Kippenberg, “Counting the cycles of light using a self-referenced optical microresonator,” *Optica*, vol. 2, no. 8, pp. 706–711, 2015.
- [17] P. Del’Haye, A. Coillet, T. Fortier, K. Beha, D. C. Cole, K. Y. Yang, H. Lee, K. J. Vahala, S. B. Papp, and S. A. Diddams, “Phase-coherent microwave-to-optical link with a self-referenced microcomb,” *Nature Photonics*, vol. 10, no. 8, p. 516, 2016.
- [18] V. Brasch, E. Lucas, J. D. Jost, M. Geiselmann, and T. J. Kippenberg, “Self-referenced photonic chip soliton kerr frequency comb,” *Light: Science & Applications*, vol. 6, no. 1, p. e16202, 2017.
- [19] E. S. Lamb, D. R. Carlson, D. D. Hickstein, J. R. Stone, S. A. Diddams, and S. B. Papp, “Optical-frequency measurements with a kerr microcomb and photonic-chip supercontinuum,” *Physical Review Applied*, vol. 9, no. 2, p. 024030, 2018.
- [20] S. Saito, Y. Yamamoto, and T. Kimura, “Optical heterodyne detection of directly frequency modulated semiconductor laser signals,” *Electronics Letters*, vol. 16, no. 22, pp. 826–827, 1980.
- [21] D. Spencer, T. Drake, T. Briles, J. Stone, L. Sinclair, C. Fredrick, Q. Li, D. Westly, B. Ilic, A. Bluestone *et al.*, “An optical-frequency synthesizer using integrated photonics.” *Nature*, vol. 557, no. 7703, pp. 81–85, 2018.

- [22] D. T. Spencer, A. Bluestone, J. E. Bowers, T. C. Briles, S. A. Diddams, T. Drake, R. Ilic, T. J. Kippenberg, T. Komljenovic, S. H. Lee *et al.*, “Towards an integrated-photonics optical-frequency synthesizer with 1 hz residual frequency noise,” in *2017 Optical Fiber Communications Conference and Exhibition (OFC)*. Ieee, 2017, pp. 1–3.
- [23] J. Bowers, A. Beling, D. Blumenthal, A. Bluestone, S. Bowers, T. Briles, L. Chang, S. Diddams, G. Fish, H. Guo *et al.*, “Chip-scale optical resonator enabled synthesizer (cores) miniature systems for optical frequency synthesis,” in *2016 IEEE International Frequency Control Symposium (IFCS)*. IEEE, 2016, pp. 1–5.
- [24] L. Chang, Y. Li, N. Volet, L. Wang, J. Peters, and J. E. Bowers, “Thin film wavelength converters for photonic integrated circuits,” *Optica*, vol. 3, no. 5, pp. 531–535, 2016.
- [25] M. Darvishi, R. van der Zee, and B. Nauta, “Design of active n-path filters,” *IEEE journal of solid-state circuits*, vol. 48, no. 12, pp. 2962–2976, 2013.
- [26] M. L. Davenport, S. Skendžić, N. Volet, J. C. Hulme, M. J. Heck, and J. E. Bowers, “Heterogeneous silicon/iii–v semiconductor optical amplifiers,” *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 22, no. 6, pp. 78–88, 2016.
- [27] J. Hulme, J. Doylend, and J. Bowers, “Widely tunable vernier ring laser on hybrid silicon,” *Optics express*, vol. 21, no. 17, pp. 19 718–19 722, 2013.
- [28] T. Komljenovic, S. Srinivasan, E. Norberg, M. Davenport, G. Fish, and J. E. Bowers, “Widely tunable narrow-linewidth monolithically integrated external-cavity semiconductor lasers,” *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 21, no. 6, pp. 214–222, 2015.
- [29] “Axi reference guide,” *Xilinx, UG761*, v13.1, 2011.
- [30] R. Dorf and R. Bishop, *Modern Control Systems*. Prentice Hall New Jersey, 2001, vol. 9.
- [31] M. Sullivan, *Fundamentals of Statistics*. Pearson Education New Jersey, 2008.
- [32] A. Bluestone, A. Jain, N. Volet, D. T. Spencer, S. B. Papp, S. A. Diddams, J. E. Bowers, and L. Theogarajan, “Heterodyne-based hybrid controller for wide dynamic range optoelectronic frequency synthesis,” *Optics Express*, vol. 25, no. 23, pp. 29 086–29 097, 2017.
- [33] “Adrf6820: 695 mhz to 2700 mhz, quadrature demodulator with integrated fractional-n pll and vco,” *Analog Devices*, Rev. C, 2016.
- [34] J. E. Volder, “The cordic trigonometric computing technique,” *IRE Transactions on electronic computers*, no. 3, pp. 330–334, 1959.

- [35] R. A. Pease *et al.*, “A comprehensive study of the howland current pump,” *National Semiconductor. January*, vol. 29, 2008.
- [36] K. Uesaka, E. Banno, H. Shoji, H. Matsuura, H. Kuwatsuka, K. Tanizawa, and S. Namiki, “Laser apparatus and method to re-tune emission wavelength tunable ld,” Sep. 13 2016, uS Patent 9,444,221.
- [37] R. Costanzo and S. M. Bowers, “A current reuse regulated cascode cmos transimpedance amplifier with 11-ghz bandwidth,” *IEEE Microwave and Wireless Components Letters*, vol. 28, no. 9, pp. 816–818, 2018.
- [38] J. Jalil, M. B. I. Reaz, and M. A. M. Ali, “Cmos differential ring oscillators: Review of the performance of cmos ros in communication systems,” *IEEE microwave magazine*, vol. 14, no. 5, pp. 97–109, 2013.
- [39] B. Razavi, *RF Microelectronics*. Prentice Hall New Jersey, 1998, vol. 2.
- [40] P. Larsson, “High-speed architecture for a programmable frequency divider and a dual-modulus prescaler,” *IEEE Journal of Solid-State Circuits*, vol. 31, no. 5, pp. 744–748, 1996.
- [41] G. Han and E. Sanchez-Sinencio, “Cmos transconductance multipliers: A tutorial,” *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 45, no. 12, pp. 1550–1563, 1998.
- [42] B. A. Chappell, T. I. Chappell, S. E. Schuster, H. M. Segmuller, J. W. Allan, R. L. Franch, and P. J. Restle, “Fast cmos ecl receivers with 100-mv worst-case sensitivity,” *IEEE Journal of Solid-State Circuits*, vol. 23, no. 1, pp. 59–67, 1988.
- [43] J.-F. Genat and F. Rossel, “Ultra high-speed time-to-digital converter,” Jan. 12 1988, uS Patent 4,719,608.
- [44] P. Dudek, S. Szczepanski, and J. V. Hatfield, “A high-resolution cmos time-to-digital converter utilizing a vernier delay line,” *IEEE Journal of Solid-State Circuits*, vol. 35, no. 2, pp. 240–247, 2000.
- [45] J. Yu, F. F. Dai, and R. C. Jaeger, “A 12-bit vernier ring time-to-digital converter in 0.13 μm cmos technology,” *IEEE journal of solid-state circuits*, vol. 45, no. 4, pp. 830–842, 2010.
- [46] L. S. Theogarajan, “A low-power fully implantable 15-channel retinal stimulator chip,” *IEEE Journal of Solid-State Circuits*, vol. 43, no. 10, pp. 2322–2337, 2008.
- [47] J. Doernberg, P. R. Gray, and D. A. Hodges, “A 10-bit 5-msample/s cmos two-step flash adc,” *IEEE Journal of Solid-State Circuits*, vol. 24, no. 2, pp. 241–249, 1989.

- [48] B. Razavi, “The strongarm latch [a circuit for all seasons],” *IEEE Solid-State Circuits Magazine*, vol. 7, no. 2, pp. 12–17, 2015.
- [49] H. R. Rategh and T. H. Lee, “Superharmonic injection-locked frequency dividers,” *IEEE Journal of Solid-State Circuits*, vol. 34, no. 6, pp. 813–821, 1999.
- [50] M. Acar, D. Leenaerts, and B. Nauta, “A wide-band cmos injection-locked frequency divider,” in *2004 IEEE Radio Frequency Integrated Circuits (RFIC) Systems. Digest of Papers*. IEEE, 2004, pp. 211–214.
- [51] R. W. Rhea, *Discrete oscillator design: linear, nonlinear, transient, and noise domains*. Artech House, 2010.
- [52] M. Farazian, P. S. Gudem, and L. E. Larson, “Stability and operation of injection-locked regenerative frequency dividers,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 8, pp. 2006–2019, 2010.
- [53] F. Leens, “An introduction to i²c and spi protocols,” *IEEE Instrumentation & Measurement Magazine*, vol. 12, no. 1, pp. 8–13, 2009.
- [54] “2.5v / 3.3v dual channel programmable clock/data delay with differential cml outputs,” *ON Semiconductor*, Rev. 5, 2012.