FPGA IMPLEMENTATION OF LOW LATENCY AND HIGH SFDR CORDIC BASED DIRECT DIGITAL SYNTHESIZER

A Thesis

by

Nur Fajar Rizqi Annafianto

Submitted to the
Graduate School of Sciences and Engineering
In Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in the
Department of Electrical and Electronics Engineering

Özyeğin University
August 2020

FPGA IMPLEMENTATION OF LOW LATENCY AND HIGH SFDR CORDIC BASED DIRECT DIGITAL SYNTHESIZER

Advisor: Prof. H. Fatih UĞURDAĞ

Co-Advisor: Assoc. Prof. Sergey POLYAKOV

Approved by:

Prof. H. Fatih Uğurdağ, Advisor
Department of Electrical and Electronics Engineering
Özyeğin University

Assistant Prof. Göktürk Poyrazoğlu
Department of Electrical and Electronics Engineering
Özyeğin University

Assistant Prof. Vecdi Emre Levent
Dept. of Computer Science
Fenerbahçe University

Date Approved: 25 August 2020

To my family, friends, and teachers

ABSTRACT

In most modulation schemes for a telecommunication system, a fast and efficient sinusoidal signal generator is needed. In this thesis, we report on an FPGA implementation of a versatile Coordinate Rotation Digital Computer (CORDIC) based Direct Digital Synthesizer (DDS). The rapid expansion of the Internet requires significant improvements in energy and bandwidth efficiency. Therefore, a new class of communication systems, namely, quantum-measurement enhanced optical communication systems are being actively pursued. In such systems, a classical receiver is replaced with a quantum receiver, while transmitters are classical. Digital synthesis of these signals requires a versatile DDS, the development of which is reported in this thesis. By design, the DDS can generate signals with a nearly arbitrary combination of phase and frequency modulations. Many other applications such as software-defined radio, wireless satellite transceiver, etc. can take advantage of high spectral purity, low latency, and reconfigurable sine wave generation DDS. A CORDIC technique offers low complexity and memory-free trigonometric calculation approach. The pipeline structure maximizes the modulation capabilities. In the quantum-enhanced communication system, the alphabet length could be quite large (up to 16 in our implementation). Since some components of such communication system use large amounts of FPGA resources, it is critical that we come up with a low-area DDS. In regards to that, our design's FPGA resource usage consumes as low as the state of the art in the literature, while we achieve 64% latency reduction compared to that of the conventional CORDIC and maintain a 72 dB SFDR value.

ÖZETÇE

Bir telekomünikasyon sistemi için çoğu modülasyon şemasında, hızlı ve verimli bir sinüzoidal sinyal oluşturucuya ihtiyaç vardır. Bu tezde, çok yönlü bir Koordinat Rotasyonlu Dijital Bilgisayar (CORDIC) tabanlı Doğrudan Dijital Sentez'leyicinin (DDS) FPGA uygulamasını raporluyoruz. Internet'in hızla genişlemesi, enerji ve bant genişliği verimliliğinde önemli gelişmeler gerektirir. Bu nedenle, yeni bir iletişim sistemleri sınıfı, yani kuantum ölçümlü optik iletişim sistemleri aktif olarak takip edilmektedir. Bu tür sistemlerde, klasik bir alıcının yerini bir kuantum alıcısı alırken, vericiler klasiktir. Bu sinyallerin dijital sentezi, gelişimi bu tezde bildirilen çok yönlü bir DDS gerektirir. Tasarım gereği DDS, neredeyse keyfi bir faz ve frekans modülasyon kombinasyonu ile sinyaller üretebilir. Yazılım tanımlı radyo, kablosuz uydu alıcı vericisi, vb. diğer birçok uygulama, yüksek spektral saflık, düşük gecikme süresi ve yeniden yapılandırılabilir sinüs dalgası oluşturma DDS'den yararlanabilir. Bir CORDIC tekniği, düşük karmaşıklık ve hafızasız trigonometrik hesaplama yaklaşımı sunar. Boru hattı yapısı modülasyon yeteneklerini en üst düzeye çıkarır. Kuantumla geliştirilmiş iletişim sisteminde, alfabe uzunluğu oldukça büyük olabilir (uygulamamızda 16'ya kadar). Bu tür bir iletişim sisteminin bazı bileşenleri büyük miktarda FPGA kaynağı kullandığından, düşük alanlı bir DDS bulmamız çok önemlidir. Bununla ilgili olarak, tasarımın FPGA kaynak kullanımı literatürdeki en son teknoloji kadar düşük tüketirken, geleneksel CORDIC'e kıyasla %64 gecikme azaltımı elde ediyor ve 72 dB SFDR değerini koruyoruz.

ACKNOWLEDGEMENTS

Special thanks to my thesis advisor Prof. H. Fatih Uğurdağ. I would like to equally thank to my co-advisor Assoc. Prof. Sergey Polyakov. I am indebted for all their precious assists, support, and advice. Additionally, I am grateful to Assistant Prof. Göktürk Poyrazoğlu and Assistant Prof. Vecdi Emre Levent for attending my thesis defense presentation as examination committee members. Moreover, my thanks to my lab mates at Özyeğin University and NIST, who have always supported me and had significant contributions to my research work and studies.

Initially, I was a full-time teaching assistant at Özyeğin University from Feb. 2018 until Feb. 2019. Then, I became a research assistant there, from Feb. 2019 until Aug. 2019, and I was supported by a TÜBİTAK project (no. 215E311) under the supervision of Prof. Murat Uysal. During that time, I was engaged in an optical communication project at OKATEM, a research center at Özyeğin University. In that project, my project mates and I implemented an OFDM algorithm on FPGA. I appreciate the opportunity of being involved in that project, because a great amount of my research skills were obtained there. For that reason, I am thankful to Vecdi Emre Levent, Burak Kebapçı, Furkan Aydın, and Shewit Tesfay, as well as Prof. Murat Uysal and Prof. H. Fatih Uğurdağ.

Since Aug. 2019, I have been involved in a novel communication project at NIST, Maryland, USA. This work is partially supported by NSF (project no. ECCS 1927674). I am so grateful to Assoc. Prof. Sergey Polyakov, who allowed me to be a part of the quantum communication project at NIST, which is the base-ground of this thesis. Finally, I would like to offer my greatest thanks to my family for the support and motivation they provided.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

GLOSSARY

AOM Acousto-Optical Modulator, p. 4.

CFSK Coherent Frequency Shift Keying, p. 4.

CORDIC Coordinate Rotation Digital Computer, p. 1.

DAC Digital-to-Analog Converter, p. 7.

DDS Direct Digital Synthesizer, p. 1.

FCW Frequency Control Word, p. 7.

FPGA Field Programmable Gate Array, p. 2.

FSK Frequency Shift Keying, p. 2.

HB Helstrom Bound, p. 6.

HFPSK Hybrid Frequency-Phase Shift Keying, p. 4.

ILA Integrated Logic Analyzer, p. 31.

LO Local Oscillator, p. 4.

LUTs Look-Up Tables, p. 11.

PSK Phase Shift Keying, p. 2.

ROM Read-Only Memory, p. 11.

RTL Register Transfer Level, p. 31.

SFDR Spurious Free Dynamic Range, p. 8.

SM State Machine, p. 39.

SNL Shot Noise Level, p. 5.

SPD Single-Photon Detector, p. 4.

SQL Standard Quantum Limit, p. 5.

CHAPTER I

INTRODUCTION

This chapter introduces the problem and gives an overview of the Quantum-enhanced Communication Protocol and Direct Digital Synthesizer (DDS). Additionally, Coordinate Rotation Digital Computer (CORDIC) concept, contributions of the Thesis and outline of the Thesis are also presented.

1.1 Problem Statement

The demands of high throughput communication channel have always been increasing over time as the continuous development of novel applications accelerates. Communication becomes the medium of two or more parties to share information. The form of communication channel varies with the available resources and the opportunities that technologies and science have offered. Communication has already been the agent of information flow in human history. With the provided abilities many communication protocols and strategies have been invented. Additionally, applications are also developed on top of the provided work-frame. During recent decades, the Internet has attracted significant attention from industry and academia. The rapid flow of data and a wide range of accessibility made it convenient for the global citizen to reach one another and to share information. As the number of applications and customers increases rapidly, the requirement of throughput also elevates significantly. Hence new communication protocol that leverages the quantum's properties has been developed, it has been proven to beat the performance of classical communication protocol. However, the other hardware components might not be able to optimize the utilization of quantum instruments because of their physical constraints. Hence, the best option to optimize the utilization is to use Field Programmable Gate Array

(FPGA). With the parallelization and the high freedom of programmability, FPGA enhances the optimum integration with the quantum instruments. However, the correct choice of architecture, design techniques, and algorithmic strategies should be done properly.

Majority of communication protocol requires the sinusoidal wave generator, including quantum-enhanced communication protocol. The most popular approach of sine wave generation in FPGA is to use a DDS. It provides a convenient and arbitrary sinusoidal wave generation, given that immediate transition of the output signal to any desired frequency can be done. DDS implementation can use one of several strategies that will be explained later on. According to the requirements of the quantum-enhanced communication protocol, the CORDIC method is chosen. CORDIC minimizes memory usage, but it comes with several drawbacks and the most important one for our purpose is the high latency, but still needs to attain high spectral purity to maximize the efficiency of quantum measurement instruments.

1.2 Quantum-enhanced Communication

Most commercial lightwave communication systems use standard modulation protocols, such as Phase-Shift Keying (PSK) and Frequency-Shift Keying (FSK), whose implementation is supported by specialized dedicated hardware, the rapid expansion of the Internet requires significant improvement in energy and bandwidth efficiency. Therefore, a new class of communication systems, namely, quantum-measurement enhanced optical communication systems are being actively pursued. In those systems, a classical receiver is replaced with a quantum receiver, while the transmitter remains similar. Recognizing that properties of quantum measurement are in general different from that of a classical measurement, more complex modulation schemes than PSK and FSK turns out to be more beneficial [2].

With the larger area the network cover and due to the demands of industry and

commercial usage, the Internet has been an attractive area for many researchers to develop a more efficient communication system. Due to the increasing growth of network usage, the load on the physical layer of the network has also increased and requires improvements and modifications. These developments can be in the level of algorithm, protocol, or hardware instrumentation. Such communication systems are usually evaluated in terms of bandwidth and energy/power. Hence, optical data transmission offers efficiency and opportunity for the communication system's improvement. Because optical data transmission provides lower attenuation and allows broad occupation of bandwidths [2]. This also comes with a considerable drawback in its development, such that pulses have been used for bits transmission that bears the information inside. However, that pulses usually have a certain energy and resides in particular bandwidth for reliable transmission [2]. Through the history of research in optical communication, classical measurement protocols have always been considered and explored. However, the current state of research on quantum measurement results on a more efficient measurement opportunity compared to the capabilities that classical measurement can offer. Hence, to observe and to evaluate the effects on the communication system quantum measurement instruments have been utilized and experiments have been conducted, but to make fair comparison similar classical communication protocol has been adopted. It has been proven that by using a quantum communication protocol, we can beat and surpass the performance of the rest of classical measurement versions. Hence, it acquires the highest energy efficiency while in the condition of great signal loss that makes the optical signal becomes faint[2]. Additionally, the quantum communication protocol is also exposed to significant loss that can be observed and found in the implementation of other communication channels. The quantum communication protocol that utilizes the quantum measurement instruments possesses different realizations in the receiver part and for transmitter adopts similar to that of other classical communication protocols. However, the development

still encounters several limitations and it is open to research and exploration.

To make a fair comparison and the experiment can be fitted to the other modulation schemes, a coherent state modulation scheme with phase and frequency keying is adopted. This way, the superiority of quantum measurement is analyzed and optimized by taking into consideration of the properties of quantum measurement and also the particular properties of coherent states when it is implemented with quantum measurement. As mentioned above, the communication system can extract information embedded in the optical signal even in the faint version of the signal. Thanks to the photon detection property of the quantum measurement instrument that provides information of the signal based on the arrival time of the pulse, and this gives dominancy to the system over other receivers explored till now [2]. The classical communication protocol has inherent limitation and with the help of quantum measurement, the communication system can surpass that limitation which is analyzed based on the communication alphabet length. Particularly in the larger alphabets which is significant for realization and application in real-life practice.

Practically, with the advantage given by double Acousto-Optical Modulator (AOM), this quantum communication protocol can be implemented for PSK, Coherent Frequency Shift Keying (CFSK), and Hybrid Frequency-Phase Shift Keying (HFPSK). To meet the rapid response and real-time requirements, FPGA comes to aid and it's also responsible for the discrimination algorithm and radio-frequency synthesis. For experimental purposes, the two radio-frequency signals synthesized by FPGA are fed into two identical AOMs and passed to Local Oscillators (LO). By performing quantum displacement and taking the signal interference between the test signal and discrimination signal from LOs, the feedback signal is fed to beamsplitter and eventually arrives at Single-Photon Detector (SPD). The arrival time of feedback output pulse signal from SPD bears information of the state of input signal and hence FPGA

performs discrimination to predict the current input state. With this setup the quantum communication protocol achieves discrimination error lower than that of classical communication protocol's Shot Noise Level (SNL)[3]

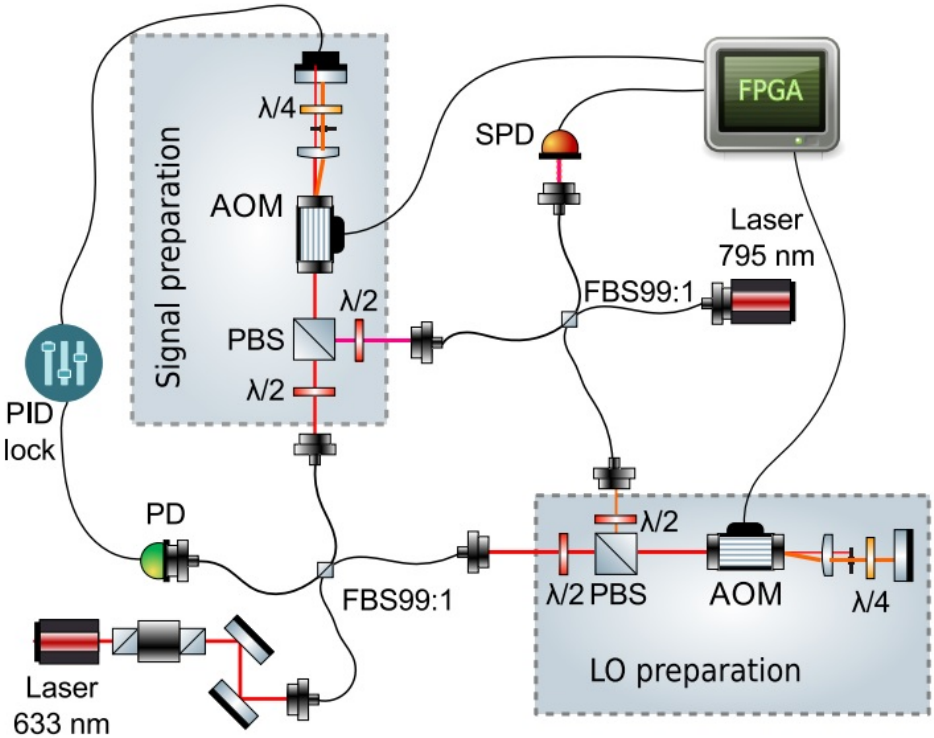

Figure 1: Quantum-enhanced communication setup.

In the optical experiment, all M states are prepared in parallel at all times, and the active output state is picked according to the encoding and measurement protocols.

To make a quantum measurement enhanced transceiver, we choose the modulation scheme which includes choosing the number of states M, the frequency, and the initial phase detuning between the adjacent states and other communication parameters.

In communication links, sensitivity is often measured as the probability to receive an erroneous symbol with certain energy at the receiver. Classical receivers have a sensitivity limit known as the Standard Quantum Limit (SQL). This limit arises from the inevitable shot noise on the idealized classical receiver scheme - a

homodyne measurement followed by a perfect detector with no noise of its own and with the 100% detection-efficiency. The SQL, however, is significantly above the so-called Helstrom Bound (HB) - or the fundamental sensitivity bound - accessible only through quantum measurement. With the help of the described DDS, we have implemented a quantum-measurement telecommunication testbed and demonstrated that the sensitivity of a telecommunication channel is better than SQL for many different modulation protocols, including quantum-measurement specific modulation protocols, described elsewhere [4].

A specific target application for this DDS is to build a quantum measurement enhanced transceiver. Some unconventional modulation schemes have been proposed in the literature, that are not compatible with the state of the art commercial hardware. Particularly, we intend to use modulation schemes that require a simultaneous phase and frequency modulation. In our optical experiment, all M states are prepared in parallel at all times, and the active output state is picked according to the encoding and measurement protocols. Because M could be quite large (up to 16 in our implementation) the low-resource usage DDSs are essential[5]. Our novel design achieves the shortest latencies, maximizes modulation capabilities, and uses a minimal footprint compared to other CORDIC-based DDSs.

Digital synthesis of these signals requires versatile DDS whose development is reported here. By design, it generates signals with a nearly-arbitrary combination of phase and frequency modulations. Many other applications such as software-defined radio, a wireless satellite transceiver, HDTV transmission, radar communication, etc. can take advantage of this low latency and re-configurable sine wave generation.

1.3 DDS Concept

The demand for high-throughput communication has always existed since its first invention. Various applications have been developed with the opportunity that current communication technology has given. The need for the Internet increases over the years and for that reason, many communication systems have been explored in the level of algorithm and architecture. Many communication systems use a modulation scheme that generates sinusoidal signal output. One popular approach is to use the DDS module that takes the Frequency Control Word (FCW) as input and passes the amplitude of the sinusoidal signal to the output. Fig. 2 shows the general block diagram of DDS which consists of a phase accumulator, a signal processor, a Digital-to-Analog Converter (DAC) and a low pass filter.

The phase accumulator defines the frequency of the sinusoidal signal as the increment of the output phase is dictated by the input phase step or FCW[6], hence the smaller the input the lower the resulting signal's frequency and vice versa. The generated sinusoidal wave's frequency is calculated as $f_{out} = \frac{FCW}{2^{bit-width}} * f_{system}$.

The range of DDS's output value covers the whole range of the DDS's input value [7]. The higher the bit width of input value the larger possible combination of output values. Additionally, since the mapping covers all the value of the input, there would be rounding-up in the consecutive output value, particularly for sinusoidal wave generation where we have continuous output values for adjacent inputs. So the output of the first input's value is close to the output of the last input's value to ensure continuity. However, in the digital domain, we cannot obtain a completely continuous signal wave due to the quantization of the digital variables. The quantization error can be reduced by increasing the bit width of the output, but it will increase resource utilization[8]. Hence, the proper bit-width selection is necessary and for commercial DAC, the standard bit width for input and output pins are 16 bits [9]. However, since the frequency resolution and the combination of output is also limited to the input

value's bit width. If we have small bit width for the output but high bit width for the input, then it results in the redundancy of combination of output value. Because there would be several identical values for adjacent points where it loses the resolution and sensitivity of the output signal for those corresponding points. Hence, the frequency of the output signal is determined by the FCW that is defined by the input of the phase accumulator, and also the system clock.

The low pass filter rectifies the signal processor's output that contains trivial distortions due to the techniques being employed. It ensures better spectral purity from the other module which is typically DAC module because of the errors given by quantization and the algorithm or strategy being employed. The low pass filter is typically comes after DAC such some works in the literature show them[10].

Here, we focus on the signal processor that takes phase as the input and generates sinusoidal amplitude.

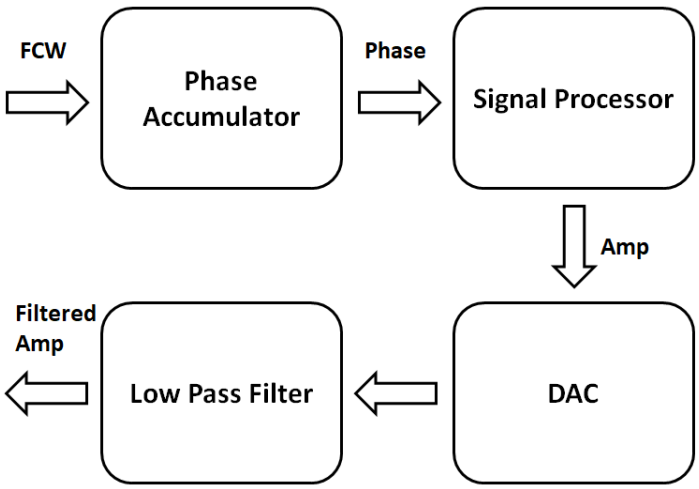

Figure 2: General block diagram of DDS.

SFDR

Spurious Free Dynamic Range (SFDR) defines the spectral purity of the produced signal. The spectral purity of a signal is significant for the overall performance of the

system. Quantum measurement instruments offer significantly more efficient properties compared to the classical measurement ones. In terms of feedback latency, which is important for FPGA to utilize the parallel computing and real-time application properties, the quantum-enhanced measurement tools are lower and can be used to increase the bandwidth of the communication system while having faint optical signal and increasing the communication alphabet number. To optimize the benefit of those properties high spectral purity is essential because introducing error to the instrument reduces the optimization of overall performance. Since there exists more than one frequency component in a signal, it is necessary to keep the desired frequency's dominance over the spurious components. SFDR implies the ratio of the power of the desired signal and the power of the strongest spurious signal[11]. Thus, the higher SFDR the smoother the output obtained, which is preferred and pursued in this work.

In Fig. 3, we depict the SFDR of a signal that has 79.52 dB value. The target signal's power is the one showed as fundamental in the figure, which is desired to be the highest. The spur signals are the embedded undesired signal that should have power as low as possible compared to the target signal's power. Hence, the larger the SFDR the smoother the output signal, the lower the effect on the error and higher optimization for the quantum measurement instruments.

1.4 CORDIC Concept

CORDIC is an algorithm that mainly used to compute trigonometric function. Moreover, by modification in the algorithmic level, it can be utilized to compute hyperbolic function, logarithmic and exponential problems. The algorithm computes the results by utilizing angular rotation that converges to the final desired value in the base of iterations. For the sake of hardware compatible implementation, modification in the algorithm's formula is performed so that it can be convenient in the realization of hardware design, such that multiplication and/or division are carried out with

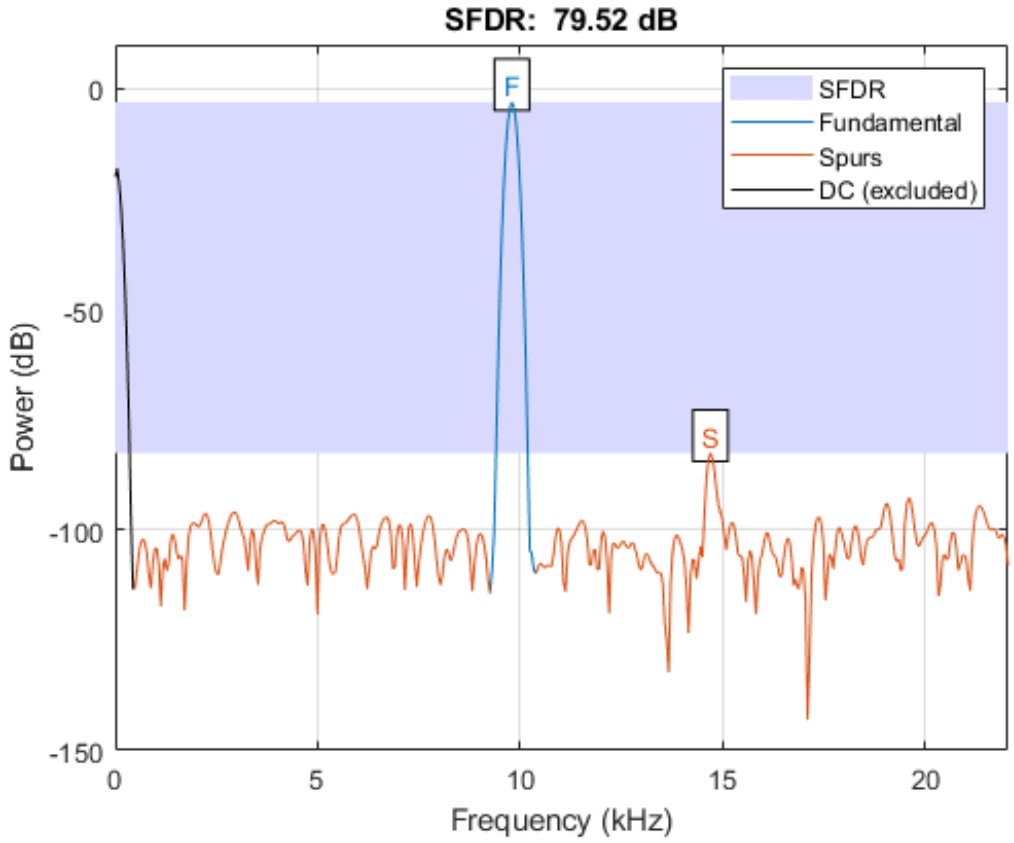

Figure 3: SFDR of a signal with value of 79.52 dB.

bit-level shifter to imitate the operation, that is so-called pseudo-operator. This way expensive resource usage of multiplier/divider can be avoided and the implementation ends up with only adder and logic shifter.

Many strategies and techniques have been developed to enhance the hardware area and speed efficiency of CORDIC algorithm implementations. The developments and research are explored in the level of algorithmic modifications and also hardware implementation technique that improves timing and area utilization's efficiency. CORDIC was initially introduced by Jack E. Volder in 1959 to calculate trigonometric functions in digital hardware devices. Thus, the CORDIC algorithm is also known as the Volder algorithm. Later, a modified version of the CORDIC algorithm was proposed by John S. Walther with the ability to calculate circular, hyperbolic, and linear

rotation systems [1]. The motivation to apply this algorithm in digital platforms has gained popularity since then[12]. Refinements on the efficiency of implementation have resulted in reduced latency and hardware area usage.

CORDIC provides an efficient hardware implementation in terms of area utilization, power consumption, and latency. To see the advantage that the CORDIC algorithm provides, we can compare the other approaches to the implementation of DDS.

There are three popular approaches to the realization of DDS: Look-Up Tables (LUTs), polynomial function (namely, Tylor series) expansion, and a CORDIC algorithm [13]. The LUTs occupy memory, namely, Read-Only Memory (ROM), to store the amplitude of the sinusoidal signal. LUTs are computationally fast, but they require a considerable amount of memory even when compression techniques are used [14]. Indeed, to get higher spectral purity, the minimized quantization error and high frequency resolution, it is necessary to increase the LUT memory widths of both the input phase and the output amplitude to yield higher precision. For these reasons, memory size grows significantly with the greater spectral purity requirement. In turn, more memory results in higher power consumption, slower operation, and lower stability [15]. Even though the advantage of having fast computation, it degrades in terms of memory usage and that also affects the undesired lower maximum frequency of overall performance and stability issue. In the quantum-enhanced communication protocol, since it already requires a considerable amount of memory to store the probability values, additional occupation in memory is not desirable and preferably avoided. In fact, that also degrades the area and energy efficiencies on the overall hardware implementation setup.

The Taylor series expansion, also called a Maclaurin series, has a complicated implementation that uses several multipliers/dividers. Therefore, in terms of maximum frequency and area utilization, it has already at a disadvantage compared to other

approaches. In addition, to get higher spectral purity, higher-order terms should be computed which also means longer latency [13]. As the order increases the bothersome fractional numbers, differential, and power of numbers terms also increase. In Eq. 1 it shows that the higher the order the higher the fractional number which also means more multiplications and that degrades the maximum frequency significantly given that each order is computed in a single clock cycle. The same argument is applied to the differential and power of numbers. To improve efficiency, some argument in the equation can be stored in memory since the computation is too expensive and as a matter of trade-off of timing requirement and latency, memory usage is much preferable.

$$f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f^3(a)}{3!}(x-a)^3 + ... \quad (1)$$

A CORDIC algorithm calculates sinusoidal amplitude by a set of rotations. The rotational angles in the set are processed in series and the accumulation of the angles approximates the desired angle that corresponds to the necessary output. The number of angles in the set determines the number of iterations, hence the latency. Thus, the correct selection of an angle set is essential. The rotational operation is carried out by adders, logic shifters, and optionally a small amount of memory that makes implementation and integration easier and simpler [16]. For these reasons, CORDIC advances in resource utilization and power consumption.

DDS performs the CORDIC algorithm based on rotation mode. In general, CORDIC has two functional modes: a vector mode and a rotation mode. Our DDS algorithm is based on rotation mode. In rotation mode, the initial vector experiences several rotations in cartesian coordinates based on the angle set and reaches the desired vector position that corresponds to the destination phase with corresponding power gain [17]. Fig. 4 shows the rotation mode with two phases in the set. In the vector mode, the target angle is estimated by using a set of designated vectors

as the reversal of the rotation mode [13]. However, CORDIC also comes with some drawbacks. First, it needs scale factor compensation due to numerical operations in the algorithm. Usually, the final result is achieved by multiplying the scale factor with the output of the series of rotation. To eliminate this requirement, the initial vector is arranged such that it has already been regulated with the scale factor prior to the calculations. Secondly, accuracy restriction: the number of rotations and the selection of the angle set impacts how close the final angle is to the desired angle [18]. A smaller angle set is beneficial for latency result. In the later section, we describe components in each stage of the hardware architecture and the mathematical operations that they perform.

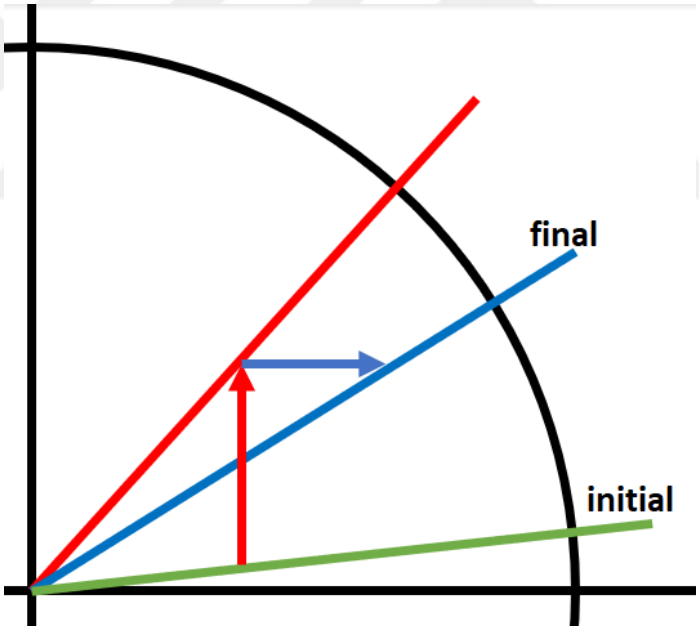

Figure 4: Rotation mode of CORDIC in DDS with two angular steps.

Since the rotations give the step of angular movement, the smaller the rotational angle the better the accuracy. However, to reach high convergence, having tiny angles would result in the increment of the number of angles in the rotational set, which implies latency. Additionally, a minute angle also makes rigid convergence to the desired vector for the algorithm.

1.5 Contributions of the Thesis

In this thesis, a new design of Direct Digital Synthesizer has been offered. Sinusoidal wave generation is a significant component of many engineering applications. Especially in the communication system, modulation schemes are often being employed. Most of the modulation schemes require a sinusoidal signal to carry out the information. For instance, the PSK communication scheme bears information in the phase of the signal and for the FSK communication scheme, the frequency embeds data. Those classical modulation schemes necessitate sinusoidal wave generation with the corresponding phase or frequency to induce the information to the signal. Therefore, DDS has been the preferred technique to meet this requirement since it offers simple and efficient applications.

The implementation of DDS to this date is based on the algorithm and approach being embraced. However, the selection of algorithms and techniques gives an impact to the performance, functionality, and also the implementation's efficiency. As described previously, the CORDIC algorithm has been proven to be area-efficient compared to the other approaches. Even though the latency is still debatable as opposed to the memory-based version, the insignificant additional latency can be tolerated for the other parameter's trade-off. The hardware realization of the algorithm also differs according to the technique and strategy being used in the hardware implementation. Hence, in this thesis, we pursue the most efficient DDS module design approach that adopts the lowest latency among CORDIC algorithms.

According to the target application, the requirement of the matrix of parameter changes. The most suitable version is chosen by looking at the parameter's trade-off. In quantum-enhanced communication receiver, additional memory utilization is being avoided to ensure the energy efficiency in the hardware domain. Because the receiver procedure has already occupied a certain amount of memory to store probability values for state discrimination purpose. The rapid feedback/response is essential to

optimize the benefit of quantum instruments. Thus, low latency sinusoidal signal generation is required. However, the spectral purity also plays a significant role in the overall performance of the system, the parameter is the so-called SFDR. Having spur signal results in a higher error, due to the increment of false and miss detection. As of the hardware domain, the common matrices are minimum period and resource utilization. Therefore, in this work, we evaluate the result based on latency, resource usage, logic operator utilization, SFDR, and maximum frequency.

The techniques and strategy being used in the hardware implementation determine the efficiency in the hardware domain result. Employing those strategies can improve the result significantly in one aspect, but it may also degrade the other factor. For this reason, the selection of techniques being employed is important. In this thesis, we overlook strategies that enhance the hardware implementation results while minimizing the possible drawbacks.

The algorithm's selection implies the efficiency of the functionality and also the output being produced. Similar to the argument in the previous paragraph, there is a certain trade-off that should be considered while choosing a suitable algorithm. Linear trade between parameters can be highly preferable when the attenuation/coefficient is large. Such that, by sacrificing the efficiency in latency, meaning longer convergence, by one clock cycle if the maximum frequency increases by a certain amount, then it should be considered as a profitable trade-off. Hence, in our case we consider the CORDIC algorithm approach which turns out to be beneficial despite its longer latency compared to memory-based approach, but the area and energy efficiency are significantly escalated. Similarly, choosing an appropriate CORDIC algorithm based on the trade-off between latency, resource usage, and produced spectral purity is important in this regard. Hence, the latency is mainly determined by the number of angles in the angle set. It results in 64% latency reduction as opposed to that of the conventional CORDIC, yet it maintains the SFDR at a considerably high level with

72.2 dB value.

Finally, we provide a new design of CORDIC based DDS design that utilizes several techniques that improve the overall results which are going to be explained in the following sections. The result of SFDR values for three CORDIC algorithm-based designs are analyzed and evaluated. To our knowledge, there are some publications in the literature that consider SFDR to be a comparison parameter of the result where we believe that it is significant for the target application. The functionality error can be observed point by point differential distance to the desired value, but it does not give the general idea about the effectiveness of the produced signal, for this reason, SFDR is significant. The three CORDIC algorithms are the conventional CORDIC, the modified virtually scaling free CORDIC, and our design. The conventional CORDIC gives the idea of the initial development of the algorithm and its implementation, and the modified virtually scaling free CORDIC is chosen because it uses a strategy that makes it achieves high maximum frequency which will be explained later and it has low latency that can compete with our design, even though still below ours. Our design generates a sinusoidal signal with a moderate-high SFDR value yet it has the lowest area usage in pipeline version and lowest latency compared to the other algorithms. The result section also provides useful parameter matrices which were listed above, that can be used for evaluation of design selection based on the target application.

1.6 Outline of the Thesis

In the next chapter, the previous works that reside in the comparison matrices will be explained. The conventional CORDIC's basic algorithm and its explanation of the equations and also its derivation of hardware compatible stage by stage equation. The modified scaling free CORDIC is elaborated by the derivation of the final equation with the strategy being adopted.

In chapter III, the implementation of our DDS's design is explained. Initially, we introduce the techniques that we chose, namely, domain folding technique, argument reduction technique, redundant CORDIC, and resource sharing technique. These techniques are selected to improve the efficiency of the hardware implementation while maintaining the functionality of the algorithm. Then, we present stages of hardware implementation. In total, we have six stages which have five different architectures. To give an insight into the results we also describe the tools and the device being used so the discussion meets the common ground.

In chapter IV, we list the results based on the parameters we described previously. The terms are also explained to avoid miss understanding.

In the final chapter, we conclude the discussion by analyzing the obtained results. Hence, one can obtain an idea about the proposed design and may consider using it for the desired application by looking at the offered advantages.

CHAPTER II

CORDIC ALGORTIHMS

In this chapter, we explain the CORDIC algorithms that are used in the Results chapter.

2.1 Conventional CORDIC

The first CORDIC algorithm was proposed by removing the burdensome cosine and sine functions multiplications with logic shift operations. Eq. 2 computes a rotation of the initial vector (x, y) to the vector (x', y') with the angular distance of θ.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{2}$$

$$\theta = \sum_{i=0}^{b-1} \alpha_i \tag{3}$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \prod_{i=0}^{b-1} \alpha_i \begin{bmatrix} 1 & -d_i tan\theta \\ d_i tan\theta & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{4}$$

The angle set of α_i converges to θ with a combination of clockwise and/or counterclockwise rotations, see Eq. 3.

Substituting (3) into (2) and taking cos α_i out of the matrix, we obtain Eq. 4, where d_i corresponds to the direction of rotation at the respective stage i. b is the angle set size and the number of iterations. In Eq. 8, z_i is the remaining phase at iteration i. The rotational direction, given as d_i={-1, 1}=sign(z_i). Our goal is to establish a recurrent formula for rotations that can be conveniently calculated on an FPGA.

$$\alpha_i = arctan\theta(2^{-i}) \tag{5}$$

$$\sum_{i=0}^{b-1} cos\alpha_i = K \tag{6}$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = K \begin{bmatrix} 1 & -d_i 2^{-i} \\ d_i 2^{-i} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{7}$$

$$z_i = \theta - \sum_{i=0}^{c-1} \alpha_i \tag{8}$$

K in the Eq. 6 is the overall scale factor that can be pre-calculated for the initial vector (x, y). Substituting Eq. 5 and Eq. 6 into Eq. 4, we obtain Eq. 7. The division by 2^{-i} can be replaced by a logic shift operator. Thus, iterations are written as:

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} = \begin{bmatrix} 1 & -d_i 2^{-i} \\ d_i 2^{-i} & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} \tag{9}$$

The block diagram in Fig. 5 shows three stages (i-1), (i) and (i+1) of a conventional CORDIC as the realization of Eq. 9 [19].

Eq. 5 implies the angle set for the iteration. By selecting 7 iterations (i=[0, 6]), we obtain the following angle set: 45, 26.565, 14.036, 7.125, 3.576, 1.789, 0.895. Note that the conventional CORDIC has 15 iterations. To ensure that z_i is approximately 0 at the end for any destination angle θ, the number of iterations (b in Eq. 4) is specified as 15 ranging from 0 to 14. Given that arctan(2^{-15}) leads in a minute rotation of 0.00699 degrees which is way below nano-rotation approximation, is is not worth of performing rotation that gives additional rotation. This number is also the accuracy limit of the digital system which uses variables with a bit width of 16 bits because shifting more than 15 bits results in 0 in such a variable and has no impact on the algorithm, the operations add redundant latency and produce no modification

on the final output. The range of convergence, that specifies the absolute value of the angle θ, is 99.882. One uses a domain folding technique with 2 blocks that cover [-90, 90) and [90, 270). This way any θ can be reached by locating an appropriate initial vector.

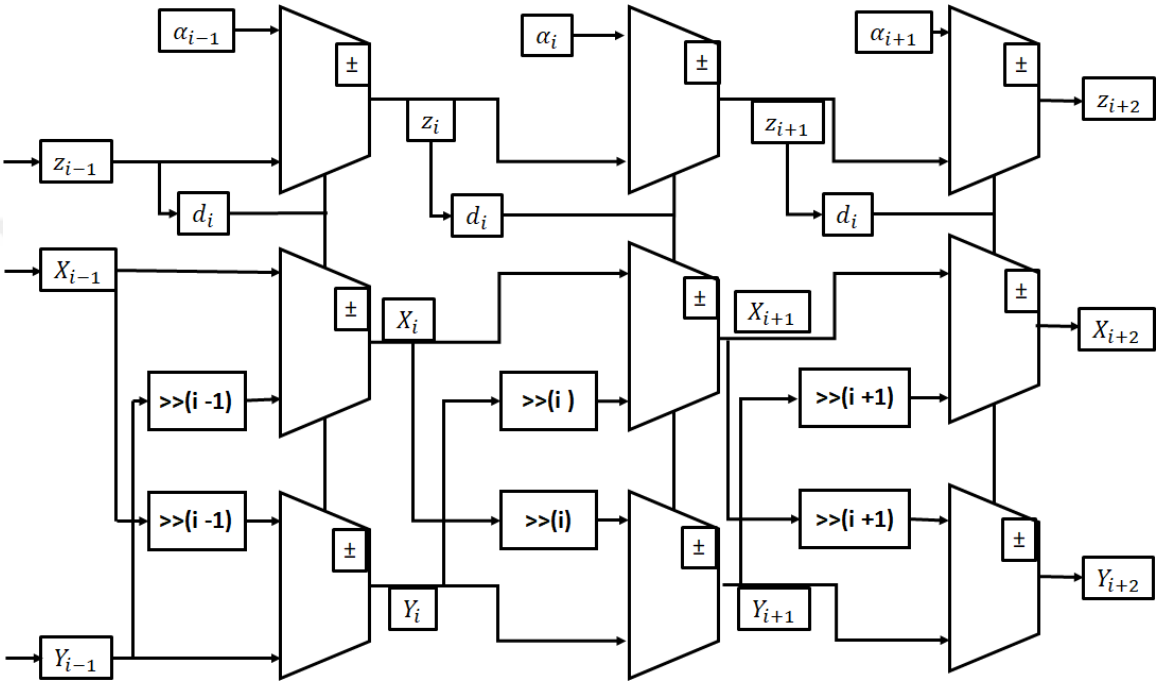

Figure 5: Block diagram of Conventional DDS.

2.2 Scaling Free CORDIC

As the name implies, scaling-free CORDIC pursues an algorithm to avoid multiplication by scale factor prior to the final output for fast performance. Scaling-free CORDIC recognizes one direction of rotation and a halting state, meaning $d_i \in \{0, 1\}$. It rotates counter-clockwise only when z_i is greater than the angle at stage i or stays at the current position otherwise. Thus, z_i is always a positive number. This makes the attainable maximum frequency higher as we will see in the result section. The sine and cosine terms can be simplified when they are considerably small.

$$\left\lceil \frac{w-log_2 6}{3} \right\rceil \leq j \leq w-1 \tag{10}$$

$$\begin{bmatrix} sin\alpha \\ cos\alpha \end{bmatrix} = \begin{bmatrix} 2^{-i} \\ 1-2^{-(2i+1)} \end{bmatrix} \tag{11}$$

The approximation in (11) is accurate if the requirement in (10) is met [1], where w is the bit width. By substituting (11) and (3) into (2) we obtain:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \prod_{i=0}^{b-1} d_i \begin{bmatrix} 1-2^{2i+1} & -2^{-i} \\ 2^{-i} & 1-2^{2i+1} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{12}$$

Then, the recursive formula is given by:

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} = d_i \begin{bmatrix} 1-2^{2i+1} & -2^{-i} \\ 2^{-i} & 1-2^{2i+1} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} \tag{13}$$

Note that (13) has no scale factor unlike in (7). Fig. 6 depicts the block diagram of the scaling-free CORDIC algorithm in stage i.

The low range of convergence compels domain folding technique to squeeze the blocks into several extra regions. Due to the condition in (10), and with bit-width (w in Eq. 10) of 16, where j is i+1, the approximation in (11) only holds for i between 3 to 14, but after 8th iteration, the logic shifter of (2i+1) results in more than 17 bits shift. Thus, iterations 9 through 14 have no effect. Therefore, iterations 9 through 14 are omitted. Similar to the previous argument, that variable has no further effect on the algorithm, the operations add more latency and produce no modification on the output. Hence i goes between 3 to 8. The range of convergence becomes [0, 22.5). The low range of convergence requires extensive use of a domain folding technique. To obtain convergence, 16 domains folding is employed and requires multiplication by a bothersome factor of $1/\sqrt{2}$. Thus, each domain's distance is 20 degrees which is within the range of convergence.

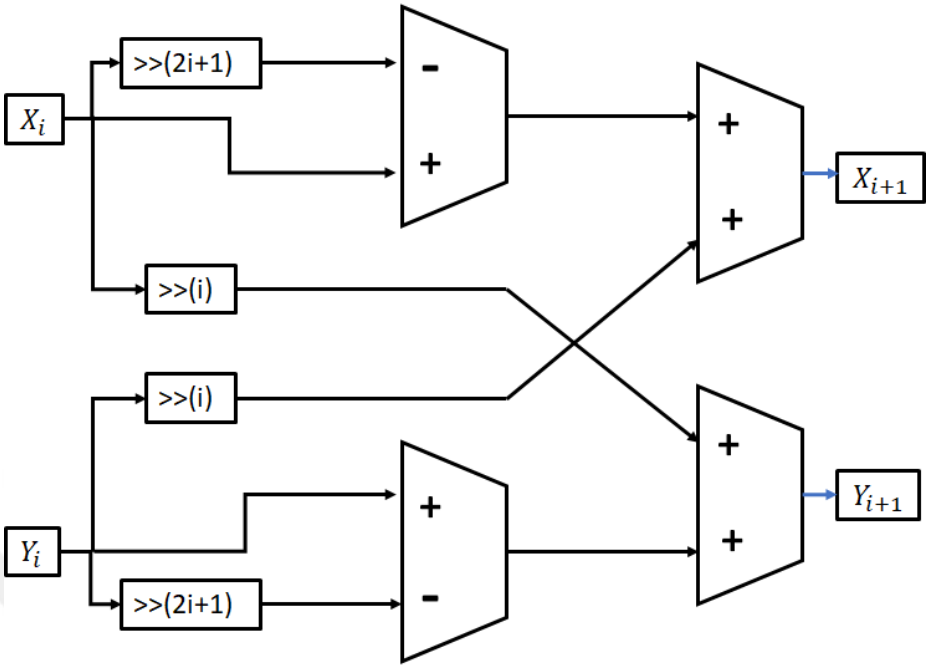

Figure 6: Block diagram of Scaling-Free CORDIC.

The argument reduction technique improves the latency of the method. Particularly, one may jump over several stages by predicting the output at the end of the skipped stages [1]. Typically, more than one computational path is possible, particularly at the early stages. This is because the initial angles are comparatively larger than that of the last stages. These computational paths can be pre-computed, and the results can be assigned using multiplexers, probable combinations of output in the earlier stages are still few and can be estimated by using multiplexers. Hence the first 3 stages are skipped and the output at the end of the 3rd stage is obtained. However, the argument reduction technique requires a variadic scale factor therefore scale factor multiplication is not avoided completely [1]. Nonetheless, this technique reduces a significant amount of latency (from 12 to 9 iterations as we see in the result section). The consequence of not reaching the desired angle due to the insufficiency of the range of convergence is to repeat iteration for the rest of the angular gap. The angular gap means the remaining angle to the desired angle that the range of

convergence couldn't cover. Thus, double and even triple latency may occur. Domain folding and argument reduction techniques are critical in this regard.

2.3 Proposed Algorithm

The angle set is 36.869/16.260/0, 7.125/0, 1.789, 0.895, S*0.112 where S is an integer in a range from 0 to 8 [20]. The range of convergence is (-57.575, 57.575). Thus, to cover the entire space, domain folding is being adopted. Domain folding occurs at the first stage. The computation of each phase assumes different strategies.

Friend angles: any group of angles that have identical magnitude is considered as friend angle[20]. For instance, in Cartesian coordinate R = 4 + 3i with the phase of 36.869 and R = 5 with the phase of 0 are friend angle because they have the same magnitude of 5. Thus, all angles in Fig. 2 are friend angles since they all have the same magnitude of 1. The identical magnitude is essential for the consistency of the system because different magnitudes impose divergence in power gains and result in different scale factors that make the system even more complicated.

Redundant CORDIC: conventional rotator moves vector in either direction: clockwise or counterclockwise. However, rotation with a large angular gap may require the next angles to cover up the unnecessarily extensive jump in a reverse direction. In those cases, holding the position instead of rotating is advantageous. Thus, the direction of rotation is d_i=-1, 0, 1. Adding one more "direction", that is 0 or no angular movement, we still regulate with power gain to attain consistency with the other directions.

Nano rotator: Rotation by sufficiently small angle can be approximated further. Given R=A+Si, a rotation is sufficiently small if S$\ll$A, therefore α=arctan(S/A) $\approx$ S/A. The other rotators are the same as previously explained CORDIC algorithms.

CHAPTER III

FPGA IMPLEMENTATION OF DDS

In this chapter, we explain the techniques that we used for the implementation and also the stages of the architecture.

3.1 Domain Folding Technique

A sinusoidal signal wave has an amplitude range between -1 to 1. The range can be attenuated with any coefficient. Since the signal is continuous, the magnitudes of adjacent points are close to one another due to the hardware platform's digital quantization requirement. However, the curve of the one-period sinusoidal signal has a mirror, inverse, and other symmetries, thus one can divide the phase into several domains [21]. Fig. 7 (a) depicts coordinate plane of the vector (x, y) for full phase range (0, 360] and it is an example of 8 domain divisions with 45-degree phase gaps.

In fact, one can divide even further with a smaller phase difference when noticing symmetry. To meet convergence range of an algorithm that has rotations with tiny angular steps, a higher order of domain division is necessary and it gives faster convergence and lower latency for the computation and it can eliminate re-iteration of the program due to insufficiency of convergence range that results in the increment of latency [22]. However, the drawback is that the hardware implementation encounters the burdensome multiplication to map the final value. For instance, in 16 domain division multiplication with $1/\sqrt{2}$ for each element of the vector costs significant quantization loss and operator usage.

In Fig. 7 (b), the initial vector (x, y) in domain I is mapped to vector (x', y') in domain II, III, and IV. The advantage of the 8 domain division is that the mapping assignment of the initial vector can be done with trivial swapping between the vector's

element and/or negation. Hence, it doesn't cost any additional operator usage and multiplication of irksome factor. In this work, we adopt 8 domain folding to enhance simplicity, area utilization, and timing.

3.2 Argument Reduction Technique

Many algorithms start with trivial initial calculation to prepare for the later computation that usually more sensitive and precise. Similarly, the CORDIC algorithm starts with comparatively bigger angular rotations than the later stages. Hence, the probable combinations of output in the earlier stages are few in number and one can combine the stages or predict the output of those earlier stages by using multiplexers without computational architecture design that requires certain latency and resource.

In Fig. 8, pre-processing block to forecast the output of the first three stages is done by using multiplexers and three bits S flag [1]. In this case, the algorithm accelerates the latency by converging three clock cycles into one and also removes the need for tiresome computations.

3.3 Redundant CORDIC

In its initial development, the CORDIC algorithm strives to commit a rotational move in every stage. Because the algorithm can arrive at the desired point with rotations. So it is considered a loss if there is no rotation at any respective stage. The assumption is that even if initially it rotates with a massive angle, the accumulation of remaining angles can retaliate the redundant angle that the massive angle implies. However, this requirement necessitates the remaining angles to stay intact, meaning we cannot eliminate some of them to improve latency. Thus, by not committing rotation one can reduce the number of angles in the angle set, because the reduced number of angles doesn't need to cover unnecessary angular gap [23]. Therefore, the additional rotational direction, that is no-rotation, can be beneficial in this regard. Still, that no-rotation direction has to be regulated with an identical coefficient as the other

directions to assure consistency of the overall power gain.

In Fig. 9, the red line with tag 1 is the no-rotation direction. The tag 2 is vector rotation of the initial vector at tag 1 in the counter-clockwise direction. The tag 3 is vector rotation of vector at tag 1 in a clockwise direction. Note that rotations tagged 2 and 3 have identical angles to make the hardware realization simpler by only employing multiplexer for the equation's negation part which will be explained in the next section in more detail.

3.4 Resource Sharing Technique

Algorithm level development can complete faster using a software platform since it provides abundant resources that the developer doesn't need to care about. But the real-time and resource usage aspects are often not considered. Hence, by brute-forcing the hardware implementation of the algorithm according to the software realization, the number of operators is approximately the same as the number of operations in the algorithm. One can decrease the hardware resource usage by sharing the same operator for similar operations [24]. Often times, the similar operations with only differences in operands' sign and/or swapping location occur, similar to our case of CORDIC algorithm.

In Fig. 10, two equations A = B+C and D = E+F are similar. By engaging multiplexers, these operations can share one adder given that they occur in a different clock cycle. Hence, the architecture on the left-hand side merges to the diagram on the right-hand side. This reduces the number of adders while using additional multiplexers, but the area requirement and energy consumption of multiplexers are significantly less than adders. This also improves the operator occupancy of the adder and reduces the idle time of the operator that wastes the area and energy usage while in idle state.

3.5 Stages of the Architecture

The following part explains the architecture of each stage in the design. The utilized techniques, figures and equations are also described.

Stage One

The domains folding technique, using 8 domains, retain resource efficiency for the system. Additionally, a higher maximum frequency is achieved using one-directional rotations. Folding the coordinate space into several domains leads to a smaller convergence range, but when the number of domains reaches or exceeds 16, complicated operations such as multiplication by $\frac{1}{\sqrt{2}}$ are required. Thus, we use 8 domains to ensure simplicity. The assignment of the initial vector can be done by trivial swapping between imaginary and real parts and negation as we see in Table 1. The angular range of each domain is 45 which is within the convergence range of the angle set in the counterclockwise direction: it enables one-directional rotation for the next stage.

Table 1: Eight domain folding coordinate assignment.

Domain	*X*	*Y*
0-45	X	Y
45-90	Y	X
90-135	-Y	X
135-180	-X	Y
180-225	-X	-Y
225-270	-Y	-X
270-315	Y	-X
315-360	X	-Y

Stage Two

The first rotation yields 3 phase options with angles 36.869, 16.26, 0. All rotation coefficients have the same magnitudes that imply the same power gain/scale factor. We use coefficients, with an angular magnitude of 1.5625. Hence, R = 1.25

+ 0.9375i for phase of 36.869 degrees, R = 1.5 + 0.4375i for phase of 16.26 degrees, and R = 1.5625 for phase of 0 degrees. Eq. 13 is modified to optimize the hardware implementation for the above three angles such as:

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} 1+2^{-2} & -1+2^{-4} \\ 1-2^{-4} & 1+2^{-2} \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \tag{14}$$

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} 2^{-1}+1 & -2^{-1}+2^{-4} \\ 2^{-1}-2^{-4} & 2^{-1}+1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \tag{15}$$

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} 2^{-1}+1+2^{-4} \\ 2^{-1}+1+2^{-4} \end{bmatrix}^T \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \tag{16}$$

Eq. 14, Eq. 15, and Eq. 16 can be implemented with just logic shifter and adder.

Resource sharing eliminates redundancy in resource usage. In Fig. 11, we use 6 logic shifters as some operators share the same logic shifter's output. The switching rules for the multiplexers are shown as numbers 0, 1, 2, where 0, 1, 2 encodes the jump angle 36.87, 16.26, 0, respectively. The architecture of stage 2 is somewhat complex, which may impact the maximum frequency of the hardware implementation. Thus, having a one-directional rotation approach shortens the longest path of the architecture, and in our case, we have 3 rotational options instead of 6 in the regular mode, which shrinks the area usage and improves the speed of this segment.

Stage Three

In this stage, we adopt redundant CORDIC to eliminate several rotations and guarantee convergence provided by remaining angles in the set. The coefficients of this rotator have an angular magnitude of 1.0078125: R = 1 + 0.125i for a phase of 7.125 degrees, and R = 1.0078125 for a phase of 0 degrees. Eq. 12 turns into:

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & -d2^{-3} \\ d2^{-3} & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} \tag{17}$$

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1+2^{-7} \\ 1+2^{-7} \end{bmatrix}^T \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} \tag{18}$$

Hardware implementation of Eq. 17 and Eq. 18 requires just 2 logic shifters per coordinate. The direction d in (17) can be -1, 1. The rotation by 0 degrees (described by Eq. 18) is equivalent to a no-rotation choice. Such redundancy is tolerable because we end up with three jumping options similar to that of the previous stage. In this regard, no degradation in the maximum frequency of the design results from this architecture. The coefficients in stages 2 and 3 ensure consistency of scale factor as the friend angle's condition is fulfilled.

The block diagram for this stage in Fig. 12 shows 4 multiplexers where two of them have the tag numbers. 0 indicates the halting condition for no rotation of the current vector. Note that the appropriate power gain is imposed. 1 indicates either clockwise or counterclockwise rotation, which is determined by the sign of active phase z.

Stage Four

Entering this stage, the residual angle gap's range is 3.58 which is within the range of convergence of the remaining angle in the set. Hence this stage requires no redundant CORDIC rotation: d={-1, 1}. In this stage, we adopt the conventional CORDIC architecture at the 5th iteration. The coefficient is R = 1 + 0.03125i for a phase of 1.789 degrees, and the hardware compatible computation is given by Eq. 19:

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} = \begin{bmatrix} 1 & -d2^{-j} \\ d2^{-j} & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} \tag{19}$$

The hardware implementation requires two shifters, Fig. 13. For this stage i = 2 and j = 5.

Stage Five

We reuse the conventional CORDIC architecture similar to the previous stage but with R = 1 + 0.015625 for a phase of 0.895 degrees. The hardware compatible computation is also given by Eq. 18, but here i = 3 and j = 6. The block diagram is identical to that of stage 4, Fig. 13.

Stage Six

We are left with a residual angle gap, whose range is 0.875. For this reason, the rotator takes advantage of a nano-rotator approximation with a non-constant, adaptive scaling coefficient: R = 1+ (S * 0.001953125i) for variadic phase, where S ∈ [0, 8].Considering the allowed values of S, the range of convergence is (-0.895, 0.895). The hardware compatible version of the coefficient is given in Eq. 20 and its architecture is shown in Fig. 14. The stage 6 implementation requires extra logic: a scale decoder and an attenuation block.

$$\begin{bmatrix} x_5 \\ y_5 \end{bmatrix} = \begin{bmatrix} 1 & -d2^{-9} * S \\ d2^{-9} * S & 1 \end{bmatrix} \begin{bmatrix} x_4 \\ y_4 \end{bmatrix} \tag{20}$$

The scale decoder block determines the magnitude of the adaptive coefficient of S using the remaining phase, to make the residual of Z as close to 0 as possible. 9 combinations of S are obtained, see Table 2.

Table 2: Remaining angle range for S

Range	S
(0, 0.0988]	0
(0. 0988, 0.1977]	1
(0. 1977, 0.2966]	2
(0. 2966, 0.3955]	3
(0. 3955, 0.4998]	4
(0. 4998, 0.5987]	5
(0. 5987, 0.6976]	6
(0. 6976, 0.7965]	7
(0.7965, 0.895]	8

The attenuation block multiplies the adaptive scale S to the shifted coordinate

variables X and Y.

The FPGA implementation integrates all these stages in series to complete the design. Given the combination of coefficients of all stages, the cumulative scale factor is K = 1.575. Hence, we modify the initial vector in stage 1 by pre-dividing the values by this scale factor, which eliminates the other extra multipliers/dividers.

3.6 Tools and the Device

To ensure a successful implementation, we have chosen the workflow depicted in Fig. 15. We implemented the algorithm as a MATLAB script and simulated the code, taking advantage of functions that ultimately are not feasible in the hardware platform, such as floating-point, exponentiation, and numerous other operators. This reduces design effort and completion time. Then, we verify the result and evaluate the performance in the software domain, which gives us an insight into the possible performance in the hardware domain. We wrote the Register Transfer Level (RTL) implementation of the design in Xilinx ISE using Verilog HDL. Then, we designed the testbench to simulate the RTL implementation and confirm its functionality. Since all variables in the hardware are in integer, we map the hardware simulation results to that of MATLAB simulation for consistency. The hardware platform we utilized the Xilinx Virtex-6 ML605 FPGA board. As a next step, we verify the FPGA's functionality using an Integrated Logic Analyzer (ILA). We store and extract the output values from the ILA signal analyzer, compare the results with that of RTL simulation, and assess the data for evaluation as shown in Fig. 16.

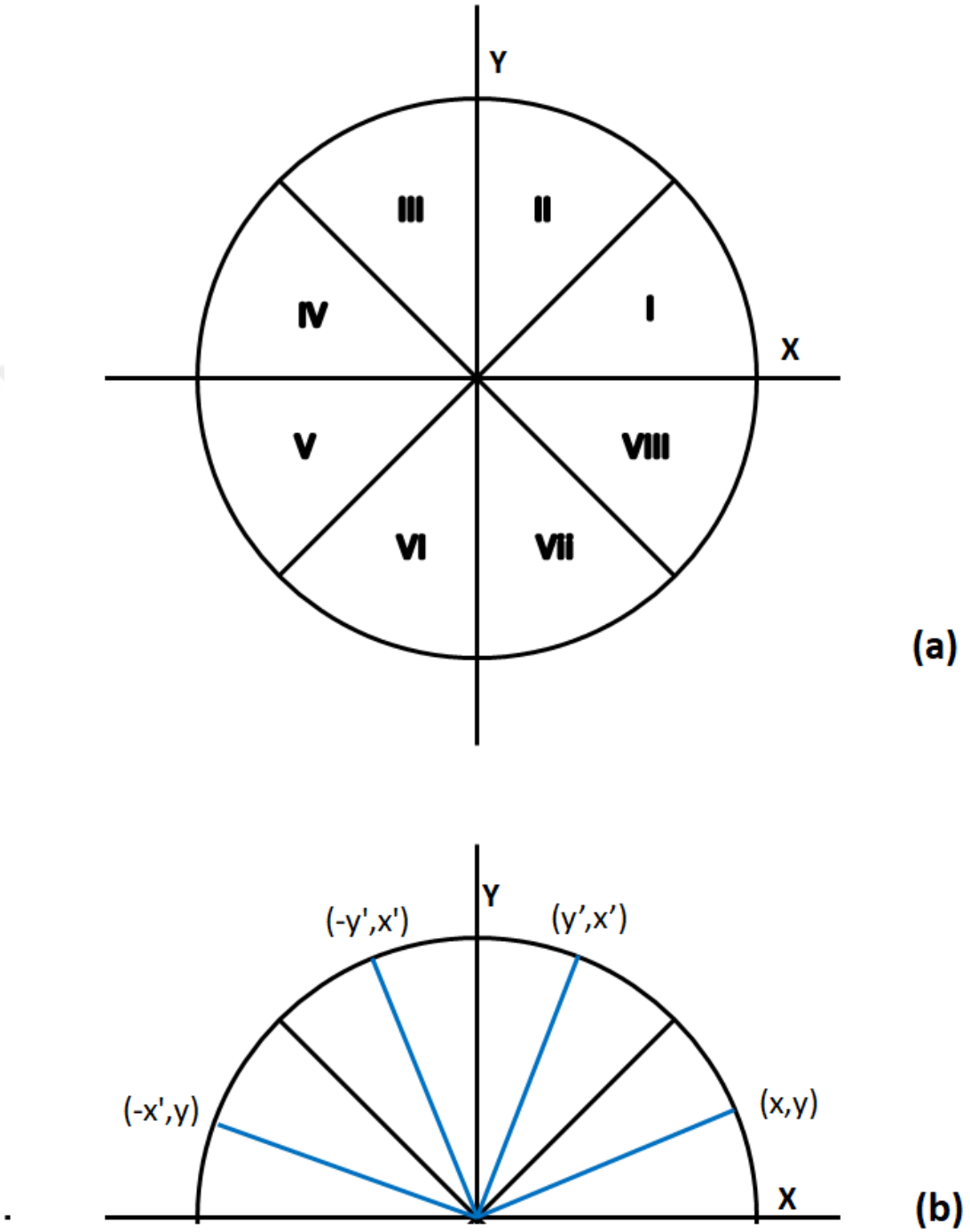

Figure 7: a) 8 domains mapping. b) Mapping of (x, y) to first 4 domains.

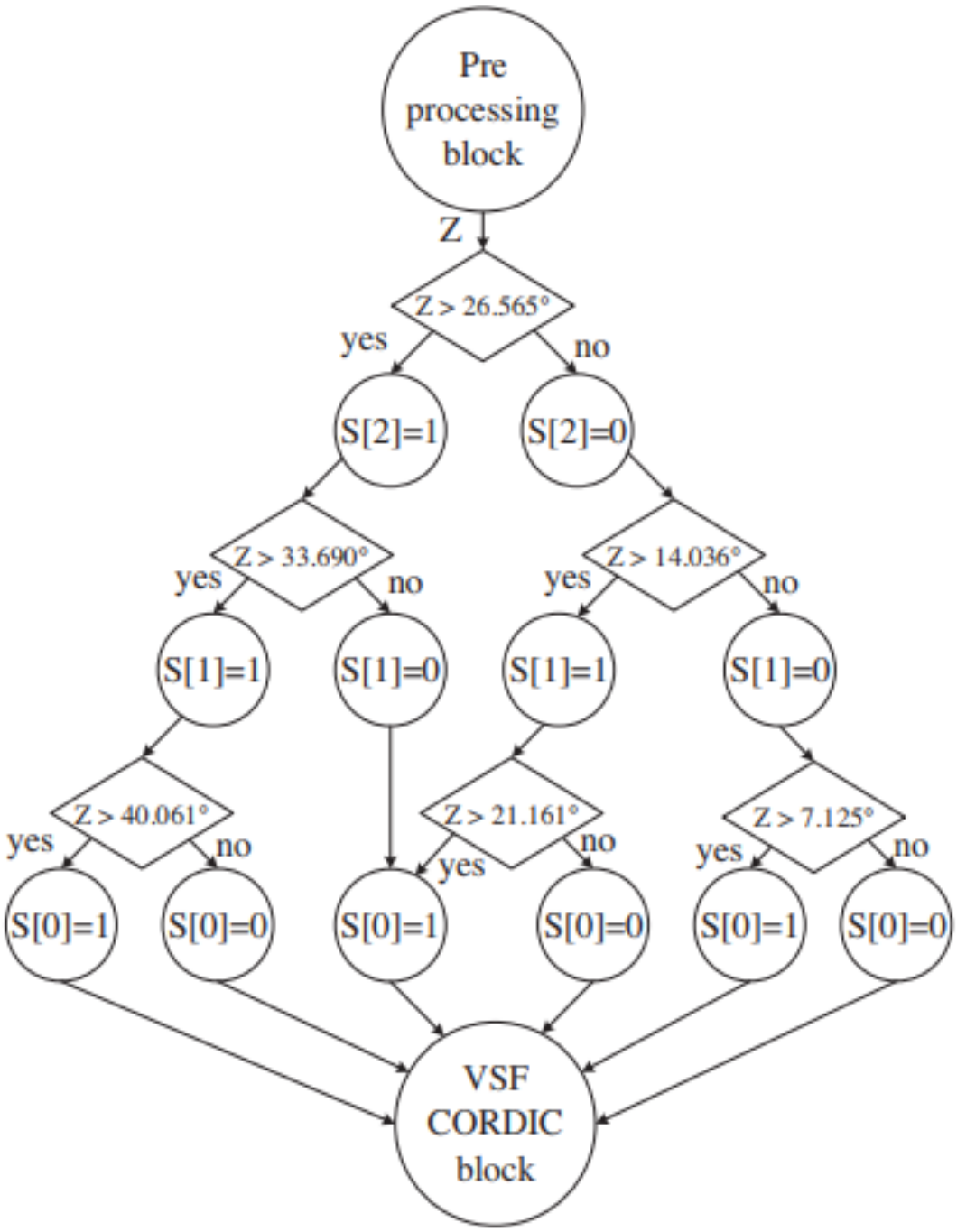

Figure 8: Argument reduction technique to skip 3 stages[1].

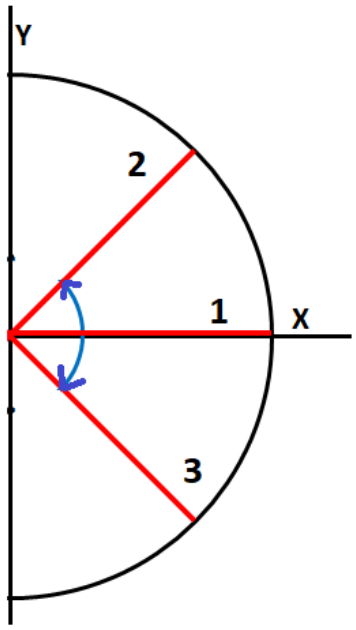

Figure 9: Redundant CORDIC with 3 angular movement options.

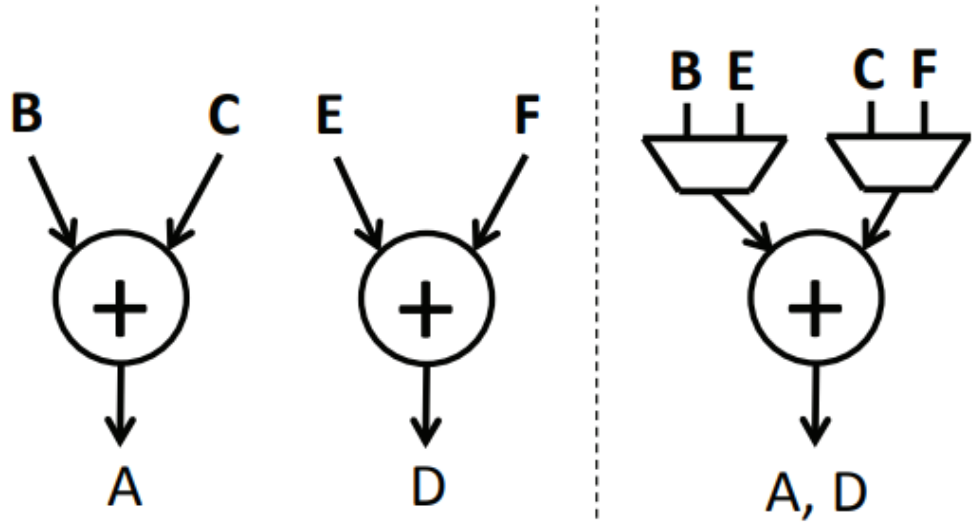

Figure 10: Resource sharing technique.

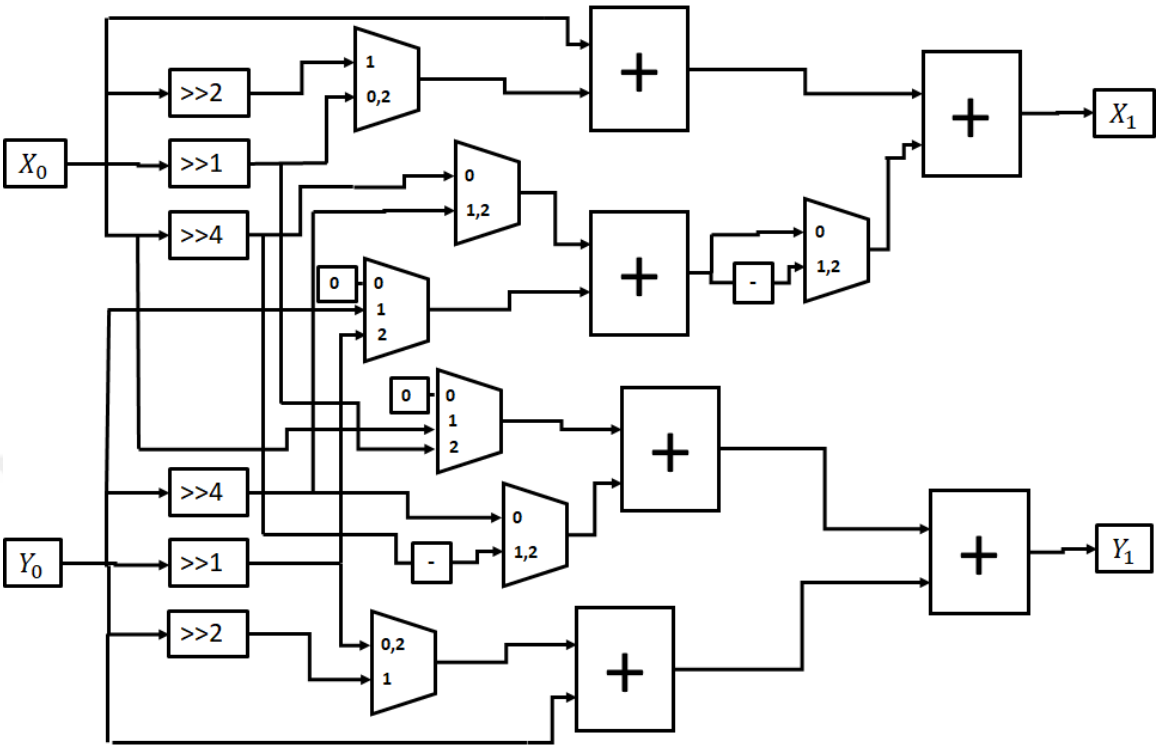

Figure 11: Stage 2 block diagram.

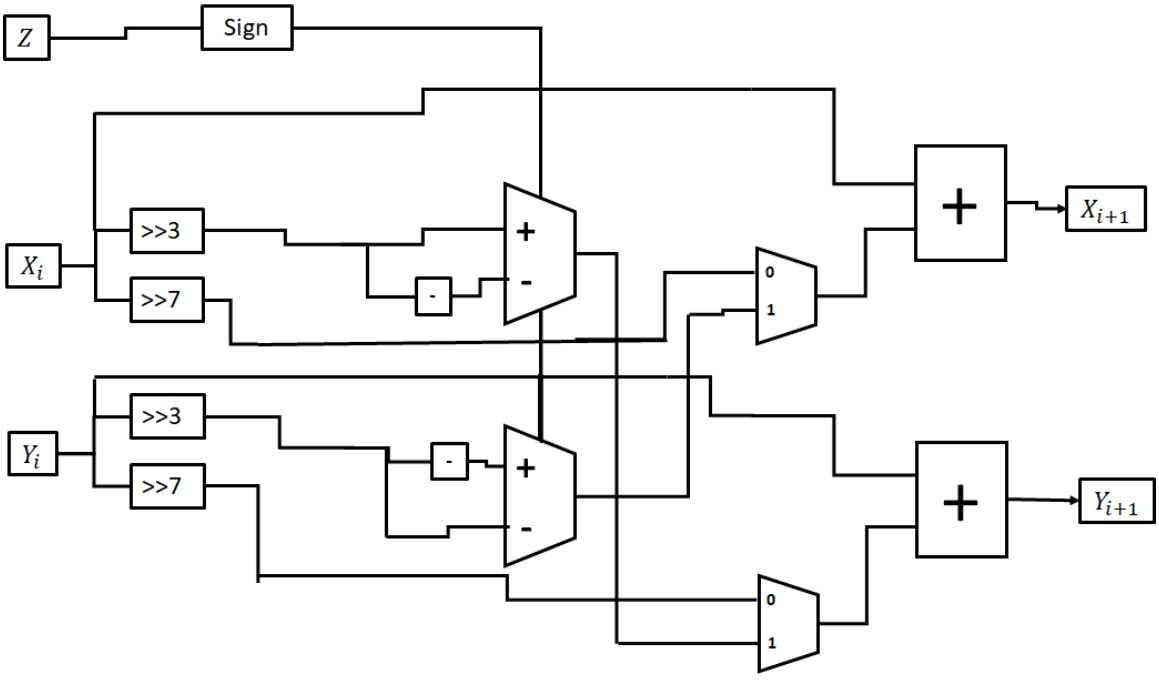

Figure 12: Stage 3 block diagram.

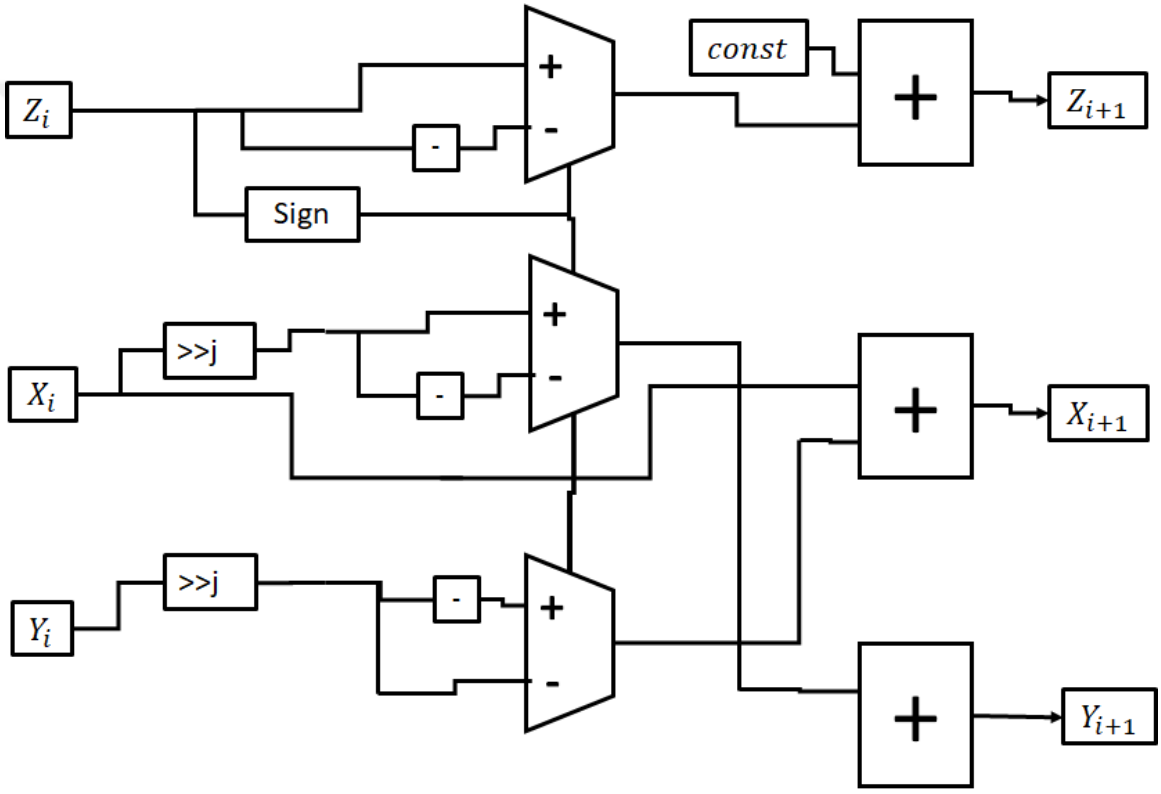

Figure 13: Stage 4 and 5 block diagrams.

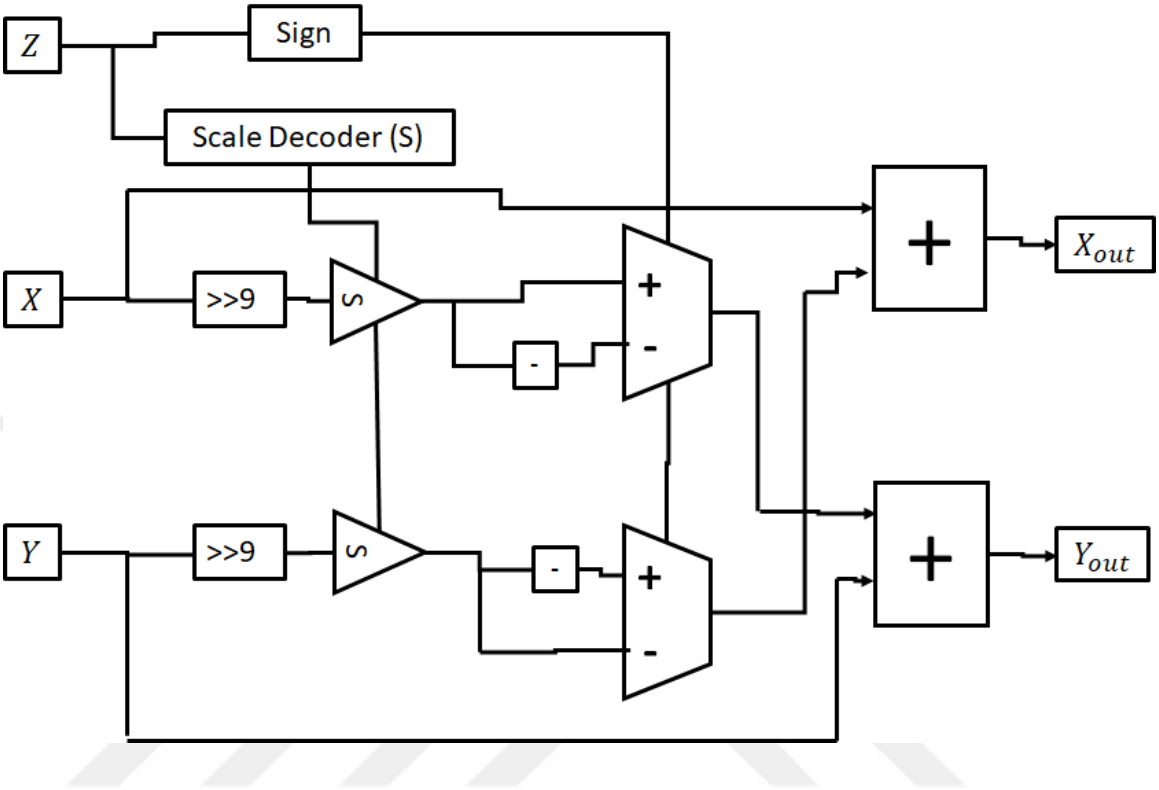

Figure 14: Stage 6 block diagram.

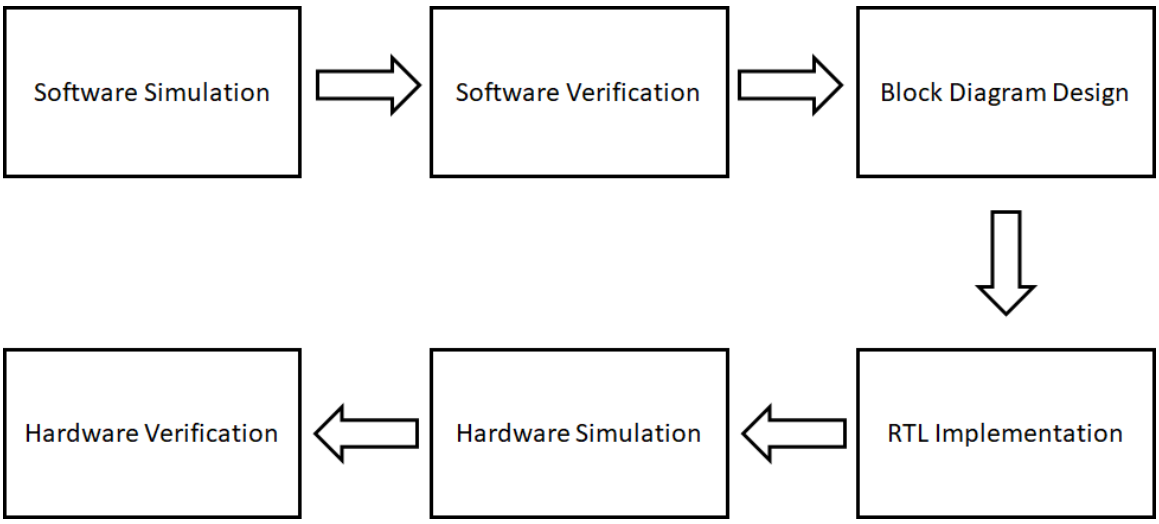

Figure 15: Workflow.

CHAPTER IV

RESULTS

We evaluate the design's performance by measuring the latency, resource usage, logic operator utilization, SFDR, and maximum frequency. These parameters provide trade-off considerations for a target application with specific requirements. For the sake of comparison, we provide values for three different CORDIC-based DDS implementations with and without a pipeline in the architecture. These are conventional CORDIC, modified scaling-free CORDIC, and our proposed design.

Table 3 shows resource utilization based on the number of registers, LUTs, LUT-FF pairs, and RAM for a given target device. Here, the LUTs are the slice logic which is not necessarily using memory. We use ROM as memory: here its usage is measured in bits. In the Table, CORDIC represents the conventional CORDIC (it stores 15 phases for the angle set and assumes 16 bits of variable's width). SF-CORDIC stands for modified scaling free CORDIC algorithm. The "P" next to the algorithm's name indicates the pipelined version. The initiation interval of every pipelined algorithm is 1, meaning the module can take input every clock cycle with no extra delay.

Table 3: Resource Utilization

Algorithm	LUTs	FFs	ROM
CORDIC	764	84	240
CORDIC	2506	840	
SF-CORDIC	479	98	96
SF-CORDIC P	835	340	
Proposed	400	140	
Proposed P	498	206	

Table 4 presents the utilization of logic operators. Mult, Add, Comp, Mux, and Shift stand for the multiplier, adder, comparator, multiplexer, and logic shifter. All

these logic operators run with variables of 16 bits. The logic shifter accepts the variadic length of shift argument.

Table 4: Logic Operator Usage

Algorithm	Mult	Add	Register	Comp	Mux	Shift
CORDIC	1	7	5	4	21	2
CORDIC P	1	100	65	19	60	
SF-CORDIC		10	6	14	58	4
SF-CORDIC P		28	63	19	49	
Proposed	2	16	31	20	42	
Proposed P	2	24	68	22	32	

In Table 5, the values of SFDR are specified in dB. We compute the SFDR by fetching the results obtained from ILA signal analyzer in the MATLAB platform.

Iteration in Table 6 indicates the number of rotations, this number is equal to the number of phases in the set. Latency is specified in the number of clock cycles. It represents the overall delay due to iterations and additional strategies such as domain folding and argument reduction techniques.

Table 7 lists the maximum frequency in MHz if implemented of a Xilinx Virtex-6 FPGA. The first column shows the maximum frequency for State Machine (SM) based DDS and the second one lists that of the pipelined version.

Table 5: SFDR Results

Algorithm	SFDR
CORDIC	92.7
SF-CORDIC	56.8
Proposed	72.2

Table 6: Iteration and Latency

Algorithm	Iteration	Latency
CORDIC	15	17
SF-CORDIC	6	9
Proposed	5	6

Table 7: Iteration and Latency

Algorithm	Max Frequency	Max Frequency P
CORDIC	180	240
SF-CORDIC	226	354
Proposed	211	251

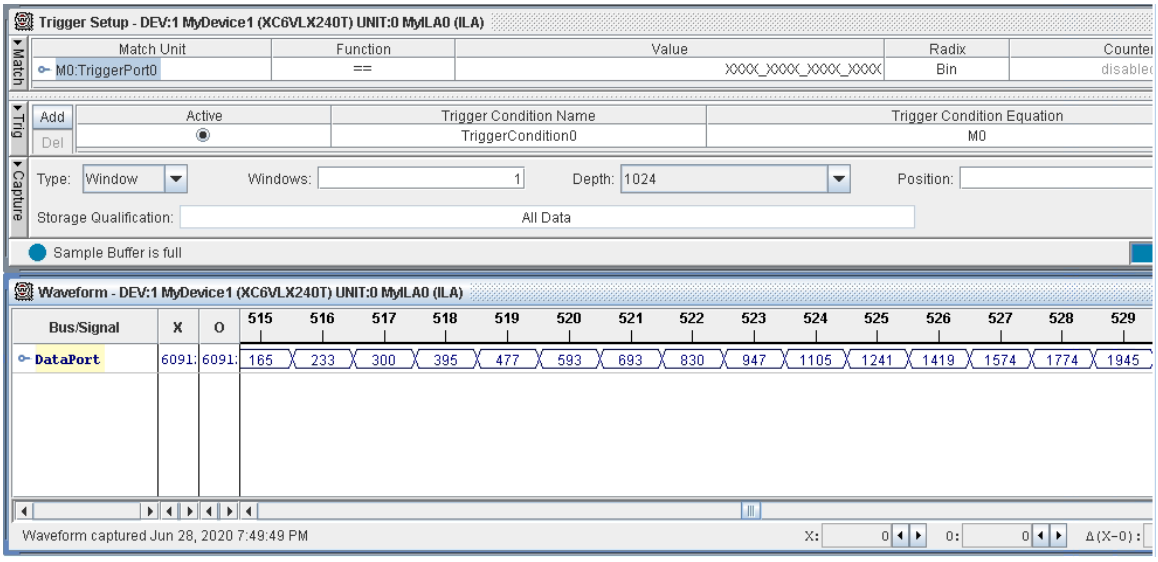

Figure 16: ILA Analyzer.

CHAPTER V

CONCLUSION

In terms of resource utilization proposed design advances in both pipeline based and SM based. The conventional CORDIC occupies the largest memory usage due to numerous angles in the set. Although it uses no memory for the pipelined version the high number of iterations makes the increment of the other resources enormous. Memory is no longer needed because every stage is implemented separately and is active simultaneously, hence each stage is pre-assigned with a constant angle. SM based modified SF CORDIC has the lowest resource as compared to our design, but its pipeline based is behind the proposed design since it employs more iterations than ours. Our SM based design virtually doesn't need iteration because each stage employs a different type of rotator, which makes the resource usage close to that of pipeline-based design. To no surprise, our design occupies the smallest area and provides an energy consumption advantage.

SM based Conventional CORDIC has the least overall logic operators, while the proposed design compares positively to the modified pipelined SF CORDIC. The pipelined version of conventional logic uses significantly more logic operators compared to the SM based one because 15 iterations that run on a set of resources are expanded into 15 identical sets of resources. The same explanation holds for the close figures of logic operator usage in the SM based and pipeline based version of the proposed design. Here, the synthesize tool optimizes the resource allocation by substituting shifter by a concatenation operator due to constant bit shifting. Consequently, the number of logic shifters may not be the same as shown in the block diagrams.

Low latency is desired to enhance the throughput and efficiency of the communication system because delay in the system slows down quantum feedback - a bottleneck and great challenge for quantum-enhanced communication systems. With a latency of just 6 clock cycles, the proposed design is superior to the other algorithms. Although the number of iterations of modified SF-CORDIC is very close to that of the proposed design, there is an extra delay of 3 clock cycles due to a required additional compensation to the rotation.

The modified SF CORDIC has the highest maximum frequency due to one-directional rotation. Our design has a moderate maximum frequency for its implementation. The conventional CORDIC achieves the highest SFDR value due to the high number iteration. Our design achieves moderate SFDR yet the lowest latency, with approximately 20 dB SFDR and 64% latency reductions compared to that of the conventional CORDIC [25].

In conclusion, we report a new memory free low latency DDS architecture. The DDS architecture produces output with high spectral purity, even with low latency performance. Generally, complex value computations could be employed to calculate the trigonometric equation, but those calculations are computationally difficult and energy inefficient. To avoid calculation-related inefficiencies, the common approach is to use a LUT with phase being the input and amplitude being the output. However, the quantization of the LUT limits both the modulation capabilities and the SFDR of the signal. Therefore, to generate a desired smooth radio-frequency signal small-step quantization is needed, requiring a larger LUT. The LUT requirement leads to an increase in memory use and may lead to the reduction of the maximum frequency of the FPGA design which would also limit modulation capabilities. On the other hand, the CORDIC technique offers low complexity and memory-free trigonometric calculation approach, with the expense of extra latency to complete the computation.

With proper angle set selection, it results in shorter latency. We enhance the proposed design with a pipelined approach to shorten latency even more. Thus, in our design, the sinusoidal wave amplitude is obtained every cycle, thus maximizing our modulation capabilities. Hence, the low-resource usage, low latency and high SFDR DDSs are essential for this purpose.

Bibliography

[1] Y. Xue and Z. Ma, "Design and implementation of an efficient modified CORDIC algorithm," in *Proc. of IEEE Int. Conf. on Signal and Image Processing (ICSIP)*, (Wuxi, China), pp. 480–484, 2019.

[2] I. A. Burenkov, M. V. Jabir, N. F. R. Annafianto, A. Battou, and S. V. Polyakov, "Experimental demonstration of time resolving quantum receiver for bandwidth and power efficient communications," in *Proc. of Conf. on Laser Science to Photonic Applications*, (California, USA), 2020.

[3] I. A. Burenkov, M. V. Jabir, D. E. Idrissi, A. Battou, and S. V. Polyakov, "Toward experimental implementation of quantum-enabled, bandwidth and power efficient communications," in *Proc. of Conf. on Lasers and Electro-Optics (CLEO)*, (California, USA), pp. 1–2, 2019.

[4] I. A. Burenkov, O. V. Tikhonova, and S. V. Polyakov, "Quantum receiver for large alphabet communication," in *Optica*, vol. 5, pp. 227–232, 2018.

[5] B.-D. Yang, J. Choi, S.-H. Han, L.-S. Kim, and H.-K. Yu, "An 800-MHz low-power direct digital frequency synthesizer with an on-chip D/A converter," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 5, pp. 761–774, 2004.

[6] F. Cardells Tormo and A. Valls Coquillat, "Optimized FPGA-implementation of quadrature DDS," in *Proc. of IEEE Int. Symp. on Circuits and Systems*, vol. 5, (Arizona, USA), p. 5, 2002.

[7] E. O. Garcia, R. Cumplido, and M. Arias, "Pipelined CORDIC design on FPGA for a digital sine and cosine waves generator," in *Proc. of Int. Conf. on Electrical and Electronics Engineering*, (Veracruz, Mexico), pp. 1–4, 2006.

[8] Z. Jiang, R. Xu, H. Huang, and C. Dong, "Design of a ROM-less direct digital frequency synthesizer on FPGA," *Int. Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 8, pp. 327–340, 2015.

[9] N. Prasad, A. K. Swain, and K. K. Mahapatra, "FPGA implementation of pipelined CORDIC based quadrature direct digital synthesizer with improved SFDR," in *Proc. of Int. Conf. on Circuits, Power and Computing Technologies (ICCPCT)*, (Nagercoil, India), pp. 756–760, 2013.

[10] M. Madheswaran and T. Menakadevi, "An improved direct digital synthesizer using hybrid wave pipelining and CORDIC algorithm for software defined radio," in *Circuits, System, and Signal Processing*, pp. 1219–1238, 2013.

[11] M. Genovese and E. Napoli, "Direct digital frequency synthesizers implemented on high end fpga devices," in *Proc. of the Conf. on Ph.D. Research in Microelectronics and Electronics (PRIME)*, (Villach, Austria), pp. 137–140, 2013.

[12] C. Shan, Z. Chen, H. Yuan, and W. Hu, "Design and implementation of a FPGA-based direct digital synthesizer," in *Proc. of Int. Conf. on Electrical and Control Engineering*, (Yichang, China), pp. 614–617, 2011.

[13] M. M. Anas, R. S. Padiyar, and A. S. Boban, "Implementation of CORDIC algorithm and design of high speed CORDIC algorithm," in *Proc. of Int. Conf. on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, (Chennai, India), pp. 1278–1281, 2017.

[14] Y. S. Gener, S. Gören, and H. F. Uğurdağ, "Lossless look-up table compression for hardware implementation of transcendental functions," in *Proc. of IFIP/IEEE Int. Conf. on Very Large Scale Integration (VLSI-SoC)*, (Cuzco, Peru), pp. 52–57, 2019.

[15] W. Shuqin, H. Yiding, Z. Kaihong, and Y. Zongguang, "A 200 MHz low-power direct digital frequency synthesizer based on mixed structure of angle rotation," in *Proc. of IEEE Int. Conf. on ASIC*, (Changsha, China), pp. 1177–1179, 2009.

[16] R. Xin, X. Zhang, Q. Wang, H. Li, and Z. Li, "An area optimized direct digital frequency synthesizer based on improved hybrid CORDIC algorithm," in *Proc. of Int. Workshop on Signal Design and Its Applications in Communications*, (Chengdu, China), pp. 243–246, 2007.

[17] B. B. Godbole and R. H. Nikam, "FPGA implementation of CORDIC algorithm used in DDS based modulators," *Int. Journal of Advanced Research in Computer and Communication Engineering*, pp. 94–97, 2015.

[18] K. Maharatna, S. Banerjee, E. Grass, M. Krstic, and A. Troya, "Modified virtually scaling-free adaptive CORDIC rotator algorithm and architecture," in *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 15, pp. 1463–1474, 2005.

[19] T. Adiono and R. S. Purba, "Scalable pipelined CORDIC architecture design and implementation in FPGA," in *Proc. of Int. Conf. on Electrical Engineering and Informatics*, vol. 02, (Selangor, Malaysia), pp. 646–649, 2009.

[20] M. Garrido, P. Källström, M. Kumm, and O. Gustafsson, "CORDIC II: A new improved CORDIC algorithm," in *IEEE Trans. on Circuits and Systems II: Express Briefs*, vol. 63, pp. 186–190, 2016.

[21] W. Chen, T. Wu, W. Tang, K. Jin, and G. Huang, "Implementation method of CORDIC algorithm to improve DDFS performance," in *Proc. of Int. Conf. on Electronics Technology (ICET)*, (Chengdu, China), pp. 58–629, 2020.

[22] Y. J. Cao, Y. Wang, and T. Y. Sung, "A ROM-less direct digital frequency synthesizer based on a scaling-free CORDIC algorithm," in *Proc. of Int. Forum on Strategic Technology*, (Harbin, Heilongjiang), pp. 1186–1189, 2011.

[23] A. Wassatsch, S. Dolling, and D. Timmermann, "Area minimization of redundant CORDIC pipeline architectures," in *Proc. of Int. Conf. on Computer Design. VLSI in Computers and Processors*, (Texas, USA), pp. 136–14, 1998.

[24] S. Hadjis, A. Canis, J. H. Anderson, J. Choi, K. Nam, S. D. Brown, and T. S. Czajkowski, "Impact of FPGA architecture on resource sharing in high-level synthesis," in *Proc. of the ACM/SIGDA Int. Symp. on FPGA*, (California, USA), pp. 111–114, 2012.

[25] N. F. R. Annafianto, I. A. Burenkov, H. F. Uğurdağ, and S. V. Polyakov, "FPGA implementation of a low latency and high SFDR direct digital synthesizer for resource-efficient quantum-enhanced communication," in *Proc. of IEEE East-West Design & Test Symp. (EWDTS)*, 2020.

VITA

Name: Nur Fajar Rizqi Annafianto

Date of Birth: 05/19/1993

Language: Turkish, English, Javanese, Bahasa

EDUCATION

- MS: Ozyegin University, Istanbul, Turkey, 2018-2020
- BS: Bilkent University, Ankara, Turkey, 2013-2017
- High School: Mustafa Germirli IHL, Kayseri, Turkey, 2008-2012

Honor

- 100 % Tuition waiver for both BS and MS

Work Experiences

National Institute of Standards and Technology, Maryland, USA, 2019-ongoing

- Working on FPGA implementation for quantum-enhanced communication protocol. Performing modulation and demodulation scheme for the experimental setup. For the realization of novel receiver protocol, designing a discrimination algorithm and module to achieve rapid response given the feedback from quantum measurement instruments, namely, the single-photon detector.
- Working on the evaluation of network security system on the physical layer using FPGA.

Ozyegin University, Istanbul, Turkey, 2018-2019

- Teaching assistant for several computer science and electrical and electronics engineering departments' courses.
- Working on FPGA implementation of orthogonal frequency division multiplexing project at OKATEM laboratory.

TRT, Ankara, Turkey, 2016

- Internship program; working on the Internet of things application to monitor energy consumption of the transmitter tower and designing the circuit of sensor and power measuring circuit.
- Researched wind turbine technologies.

Eldaş A.Ş, Calibration, LVD and Electromagnetic testing, Ankara, Turkey, 2016

- Internship program; Observed and learned to perform LVD and Electromagnetic testing for different classes of device.
- Contributed to the problem solving and the development of devices that do not pass the tests and helped the design engineer to understand the problem and possible solution.

TECHNICAL EXPERIENCE

- Java, MATLAB, VHDL, Assembly8051, Python, Verilog, C++
- LTSpice, COMSOL, Candence, Arduino, Xilinx ISE, Vivado HLS, Quartus Prime, Raspberry Pi

Publications

- N. F. R. Annafianto, M. V. Jabir, I. A. Burenkov, H. F. Uğurdağ, A. Battou, and S. V. Polyakov, "FPGA implementation of a low latency and high SFDR direct

digital synthesizer for resource-efficient quantum-enhanced communication," in *Proc. of IEEE East-West Design & Test Symp. (EWDTS)*, 2020.

- I. A. Burenkov, M. V. Jabir, N. F. R. Annafianto, A. Battou, and S. V. Polyakov, "Experimental demonstration of time resolving quantum receiver for bandwidth and power efficient communications," in *Proc. of Conf. on Lasers and Electro Optics*, (Optical Society of America), p. FF1D.1, 2020.

- M. V. Jabir, I. A. Burenkov, N. F. R. Annafianto, A. Battou, and S. V. Polyakov, "Experimental demonstration of a scalable M-ary quantum receiver," in *Proc. of Photonics West*, (California, USA), p. 8, 2020.