RF Signal Processing in UAVs using FPGAs and Exploration of High-Speed ADC/DAC ICs for Phased Array Applications

MTech. Thesis

By

Archishman Guha

Roll -2102121005

DISCIPLINE OF Astronomy, Astrophysics & Space Engineering

INDIAN INSTITUTE OF TECHNOLOGY INDORE

November, 2022

RF Signal Processing in UAVs using FPGAs and Exploration of High-Speed ADC/DAC ICs for Phased Array Applications

A THESIS

Submitted in partial fulfillment of the
requirements for the award of the degree
of
Master of Technology
Space Engineering
by

Archishman Guha
Roll – 2102121005

DISCIPLINE OF Astronomy, Astrophysics & Space Engineering
INDIAN INSTITUTE OF TECHNOLOGY INDORE

November, 2022

INDIAN INSTITUTE OF TECHNOLOGY INDORE

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the thesis entitled "**Stage 1: Drone/UAV based RF signal acquisition & processing through Red Pitaya and Raspberry Pi, Stage 2: Implementation of a Phased Array with Analog Devices Quad MxFE paired with the Xilinx Ultrascale+ VCU118**" in the partial fulfillment of the requirements for the award of the degree of **MASTER OF TECHNOLOGY** and submitted in the **DISCIPLINE OF Astronomy, Astrophysics & Space Engineering, Indian Institute of Technology Indore**, is an authentic record of my own work carried out during the time period from August 2021 to November 2022 under the supervision of **Prof Abhirup Datta**, HoD, Professor, DAASE, IIT Indore and Mr. Sai Jagini, Senior Engineer, Analog Devices.

The matter presented in this thesis has not been submitted by me for the award of any other degree of this or any other institute.

Archishman Guha

20/04/2023

Signature of the student with date
(Archishman Guha)

This is to certify that the above statement made by the candidate is correct to the best of my/our knowledge.

Signature of the Supervisor of
M.Tech. thesis #1 (with date)
(Prof. Abhirup Datta)

Signature of the Co- Supervisor of
M.Tech. thesis #2 (with date)
(Sai Jagini)

Archishman Guha has successfully given his/her M.Tech. Oral Examination held on **24th April 2023**.

Signature(s) of Supervisor(s) of M.Tech. thesis
Date:

Convener, DPGC
Date: 02/06/2023

Signature of PSPC Member #1
Date:

Signature of PSPC Member #1
Date:

ACKNOWLEDGEMENTS

I would like to thank and acknowledge my supervisor Prof. Abhirup Datta for allowing me the opportunity to work on the thesis which has surely been a great addition to my skillset and knowledge bank. I duly thank IIT Indore for providing me with the resources without which working on this thesis wouldn't have been possible.

I thank Analog Devices, Inc for providing me the opportunity to work with them towards the completion of my thesis. The opportunity would prove to be an instrumental turning point in my career in the form of developing a deeper understanding of my field of work.

Last, but no the least, I would like to thank everyone at DAASE, IIT Indore for their support and guidance.

Thank you,

Archishman Guha

MTech, Space Engineering

DAASE, IIT Indore

Roll: 2102121005

Abstract

This thesis presents a method to acquire and process RF signals, in real-time, for various applications like spectrum sensing, communication purposes, electronic warfare and many more. An RF signal acquisition system starts with the analog front-end constituting the LNA, Mixers, Band-pass filters etc. In the backend, we have an ADC that converts the continuous RF signal to digital form so that they can be processed using a digital devices like CPUs (Raspberry Pi, Intel, AMD etc.) & FPGAs (ZCU104, PYNQ Z1, ZCU102, VCU118).

The project first introduces a full signal acquisition system with a basic analog front-end and the Red Pitaya STEM Lab 125-10 as the signal acquisition device and the acquired samples are sent to a Raspberry Pi 4 (On-Board Computer) for software-based signal processing applications. A small PCB based signal generator has been used as the local oscillator for the system. The Red Pitaya has a sampling rate of 125 MSPS, with a 50 MHz LPF at the RF input. The system consisting of the Red Pitaya, Raspberry Pi, PCB Signal Generator along with a RF amplifier and a mixer will be used for remote data acquisition applications like UAVs, Drones etc.

The Red Pitaya comes with its limitations of low-sampling rate, limited number of input and output ports which stand as a hindrance for high-speed direct RF sampling and implementation of phased arrays for defense and Space Applications. Hence, we plan to use the Quad MxFE from Analog Devices, Inc for this purpose. The Quad MxFE boasts of 16 Rx and 16 Tx channels with a sampling rate ranging from 1.5 GSPS and 4 GSPS. With this we plan to implement a scalable phased array system for Satcom, Electronic Warfare, Radar applications for the L/S/C bands.

Four MxFE software-defined, direct RF sampling transceivers are included in Quad-MxFE, together with related RF front ends, clocking, and power circuits. Phased array radars, electronic warfare, and ground-based SATCOM are the intended uses; more precisely, a 16 transmit/16 receive channel direct sampling phased array at L/S/C band (0.1 GHz to 5GHz) is the intended use. Depending on the user's application, the Rx & Tx RF front-end includes drop-in configurations that enable configurable frequency ranges. In this research, methods for merging multiple element array tiles onto a single processor are presented. There are usual challenges for phased array systems, such as synchronization and stability of the

relative phases of the signals. To support beamforming for radar operations, phase control also needs very accurate knowledge of the relative positions of the array's nodes.

The coordination issues for closed-loop applications, such dispersed communications, have been addressed by several technologies in recent years. More recently, there has been an increase in interest in emerging technologies for open-loop applications, like radar and remote sensing.

In addition to high-speed acquisition and transmission, the AD9081 come with features enabling multichip synchronization. This gives the user the ability to establish large radar systems, the multichip capabilities have been discussed in detail in the thesis. The primary requirement in synchronizing large mixed signal systems is the synthesis of a clock tree to maintain phase alignment of clocks emanating from different clocking ICs. The synthesis of a clock tree has also been explored in this thesis.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

Real-Time Signal processing, in today's world, is one of the most sought-after capabilities in the realm of signal processing. It allows the leveraging of processing signals at locations that aren't within the reach of human beings and sophisticated instruments. Processing some information on-spot instead of sending all the data back to the base station reduces the need of large volume storage spaces. The ability to run a software program on a drone enables us to do the major chunk of processing on the fly. We only take the pain of processing or analyzing the yield of the processed data in the drone. Processing signals in real-time reduces the need for human intervention thus reducing requirement of manpower. We use a Red Pitaya with an ADC sampling at a maximum rate of 125MHz to acquisition signal and then send it over to the Raspberry Pi over ethernet.

1.1: Red Pitaya and Raspberry Pi for data acquisition and processing in drones and UAVs.

The Red Pitaya device uses the AD9608 ADC from Analog Devices to sample RF signals through its input port. The ADC used on the board has a maximum sampling rate of 125MHz, with a 50MHz LPF at the input. The device also houses networking capabilities like Wi-Fi and ethernet that help in transmission of data, here we used an ethernet cable to fetch data into the Raspberry Pi from the Red Pitaya. FPGA programming though Xilinx Vivado is also possible on this device.

The Raspberry Pi on the other hand houses an ARM-Cortex processor as its main processing unit. The Raspberry Pi runs a version of Ubuntu on it. It is on this device where the processing of the data acquired from the Red Pitaya takes place. Research shows the extensive use of Raspberry Pi in this field [1-2]. In this paper we propose an integration of the devices discussed above as a part of our applications. The setup, after being flown on a drone, could be used to locate distress [3] signals acquired by the Tri-Axis Dipole used for direction-of-arrival detection. The drone would carry a receiver antenna along with analog RF signal chain and the Red Pitaya and the Raspberry Pi. The whole setup can also be just used a frequency detector that simply senses and notifies the base user about the presence of any frequency.

The Red Pitaya has only two input RF ports with a low sampling rate of 125 MSPS, this prevents us from using this device for high-speed direct RF digitizing.

1.2: Analog Devices Quad MxFE for RF applications in L/S/C band

To be able to implement phased arrays using high speed-converters we need more sophisticated tools, which are addressed by the Analog Devices Quad Mixed Signal Front-End (MxFE). The Quad MxFE has 4 AD9081 ADCs, with 16 Rx and 16 Tx channels. The device supports flexible RF front ends for filtering, amplification and gain control. Four MxFE® software defined, direct RF sampling transceivers, along with related RF front ends, clocking, and power circuits, are included in the Quad-MxFE System Development Platform. Phased array radars, electronic warfare, and ground-based SATCOM are the intended uses; more precisely, a 16 transmit/16 receive channel direct sampling phased array at L/S/C band (0.1 GHz to 5GHz) is the intended use [4].

The system can be used to enable quick time-to-market development programs for applications like:

- Phased-Array, RADAR, EW, SATCOM
- Communications Infrastructure (Multiband 5G and mmWave 5G)
- Electronic Test and Measurement

1.3: Organization of the thesis

The thesis first introduces the topic and establishes the rationale as to why this deserves attention and effort. First the implementation of a drone/UAV based data acquisition and on-board processing system has been discussed. Following that, citing the shortcomings of the Red Pitaya as a sampling device, we moved onto the Analog Devices Quad MxFE with a higher end ADC chip. Chapter 2 discusses the related work that has been done already and areas where this thesis can be implemented to make systems real-time and more dynamic. The area of phased array for military electronic warfare, radar systems will be explored through the second half of the project. Chapter 3 introduces all the devices used along with their features. Chapter 4 discusses the implementation methodology of the entire project. The building of the signal chain and the hardware design in Vivado have been discussed in this chapter. Chapter 5 elaborates on the results obtained and thesis is concluded by the scope for further developments and the remaining future work.

Chapter 6 shows various example areas where the work has been tested. Chapter 8 introduces the AD9081 IC and is followed by chapter 9 discussing the related eval board. Chapter 10 shows the results of a loopback test performed with the DAC and ADC. Chapters 11-13 explore the different clocking solutions and show the possibility of system expansion through a clock tree synthesis in master-slave mode. To conclude, the final chapter, 14, introduces the Quad MxFE and discusses the results obtained. It also shows the muti-chip sync feature needed for large system integration. The conclusion summarizes the thesis and discusses where the work will find relevance and points out further applications where it will be used.

Chapter 2

Previous Work and Problem Statement Formulation

Real-Time Data acquisition and processing finds multitude of applications in communication systems, military applications like Radars, Space Applications such as SATCOM and Navigation. The plethora of possible technological advancements in this field has drawn huge investments in terms of already completed and on-going research. [1] & [2] talk about how the use of Raspberry Pi is possible as an On-Board Computer (OBC). The usage of Raspberry Pi enables us to run python scripts to implement our software-based signal processing applications. When clubbed with an appropriate RF acquisition device we would be able to run signal processing scripts in real time.

An important military application is detection of unwanted UAVs in the airspace. [5] talks about methods to use signals acquired by an RF radar sensor to detect and classify the type of drone. Here, we may implement the work of this thesis in a manner such that, the entire setup can be flown on a drone and the airspace can be monitored from a remote location (base station). The signals then captured can be passed onto the work of [5] for real-time drone detection and classification. For [5], the DroneRF dataset was sampled at a rate of 40 MHz which is well withing the range of the Red Pitaya Device.

Automatic Modulation Classification is another important RF application. In [6] we see the implementation of an CNN based approach to classify RF signals based on their modulation types. The DeepSig dataset has been used to train and test the model. Once the DL based model is ready, we may test it with the signals acquired remotely on a drone and use to classify it.

Along the same lines, Phased Radars in the L/S/C bands find heavy use in Military (Electronic Warfare), Satellite Communications and other communication systems. Phased Arrays have the ability to alter the direction & shape of the beam. This can be achieved by keeping the antenna elements at a certain distance from each other and applying a phase shift between two consecutive antenna elements [7]. This has been demonstrated for a 32-element phased array of dipole antennas at 300 MHz later in the following sections. We have shown, in collaboration with Analog Devices, Inc, a phased array radar setup for SatCom, EW (Electronic Warfare), Radar Systems on the Quad MxFE (Mixes Signal Front-End).

Chapter 3

Getting to Know the Devices Used

3.1: Red Pitaya STEM Lab 125-10

1. Dual-Core ARM Cortex-A9 MP Core
2. USB, Ethernet, Wi-Fi (external dongle)
3. 2 channels
4. Input Ports:
    i. Sample Rate: 125 MSPS
    ii. ADC resolution: 10 bit
    iii. Full Scale Voltage Range: +-1V / +-20V
    iv. BW: DC-50 MHz
    v. Input Impedance: 1 Mohm
5. Output Ports:
    i. Sample Rate: 125 MSPS
    ii. ADC Resolution: 10 bit
    iii. Full Scale Voltage: +-1V
    iv. Load Impedance: 50 ohm
    v. BW: DC-40 MHz
6. Trigger Input for Synchronization: Through extension connector

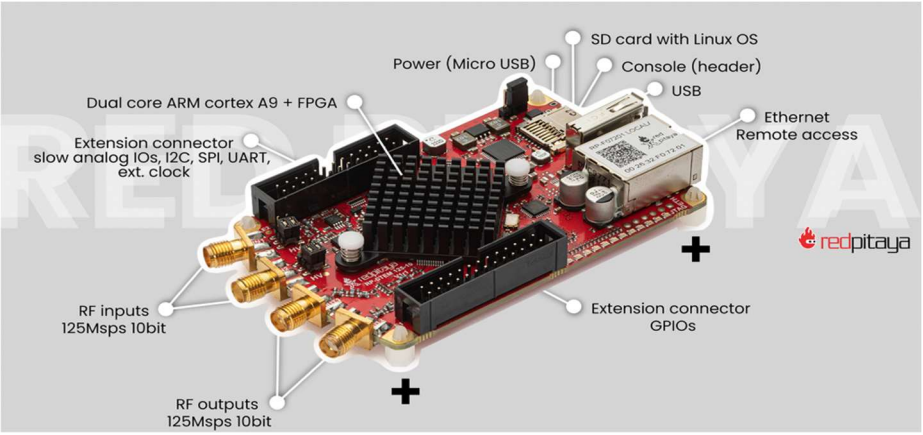

Figure 3.1: The Red Pitaya Stem Lab 125-10, Credits: Red Pitaya

3.2 Analog Devices AD9608 ADC IC

This ADC IC has been used to sample at both the input ports of the RF ADC.

1. 1.8 V supply operation
2. 1.8 V CMOS or LVDS output
3. SNR = 61.7 dBFS at 70 MHz
4. SFDR = dBc at 70 Mhz
5. Digital Non – Linearity = +-0.13 LSB
6. Pipelined ADC, CMOS/LVDS output buffers.
7. User Configuration through SPI interface.

3.3 Analog Devices AD9081 ADC IC

This ADC IC has been used to sample on the Analog Devices Quad MxFE.

1. Each IC has a DACs & 4 ADCs.
2. Maximum DAC sample rate up to 12 GSPS using JESD204C
3. Maximum ADC sample rate up to 4 GSPS using JESD204C
4. Noise Figure: 26.8 dB
5. HD2 & HD3: -67dBFS & -73dBFS
6. SFDR: 3.7 GHz at -70dBc

3.4 Monopole Antenna

Figure 3.2: Monopole Antenna

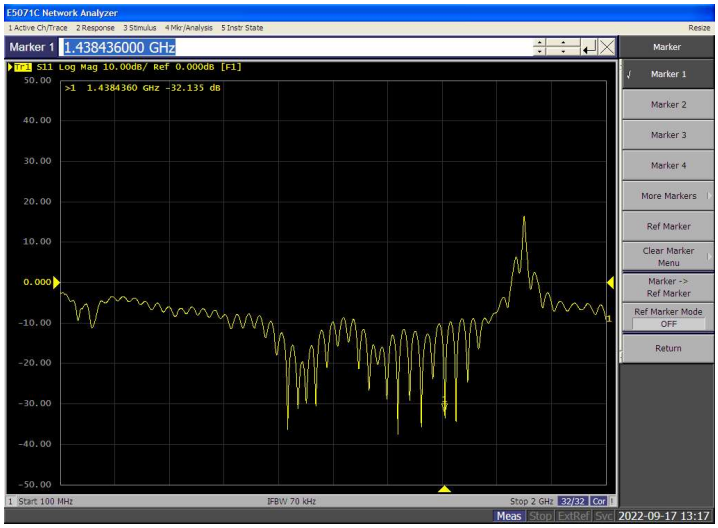

Figure 3.3: S11 characteristics of Monopole Antenna

3.5 RF Amplifier

1. Frequency Range: 0.1 – 4000 MHz
2. Gain: 20 dB

Figure 3.4: RF amplifier

3.6 Mini Circuits RF Mixer ZEM – 4300MH+

1. LO/RF: 200-4300 MHz
2. IF: DC-1000MHz
3. LO-RF Isolation:

i) Typical: 40 dB

ii) Minimum: 20 dB

4. LO-IF isolation: Typ: 14 dB Min: 7 dB

Figure 3.5: RF Mini Circuits Mixer ZEM – 4300 MH+, Credits: Mini-Circuits

3.7 Raspberry Pi

1. Form-factor correct for OBC.
2. Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz.
3. 2.4 GHz Bluetooth.
4. Gigabit Ethernet
5. 2 USB 3.0 ports; 2 USB 2.0 ports.

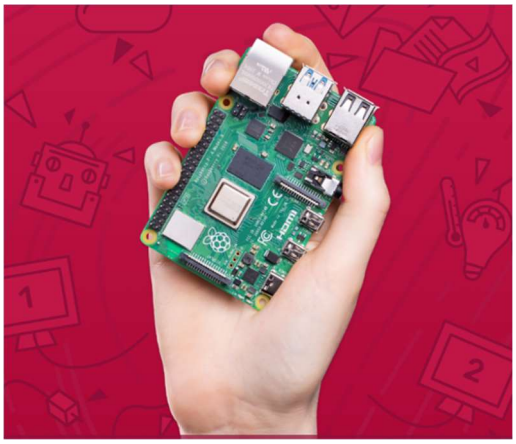

Figure 3.6: Raspberry Pi 4, Credits: Raspberry Pi

3.8 Analog Devices Quad MxFE:

1. 16 RF Receiver 1.5 GSPS to 4 GSPS
2. 48x Digital Down Converters (DDCs)
3. 16x Programmable FIRs

The Quad-MxFE System Evaluation Board, which includes four AD9081 software defined, direct RF sampling transceivers as well as related RF front-end, clocking, and power circuits, serves as the primary source of information for system engineers and software developers. Phased array radars, electronic warfare, and ground-based SATCOM are the intended uses; more precisely, a 16 transmit/16 receive channel direct sampling phased array at L/S/C band (0.1 GHz to 5GHz) is the intended use. Depending on the user's application, the Rx & Tx RF front-end includes drop-in configurations that enable configurable frequency ranges.

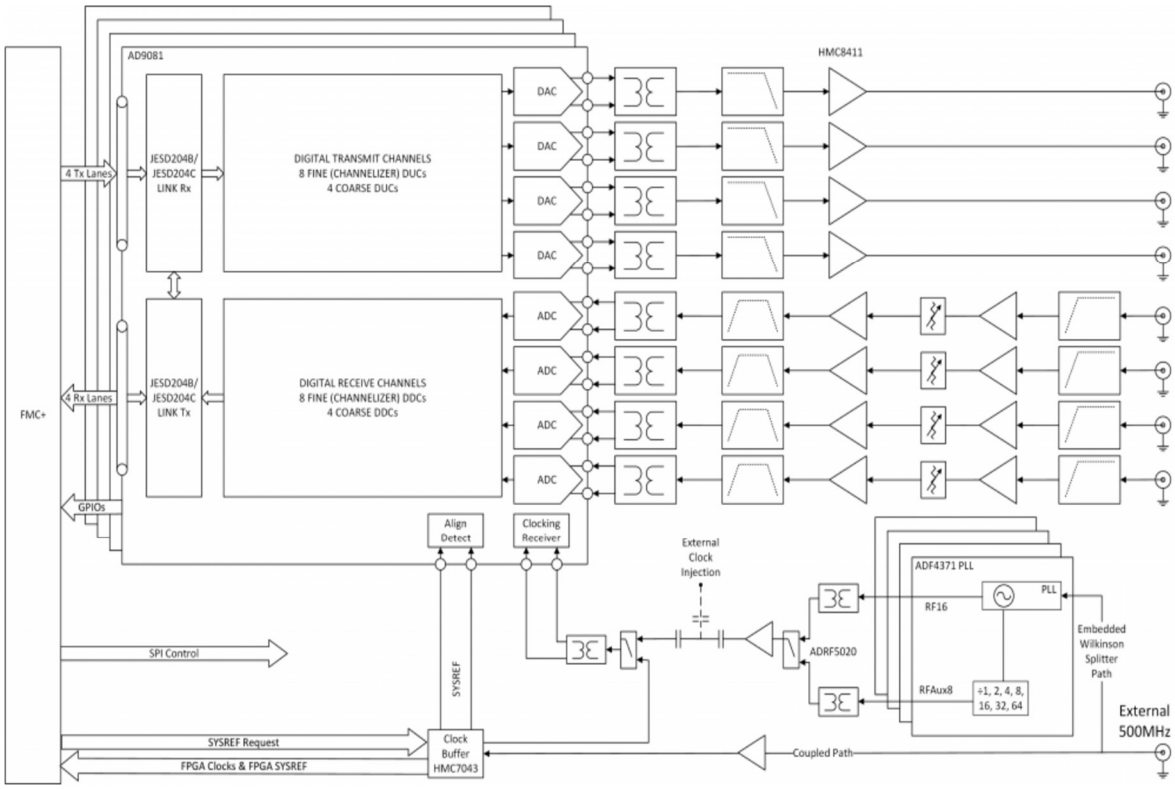

Figure 3.7: High level block diagram, Credits: Analog Devices

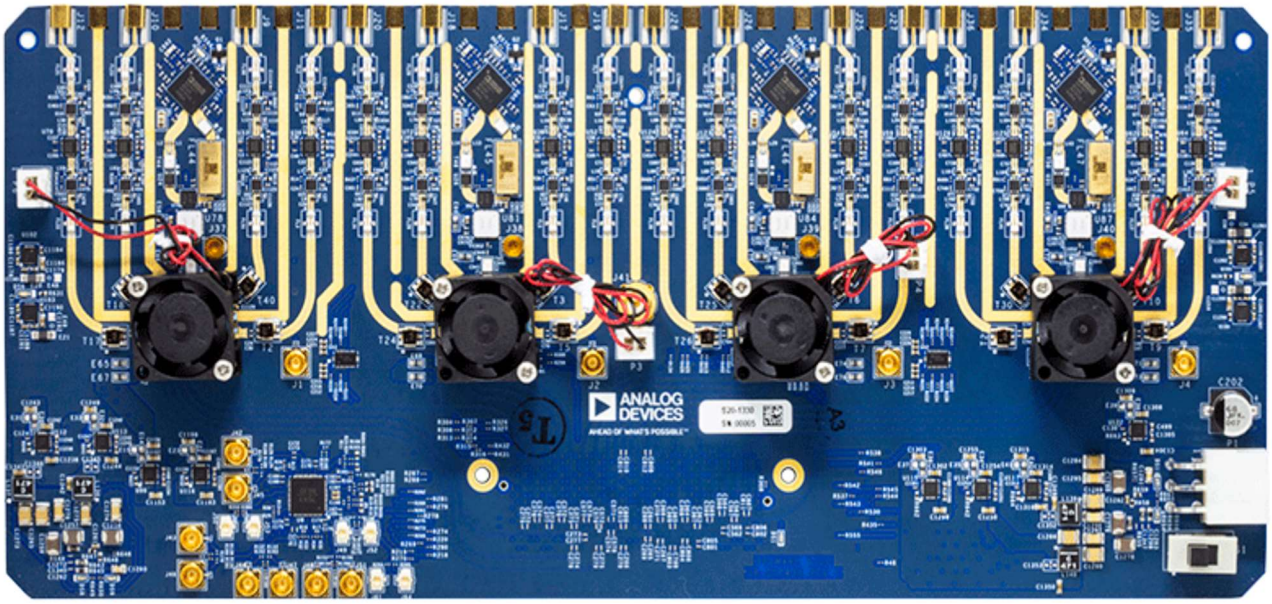

Figure 3.8: Analog Devices Quad MxFE, Credits: Analog Devices

3.9 Xilinx Virtex UltraScale+ FPGA VCU118

1. System Logic Cells: 2586
2. DSP Slices: 6840
3. Memory: 345.9 Mb
4. I/O 832

Figure 3.8: Xilinx Virtex UltraScale+ FPGA VCU118, Credits: Xilinx

Chapter 4

Methodology

First, we have to select an antenna for our specific purpose. This antenna must have a good (very small) S11 parameter in the frequency of interest. One of the most important aspects of acquiring RF signals is

building the signal chain. The analog front-end of the signal chain starts with the receiving antenna, followed by an RF amplifier. For noise sensitive measurements, a RF Low Noise Amplifier, only if the signal of interest isn't strong enough, can be included to boost the signal received. The Red Pitaya can only receive RF signals up to 50MHz, this constraint comes from the sampling rate of the ADC housed in the Red Pitaya Board. To detect frequencies higher than 50MHz, an RF mixer has to be used. The next complexity comes in the form of the requirement of the frequency synthesizer that can generate the local oscillator frequency.

This is achieved through a PCB Signal generator. The output of this generator is fed into the LO port of the mixer. The IF port is connected to the Red Pitaya input port.

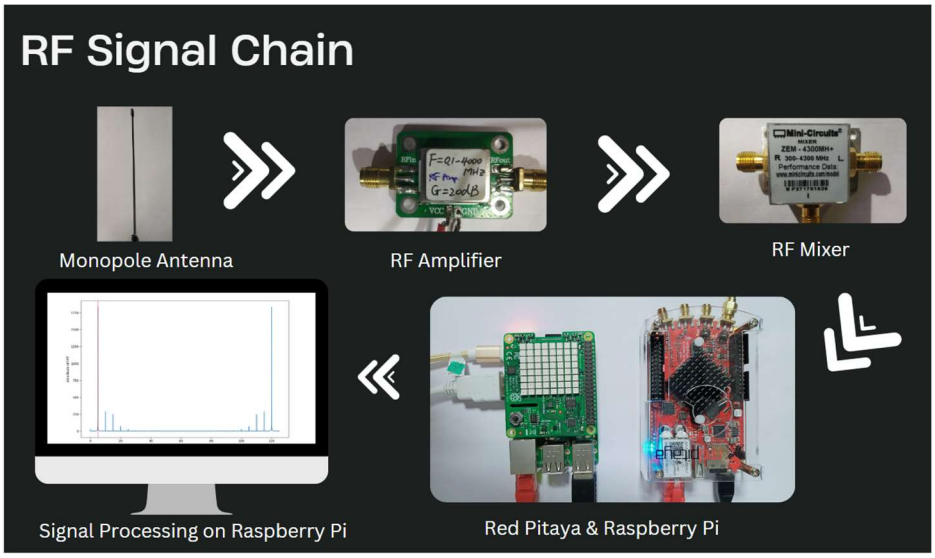

Figure 3.9: RF Signal chain.

In order to achieve more efficiency than is achievable with software operating on a general-purpose CPU alone, a technique known as hardware acceleration refers to the process by which an application will offload specific computing duties onto specialized hardware components within the system. Numerous industries, such as surveillance, automotive, robotics, and medical diagnosis, use hardware accelerators. Hardware accelerators offer the possibility to decrease latency and improve throughput for a particular computation.

The GPU (Graphics Processing Unit), CPU (Central Processing Unit), FPGA (Field Programmable Gate Array), and ASIC (Application Specific IC) are the most often used hardware for AI acceleration. FPGA and ASIC are the alternatives to CPU and GPU because they are sequential in nature and use more power. FPGA is a middle alternative for hardware acceleration because ASIC is not reconfigurable, whereas FPGA is reprogrammable and reconfigurable. Verilog and VHDL are examples of hardware description languages (HDLs) that are specialized in the same semantics as software and synthesize designs into a netlist that can be programmed into an FPGA.

This paper focuses on a processing comparison of how hardware and software in Xilinx platforms will be shown, as well as how to manage which operations are carried out on the FPGA. The proposed algorithms are implemented on Xilinx ZYNQ-7000 series XC7Z020 SoC, called as PYNQ board. PYNQ is a Xilinx open-source project, which uses high-level programming languages such as Python to make it easier to program their platform's language and its libraries, designers can take advantage of programmable logic and parallel processing to create more electronic systems easily.

The filter, known as Finite Impulse Response (FIR), will be described in this paper. Because it is well-known and can be effectively compared, implemented, and used with Python libraries and FPGA, this application was chosen. The FIR (Finite Impulse Response) filter is a finite-length unit impulse response filter, it is non-recursive. FIR filters are one of two types of digital filters commonly used in Digital Signal Processing (DSP) applications, the other being IIR filters. It can ensure arbitrary amplitude-frequency characteristics while maintaining strict linear phase-frequency characteristics, and its unit sampling response is finite, making the filter a stable system. As a result, FIR filters are widely used in fields such as communication, image processing, pattern recognition, and others. FPGA logic fabric enables the implementation of a high-speed, fully parallel version of a design. In the case of FIR filtering, this 'off-loading' of operations from the processor into hardware has the potential to significantly reduce overall execution time.

Filter Implementation

A low-pass finite impulse response filter with a pass band of 0-10 MHz and passband ripple of 2.15 dB was designed in the Tfilter website. The filter coefficients were generated and used in the corresponding hardware design. A stop-band attenuation was -40.83 dB was used. The frequency response from the Tfilter website has been shown in Figure 4.1. Figure 4.1 also has the filter coefficients written in the 16-bit int form. There are 31 taps in the filter.

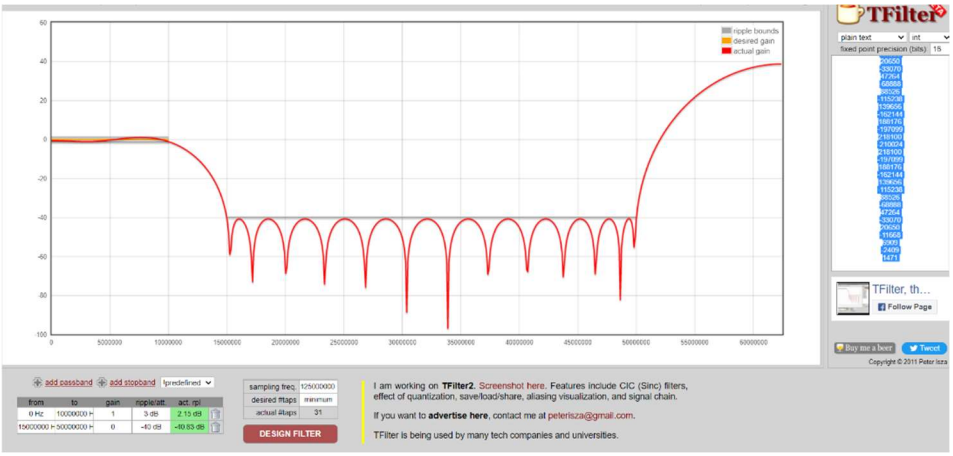

Figure 4.1: Filter Implementation

Hardware Implementation

The hardware design was carried out in Xilinx Vivado software. The free Xilinx FIR filter IP was used (figure 4.2). The LPF is a 32-bit fixed point implementation with a sampling rate of 125 MHz. The filter is a 31-tap filter with integer coefficients. The coeffcients are symmetric and the latency of the design is 26 clock cycles.

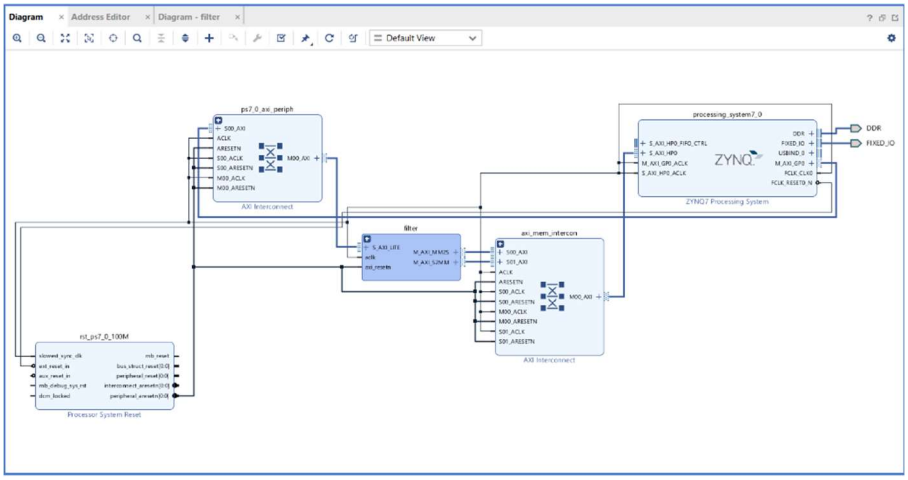

Figure 4.2: Hardware Implementation of FIR filter in Xilinx Vivado

Analog Devices: AD9081

The AD9081 with integrated ADC and DAC cores, has been tested and its features explored through the *Analysis Control & Evaluation Software* from Analog Devices. The procedure involves, first, sleecting a band of interest and then setting up the ADC and DAC configuration accordingly. For our use case, a full bandwidth implementation has been shown.

The control and multi-chip sync of the Quad MxFE has been shown through the implementation of MATLAB scripts. The architecture of the IC has been discussed in detail in the following chapters of the theiss.

Chapter 5:

Takeaways from Stage 1 and Scope of Stage 2:

STAGE 1

Real Time Signal Acquisition & Processing
Working with ADC data
Developing a python pipeline to capture and process data
Interface the Red Pitaya with an On-Board computer, Raspberry Pi
Understanding features of an ADC
Building a signal chain to acquire RF Signals, Antenna, LNAs, RF Amplifiers, Mixers etc.
Understanding design limitations for volume constrained environments, like payloads/drones

Table 5.1: Stage 1 developments

STAGE 2

Implementation of a Phased Array
Working with multiple ADC ICs, understanding multiple-chip synchronization
Developing a software code to implement the Analog Devices Quad MxFE
Interfacing the Quad MxFE with the Xilinx UltraScale + VCU118 FPGA
Understanding features of an ADC
Understanding & Implementing the JESD communication interface
Working with Verilog HDL code to implement designs on the VCU118 FPGA

Table 5.2: Stage 2 targets and scope

Chapter 6

Results and Discussions

Stage 1 Results:

We have been able to successfully to acquire and process RF signals using the Red Pitaya and the Raspberry Pi 4. Here the Red Pitaya has been used as the sampling device and the Raspberry Pi as the on-board computer. The sampled RF signals are passed onto the Raspberry Pi over ethernet and a python script does the necessary processing.

Another utility of On-Board processing capabilities is that we can do a basic processing before dumping everything back to the base station. Real-Time filtering for frequencies of interest and to omit out frequencies that do not interest. This allows us to communicate only when something of interest is caught rather than being in continuous touch eating away valuables storage volume.

As a test, a 1.4 GHz signal was passed into a transmitting monopole. This signal was then captured by the receiving antenna attached to our setup. The PCB signal generator was set to a frequency of 1.43 GHz, thus we received a down-converted frequency of 30 MHz. This is shown in figure 5.1.

To target this for emergency beacons operating at 466 MHz, we can use the same setup by just changing the LO frequency from 1.43 GHz to 496 MHz using the programmable PCB signal generator.

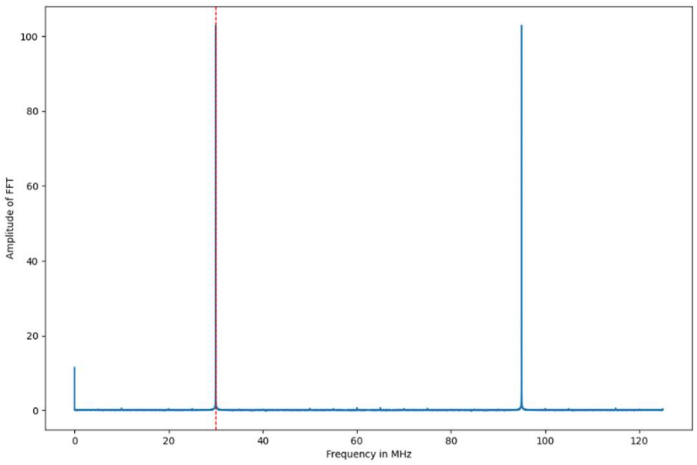

Figure 6.1: The 30 Mhz signal as captured.

As an example of On-Board Signal processing capabilities, we have taken up a low pass filter and filtered out the captured 30 MHz signal from above. Figure 5.2 shows the filter response and figure 5.3 shows the filtered output.

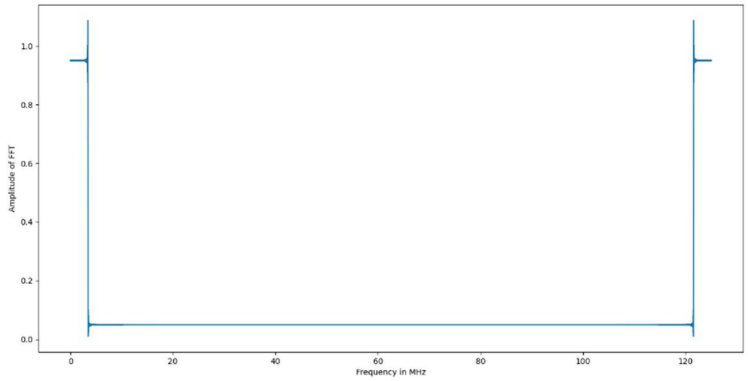

Figure 6.2: LPF Frequency Response (Cut off at 7 MHz)

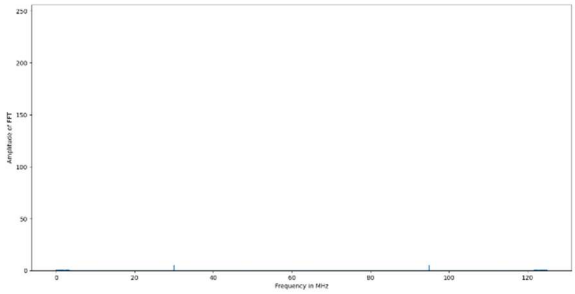

Figure 6.3: Attenuated 30 MHz signal due to response of filter.

Application for SARSAT, targeted for EPIRB, PLB, ELT

The setup has been tested at the 406 MHz frequency used for emergency beacons. The Emergency Position Indicating Radio Beacons (EPIRB), Personal Locator Beacons (PLB), Emergency Locator Beacons (ELB) use this frequency to signal the SARSAT.

A local oscillator at 436 MHz was used to donw-convert the 406 MHz to an IF of 30 MHz, as shown in figure 5.4.

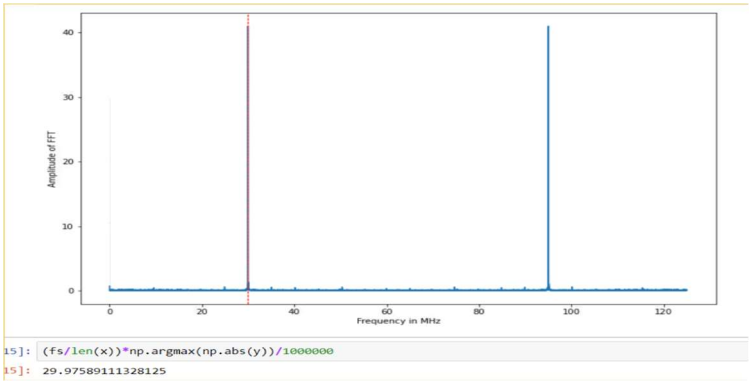

Figure 6.4: 30 MHz IF sampled signal.

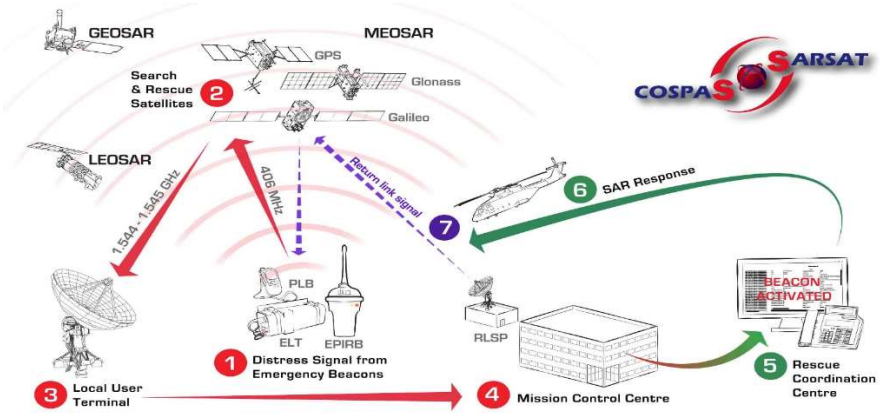

Figure 6.5: SARSAT System Overview, Credits: Credits: SARSAT, www.sarsat.noaa.gov.in

Acceleration of LPF on the Xilinx PYNQ Z1

As discussed before, the hardware for an LPF was implemented on the PYNQ board to benchmark the hardware and the software implementations of the same filter. The attached figures show the results of the filtering operation and the benchmarking results.

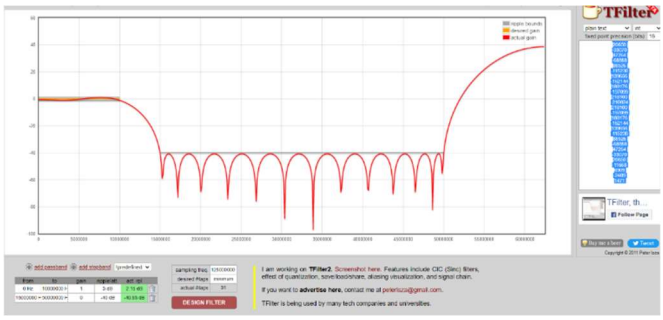

Figure 6.6: Filter frequency response

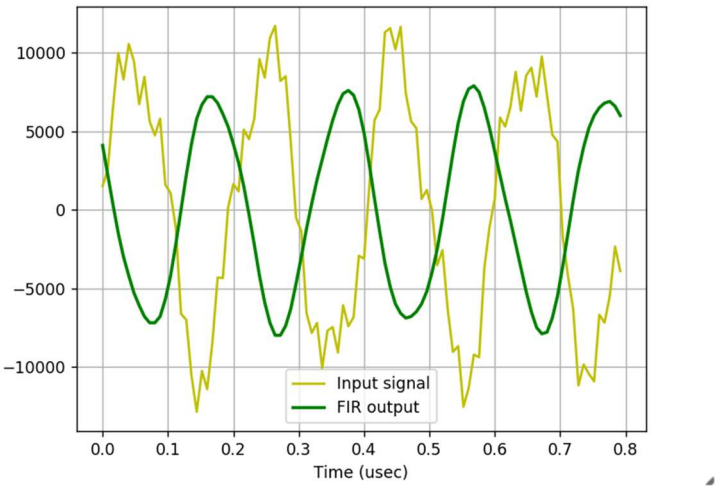

Figure 6.7: Noisy and filtered out signal.

A signal containing a sinusoid at 5 MHz, 12 MHz, 46 MHz was input to the design. A filter using the lfilter function from the SciPy library was used for the software implementation. The input & output signals have been shown in Figure 5.5. Since the filter was designed with a passband of 10 MHz, only the 5 MHz sinusoid is seen in the output and the higher frequencies have been filtered out. Figure 5.6 shows the benchmarking results.

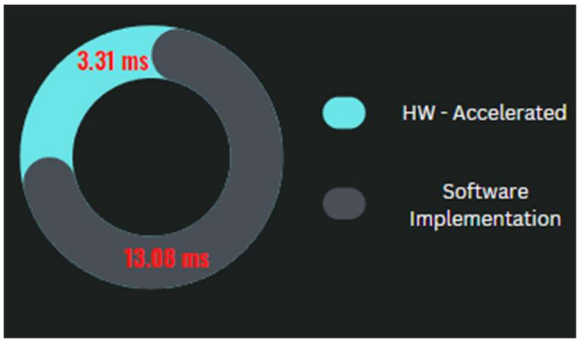

Figure 6.8: Benchmarking results.

Chapter 7

Conclusion of Stage 1 and Proceedings of Stage 2

To take my work forward, I would be working, in collaboration with Analog Devices, Inc., on the Quad MxFE to implement a phased array platform. The Quad MxFE board has 4 AD9081 ADC ICs that are the heart of the device. As a simple demonstration we started with plotting the beam patterns for different values of phase shifts for a 32-element array, the plots of which have been attached below for reference. A striking feature of the MxFE is the communication with JESD. This interface enables high speed data data transfers. This interface addresses problems like low power usage, a smaller number of pins, scalable to higher frequencies & common interfacing with FPGAs. Other common digital interfaces are CMOS, LVDS (Serial & Parallel) & Current Mode Logic (CML, used in JESD). A tabular comparison between CMOS, LVDS and versions of JESD has been shown in figure 6.1.

Function	CMOS	Serial LVDS	JESD204	JESD204A	JESD204B	JESD204C
Specification Release	1990	2001	2006	2008	2011	2017
Maximum Lane Rate (Gbps)	0.2	1.0	3.125	3.125	12.5	32
Multiple Lanes	No	No	No	Yes	Yes	Yes
Lane Synchronization	No	No	No	Yes	Yes	Yes
Multidevice Synchronization	No	No	Yes	Yes	Yes	Yes
Deterministic Latency	No	No	No	No	Yes	Yes
Harmonic Clocking	No	No	No	No	Yes	Yes

Figure 7.1: Tabular comparison of digital interface types, Credits: Siddharth Shah, Analog Devices

The JESD204 interface is used to interface high-speed converters with a host like an FPGA or ASIC. It also uses the 8b/10b encoding which allows more balanced 0 and 1 transition, enabling the Clock Data Recovery.

The python plots for beam pattern for antenna arrays have been shown in Figure 6.2a, 6.2b, 6.2c.

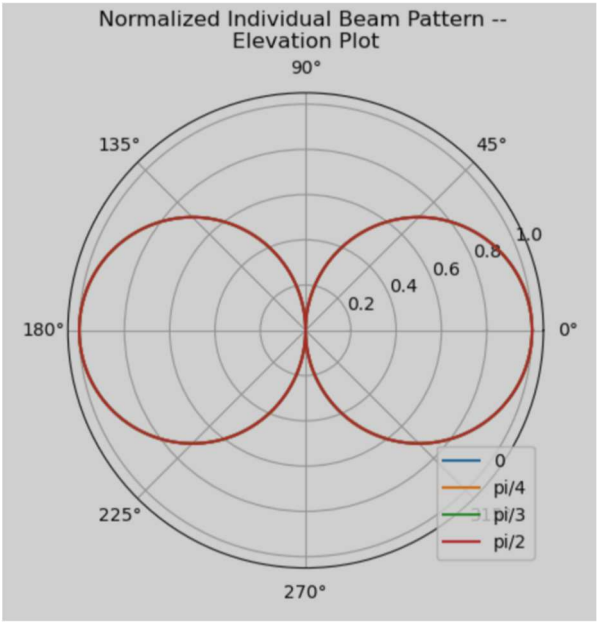

Figure 7.2a: Dipole Beam Pattern for Individual element.

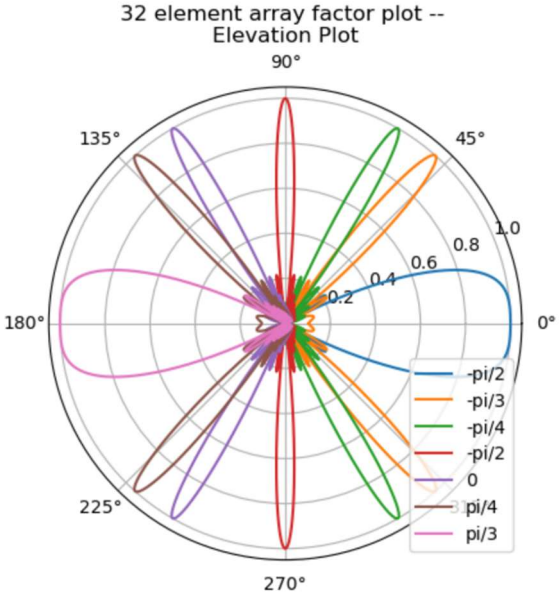

Figure 7.2b: Array factor for 32-element for different phase shifts.

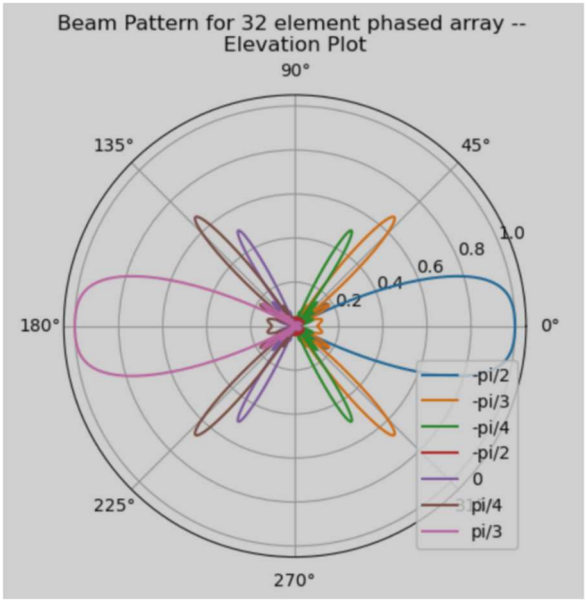

Figure 7.2c: Final Response of Phased Array.

Alongside, stage II will explore different aspects of clocking for RF and associated digital systems, DSP on Silicon – offloading from FPGAs, multi-chip synchronization and much more. The applications of high-speed ADCs & DACs and their performance will also be explored.

The architecture of AD9081 will be the start and delving deep into the working and capabilities of the Mixed Signal Front End integrated DAC and ADC IC will be an interesting exploration. Along with it, stage 2 will also discuss and perform clock tree synthesis and explain the architecture of modern clocking ICs.

Chapter 8: Introduction to AD9081/9082 MxFE

The AD9081Mixed Signal Front-End from Analog Devices has 4 ADCs sampling at 4 GHz and 4 DACs with a sampling frequency of 12 GHz. The device features communication through JESD204C and JESD204B. The JESD204B supports a transfer rate of 15.5 Gbps/Lane and the JESD204C supports 24.75 Gbps/Lane. The JESD204x supports 8 transmit lanes and 8 receive lanes. There is an On-Chip PLL that acts as a clock multiplier and DSP capability. The 4D4A model has a 600 MHz bandwidth per channel.

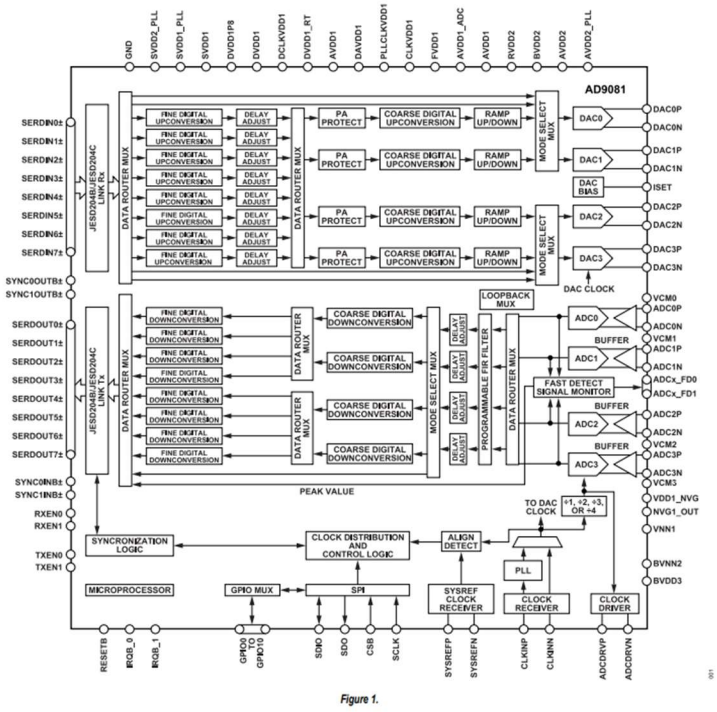

Figure 8.1: Internal Architecture of the AD9081 MxFE IC, Credits: AD9081 Datasheet

Some features of the AD9081:

1. 4 DACs and 4 ADCs
2. External RFCLK input option for off-chip PLL
3. Maximum sample rate up to 12 GSPS using JESD204C – DAC
4. Maximum sample rate up to 4 GSPS using JESD204C – ADC
5. ADC: HD2: −67 dBFS
6. ADC: HD3: −73 dBFS
7. SFDR: DAC: 3.7 GHz -70 dBc

In the architecture in fig 8.1, we see that there are 8 Tx/Rx lanes by the name SERDINx/SERDOUTx. These pins are differential and are based on the CML architecture. Next, up the DATA-ROUTER MUX configuration allows data to be channeled into user-configurable Fine Digital Up Converters which are 8 in number. The data-path also supports Delay Adjust blocks to provide programmable delay to each data-path.

Each ADC and DAC has a Coarse Digital Up-Converter along the datapath. The Fine Digital Up-Converter has a 48-bit NCO in the Tx path, with a frequency range between -750 MHz to 750 MHz. The Coarse Digital Up-Converter has another 48-bit NCO that has a frequency range of -6 to 6 GHz. The receive path also has 192-tap programmable FIR filters.

In the receive path again, there are 48-bit Fine and Coarse digital upconverters. The fine and coarse have frequency ranges from -750 MHz to 750 MHz & -2 to 2 GHz respectively.

To adjust with the data rates, the transmit and receive paths have interpolation and decimation features. These help in matching the IQ sample rates that come from the FPGA or to it and the DAC/ADC sample rates.

The chip features an on-chip PLL that allows an external reference clock or a method to direct clock the DAC and ADC. The ADC clock is always an integer multiple of the DAC clock. Next up, the operation of the AD9081 will be discussed based on an evaluation board.

The On-Chip PLL reference clock can be input a maximum frequency of 750 MHz, when the reference clock divider is at 1. The VCO can also output a maximum frequency of 12 GHz, when the VCO divider is set at 1.

The Maximum DAC sample rate is at 12 GHz and the minimum DAC sample rate is at 2.9 GHz. The Maximum and Minimum ADC sample rates are 4 GHz & 1.45 GHz.

Chapter 9:

Analog Devices AD9081 Eval Board

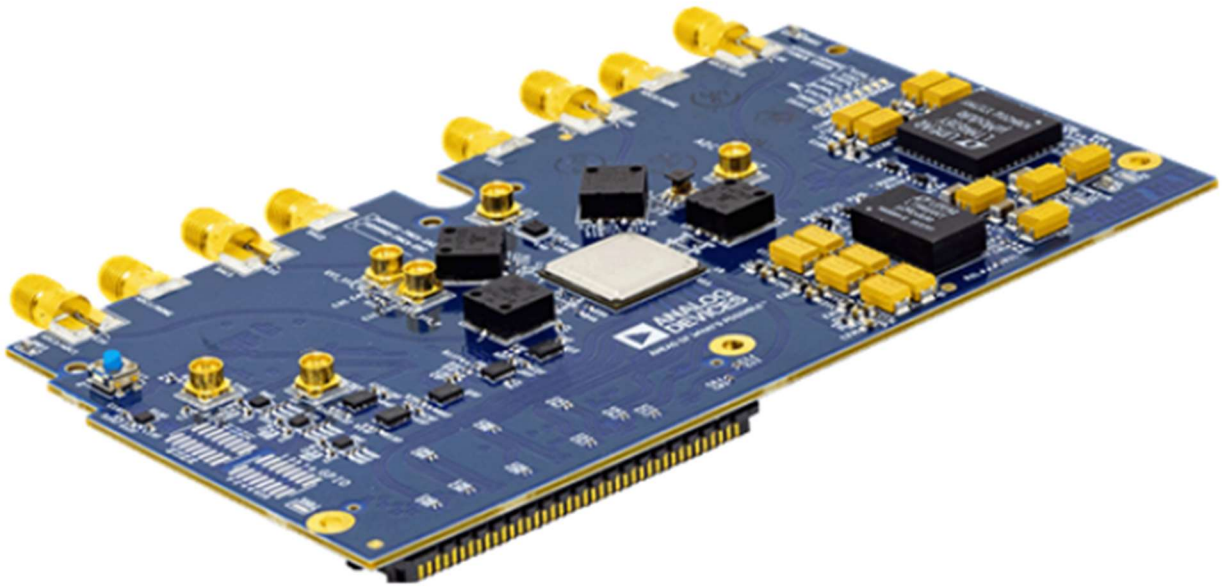

Figure 9.1: Analog Devices, AD9081/9082 MxFE Eval Board

Figure 9.1 shows the AD9081 Eval-Board used to test the AD9081 chip. This eval board is connected to the ADS9 FPGA Board that acts as an interface to the user PC and the Eval Board.

The Analog Devices Clocking IC HMC7044 which is a jitter cleaner two PLL clock IC, this will be discussed in detail in the later chapters of this thesis. The HMC7044 has been used to generate the reference clock and the required SYSREFs for the AD9081, JESD204x interface and the ADS9 FPGA. The board also offers external direct clocking to the DAC and ADC, External Reference, and SYNC for the HMC7044 & External SYSREFs. These help in expansion to larger systems and synchronize multiple Eval Boards and host processor systems. The test results from the Eval board have been shown in the following slides.

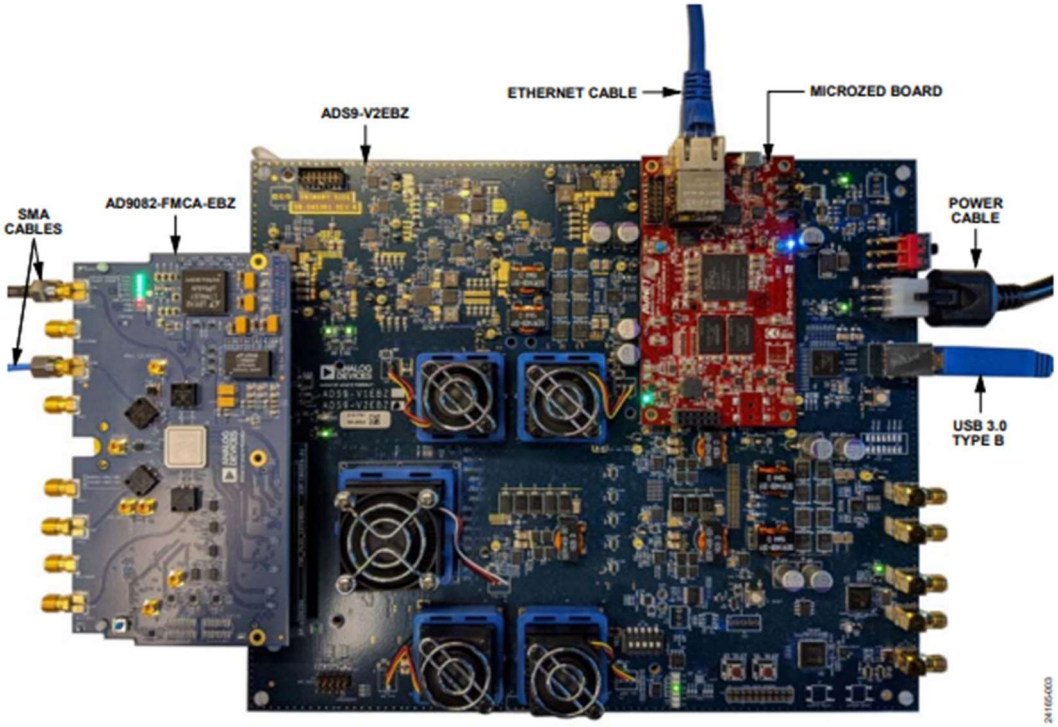

Figure 9.2: The entire AD9081 Eval Board attached to the ADS9 FPGA.

The Verilog design has been implemented on a Xilinx Ultrascale+ FPGA. The communication to the host user PC is handled through a MicroZed Board which has an ZYNQ7000 FPGA on-board.

The UltraScale+ FPGA sits below the largest fan among the 5. The rest smaller fans are the Block RAMs that act as data storage buffers for the AD9081.

Chapter 10:

Signal Generation and Acquisition using MxFE and ADS9 FPGA Board

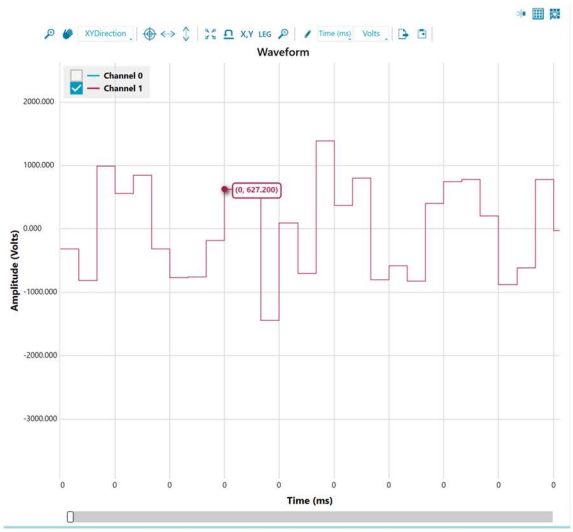

Figure 10.1: Square Wave Generation using MxFE.

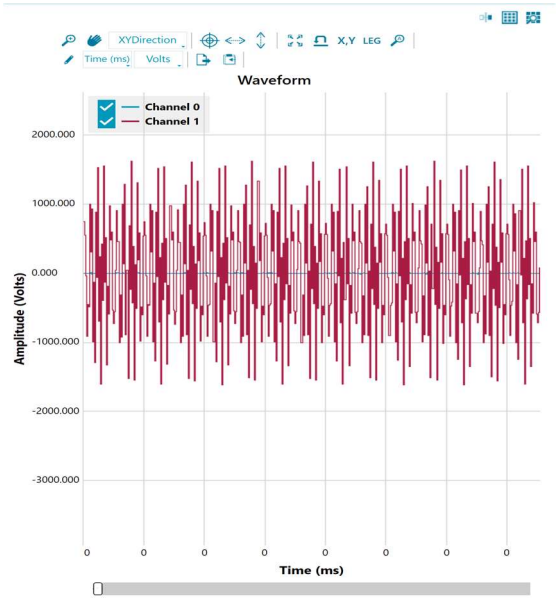

Figure 10.2: Sine Wave Generation using MxFE.

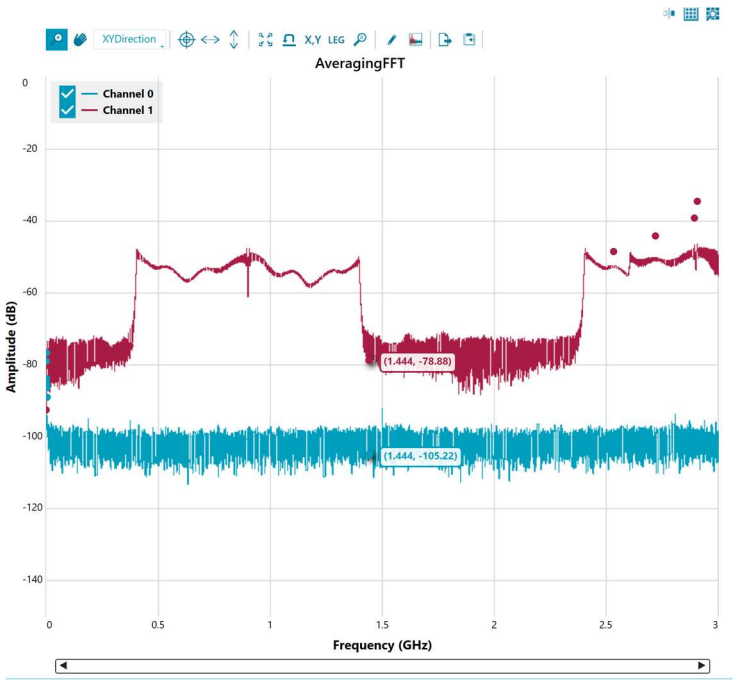

Figure 10.3: FFT of chirp waveform generated by MxFE

Results

Sample Frequency:	6 GHz
Samples:	65536
Noise / Hz:	-148.291 dBFSPerHz
Avg Bin Noise:	-98.674 dB
SNR:	47.039 dB
SNR FS:	53.52 dB
SFDR:	67.788 dB
THD:	-3.09 dBFS
SINAD:	47.04 dB
DC Frequency:	0 Hz
DC Power:	-77.986 dBFS
Fund Frequency:	2 GHz
Fund Power:	-6.481 dBFS
Harm 2 Frequency:	2 GHz
Harm 2 Power:	-6.589 dBFS
Harm 3 Frequency:	183.105 kHz
Harm 3 Power:	-77.816 dBFS
Harm 4 Frequency:	2 GHz
Harm 4 Power:	-6.548 dBFS
Harm 5 Frequency:	2 GHz
Harm 5 Power:	-13 dBFS
Worst Other Frequency:	2.129 GHz
Worst Other Power:	-74.269 dBFS

Figure 10.4: Results of Acquisition

Chapter 11:

Clocking System for ADC and DAC based Acquisition Systems

The clock is the heart of any digital system and essentially drives all the acquisition and processing on complex electronic systems. For this application, the HMC7044 was studied, and its features explored.

This chapter will delve into the architecture of clocking devices used in these RF systems. For this matter two clocking ICs, HMC7044 and a similar one, the LTC6952 will be discussed. Later, the synthesis of a clock tree using the clocking devices will also be shown to establish their utility in synchronizing multiple systems.

The HMC7044 is a 14-ouput dual-loop jitter cleaner clocking IC. This IC has a very low RMS jitter at 44fs at 2457.6 MHz at an offset of 12 kHz to 20 MHz. The device also has a very low noise floor and low phase noise. The has two PLLs with integrated VCOs. It also has an option to use an external VCO upto 6GHz.

Overview of operation of HMC 7044

1. The first PLL, PLL1 acts as the jitter cleaner.
2. Select one of the 4 references through the Reference Mux Selector. The selected Reference is then divided using the pre-scaler.
3. The signal is then fed to the Phase detector and the On board VCXO running at 122.88 MHz is fed through the N1 divider
4. A sensor watched both up and down pulses and checks is the phase error is larger than 4 ns. If yes, the charge pump is tri-stated & the holdover is initialized.
5. The PLL1 locks to the selected reference and provides a clean signal to PLL2.

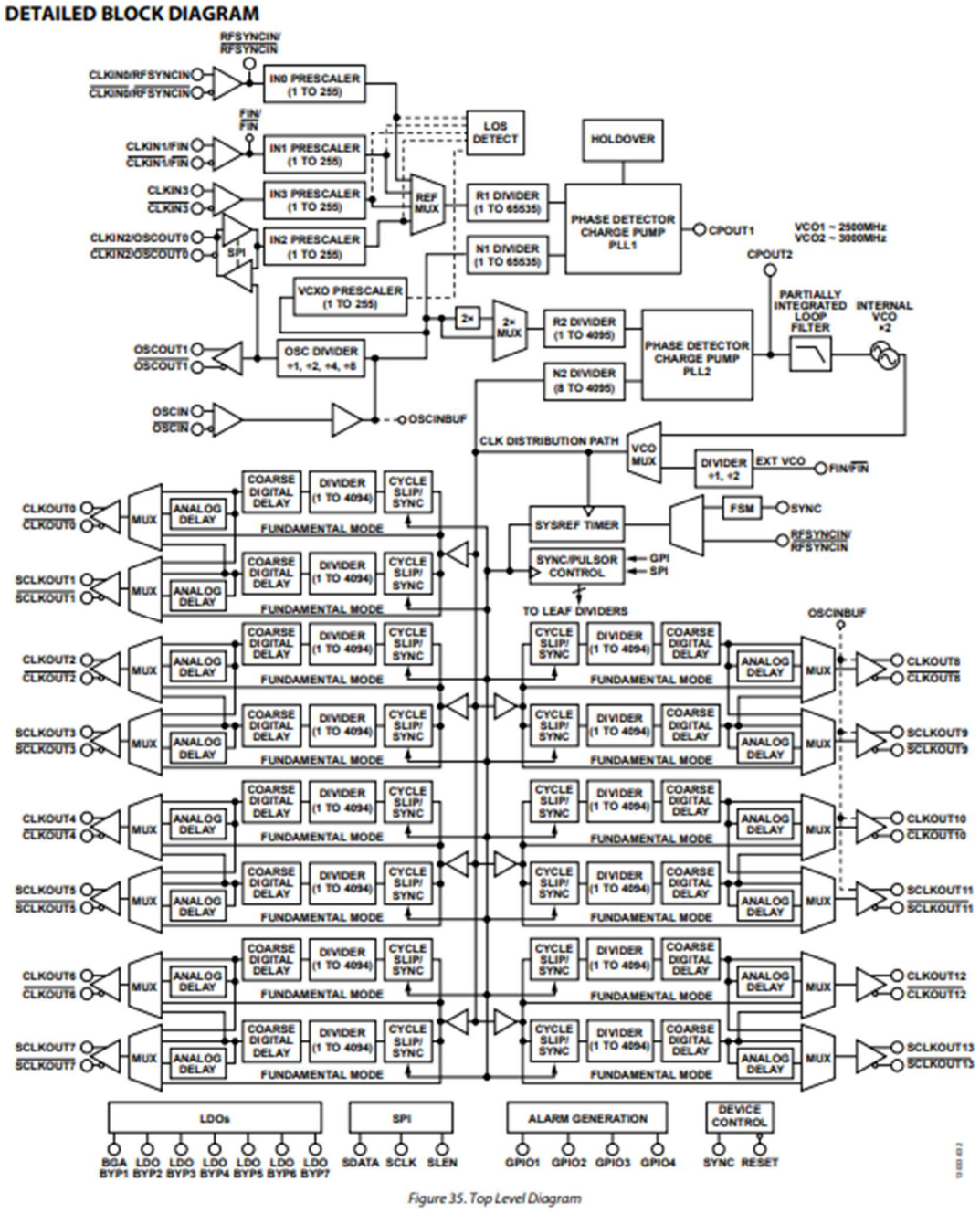

Figure 35. Top Level Diagram

Figure 11.1: Detailed internal architecture of the HMC7044

Fig 11.1 shows the internal architecture, there we see that the IC can handle 4 reference inputs and these inputs can be programmed to change their functionality. One of the references can be fed with a noisy reference clock and this clock will lock the first PLL. This is where the jitter cleaner action will take place. Now, the first PLL will output a relatively better (less noisy) clock to the second PLL loop that will generate the 14 outputs.

The outputs will startup at random phase and must be synced with an external SYNC signal or a RFSYNC.

The output PLL has the flexibility to choose two VCOs, one with higher frequency and the other with a lower frequency. This VCO, after it has locked to the cleaned reference from the first PLL, will provide outputs through the clock dividers.

An external RFSYNC or SYNC signal will reset the dividers and synchronize all the 14 outputs. This is very useful in synchronizing JESD and expansion to larger systems.

The outputs can be configured in Asynchronous or Dynamic mode. In the Asynchronous mode, the outputs startup in random phase and must be synced through a rising edge in the SYNC pin or an external RFSYNC through CLKIN0. In the Dynamic mode of operation, the outputs will start only when a SYSREF request is applied. The outputs configured in the dynamic mode will output a specified number of pulses.

The next section will talk about the results and outputs of the HMC7044 through an evaluation board.

Chapter 12: HMC7044 Eval Board Results

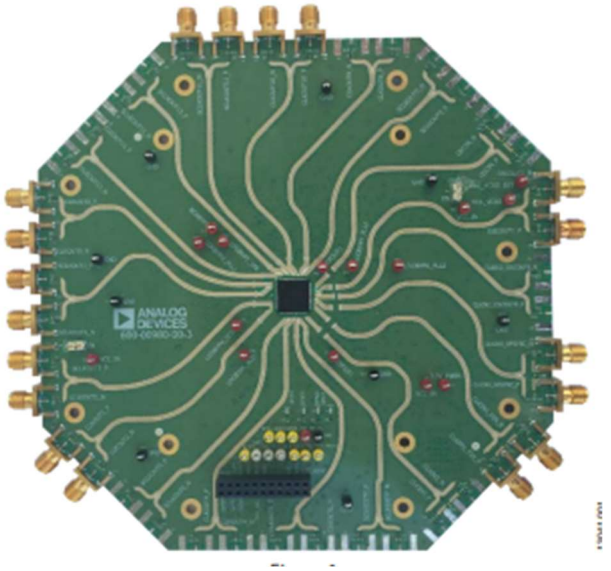

Figure 12.1: HMC7044 Eval Board

The Eval Board has a 122.88 MHz Oscillator that acts as the VCXO for the first PLL. One of the reference inputs must be fed with a frequency which is a sub-multiple of 122.88 MHz. As discussed in the previous section, the first PLL will lock and provide a reference to the second PLL for generation of the outputs.

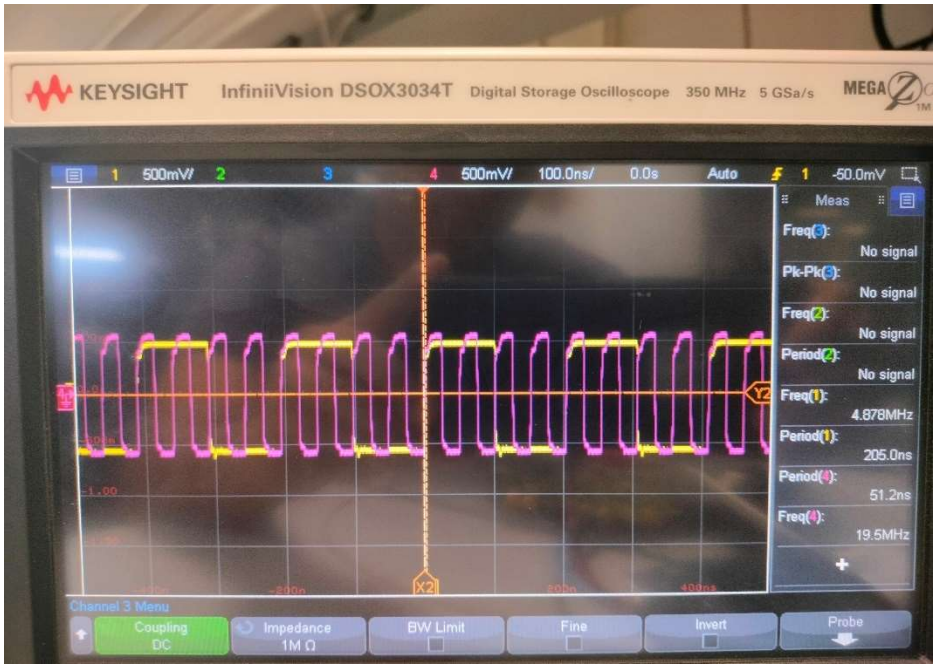

Figure 12.2: Two channel outputs running at different frequencies of the HMC7044

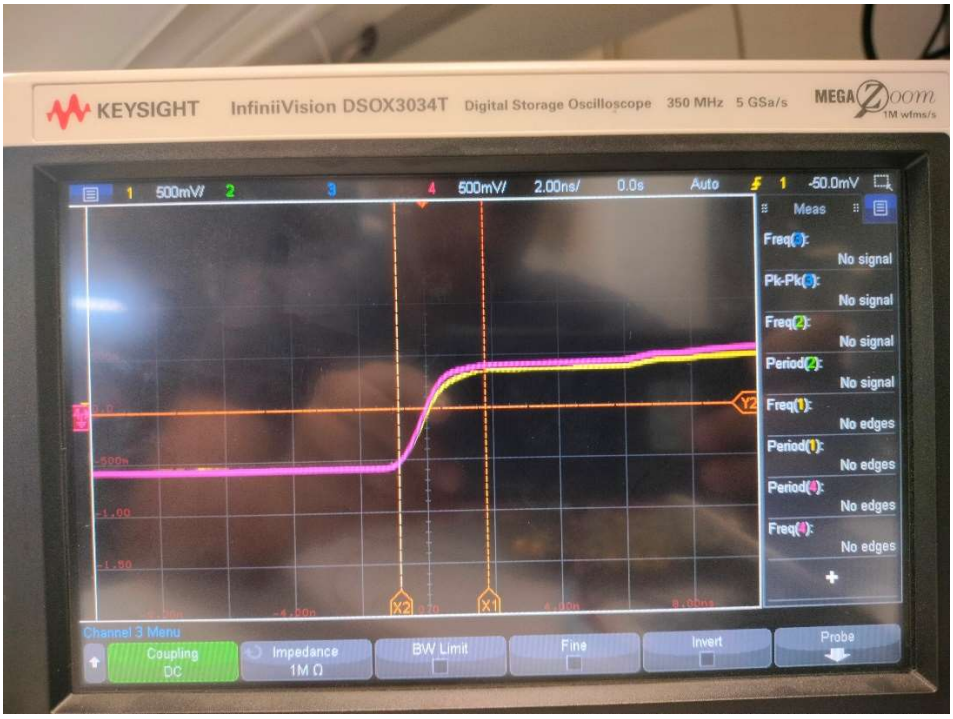

Figure 12.3: Synchronized Outputs of the HMC7044 (Overlapping and Synced rising edges)

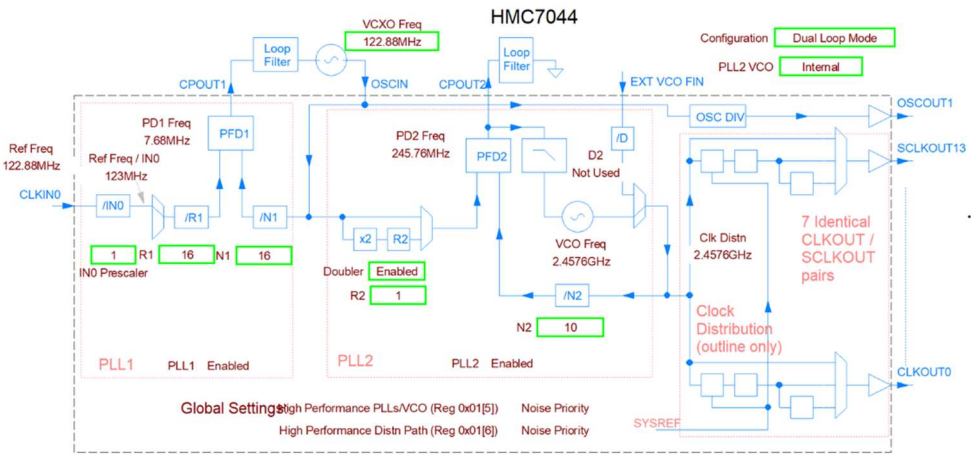

Figure 12.4: An example configuration for the HMC7044

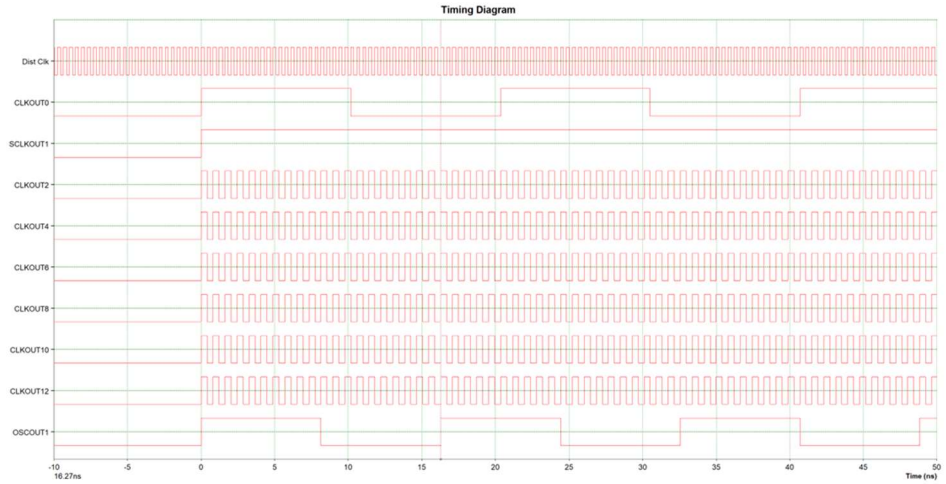

Figure 12.5: Simulated Ouputs of the HMC7044 in the ADIsimCLK software.

This explains the typical architecture and working of an PLL based clocking IC. The next section will explore the synthesis of a clock tree to sync multiple RF systems to a master clock using the LTC6952.

Chapter 13: Analog Devices LTC6952 Clocking IC

The LTC6952 is a high performance, ultralow jitter, JESD204B/C clock generation and distribution IC. It includes a Phase Locked Loop (PLL) core, consisting of a reference divider, phase-frequency detector (PFD) with a phase-lock indicator, ultralow noise charge pump and integer feedback divider. The LTC6952's eleven outputs can be configured as up to five JESD204B/C subclass 1 device clock/SYSREF pairs plus one general purpose output, or simply eleven general purpose clock outputs for non-JESD204B/C applications. Each output has its own individually programmable frequency divider and output driver. All outputs can also be synchronized and set to precise phase alignment using individual coarse half-cycle digital delays and fine analog time delays. For applications requiring more than eleven total outputs, multiple LTC6952s can be connected using the EZSync or ParallelSync synchronization protocols.

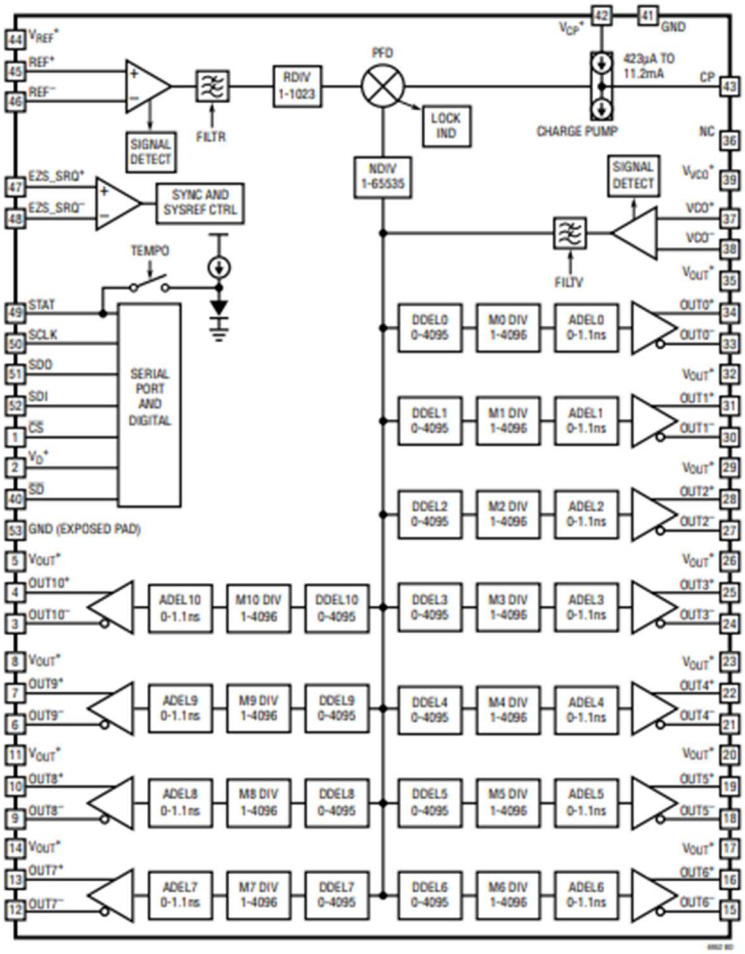

Figure 13.1: The internal architecture of the LTC6952, Credits: LTC6952 Datasheet

The IC has the feature of both Digital and Analog Delays, along with the output dividers to achive a desired frequency that must be an integer sub-multiple of the PLL VCO output.

This chip has been used to synthesize a clock tree to synchronize two slave eval boards to a master eval board.

13.1 Parallel SYNC based Clock tree synthesis:

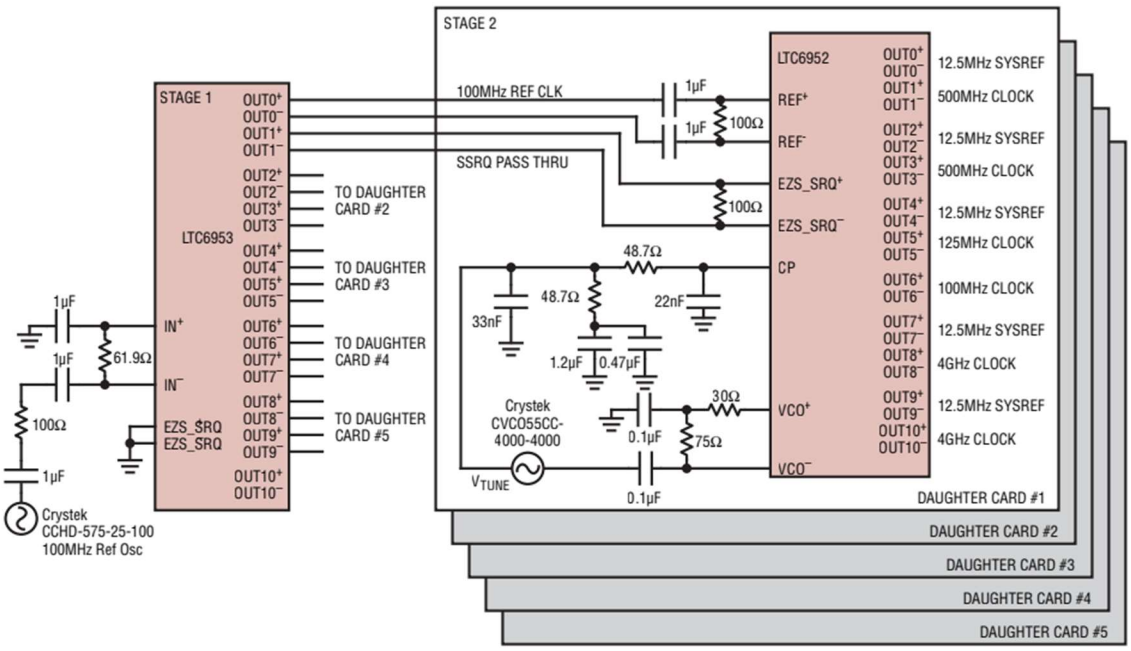

Figure 13.1.1: Parallel Sync Setup for Multi-Chip sync.

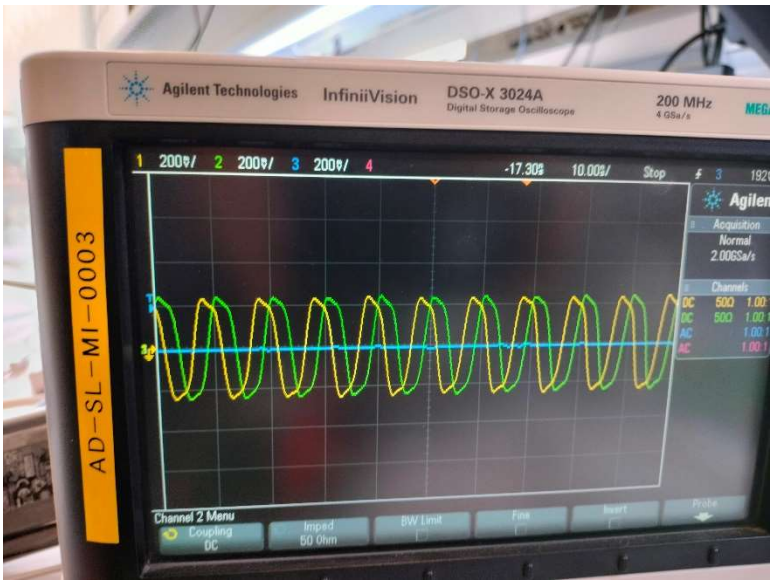

Figure 13.1.2: Outputs of LTC6952 from Slaves Before SYNC, Yellow is from Slave 1 and Green is from Slave 2.

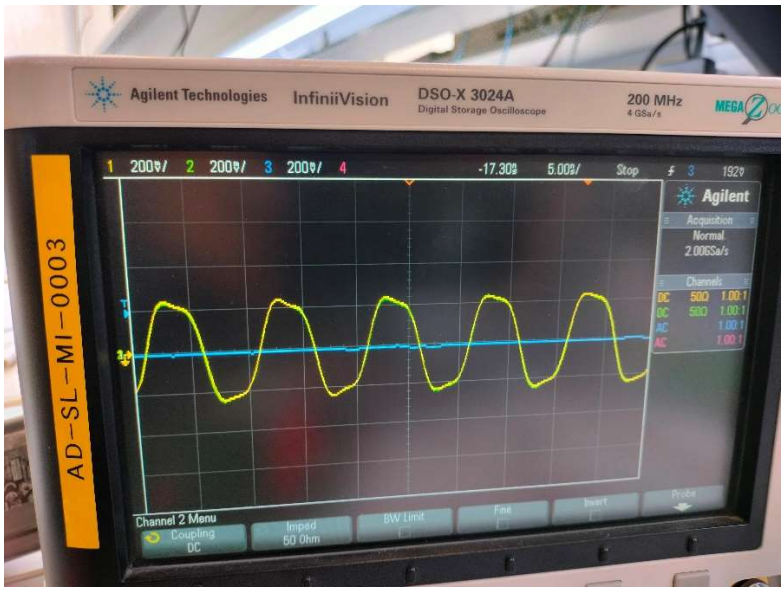

Figure 13.1.3: Overlapping SYNCed Outputs from both slaves.

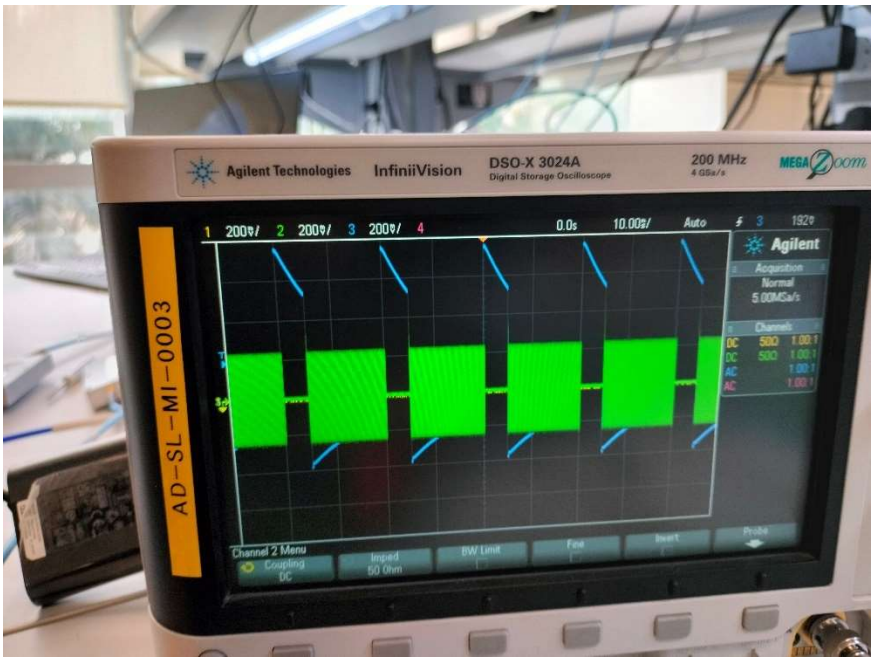

Figure 13.1.4: The SYNC signal and the overlapped outputs.

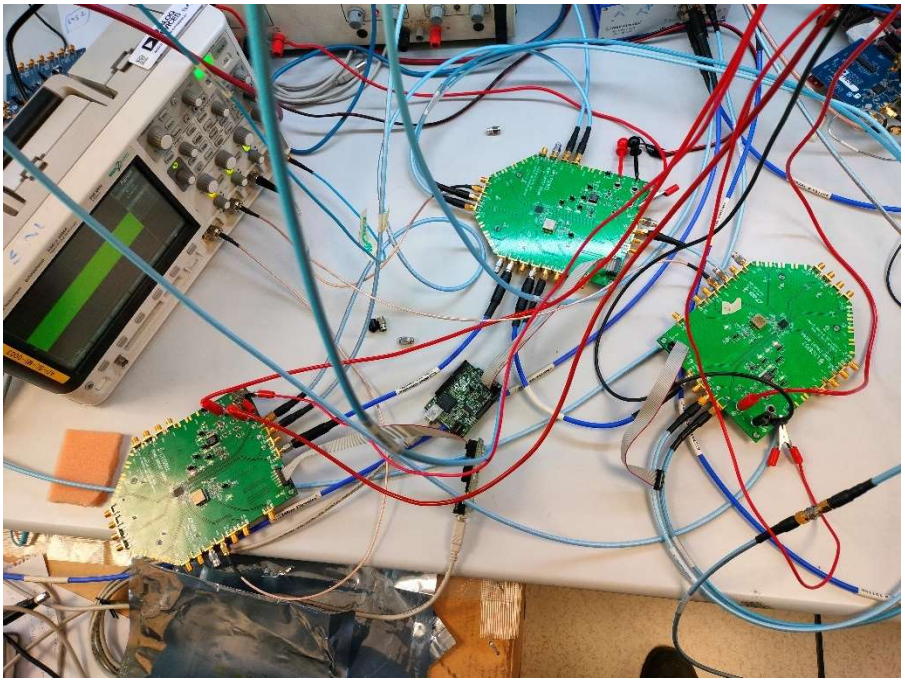

Figure 13.1.4: Clock Tree Lab Setup.

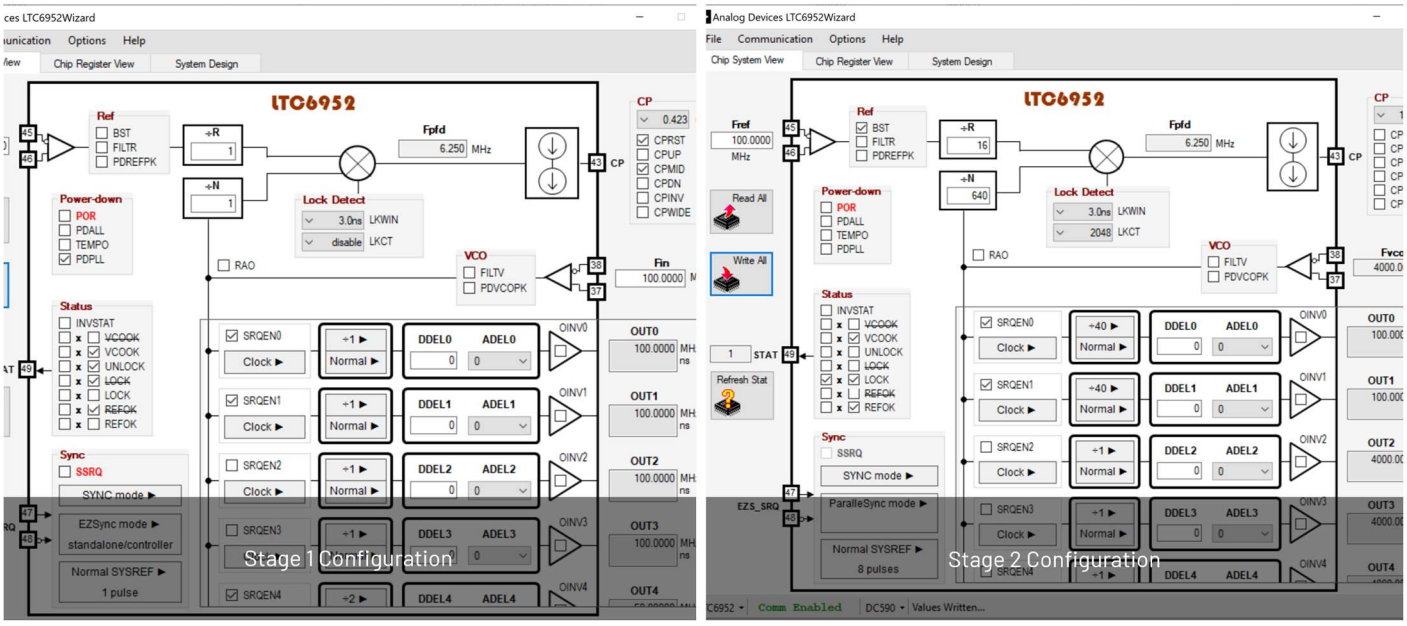

Figure 13.1.5: Configurations of Master (Stage 1) and Slave (Stage 2)

Figure 13.1.6: Phase Noise Measurements at 100 MHz

Figure 13.1.7: Phase Noise Measurements at 4 GHz with respect to Signal Generator at 4GHz.

Figure 13.1.8: Phase Noise Measurements at 400 MHz.

Chapter 14: Analog Devices Quad MxFE

The Quad-MxFE System Development Platform contains four MxFEs, direct RF sampling transceivers, as well as RF front ends, clocking, & circuitry taking care of power supplies. This finds application phased array radars, electronic warfare, and ground-based SATCOM, a 16 transmit/16 receive channel direct RF sampling phased array at L/S/C bands (0.1 GHz to ~5GHz).

The Quad MxFE has displayed multi-chip synchronization, beam-forming, direct RF sampling, band-selection through NCOs, down-conversion through DDCs and Up-conversion through DUCs .

The package also includes a 16Tx/16Rx Calibration Board, which can be used to create system-level calibration algorithms or to demonstrate power-up phase determinism more readily in circumstances relevant to the user's use case. When linked to the PMOD interface of the VCU118, the calibration board also enables the user to show combined-channel dynamic range, spurious, and phase noise improvements. It can also be operated via a free MATLAB add-on.The system can be used to enable quick time-to-market development programs for applications like:

- ADEF (Phased-Array, RADAR, EW, SATCOM)
- Communications Infrastructure (Multiband 5G and mmWave 5G)
- Electronic Test and Measurement

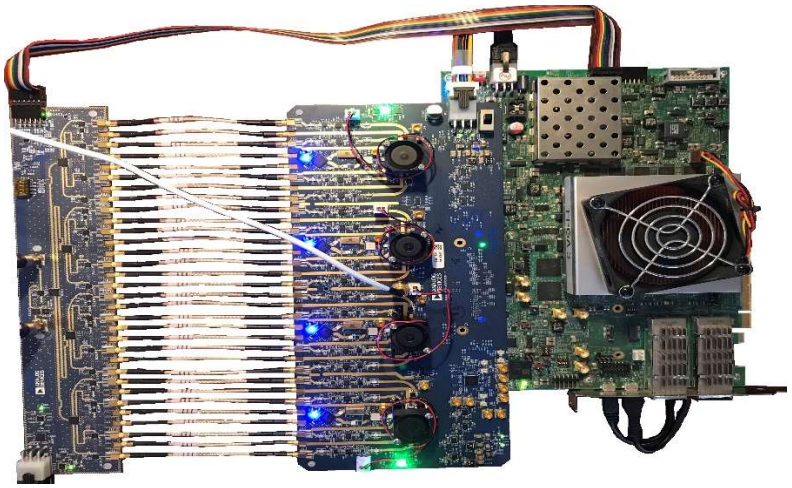

Figure 14.1: Quad MxFE AD9081 Setup

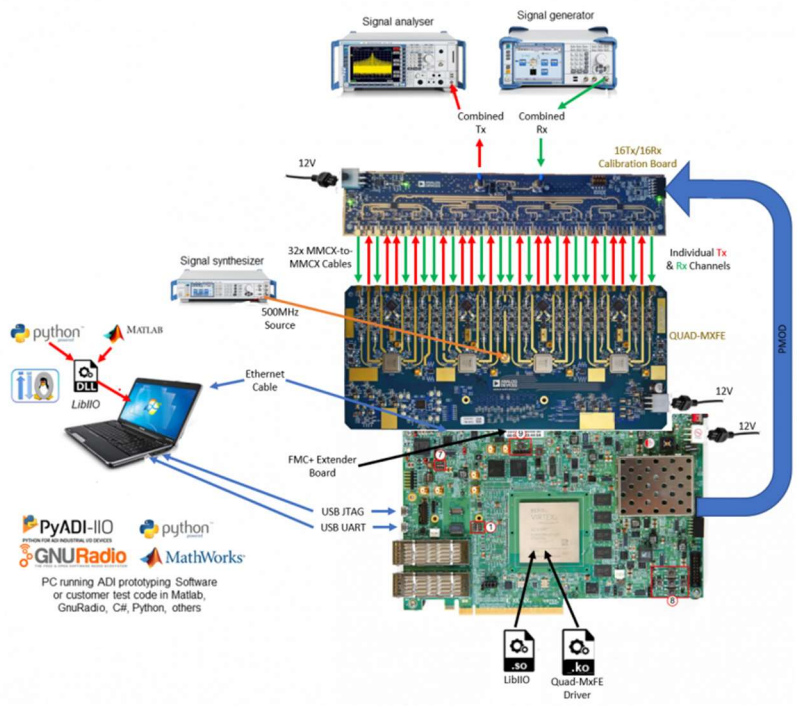

Figure 14.2: Setup to run the Quad MxFE

The 500 MHz will be the reference signal to all the PLLs of the 4 AD9081 IC. The calibration board is used to test the scripts and show calibration in the Rx and Tx paths.

14.1: Multi-Chip Synchronization on the Quad MxFE and the Calibration Board.

It is possible to synchronize multiple AD9081s to cater to needs of large radar and communication systems requiring multiple ADCs and DACs. The process of multi-chip synchronization involves two steps:

A. One Shot SYNC – This synchronizes the baseband data and the internal digital clocks. This will ensure sync of receive and transmit paths if the NCOs are bypassed.
B. Master Slave NCO SYNC – To synchronize multiple NCOs in different ICs, must first perform the above-mentioned One-Shot sync procedure to align all the clocks. Next, one of the ICs will act as master where the rising edge of an internal SYNC signal will trigger the ADC and DAC reset algorithm of Rx and Tx paths. One GPIO pin from the master will be routed to the slaves which will also be synced when the master GPIO responds through this GPIO.

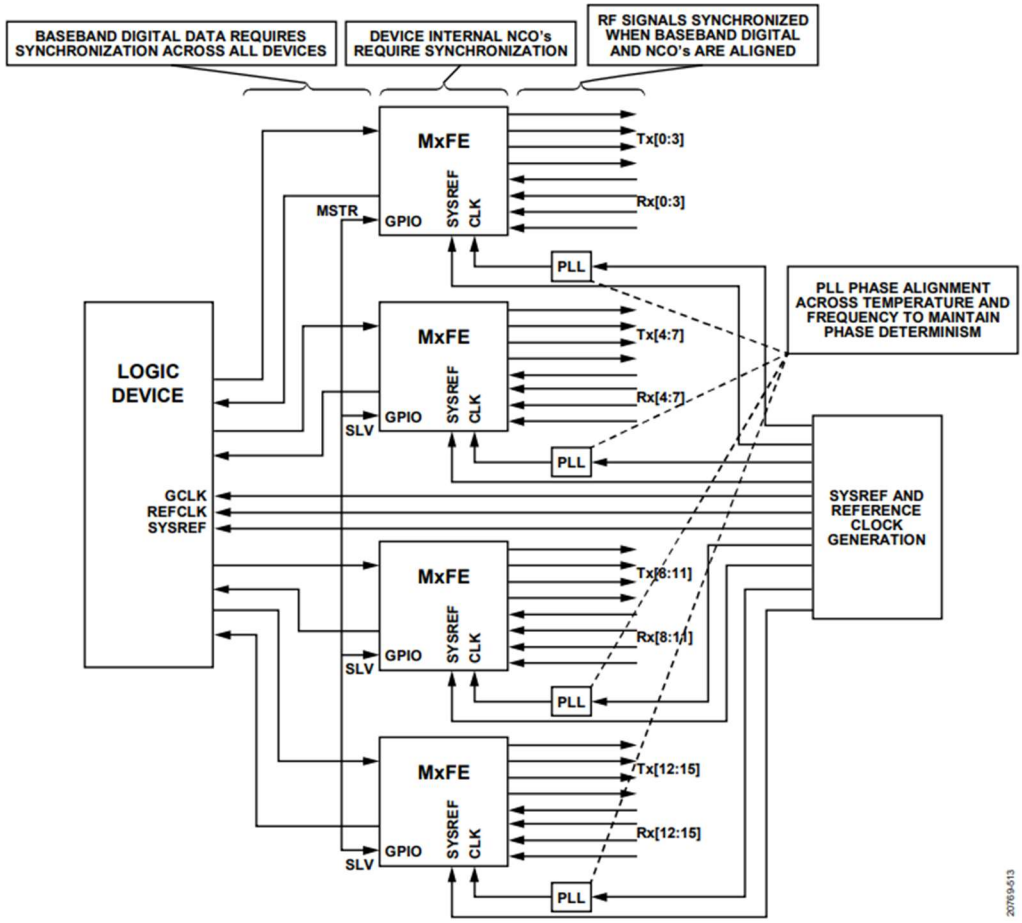

Figure 14.3: AD9081 Quad MxFE High Level MCS diagram, Credits: User Guide, Analog Devices

Figure 14.3 shows the diagram to achieve multi-chip synchronization, The SYSREF and clock generation IC (HMC7043) distributes the reference clock to all the PLLs of the AD9081s. The distribution IC also provides FPGA and SYSREF clocks. The HMC7043 has syncing capabilities that phase aligns all the ouputs. This aligns all the baseband communication clocks.

The 'MSTR' stands for master and this GPIO is configured to be an output and is eventually fed into the slaves. On the rising edge of an internal synchronizing signal, the NCOs go through a reset cycle.

14.2: PLL Synthesizer for Temperature Compensation:

PLL Phase drifts due to temperature variations are taken care of through a feedback mechanism in which the first transmit channel is phase aligned with the first transmit channel of the first MxFE. The four transmit channel data is then simultaneously read through one receive channel of the first MxFE. Then by applying cross-correlation the required phase offsets are determined and applied to the PLL phases. This method ensures same baseline for the first transmit channels of all the MxFEs.

14.3: System Level Calibration Algorithm:

After performing the MCS process, deterministic phase will be achieved between all the Rx and Tx channels of the Quad MxFE. In this method, all the transmit channels transmit a pulse one-after the other. All the Tx pulses are combined in the Calibration board and then fed to all the Rx channels.

Then, the first, Tx0 pulse is located and after that all the pulses subsequently. The delays (phases) are then stored in a 1*16 array. Similarly, the data is horizontally observed to determine the delays of the Tx0 in all the other channels with respect to Rx0. The data is stored in a 1*16 array and then is applied to the complex NCO phases, thus achieving total transmit and receive sync. This method is very helpful when there are different delays of clock due to different PCB trace delays.

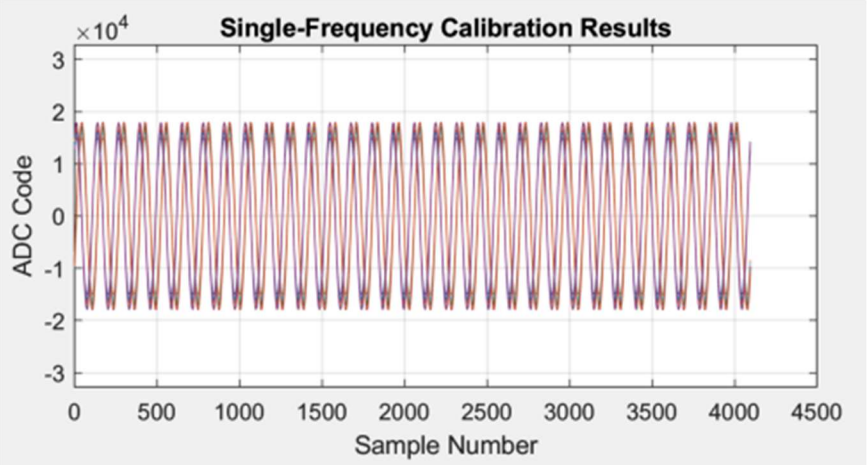

Figure 14.3: IQ Overlap in 16 Channels

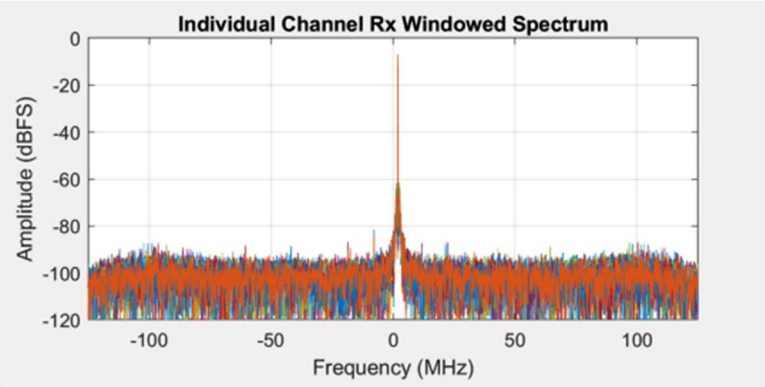

Figure 14.4: Individual Spectrum of Rx channels

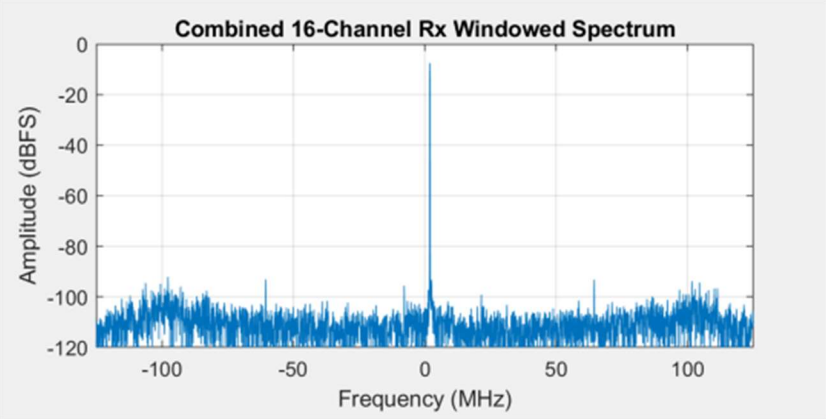

Figure 14.5: Combined 16 channel – Lowered Noise Floor

Figures 14.3 to 14.5 show the Multi-Chip Synchronization results in the lab with the Quad MxFE. Figure 14.3 shows the overlap of I and Q from all 16 channels, figure 14.4 shows the spectrum from a single channel and figure 14.5 shows the combined spectrum of all 16 channels. This shows the lowering of the noise floor.

The scripts run in MATLAB implement the MCS algorithm as discussed previously in section 14.1.

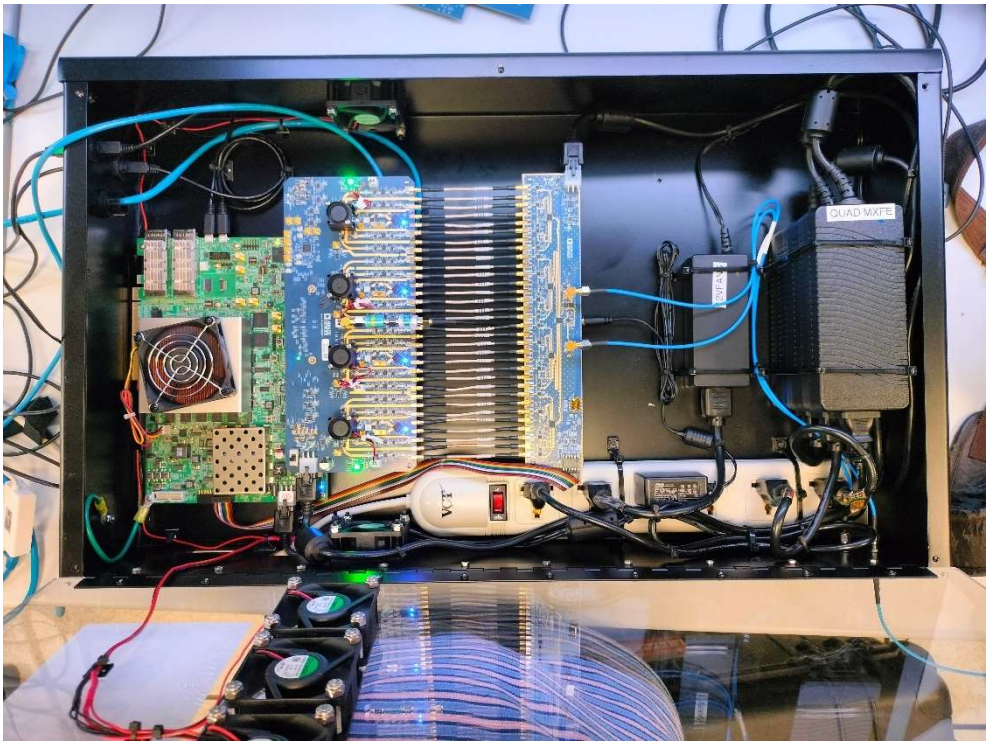

Figure 14.6: Ruggedized Quad MxFE Setup with the VCU118.

Figure 14.6: shows the entire setup in the lab. The setup has been ruggedized for field applications.

Chapter 15: Conclusion and Future Developments

This thesis presents solution for state-of-the-art RF solutions for communications, defense, and UAV applications. The work first explores a solution with the Red Pitaya 125-10 and Raspberry Pi for on-board signal acquisition and processing on drones, UAVs, and unmanned/remote locations. The Red Pitaya comes with an Analog Devices 9608 ADC. In this implementation the Raspberry Pi acts as the On-Board computer, on this the ability to process signals in real-time has been shown. The signals acquired by the Red Pitaya is sent to the OBC (On-Board Computer) for processing. This enables us to implement filters to suppress unwanted frequencies before sending the data back to the base station. This will find extensive applications in remote spectrum monitoring, intelligent electronic warfare systems, real-time processing applications on the fly and much more. As an example, the work has been tested in the frequency range used by the Search and Rescue Satellite (SARSAT) at 406 MHz. The work has also been demonstrated to the military and have found areas in their use cases for which this is a perfect fit.

The next stage of the thesis involves developing the work with the Red Pitaya and Raspberry Pi to high end mixed signal front end ICs with Analog Devices. The AD9081 IC is a Mixed Signal Front End (MxFE) that houses ADCs, DACs, DSP applications and other components of the signal chain. The receive and transmit paths have NCOs (fine and coarse), gain adjusts, multiple JESD204x modes to enable multitude of use cases and sampling rates. The IC has been interfaced with a Xilinx UltraScale+ FPGA that has the JESD IP in the Verilog code itself. This part of the thesis explores implementing signal processing algorithms, in real-time, on an FPGA.

Next up to implement larger systems, we need to achieve synchronized clocks so that all the systems. This will be the heart of the multi-chip synchronization that is implemented in the Quad MxFE. For this the LTC6952 from Analog Devices has been used. The thesis shows the synchronization of three LTC6952 Eval Boards in master slave configuration. This shows the synchronization of the outputs of two slaves which see synced reference clocks from the master. In addition, the HMC7044 and its working has also been shown as a part of this work. The HMC7044 is a dual PLL jitter attenuator clocking IC.

Finally, the thesis turns towards the Quad MxFE which utilizes all the work discussed before and establishes the multi-chip synchronization feature of the AD9081 to show synchronization between 4 AD9081 ICs. The catch of this part is that the entire setup is based on a single baseband processor – the VCU118 FPGA. This will be very useful in applications where multiple systems are integrated and must be synchronized.

The work presented in this thesis is highly relevant to today's needs of accurate, low-noise, fast communication, military, warfare, space applications. This can be implemented for Radars, SatCom and 5G applications. The phase changing abilities in both Tx & Rx datapaths enable this work to be implemented in beam-steering applications. The ability to change the phases in real-time allows this be flown in a dynamic system like an airplane, satellite, or drone. The present roll-pitch-yaw information from an external IMU can be used to steer beams in real time from a remote base station.

Another important electronic application in electronic warfare is fast frequency hopping that allows the carrier frequency to 'hop' around different values making it impossible for enemy to intercept the frequency and band of communication. It can also be used to acquire RF signals and perform modulation classification by feeding the captured IQ samples to a pre-trained Deep Learning model.

REFERENCES

1. Mitrovic, Andrija, Slobodan Djukanovic, and Milutin Radonjic," Implementation of signal classification using frequency spectrum features on the Raspberry Pi platform", Telecommunication Forum, pp. 1-4, 2017.
2. JPatange,P. Pranjali, and John Sahaya Rani Alex. " Implementation of ANN based speech recognition system on an embedded board", International Conference Nextgen Electronic Technologies: Silicon to Software (ICNETS2), , pp. 408-412, 2017.
3. A. Deleforge, D. Di Carlo, M. Strauss, R. Serizel and L. Marcenaro, "Audio-Based Search and Rescue with a Drone: Highlights from the IEEE Signal Processing Cup 2019 Student Competition [SP Competitions]," in IEEE Signal Processing Magazine, vol. 36, no. 5, pp. 138-144, Sept. 2019, doi: 10.1109/MSP.2019.2924687.
4. Analog Devices (1995), https://www.analog.com/en/design-center/evaluation-hardware-and-software/evaluation-boards-kits/quad-mxfe.html#eb-documentation
5. Rabiye Kılıç, Nida Kumbasar, Emin Argun Oral, Ibrahim Yucel Ozbek, Drone classification using RF signal based spectral features, Engineering Science and Technology, an International Journal, Volume 28, 2022, 101028, ISSN 2215-0986, https://doi.org/10.1016/j.jestch.2021.06.008
6. Zdunek, Rafal, Wu, Peng, Sun, Bei, Su, Shaojing, Wei, Junyu, Zhao, Jinhui, Wen, Xudong, 2020, 2020/11/27, Automatic Modulation Classification Based on Deep Learning for Software-Defined Radio, https://doi.org/10.1155/2020/2678310, doi: 10.1155/2020/2678310, Mathematical Problems in Engineering, Hindawi
7. EverythingRF, https://www.everythingrf.com/community/what-is-phased-array-antenna
8. Red Pitaya, https://redpitaya.com/stemlab-125-10/
9. Mini – Circuits, ZEM 4300MH+, https://www.minicircuits.com/pdfs/ZEM-4300MH+.pdf.
10. SARSAT, www.sarsat.noaa.gov.in.
11. https://wiki.analog.com/resources/eval/user-guides/quadmxfe/quickbringup

12. https://wiki.analog.com/resources/eval/user-guides/quadmxfe
13. https://wiki.analog.com/resources/eval/user-guides/quadmxfe/multichipsynchronization
14. https://wiki.analog.com/resources/eval/user-guides/quadmxfe/boardhardwaredetails
15. https://www.analog.com/media/en/technical-documentation/application-notes/47535474114416530477570974069213146 1831AN823_0.pdf
16. https://www.analog.com/media/en/training-seminars/tutorials/mt-017.pdf
17. https://www.analog.com/media/en/technical-documentation/data-sheets/ad9081.pdf
18. https://www.analog.com/media/en/technical-documentation/user-guides/ad9081-ad9082-ug-1578.pdf
19. https://www.analog.com/media/en/technical-documentation/user-guides/eval-ad9081-9082-9986-9988-ug-1829.pdf
20. https://www.analog.com/en/products/ad9081.html#product-overview