Addis Ababa University

Addis Ababa Institute of Technology

School of Electrical and Computer Engineering

Performance Comparison of Multi-Mode Modulation Techniques for SDR Using FPGA

By

Sisay Bogale

Adviser: Dr. -Ing. Yihenew Wondie

A thesis submitted to the School of Electrical and Computer Engineering in Partial Fulfillment of the requirements for the Degree of Masters of Science in Communication Engineering.

November, 2023

Addis Ababa University

Addis Ababa Institute of Technology

School of Electrical and Computer Engineering

Performance Comparison of Multi-Mode Modulation Techniques for SDR Using FPGA

By

Sisay Bogale

Approved by Board of Examiners

_____	_____
Chair Person	Signature
Dr.-Ing. Yihenew Wondie	_____
Adviser	Signature
_____	_____
Examiner	Signature
_____	_____
Examiner	Signature

Declaration

In accordance with globally recognized standards, I, the one who signs this document, certify that the thesis is an original work of mine, and I have appropriately acknowledged and cited all sources consulted during the writing process.

Sisay Bogale _____

Name

_____

Signature

Place: **Addis Ababa University, Ethiopia**

Date of Submission: ----------

This thesis has been submitted for examination with my approval as a university advisor.

Dr.-Ing. Yihenew Wondie _____

Adviser Signature

Abstract

Radio devices that were previously built in hardware have been replaced in recent years by reconfigurable software defined radio (SDR) systems. Conventional hardware-based radios have restricted multi-functionality and are physically changeable only. This leads to an increase in production expenses and a reduction in the number of waveform standards that can be supported. A rapid and affordable answer to this issue is provided by software-defined radio technology, which enables software upgrades for multi-mode, multi-band, and multi-functional wireless devices. In SDR, different modulation techniques are used to achieve efficient communication over a radio channel. Multi-mode modulation is an approach that allows the use of multiple modulation schemes in a single system, which can enhance the flexibility and resilience of communication systems. This paper presented a design and implementation of multi-mode modulation techniques for SDR using FPGA and analyze the performance based on the FPGA resource utilization. It combines six modulation schemes: QASK, QPSK, QAM, AM, PM and FM to create multi-mode modulation system. The performance of this multi-mode modulation system is evaluated in terms of FPGA resource utilization such as total computational power, total number of Look Table (LUT) or memory used, Flip Flops (FF) and Input/Output (IO) port usage. Xilinx Vivado system generator for DSP with MATLAB/Simulink is used to design, simulate and verify the multi-mode modulator, which would then be implemented on a Xilinx Zedboard FPGA hardware. A total of 0.225W power, 844 number of LUT and 1 IO port is utilized by the implemented design. The biggest thing we achieved in this research is that we saved computational power. 1.572W and 1.134W amount of power is saved by our design as compared to previous two studies.

Keywords: *FPGA, AM, FM, PM, 4ASK, QPSK, 4QAM, SDR, Multi-mode modulation*

Acknowledgements

I owe a debt of gratitude to Dr.-Ing. Yihenew Wondie, my master's program advisor, for all of his help and advice. Their knowledge and tolerance have been really helpful to me and have been essential to the completion of this thesis.

I am appreciative of Information Network Security Administration (INSA) for giving me the chance to work on my master's program and my research, as well as for all of the tools and assistance they offered.

Additionally, I would like to thank Bethlehem Seifiu, a PhD candidate, Dr. Tsegamlak Terefe, and Dr. Dereje Hailemarim for serving on my thesis committee and offering insightful criticism and recommendations.

I sincerely appreciate my wife, Mss. Haymanot Fissha, and my aunt, Mss. Tsehay Mindahun, for their love and support throughout this journey.

Lastly, I just want to say how much I appreciate each and every one of the study volunteers. They have been incredibly helpful to my study and have contributed to the success of this thesis with their readiness to share their experiences and insights. I appreciate all of your help and attention. I am appreciative of everyone who has helped me along the way.

Table of Contents

List of Tables

List of Figures

Acronyms

AM	Amplitude Modulation
ADC	Analog to Digital Converter
ASIC	Application Specific Integrated Circuit
BPSK	Binary Phase shift keying
CAD	Computer Aided Design
CLB	Configurable Logic Blocks
DAC	Digital to Analog Converter
DDC	Digital Down Converter
DDR	Dynamic Random-Access
DDS	Direct Digital Synthesizer
DSP	Digital Signal Processor
DUC	Digital Up Converter
FM	Frequency Modulation
FF	Flip Flop
FPGA	Field Programmable Gate Array
FSK	Frequency Shift Keying
GPP	General Purpose Processors
HDL	Hardware Description Language
IO	Input/Output
IP	Intellectual Property
LUT	Lookup Table
NCO	Numerically Controlled Oscillator
NGC	Native Generic Circuit
PM	Phase Modulation

QPSK	Quadrature Phase Shift Keying
QAM	Quadrature Amplitude Modulation
RF	Radio Frequency
SDR	Software Defined Radio
SFDR	Spurious Free Dynamic Range
SoC	System on Chip
JTAG	Joint Test Action Group
VHDL	Very High Speed Integrated Circuit Hardware Description Language
XST	Xilinx Synthesis Technology

Chapter 1: Introduction

1.1 Introduction

With the introduction of cellular communications twenty years ago, the use of wireless communication networks has grown. Hardware flexibility is now necessary due to improvements in wireless technologies and applications [1]. Developing new radios in response to evolving wireless standards and applications is highly costly and time-consuming. By transferring analog circuit components into the digital realm, software-defined radio can provide answers to these issues [1]. Logic devices that can be programmed, such that FPGAs, can be used to perform radio functionalities. [1].

The SDR is a design paradigm for wireless communications devices. The term's originator, Joseph Mitola, established it as a designation for a type of radios that could be software-reprogrammed and reconfigured in the early 1990s [8]. An antenna and an analog to digital converter (ADC) on the receiving end were the sole physical components of Mitola's ideal Software Defined Radio. The transmitter would also feature a transmitting antenna in addition to a Digital to Analog Converter (DAC). Reprogrammable Processors would take care of the remaining tasks [8].

The recent advances in FPGA and processor technology have led to a surge in interest in Software Defined Radio, or SDR. Functions that were previously developed utilizing analog hardware components can now be inexpensively implemented thanks to FPGAs [1]. FPGAs are composed of a matrix of programmable connections connecting a range of programmable logic cells. A designer's CAD tool defines a basic logic function that each cell can implement. These cells can be utilized to create intricate digital circuits, and a typical programmable circuit contains 64–300,000 of them. The designer can create a custom processor to effectively accomplish the intended function by manipulating the logic at the gate level. Generally, designs aimed at FPGAs are expressed by Hardware Description Languages, or HDLs [9].

The process of varying a carrier wave's characteristics—typically its amplitude, frequency, or phase— according to the immediate value of the modulating signal is known as modulation. Modulators are the systems that carry out modulation and are regarded as the most important component of any communication system. These modulators are widely employed in communication systems for a wide range of applications, including mobile, undersea, radar, aerospace, naval/maritime, and underground communication. Among the most fundamental and widely used modulation techniques are AM, FM, ASK, FSK, PSK, and QAM [2].

All modulation and demodulation in conventional designs takes place in the analog domain. In recent years, digital radio systems have supplanted analog radio systems for a variety of radio applications in commercial, military, and civilian industries. Furthermore, digital radio systems are using programmable hardware modules at various functional levels on an increasing basis [1].

Analog and digital modulators have previously been developed and created using Xilinx hardware and software resources. In those works, the modulators are implemented as a single modulator or as a reconfigurable module. Using a control signal, the reconfigurable modulator selects each modulation type one by one. In this thesis, a digital and analog modulator will be designed and developed as a single module with six inputs and outputs. These modulators will be developed and implemented using the Vivado system generator software and the Xilinx Zedboard FPGA. The multi-mode modulation system diagram is shown in Figures 1.1 and 1.2. This thesis focus on the baseband section of the SDR system.

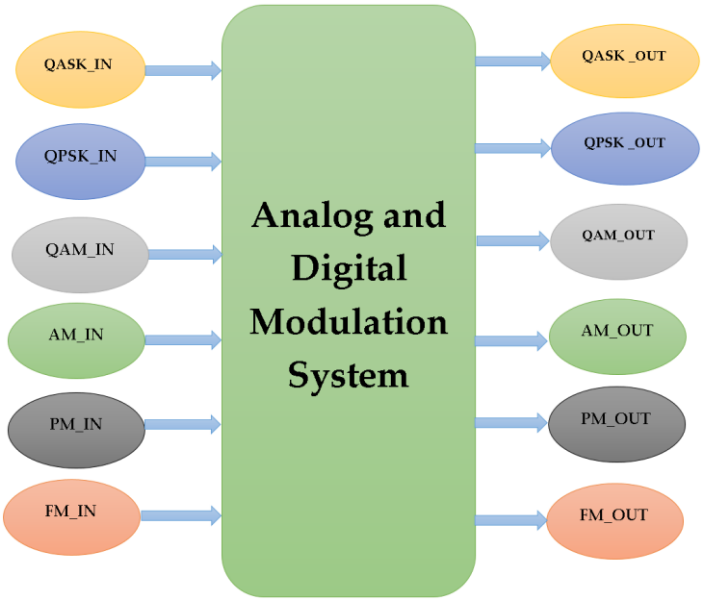

Figure 1.1: Analog and Digital Modulation Systems Block [4].

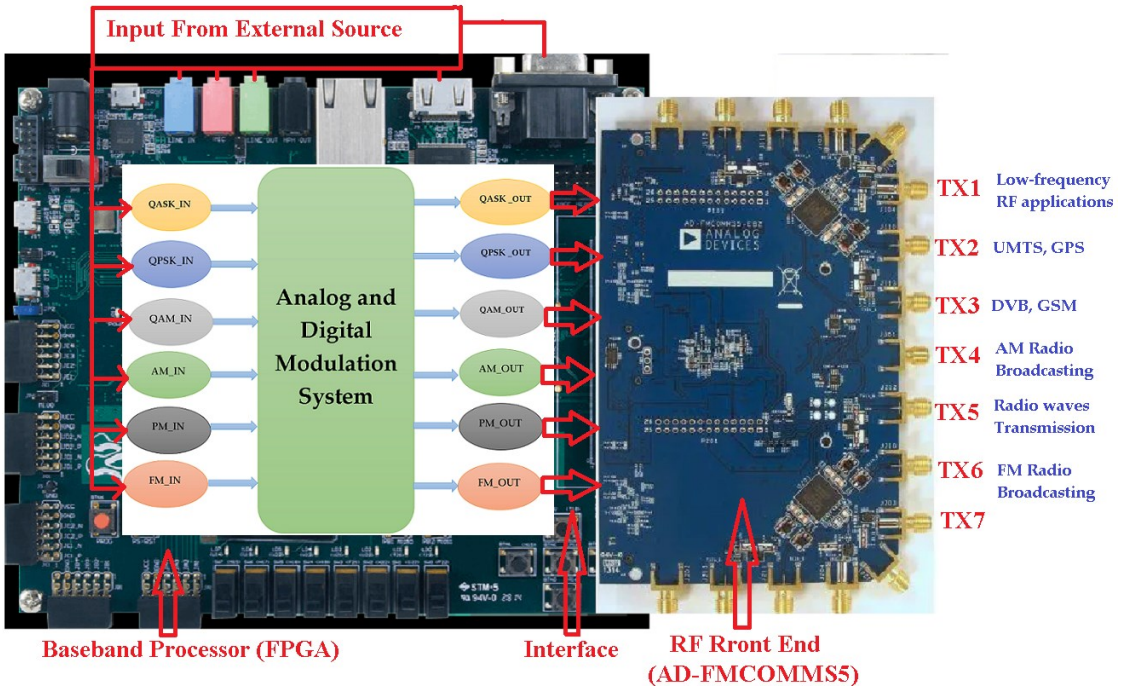

Figure 1.2: Multi-mode modulator with Applications [11].

AM, FM, and PM modulators will be built in Vivado system generator utilizing Simulink blocks produced by Xilinx Company for the analog modulation section. The

designed modulation system is first validated in simulation before being implemented on the Zedboard FPGA.

In the digital modulation system QPSK, QASK and 4QAM modulators will be created in Vivado a system generator using Simulink block which is prepared by Xilinx Company. The designed modulation scheme is then validated through simulation before being implemented on the Zedboard FPGA.

The multi-mode modulators that have been developed will be used in a variety of wireless communication applications. The applications of the implemented modulators are summarized in Table 1.1 and Table 1.2.

Table 1.1: Applications of digital modulators [12].

Modulation Type	**Applications**
QPSK	UMTS, GPS, CDMA
4QAM	DVB, GSM
QASK	Used in low-frequency RF applications

Table 1.2: Applications of Analog modulators [13].

Modulation Type	**Applications**
AM	AM radio broadcasting
FM	FM radio broadcasting
PM	For transmitting radio waves

1.2 Background

SDR is a versatile technology that allows digital signal processing (DSP) algorithms to be implemented in software instead of hardware. SDR enables reconfigurable radios, where the same hardware could be used to implement different wireless communication standards, protocols, and waveform types. Multi-mode modulation techniques are a subset of SDR techniques that allow multiple modulation schemes to be implemented for flexible communication. Multi-mode modulation techniques could be a better fit for applications that require continuous modulation adaptation, such as cognitive radio, adaptive communication systems, and hybrid communication networks [27].

FPGAs are often used as the hardware platform for SDR because of their high-performance computing capabilities, re-configurability, and low-power consumption. FPGAs enable DSP algorithms to be executed in parallel, which speed up the processing time and reduce the latency of the system [28].

This research aims to combine six modulation techniques to create multi-mode modulation system and compare with existing researches using FPGA as a hardware platform. The aim is to evaluate the performance of the multi-mode modulation system in terms of computational power, total number of Look Table (LUT), Flip Flops (FF) and Input/Output (IO) port usage. The comparison will be based on simulations and measurements using a real-time FPGA implementation of the different modulation schemes. This research will help for wireless communications applications that require multi-mode communication.

1.3 Statement of the Problem

Modifying DSP hardware components easily and affordably has become essential for businesses. Traditional designs operate entirely in the analog domain for both modulation and demodulation. The military, commercial, and civilian sectors have all moved on from analog radio systems in favor of digital radio systems in recent years. Moreover, digital radio systems at different functional levels are using programmable hardware modules more and more frequently. Cross-functionality is restricted in conventional hardware-based radio devices because they can only be physically altered. As a result, production costs rise and there is less flexibility in supporting a wide range of waveform standards. On the contrary, FPGA technology offers an effective and reasonably priced answer to this issue, enabling wireless devices with multiple modes, bands, and/or functions that may be changed through software updates.

This research has done the integration of analog and digital modulation techniques into a single module for FPGA implementation to save resource. Integrating analog and digital modulation systems for FPGA implementation into one module can have the following advantages.

- ★ **Increased Efficiency:** By integrating analog and digital modulation systems into a single FPGA module, efficiency can be improved in terms of resource utilization, power consumption, and overall system performance.
- ★ **Complexity Reduction:** Integrating analog and digital modulation systems into one module simplifies the overall design and reduces complexity. With a unified module, it becomes easier to manage and maintain the FPGA implementation, resulting in reduced development effort and potential errors.
- ★ **Cost-effectiveness:** The integration of analog and digital modulation systems on a single FPGA module can lead to cost savings. It eliminates the need for multiple

FPGA devices or additional hardware for separate modulation systems, reducing manufacturing costs, and making the implementation more affordable.

- ★ **Improved Flexibility:** Integration allows for greater flexibility in terms of signal processing and modulation techniques. It enables the module to support a variety of modulation schemes, allowing for seamless switching between analog and digital modes based on specific requirements or signal conditions.
- ★ **Enhanced Performance:** The integration of analog and digital modulation systems can lead to improved overall system performance. By utilizing the strengths of both analog and digital modulation techniques, it becomes possible to achieve better signal quality, reduced noise, improved spectral efficiency, and enhanced transmission reliability.

So, this research considered all the above concerns to provide a multi-mode modulation system as a single module.

1.4 Thesis Objectives

1.4.1 General Objective

- To create multi-mode modulation techniques by combining three analog and three digital modulation techniques into a single module. Implement on FPGA hardware, improve the performance and compare with existing systems based on the FPGA resource utilization.

1.4.2 Specific Objectives

- Simulate AM, FM and PM in Matlab and convert the simulated waveform to HDL (bitstream).
- Simulate QASK, QPSK and 4QAM in Matlab and convert the simulated waveform to HDL (bitstream)

- Combine the analog and digital modulators to create a multi-mode modulator, simulate, generate bitstream and implement on Zed board FPGA.
- Based on the usage of FPGA resources such as total computational power total number of Look Table (LUT), Flip Flops (FF) and Input/Output (IO) port usage, compare the performance of the multi-mode modulation techniques to earlier studies.

1.5 Literature Review

Earlier studies have demonstrated the implementation of modulation techniques on FPGA through the use of the Vivado System Generator and the Matlab/Simulink environment.

In [30] the FPGA implementation of a low-power PSK modulator using Xilinx FPGA is presented. The authors implemented a number of power-saving techniques to reduce the power consumption of the PSK modulator. As a result, their implementation can achieve a power consumption of less than 1 W. The authors evaluated the FPGA resource utilization of their implementation and found that it is very efficient.

In [31] comprehensive overview of the FPGA implementation of digital modulation on FPGA is presented. The authors discusses the different types of digital modulation techniques, the FPGA architectures that are commonly used for their implementation, and the various optimization techniques that can be used to improve the FPGA resource utilization. The paper also presents a comparative analysis of the FPGA resource utilization of different digital modulation techniques implemented on Xilinx FPGAs.

In [32] Implementation of Multi-Mode Modulation Techniques for Software Defined Radio Using Xilinx FPGAs is done. This paper focuses on the FPGA implementation of multi-mode modulation techniques for SDR using Xilinx FPGAs. It discusses the challenges of implementing multi-mode modulation techniques on FPGAs and the various techniques that can be used to improve the FPGA resource utilization.

In [33] FPGA Implementation of Digital Modulation Techniques for IoT Applications is done. This paper reviews the FPGA implementation of digital modulation techniques for IoT applications. It discusses the different types of digital modulation techniques that are commonly used in IoT applications and the challenges of implementing them on FPGAs. The paper also presents a comparative analysis of the FPGA resource utilization of different digital modulation techniques implemented on Xilinx FPGAs for IoT applications.

In [3], Hamsa Abdulkareem and Raya Kahtan Mohammed developed and utilized an FPGA for implementing digital and analog modulation techniques. This paper introduces the hardware implementation of both analog and digital modulation. This paper's primary contribution is the application of digital platforms for analog modulation. FPGA is the foundation of the implementation process, offering a fast, adaptable, and adaptable experimental setup at a low cost. With the proper FPGA board and clock requirements, all of the designed digital and analog modulation systems were converted to VHDL codes utilizing HDL Netlist tools in the system generator token. Next, bitstreams that are downloaded to the NEXYS4 DDR FPGA Board device are created by synthesizing, implementing, and converting VHDL codes. The designed modulation algorithms were simulated and implemented using Vivado System Generator and MATLAB. The results were shown at the scope output of the system generator.

Poojan Murthy in [4] used Vivado System Generator to create and simulate a programmable digital modulator. The input data stream, the reconfigurable block with various modulations, and the output data stream comprise the modulation system. The demultiplexer receives a binary data stream with 2^n bits as input. The control signal's purpose is to transmit the necessary number of bits using the specified modulation. The output data stream provides the necessary modulation signal after gathering the bits in accordance with the control signal's instructions. The reconfigurable block is made up of QAM, FSK, PSK, and ASK modulation techniques. MATLAB and the Vivado System Generator 2015.3 version were used to simulate each of these modulation approaches, and the result was seen at the scope output of the system generator.

In [5] Bhaskara Rao Jammu et al used the Nexys 2 Spartan 3E FPGA board and the Xilinx ISE system generator to develop and implement BPSK, DPSK, BFSK, and APSK. Since there are limited programmable routing resources available on FPGAs, it is not feasible to utilize 100% of their logic cell in practice. The experimental results were performed using SPARTAN FPGA board and the result is viewed on oscilloscope. After implementation on FPGA comparisons done among the implemented modulation techniques based on FPGA resource utilization. According to the results, the metrics in BFSK and BPSK are the same, including the number of IO blocks, the number of slice LUTS, the total on-chip power, and the junction temperature. When comparing BASK modulation to BFSK, BPSK, and DPSK modulation, it is discovered that the junction temperature is higher. When using BASK modulation, there is a greater on-chip power usage.

In [6], B.K.V. Prasad and R. Sai Priya used MATLAB/Simulink and a Vivado system generator with a spartran-3e FPGA board to develop and implement digital modulation and demodulation. Algorithms for BPSK, ASK, and FSK have been

developed and evaluated. The signals that are obtained after implementation are compared with the modulated signals that are obtained from these simulations. The FPGA's output fits the simulation's results, and it also completes the reconfiguration process—which is necessary to choose the necessary modulation for optimization—something that is not possible with a single FPGA. The best modulation is chosen during the reconfiguration process depending on the available bandwidth, bit-error-rate, and signal-to-noise ratio.

However the research work listed above, on the other hand, don't deal with combining different modulation methods into a single module to produce a multi-mode modulation system. Additionally, they do not take into account utilizing the minimal amount of Look Table (LUT), Flip Flops (FF), Input/Output (IO) port consumption, and total processing power for FPGA hardware implementation. Thus, the goal of this research is to investigate each of these issues in order to offer a single module that combines analog and digital modulation. Three digital and three analog modulations will be constructed as a single module in this thesis. A hardware programmable device (FPGA) with a fast, adaptable, and customizable implementation technique becomes the basis for the experimental setup.

1.6 Scope and Limitation

1.6.1 Scope

This thesis focused only the modulation part. Six modulation techniques are selected and combined together to create multi-mode modulation system. The demodulation is not part this thesis. In addition to that this thesis only combines six modulation approaches, but depending on the demands of real-time applications, that number could be extended to 8, 16, and so on in the future.

1.6.2 Limitation

As with most studies the design of the proposed system has some limitations. Due to lack of DAC the output from the FPGA cannot be tested in real oscilloscope. Only the Hardware co-simulation is done.

1.7 Contributions

Combining three analog and three digital modulation techniques and creating multi-mode modulation (six input and six output) system and saving FPGA resource for implementation is the contribution of this thesis.

1.8 Significance of the study

- The importance of this study is found in its potential to provide critical insights into the suitability of various modulation techniques for software-defined radios (SDRs) platforms. The selection of the most suitable modulation technique for SDR depends on various factors, including the complexity of the modulation scheme, the computational power, and resource utilization.
- This study's significance also extends to the broader field of wireless communication systems, where the findings can be useful in improving the performance of communication systems that use software-defined radios. The study's outcomes can help in multiple wireless communication transmission using a single SDR platform resulting in more efficient communication systems with a lower number of Look Up Table (LUT) or memory size, Flip Flops (FF) and Input/Output (IO) port usage and reduced computational power.

1.9 Thesis Outline

Five chapters make up the remaining portion of the thesis. In chapter 2 software defined radio and FPGA background is presented. Chapter 3 presented the methodology.

Chapter 4 presented system design and modeling. Chapter 5 describe the result and discuss the performance in detail and the last chapter conclude about the thesis and describe the future work.

Chapter 2: Software Defined Radio (SDR)

2.1 Introduction and Historical Background

There are various definitions for SDR. The SDR Forum, in partnership with the Institute of Electrical and Electronic Engineers (IEEE) P1900.1 group, define SDR as "Radio in which some or all of the physical layer functions are software defined [21]".

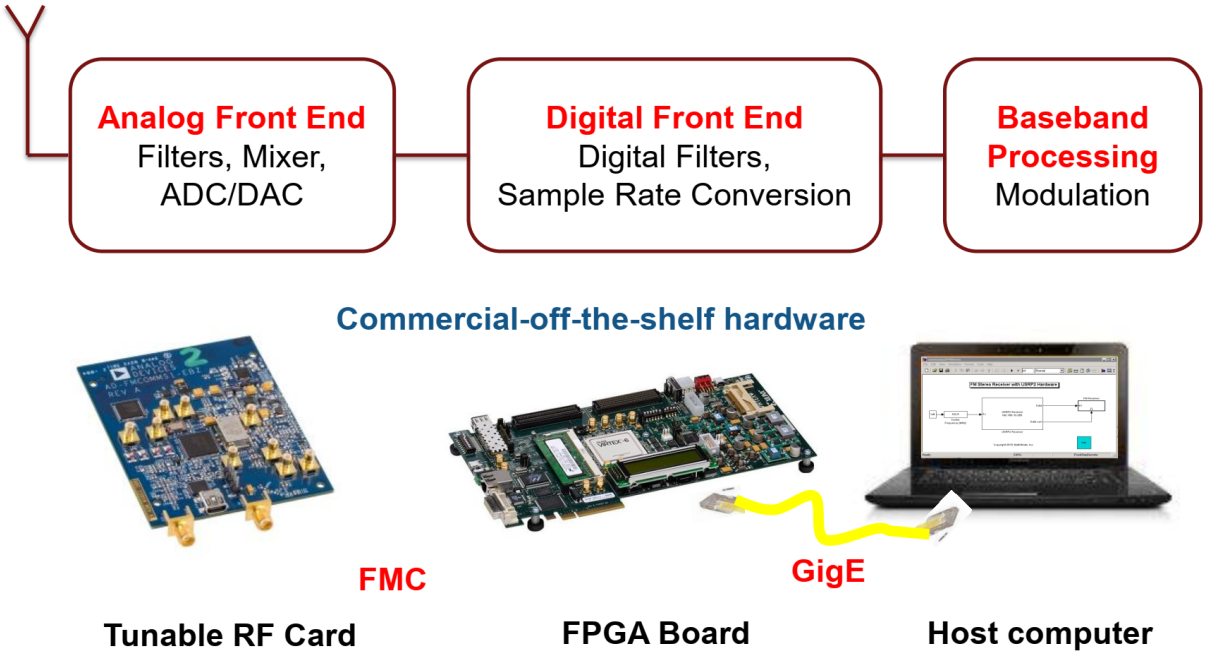

Figure 2.1: Typical SDR System [20].

SDR refers to a set of hardware and software technologies in which some or all of the radio's operational tasks (also known as physical layer processing) are accomplished using customizable software running on programmable processor technologies. An ideal SDR does not have any analog stages for signal processing, except antenna, power amplifier and microphone/loudspeaker (Figure 2.2) [22].

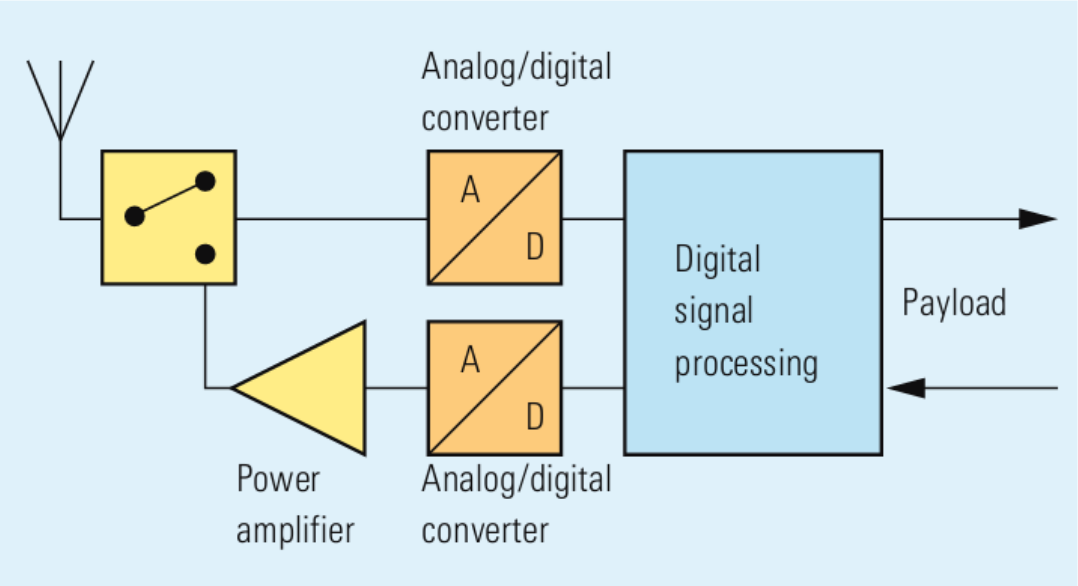

Figure 2.2: The "ideal" software defined radio [22].

2.2 Software Defined Radio Architecture

2.2.1 SDR Transmitter

The input to the Tx side of an SDR system is a digital baseband signal, typically generated by a DSP stage as shown in Figure 2.3 below.

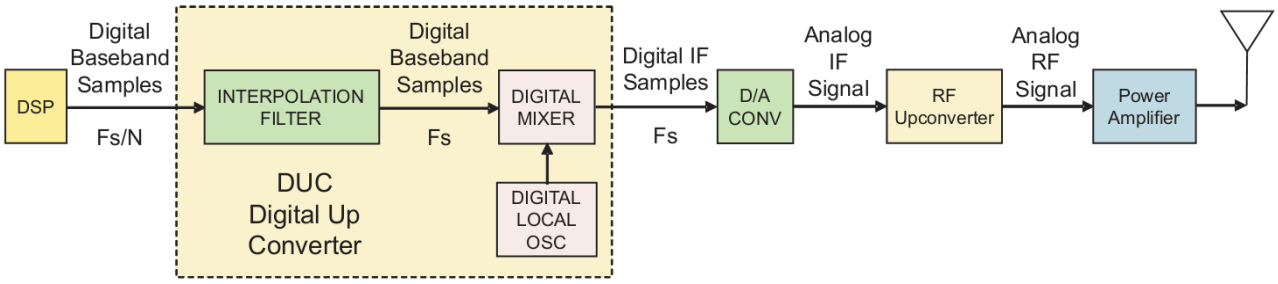

Figure 2.3: Software defined radio Transmitter [23].

A DUC (digital up converter) is the digital hardware block shown in dotted lines; it converts the baseband signal to the IF frequency. The digital IF samples are transformed into the analog IF signal by the subsequent D/A converter. The analog IF signal is then converted to RF frequencies by the RF up converter [23].

2.2.2 SDR Receiver

A software-defined radio receiver's block diagram is displayed in Figure 2.4. Similar to the first three stages of an analog receiver, the RF tuner converts analog RF signals to analog IF frequencies [23].

The digital downconverter (DDC) is typically a single monolithic chip or FPGA IP (Intellectual Property), and it is a key part of the SDR system [23].

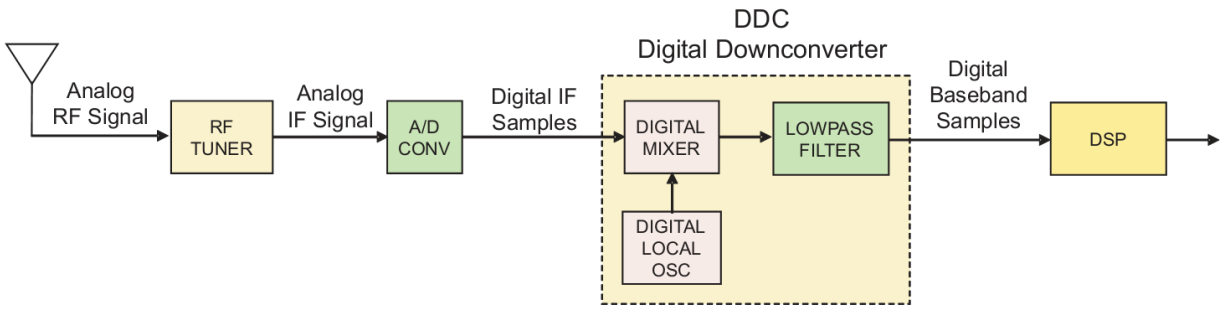

Figure 2.4: Software defined radio receiver [23].

A conventional DDC has three major sections:

- A digital mixer
- A digital local oscillator
- An FIR low pass filter [23].

2.3 Field Programmable Gate Array (FPGA)

A logic device known as a Field-Programmable Gate Array (FPGA) is made up of programmable switches and a two-dimensional array of generic logic cells. It is possible to program a logic cell to carry out a basic task, and a programmable switch can be made to provide connections between the logic cells. By defining the purpose of every logic cell and adjusting the connections of each programmable switch individually, a custom design may be put into practice. After the design and synthesis are finished, we can

download the required logic cell and switch configuration to the FPGA device and create the custom circuit using a straightforward adapter cable [24].

2.3.1 FPGA Architectures

The basic structure of an FPGA is composed of the following elements:

- ★ **Configurable Logic Blocks (CLB):** CLB contain the logic for the FPGA [25].
- ★ **Look-up table (LUT):** performs logic operations.
- ★ **Flip-Flop (FF):** stores the result of the LUT.
- ★ **Programmable interconnect:** connect elements to one another.
- ★ **Configurable I/O blocks:** ports get data in and out of the FPGA [25].
- ★ **On-chip memory:** To provide on-chip storage
- ★ **Hard macro intellectual property (IP) cores:** provide efficient complex functions [26].
- ★ **Clock management resources:** Clock distribution and frequency synthesis and clock shifting capabilities [26].

The fundamental FPGA architecture is depicted in Figure 2.5.

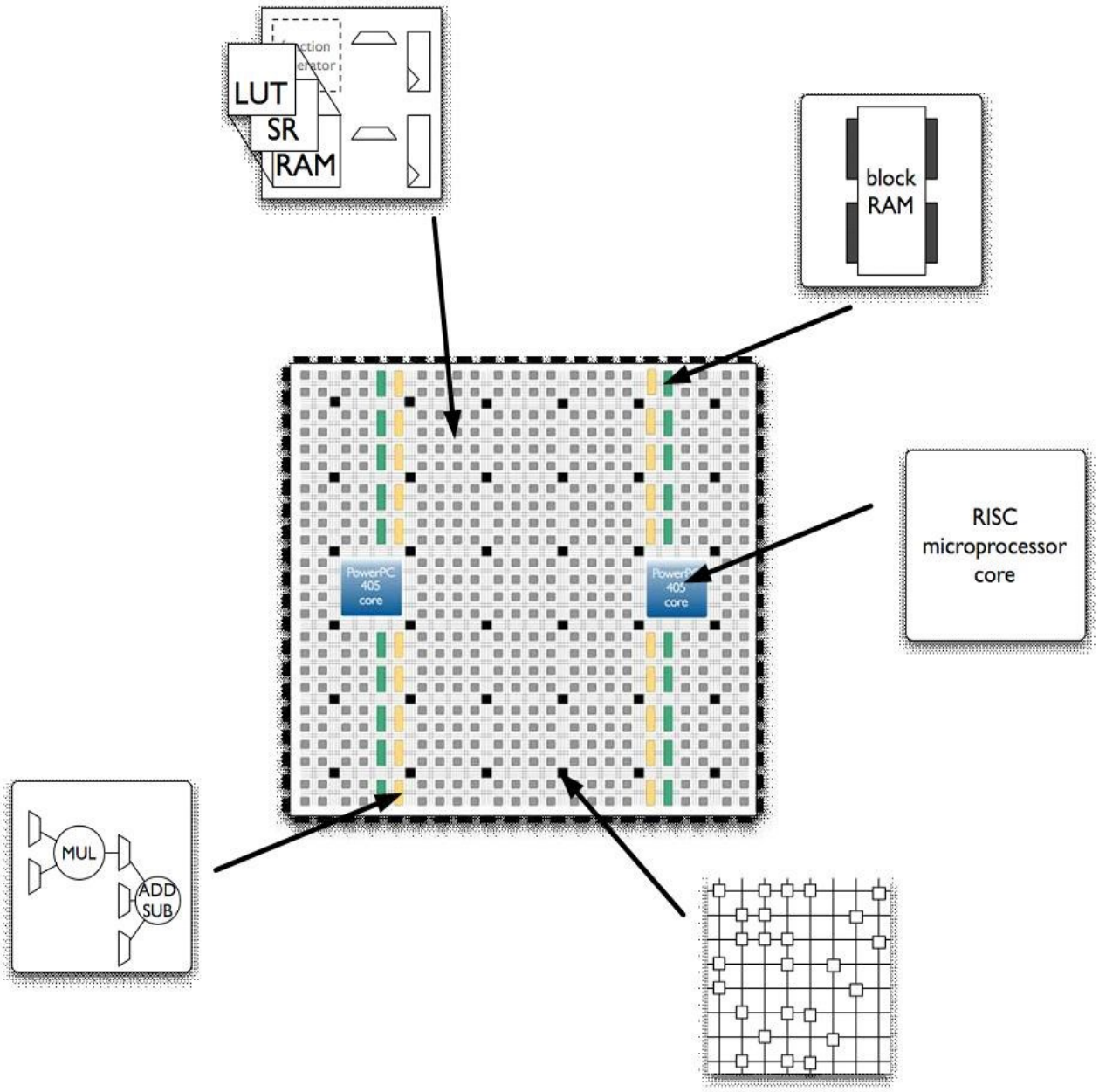

Figure 2.5: General FPGA architecture [26].

Chapter 3: Methodology

3.1 Introduction

Experimental approach is used to implement this thesis. Simulation and hardware implementation is conducted. The input for the system is an artificial data taken from MATLAB which is used to test the system functionality.

To implement the multi-mode modulation techniques on the FPGA, two different software packages are used: Vivado Design Suite 2019.2 using Xilinx's System Generator for DSP tool and Matlab/Simulink 2019a. Matlab/Simulink is used to verify system functionality and design the system, while Vivado Design Suite is used to configure the program to run on the FPGA. System Generator, a Matlab component that translates Simulink math code into VHDL code that the Vivado software can understand, serves as the fundamental connection between the two packages.

3.2 Overall Design Flow Using FPGA

A design tool in the Vivado Design Suite called System Generator for DSP enables programmers to use the Simulink design environment for FPGA design, which is based on models and is built on Math Works. To generate the FPGA programming file, all design and implementation procedures such as synthesis, place, and route are carried out automatically. Figure 3.1 [10] shows the system design process as a whole.

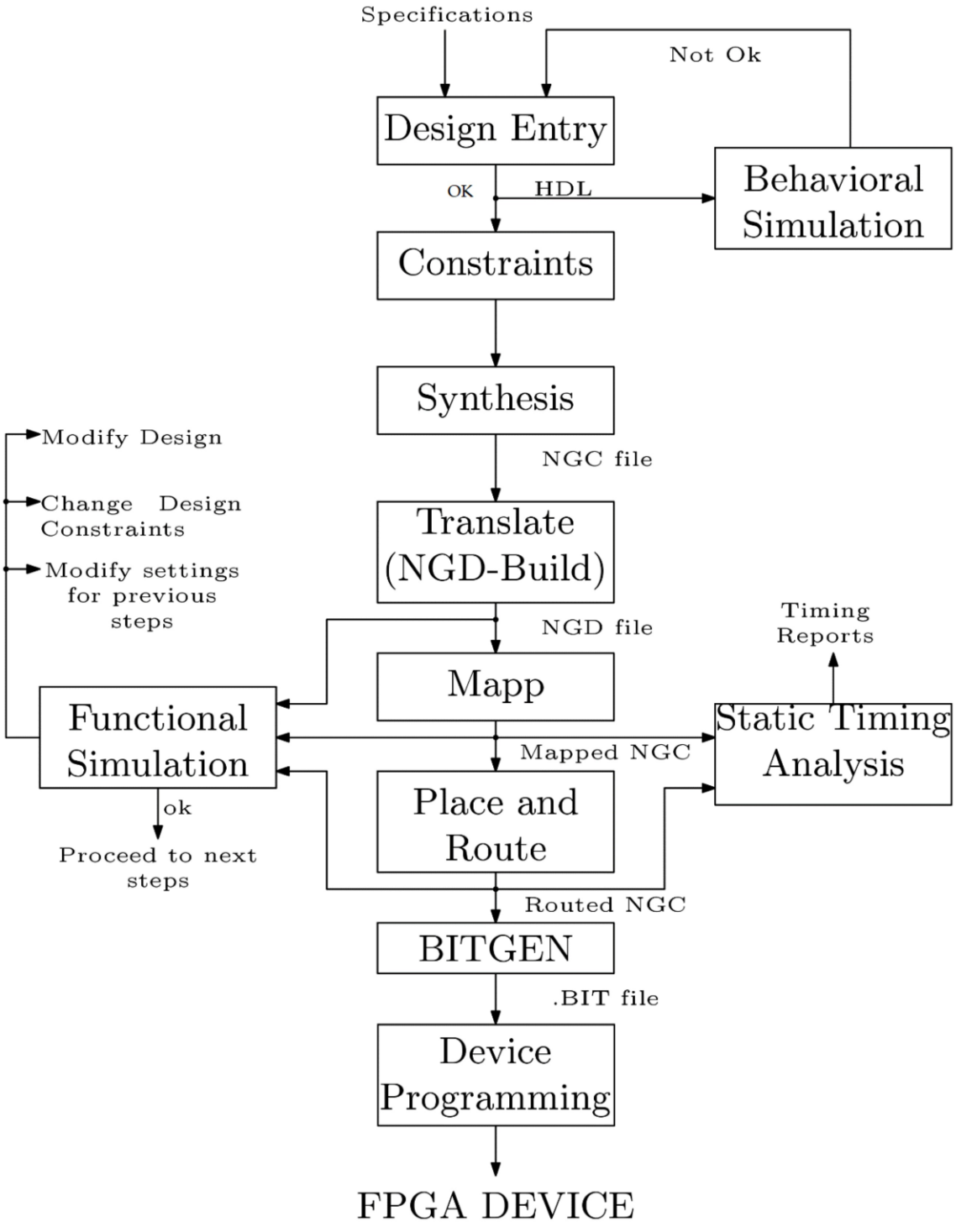

Figure 3.1: Design Flow using the Xilinx System Generator for DSP [34].

a) Design Entry

There are different techniques for design entry.

- Schematic based,
- Hardware Description Language (VHDL, VERILOG)
- Combination of both.

The method selected is determined by both the design and the designer.

b) Synthesis

The act of translating VHDL or Verilog code into a device netlist format—that is, a whole circuit containing logical components for the design, such as gates and flip flops—is called synthesis.

c) Implementation

This process consists of a sequence of three steps

- Translate: combines all the input netlists and constraints to a logic design file.
- Map: divides the whole circuit with logical elements into sub blocks such that they can be fit into the FPGA logic blocks.
- Place and Route

d) Device Programming

In this step the FPGA is programmed using the bitstream file generated from the design.

e) Design Verification

Verification can be done at different stages of the process steps.

- **Behavioral Simulation (RTL Simulation):** this simulation is run to validate the RTL (behavioral) code to make sure the design is operating as planned [34].
- **Functional simulation:** Functional simulation gives information about the logic operation of the circuit [34].

3.3 Methodology for Simulation and Implementation

The six modulation techniques are first constructed in Matlab system generator for simulation then after the hardware programing file will be generated from the design. Three multipliers is used to design the multi-mode modulation techniques. Those multipliers are shared among the six modulation techniques. Figure 3.2 shows the design flow for this multi-mode modulation techniques. Simulation and hardware implementation result is analyzed using oscilloscope.

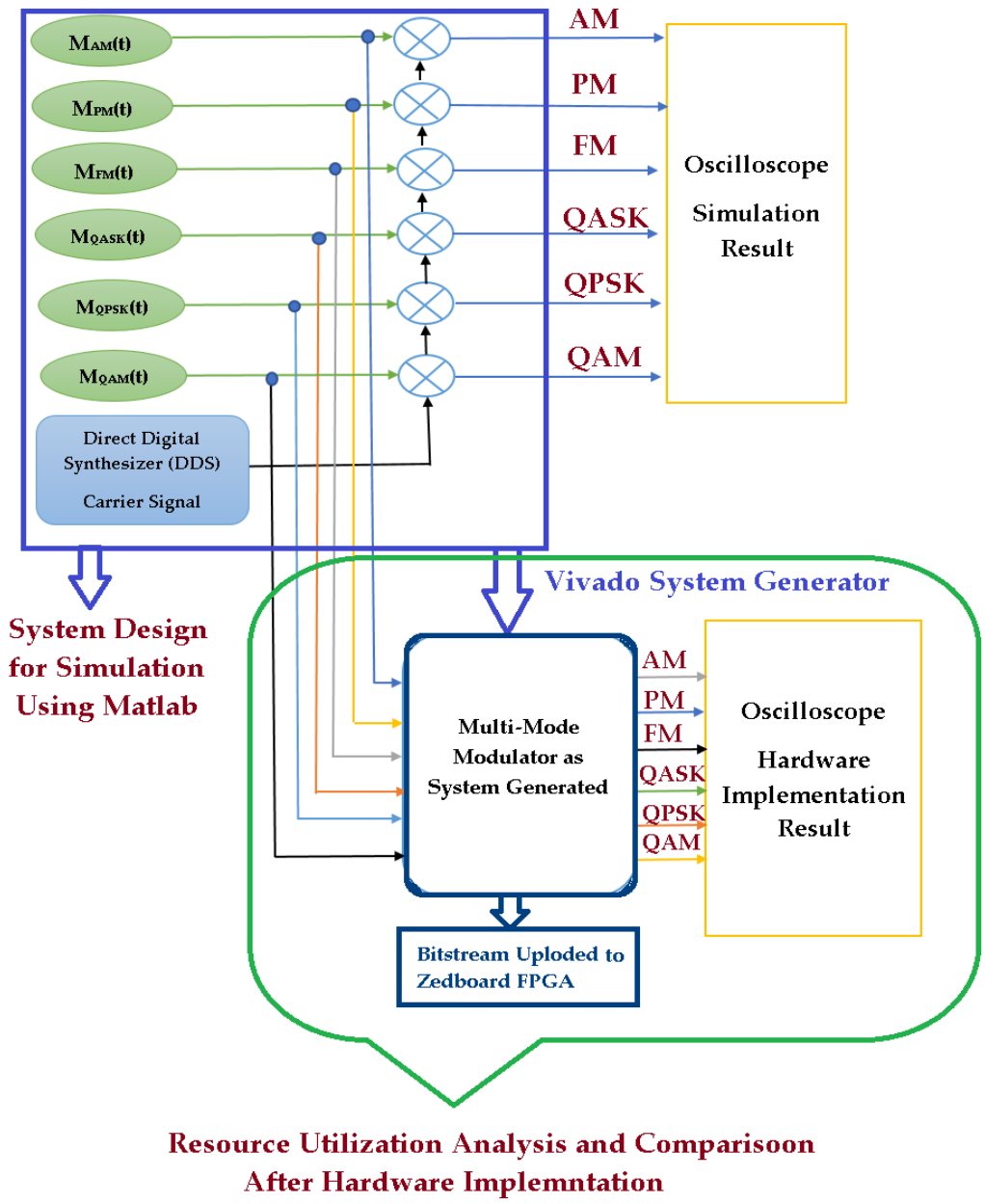

Figure 3.2: Methodology

Chapter 4: System Design and Modeling

4.1 System Flowchart

Figure 4.1 describe the system flowchart for the overall system. It briefly describe the design and implementation process of the implemented multimode modulation system.

The first step is to generate the carrier signal, which is done by the Direct Digital Synthesizer (DDS) signal generator.

The second step is to generate message signals for the analog and digital modulation for the input.

Next is to mix the carrier signal and the message signal to generate the six modulation types. After that, the six types of modulation are combined to form multimode modulation.

The next step is hardware simulation to verify the design. If the simulation result is as expected and all six modulation types are correctly simulated, an HDL file or the intellectual property (IP block) is generated for hardware implementation design.

Next, using the HDL file or IP block, a design for hardware implementation is prepared. Then, the design is implemented on FPGA hardware and the result is checked on an oscilloscope. If the hardware simulation result and the implementation result are the same, it means that the design is done correctly.

Finally, we go to the Results Analysis part. It means that we will take the resource utilization report from the implemented design and compare it with previous researches and prepare the summary of the study.

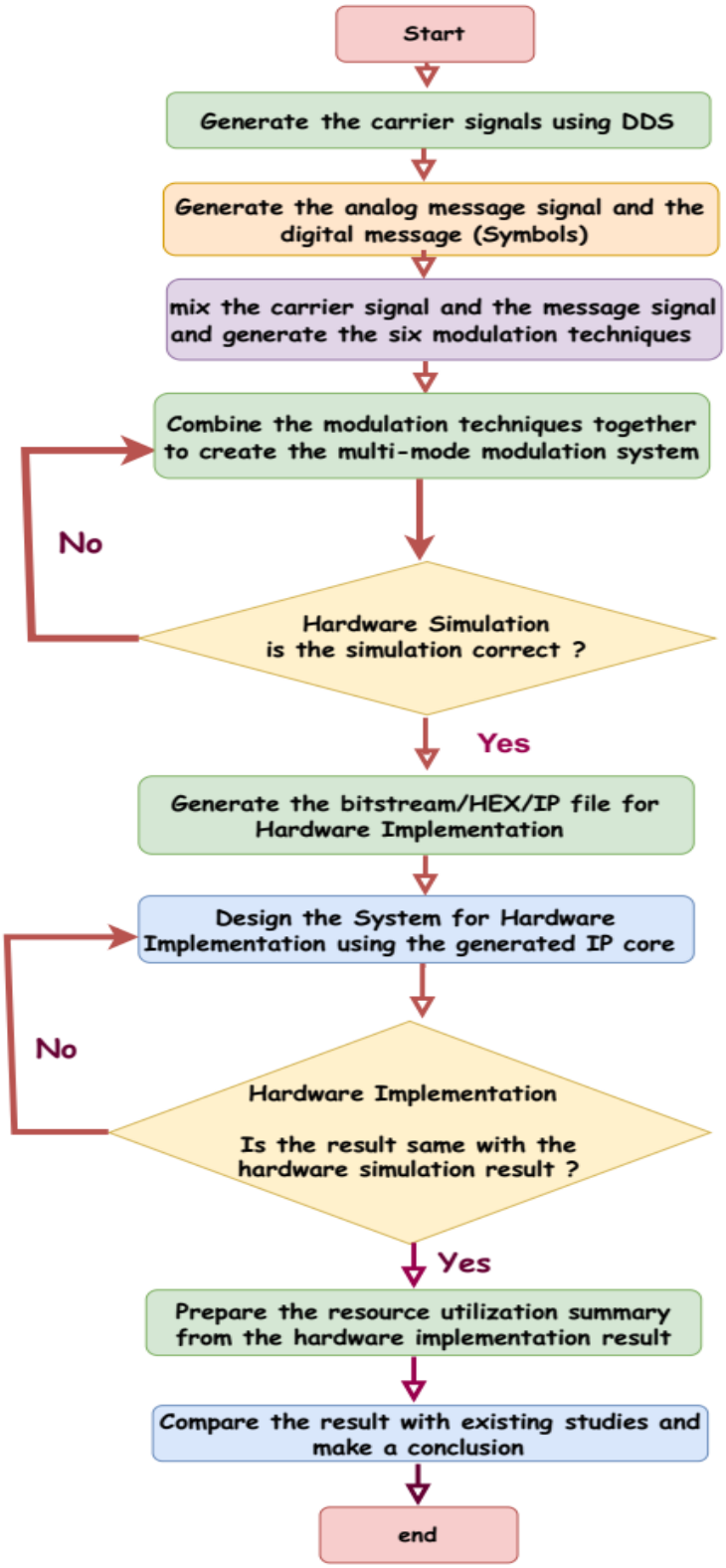

Figure 4.1: System Flow Chart

4.2 System Model

4.2.1 Analog Modulations System Modeling

Three analog modulation techniques (AM, FM and PM) are combined together to create an analog modulation system. Direct Digital Synthesizer (DDS), Cosine wave generator, multiplexers, multiplier and inverter are used to design the system. The mathematical analysis for those modulation techniques are given in following equations.

a) Amplitude Modulation (AM)

AM transmits information by varying the amplitude of the carrier signal while the phase and frequency is constant [1]. Let the information signal $m_{AM}(t)$ be message signal which is given by Equation 4.1

$$m_{AM}(t) = A_{AM}\, cos(2\pi f_{AM} t) \tag{4.1}$$

Where:

A_{AM} *is the Amplitude of the modulating signal*

f_{AM} *is the frequency of the modulating signal*

$\theta_{AM}(t)$ *is the phase of the modulating signal*

The carrier signal generated from Direct Digital Synthesizer (DDS) is represented in Equation 4.2.

$$C(t) = A_c\, cos\big(2\pi f_c t + \theta_C(t)\big) \tag{4.2}$$

Where:

A_c *is the Amplitude of the carrier signal*

f_C *is the frequency of the carrier signal*

$\theta_C(t)$ *is the phase of the carrier signal*

The standard carrier frequency in the baseband is in the range of the 550 KHz to 1720 KHz [1]. It is expected that $f_{AM} \ll f_c$. Therefore the Amplitude Modulated signal can be obtained by multiplying the carrier signal with the message signal assuming $\theta_C(t) = 0$ which is given by Equation 4.3.

$$S_{AM}(t) = m_{AM}(t) * C(t) = A_c m(t) \cos(2\pi f_c t) \tag{4.3}$$

Where:

$S_{AM}(t)$: represent the AM modulated waveform.

For AM modulation one multiplier is used to mix the message signal with the carrier signal. The system can be modeled as shown in figure 4.2.

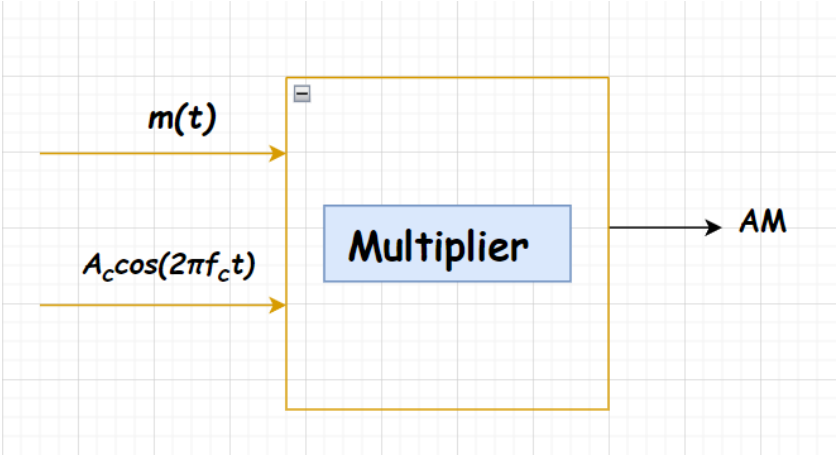

Figure 4.2: System model for AM modulation

b) Phase Modulation (PM)

PM transmits information by varying the phase of the carrier signal while the amplitude and frequency is constant [1]. Let the information signal $m_{PM}(t)$ be message signal which is given by Equation 4.4.

$$m_{PM}(t) = A_{PM}\, cos\big(2\pi f_{PM} t + \theta_{PM}(t)\big) \tag{4.4}$$

Where:

A_{PM} *is the Amplitude of the modulating signal*

f_{PM} *is the frequency of the modulating signal*

$\theta_{PM}(t)$ *is the phase of the modulating signal*

The carrier signal generated from Direct Digital Synthesizer (DDS) is also represented in Equation 4.2. Therefore the Phase Modulated signal is obtained by multiplying the carrier signal with the message signal. In the system model 2 by 1 multiplexer is used to design the PM modulator. Based on the input (message signal) for the multiplexer the two waveforms which has 90^0 phase difference is selected. Inverter is used to change the phase of the carrier signal. Therefore the phase modulated waveform is represented as shown in Equation 4.5. The system can be designed as shown in figure 4.3.

$$S_{Pm}(t) = A_c \cos(2\pi f_c t + k_{PM} m(t)) \tag{4.5}$$

Where:

$S_{PM}(t)$: represent the phase modulated waveforms.

K_{Pm} : Represent the constant of proportionality for phase modulation.

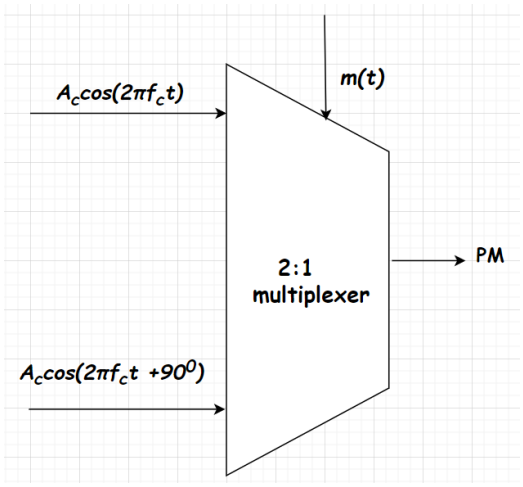

Figure 4.3: System model for PM modulation

c) *Frequency Modulation (FM)*

Frequency modulation transmits information by varying the frequency of the carrier signal by keeping the amplitude and phase constant [1]. Let the information signal $m_{FM}(t)$ be an arbitrary waveform, called message signal which is given by Equation 4.6.

$$m_{FM}(t) = A_{FM}\, cos\big(2\pi f_{FM} t + \theta_{FM}(t)\big) \tag{4.6}$$

Where:

A_{FM} *is the Amplitude of the modulating signal*

f_{FM} *is the frequency of the modulating signal*

$\theta_{FM}(t)$ *is the phase of the modulating signal*

The carrier signal generated from Direct Digital Synthesizer (DDS) is shared between AM and PM modulators and represented in Equation 4.2. Therefore the frequency Modulated signal can be obtained by multiplying the carrier signal with the message signal. In the system model 2 by 1 multiplexer is used to design the FM modulator. Based on the input for the multiplexer the carrier is varied. The FM modulated waveform is represented by equation 4.7 and it can be modeled as shown in figure 4.4.

$$S_{Fm}(t) = A_c \cos\left(2\pi f_c t + 2\pi k_f \int m(t)dt\right) \tag{4.7}$$

Where:

$S_{FM}(t)$: represent the Frequency modulated waveform.

k_f is the frequency deviation constant .

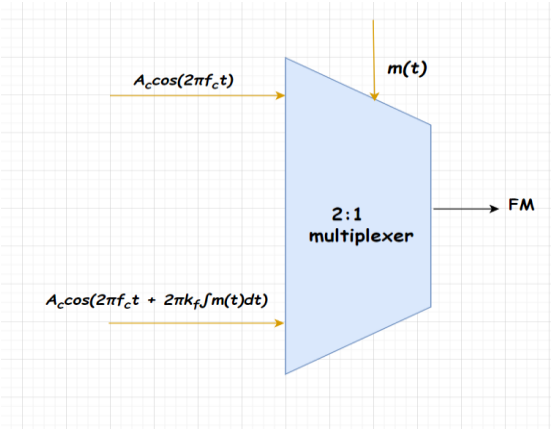

Figure 4.4: System model for FM modulation

4.2.2 Digital Modulations System Modeling

Three digital modulation techniques (QASK, QPSK and QAM) are combined together to create Digital modulation system. The mathematical analysis for this modulation techniques are given in Equation 4.8 to 4.20.

For any set of M real signals $S = \{s_1(t) \ldots s_M(t)\}$ defined on [0, T), we can define M linearly independent signals $\{s_i(t)\}$,

$$s_i(t) = s_{i1}(t)g(t)\cos(2\pi f_c t) + s_{i2}(t)g(t)\sin(2\pi f_c t) \qquad for\ i = 1,2,\ldots M \tag{4.8}$$

So, if we have M=4 (2 bits per symbol) we will have four signal outputs.

$\{S = s_1(t), s_2(t), s_3(t), s_4(t)\}$

For each digital modulation selected the $s_i(t)$ representation will be given in the following equations.

a) M-ASK (no quadrature components in here)

In Amplitude Shift Keying (QASK) the amplitude of the carrier is changed in response to the digital data stream (information signal) which is represented as $m_{QASK}(t)$. The carrier signal generated from Direct Digital Synthesizer (DDS) can be shared with

the analog modulation and represented in equation 4.2. A_1 and A_2 are multipliers used to control the amplitude of the carrier signal. Therefore the QASK waveform is defined as:

$$s_i(t) = A_i g(t)\cos(2\pi f_c t) \quad (4.9)$$

$$Where\ A_i = (2i - 1 - M)d\ for\ i = 1,2, \ldots M$$

For M=4 we will have 4 outputs $A_i = \{-3d, -1d, 1d, 3d\}$, taking d=1 and g(t) is just pulse shaping function the amplitude will be $\{-3, -1, 1, 3\}$ so the waveform will looks like

$$s_1(t) = A_1 g(t)\cos(2\pi f_c t) = -3g(t)\cos(2\pi f_c t) = -3\cos(2\pi f_c t) \ldots .00$$
$$s_2(t) = A_2 g(t)\cos(2\pi f_c t) = -1g(t)\cos(2\pi f_c t) = -\cos(2\pi f_c t) \ldots . 01$$
$$s_3(t) = A_3 g(t)\cos(2\pi f_c t) = 1g(t)\cos(2\pi f_c t) = \cos(2\pi f_c t) \ldots \ldots \ldots .10$$
$$s_4(t) = A_4 g(t)\cos(2\pi f_c t) = -3g(t)\cos(2\pi f_c t) = 3\cos(2\pi f_c t) \ldots . . 11$$

So the constellation diagram is shown in figure 4.5.

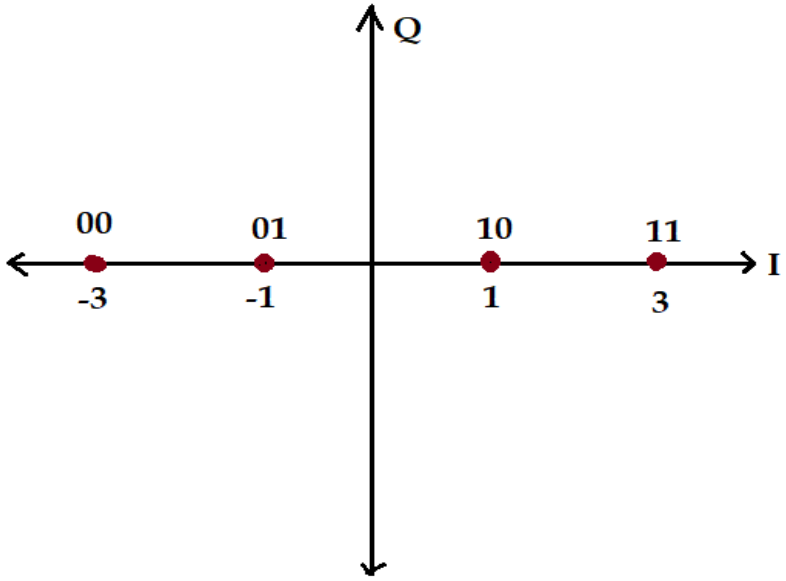

Figure 4.5: Constellation diagram of 4-ASK

So as we can see from the equation and constellation diagram 4-ASK need two multipliers and two inverters. The system model diagram is shown in figure 4.6.

$$Where\ the\ value\ of\ \ A_1 = 3\ and\ A_2 = \ 1$$

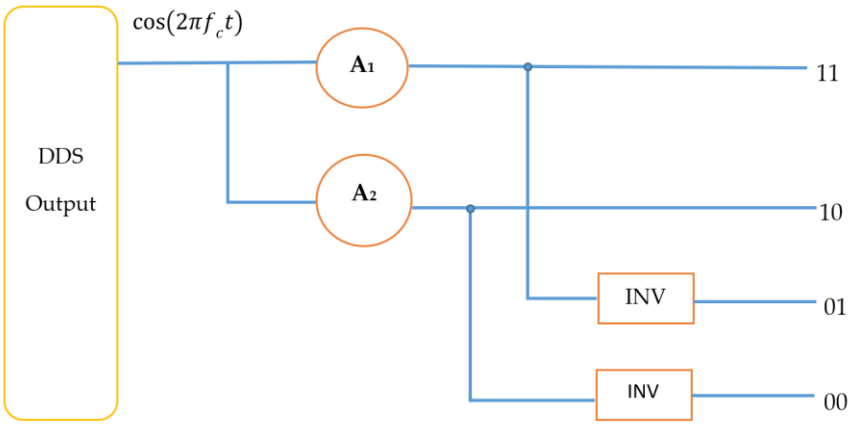

Figure 4.6: System model for 4-ASK

b) M-PSK

In Phase Shift Keying (PSK) the phase of the carrier is changed in response to the digital data stream (information signal). The carrier signal generated from Direct Digital Synthesizer (DDS) which is shared with the analog modulation were represented in equation 4.2. Two inverters are used to change the phase of the carrier signal. Therefore the QPSK waveform are defined as:

$$s_i(t) = A_1 \cos(\theta_i)\, g(t) \cos(2\pi f_c t) + A_1 \sin(\theta_i)\, g(t) \sin(2\pi f_c t) \qquad (4.10)$$

$$\text{Where } \theta_i = \frac{2\pi(i-1)}{M} \; for \; i = 1,2,\dots,M$$

Similarly for M=4 we will have 4 outputs as with $\theta_i = \{0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}\}$

So by taking A_1=1 the four waveforms will look like:-

$$s_1(t) = A_1 \cos(0)\, g(t) \cos(2\pi f_c t) + A_1 \sin(0)\, g(t) \sin(2\pi f_c t) = A_1 \cos(2\pi f_c t) \dots\dots\dots\dots 00$$

$$s_2(t) = A_1 \cos(\pi/2)\, g(t) \cos(2\pi f_c t) + A_1 \sin(\pi/2)\, g(t) \sin(2\pi f_c t) = A_1 \sin(2\pi f_c t) \dots 01$$

$$s_3(t) = A_1 \cos(\pi)\, g(t) \cos(2\pi f_c t) + A_1 \sin(\pi)\, g(t) \sin(2\pi f_c t) = -A_1 \cos(2\pi f_c t) \dots\dots\dots\dots 10$$

$$s_4(t) = A_1 \cos(3\pi/2)\, g(t) \cos(2\pi f_c t) + A_1 \sin(3\pi/2)\, g(t) \sin(2\pi f_c t) = -A_1 \sin(2\pi f_c t) .. 11$$

So the constellation diagram is drawn as shown in figure 4.7.

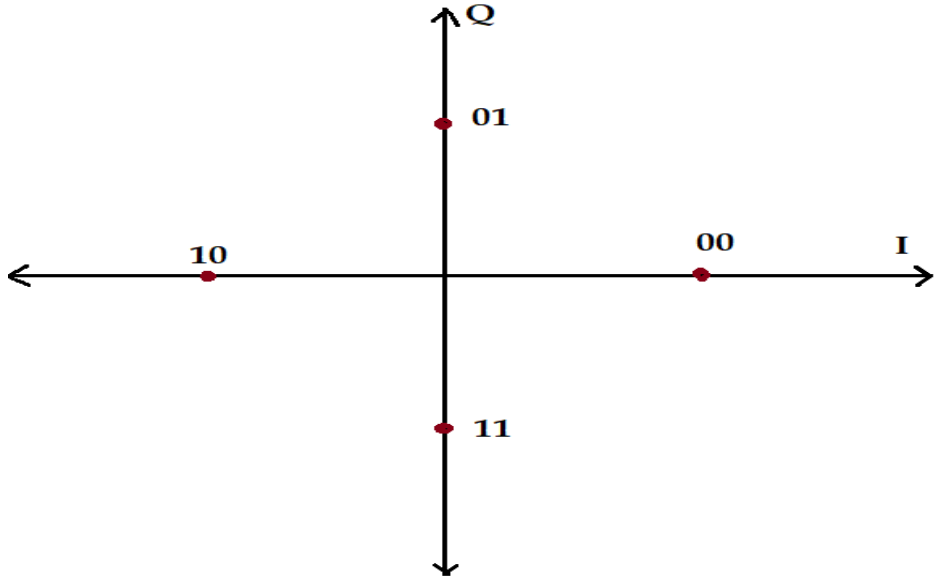

Figure 4.7: Constellation diagram of 4-PSK

Taking A_1=1 will result

$$s_1(t) = \cos(2\pi f_c t) \dots \dots 00$$

$$s_2(t) = \sin(2\pi f_c t) \dots \dots .01$$

$$s_3(t) = -\cos(2\pi f_c t) \dots \dots .10$$

$$s_4(t) = -\sin(2\pi f_c t) \dots \dots 11$$

c) M – QAM

In Quadrature Amplitude Modulation (QAM) the phase and amplitude of the carrier is changed in response to the digital data stream (information signal). The carrier signal generated from Direct Digital Synthesizer (DDS) which is shared with the analog modulation is represented in Equation 4.2. Two inverters which are used in QPSK are also used in QAM to change the phase of the carrier signal. Therefore the QAM waveform can be defined as:

$$s_i(t) = A_k \cos(\theta_k)\, g(t) \cos(2\pi f_c t) + A_k \sin(\theta_k)\, g(t) \cos(2\pi f_c t) \qquad (4.11)$$

Where $\theta_k = \frac{2\pi(k-1)}{M} and\ A_k = (2k-1-K)d\ \ and\ for\ k = 1,2,\ldots K\ where\ K = \log_2 M$

So for M=4, $A_k = \{-1d, 1d\}$ and $\theta_k = \{0, \frac{\pi}{2}\}$, the 4 waveform will looks like

$$s_1(t) = A_1 \cos(\theta_1)\, g(t) \cos(2\pi f_c t) + A_1 \sin(\theta_1)\, g(t) \sin(2\pi f_c t) = -A_1 \cos(2\pi f_c t) \ldots .00$$

$$s_2(t) = A_1 \cos(\theta_2)\, g(t) \cos(2\pi f_c t) + A_1 \sin(\theta_2)\, g(t) \sin(2\pi f_c t) = -A_1 \sin(2\pi f_c t) \ldots .01$$

$$s_3(t) = A_2 \cos(\theta_1)\, g(t) \cos(2\pi f_c t) + A_2 \sin(\theta_1)\, g(t) \sin(2\pi f_c t) = A_2 \cos(2\pi f_c t) \ldots \ldots\ 10$$

$$s_4(t) = A_2 \cos(\theta_2)\, g(t) \cos(2\pi f_c t) + A_2 \sin(\theta_2)\, g(t) \sin(2\pi f_c t) = A_2 \sin(2\pi f_c t) \ldots \ldots .11$$

Taking $A_1 = 1$ and $A_2 = -1$ will result

$$s_1(t) = \cos(2\pi f_c t) \ldots \ldots 00$$

$$s_2(t) = \sin(2\pi f_c t) \ldots \ldots 01$$

$$s_3(t) = -\cos(2\pi f_c t) \ldots \ldots 10$$

$$s_4(t) = -\sin(2\pi f_c t) \ldots \ldots 11$$

So as we can see from the derivation 4-PSK and 4-QAM have identical representation of the constellation and carrier signals for the four symbols. So that the system model is built as shown in figure 4.8.

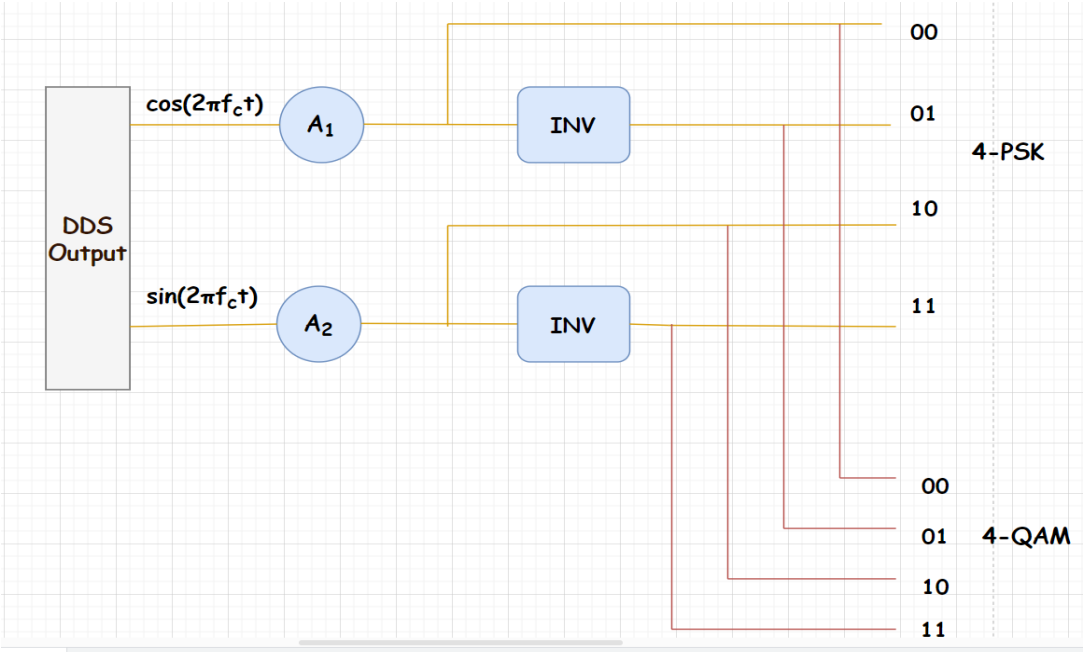

Figure 4.8: System model for 4-PSK and 4-QAM

4.2.3 Multi-Mode modulation System Modeling

The multi-mode modulation technique designed and implemented in this thesis are created by combining the above analog and digital modulation system. The main building blocks used to model the system is message signal generator, Direct Digital synthesizer for carrier signal generation, multipliers, inverters and multiplexers. Figure 4.9 shows the system model for proposed multi-mode modulation system.

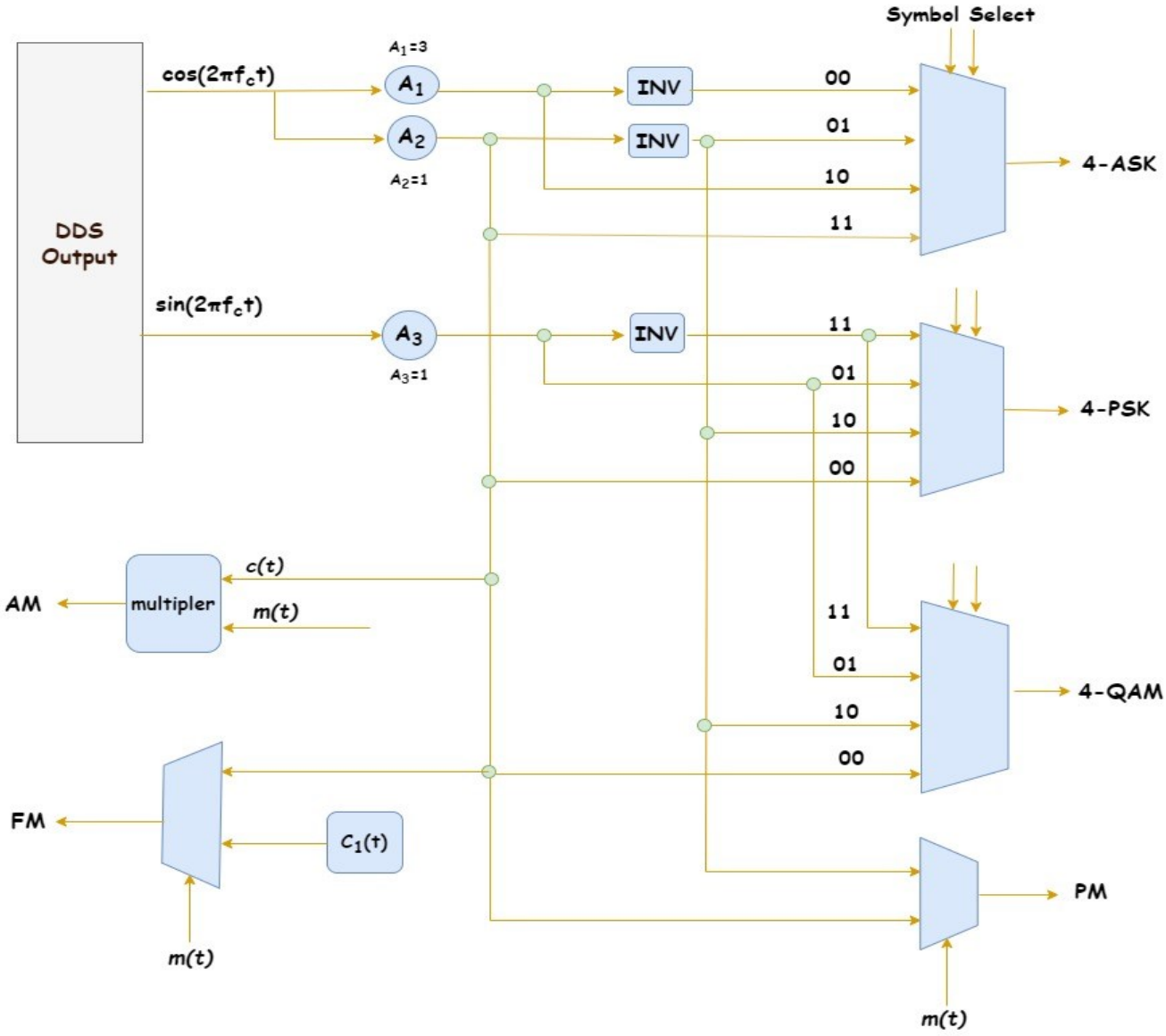

Figure 4.9: Multi-Mode Modulations System Model

So, Combination of Multipliers with the Increase in digital modulation order is formulated as shown below:

- ***8-ASK***

For M=8, Using equation 4.9 the amplitude values are calculated and the number of multipliers required is indicated as follow.

For M=8 we will have 4 outputs $A_i = \{-7d, -5d, -3d, -1d, 1d, 3d, 5d, 7d\}$, so 8-ASK needs 4 multipliers and 4 inverters

- ***8-PSK***

For M=8, Using Equation 4.10 the phase values are calculated and the number of multipliers required is indicated as follow.

For M=8 we will have 8 different phase outputs as with $\theta_i = \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}, \pi, \frac{5\pi}{4}, 2\pi, \frac{7\pi}{4}\}$

So, 8-ASK needs 1 multiplier and 8 different phases values

- ***8-QAM***

For M=8, Using equation 4.11 the amplitude and phase values are calculated and the number of multipliers required is indicated as follow.

$$\theta_k = \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}$$

$$A_k = \{-3d, -1d, 1d, 3d\}$$

So, 8-QAM needs 2 multipliers with 2-inverter and 4 different phase values. As we can see from the calculation the amplitude value in 8-QAM and 8-ASK have common values so that this modulation techniques can share the common values. So, for the higher modulation order the number of multipliers required can be calculated as a function of modulation order as mentioned in Equation 4.12.

$$\left(\frac{m}{2}+1\right)+inverters \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \; 4.12$$

The combination of multipliers with the increase in digital modulation order is given in table 4.1.

Table 4.1: Combination of multipliers with the increase in digital modulation order

Modulation order	Number of Multipliers Required
4	$3 + Inverters$
8	$5 + inverters$
16	$9 + inverters$
. . . m	. . . $\left(\frac{m}{2}+1\right)+inverters$

4.3 System Design

4.3.1 Tools

Xilinx's System Generator is a DSP design tool that allows FPGA designers to leverage the Math Works model-based Simulink design environment [17].

The multi-mode modulation techniques built in this thesis have six input and six output. Multipliers and DDS module are shared among the six modulation techniques. Sharing this modules greatly saved resources, which is the main assumption of this thesis.

The main building blocks used to design this modulation systems are gateway in, gateway out, DDS, sine wave generator, multiplier, multiplexer, inverters and Oscilloscope.

- ***Xilinx Gateway in and Xilinx Gateway out:*** - are used to define the inputs and outputs to the Xilinx design [1].

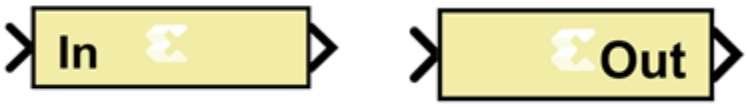

Figure 4.10: Xilinx Gateway in and Xilinx Gateway out block

- ***Xilinx Direct Digital Synthesizer (DDS) Compiler:*** **-** is used to generate frequencies [18].

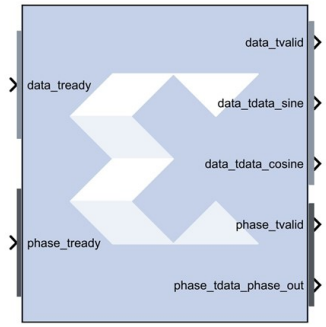

Figure 4.11: Xilinx Direct Digital Synthesizer (DDS) Compiler [18].

- ***Xilinx Sine wave generator:*** **-** generates a sine wave [19].

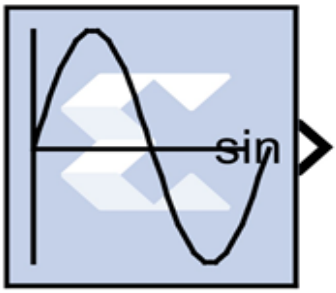

Figure 4.12: Xilinx Sine wave generator [19].

- ***Repeating Sequence Stair:*** **-** A value in Vector of output values is output at each time interval, and then the sequence repeats.

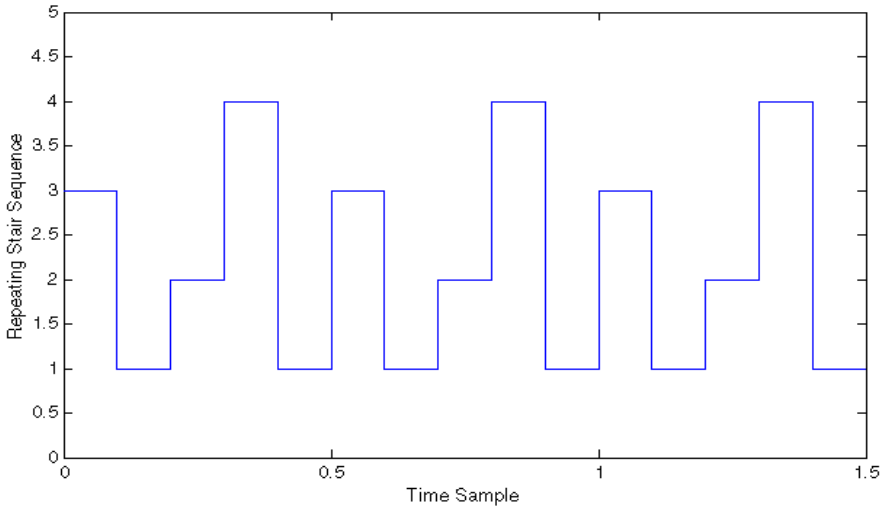

Figure 4.13: Repeating Sequence Stair

- ***Xilinx Constant Multiplier: -*** implements a gain operator.

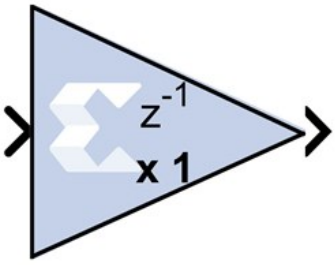

Figure 4.14: Xilinx Constant Multiplier

- ***Xilinx Multiplier: -*** It computes the product of the data on its two input ports.

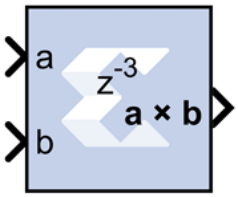

Figure 4.15: Xilinx Multiplier

- ***Xilinx Bus Multiplexer:*** **-** The Xilinx Mux block implements a multiplexer.

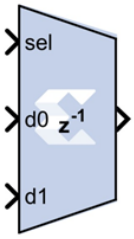

Figure 4.16: Xilinx Bus Multiplexer

- ***The Xilinx Inverter block:*** **-** calculates the bitwise logical complement of a fixed-point number.

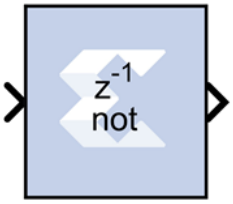

Figure 4.17: The Xilinx Inverter block

- ***System Generator Token:*** **-** serves as a control panel for controlling system and simulation parameters.

Figure 4.18: System Generator Token

4.3.2 System Design for Analog Modulation system

Based on the system model the Analog modulation system is designed in Matlab and Xilinx System Generator. Figure 4.19 shows the designed system for the analog modulation system. The DDS is used to generate the carrier signal, the analog message

signal block is used to generate the message signal for the input, and the multiplexer block is used to selects between the two input signals and forwards the selected input to a single output line. The selection is directed by the message signal known as select lines. Gate way in and gate way out at the input and at the output is used for data type conversion, i.e. from floating point to fixed point data type. The oscilloscope block is used to display and analyze the output waveform. The red-colored System Generator token in the design is used to activate the code generator for netlisting and functions as a control panel for system and simulation parameters. Figures 4.20 to 4.21a and 4.21b below display the system design parameters for the analog modulation system.

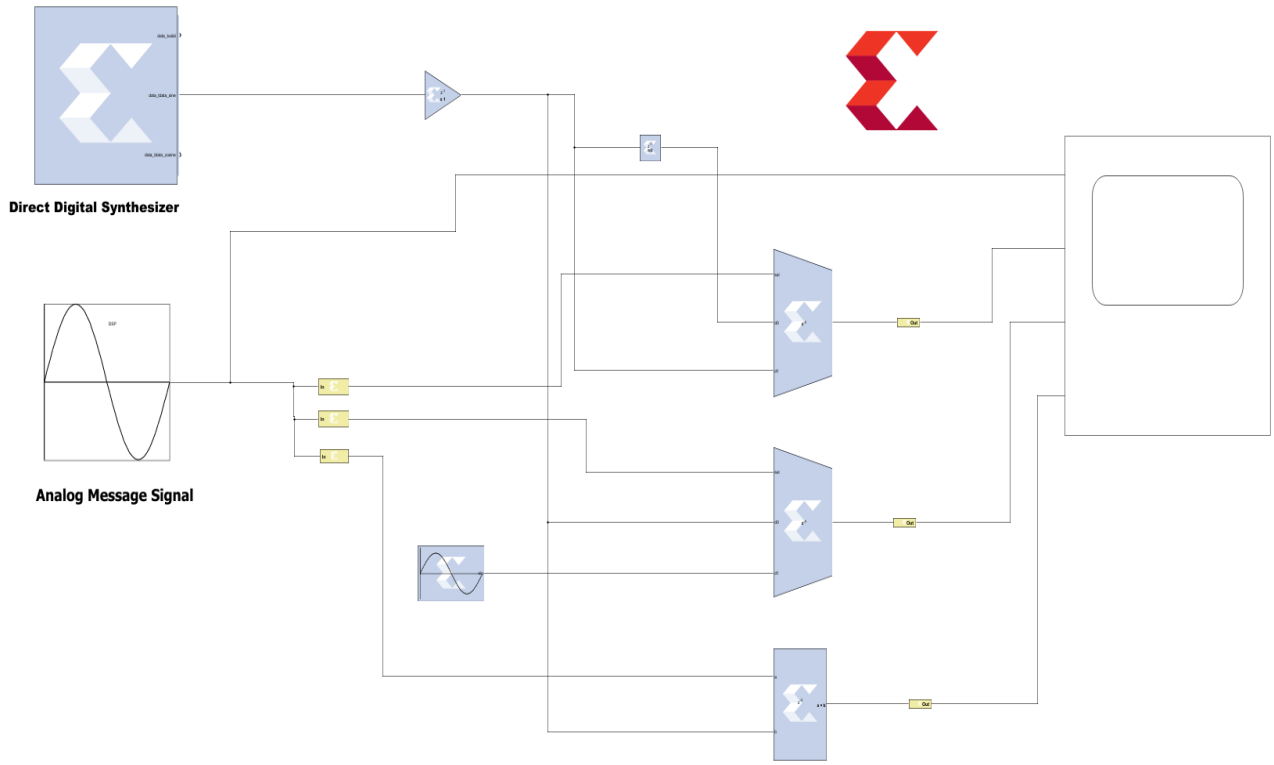

Figure 4.19: System Design for Analog Modulation system

4.3.2.1 Design parameters for Analog Modulation system

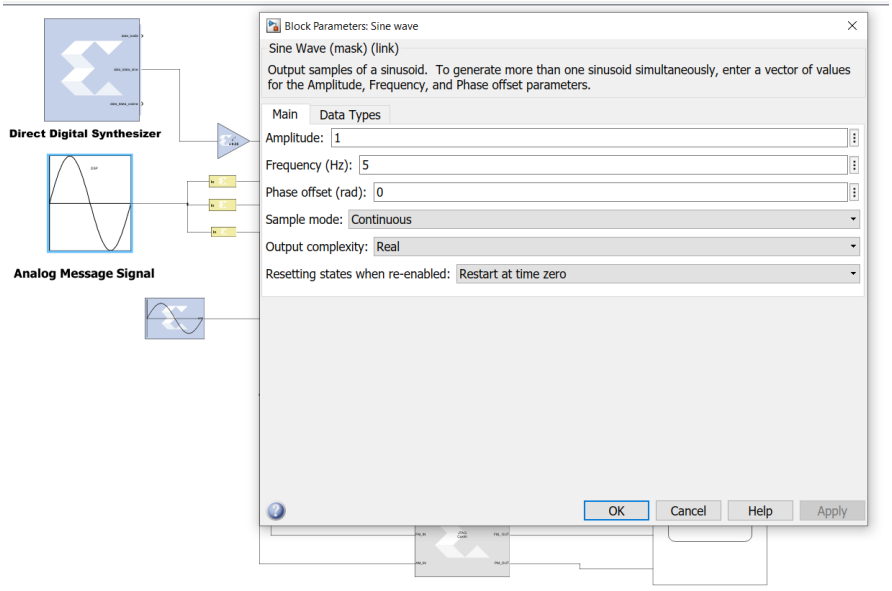

Figure 4.20: Design parameters for the analog message signal

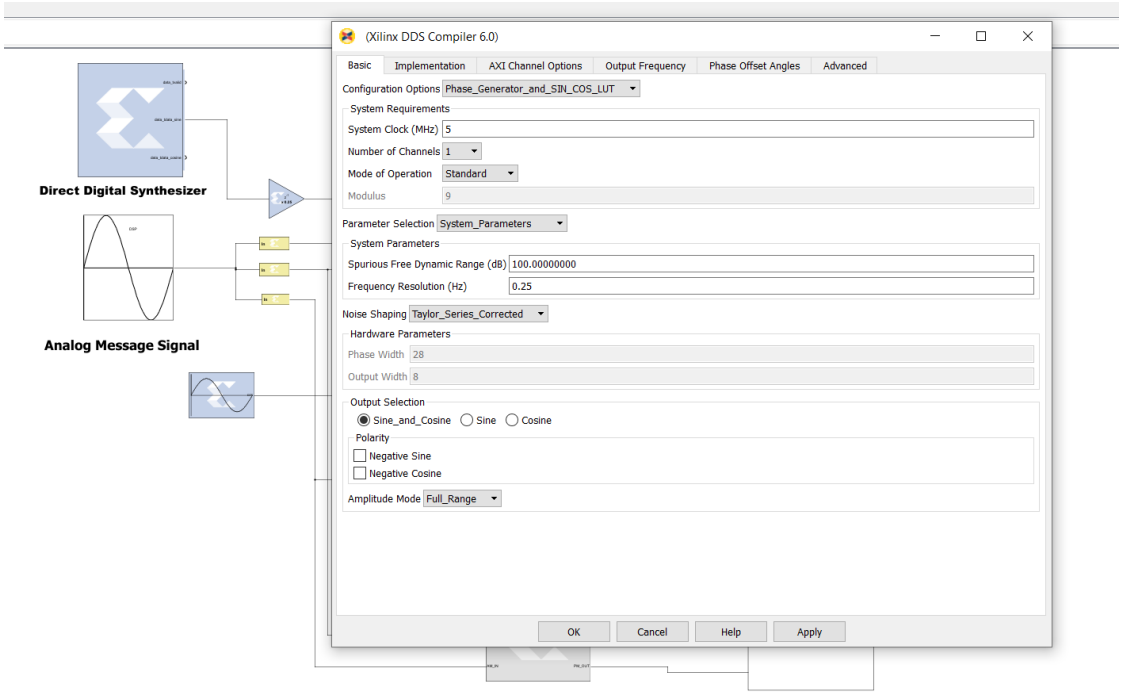

Figure 4.21 (a): Design parameters for the carrier signal-basic parameters

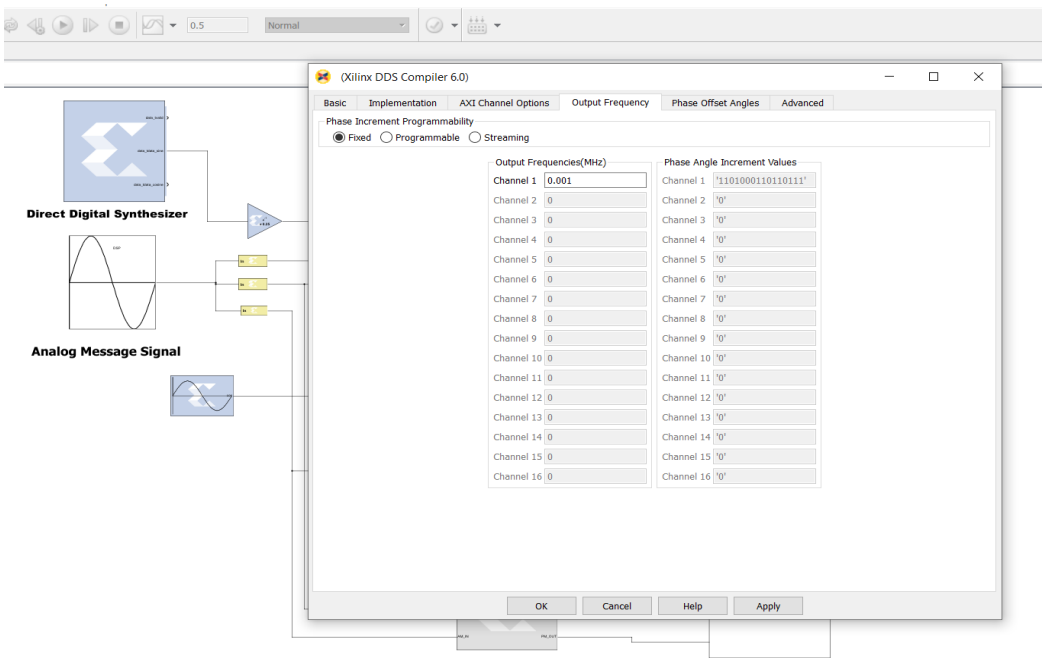

Figure 4.21 (b): Design parameters for the carrier signal-output frequency

4.3.3 System Design for Digital Modulation System

Based on the system model the digital modulation system is designed in Matlab and Xilinx System Generator. Figure 4.22 shows the designed system for the digital modulation system.

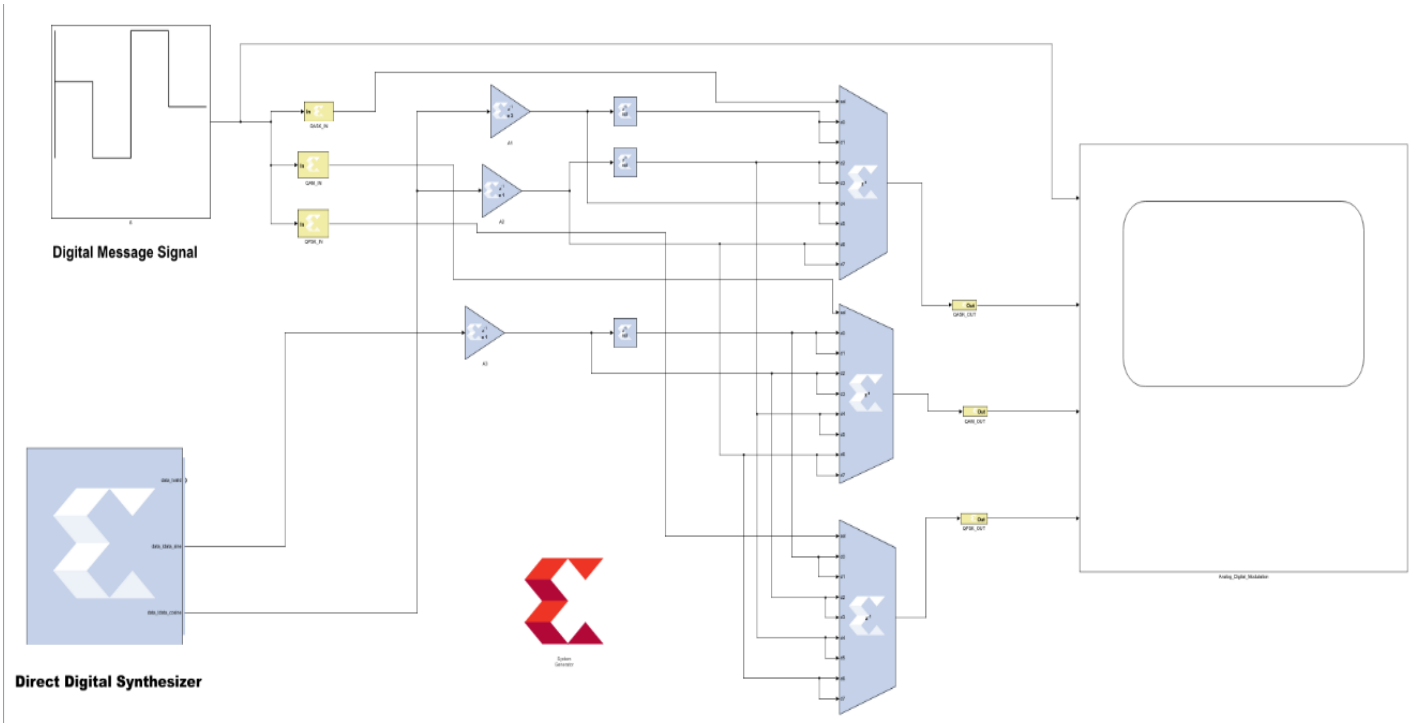

Figure 4.22: System Design for Digital Modulation system

The DDS block is used to generate the carrier signal, the digital message signal block the four digital symbols for the input. The multiplexer block is used to selects and multiplex the carrier waveform according to the four symbols input. Gate way in and gate way out at the input and at the output is used for data type conversion, i.e. from floating point to fixed point data type. The oscilloscope block is used to display and analyze the output waveform. The system design parameters for the digital modulation system are shown in figures 4.23 below.

4.3.3.1 Design Parameters for Digital Modulation System

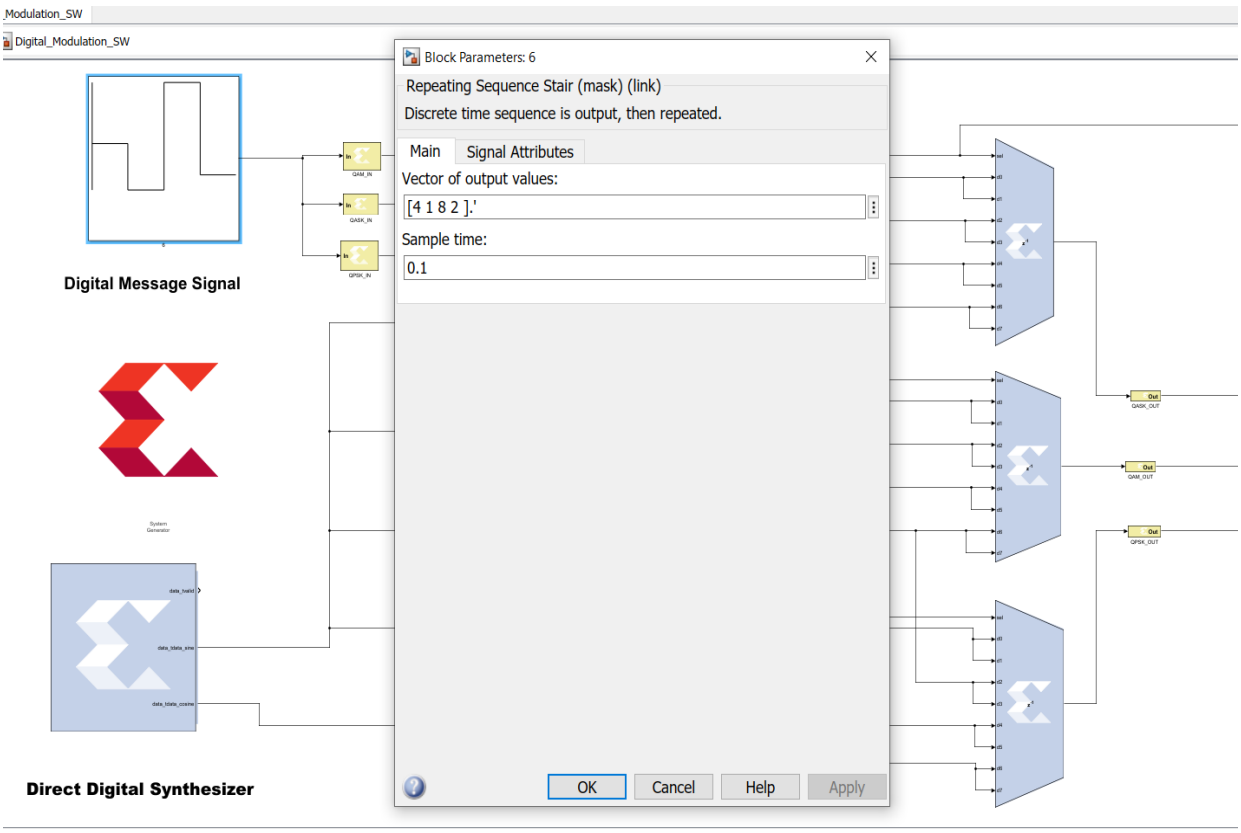

Figure 4.23: Design parameters for the digital message signal

The carrier signal is commonly shared between the analog modulation systems therefore the parameters value are the same.

4.3.4 System Design for multi-mode modulation techniques

Based on the system model the multi-mode modulation system is designed by combining the above analog and digital modulation systems. Figure 4.24 shows the designed system for the multi-mode modulation system.

When designing this multi-mode modulation system using FPGA, design parameters need to be considered. The parameters are chosen based on the system's

specific requirements and desired performance. The design parameters and its values are given in table 4.2. The parameters are selected based on the following assumption.

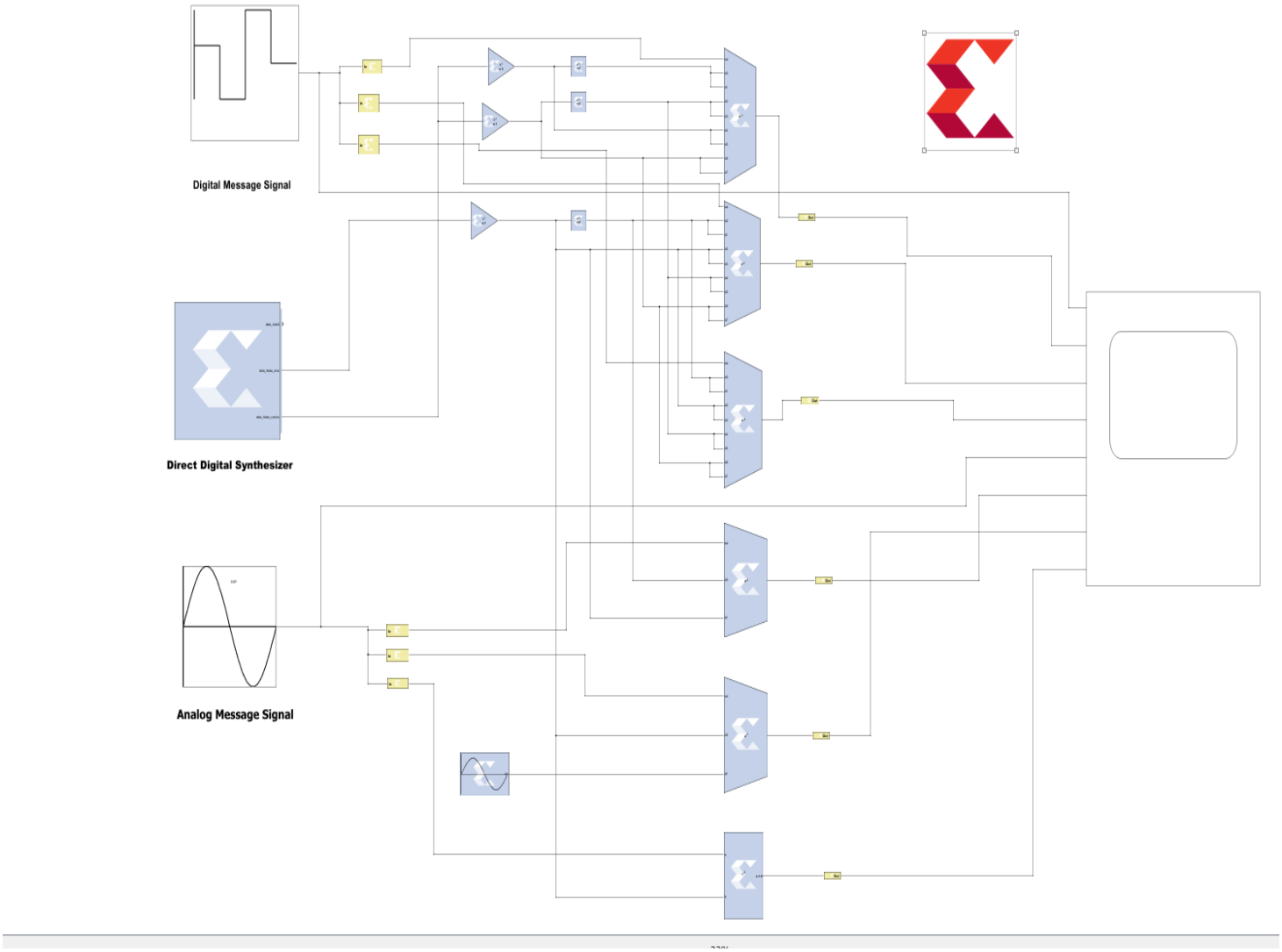

Figure 4.24: System Design for the multi-mode modulation system

- ★ **The Modulation schemes implemented:** the selected modulation schemes that we implement on FPGA module. This includes (QASK, QPSK, QAM) and analog modulation schemes (AM, FM and PM) that need to be integrated.
- ★ **Data rate:** it define the maximum data rate that the FPGA module needs to support. This parameter influences the required processing speed, signal bandwidth, and resource utilization of the FPGA.
- ★ **Resource allocation:** if design is implemented on FPGA the FPGA's available resources must be considered, such as logic elements, memory, DSP blocks, and I/O pins.
- ★ **Power consumption:** we consider the power constraints of the system and we choose design parameters that enable efficient power utilization.
- ★ **The DDS clock rate**: should be high enough to provide the desired frequency resolution and spurious-free dynamic range. Which means $f_c >> f_m$ [34].

Table 4.2: System Design parameters

System Parameters	Value
f_c	10KHz
$f_m = f_{AM}, f_{PM}, f_{FM}$	5Hz
System Clock Frequency	5MHz
A_1	3
$A_3 = A_2$	1

4.4 Design for Hardware Implementation

Figure 4.25 shows the system generator token which is used to generate the hardware programming file, for the configuration of the system, to generate the IP (Intellectual property) block and finally to program the FPGA.

Using this system generator token the FPGA board type, the compilation method and the hardware description language are selected. Then after the IP block which includes the bitstream file for programming is generated.

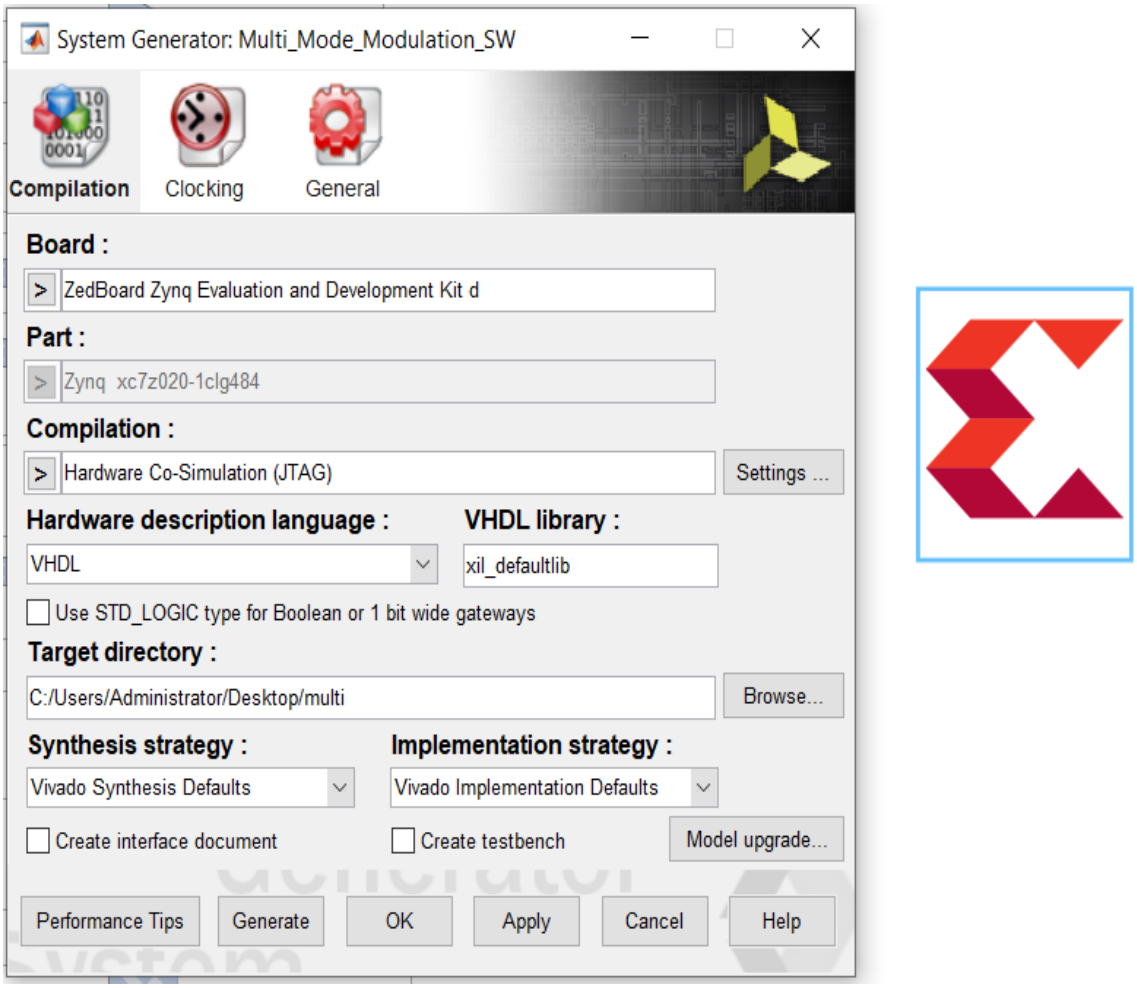

Figure 4.25: System generator token

4.4.1 Hardware Design for Analog Modulation System

Figure 4.26 is the IP generated for analog modulation system. It has 3 independent input and 3 independent output. Using this IP the analog modulation system for the hardware implementation is designed as shown in figure 4.27.

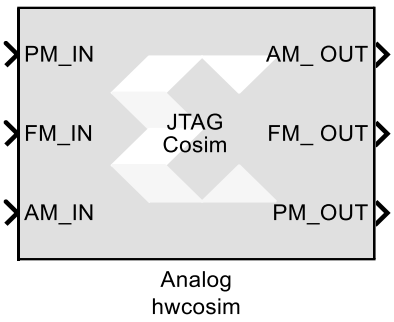

Figure 4.26: Analog modulator IP

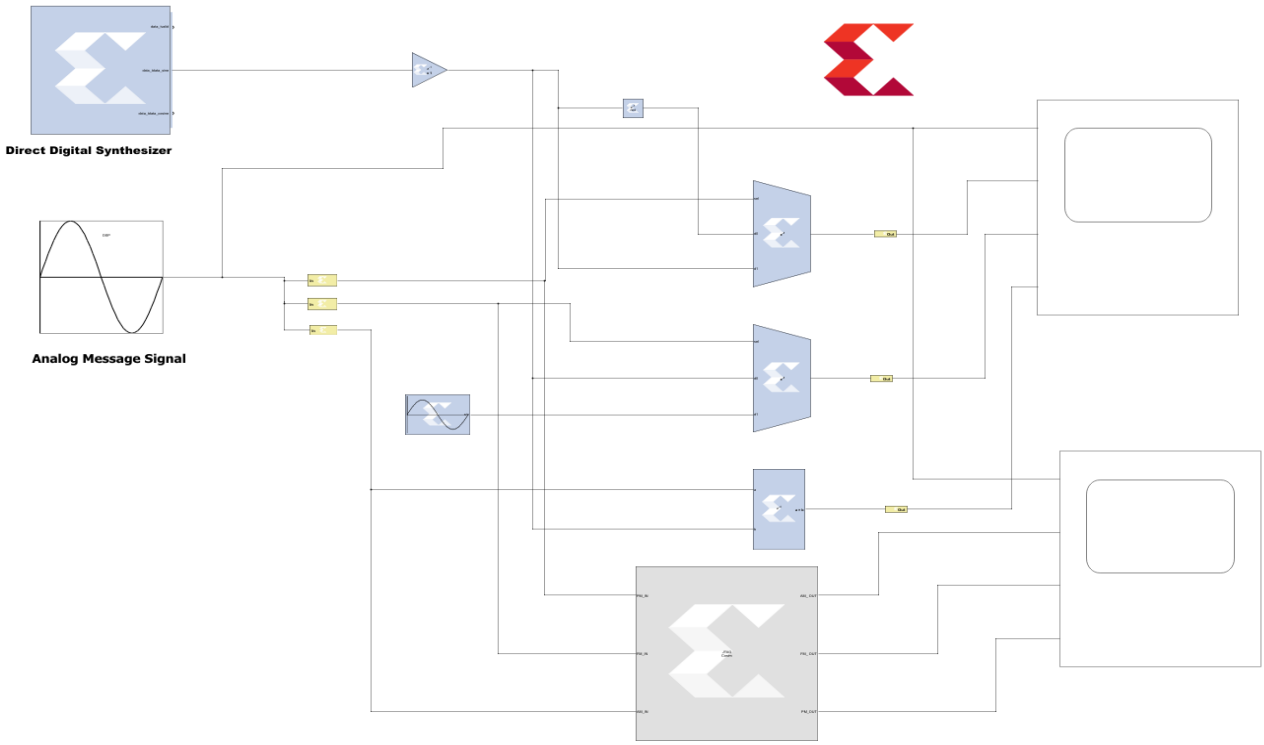

Figure 4.27: Analog modulation System Design for Hardware Implemntation

4.4.2 Hardware Design for Digital Modulation System

Figure 4.28 is the IP generated for digital modulation system. It has 3 independent input and 3 independent output. Figure 4.29 is the system design for the hardware implementation of digital modulation system using the generated IP.

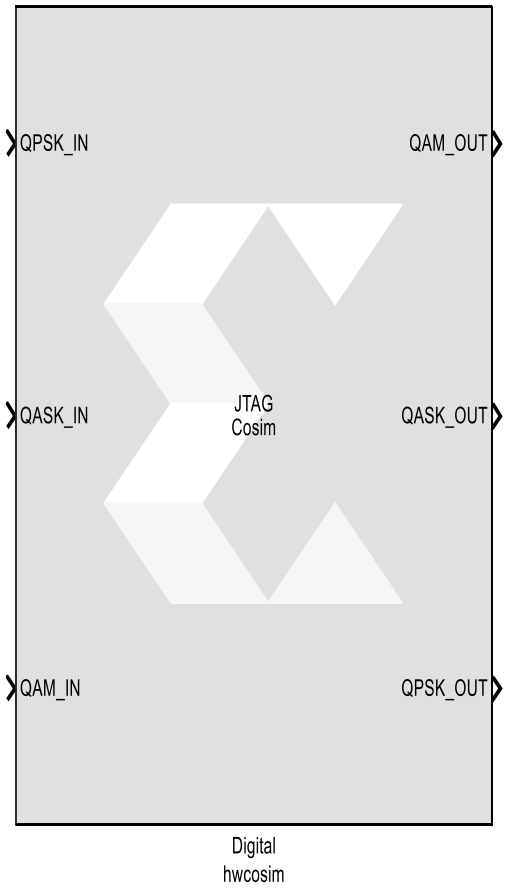

Figure 4.28: Digital modulator IP

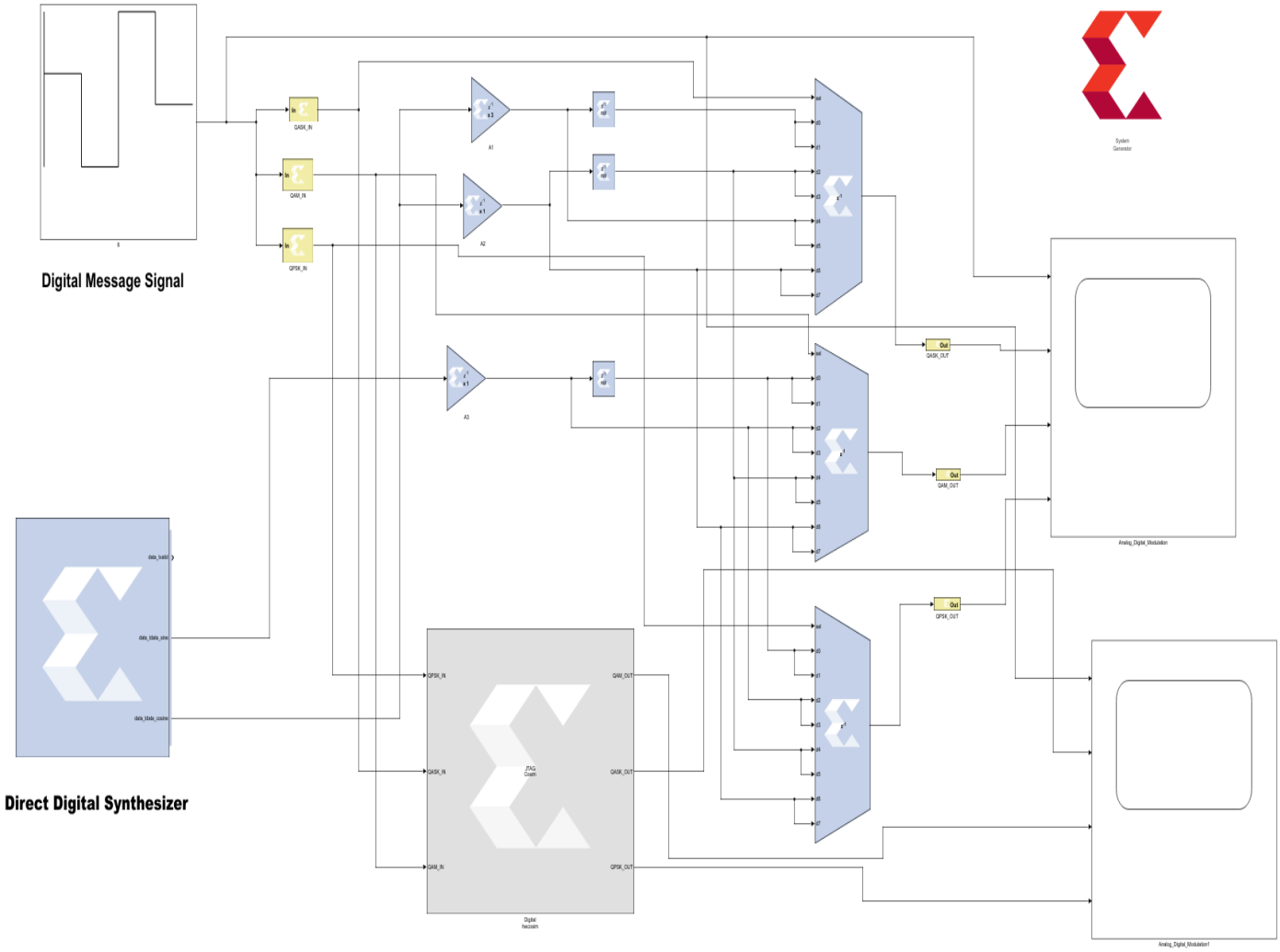

Figure 4.29: Digital modulation System Design for Hardware Implemntation

4.4.3 Hardware Design for Multi-Mode Modulation System

Figure 4.30 is the IP generated for the multi-mode modulation system. It has 6 independent input and 6 independent output. Figure 4.31 is the system design for the hardware implementation of multi-mode modulation system using the generated IP.

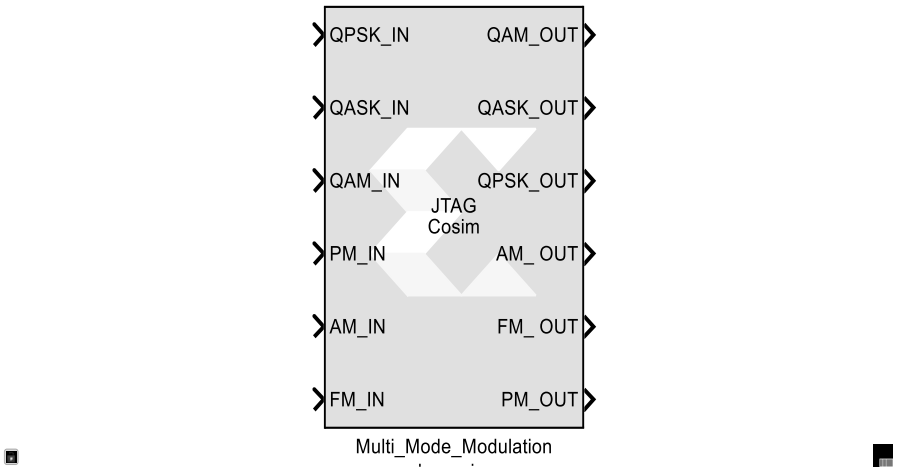

Figure 4.30: Multi-mode modulator IP

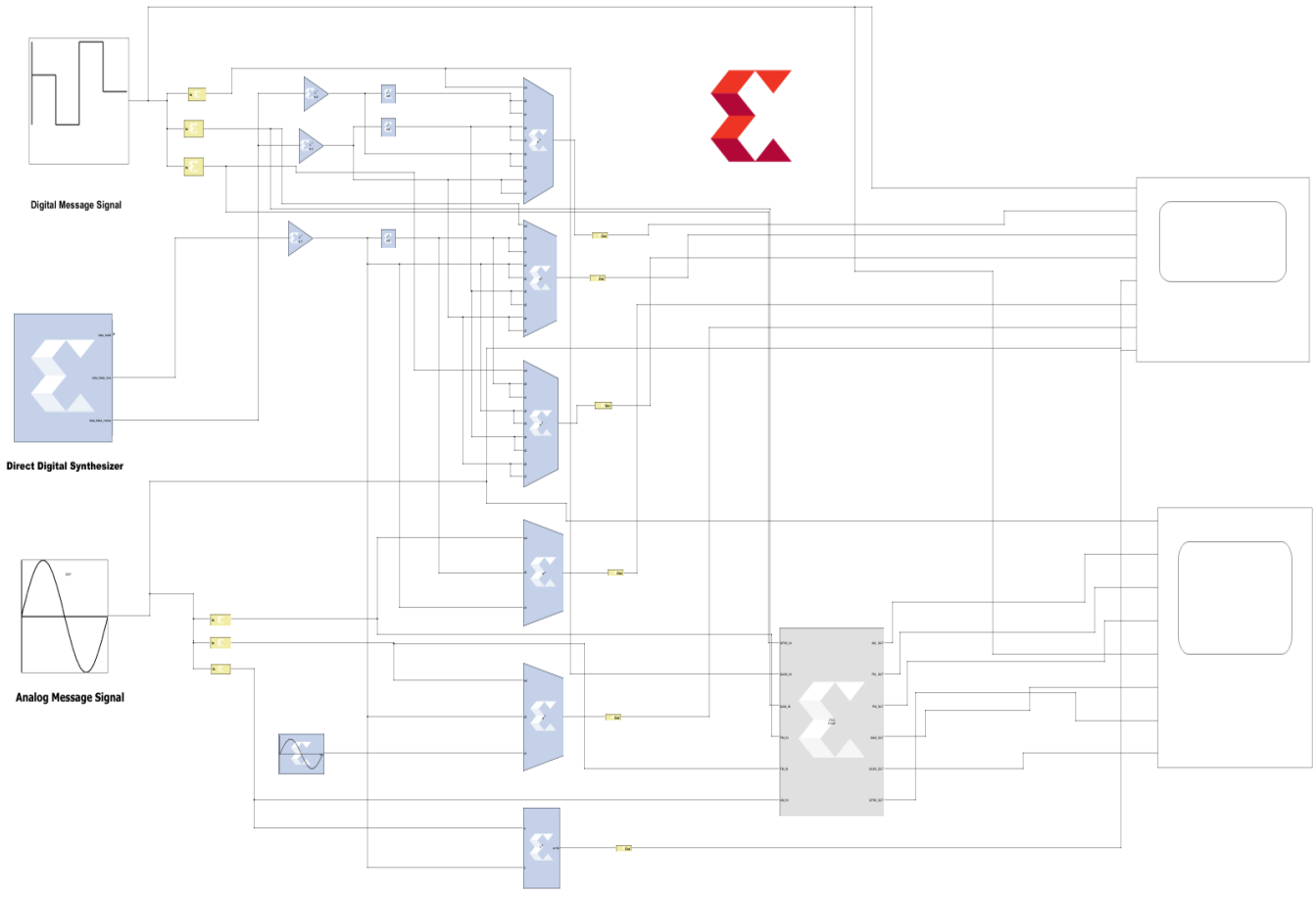

Figure 4.31: Multi-Mode Modulation System Design for Hardware Implemntation.

Chapter 5: Results and Discussion

5.1 Introduction

This chapter presents the hardware simulation results, the hardware implementation/Co-simulation results and the performance analysis of the implemented multi-mode modulation system. The results are presented in tables, graphs and figures form.

5.2 Hardware Simulation Results

Hardware simulation for FPGA is a technique for testing the performance and behavior of a digital circuit design prior to implemntation. It involves creating a virtual model of the FPGA circuit, including all its components and interconnections, and simulating its behavior.

The hardware simulation process is essential because it helps to identify and correct design errors and ensure that the final hardware implementation works as intended.

Hardware simulation for FPGA typically involves writing test benches that generate input stimuli and compare the output of the simulated design to expected results. The goal is to cover all possible scenarios that the design may encounter and ensure that it behaves correctly under all conditions.

5.2.1 Hardware Simulation Results for Analog Modulation system

Figure 5.1 is the hardware simulation result taken from the matlab Simulink oscilloscope for the analog modulation system. As we can see from the diagram the analog modulation systems are simulated as expected. We confirm this by the fact that the analog modulation that we know in theory is the same as the design result that we have prepared to implement on the hardware. As we can see in the figure, the first waveform in red is the message signal and is used as an input to the system. In The diagram what we see in pink color is the result for phase modulation, green is the result of frequency modulation and the red at the bottom is the result of amplitude modulation.

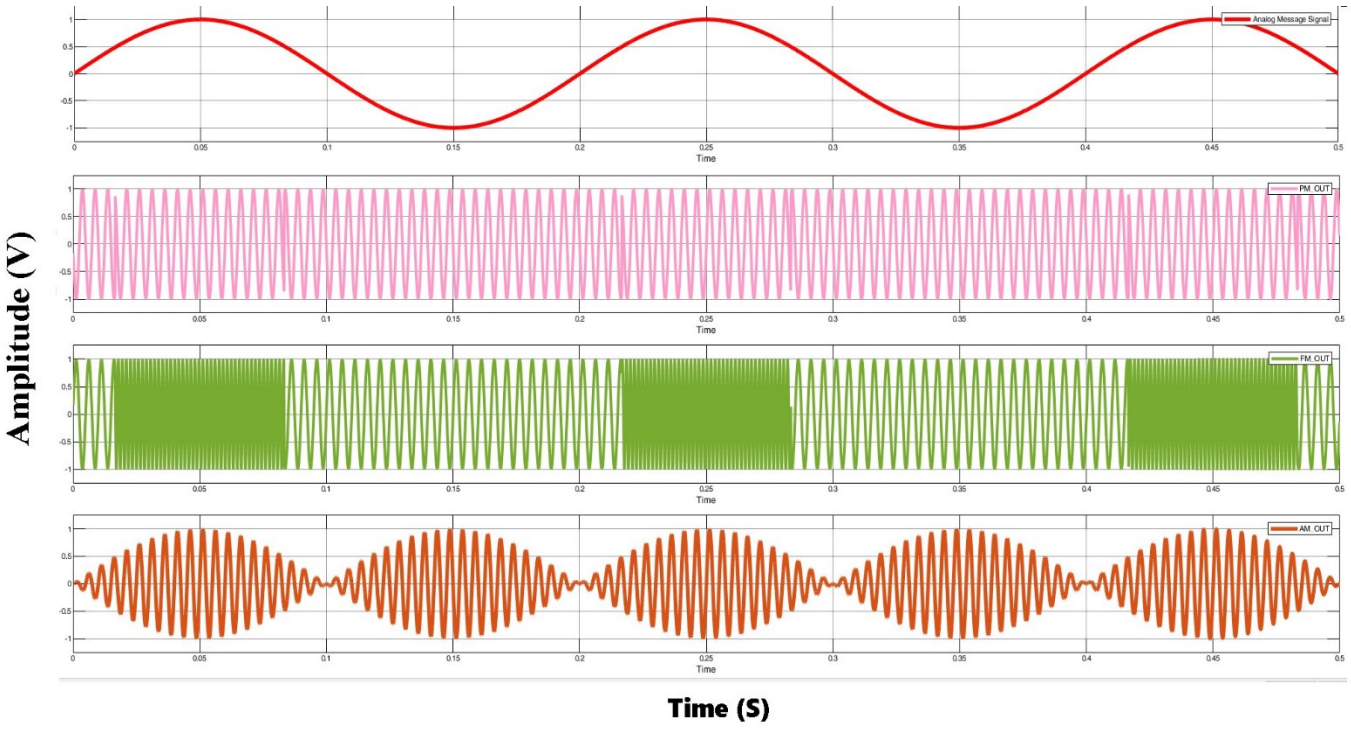

Figure 5.1: Simulation Result for Analog Modulation System

5.2.2 Hardware Simulation Results for Digital Modulation System

Figure 5.2 is the hardware simulation result taken from the matlab Simulink oscilloscope for the digital modulation system. As we can see from the diagram the digital

modulation systems are simulated as expected. We confirm this by the fact that the digital modulation that we know in theory is the same as the design result that we have prepared to implement on the hardware. As we can see in the figure, the first waveform in red is the digital message signal or symbols generated and is used as an input to the system. In The diagram what we see in blue color is the result for QASK modulation, green is the result of 4QAM modulation and the red at the bottom is the result of QPSK modulation.

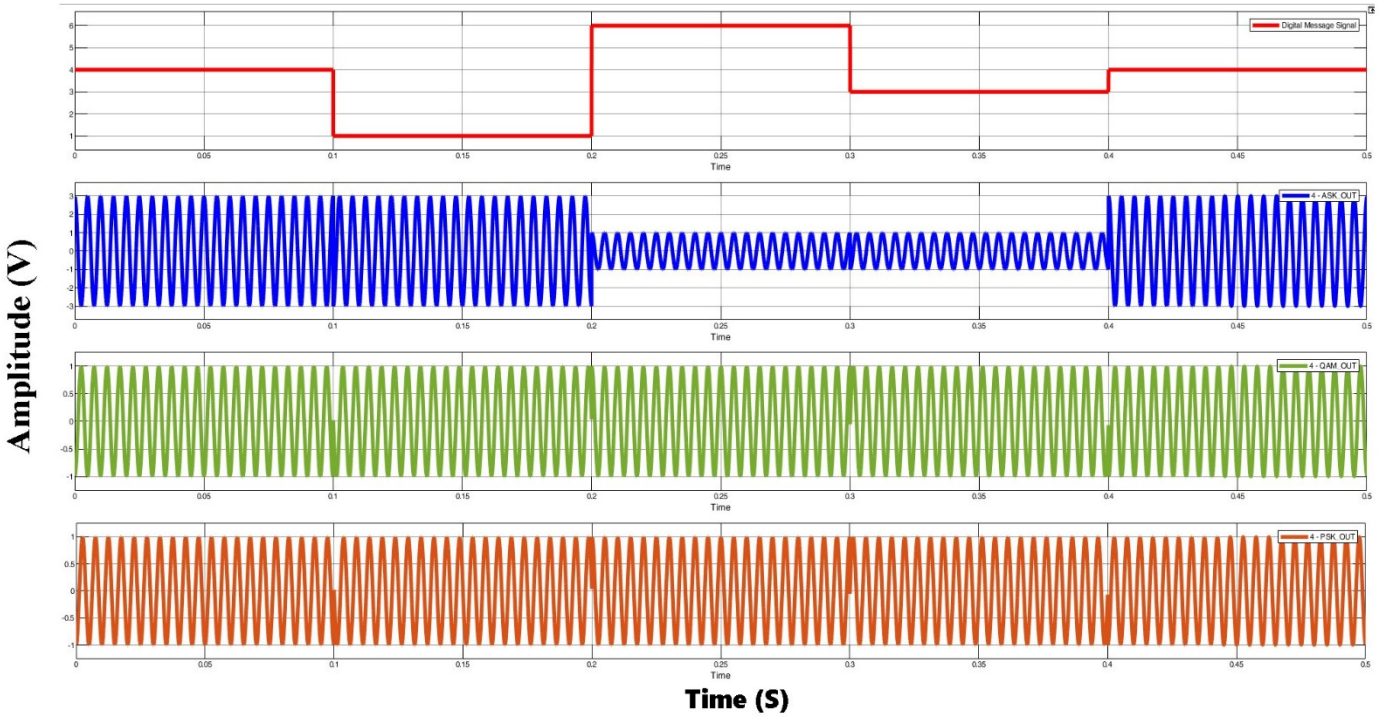

Figure 5.2: Simulation Result for Digital Modulation System

5.2.3 Hardware Simulation Results for Multi-Mode Modulation Techniques

Figure 5.3 is the hardware simulation result taken from the matlab Simulink oscilloscope for the multi-mode modulation system. This result is achieved by combining the above analog and digital modulation systems. This combination helps us to save resources for hardware implementation.

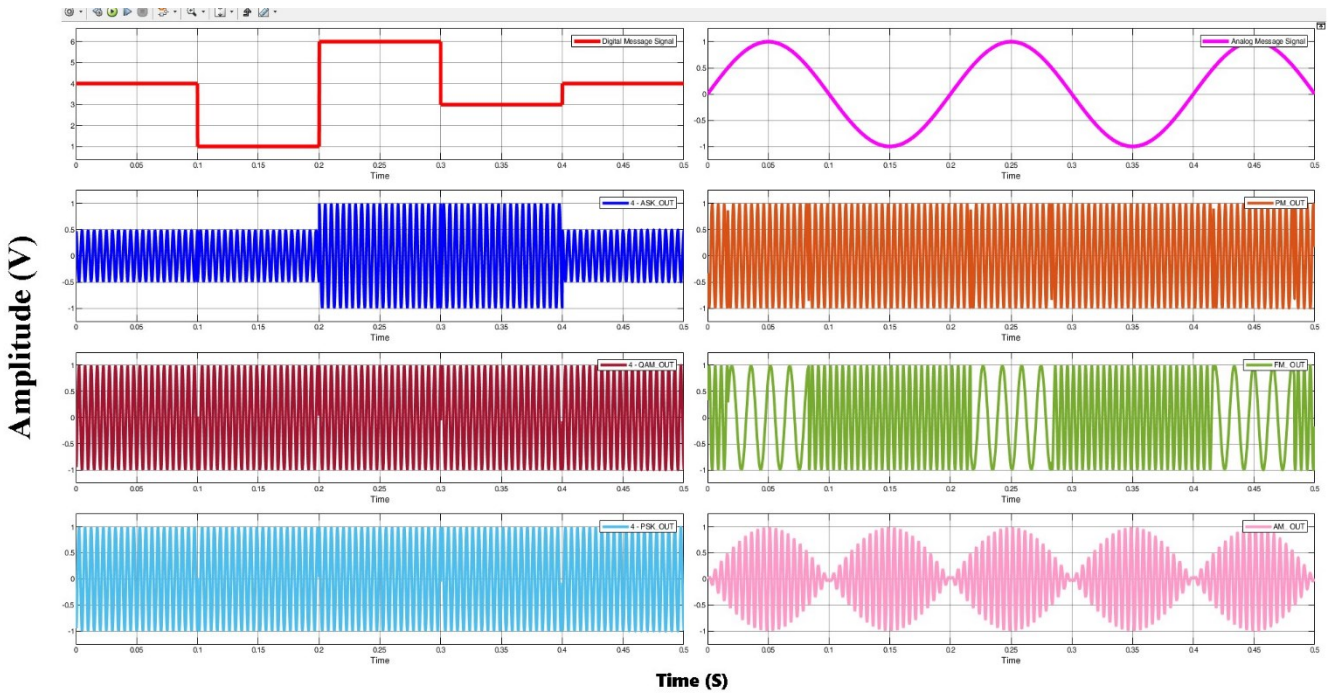

Figure 5.3: Multi-Mode modulation Simulation Result

5.3 Hardware Implementations Results

The hardware implementation of this thesis uses a Xilinx Zedbord FPGA. This board has all the components needed to develop OS - or real-time (RTOS)-based design, including Windows, Linux, Android, and others. Furthermore, the processing system and programmable logic I/Os are easily accessible to users through a number of extension connectors [36].

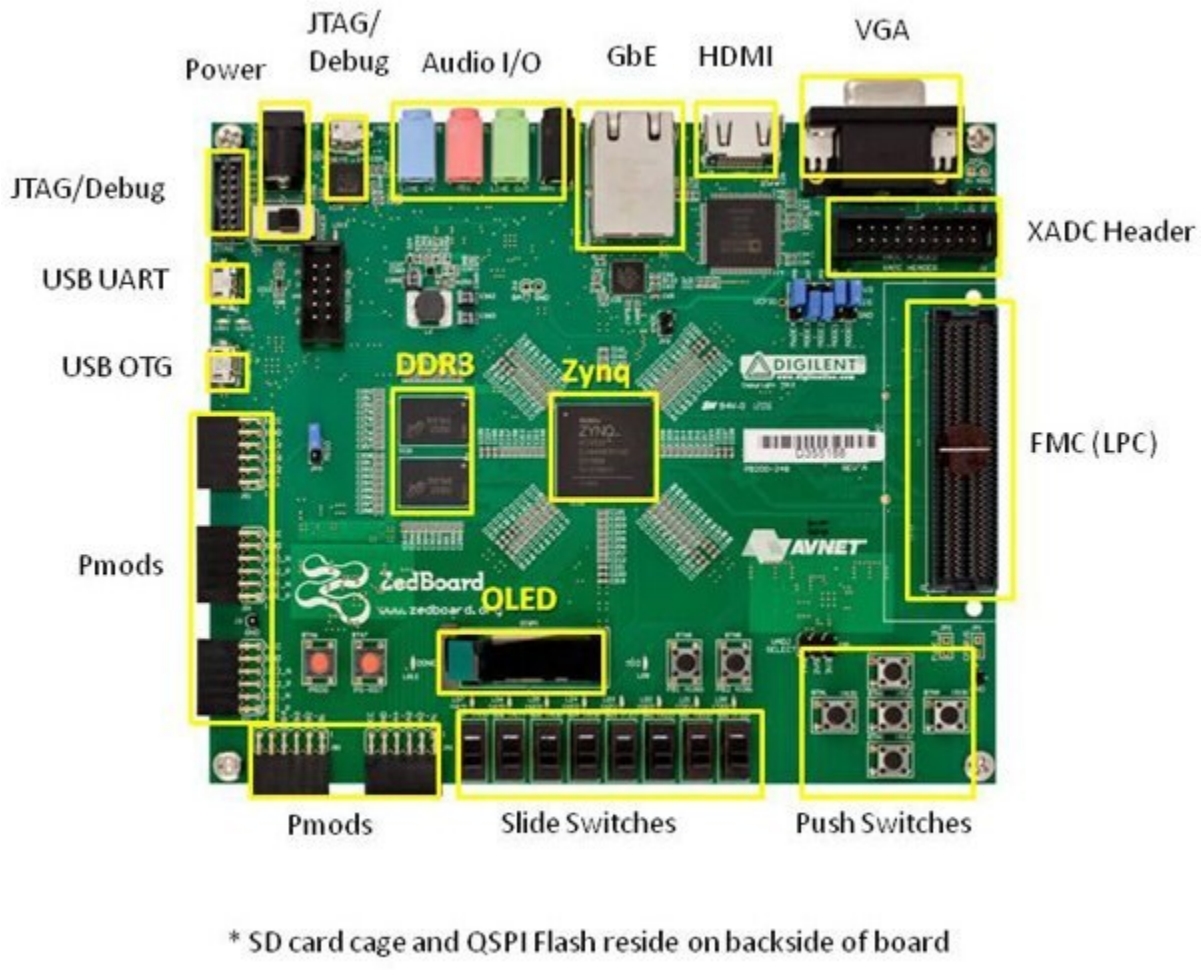

Figure 5.4: Zedboard FPGA [36]

5.3.1 Hardware Implementations Results for Analog Modulation system

Figure 5.5a and 5.5b show the results of the hardware implementation of analog modulation system. The LED which is shown in blue color in figure 5.5a is blinking when the FPGA is programmed. JTAG programming method is used to upload the bitstream to the FPGA. As can be seen, the result of hardware simulation in previous figure 5.1 and the result of hardware implementation in figure 5.5b are the same, so we can say that the implementation is successful. Therefore we can confirm that our design save resources for the implementation.

Figure 5.5 (a): Analog Modulation system result after Implementation

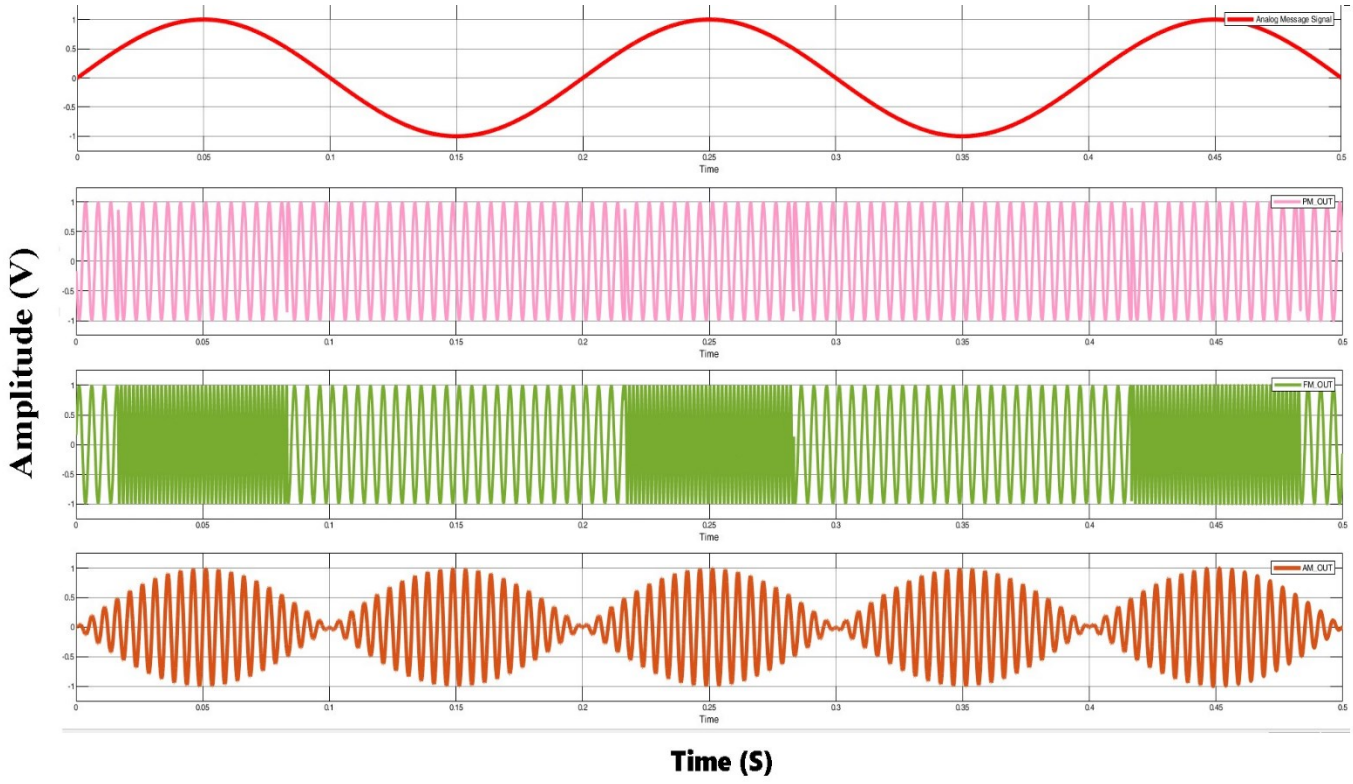

Figure 5.5 (b): Oscilloscope output for Analog modulation System after Hardware Implementation

5.3.2 Hardware Implementations Results for Digital Modulation System

Figure 5.6a and 5.6b show the results of the hardware implementation of digital modulation system. The LED which is shown in blue color in figure 5.6a is blinking when the FPGA is programmed. JTAG programming method is used to upload the bitstream to the FPGA. As can be seen, the result of hardware simulation in previous figure 5.2 and the result of hardware implementation in figure 5.5b are the same, so we can say that the implementation is successful. Therefore we can confirm that our design save resources for the implementation.

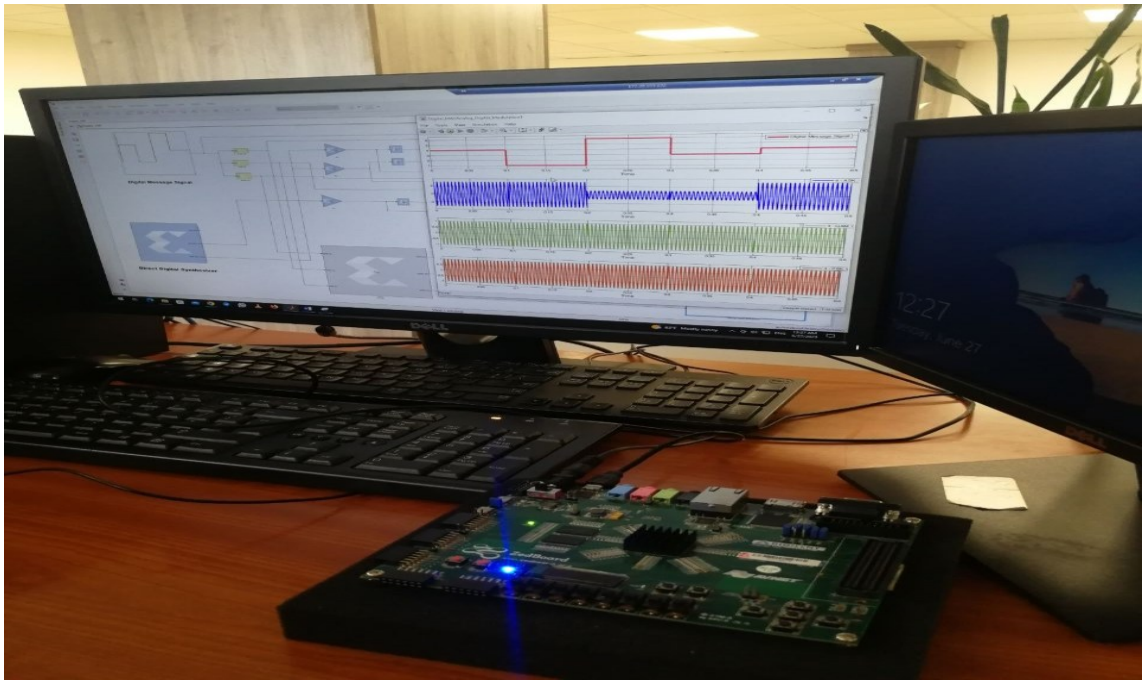

Figure 5.6(a): Digital Modulation system result after Implementation

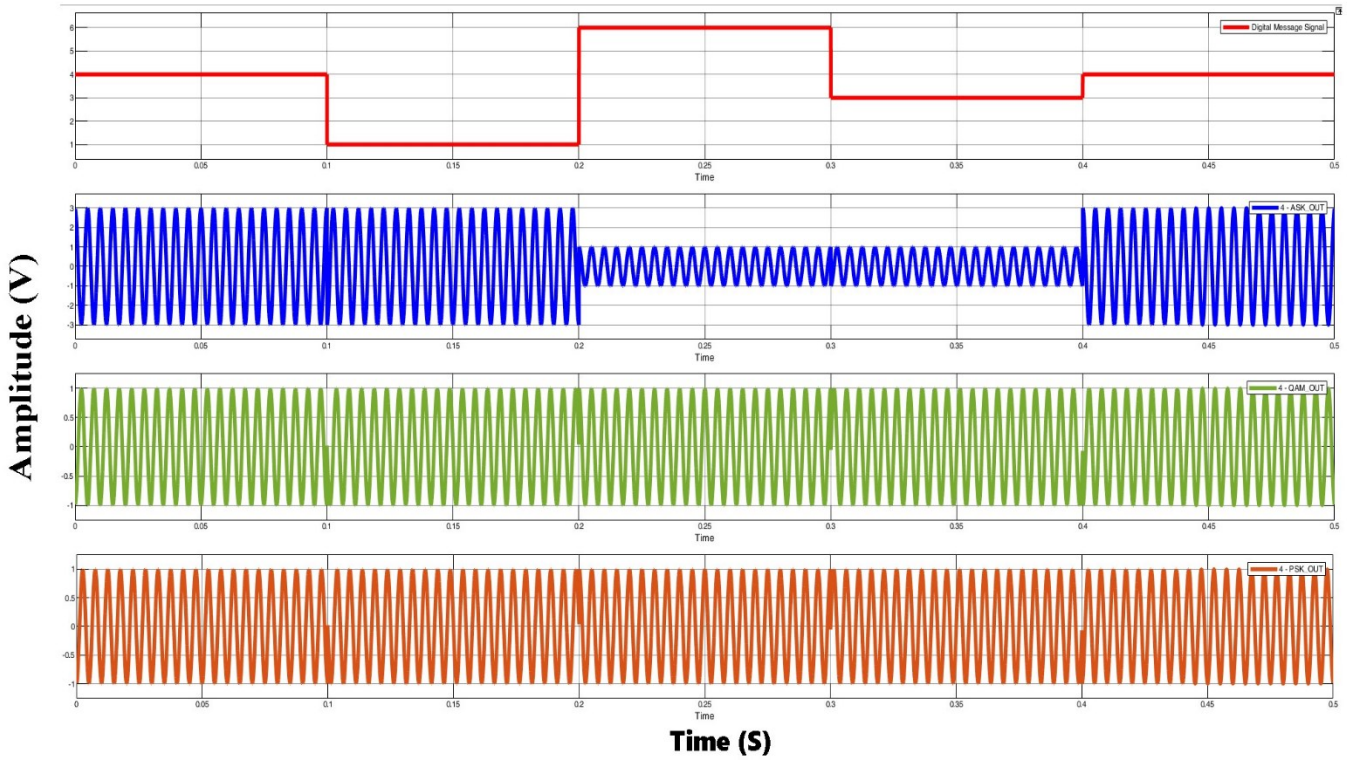

Figure 5.6 (b): Oscilloscope output for Digital modulation System after Hardware Implementation

5.3.3 Hardware Implementations Results for Multi-mode Modulation System

Figure 5.7a and 5.7b show the results of the hardware implementation of the multi-mode modulation system. The LED which is shown in blue color in figure 5.5a is blinking when the FPGA is programmed. JTAG programming method is used to upload the bitstream to the FPGA. As can be seen, the result of hardware simulation in previous figure 5.3 and the result of hardware implementation in figure 5.5b are the same, so we can say that the implementation is successful. Therefore we can confirm that our design save resources for the implementation.

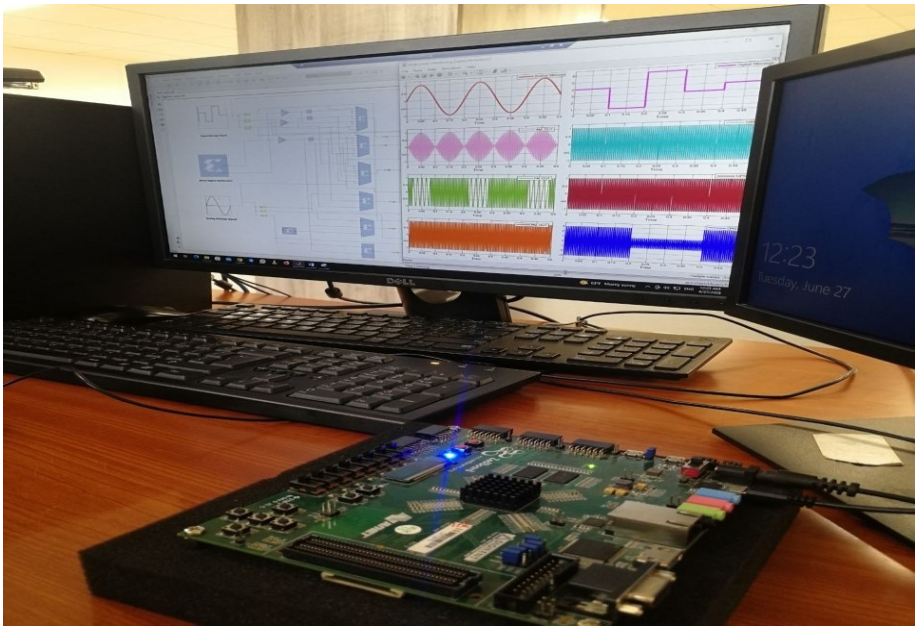

Figure 5.7(a): Multi-Mode Modulation system result after Implementation

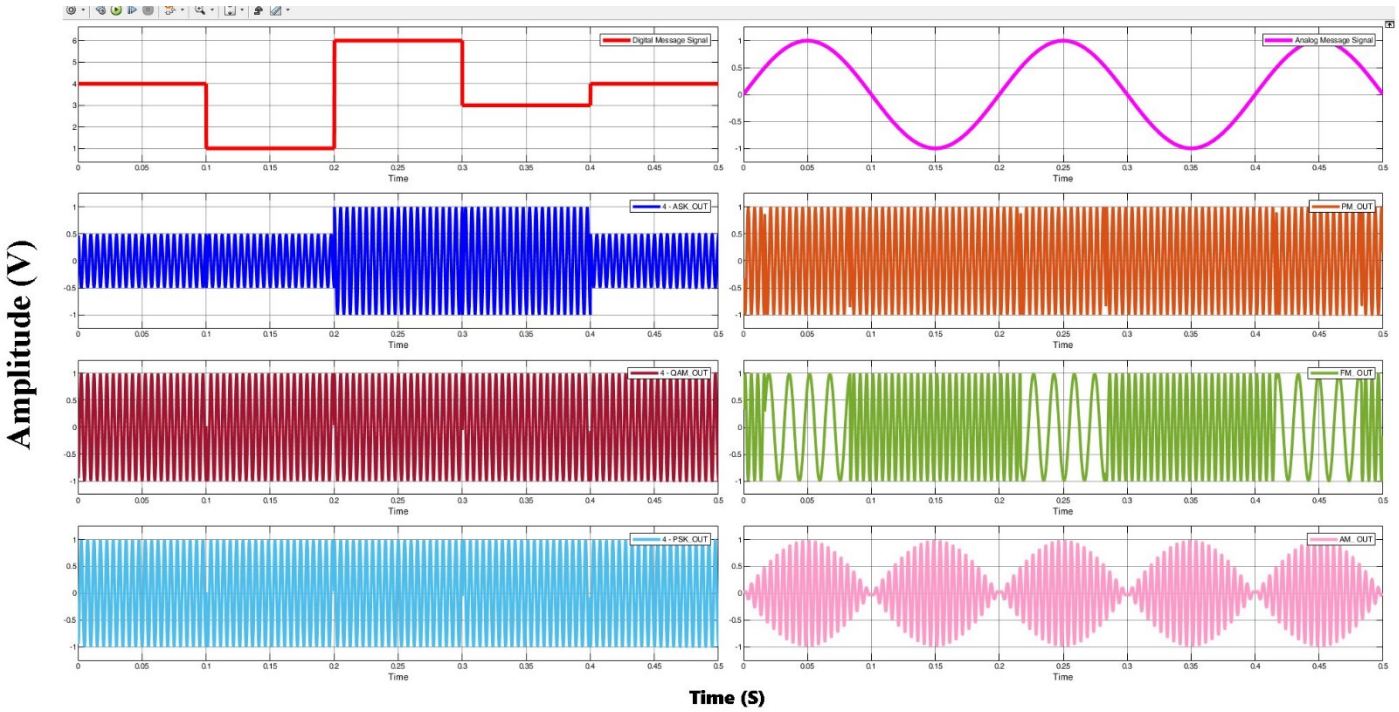

Figure 5.7(b): Oscilloscope output for Multi-Mode Modulation system after Hardware Implementation

5.4 Resource Utilization Analysis

Figure 5.8 is a screenshot taken from Vivado software after hardware implementation which shows the total power utilized by the implanted system. A total of 0.225W is used to implement the multi-mode modulation system on Zedboard FPGA. This is much better than previous research results as we compared in the following charts and tables.

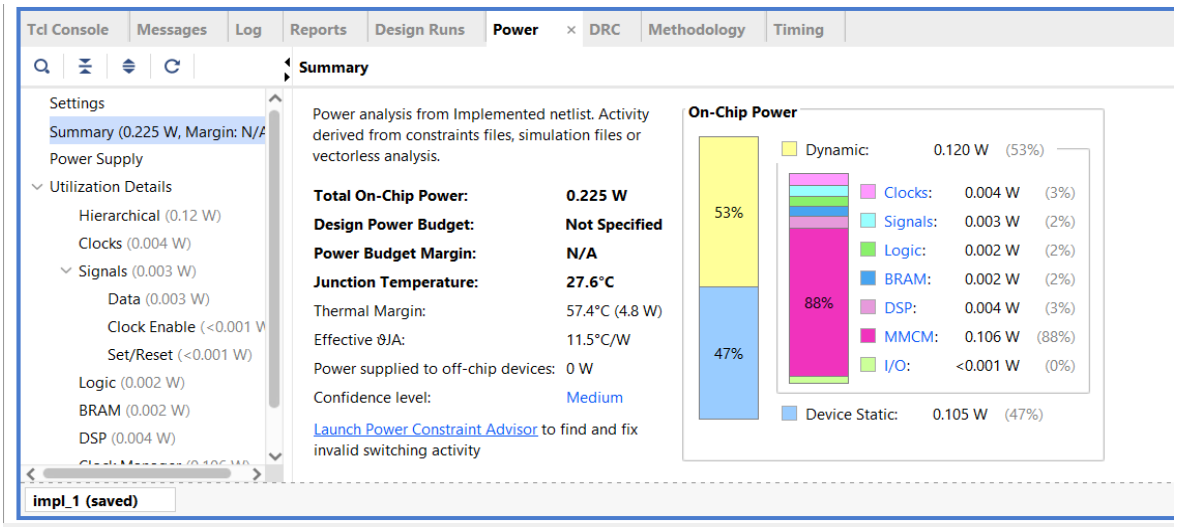

Figure 5.8: Total power consumption utilized by the system

Table 5.1 shows the Resource Utilization Summary of the implanted system. The total number FPGA resources such as Look Up Table (LUT), Flip Flop (FF) and Input/output (IO) port utilized by the implemented design are summarized in the table.

Table 5.1 Resource Utilization Summary

Resource	Utilization	Available	Utilization %
LUT	844	53200	1.59
LUTRAM	1	17400	0.01
FF	1345	106400	1.26
BRAM	3.50	140	2.50
DSP	9	220	4.09
IO	1	200	0.50
MMCM	1	4	25.00

5.5 Performance Comparison

Table 5.2 is the summary of comparison between the proposed method and the existing researches. As we can see in the table the proposed method significantly improves and save resources for the implementation.

Table 5.2: Resource Utilization Comparison

Performance Comparison Metrics	**Implemented Modulation Techniques**						
	Existing Methods					**Proposed Methods**	**Saved Resources**
	[2015] ASK,FSK, BPSK,QPSK	**[2016] BPSK,ASK,FSK**	**[2017] BASK-BFSK-BPSK-DPSK**	**[2019] BASK-BFSK-BPSK-QAM**	**[2020] BASK-BFSK-BPSK-QPSK-AM-FM-PM**	**[2023] AM,PM,FM,QASK, QAM,QPSK**	
Total on chip power	1.797W	Resource utilization summary is not included in the document but the modulation techniques can share resources for implementation.	1.359W	Resource utilization summary is not included in the document but the author used control signal and demultiplexer of the FPGA which is extra resource usage.	-	0.225W	1.572W and 1.134W Power is saved by our design as compared to the research done in 2015 and 2017
LUT	90		69		12,160	844	11,316 number of LUT is saved as compared to the research done in 2017
FF	-		-		316	1345	
IO	170		140		402	1	
BRAM	-		-		-	3.5	
Junction Temperature	-		25.5 °C		-	27.6°C	

Figure 5.9 to 5.11 is the performance comparison graph between recent studies and the proposed method in terms of total power consumption, IO port and LUT usage. The graph clearly shows that the proposed method improves the utilization of FPGA resources for the hardware implementation.

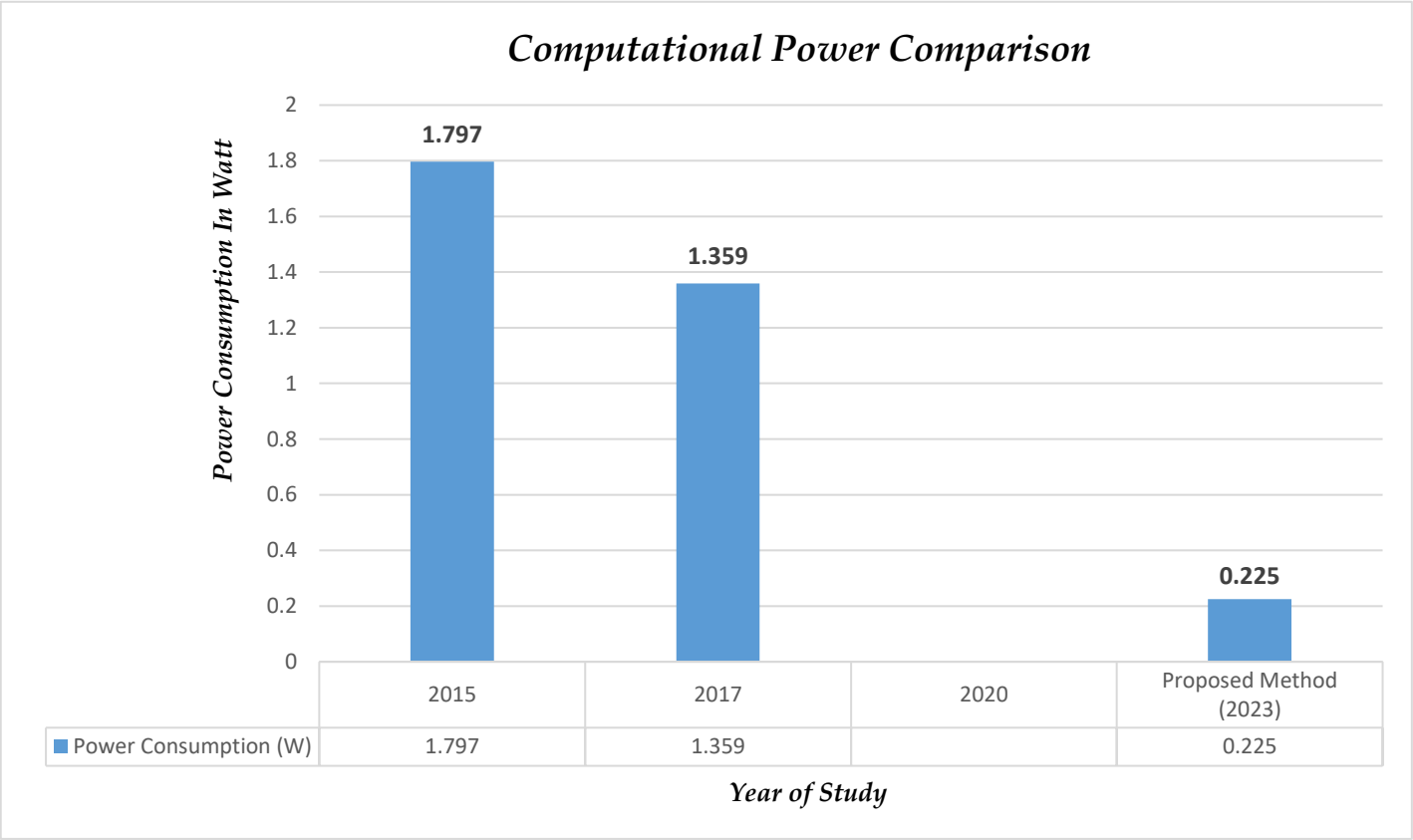

Figure 5.9: Computational Power Utilization comparison graph

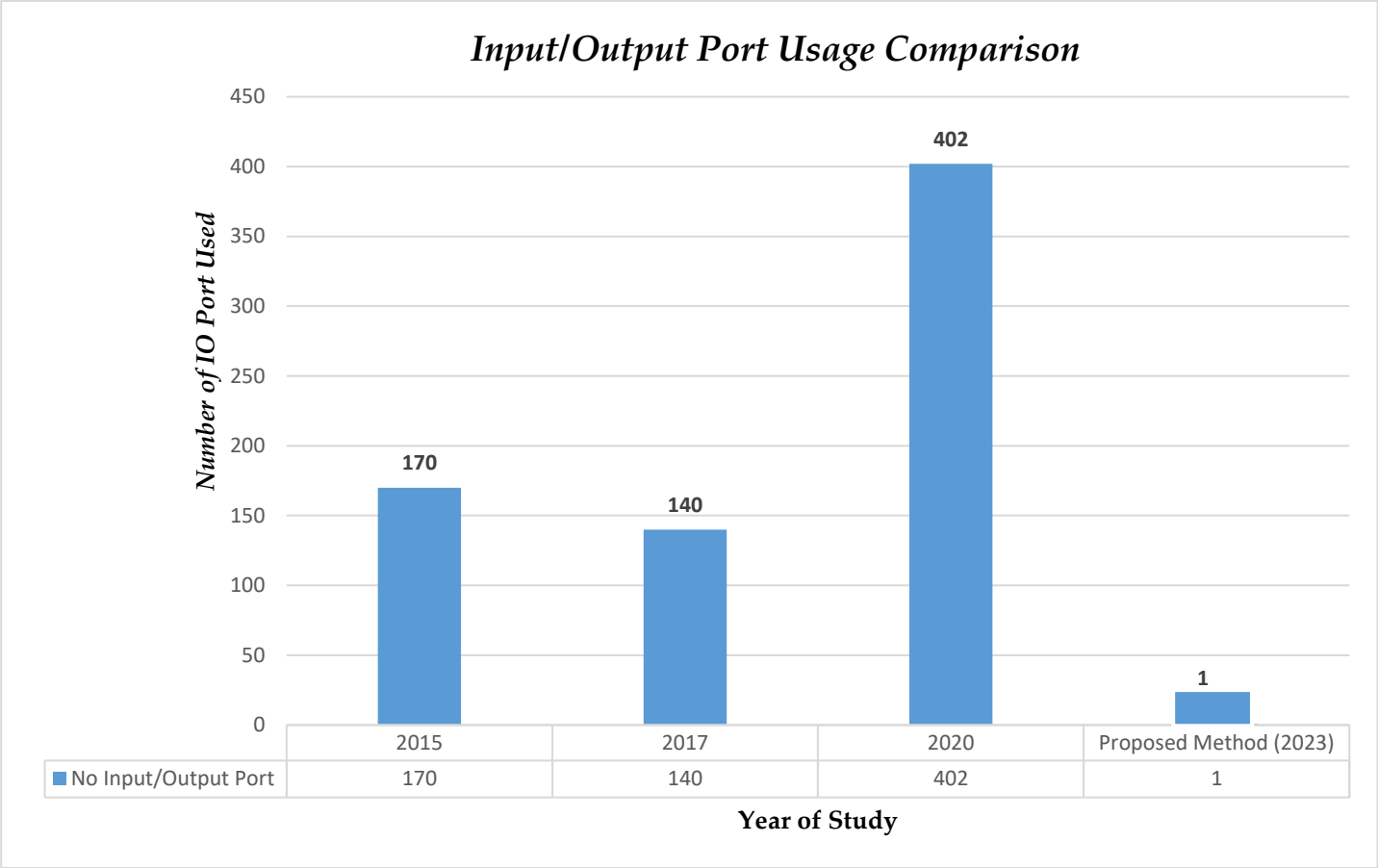

Figure 5.10: Input/Output Port Utilization comparison graph

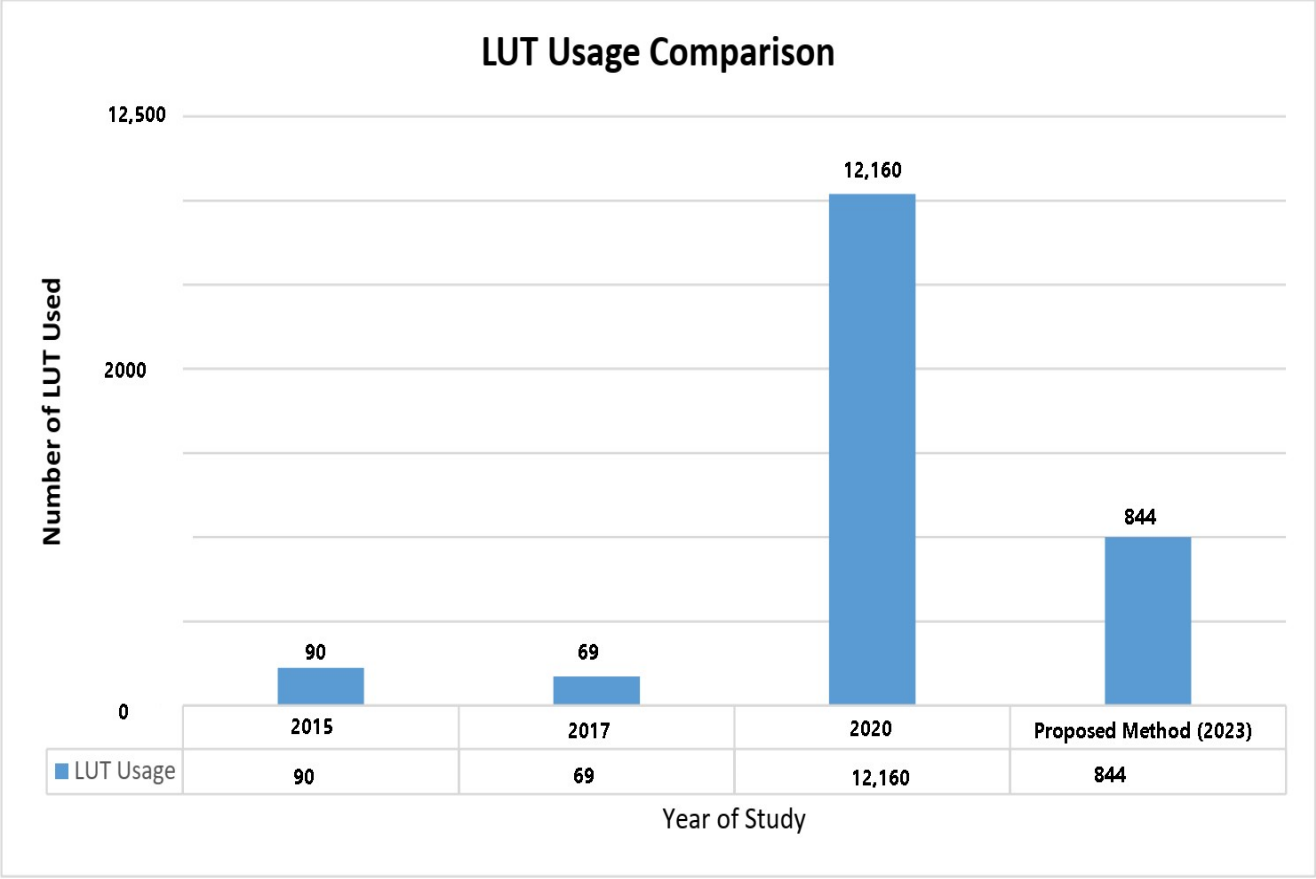

Figure 5.11: LUT Utilization comparison graph

Chapter 6: Conclusion and Future Work

6.1 Conclusion

This research has done the implementation of multi-mode modulation system. The results of hardware simulation and implementation is found to be similar, which means that the design is effective and the goal of saving resources is achieved. A total of 0.225W power, 844 number of LUT and 1 IO port is utilized by the implemented design. As we have tried to show with the numerical data in the table and graph above, this research got much better result than the work done in previous time.

This Multi-mode modulation system will be a crucial component of the communication system if parallel transmission is required. The multi-mode modulation methods that have been implemented would be usable in real-time applications like GPS, UMTS, DVB as well as FM and AM broadcasting. In terms of total number of Look Table (LUT), Flip Flops (FF), Input/Output (IO) port usage and total computational power needed, the proposed approach performs significantly better than the existing methods.

Generally, the performance a of multi-mode modulation techniques for SDR using FPGA was analyzed. The complexity of the FPGA implementation of the modulation techniques was also predicted as we tried to show when doing system modeling, and it was found that the higher-order modulation techniques will require more resources and computational power.

In conclusion, the choice of modulation technique and access scheme for an SDR system depends on the specific application requirements. The FPGA implementation of these techniques is complex and requires careful resource management to achieve optimal performance.

6.2 Future Work

Much work has gone into achieving the goal of this thesis; nonetheless, since it is always necessary to examine a particular study topic from various angles, the following themes are suggested for future study:

- Investigate the performance of more advanced modulation techniques such as 64QAM, OFDM, and DSSS in comparison to the tested techniques.
- Explore the use of different hardware platforms for implementing SDR such as DSPs and ASICs, and compare their performance with FPGA.
- Conduct experiments on a wider range of software-defined radio hardware to test the scalability of the proposed techniques.

References

[1] T. M. Patel, "Implementation of a software defined radio on FPGAs using system generator," University of Tennessee at Chattanooga, Chattanooga, 2010.

[2] M. J. a. M. J, "FPGA Implementation of Reconfigurable Modulation System," in IEEE, 100-Feet Ring Road, BSK Stage III, Bangalore 560085, India, 2015.

[3] H. A. Raya Kahtan Mohammed, "Implementation of digital and analog modulation systems using FPGA," Indonesian Journal of Electrical Engineering and Computer Science, vol. 18, pp. 485-493, April 2020.

[4] P. N. MURTHY, "Design and Simulation of Reconfigurable Digital Modulator using Vivado System Generator," in IEEE, Bangaluru, 2019.

[5] H. K. B. e. a. Bhaskara Rao Jammu, "FPGA Implementation of BASK-BFSK-BPSK-DPSK Digital Modulators using System Generator," in IEEE, -, 2017.

[6] R. S. P. B.K.V. PRASAD, "Implementation and Reconfiguration of Basic Digital Modulation Design Models," Journal of Theoretical and Applied Information Technology, vol. 90, pp. 208-218, 2016.

[7] S. Popescu et al., "Hardware Co-Simulation of the BPSK and QPSK Systems on FPGA," Politehnica University of Timisoara, Timisoara, Romania.

[8] J. R. M. Fernández, "Software Defined Radio: Basic Principles and Applications," Vols. Vol. 24, No. 38, pp. 79-96, January 2015.

[9] B. J. F. a. T. Ł. Mariusz Rawski, "Digital Signal Processing Designing for FPGA," Vols. vol. 20, no. 3, pp. 437-459, December 2007.

[10] Xilinx, "Model-Based DSP Design Using System Generator," June 12, 2020.

[11] R. K. rajesh.k, "Zedboard interfacing with AD-FCOMMS3-EBZ," Oct. 28, 2020.

[12] "Digital Modulation in Communications Systems – An Introduction," Hewlett-Packard Company, USA, 1997.

[13] "Applications of Analog and Digital Communication"Sri Eshwar college of engineering ,Comibatore

[14] R. Proloy, S. Ahmed, A. Hossain, "Comparative Analysis of Various Wireless Digital Modulation Techniques with different Channel Coding Schemes under AWGN Channel," International Journal of Computer Applications, vol. 161,no.3,pp.30-34, 2017.

[15] Vidyarani K R, Tejaswini S, Ankitha B M, Shalini D V, "Implementation of Analog Modulation Techniques using TMS320C6748 and LabVIEW," IOSR Journal of Research & Method in Education (IOSR-JRME , vol. 7, no. 4, pp. 46-50,2017.

[16] M. G. Ruppert, D. M. Harcombe, M. R. P. Ragazzon, S. O. Reza Moheimani, A. J. Fleming, "A review of demodulation techniques for amplitude-modulation atomic force microscopy," Beilstein Journal Nanotechnol, vol.8, p. 1407–1426, 2017.

[17] Vivado Design Suite User Guide: Model-Based DSP Design Using System Generator UG897 (v2020.2) November 18, 2020

[18] https://www.xilinx.com/products/intellectual-property/dds_compiler.html

[19] https://docs.xilinx.com/r/en-US/ug1483-model-composer-sys-gen-user-guide/Sine-Wave

[20] John Zhao, "Software Defined Radio with MATLAB" 2014

[21] https://www.wirelessinnovation.org/Introduction_to_SDR

[22] News from Rohde & Schwarz, Number 182 (2004/II)

[23] *Software-Defined Radio Handbook,* Pentek, Inc. One Park Way, Upper Saddle River, NJ 07458

[25] Introduction to FPGA Design with Vivado HLS www.xilinx.com 11 , UG998 (v1.0), July 2, 2013

[26] Shahnam Mirzaei, "*Design Methodologies and Architectures for Digital Signal Processing on FPGAs" June 2010*

[27] Dillinger, M., Sowerby, K. W., & Mitola, J. M. (2003). *Software Defined Radio: Concepts, Technologies, and Applications*. John Wiley & Sons.

[29] Reed, J. H. (2002). *Field Programmable Gate Arrays (FPGAs) for Software Defined Radio (SDR) *. Wiley-IEEE Press.

[30] "FPGA Implementation of a Low-Power PSK Modulator Using Xilinx FPGA" by N. Ahmad, O. Ullah, and A. Khan, published in the IEEE Transactions on Very Large Scale Integration (VLSI) Systems journal in 2023.

[31] 1. "FPGA Implementation of Digital Modulation Techniques: A Review" by A. Sharma and P. Singh, published in the International Journal of Advanced Research in Electronics and Communication Engineering in 2022

[32] "Implementation of Multi-Mode Modulation Techniques for Software Defined Radio Using Xilinx FPGAs: A Review" by B. Kumar and M. Sharma, published in the International Journal of Computer Science and Engineering in 2022

[33]"FPGA Implementation of Digital Modulation Techniques for IoT Applications: A Review" by E. Ahmed and F. Mohamed, published in the International Journal of Computer Science and Engineering in 2023

[34] https://digitalsystemdesign.in/fpga-implementation-step-by-step

[35] Xilinx's documentation and application notes on FPGA design and simulation

[36] https://www.xilinx.com/products/boards-and-kits/1-8dyf-11.html#