A Project Report

On

Implementation of Multi-Channel Time-to-Digital Converter using FPGA

Submitted in partial fulfilment of the requirement of

University of Mumbai

For the Degree of

Bachelor of Engineering

in

COMPUTER ENGINEERING

Submitted by

Aishwarya Anna Mathew
Pallavi Banagar
Roni Leonita

Supervised by

Ms. Shweta Tripathi (FCRIT)

&

Dr. Raghunandan A. Shukla (TIFR)

Department of Computer Engineering

Fr. Conceicao Rodrigues Institute of Technology
Sector 9A, Vashi, Navi Mumbai - 400703

UNIVERSITY OF MUMBAI

2020-2021

APPROVAL SHEET

This is to certify that the project entitled

"Implementation of Multi-Channel Time-to-Digital Converter using FPGA"

Submitted by

Aishwarya Anna Mathew 101701
Pallavi Banagar 101706
Roni Leonita 101745

In partial fulfilment of degree of **B.E. (Semester VII)** in **Computer Engineering** for term work of the project is approved.

Supervisors : 1. Ms. Shweta Tripathi

2. Dr. Raghunandan A. Shukla

Project Coordinator : Ms. Rakhi Kalantri

Examiners : 1. _____

2. _____

Head of Department : Dr. Lata Ragha

Date :

Place :

Declaration

We declare that this written submission for B.E. Declaration entitled **"Implementation of Multi-Channel Time-to-Digital Converter using FPGA"** represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any ideas / data / fact / source in our submission. We understand that any violation of the above will cause for disciplinary action by institute and also evoke penal action from the sources which have thus not been properly cited or from whom paper permission have not been taken when needed.

Project Group Members:

1. Aishwarya Anna Mathew, 101701

_____

2. Pallavi Banagar, 101706

_____

3. Roni Leonita, 101745

_____

Abstract

The GRAPES-3 (Gamma Ray Astronomy PeV EnergieS phase-3) experiment located at Ooty, India is designed to detect and study various properties of cosmic rays using an array of air shower detectors and a large area muon detector. One of the properties of interest is the angle of the rays that enter the atmosphere. This can be calculated if the time intervals between the detection of rays at adjacent light sensors are known.

In this project, we propose a design that implements a multi-channel time-to-digital converter (TDC) to measure and transmit these time intervals. The hybrid counter designed for measuring the time interval uses a combination of the coarse counter and the fine counter technique. The coarse counter provides a longer measuring range whereas the fine counter provides the precision required in the concerned conditions. Thus, this combined design is efficient and suitable for attaining an accuracy of sub-nanoseconds.

The aforementioned design is implemented on a Field Programmable Gate Array (FPGA). FPGAs can be configured as per the developers requirements using a Hardware Description Language (HDL). The array of programmable logic blocks and memory elements in an FPGA allows a wide variety of configurations. This is greatly advantageous for this project as it provides: (1) high parallel processing capabilities (2) high flexibility for reconfiguration (3) less development time.

The embedded system designed to be integrated with the GRAPES-3 experiment must also be able to handle the large amount of data and the data rate during communication and processing. Thus, the FPGA-based TDC system proposed is finally integrated with a network module, WIZ830MJ that includes Ethernet and UDP protocol, to facilitate smooth transmission of data to and from the end user.

Contents

List of Figures

Listings

Chapter 1

Introduction

1.1 Background

On a daily basis, the earth's atmosphere is bombarded by cosmic rays consisting of high energy particles that travel from the sun or more distant places in the universe. After a primary energetic particle enter Earth's atmosphere, it collides with nuclei and molecules in the atmosphere and produces secondary particles, which share the original primary particle's energy. The secondary particles propagate further, decay and produce more particles. This creates a shower of particles, which are called extensive air showers. These showers mainly consist of leptons (muons, electrons, neutrinos) and gamma rays. [1] Thus, on arrival at ground level, the air shower generated by a single ray spans over a large area. The detection and observation of these air showers are conducted by placing large are muon detectors at ground level.

The observation of these diffuse Galactic gamma ray flux is a powerful tool to study cosmic rays in different regions of the Galaxy. The energy and angular distributions of the photons encode information about the density and spectral shape of relativistic particles in the entire Milky Way.[2]

The GRAPES-3 experiment at Ooty, India started as a collaboration between the Tata Institute of Fundamental Research, Mumbai, India and the Osaka City University, Osaka, Japan. It is now a collaboration between many institutes from India and Japan. The experiment is designed to study cosmic rays with an array of air shower detectors and a large area muon detector. Currently, there are observations being recorded from over 400 scintillators, covering an area of 25,000 m^2, that detect charged particles contained in extensive air showers produced by interaction of high energy cosmic rays in the atmosphere. The GRAPES-3 experiment aims to study the following: (1) The origin, acceleration and propagation of >1014 eV cosmic rays in the galaxy and beyond. (2) Existence of "Knee" in the

energy spectrum of cosmic rays. (3) Production and/or acceleration of highest energy (∼1020 eV) cosmic rays in the universe. (4) Astronomy of multi-TeV γ-rays from neutron stars and other compact objects. (5) Sun the closest astrophysical object, accelerator of energetic particles and its effects on the Earth. [3]

1.2 Motivation

Precise time interval measurements are often required in various high energy physical experiments such as exploring subatomic level fine structure in fixed target experiments and collision experiments.

In this particular instance, it is used in a larger experiment for observing and studying diffuse galactic γ ray flux. The TDC is a necessary step for learning more about cosmic rays in different regions of the galaxy. The angular distributions of the photons are used to discover information about the density and spectral shape of relativistic particles in the entire Milky Way. Thus, this project plays a very small, but vital role in understanding various aspects of the universe from the tiny spec called Earth. This motivated us to choose this project.

1.3 Aim and Objective

This project aims to design and implement a multi-channel time-to-digital converter (TDC) using a Field Programmable Gate Array (FPGA). The designed hybrid counter will enable the precise measurement of time intervals between two signals (start and stop) with an accuracy of sub-nanoseconds. Furthermore, we aim to design an independent transmission module that uses Ethernet to connect the FPGA and the end user, thereby enabling effective and fast communication.

1.4 Report Outline

This report consists of 6 chapters. Starting with the introduction in Chapter 1, the background, motivation and objectives of the project are elaborated. Chapter 2 discusses the research concerning the topic. It includes an explanation of the technique adopted in the project and the other available techniques for precision time measurement and time-to-digital converters. This chapter also includes the literature survey where the review and comparison of 6 papers is included.

The proposed system of the project is explained in Chapter 3 with details about the problem statement, scope and components of the proposed system. Chapter 4 gives further details on the design of the system and its requirements. The results of the modules of the project implemented so far are presented in Chapter 5. Finally, Chapter 6 concludes the report and presents the future scope.

Chapter 2

Study Of the System

2.1 About the Technique

A TDC converts a time interval between different pulses into digital numbers. A TDC has two inputs, a start which arrives first and a stop which comes later. Difference between positive edges of start pulse to positive edge of stop pulse is called as input time interval. The resolution of this device is limited by the speed of the reference clock and it cannot be higher than a single clock period. For getting higher resolution, high clock frequency is needed which leads to major power dissipation. Constraints like on chip clock frequency of the particular device is the key factor in limit the application of this kind of architecture. Thus we make use of the tapped delay line technique to achieve resolution in the Time to Digital Converter [4].

A Tapped Delay Line (TDL) is a delay line with at least one "tap". A delay-line tap extracts a signal output from somewhere within the delay line, optionally scales it, and usually sums with other taps for form an output signal. A tap may be interpolating or non-interpolating. A non-interpolating tap extracts the signal at some fixed integer delay relative to the input. Thus, a tap implements a shorter delay line within a larger one.

A Tapped delay line is consisting of one latch chain and one buffer chain. Start signal is given to buffer chain and stop signal is given to the latch chain. multiplexers with dedicated routes can be used in FPGA for creating this kind of design [4].

The Hybrid Counter uses Carry4 primitive in the delay chain which is a carry logic already present in Vivado Design Suite.

2.2 Various Available Techniques

2.2.1 TDC Architecture Approaches

2.2.1.1 Analog TDC methods

- Time stretching method :
  In the time stretching method, the capacitor is discharged and recharged by two current sources, where the recharging time is K times longer than the discharging time. Hence, the time resolution can be improved by the stretching factor In the time stretching method, the capacitor is discharged and recharged by two current sources, where the recharging time is K times longer than the discharging time. Hence, the time resolution can be improved by the stretching factor

- Time-To-Amplitude conversion method:
  TDC based on a time-to-amplitude conversion method is realized by combining time-to-amplitude conversion and analog-to-digital converters (ADC). Through careful design and layout, analog TDC can obtain good resolution (about 8 ps) at the expense of high power consumption.

However, analog TDCs often suffer from large temperature drift and poor stability. Moreover, the area-consuming devices in analog TDC also hinder its implementation in integrated circuits.[5]

2.2.1.2 Digital TDC methods

With the development of integrated circuit (IC) technologies, recent works have concentrated on all-digital TDCs which employ standard CMOS technology to realize on-chip TDC. Due to their low area consumption and high conversion rate, TDCs based on application-specific integrated circuits (ASIC-based TDC) and field programmable gate arrays (FPGA-based TDC) have been widely used in recent years.[5]

- ASIC-based TDC:
  Compared with FPGA-based TDC, ASIC-based TDC has the merits of fully customized circuits and precise control of the internal propagation delay. The resolution obtained by ASIC-based TDC can be 1–2 ps. However, ASIC-based TDCs usually suffer from high development cost and long time-to-market.[5]

- FPGA-based TDC:
  FPGA devices have become very popular for rapid system prototyping, logic emulation, and reconfigurable computing because of their much lower manufacturing cost and shorter development time.[5]

2.2.2 For Coarse Measurement

2.2.2.1 Coarse Counter

It is the simplest counter which is edge-triggered and counts the number of clock signals that occurred between the start and stop signal. It considers the clock edge trigger that occurred after the start signal and the clock edge trigger that occurred after the stop signal, hence not an accurate counter in cases where the start and stop signal do not occur at the edge triggers of reference clock.

2.2.2.2 Statistical Counter

Since start, stop and clock signals are asynchronous, there is a uniform probability distribution of the start and stop signal-times between two subsequent clock pulses. This detuning of the start and stop signal from the clock pulses is called quantization error. For a series of measurements on the same constant and asynchronous time interval one measures two different numbers of counted clock pulses. Measuring a time interval using a coarse counter with the averaging method is relatively time consuming because of the many repetitions that are needed to determine the probabilities.

2.2.3 For Fine Measurement

2.2.3.1 Single Delay Line

In this architecture, buffers are used as a delay cells. The TDC based single delay line is suitable for technology scaling because of its fully digital organization [9]. Six buffers as delay cells are embedded into a charge-pump Delay Locked loop (DLL). Six delayed clocks with different delay can be generated by this method. A hit signal is mainly works as a sampling clock. The states of the six clocks will be samples into hit register when edge of the hit signal is positive [4].

Resolution of TDC based on single delay line is mainly dependent on clock period as well as number of buffers embedded into delay line. In FPGA clock period is limited by technology used. So by increasing number of delay elements can be useful. But Minimum number of delay cells and

its delay time is also technology dependent. Moreover, there is also a problem of mismatching a delay cells in this method which does not allow integrating large number of delay cells. So, achieving higher resolution can be difficult. To improve the time resolution of this architecture, one can try to further divide the delay of the delay cells by using an array of delay lines or other techniques [4].

2.2.3.2 Array of Delay Line

For eliminating the issue of single delay line array of delay line can be used in which several Delay Locked Loops (DLLs) are employed in the array. Resolution is depends on time difference between delay cells in delay line. The reference clock is propagated by the array of DLLs. In this method, power dissipation and area will be high because use of several DLLs in array. For overcome this issue tapped delay line and vernier delay lines are widely used approaches [4].

2.2.3.3 Tapped Delay Line

Tapped delay line is consisting of one latch chain and one buffer chain. Start signal is given to buffer chain and stop signal is given to the latch chain. Delay of latch chain and buffer chain is different. In this architecture resolution is depends on difference of the delays between latch and buffer. This architecture is more area efficient than single as well as vernier delay line [4].

Achieving higher resolutions will require more effort in terms of optimization of delay element, placement and routing of delay cells in this architecture because the elements used are of different types [4].

2.2.3.4 Vernier Delay Line

The principle of the measurements originates from the Vernier ruler[9]. Two delay lines are required. As shown in above figure, vernier delay line is consist of one latch chain and two buffer chain. Latch chain is mainly used for holding the result. Start signal and stop signal both are given to the different buffer chains. By using the vernier method, the small time difference can be measured. To realize the TDC using vernier delay line, two DLLs should be employed. Thus, the synchronization of the multi-phase clock is very important in this circuit [4].

This architecture is more area efficient than single delay line and less area efficient than tapped delay line. Compared to the tapped delay line

TDC this method requires more area in FPGA, but achieving finer resolutions is easier as the delay elements used are of the same type [4].

2.2.3.5 Gated Ring Oscillator

Gated ring oscillator based delay line contains gated ring oscillator, many counters and arbitrary adder. This TDC operates only when the "Enable" signal is high level and stops when this signal is at low level. The outputs of the gated ring oscillator can be used as the clocks which drive the counter to counting numbers [9]. Counter will get reset when the enable signal is at low level. Binary adder will obtain the total number of all counters. And the measured time interval will be proportional to sum of counted numbers by binary counter [4].

2.2.3.6 Pulse Shrinking Delay Line

In pulse shrinking delay line, non-homogeneity is used to to create delay line. In this architecture, a Reset signal is used to ensure the Tout is at Low level at the beginning. The input time interval is shrieked with each cycle in the delay line with a fixed width. The output of the delay line is then feedback to the input to AND gate for circular operation. A high-resolution counter is driven by Tout and generates digital outputs which are proportional to the measured time interval. [4]

In this kind of delay line, inverters are used as delay cells and two types of inverters are necessary. First type can be the standard inverter with the gain of one unit. The second type is the inverter with the gain of β unit. Because of the difference of the input capacitance and equivalent ON resistance, for fixed time interval pulse will be shrinking. This shrinking delay interval depends on the dimension of the transistors, threshold voltage, power supply, temperature and other parameters. Pulse shrinking delay line can be implemented in FPGA as well as standard CMOS technology whose cost is much higher than FPGA. [4]

2.2.3.7 Vernier Ring Oscillator Delay Line

The important element of this design is two precise re-triggerable ring oscillators of very small difference in time periods (ΔT). These oscillators are used to determine the time difference between two pulses START and STOP. The Start and Stop pulses will enable the slow and fast oscillators respectively. The slow oscillator (time period T1) is triggered by START and the fast oscillator (time period T2) is triggered by STOP. In this technique, T2 less than T1 and STOP arrives after START, at some point

the rising edge of the fast oscillator will coincide with rising edge of the slow oscillator. This coincident will be detected by a phase detector. In this technique, slow oscillator and fast oscillator will serve as a clock coarse counter and fine counter respectively and the number of clock pulses (n1 and n2) will be counted. These counters will stop counting when the phase detector detects the phase coincidence of the oscillators. [4]

Benefit of using the oscillators is that it reduces the matching requirements on the delay buffers used in vernier delay line. This feature is very useful to reduce the temporal uncertainties of Vernier delay line based TDCs which is mainly caused by delay variation of buffers [21]. Disadvantage of this architecture is that it takes so many cycles to complete a single time interval (longer dead time). Conversion time is also high in this type of delay line than other delay lines which can detect time interval every cycle. [4]

2.2.3.8 Oversampling method

In this method, single delay line with four buffers which can give exactly 90 degree phase shift has been used [2]. Delay locked loop has been used to achieve exact 90 degree phase shift for this technique. Delay Locked Loop provides constant phase shift to output clocks. In figure 11, DLL has a Phase Detector which measures the phase error and to convert this information into voltage charge pump is required. Clock signal is given to the delay element. In this technique, four phase shifted clocks (0, 90,180, 270 deg) are generated [2]. This four shifted clocks works as a clocks which triggers four different counters. Resulting count has four times higher resolution than reference clock frequency. [4]

2.2.3.9 Tapped Delay Line using Multiplexer

The main advantage of vernier delay line over tapped delay line is that vernier delay line has same delay elements for giving delay in both delay lines rather, tapped delay line approach has different delay elements for both delay line which requires careful delay matching between latch and buffer chains. But tapped delay line is more area efficient as it uses fewer elements than vernier delay line [7]. By using only a same element as latch and buffer, it could possible to narrow down the delay difference. Therefore use of multiplexers instead of both latch and buffer is much useful [7]. Carry chain multiplexers with dedicated routes can be used in FPGA for creating this kind of design. [4]

The main advantage is that a multiplexer can be easily configured to

behave as a buffer as well as a latch, just by changing the way the output is connected. This allows the two parts of the chain to be very precisely matched. It can be also called as hybrid approach. The latch and buffer chains are based on the Tapped Delay Line approach, but the individual delay elements are realized using the same underlying hardware, thus giving delay matching similar to the VDL method. [4]

2.3 Related Works

2.3.1 Low resource FPGA-based Time to Digital Converter

Authors: Alessandro Balla, Matteo Beretta, Paolo Ciambrone, Maurizio Gatta, Francesco Gonnella, Lorenzo Iafolla, Matteo Mascolo, Roberto Messi, Dario Moricciani, Domenico Riondino

In this paper, they have developed and implemented a 32-channel TDC on a Xilinx Virtex-5 FPGA with a precision of 255 ps and low non-linearity effects. They system was implemented on the FPGA due to the necessity of a custom data acquisition system and the interface. The architecture implemented for the TDC used the Nutt interpolation method and 4xOver-sampling technique [6].

Differential NonLinearity (DNL) is the deviation of a single quantization step from the ideal value of 1 Last Significant Bit (LSB) and Integral NonLinearity (INL) is the deviation of the input-output characteristic from the ideal straight line. This particular paper was reviewed for studying the details concerning differential nonlinearity, integral nonlinearity and its testing and results.

2.3.2 Pico-TDC: a novel FPGA-based TDC with 2.2ps RMS timing resolution

Authors: T. Sui[1], Z. Zhao[2], S. Xie[1], Q. Huang[2], J. Xu[1], Q. Peng[3]
[1] Huazhong University of Science and Technology, Wuhan, China
[2] Shanghai Jiaotong University, Shanghai, China
[3] Lawrence Berkeley National Laboratory, Berkeley, USA

The purpose of their work was to develop a low-cost high-performance TDC to meet the requirement from the next generation sub-10ps TOF-PET camera. In the paper they have presented a new method called Pico-TDC method to construct the TDCs in FPGA. The uniqueness of the Pico-TDC method is to use single registers in the FPGA as low-precision

TDCs, and to combine many of those low precision TDCs to construct a high-precision TDC [7]. This paper was reviewed to explore the different possible approaches to increasing performance of the TDC.

2.3.3 264 Channel TDC Platform Applying 65 Channel High Precision (7.2 psRMS) FPGA Based TDCs

Authors: Cahit Ugur[1], Grzegorz Korcyl[2], Jan Michel[3], Manuel Penschuk[3] and Michael Traxler[1]

[1] GSI Helmholtz Centre for Heavy Ion Research, Darmstadt, Germany

[2] Jagiellonian University, Krakow, Poland

[3] Goethe-University, Frankfurt, Germany

In this paper the architecture of a 65-channel TDC implemented in a single FPGA is discussed in detail and the test results displaying some of the quality measurements are presented. The TDC applies the interpolation method for time measurements and uses Wave Union Launcher method to increase its precision. The TDC has a maximum precision of 7.2 ps RMS and $<$ 14 ps RMS on all channels [8].

In order to overcome the minimum pulse width limitation, a semi-asynchronous pulse stretcher is implemented which has been verified to allow a measurement of a pulse width $<$500 ps. This paper details the Semi-asynchronous stretcher for minimum pulse width limitation which was useful.

2.3.4 A Multichannel High-Resolution (less than 5 ps RMS between two channels) Time-to-Digital Converter (TDC) Implemented in a Field Programmable Gate Array (FPGA)

Authors: Eugen Bayer[1], Peter Zipf [1] and Michael Traxler[2]

[1] University of Kassel, Germany

[2] GSI Helmholtz Centre for Heavy Ion Research, Darmstadt, Germany

This paper presents a new design that uses dedicated carry-chains for time interpolation purposes and is able to perform two time-measurements in a single carry-chain per hit. In this design multiple ($>$2) measurements can be made in a single chain per hit reaching a time resolution of $\sim$2 ps RMS between two channels.

This paper was reviewed to study in detail some advanced interpolation techniques for the tapped delay line method. These techniques are used to improve the single channel resolution of a TDC channel based on the

TDL method. In all techniques an incoming start signal initiates multiple interpolations and produces N measuremnets which are averaged and recorded.[9]

2.3.5 Time-to-digital-converter based on multiple-tapped-delay-line

Authors: Dariusz Chaberski[1]
[1] Nicolaus Copernicus University, Poland

This article contains the description of the idea, operation, analysis, design and test-results of a time-to-digital converter (TDC) based on multiple-tapped-delay-line (MTDL). The use of multiple of relatively low equivalent-resolution (323 ps on average) TDLs allowed obtaining measurement equivalent-resolution of about 5:8 ps [10].

The idea of measurement consists in combining all tapped-delay-line results into one high-precision time-stamp for every hit. This paper contained elaborate mathematical analysis necessary to perform this operation.

2.3.6 A Coarse-Fine Time-to-Digital Converter

Authors: Ya-Qian Chen[1], Li-Ya Meng[1] and Xiao-Gang Lin[1]
[1] Key Laboratory of Optoelectronic Technology and System, P.R. China

In this paper, a high-precision TDC using a three-level conversion scheme was proposed. With the development of integrated circuit technology, a TDC with both high resolution and wide dynamic range is need [11]. Achieving only one target is not enough. Considering both precision and dynamic range, the Coarse-Fine TDC proposed in this paper included three stages same as in the nutt interpolation method. This paper was studied to better understand these 3 stages.

Chapter 3

Proposed System

3.1 Problem Statement

To develop a multi-channel time-to-digital converter (TDC) with the help of a Field Programmable Gate Array (FPGA) device. The designed TDC will enable us to measure time delays between any two signals (start and stop) with an accuracy of sub-nanoseconds. Also, integrate the FPGA-based TDC system with a network module, WIZ830MJ that includes Ethernet connection.

3.2 Scope

The scope of our project is limited to the generation of a 'Multi-channel Time to Digital Converter (TDC) integrated with Ethernet communication'. This means that our project scope is limited to the creation of Coarse Counter and Fine Counter along with their integration with ethernet communication. The above mentioned Coarse Counter will be used to measure longer time intervals (10 ns steps) and the Fine Counter will be used to measure smaller time interval (order of pico-seconds). This system can be deployed where precision timing measurement and large data transmission is required.

3.3 Proposed System

Our proposed system is a hybrid counter which uses the Nutt interpolation method for time measurement. The circuit for this will be developed on an FPGA board (Nexys A7 FPGA). Finally the communication in this systm wil be done using Ethernet.

3.3.1 FPGA (Field Programmable Gate Array)

Benefitting from the FPGA platform, the proposed TDC has superiorities in easy implementation, low cost, and short development time.

The semiconductor technology has been developing constantly. FPGA devices have become very popular for rapid system prototyping, logic emulation, and reconfigurable computing because of their much lower manufacturing cost and shorter development time. The best time resolution of previous FPGA-based TDCs is about 10 ps. Our FPGA-based TDC uses tapped delay lines (TDLs) based on carry logic. [5]

3.3.2 Nutt Interpolation Method

The hybrid counter is implemented using the nutt interpolation method. The main advantage of this is that it can measure long ranges using the coarse counter and also get resolution finer than the clock cycle period using the fine counter.

The time interval T to be measured is divided into three intervals as shown in Figure 3.1 [6]. One interval Δt_{12} (which may be quite long) is measured in real time by the coarse counter; the remaining two short intervals, Δt_1 and Δt_2 (at the beginning and at the end of the interval T), are measured by a high resolution TDC (fine counter). The fine counter measures the time between the STOP/START pulse and the next positive edge of the clock.

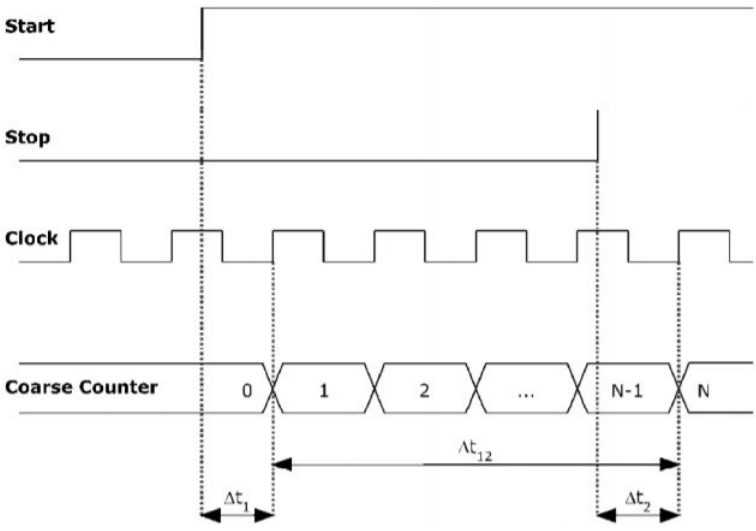

Figure 3.1: Nutt Interpolation Method. The time to be measured, T = Δt_{12} + Δt_1 - Δt_2

In our system, the coarse counter measures time ranges in the multipltes of 10ns. This is an important component as the length of the delay chain cannot be increased to very high timing intervals. The coarse counter is a simple counter that measures time intervals by simply counting the FPGA clock.

The fine counter is used to measure time intervals in the range of pico seconds. This timing measurement is difficult to achieve as the FPGA internal clock frequency is 100Mhz and thus direct counting at pico-second interval is not possible. This fine counter is implemented using tapped delay lines. The delay produced in each delay primitive is in the range of pico-seconds. The fine counter is designed to count the number of delay primitive.

3.3.3 Communication System

We are using WIZ830MJ, which is the network module that includes W5300 (TCP/IP and Ethernet PHY) and MAG-JACK (RJ45 with X'FMR) with other glue logics. No effort is required to interface W5300 and MAG Jack. We use this module to develop the data transmission system with high network performance. UDP protocol has been used for this purpose.

Chapter 4

Design Of the System

4.1 Requirement Engineering

4.1.1 Requirement Elicitation

The final system to be deployed is a time-to-digital converter implemented on a Field Programmable Gate Array (FPGA). To achieve this goal the primary requirement is the Nexys A7 FPGA board. For transmission of data between the FPGA and the end user (i.e. PC), a WIZ830MJ module, a PCB and an RJ45 Ethernet cable is required. The PCB designed connects all the data lines, address lines and control lines required for the communication protocol. In software requirements, he Vivado Design Suite Software will be needed to develop and deploy the hardwae circuit on the Nexys A7 FPGA board.

4.1.2 Software Life Cycle Model

The software life cycle model used here is the Agile development model. Agile methodology is based on collaborative decision making between requirements and solutions teams, and a cyclical, iterative progression of producing working software.

Work was done in regularly iterated cycles, or sprints, that usually lasted one to two weeks. Each sprint had a stack of new and old requirements known as the Backlog. Regular Scrum meetings (Daily meetings) took place between the development team members during a sprint. The Scrum meetings were performed to share the status of the work being completed on the backlog of the sprint and to identify potential issues to be added to the backlog of the next sprint.

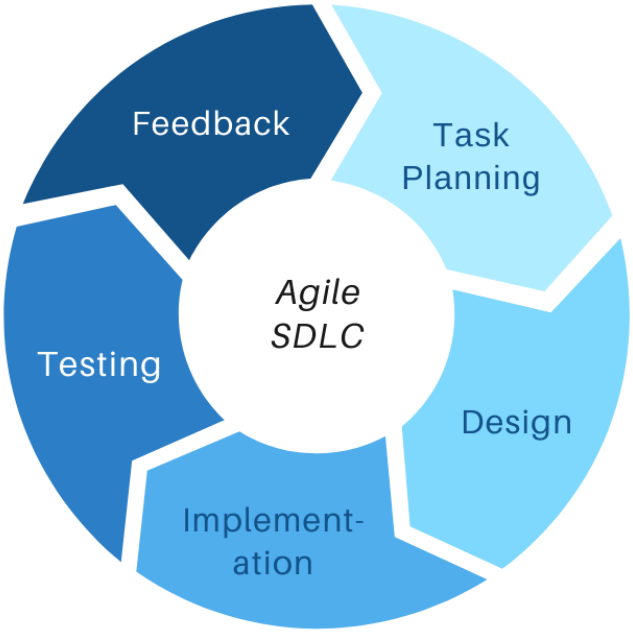

Figure 4.1: The Agile Software Development Life Cycle Model that was followed

4.1.3 Cost Analysis

The major cost of the project is of the Nexys A7-100T FPGA Kit. The entire project can be deployed on the FPGA kit along with a WIZ830MJ module, a passive communication PCB and an RJ45 Ethernet cable. As of 18 Nov, 2020, the Nexys A7-100T FPGA Kit costs $265. An additional cost of $52 would be incurred for the WIZ830MJ module, PCB manufacturing process and RJ45 Ethernet Cable. Thus, the total cost amounts to $317.

4.1.4 UML diagrams

4.1.4.1 Use Case Diagram

Figure 4.2 depicts how the system interacts with the external environment. In this diagram, there are two actors, the astronomical observatory and the user. The end user here may be another receiving end of the same observatory.

The observatory detects the light rays using optical sensors and passes the start and stop pulses to the TDC system. The TDC system is fabricated on the FPGA and uses a hybrid counter which is a combination of the coarse and fine counter to measure and output the final time interval. This result is written into the WIZ830MJ module so that it may be read by the end user.

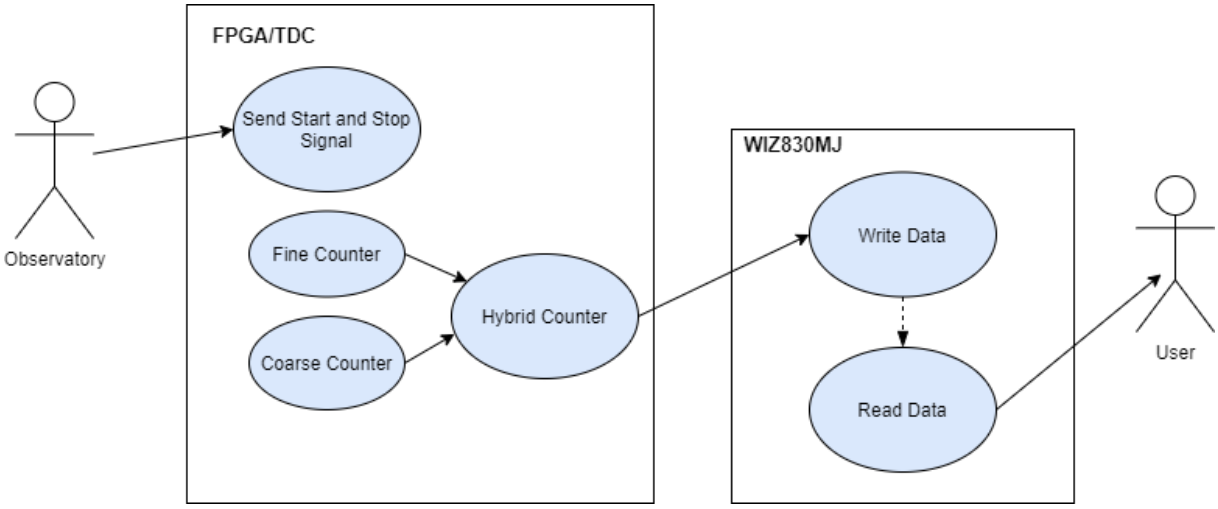

Figure 4.2: Use Case Diagram

4.1.4.2 Deployment Diagram

Figure 4.3 is the deployment diagram which visualises the physical hardware and software of the system. It depicts the execution architecture of the TDC system, including nodes for the optical sensor at the observatory, the FPGA, the WIZ830MJ module, and the wires/cables connecting them.

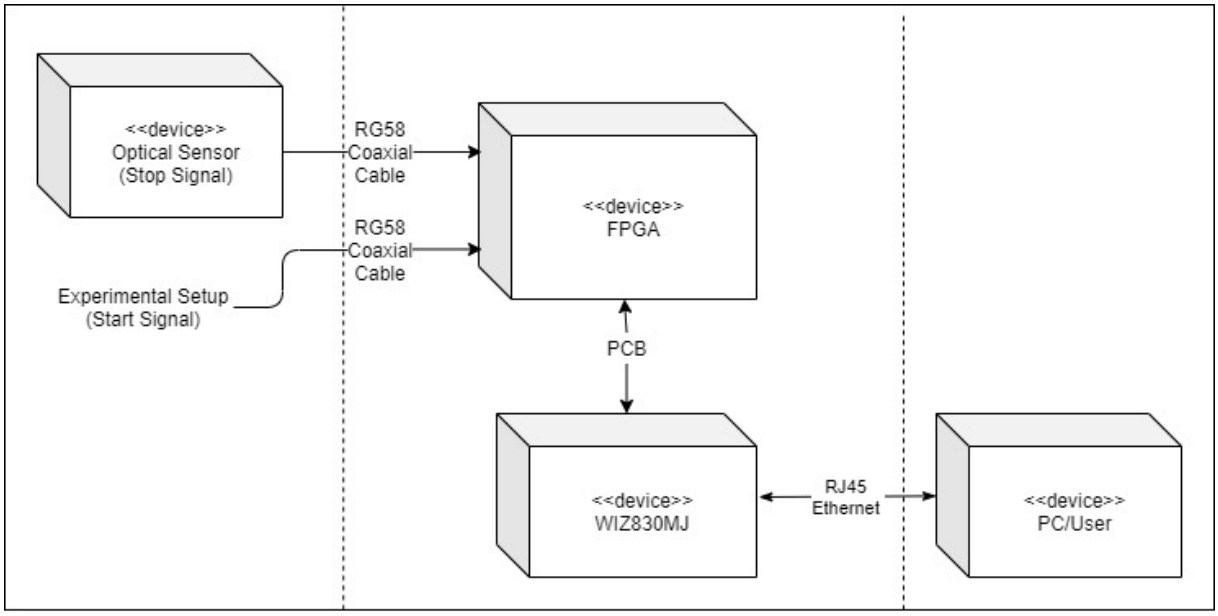

Figure 4.3: Deployment Diagram

4.1.4.3 Activity Diagram

Figure 4.4 shows the Activity Diagram with the workflow within the system. It shows the choices, decisions, behaviour and control flow from starting point to ending point of the execution.

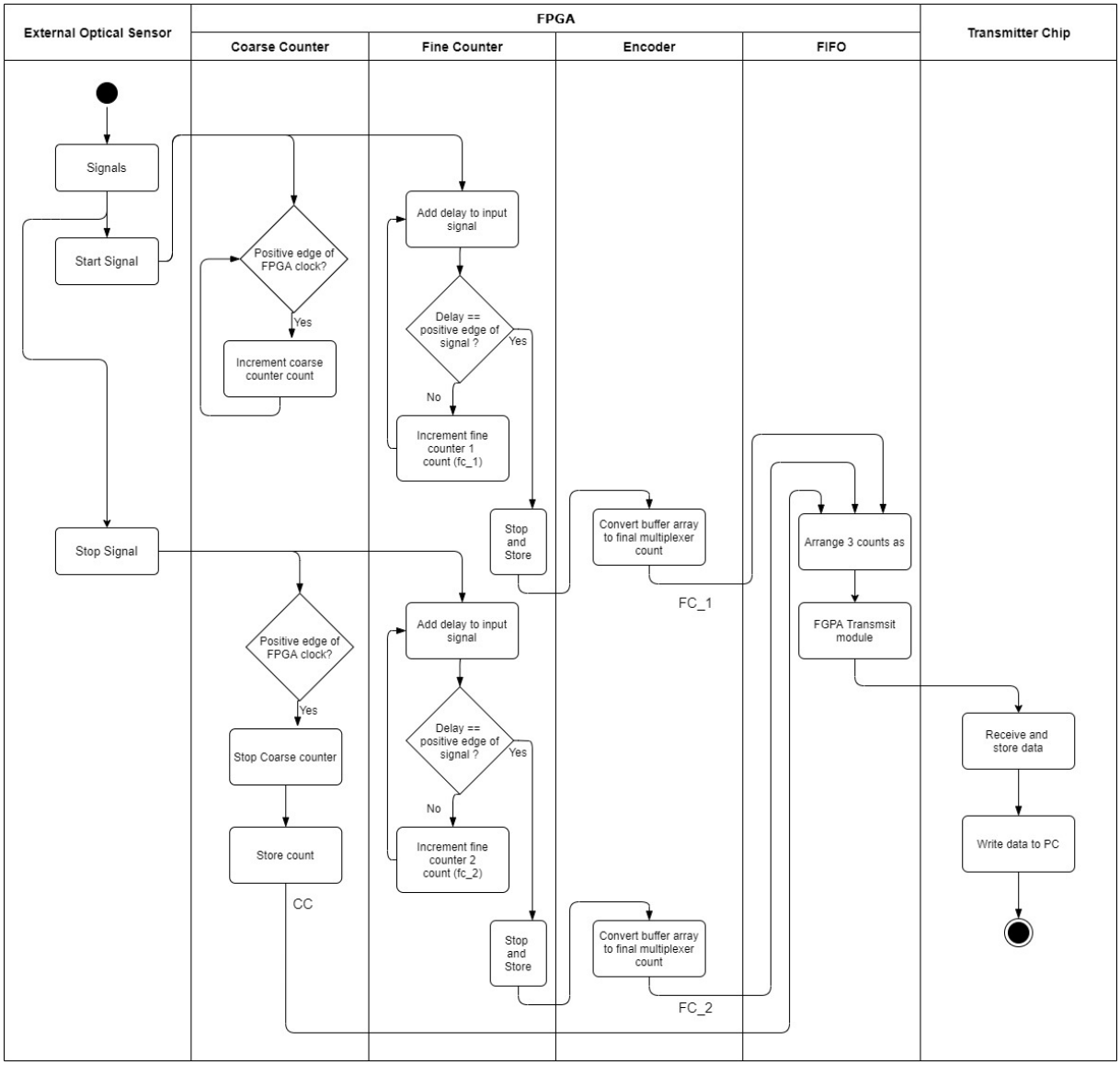

Figure 4.4: Activity Diagram

4.1.4.4 Sequence Diagram

Figure 4.5 shows the Sequence Diagram with object interactions arranged in time sequence. It is used represent the the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.

The start of the experiment triggers the start pulse and the detection of light by the optical sensors creates the stop pulse. These two serve as input

to the FPGA where the TDC counts the time interval between the two. The FPGA divides the time interval into 3 parts and generates 3 counts (Δt_c, Δt_1 and Δt_2). These are sent to the WIZ830MJ through which the end user (PC) receives it. The values are repeatedly generated and stored accordingly.

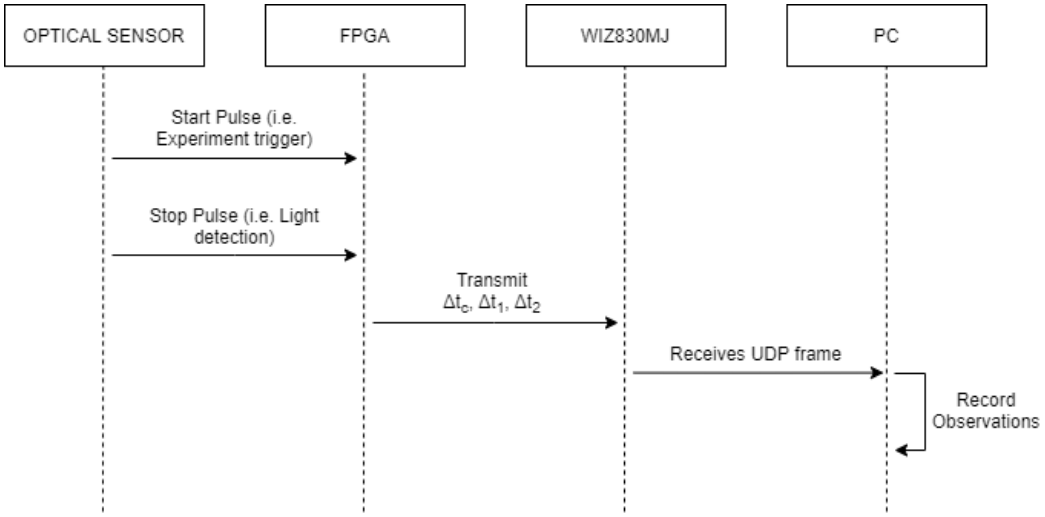

Figure 4.5: Sequence Diagram

4.1.5 Hardware and Software Requirements

The following are the mandatory minimum hardware and software requirements to run the project.

1. Hardware Requirements
    - Nexys A7-100T FPGA Kit: FPGA to write the bit file on
    - WIZ830MJ: Ethernet chip for communication between FPGA and User/PC,
    - PCB: To connect FPGA and WIZ830MJ,
    - RJ45 ethernet cable: To connect WIZ830MJ and User/PC,
2. Software Requirements
    - Vivado Design Suite: Required only to edit the project.
    - Python3: Required only to run the UART receiver code created.
    - Adept2: Optional. To be used in windows only when Vivado is not available.

4.2 System architecture

4.2.1 Block Diagram

The start and stop pulses are transmitted to FPGA from the observatory. The fine counter module 1 is given start and clock pulse as input whereas fine counter module 2 is given stop and clock pulse as input. The fine counter outputs the time intervals Δt_1 and Δt_2. These are the count of picoseconds from the edge of the start and stop pulse respectively to the next positive edge of clock. The start, stop and clock pulse are given as input to the coarse counter module. This outputs the Δt_c time interval which is the time between start and stop pulse in the order of 10ns. The coarse counter output and encoder outputs are passed to FIFO module which manages and arranges the data after its arrival. This final output is transmitted to the external Ethernet chip (WIZ830MJ) via Ethernet transmitter module from where the User/PC can read it. The final count is calculated as follows :

$$\Delta T = \Delta t_c + \Delta t_1 - \Delta t_2$$

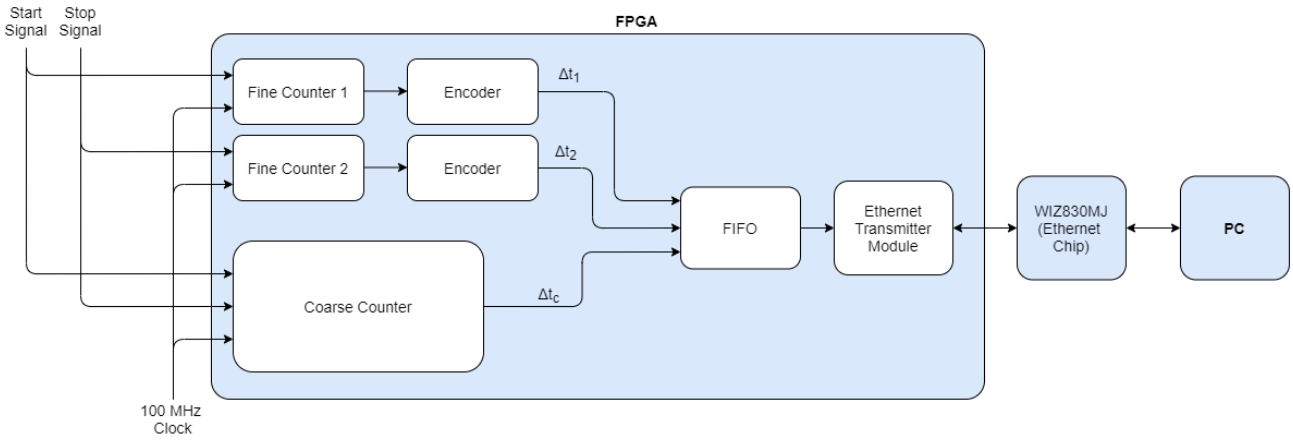

Figure 4.6: Block Diagram

Chapter 5

Result and Discussion

The coarse counter technique designed for this project was implemented as a gated counter. There are two inputs, the source signal and gate signal. The function of the gated counter is to count the number of source pulses occurring when the gate is open (HIGH). This is done by incriminating the count register if a positive edge of the source signal is encountered while the gate is open. The count is updated at each positive edge of the clock cycle.

The logic for the task was implemented using a finite state machine with 3 states (idle, counting and reset states). The circuit was described using Verilog Hardware Description Language in Vivado Design Suit software.

5.1 Screenshots of the System

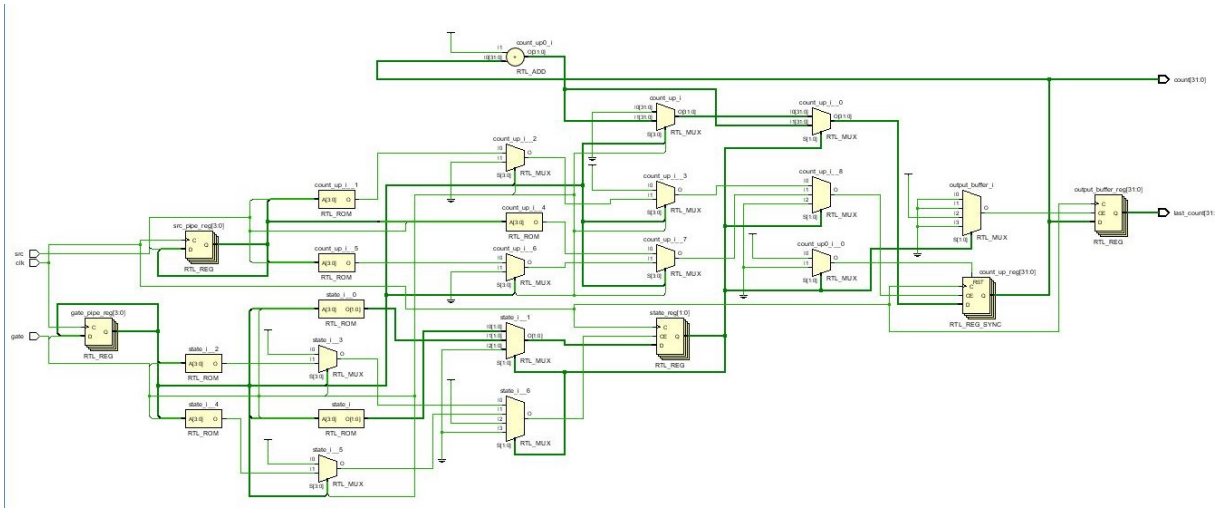

Figure 5.1: Detailed RTL Schematic of Gated Counter module

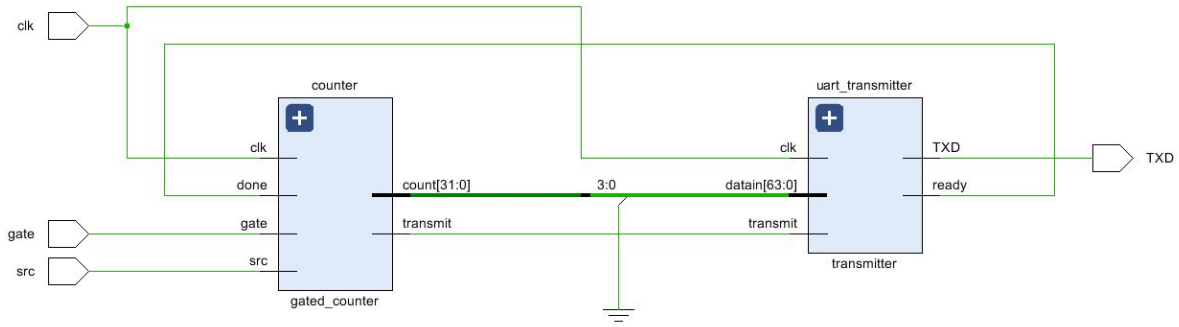

Figure 5.2: RTL Schematic of top module in Gated Counter

5.2 Sample Code

```
‘timescale 1ns / 1ps

module gated_counter(
    input wire clk,
    input wire gate,
    input wire src,
    //input wire done,
    output wire [31:0] count,
    output wire [31:0] last_count
    //output wire transmit
    );

reg [31:0] count_up;
reg [31:0] output_buffer;
reg [3:0] gate_pipe;
reg[1:0]  state;
reg[3:0] src_pipe;
reg transmit_reg;

localparam[1:0] idle = 2’b00;
localparam[1:0]  counting = 2’b01;
localparam[1:0]  reset = 2’b10;

initial begin
    state = idle;
    count_up = 32’d0;
    gate_pipe = 4’b0000;
    output_buffer=32’d0;
    src_pipe= 4’b0000;
    transmit_reg=1’b0;
end

always @ (posedge clk) begin

    gate_pipe[3:0] = { gate_pipe[2:0], gate };
    src_pipe[3:0] = {src_pipe[2:0], src };

    case(state)
```

```
39
40      idle: begin
41          transmit_reg<=0;
42          if(gate_pipe==4'b0000) begin
43              count_up<=32'd0;
44              state <= idle;
45          end
46          else if(gate_pipe==4'b0111) begin
47              case(src_pipe)
48                  4'b0110:    begin
49                      count_up<=count_up+1'b1;
50                  end
51                  4'b0011:    begin
52                      count_up<=count_up+1'b1;
53                  end
54                  4'b1011:    begin
55                      count_up<=count_up+1'b1;
56                  end
57                  4'b0111:    begin
58                      count_up<=count_up+1'b1;
59                  end
60              endcase
61              state <= counting;
62          end
63          end
64
65      counting:
66          begin
67          if(gate_pipe==4'b1111) begin
68              case(src_pipe)
69              4'b0011: begin
70                  count_up<=count_up+1'b1;
71              end
72              4'b1011: begin
73                  count_up<=count_up+1'b1;
74              end
75              endcase
76              state<=counting;
77          end
78          else if(gate_pipe==4'b1110) begin
79
80             if(src_pipe==4'b0110) begin
81                  count_up<=count_up+1'b1;
82             end
83
84             state <= reset;
85          end
86          end
87
88      reset:
89          begin
90          output_buffer <= count_up;
91          count_up <= 32'd0;
92          state <= idle;
93          end
94
95      endcase
96      $display("gate=%0b,%0b,%0b,%0b,src=%0b,%0b%0b,%0b,state=%0d,count=%0
    d",gate_pipe[3],gate_pipe[2],gate_pipe[1],gate_pipe[0],src_pipe[3],
```

```
    src_pipe[2],src_pipe[1],src_pipe[0],state,count_up);
end

assign count = count_up;
assign last_count = output_buffer;
assign transmit=transmit_reg;

endmodule
```

Listing 5.1: Gated Counter Circuit Design

```
`timescale 1ns / 1ps

module gated_counter_tb;

reg clk;
reg gate;
reg src;
wire TXD;

gated_counter Counter(clk,gate,src,count,last_count);

initial begin
    clk=1'b0;
    gate=1'b0;
    src=1'b0;

end

always begin
   #2 clk <= ~clk;
end

always begin
   #17 src <= ~src;
end

always begin
   #100 gate <= ~gate;
end

endmodule
```

Listing 5.2: Vivado Testbench code for simulating Gated Counter

Figure 5.3 shows the timing diagram obtained when the above gated counter was simulated with sample signals. As shown, the count increments at every positive edge of clock if both gate and src pulse are HIGH.

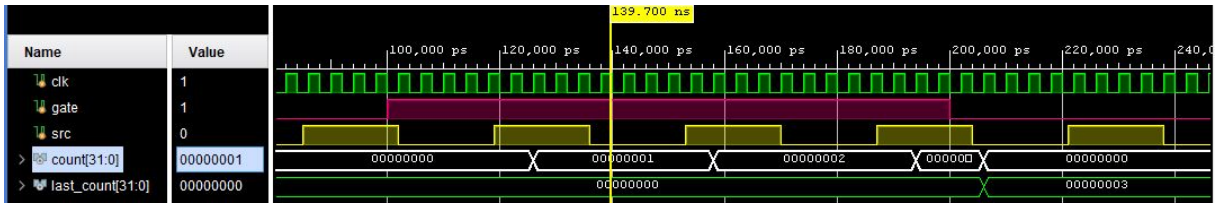

Figure 5.3: Timing Diagram Simulated for Gated Counter

Chapter 6

Conclusion & Future Scope

In this report, the study of different techniques and devices used for implementing coarse counters, fine counters and time-to-digital converters has been presented. The advantages of FPGA-based TDCs are highlighted and explained. The coarse counter designed is implemented and tested. The report includes the results for the same. A hybrid method of counting using Nutt interpolation method and is also proposed.

The implementation of the multi-channel TDC will be completed shortly. An efficient method of Ethernet communication has been implemented and will be connected to the multi-channel TDC. The main focus for the near future will be implementing the multiple channels for processing start and stop signals.

The future scope includes integration of the final system with the TIFR's observatory at Ooty for the GRAPES-3 experiment. A similar but advanced design can also be used in the Compact Muon Solenoid (CMS) experiment built on the Large Hadron Collider (LHC) at CERN in Switzerland and France.

References

[1] Y.-H. Kuo, "Determination of the angular distribution of cosmic rays at sea level," 02 2011.

[2] P. Lipari and S. Vernetto, "Diffuse galactic gamma-ray flux at very high energy," *Physical Review D*, vol. 98, Aug 2018.

[3] B. Hariharan, S. Ahmad, M. Chakraborty, A. Chandra, S. R. Dugad, S. K. Gupta, Y. Hayashi, H. Kojima, S. S. R. Inbanathan, P. Jagadeesan, and et al., "Energy sensitivity of the grapes-3 eas array for primary cosmic ray protons," *Experimental Astronomy*, vol. 50, p. 185–198, Sep 2020.

[4] Z. Soni, A. Patel, D. Panda, and A. B. Sarbadhikari, "Comparative study of delay line based time to digital converter using fpga," 2017.

[5] L. Y. Zhang M, Wang H, "A 7.4 ps fpga-based tdc with a 1024-unit measurement matrix," *Measurement*, vol. 17(4): 865, 04 2017.

[6] A. Balla, M. Beretta, P. Ciambrone, M. Gatta, F. Gonnella, L. Iafolla, M. Mascolo, R. Messi, D. Moricciani, and D. Riondino, "Low resource fpga-based time to digital converter," 06 2012.

[7] T. Sui, Z. Zhao, S. Xie, Q. Huang, J. Xu, and Q. Peng, "Pico-tdc: a novel fpga-based tdc with 2.2ps rms timing resolution," in *2017 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, pp. 1–3, 2017.

[8] C. Uğur, G. Korcyl, J. Michel, M. Penschuk, and M. Traxler, "264 channel tdc platform applying 65 channel high precision (7.2 psrms) fpga based tdcs," in *2013 IEEE Nordic-Mediterranean Workshop on Time-to-Digital Converters (NoMe TDC)*, pp. 1–5, 2013.

[9] E. Bayer, P. Zipf, and M. Traxler, "A multichannel high-resolution (<5 ps rms between two channels) time-to-digital converter (tdc) implemented in a field programmable gate array (fpga)," in *2011 IEEE Nuclear Science Symposium Conference Record*, pp. 876–879, 2011.

[10] D. Chaberski, "Time-to-digital-converter based on multiple-tapped-delay-line," *Measurement*, vol. 89, 03 2016.

[11] Y.-Q. Chen, L.-Y. Meng, and X.-G. Lin, "A coarse-fine time-to-digital converter," *ITM Web of Conferences*, vol. 11, p. 08006, 01 2017.

Acknowledgement

The success of a project like this involving high technical expertise is only possible when the whole team works together. We would like to take this opportunity to express our immense gratitude to those who have been instrumental in the successful completion of this project. We would like to show our appreciation to our guides for their tremendous support and help. Thank you to **Ms. Shweta Tripathi** for her support, encouragement and advice. We are grateful for the opportunity we received to work with Tata Institute of Fundamental Research, Mumbai under **Dr. Raghunandan A. Shukla**. We extremely thankful for his time, patience and constant guidance. This project would not have been possible without it.

We would also like to thank our project coordinator **Mrs. Rakhi Kalantri** for providing us with regular inputs about documentation and project timeline. A big thanks to our Head Of Department, **Dr. Lata Ragha**, for all the encouragement given to our team. We would also like to thank our principal, **Dr. S. M. Khot**, and our college, **Fr. C. Rodrigues Institute of Technology, Vashi**, for giving us the opportunity and the environment to learn and grow.

Project Group Members:

1. Aishwarya Anna Mathew, 101701

_____

2. Pallavi Banagar, 101706

_____

3. Roni Leonita, 101745

_____

Appendix A : Timeline Chart

	TASK TITLE	START DATE	DUE DATE
1	Meeting with Seniors	02-06-2020	13-06-2020
2	Study of Implementation	14-06-2020	09-07-2020
3	Installation	10-07-2020	16-07-2020
4	Study of Uart Communication	17-07-2020	23-07-2020
5	Verilog Assignments	24-07-2020	23-08-2020
6	Gated Counter	24-01-2020	06-09-2020
7	Gated Counter using FSM	07-09-2020	20-09-2020
8	BRAM & Vio	04-10-2020	11-10-2020
9	Ethernet Communication	12-10-2020	21-10-2020
10	Designed PCB	22-10-2020	30-10-2020
11	Integrating Gated Counter & Function Gen	28-11-2020	08-12-2020
12	Report	09-12-2020	12-12-2020

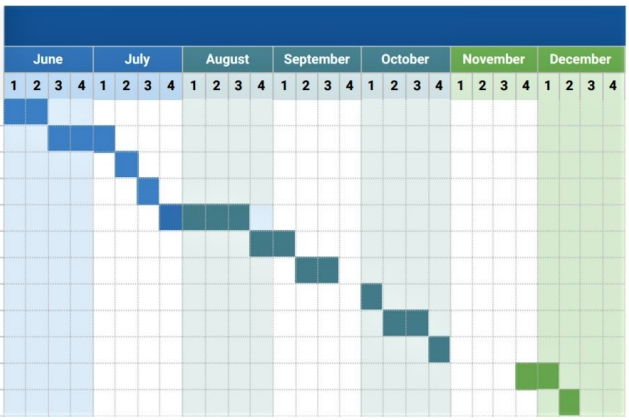

Figure 6.1: Gantt Chart for the project