

**DESIGN METHOD TO TRANSMIT AND  
RECEIVE SOURCE SYNCHRONOUS SIGNALS  
USING SOURCE ASYNCHRONOUS  
TRANSCEIVER CHANNELS**

**By**

**NATHAN RAMACHANDRAN**

**A dissertation submitted for partial fulfillment of the requirement  
for the degree of Master of Science**

**July 2013**

## **ACKNOWLEDGMENTS**

I will like to thank my supervisor, Dr. Wan Mohd Yusof Rahiman Wan Abdul Aziz whose help, stimulating suggestions and encouragement helped me in all my time of research for and writing this thesis. It was indeed an honor to work under him. I would also like to thank my sponsor, USM for giving me a chance to pursue my studies in this area.

I would like to thank the entire Electrical and Electronics Engineering Faculty for making such an enjoyable place to work.

Lastly, I would like to thank my parents, Mr and Mrs Ramachandran, wife, Yoges Mariarani, kids, Dhaanya Nathan and Abbinaya Nathan and my siblings for their prayers and full support throughout the year. Their constant affection and encouragement have helped me achieve my goals.

Nathan Ramachandran

July 2013

## TABLE OF CONTENTS

ACKNOWLEDGEMENTS	ii
TABLE OF CONTENTS	iii
LIST OF TABLES	iv
LIST OF FIGURES	v
LIST OF ABBREVIATIONS	vi
ABSTRACT	vi
ABSTRAK	ix
CHAPTER 1 INTRODUCTION	1
1.1 Overview	1
1.2 Research Motivation	2
1.3 Thesis Objectives	3
1.4 Requirements	4
1.5 Research Methodology	4
1.6 Thesis Outline	8
CHAPTER 2 LITERATURE REVIEW	9
2.1 Introduction	9
2.2 Components of Source Synchronous System	16
2.2.1 LVDS Transmitter	16
2.2.2 LVDS Receiver	17
2.3 Components of Source Asynchronous System	20
2.3.1 Source Asynchronous Transmitter	21
2.3.2 Source Asynchronous Receiver	21
2.4 Conclusions	24
CHAPTER 3 SOFTWARE AND HARDWARE DESIGN	26
3.1 Introduction	26
3.2 Design Specification	27
3.3 Design Methodology	28
3.4 Software Design	29
3.2 Software Development Tools	35
3.5.1 Model Sim v6.2b	35

3.5.2	Quartus 12.1 SP1	36
3.6	Hardware Setup	38
3.7	Methodology of Hardware Issue Debug	41
CHAPTER 4 RESULTS AND DISCUSSION		35
4.1	Overview	44
4.2	Results and Discussions	45
4.2.1	Prove of Concept	45
4.2.2	Measurement at 5 Gbps Across 5 Inches of Backplane	46
4.2.3	Measurement at 6 Gbps Across 10 Inches of Backplane	47
4.2.4	Performance Evaluation	49
4.2.5	Discussion on Advantages and Disadvantages of Source	54
Synchronous Systems		
4.3	Summary	55
CHAPTER 5 CONCLUSIONS AND FUTURE SCOPE		56
5.1	Conclusions	56
5.2	Future Scope of the Project	57
References		

## LIST OF TABLES

Table 4.1: Effect of Increasing Data Rate for Signals at Far End using 5 Inches of Backplane	50
Table 4.2: Effect of Increasing Transmission Backplane Length for Signal Transmitting at 5 Gbps	52
Table 4.3: Time Taken to Achieve Data and Strobe Synchronization	53

## LIST OF FIGURES

Figure 1.1: Design methodology flow chart	4
Figure 2.1: Block Diagram of LVDS Transmitter	9
Figure 2.2: Block Diagram of LVDS Receiver	10
Figure 2.3: Receiver Input, Output , Clock and Alignment Signaling	12
Figure 2.4: Preset Rollover Point Indicated by <i>rx_cda_max</i>	12
Figure 2.5: Source Asynchronous Transceiver Channel Components and Datapath	13
Figure 2.6: Word Aligner Configured in Bit Slip Mode	15
Figure 3.1: Block Diagram showing System Implementation	29
Figure 3.2: Logic Implementation in Data Channel and Strobe Channel for Synchronization and Word Boundary Alignment	32
Figure 3.3: Logic Implementation Monitoring Amount of Bit Slip in Data and Strobe Channels	34
Figure 3.4: Model Sim v6.2b Tool	36
Figure 3.5: Quartus II 12.1 SP1 Tool	37
Figure 3.6: Signal Tap Tool	37
Figure 3.7: Hardware Setup	38
Figure 3.8: Transceiver Instance Configured Using Quartus II Tool Phy IP in Megawizard	40
Figure 3.9: Hardware Issue Debug Flow	42
Figure 4.1: Cyclone V GT Eye Mask Specification	44
Figure 4.2: Signal Tap Tool Snapshot Showing the Prove of Concept	45

Figure 4.3: Near End Measurement Eye Diagrams after 5 Inches of Backplane at 5 Gbps on Source Synchronous vs Source Asynchronous Systems	46
Figure 4.4: Far End Measurement Eye Diagrams after 5 Inches of Backplane at 5 Gbps on Source Synchronous vs Source Asynchronous Systems	46
Figure 4.5: Signal Tap Snapshot of <i>error_counter</i> after 5 Inches of Backplane at 5 Gbps on Source Synchronous vs Source Asynchronous Systems	47
Figure 4.6: Near End Measurement Eye Diagrams after 10 Inches of Backplane at 6 Gbps on Source Synchronous vs Source Asynchronous Systems	48
Figure 4.7: Far End Measurement Eye Diagrams after 10 Inches of Backplane at 6 Gbps on Source Synchronous vs Source Asynchronous Systems	48
Figure 4.8: Signal Tap Snapshot of <i>error_counter</i> after 6 Inches of Backplane at 6 Gbps on Source Synchronous vs Source Asynchronous Systems	49
Figure 4.9: Graphical Representation of Data Rate vs Total Jitter for Signals at Far End using 5 Inches of Backplane	51
Figure 4.10: Far End Measurement Eye Diagrams after 5 Inches of Backplane at 7.25 Gbps on Source Synchronous vs Source Asynchronous Systems	51
Figure 4.11: Graphical Representation of Backplane Length vs Total Jitter for Signals Transmitting at 5 Gbps at Far End	52

## **LIST OF ABBREVIATIONS**

ASIC: Application Specific Integrated Circuit

CDR: Clock Domain Recovery

DC: Direct Current

DPA: Dynamic Phase Aligner

FIFO: First In First Out

FPGA: Field Programmable Gate Array

Gbps: Gigabit Per Seconds

IC: Integrated Circuit

ID: Identification

IP: Intellectual Property

IO: Input Output

LVDS: Low Voltage Differential Signaling

PCB: Printed Circuit Board

PCIe: Peripheral Component Interconnect Express

PCS: Physical Coding Sublayer

PD: Phase Detector

PFD: Phase Frequency Detector

PI: Phase Interpolator

PMA: Physical Medium Access

PPM: Parts Per Million

PRBS: Pseudo Random Bit Sequence

RTL: Register Timing Level

Tj: Total Jitter

VCO: Voltage Controlled Oscillator

vs: Versus

# **DESIGN METHOD TO TRANSMIT AND RECEIVE SOURCE SYNCHRONOUS SIGNALS USING SOURCE ASYNCHRONOUS TRANSCEIVER CHANNELS**

## **ABSTRACT**

Lower cost Field Programmable Gate Array (FPGA) devices offer limited data rate speed for source synchronous Low-Voltage Differential Signaling (LVDS) Input-Output (IO) interfaces but higher data rate speeds for source asynchronous transceivers channels. Cyclone V which is a low cost FPGA device supports LVDS IO channels for data rates up-till 1.25 Gigabit per second (Gbps) meanwhile the transceiver channels support data rates up-till 5 Gbps. In general, another known limitation of source synchronous system is the clock transmission path need to be as short as possible to eliminate high skew between data channel and clock channel. Hence, this research objective is to presents a solution to transmit and receive source synchronous signals at higher data rates using the available source asynchronous channels in the FPGA devices. The solution will also address the limitation of clock transmission path length. Overall, this will enable FPGA application developers to select lower cost devices to meet higher speed source synchronous data transmission requirements. The method used in this research is by transmitting the clock signal as a data signal. A series of digital logics are used to synchronize and align the recovered clock and data signal after the receiver for an error free transmission. The proposed solution is evaluated to support transmission of source synchronous signals up-till 6 Gbps without bit error using source asynchronous transceiver channels across a 10 inch backplane.

# **REKABENTUK UNTUK MENGHANTAR DAN MENERIMA ISYARAT SUMBER SEGERAK MENGGUNAKAN SALURAN SUMBER TIDAK SEGERAK**

## **ABSTRAK**

Field Programmable Gate Array (FPGA) yang berkos rendah menawarkan data dengan kelajuan terhad untuk saluran sumber segerak Low-Voltage Differential Signaling (LVDS) Input-Output (IO) tetapi kelajuan lebih tinggi untuk saluran sumber tidak segerak. Cyclone V adalah peranti berkos rendah yang menawarkan saluran LVDS IO yang menyokong kadar kelajuan 1.25 Gigabit sesaat (Gbps) tetapi saluran tidak segeraknya menyokong kadar kelajuan 5 Gbps. Secara umum, satu lagi had sumber sistem segerak adalah jarak penghantaran saluran jam perlu sependek yang mungkin untuk menghapuskan condong antara saluran data dan saluran jam. Maka, objektif kajian ini adalah membentangkan penyelesaian untuk menghantar dan menerima sumber isyarat segerak pada kelajuan yang lebih tinggi menggunakan saluran sumber tidak segerak yang terdapat dalam peranti FPGA. Penyelesaian yang dicadangkan juga akan membolehkan jarak penghantaran saluran jam yang lebih panjang digunakan. Secara keseluruhan, penyelesaian ini membolehkan pemaju aplikasi FPGA untuk memilih peranti yang berkos rendah untuk mencapai kelajuan sumber segerak yang lebih tinggi. Kaedah yang digunakan dalam kajian ini adalah menghantar isyarat jam sebagai isyarat data. Isyarat jam dan data diselaraskan oleh logik digital. Penyelesaian yang dicadangkan dalam kajian ini telah dinilai untuk menyokong penghantaran sumber isyarat segerak sehingga 6 Gbps tanpa ralat menggunakan sumber saluran tidak segerak untuk jarak penghantaran sejauh 10 inci.

# CHAPTER 1

## INTRODUCTION

### 1.1 Overview

Expansion in the telecommunications market and growth in internet use requires systems to move more data faster than ever. Hence, this creates new challenges to existing transmission systems such as high speed differential source synchronous systems. Source synchronous transmission refers to the technique of sourcing a clock along with the data. The clock is often referred to as a strobe. The traditional source synchronous interfaces in a digital system restrict the overall system performance and limit the printed circuit board (PCB) trace length. This method restricts system designers to achieve the high speed data signaling that today's market demands. As a solution, the system designers are turning to source synchronous system designs that demonstrate the high interconnect speed at distances of few meters. However, high speed data signaling in Gbps range shows other problems such as managing the skew between clock and data signals. The solution to this challenge is to use clock domain recovery (CDR) unit with dynamic phase alignment mechanism to eliminate the skew between data channels and clock channels. However, a CDR unit with dynamic phase alignment capabilities needs to incorporate a phase interpolator. CDR unit with a phase interpolator has a drawback of limited tracking frequency bandwidth. However, a high frequency bandwidth CDR unit without phase interpolator can be used if the recovered clock and strobe clock frequency phase is aligned externally.

High speed data signaling in Gbps is more popularly implemented with source asynchronous systems to overcome the challenge of transmitting clocks at high speed. At higher speeds in Gbps, the clock speed is in few hundred MHz range that it becomes a challenge to transmit it across a PCB trace. Hence, many designers started to move to source asynchronous systems. In source asynchronous systems, only the data is transmitted and the clock is later recovered using a CDR unit. However, source asynchronous systems have to incorporate some level of encoding and training sequence upon power up to ensure the clock can be recovered for sampling correctly at the receiver end. Most source asynchronous systems incorporate some level of physical sub-coding layer (PCS) to perform functions such as encoding, word alignment, rate matching and phase compensation before data is sampled with the recovered clock.

Hence, the transmission length and data rate of a source synchronous system can be further increased by incorporating some of the features offered in source asynchronous systems.

The FPGA devices nowadays are being offered with both source synchronous and source asynchronous transmission system. Hence, this research will take the features offered in the source asynchronous transmission systems to further boost the capability of source synchronous systems in terms of transmission speed and length.

## **1.2 Research Motivation**

Lower cost FPGA devices such as Cyclone IV GX, Cyclone V GT and Artix offer limited data rate speed for LVDS IO interfaces but higher data rate speeds for transceivers channels. Example, Cyclone V GT offers LVDS IO only up-till 1.25 Gbps but transceiver channels that can be configured to LVDS IO standard up-till 5

Gbps. Hence, this research is meant to be a solution to boost the device capability in terms of transmitting and receiving higher data rate of source synchronous signals using the available source asynchronous channels available in the FPGA devices. Overall, this will actually enable FPGA users to select lower cost devices to meet their high speed data transmission requirements.

### **1.3.1 Thesis Objectives**

The general aim of this project is to develop a solution that addresses the following objectives:

1. To transmit and receive source synchronous data with zero bit error using low cost FPGA device at 5 Gbps.
2. To transmit the clock over a minimum length of 5 inches.
3. To leverage features offered in source asynchronous transceiver channels to avoid high logic utilization in FPGA fabric.
4. To customize the design to be scalable for implementation in any FPGA or Application Specific Integrated Circuit (ASIC) devices.

The target data rate 5 Gbps is selected because Cyclone V GT device transceivers can operate up-till this speed. The clock transmission path length of 5 inches is selected because source synchronous protocols such as Peripheral Component Interconnect Express (PCIe) at 5 Gbps requires minimum of 5 inches backplane drive capability.

## **1.4 Requirements**

These are the requirements of this thesis.

1. No bit errors should be observed for half an hour of transmission with stressed data pattern, Pseudo Random Bit Sequence (PRBS) pattern  $2^{23}$  at data rate 5 Gbps.
2. Word alignment pattern “5555 5555 5555 000FF” is send upon power up repeatedly for link training. The time taken to repeat the link training pattern should be kept as short as possible around 1us. This is to avoid link up time that is higher than 10us. Word alignment can only be achieved after receiving the training pattern continuously for more than 20 times.
3. LVDS IO standard should be used for both transmitter and receiver.

## **1.5 Research Methodology**

The development of this project has been separated into four parts. The first part focuses on project planning. Planning involves timeline creation, breaking down task to the smallest task and selecting devices and tools based on availability search.

The second part is creating a methodology to implement the project. This is done by in-depth research on components, limitation, advantages and applications of source synchronous and asynchronous systems. The third part is creation of RTL code to implement the methodology. A test bench will be created to validate the functionality of the register transfer level (RTL) design code in simulation. The last part is to transfer the code to hardware and validate if the project requirements are met.

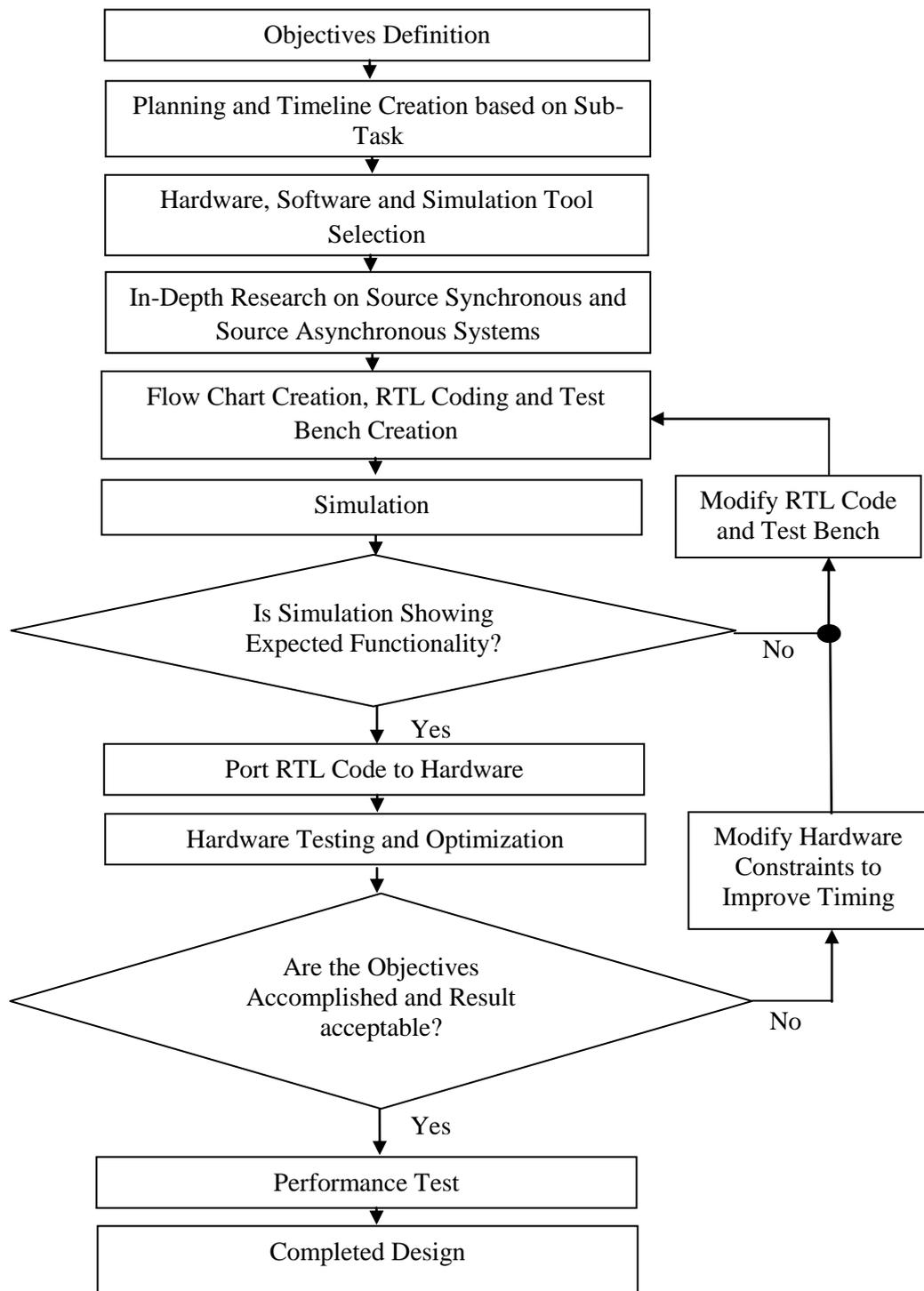


Figure 1.1: Design methodology flow chart

The design methodology is described by the flow chart in Figure 1.1. This research is started off by defining the objectives of the research clearly. Once the end

goal was clear, the planning was done. The project implementation was divided into task and subtask. The details on task and subtask were used to plan a realistic timeline.

The next step was hardware selection. This part was critical because hardware availability determines whether an existing FPGA development kit could be leveraged or need to build a new PCB. Building a new PCB board will be more costly and require more effort and time. However, if an existing FPGA board is used, the hardware must be able to meet all the research requirements from start till end. It will be a waste of time if the hardware is found to be unable to meet the research requirements at a later stage.

After selecting the hardware, the proper software for RTL code compilation, simulation and hardware programming file configuration is determined. The software licensing and availability is also considered as each software tool is required uniquely at different stages of the research.

The next step was to go into in-depth research of source synchronous and source asynchronous systems. This was critical to gather as many ideas as possible on how to achieve the research objective. New knowledge on both source synchronous and source asynchronous systems in terms of implementation and application was gained in this stage. This step also ensures this research does not repeat someone else work and its unique.

Moving on, this research enters its most critical step which determines the design methodology. The flow chart defining the design was created and updated multiple rounds. The flow chart creation was done keeping into consideration the design will be implemented using Verilog code. Hence, the flow chart is simple enough to translate to state diagrams. After this, RTL code was written in Verilog

based on the flow chart. The next step was the test bench creation. The test bench was written to check as many paths and loopbacks in the flow chart.

Multiple simulations were done in the next step to verify the health of the RTL code. The simulations also served to perform functional verifications on the design. The simulation results were checked to see if it displayed the expected functionality. For every case that did not meet the functional expectation, the part of the RTL code describing the case was identified and modified. Functional simulation was continued till the entire design met the expected functionality.

The next step was to port the verified RTL code into hardware. A software tool was used to convert the synthesized netlist of the RTL code into a hardware programming file format. This step also involves programming the hardware configuration, timing constraints and pin assignment into the software tool.

Finally, the hardware is setup for evaluation. The hardware is programmed with the programming file and the design is tested in hardware. Modification on the hardware system setup and test environment is made to validate the design in hardware. The design is now validated in hardware to check if the research objectives are met and the results are acceptable. The part of the design that was failing in hardware was identified and hardware constraint modifications were done to improve the timing. After that, the part of the RTL code related to the hardware failure was updated. Hardware testing is only performed on the updated RTL after software simulation is rerun again and shows no issue. This step was continued till the design met all the design objectives and the results were acceptable. Then, the design performance was tested with higher transmission speed and longer transmission path.

## **1.6 Thesis Outline**

The thesis report has been organized into five chapters as follows. The literature review is presented in Chapter 2. The description on key components of source synchronous and source asynchronous systems are included. Chapter 3 describes the specification and operation of the key components used in the project. Chapter 3 also demonstrates the detailed aspects of hardware implementation, software development and implementation for the proposed digital logic system. Chapter 4 presents the results and discussion of the testing that have been conducted on the system. The final chapter provides the conclusions and suggestions for future work that can be realized for this project.

## CHAPTER 2

### LITERATURE REVIEW

#### 2.1.1 Introduction

The mobile communication and data communication industry has been continuously growing in the last few years. This brings up a demand for source synchronous systems to transmit more data at a faster rate. In recent years source synchronous signals have been transmitted faster by incorporating features such as differential signaling, dynamic phase alignment, clock domain recovery and clock jitter elimination [Kurd & Tierno, 2011; Loh & Neyestanak, 2008]. In the present paper it is hypothesized that source synchronous signals can be transmitted faster and further if a system incorporates features to manage the skew between clock and data signal. The following four literature reviews attempt to demonstrate and support the hypothesis. In a research article by Shijie Hu (2012), two specific points were stressed related to source synchronous system clocking at 10 Gbps. The research covers the advantages and disadvantages of using Phase Interpolator (PI) based CDR. A CDR is used to recover the clock from the received data. This recovered clock will be used to sample the data after the receiver. The CDR tracks the data by using a clock with same phase as the sampling clock on the transmitter side. Hence, an important requirement for a CDR is its input reference clock frequency must be same as the Phase Locked Loop (PLL) clock that sampled the transmitted data. Most CDR designs allow a certain Parts Per Million (PPM) threshold difference between the CDR reference clock and the transmitter's sampling clock. If the PPM threshold is within the allowed range, the CDR unit will be able to track the data correctly.

However, the clock phase the CDR tracks the data will be different from the transmitter's PLL clock. The article focuses on studying how critical is the phase difference at higher data rates such as 10 Gbps. The article compares the phase of the expected clock against the measured recovered clock. The research uses two 65nm CMOS process CDR that is able to track signals between 5 Gbps to 10 Gbps. The first is an analogue CDR and the second is a PI based CDR. The measurements show that at 5 Gbps, the phase difference is exceeding half the phase of the original signal for the analogue CDR. However, the PI based CDR is able to maintain phase difference within 10% of the original clock signal up-till 10 Gbps by changing the PI settings. This results show that an analogue CDR cannot be used to maintain the phase difference between the original transmitted clock and CDR recovered clock at high speeds. The analogue CDR is designed using a Phase Detector (PD) and Phase Feedback Detector (PFD). The PD tracks the data using a Voltage Controller Oscillator (VCO). The VCO continuously oscillates and generates different clock phases based on the reference clock input. This causes the PD to attempt to track the data with a series of different clock phases generated by the VCO. When the PD determines a certain clock phase that can track the data, it will fix the clock phase and track the data with that setting. However, the data will be continuously monitored by the PFD to check if the clock phase used is correct. A slight change in the Process, Voltage or Temperature (PVT) conditions actually causes the PFD to trigger PD to re-track the data with a new phase clock. The PFD acts as a feedback path to ensure the data is continuously tracked with the correct recovered clock phase. This improves the analogue CDR in terms of jitter tolerance and bandwidth. The continuously oscillating VCO also causes the analogue CDR to be sensitive towards conditions such as long idling of logic zero or one. Hence, this causes the

analogue CDR to have very high phase difference with the transmitted data's sampling clock. This is an issue for source synchronous signals but is not a concern for source asynchronous systems. Hence, the point to note from this research is, source synchronous systems will be better suited to use a CDR with PI compared to an analogue CDR. If an analogue CDR is required for source synchronous system, the phase difference need to be addressed externally outside the receiver. Clock phase difference between the transmitted clock source and recovered clock need to be controlled only when the skew between data and clock channel is high. The clock phase contributed by the blocks within the receiver and transmitter is minimum compared to phase difference contributed by the data and clock channel skew. The PI based CDR is made up of PI and PD. The PI will divide down the input reference clock to multiple clock phases. Similar to the analogue CDR, the PD will determine the best clock phase that is able to track the data correctly. However, once the clock phase is determined, the PD will continuously use the same clock phase until the next reset cycle. There is no feedback mechanism for PVT conditions in PI based CDR. However, this translates to the CDR being able to track signal with less than 10% clock phase difference between source and recovered clock up-till 10 Gbps. A point to note is when the impact of PVT condition increases exponentially with data rate variation. CDR operation at higher speed translates to lower jitter and noise tolerance at higher speeds with PVT variations. Hence, this research provides a specification for skew control between data and clock channel for data rates beyond 5 Gbps when using CDR unit with PI. Hence, research article by Shijie Hu (2012) points out that source synchronous system requires the clock phase of the source and recovered clock to be within certain percentage. If analogue based CDR is used, it will have the advantage of better clock jitter or noise condition tolerance. However, the power

consumption will be higher as the analogue CDR has a feedback loop and an external phase compensation mechanism needed to be implemented. The CDR with PI however is more suited for source synchronous systems. The advantage is the power consumption of the CDR is lower. However, the drawbacks are more controlled PVT conditions need to be met and skew between data and clock channel need to be taken care to be within the specified range. Hence, this research proves the hypothesis that source synchronous signals can be driven faster by managing the skew between clock and data channel.

Research article by Agarwal (2008) highlights that as data rates increase, successful data recovery in a jittery environment requires precise positioning of the sampling clock. Receivers need to perform skew compensation between data and clock channel for every IO pin while preserving the correlation in the jitter between the transmitted clock and data. Source synchronous receiver channels are mentioned to widely use multi-phase clock generators to drive phase interpolators. Multiple clock phases are also required when interleaved samplers are employed to easily accommodate high off-chip data rates. The advantage of using multiple clock phases to drive phase interpolators are more clock phase divisions can be obtained. This will enable better jitter tolerance because of more precise position of the sampling clock. However, this will cause the tracking time taken by data to track the correct clock frequency phase to increase. Having a better system jitter tolerance is shown in the research to increase the transmitted path length. Hence, this paper also further proves the hypothesis at the beginning of Chapter 2.

Research article by Paul Teehan (2009) shows the Dynamic Phase Alignment (DPA) feature is able to increase source synchronous system efficiencies in terms of transmission bandwidth. The research highlights that new emerging high speed

interface standards such as Rapid IO, Small Form Factor (SFF) and Gigabit Ethernet requires source synchronous systems with data rates exceeding 5 Gbps and auto negotiation support to lower data rates. Hence, this creates a requirement for source synchronous systems to not only go faster but run slower when required. In another word, the transmission bandwidth is also becoming a concern for source synchronous signals. DPA is a feature commonly found in receiver systems to help with phase alignment of the clock and data channel to clean up the clock skew at receiver end. A typical problem faced with high-speed source-synchronous systems is when clock or data signal transitions occur at different times with respect to each other. When this happens, the receiver does not sample the data at the correct time, causing system bit errors. This problem is mainly contributed by inherited skew on the clock transferred out from the transmitter devices. The research shows the skew to be as high as  $0.2UI$  or almost 20 percent of the signal at 5 Gbps for LVDS IO pins of 40nm CMOS process. This results in inaccurate data transmission from one point to another and interrupted communication between components within the system. The DPA corrects the clock skew in reference to the data with using an individual PLL. Hence, this translates to better power consumption. PLL units are generally high power consumption units. A typical DPA contains a PI, dynamic phase selector, synchronizer and a data re-aligner unit. The clock from the strobe is used by the PI to generate multiple phases. The dynamic phase selector will select the clock phase that can track the data and send the data over to the synchronizer. The clock supplied to the synchronizer is the PI generated clock phase that matches the data. The synchronizer will monitor the control signals in the data for synchronization code. If synchronization codes cannot be received, the synchronizer will request the PI to reselect a new clock phase that matches the data. This block provides some sort of

feedback mechanism for the PI block to compensate for PVT variations. Finally the data and clock is transmitted over to the data re-aligner unit where the data word boundary will be aligned to the clock. This is done by waiting for a known word alignment pattern. Hence, to use a DPA, the source synchronous system needs to transmit additional control characters which are the data synchronization code and word alignment pattern. The advantage of using DPA is higher transmission speeds and bandwidth can be achieved with no skew control requirements between data and clock channels. The DPA also is not sensitive to PVT condition variations and does not require additional circuitry for phase control between clock and data channel. However, the biggest disadvantage of a DPA is reduced payload with additional control characters. The increased control characters in a source synchronous system will translate to smaller effective bandwidth transfer and complex lower level data management scheme. A key takeaway from this research article is higher transmission speed is obtainable when the skew between data and clock channel is managed. This further proves the hypothesis in the beginning of Chapter 2 again.

Research article by Muhammad Elraba (2006) shows how digital logics are utilized to address concern on bad clock quality at receiver. The clock signal deterioration due to long transmission path and skew between data and clock channel could cause bit errors. This issue could lead to timing failure on the receiver circuitry due to non-synchronous data clock at receiver side. Hence, this research provides a method without using an analogue PLL to clean up the clock jitter. Analogue PLL's have very large circuits and consume high power. The digital logic solution proposed is a fully digital, hence its smaller in size with much lower power consumption. The solution proposed is to implement a digital clock re-timer. The digital clock re-timer comprises of a Phase Capturing and Clock Muxing (PCCM) circuit, matched delay

line circuits, a fixed delay circuit, a matched delay circuit and double edge detector. The data is sent through a double edge detector whereby the data will be sampled across both the clock edges. Double sampling of data helps to address issue of missing data cycle. The PCCM circuit will generate multiple phase of clocks from the strobe signal. The multipliers inside the PCCM circuit will further generate even more precise clock phases. All the clock phases generated are transmitted through a matched delay line before checking which clock tracks the data correctly. This ensures more precise positioning of the sampling clock. The clock is tracked against the data at both edges. Clock tracked at both data edges will incorporate an averaging method to determine the correct logic for that sampling period. Hence, this allows more margin for setup and hold time to meet the timing requirement which translates to better reduced clock jitter. When the best clock phase is determined, the data is transmitted through a matched delay to the subsequent circuitry. This is to let the clock be ready before data arrives in the subsequent circuitry. This method has been proved to clean clock signals over a 10 inch PCB board for data rate up-till 5 Gbps without using an analogue PLL for clock cleanup. This research proves the hypothesis in the beginning of Chapter 2 again.

All of these results combined confirm the hypothesis that managing the data and clock channel skew is the key to achieve faster and longer transmission in a source synchronous system. Several methods have been introduced from previous research. For this research, the FPGA device Cyclone V GT is used. Hence, the features offered by Cyclone V GT LVDS IO and transceivers will be discussed in the next part of the chapter.